MA 5124 Financial Time Series Analysis & Forecasting

Dr. Priyanga D. Talagala

2021-03-28

Contents

Course Syllabus

Module Code: MA 5124

Title: Financial Time Series Analysis & Forecasting

Credits: 4

Pre-requiites

None

Learning Objectives

- The purpose of this course is to provide students with introductory tools for the time series analysis of financial time series.
- Analyze of data series based on stochastic and non stochastic models

Learning Outcomes

- On successful completion of this course, students will be able to provide more than an introductory treatment of the topics.
- Students are encouraged to pursue further study in this area if they find that the topics covered in this course.

Outline Syllabus

- Definition and examples of time series
- back-shift and differencing-operators, - strong and weak stationarity, definition of ACF, PACF.

- Definitions and properties of the $MA(q), MA(\infty), AR(p), AR(\infty)$ and $ARMA(p, q)$,in particualr their acf's
- causal stationarity of AR
- invertibility of MA models and causal stationarity and invertibility of ARMA; - concept of spectral density function and its applications
- definition and properties of integrated $ARIMA(p, d, q)$ processes
- definition and properties of random walks with or without drift.
- Model selection following the AIC and BIC
- brief introduction to linear prediction and calculation of forecasting intervals for normal ARMA models
- point and interval forecasts for normal random walks with or without drift.
- Definition and properties of the VAR (vector autoregressive) model, arrange a univariate time series as a multivariate Markov model.
- Nonlinear properties of financial time series
- definition and properties of the well known ARCH, GARCH etc.
- Cointegration in Single Equations, Modeling and Forecasting Financial Time Series.

Method of Assessment

- Assignment 30%
- End-semester examination 70%

Lecturer

Dr. Priyanga D. Talagala

Schedule

Lectures:

- Sunday [9.00am -12.00 noon]

Copyright Notice

My lectures and course materials, including presentations, tests, exams, outlines, and similar materials, are protected by copyright. I am the exclusive owner of copyright in those materials I create. I encourage you to take notes and make copies of course materials for your own educational use. However, you may not,

Chapter 1

Intordution to Time Series Forecasting

Chapter 2

ARIMA models

- **AR**: autoregressive (lagged observations as inputs)
- **I**: integrated (differencing to make series stationary)
- **MA**: moving average (lagged errors as inputs)

An ARIMA model is rarely interpretable in terms of visible data structures like trend and seasonality. But it can capture a huge range of time series patterns.

2.1 Stationarity and differencing

2.1.1 Stationarity

Definition

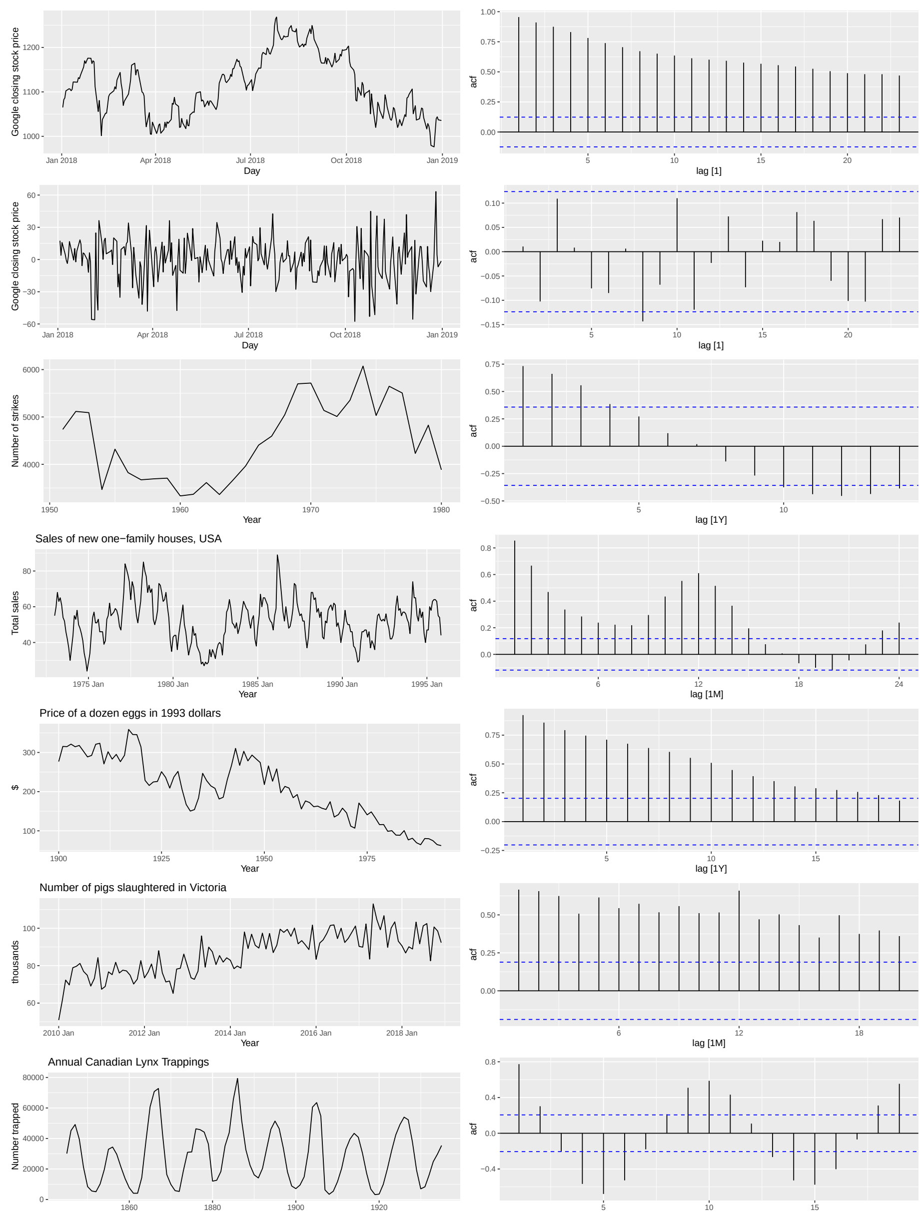
If $\{y_t\}$ is a stationary time series, then for all s, the distribution of $(y_t, \dots, y_{t+s})$ does not depend on t.

A **stationary series** is:

- roughly horizontal
- constant variance
- no patterns predictable in the long-term
- Transformations help to stabilize the variance.
- For ARIMA modelling, we also need to stabilize the mean.

Identifying non-stationary series

- time plot.
- The ACF of stationary data drops to zero relatively quickly
- The ACF of non-stationary data decreases slowly.
- For non-stationary data, the value of r1 is often large and positive.



- A time series, $\{Y_t, t = 0, \pm 1, ...\}$ is said to be **strict stationary**, if $(Y_1, ..., Y_n)$ and $(Y_{1+h}, ..., Y_{n+h})$ have the same joint distribution for all integers h and $n > 0$.

2.1.1.1 Weak Stationarity

Definition: Covariance function (in (Brockwell et al., 2016), p. 15; the notations have been changed for consistency within this note)

Let $\{Y_t\}$ be a time series with $E(Y_t^2) < \infty$. The **mean function** of $\{Y_t\}$ is

$$\mu_Y(t) = E(Y_t)$$

The **covariance function** of $\{Y_t\}$ is

$$\gamma_Y(r, s) = Cov(Y_r, Y_s) = E[(Y_r - \mu_Y(r))(Y_s - \mu_Y(s))]$$

for all integers r and s.

Definition: Weakly stationary (in (Brockwell et al., 2016), p. 15; the notations have been changed for consistency within this note)

$\{Y_t\}$ is **weakly stationary** if

1. $\mu_Y(t)$ is independent of t,

and

2. $\gamma_Y(t + h, t)$ is independent of t for each h.

- Unless specifically indicate otherwise, whenever we use the term *stationary* we shall mean *weakly stationary.*

2.1.2 Differencing

- Differencing helps to **stabilize the mean**.
- The differenced series is the *change* between each observation in the original series: $y'_t = y_t - y_{t-1}$.
- The differenced series will have only $T - 1$ values since it is not possible to calculate a difference y'_1 for the first observation.

2.1.2.1 Second-order differencing

Occasionally the differenced data will not appear stationary and it may be necessary to difference the data a second time:

$$\begin{aligned} y''_t &= y'_t - y'_{t-1} \\ &= (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) \\ &= y_t - 2y_{t-1} + y_{t-2}. \end{aligned}$$

- y''_t will have $T-2$ values.
- In practice, it is almost never necessary to go beyond second-order differences.

2.1.2.2 Seasonal differencing

A seasonal difference is the difference between an observation and the corresponding observation from the previous year.

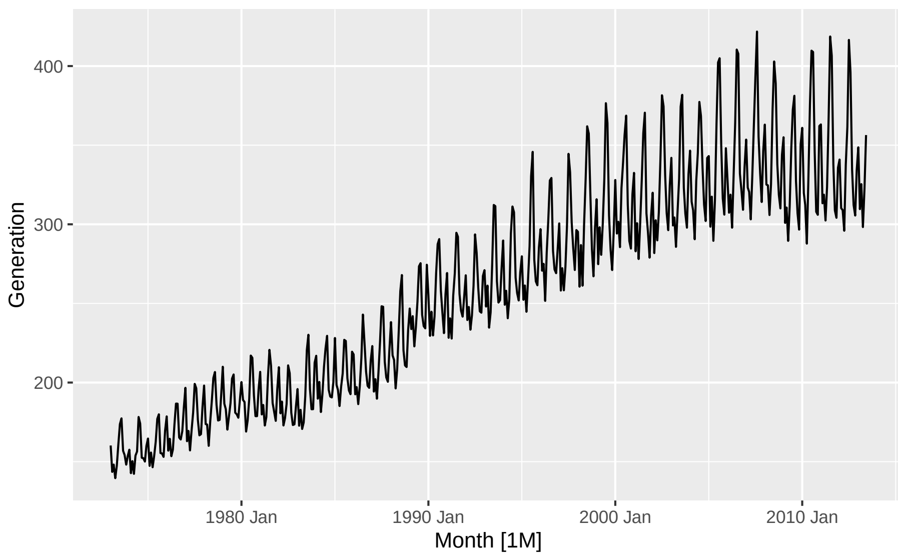
$$y'_t = y_t - y_{t-m}$$

where $m =$ number of seasons.

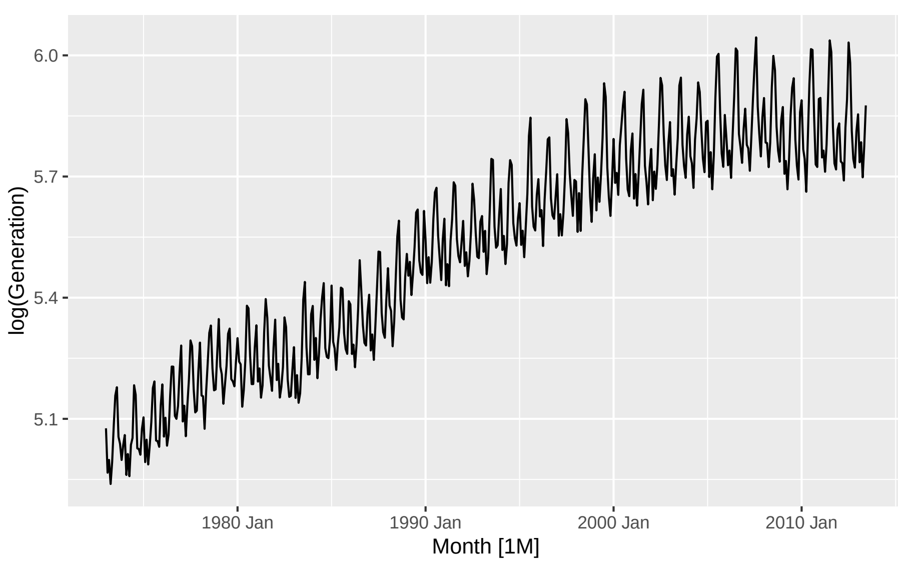
- For monthly data $m = 12$.
- For quarterly data $m = 4$.

Example : Electricity production

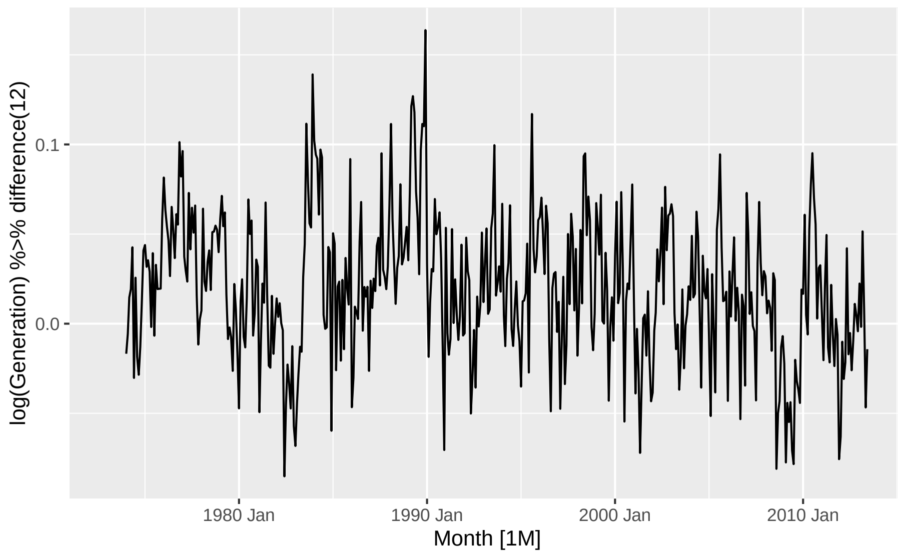
```
usmelec %>% autoplot(Generation)
```



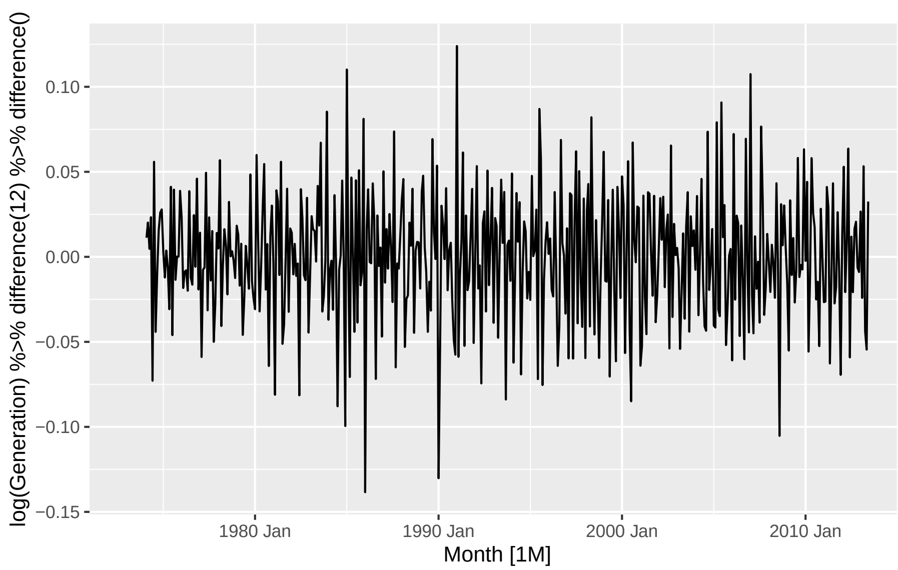

```
usmelec %>% autoplot(log(Generation))
```



```
usmelec %>% autoplot(log(Generation) %>% difference(12))
```




```
usmelec %>% autoplot(log(Generation) %>% difference(12) %>% difference())
```



- Seasonally differenced series is closer to being stationary.
- Remaining non-stationarity can be removed with further first difference.

If $y'_t = y_t - y_{t-12}$ denotes seasonally differenced series, then twice-differenced series is

$$\begin{aligned} y^*_t &= y'_t - y'_{t-1} \\ &= (y_t - y_{t-12}) - (y_{t-1} - y_{t-13}) \\ &= y_t - y_{t-1} - y_{t-12} + y_{t-13}. \end{aligned}$$

When both seasonal and first differences are applied ...

- it makes no difference which is done first—the result will be the same.
- If seasonality is strong, we recommend that seasonal differencing be done first because sometimes the resulting series will be stationary and there will be no need for further first difference.

It is important that if differencing is used, the differences are interpretable.

2.1.2.3 Interpretation of differencing

- first differences are the change between one observation and the next;
- seasonal differences are the change between one year to the next.

But taking lag 3 differences for yearly data, for example, results in a model which cannot be sensibly interpreted.

2.1.3 Backshift notation

A very useful notational device is the backward shift operator, B, which is used as follows:

$$By_t = y_{t-1}$$

In other words,

- B, operating on y_t, has the effect of **shifting the data back one period**.
- Two applications of B to y_t **shifts the data back two periods**:

$$B(By_t) = B^2 y_t = y_{t-2}$$

- For monthly data, if we wish to shift attention to "the same month last year", then B^{12} is used, and the notation is

$$B^{12} y_t = y_{t-12}$$

.

- The backward shift operator is convenient for describing the process of *differencing*.
- A first difference can be written as

$$y'_t = y_t - y_{t-1} = y_t - By_t = (1-B)y_t$$

- Note that a first difference is represented by $(1-B)$.
- Similarly, if second-order differences (i.e., first differences of first differences) have to be computed, then:

$$y''_t = y_t - 2y_{t-1} + y_{t-2} = (1-B)^2 y_t$$

- Second-order difference is denoted $(1-B)^2$.
- *Second-order difference* is not the same as a *second difference*, which would be denoted $1-B^2$;
- In general, a dth-order difference can be written as

$$(1-B)^d y_t$$

* A seasonal difference followed by a first difference can be written as

$$(1-B)(1-B^m)y_t$$

- The "backshift" notation is convenient because the terms can be multiplied together to see the combined effect.

$$\begin{aligned}(1-B)(1-B^m)y_t &= (1-B-B^m+B^{m+1})y_t \\ &= y_t - y_{t-1} - y_{t-m} + y_{t-m-1}.\end{aligned}$$

- For monthly data, $m = 12$ and we obtain the same result as earlier.

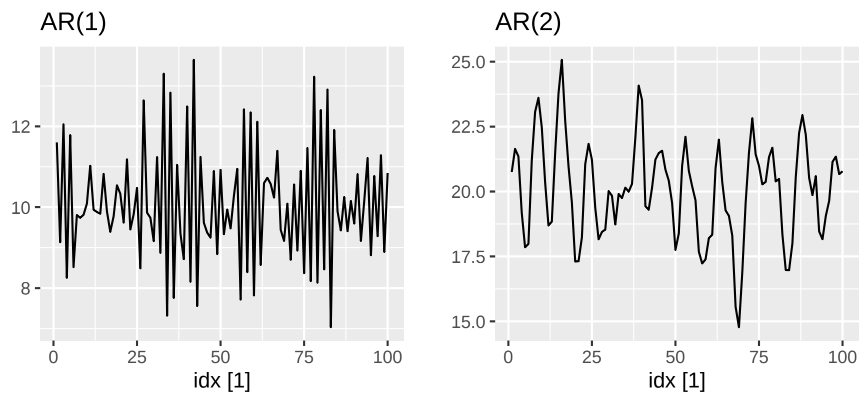
2.2 Non-seasonal ARIMA models

2.2.1 Autoregressive models

Autoregressive (AR) models:

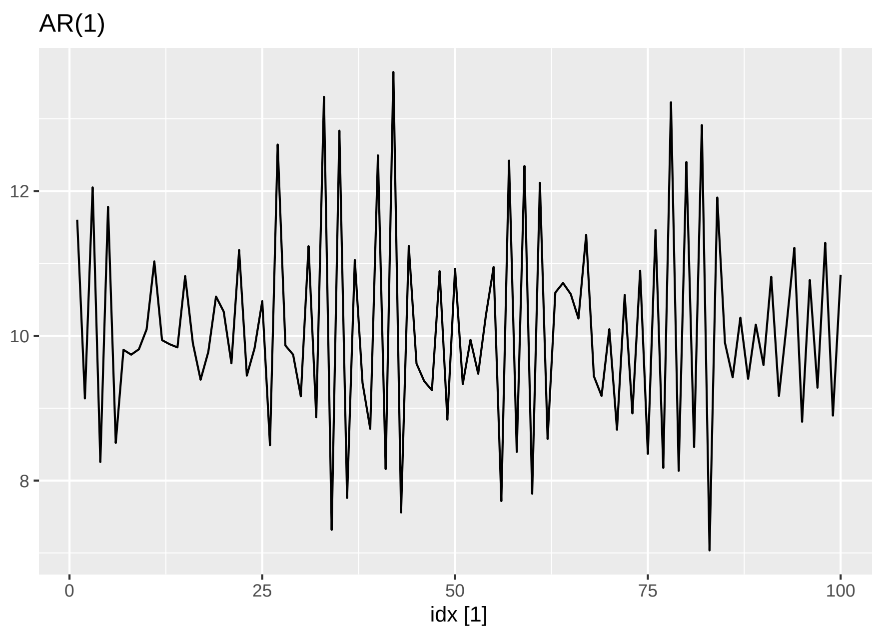
$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t,$$

where ε_t is white noise. This is a multiple regression with **lagged values** of y_t as predictors.



2.2.1.1 AR(1) model

$$y_t = 18 - 0.8y_{t-1} + \varepsilon_t$$
$$\varepsilon_t \sim N(0, 1), \quad T = 100.$$



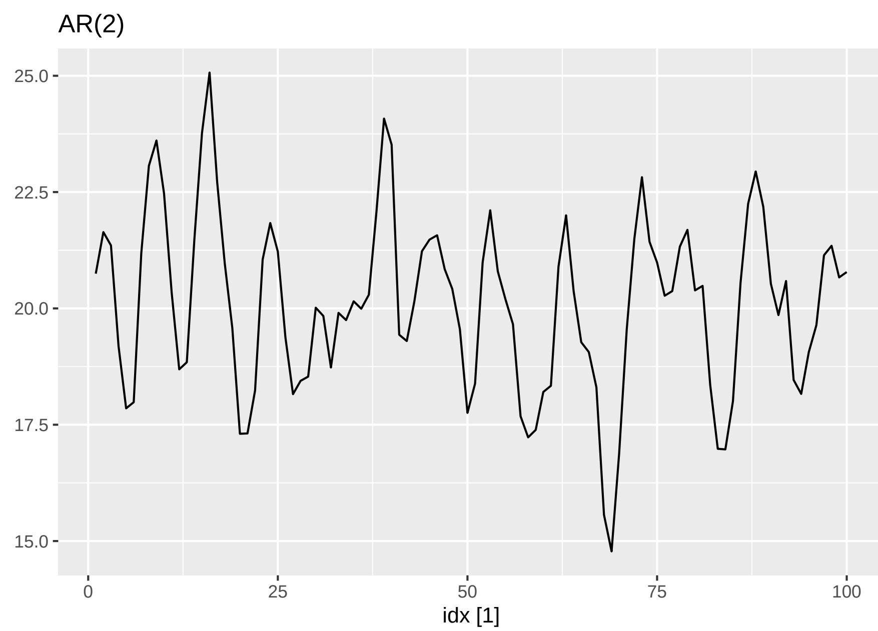
$$y_t = c + \phi_1 y_{t-1} + \varepsilon_t$$

- When $\phi_1 = 0$, y_t is **equivalent to WN**

- When $\phi_1 = 1$ and $c = 0$, y_t is **equivalent to a RW**
- When $\phi_1 = 1$ and $c \neq 0$, y_t is **equivalent to a RW with drift**
- When $\phi_1 < 0$, y_t tends to **oscillate between positive and negative values**.

2.2.1.2 AR(2) model

$$y_t = 8 + 1.3y_{t-1} - 0.7y_{t-2} + \varepsilon_t$$
$$\varepsilon_t \sim N(0,1), \qquad T = 100.$$



2.2.1.3 Stationarity conditions

- We normally restrict autoregressive models to stationary data, and then some constraints on the values of the parameters are required.

General condition for stationarity

Complex roots of $1 - \phi_1 z - \phi_2 z^2 - \cdots - \phi_p z^p$ lie outside the unit circle on the complex plane.

- For $p = 1$: $-1 < \phi_1 < 1$.
- For $p = 2$:
  $-1 < \phi_2 < 1 \qquad \phi_2 + \phi_1 < 1 \qquad \phi_2 - \phi_1 < 1$.

- More complicated conditions hold for $p \geq 3$.
- Estimation software takes care of this.

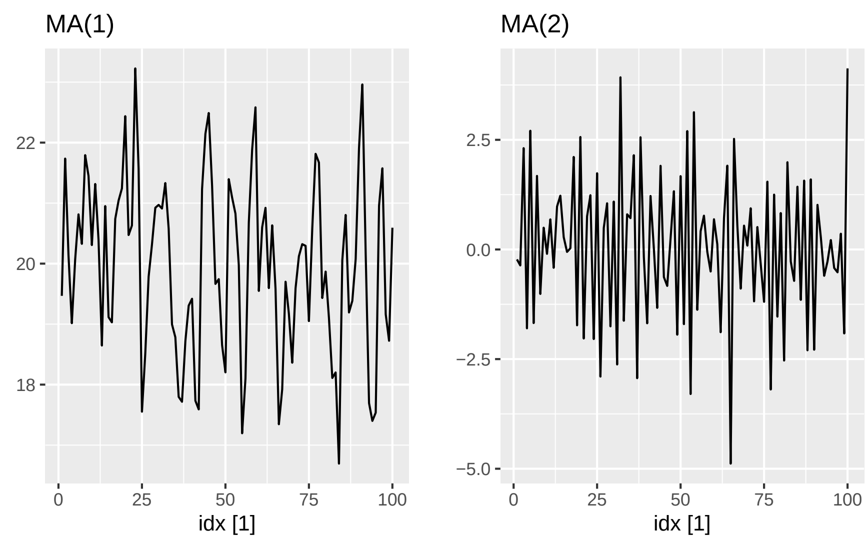
2.2.2 Moving Average (MA) models

Moving Average (MA) models:

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q},$$

where ε_t is white noise. This is a multiple regression with **past errors** as predictors.

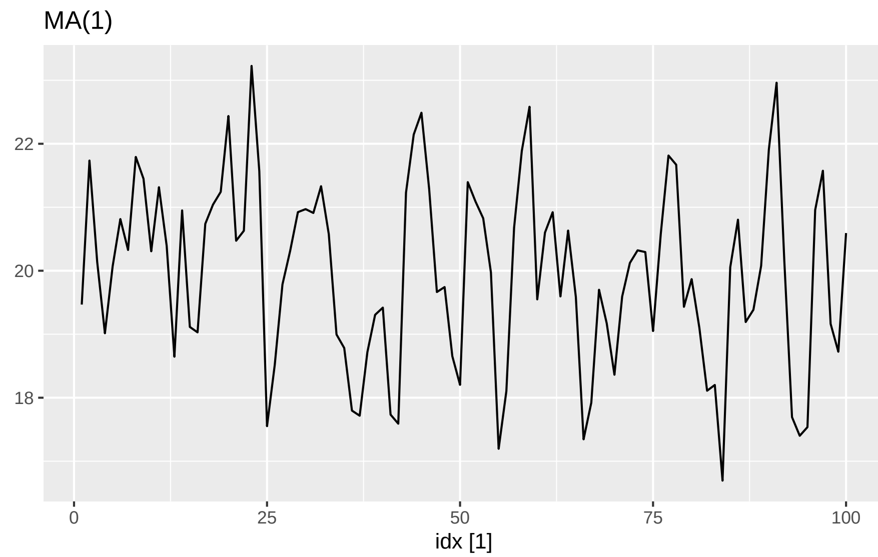
- Don't confuse this with moving average smoothing!



2.2.2.1 MA(1) model

$$y_t = 20 + \varepsilon_t + 0.8\varepsilon_{t-1}$$

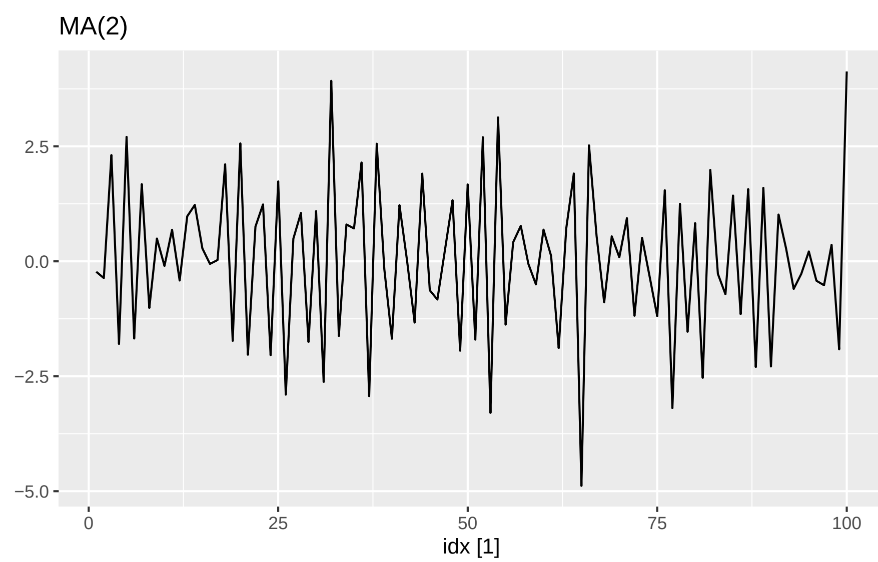
$$\varepsilon_t \sim N(0,1), \quad T = 100.$$



2.2.2.2 MA(2) model

$$y_t = \varepsilon_t - \varepsilon_{t-1} + 0.8\varepsilon_{t-2}$$

$$\varepsilon_t \sim N(0,1), \quad T = 100.$$



2.2.2.3 MA(∞) models

It is possible to write any stationary AR(p) process as an MA(∞) process.

Example: AR(1)

$$\begin{aligned} y_t &= \phi_1 y_{t-1} + \varepsilon_t \\ &= \phi_1(\phi_1 y_{t-2} + \varepsilon_{t-1}) + \varepsilon_t \\ &= \phi_1^2 y_{t-2} + \phi_1 \varepsilon_{t-1} + \varepsilon_t \\ &= \phi_1^3 y_{t-3} + \phi_1^2 \varepsilon_{t-2} + \phi_1 \varepsilon_{t-1} + \varepsilon_t \\ &\dots \end{aligned}$$

Provided $-1 < \phi_1 < 1$:

$$y_t = \varepsilon_t + \phi_1 \varepsilon_{t-1} + \phi_1^2 \varepsilon_{t-2} + \phi_1^3 \varepsilon_{t-3} + \cdots$$

2.2.3 Invertibility

- Any MA(q) process can be written as an AR(∞) process if we impose some constraints on the MA parameters.
- Then the MA model is called "invertible".
- Invertible models have some mathematical properties that make them easier to use in practice.
- Invertibility of an ARIMA model is equivalent to forecastability of an ETS model.

General condition for invertibility

Complex roots of $1 + \theta_1 z + \theta_2 z^2 + \dots + \theta_q z^q$ lie outside the unit circle on the complex plane.

- For $q = 1$: $-1 < \theta_1 < 1$.
- For $q = 2$:
  $-1 < \theta_2 < 1 \qquad \theta_2 + \theta_1 > -1 \qquad \theta_1 - \theta_2 < 1$.
- More complicated conditions hold for $q \geq 3$.
- Estimation software takes care of this.

2.2.4 ARIMA models

Autoregressive Moving Average models:

$$\begin{aligned} y_t &= c + \phi_1 y_{t-1} + \dots + \phi_p y_{t-p} \\ &+ \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} + \varepsilon_t \end{aligned}$$

- Predictors include both **lagged values of y_t and lagged errors.**
- Conditions on coefficients ensure stationarity.
- Conditions on coefficients ensure invertibility.

Autoregressive Integrated Moving Average models

- Combine ARMA model with **differencing**.
- $(1-B)^d y_t$ follows an ARMA model.

Autoregressive Integrated Moving Average models

ARIMA(p,d,q) model

- **AR:** $p =$ order of the autoregressive part
- **I:** $d =$ degree of first differencing involved
- **MA:** $q =$ order of the moving average part.
  - White noise model: ARIMA(0,0,0)
  - Random walk: ARIMA(0,1,0) with no constant
  - Random walk with drift: ARIMA(0,1,0) with const.
  - AR(p): ARIMA(p,0,0)
  - MA(q): ARIMA(0,0,q)

2.2.5 Backshift notation for ARIMA

- **ARMA model:**

$$y_t = c + \phi_1 B y_t + \cdots + \phi_p B^p y_t + \varepsilon_t + \theta_1 B \varepsilon_t + \cdots + \theta_q B^q \varepsilon_t$$

$$\text{or} \quad (1 - \phi_1 B - \cdots - \phi_p B^p) y_t = c + (1 + \theta_1 B + \cdots + \theta_q B^q)\varepsilon_t$$

ARIMA(1,1,1) model:

$$(1 - \phi_1 B)(1 - B) y_t = c + (1 + \theta_1 B)\varepsilon_t$$

NOTE:

Written out:

$$y_t = c + y_{t-1} + \phi_1 y_{t-1} - \phi_1 y_{t-2} + \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

2.3 Estimation and order selection

2.3.1 Maximum likelihood estimation

Having identified the model order, we need to estimate the parameters c, $\phi_1, \dots, \phi_p$, $\theta_1, \dots, \theta_q$.

- MLE is very similar to least squares estimation obtained by minimizing

$$\sum_{t-1}^{T} e_t^2$$

- The `ARIMA()` function allows CLS or MLE estimation.
- Non-linear optimization must be used in either case.
- Different software will give different estimates.

2.3.2 Partial autocorrelations

Partial autocorrelations measure relationship between y_t and y_{t-k}, when the effects of other time lags — $1, 2, 3, \dots, k-1$ — are removed.

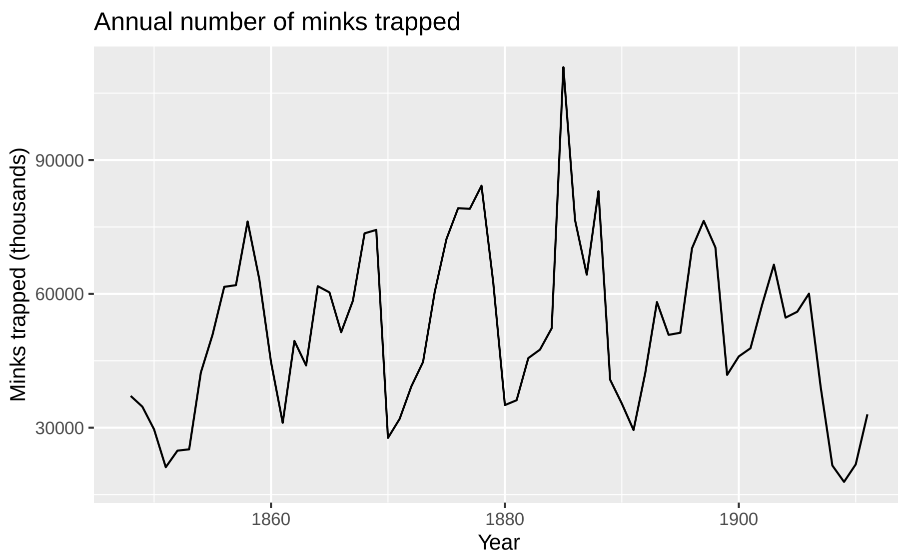
$$\alpha_k = k\text{th partial autocorrelation coefficient}$$

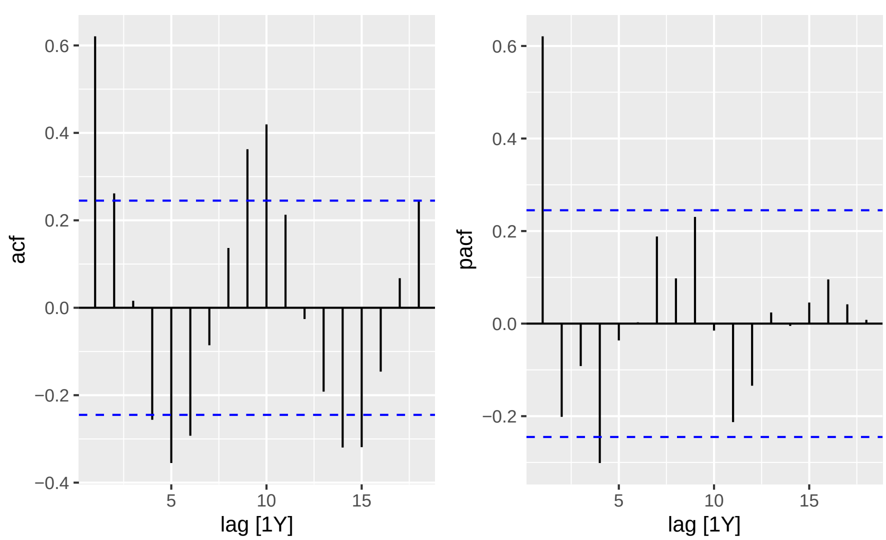
$$= \text{equal to the estimate of } \phi_k \text{ in regression:}$$

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_k y_{t-k}.$$

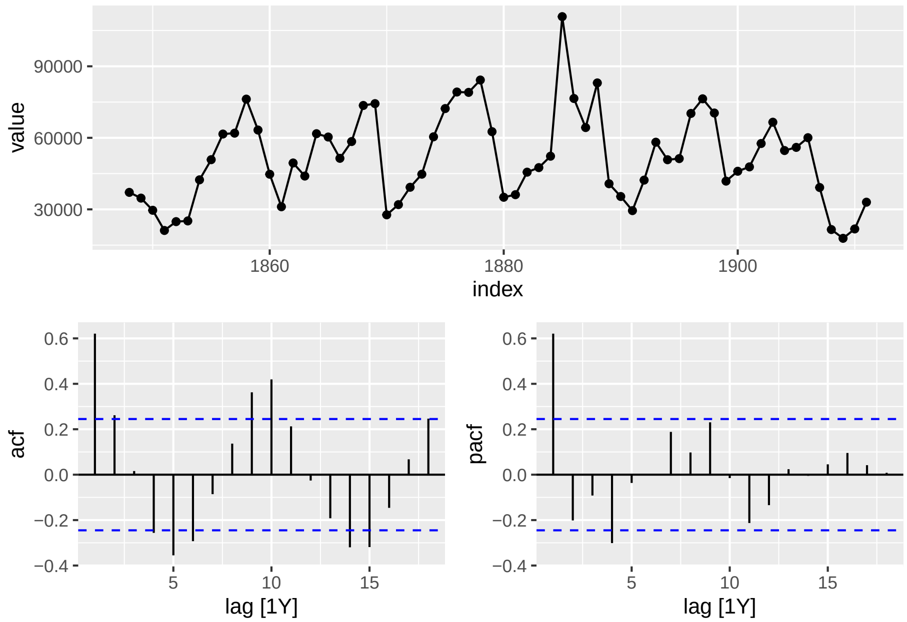
- Varying number of terms on RHS gives α_k for different values of k.
- $\alpha_1 = \rho_1$
- same critical values of $\pm 1.96/\sqrt{T}$ as for ACF.
- Last significant α_k indicates the order of an AR model.

2.3.2.1 Example: Mink trapping






```
mink %>% gg_tsdisplay(value, plot_type='partial')
```



2.3.3 ACF and PACF interpretation

AR(1)

$$\rho_k = \phi_1^k \qquad \text{for } k = 1, 2, \dots;$$

$$\alpha_1 = \phi_1 \qquad \alpha_k = 0 \qquad \text{for } k = 2, 3, \dots.$$

So we have an AR(1) model when

- autocorrelations exponentially decay
- there is a single significant partial autocorrelation.

AR(p)

- ACF dies out in an exponential or damped sine-wave manner
- PACF has all zero spikes beyond the pth spike

So we have an AR(p) model when

- the ACF is exponentially decaying or sinusoidal

- there is a significant spike at lag p in PACF, but none beyond p

MA(1)

$$\rho_1 = \theta_1 \qquad \rho_k = 0 \qquad \text{for } k = 2, 3, \dots;$$

$$\alpha_k = -(-\theta_1)^k$$

So we have an MA(1) model when

- the PACF is exponentially decaying and
- there is a single significant spike in ACF

MA(q)

- PACF dies out in an exponential or damped sine-wave manner
- ACF has all zero spikes beyond the qth spike

So we have an MA(q) model when

- the PACF is exponentially decaying or sinusoidal
- there is a significant spike at lag q in ACF, but none beyond q

2.3.4 Information criteria

Akaike's Information Criterion (AIC)

$$\text{AIC} = -2\log(L) + 2(p + q + k + 1),$$

where L is the likelihood of the data, $k = 1$ if $c \neq 0$ and $k = 0$ if $c = 0$.

Corrected AIC:

$$\text{AICc} = \text{AIC} + \frac{2(p + q + k + 1)(p + q + k + 2)}{T - p - q - k - 2}.$$

Bayesian Information Criterion:

$$\text{BIC} = \text{AIC} + [\log(T) - 2](p + q + k + 1).$$

- Good models are obtained by minimizing either the AIC, AICc or BIC.
- Our preference is to use the AICc.

2.4 Seasonal ARIMA models

ARIMA	$\underbrace{(p,d,q)}$	$\underbrace{(P,D,Q)_m}$
	$\uparrow$ Non-seasonal part of the model	$\uparrow$ Seasonal part of of the model

where m = number of observations per year.

Example: ARIMA$(1,1,1)(1,1,1)_4$ model (without constant)

$$(1-\phi_1 B)(1-\Phi_1 B^4)(1-B)(1-B^4)y_t \;=\; (1+\theta_1 B)(1+\Theta_1 B^4)\varepsilon_t.$$

All the factors can be multiplied out and the general model written as follows:

$$y_t = (1+\phi_1)y_{t-1} - \phi_1 y_{t-2} + (1+\Phi_1)y_{t-4} - (1+\phi_1+\Phi_1+\phi_1\Phi_1)y_{t-5} + (\phi_1+\phi_1\Phi_1)y_{t-6}$$

$$-\Phi_1 y_{t-8} + (\Phi_1+\phi_1\Phi_1)y_{t-9} - \phi_1\Phi_1 y_{t-10} + \varepsilon_t + \theta_1\varepsilon_{t-1} + \Theta_1\varepsilon_{t-4} + \theta_1\Theta_1\varepsilon_{t-5}.$$

2.4.1 Common ARIMA models

The US Census Bureau uses the following models most often:

ARIMA(0,1,1)(0,1,1)$_m$	with log transformation
ARIMA(0,1,2)(0,1,1)$_m$	with log transformation
ARIMA(2,1,0)(0,1,1)$_m$	with log transformation
ARIMA(0,2,2)(0,1,1)$_m$	with log transformation
ARIMA(2,1,2)(0,1,1)$_m$	with no transformation

2.4.2 Seasonal ARIMA models

The seasonal part of an AR or MA model will be seen in the seasonal lags of the PACF and ACF.

ARIMA(0,0,0)(0,0,1)$_{12}$ will show:

- a spike at lag 12 in the ACF but no other significant spikes.
- The PACF will show exponential decay in the seasonal lags; that is, at lags 12, 24, 36, ….

ARIMA(0,0,0)(1,0,0)$_{12}$ will show:

- exponential decay in the seasonal lags of the ACF
- a single significant spike at lag 12 in the PACF.

2.5 Theoretical properties of the models

2.5.1 Autoregressive (AR) models

2.5.1.1 Properties of AR(1) model

Consider the following $AR(1)$ model.

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \epsilon_t \tag{2.1}$$

where ε_t is white noise.

2.5.1.1.1 Mean

Assuming that the series is weak stationary, we have $E(Y_t) = \mu$, $Var(Y_t) = \gamma_0$, and $Cov(Y_t, Y_{t-k}) = \gamma_k$, where μ and γ_0 are constants. Given that ϵ_t is a white noise, we have $E(\epsilon_t) = 0$. The mean of $AR(1)$ process can be computed as follows:

$$\begin{aligned} E(Y_t) &= E(\phi_0 + \phi_1 Y_{t-1}) \\ &= E(\phi_0) + E(\phi_1 Y_{t-1}) \\ &= \phi_0 + \phi_1 E(Y_{t-1}). \end{aligned}$$

Under the stationarity condition, $E(Y_t) = E(Y_{t-1}) = \mu$. Thus we get

$$\mu = \phi_0 + \phi_1 \mu.$$

Solving for μ yields

$$E(Y_t) = \mu = \frac{\phi_0}{1 - \phi_1}. \tag{2.2}$$

The results has two constraints for Y_t. First, the mean of Y_t exists if $\phi_1 \neq 1$. The mean of Y_t is zero if and only if $\phi_0 = 0$.

2.5.1.1.2 Variance and the stationary condition of AR (1) process

First take variance of both sides of Equation (2.1)

$$Var(Y_t) = Var(\phi_0 + \phi_1 Y_{t-1} + \epsilon_t)$$

The Y_{t-1} occurred before time t. The ϵ_t does not depend on any past observation. Hence, $cov(Y_{t-1}, \epsilon_t) = 0$. Furthermore, ϵ_t is a white noise. This gives

$$Var(Y_t) = \phi_1^2 Var(Y_{t-1}) + \sigma^2.$$

Under the stationarity condition, $Var(Y_t) = Var(Y_{t-1})$. Hence,

$$Var(Y_t) = \frac{\sigma^2}{1 - \phi_1^2}.$$

provided that $\phi_1^2 < 1$ or $|\phi_1| < 1$ (The variance of a random variable is bounded and non-negative). The necessary and sufficient condition for the $AR(1)$ model in Equation (2.1) to be weakly stationary is $|\phi_1| < 1$. This condition is equivalent to saying that the root of $1 - \phi_1 B = 0$ must lie outside the unit circle. This can be explained as below

Using the backshift notation we can write $AR(1)$ process as

$$Y_t = \phi_0 + \phi_1 BY_t + \epsilon_t.$$

Then we get

$$(1 - \phi_1 B)Y_t = \phi_0 + \epsilon_t.$$

The $AR(1)$ process is said to be stationary if the roots of $(1 - \phi_1 B) = 0$ lie outside the unit circle.

2.5.1.1.3 Covariance

The covariance $\gamma_k = Cov(Y_t, Y_{t-k})$ is called the lag-k autocovariance of Y_t. The two main properties of γ_k: (a) $\gamma_0 = Var(Y_t)$ and (b) $\gamma_{-k} = \gamma_k$.

The lag-k autocovariance of Y_t is

$$\begin{aligned}
\gamma_k &= Cov(Y_t, Y_{t-k}) \\
&= E[(Y_t - \mu)(Y_{t-k} - \mu)] \\
&= E[Y_t Y_{t-k} - Y_t \mu - \mu Y_{t-k} + \mu^2] \\
&= E(Y_t Y_{t-k}) - \mu^2.
\end{aligned} \tag{2.3}$$

Now we have

$$E(Y_t Y_{t-k}) = \gamma_k + \mu^2 \tag{2.4}$$

2.5.1.1.4 Autocorrelation function of an AR(1) process

To derive autocorrelation function of an AR(1) process we first multiply both sides of Equation (2.1) by Y_{t-k} and take expected values:

$$E(Y_t Y_{t-k}) = \phi_0 E(Y_{t-k}) + \phi_1 E(Y_{t-1} Y_{t-k}) + E(\epsilon_t Y_{t-k})$$

Since ϵ_t and Y_{t-k} are independent and using the results in Equation (4.3)

$$\gamma_k + \mu^2 = \phi_0 \mu + \phi_1(\gamma_{k-1} + \mu^2)$$

Substituting the results in Equation (4.2) to Equation (4.3) we get

$$\gamma_k = \phi_1 \gamma_{k-1}. \tag{2.5}$$

The autocorrelation function, ρ_k, is defined as

$$\rho_k = \frac{\gamma_k}{\gamma_0}$$

.

Setting $k = 1$, we get $\gamma_1 = \phi_1 \gamma_0$. Hence,

$$\rho_1 = \phi_1.$$

Similarly with $k = 2$, $\gamma_2 = \phi_1 \gamma_1$. Dividing both sides by γ_0 and substituting with $\rho_1 = \phi_1$ we get

$$\rho_2 = \phi_1^2.$$

Now it is easy to see that in general

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \phi_1^k \tag{2.6}$$

for $k = 0, 1, 2, 3, ...$

Since $|\phi_1| < 1$, the autocorrelation function is an exponentially decreasing as the number of lags k increases. There are two features in the ACF of AR(1) process depending on the sign of ϕ_1. They are,

1. If $0 < \phi_1 < 1$, all correlations are positive.
2. if $-1 < \phi_1 < 0$, the lag 1 autocorrelation is negative ($\rho_1 = \phi_1$) and the signs of successive autocorrelations alternate from positive to negative with their magnitudes decreasing exponentially.

2.5.1.2 Properties of AR(2) model

Now consider a second-order autoregressive process (AR(2))

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \epsilon_t. \tag{2.7}$$

2.5.1.2.1 Mean

Question 1: Using the same technique as that of the AR(1), show that

$$E(Y_t) = \mu = \frac{\phi_0}{1 - \phi_1 - \phi_2}$$

and the mean of Y_t exists if $\phi_1 + \phi_2 \neq 1$.

2.5.1.2.2 Variance

Question 2: Show that

$$Var(Y_t) = \frac{(1-\phi_2)\sigma^2}{(1+\phi_2)((1+\phi_2)^2-\phi_1^2)}.$$

Here is a guide to the solution

Start with

$$Var(Y_t) = Var(\phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \epsilon_t)$$

Solve it until you obtain the Eq. (a) as shown below.

$$\gamma_0(1-\phi_1^2-\phi_2^2) = 2\phi_1\phi_2\gamma_1 + \sigma^2. \tag{a}$$

Next multiply both sides of Equation (2.7) by Y_{t-1} and obtain an expression for γ_1. Let's call this Eq. (b).

Solve Eq. (a) and (b) for γ_0.

2.5.1.2.3 Stationarity of AR(2) process

To discuss the stationarity condition of the $AR(2)$ process we use the roots of the characteristic polynomial. Here is the illustration.

Using the backshift notation we can write $AR(2)$ process as

$$Y_t = \phi_0 + \phi_1 BY_t + \phi_2 B^2 Y_t + \epsilon_t.$$

Furthermore, we get

$$(1-\phi_1 B - \phi_2 B^2)Y_t = \phi_0 + \epsilon_t.$$

The **characteristic polynomial** of $AR(2)$ process is

$$\Phi(B) = 1 - \phi_1 B - \phi_2 B^2.$$

and the corresponding **AR characteristic equation**

$$1 - \phi_1 B - \phi_2 B^2 = 0.$$

For stationarity, the roots of AR characteristic equation must lie outside the unit circle. The two roots of the AR characteristic equation are

$$\frac{\phi_1 \pm \sqrt{\phi_1^2 + 4\phi_2}}{-2\phi_2}$$

Using algebraic manipulation, we can show that these roots will exceed 1 in modulus if and only if simultaneously $\phi_1 + \phi_2 < 1$, $\phi_2 - \phi_1 < 1$, and $|\phi_2| < 1$. This is called the stationarity condition of $AR(2)$ process.

2.5.1.2.4 Autocorrelation function of an AR(2) process

To derive autocorrelation function of an AR(2) process we first multiply both sides of Equation (2.7) by Y_{t-k} and take expected values:

$$\begin{aligned} E(Y_t Y_{t-k}) &= E(\phi_0 Y_{t-k} + \theta_1 Y_{t-1} Y_{t-k} + \theta_2 Y_{t-2} Y_{t-k} + \epsilon_t Y_{t-k}) & (2.8) \\ &= \phi_0 E(Y_{t-k}) + \phi_1 E(Y_{t-1} Y_{t-k}) + \phi_2 E(Y_{t-2} Y_{t-k}) + E(\epsilon_t Y_{t-k}). & (2.9) \end{aligned}$$

Using the independence between ϵ_t and Y_{t-1}, $E(\epsilon_t Y_{t-k}) = 0$ and the results in Equation (4.3) (This is valid for AR(2)) we have

$$\gamma_k + \mu^2 = \phi_0 \mu + \theta_1(\gamma_{k-1} + \mu^2) + \phi_2(\gamma_{k-2} + \mu^2).$$

(Note that $E(X_{t-1}X_{t-k}) = E(X_{t-1}X_{(t-1)-(k-1)} = \gamma_{k-1}))$

Solving for γ_k we get

$$\gamma_k = \phi_1 \gamma_{k-1} + \phi_2 \gamma_{k-2}. \tag{2.10}$$

By dividing both sides of Equation (2.10) by γ_0, we have

$$\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2}. \tag{2.11}$$

for $k > 0$.

Setting $k = 1$ and using $\rho_0 = 1$ and $\rho_{-1} = \rho_1$, we get **the Yule-Walker equation for $AR(2)$ process.**

$$\rho_1 = \phi_1 + \phi_2 \rho_1$$

or

$$\rho_1 = \frac{\phi_1}{1 - \phi_2}.$$

Similarly, we can show that

$$\rho_2 = \frac{\phi_2(1-\phi_2)+\phi_1^2}{(1-\phi_2)}.$$

2.5.1.3 Properties of AR(p) model

The pth order autoregressive model can be written as

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + \epsilon_t. \tag{2.12}$$

The AR characteristic equation is

$$1 - \phi_1 B - \phi_2 B^2 - ... - \phi_p B^p = 0.$$

For stationarity of $AR(p)$ process, the p roots of the AR characteristic must lie outside the unit circle.

2.5.1.3.1 Mean

Question 3: Find $E(Y_t)$ of $AR(p)$ process.

2.5.1.3.2 Variance

Question 4: Find $Var(Y_t)$ of $AR(p)$ process.

2.5.1.3.3 Autocorrelation function (ACF) of an AR(p) process

Question 5: Similar to the results in Equation (2.11) for $AR(2)$ process, obtain the following recursive relationship for $AR(p)$.

$$\rho_k = \phi_1\rho_{k-1} + \phi_2\rho_{k-2} + ... + \phi_p\rho_{k-p}. \tag{2.13}$$

Setting $k = 1, 2, ..., p$ into Equation (2.13) and using $\rho_0 = 1$ and $\rho_{-k} = \rho_k$, we get the Yule-Walker equations for $AR(p)$ process

$$\begin{aligned}
\rho_1 &= \phi_1 + \phi_2\rho_1 + ... + \phi_p\rho_{p-1} \\
\rho_2 &= \phi_1\rho_1 + \phi_2 + ... + \phi_p\rho_{p-2} \\
&... \\
\rho_p &= \phi_1\rho_{p-1} + \phi_2\rho_{p-2} + ... + \phi_p
\end{aligned} \tag{2.14}$$

The Yule-Walker equations in (2.14) can be written in matrix form as below.

$$\begin{bmatrix} \rho_1 \\ \rho_2 \\ \cdot \\ \cdot \\ \cdot \\ \rho_p \end{bmatrix} = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdot & \cdot & \cdot & \rho_{p-1} \\ \rho_1 & 1 & \rho_1 & \cdot & \cdot & \cdot & \rho_{p-2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \rho_{p-1} & \rho_{p-2} & \rho_{p-3} & \cdot & \cdot & \cdot & 1 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \cdot \\ \cdot \\ \cdot \\ \phi_p \end{bmatrix}$$

or

$$\boldsymbol{\rho}_\mathbf{p} = \mathbf{P}_\mathbf{p}\boldsymbol{\phi}.$$

where,

$$\boldsymbol{\rho}_\mathbf{p} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \cdot \\ \cdot \\ \cdot \\ \rho_p \end{bmatrix}, \mathbf{P}_\mathbf{p} = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdot & \cdot & \cdot & \rho_{p-1} \\ \rho_1 & 1 & \rho_1 & \cdot & \cdot & \cdot & \rho_{p-2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \rho_{p-1} & \rho_{p-2} & \rho_{p-3} & \cdot & \cdot & \cdot & 1 \end{bmatrix}, \boldsymbol{\phi} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \cdot \\ \cdot \\ \cdot \\ \phi_p \end{bmatrix}$$

The parameters can be estimated using

$$\boldsymbol{\phi} = \mathbf{P}_\mathbf{p}^{-1}\boldsymbol{\rho}_\mathbf{p}.$$

Question 6: Obtain the parameters of an $AR(3)$ process whose first autocorrelations are $\rho_1 = 0.9$; $\rho_2 = 0.9$; $\rho_3 = 0.5$. Is the process stationary?

2.5.1.3.4 The partial autocorrelation function (PACF)

Let ϕ_{kj}, the jth coefficient in an $AR(k)$ model. Then, ϕ_{kk} is the last coefficient. From Equation (2.13), the ϕ_{kj} satisfy the set of equations

$$\rho_j = \phi_{k1}\rho_{j-1} + ... + \phi_{k(k-1)}\rho_{j-k+1} + \phi_{kk}\rho_{j-k}, \tag{2.15}$$

for $j = 1, 2, ...k$, leading to the Yule-Walker equations which may be written

$$\begin{bmatrix} \rho_1 \\ \rho_2 \\ \cdot \\ \cdot \\ \cdot \\ \rho_k \end{bmatrix} = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdot & \cdot & \cdot & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & \cdot & \cdot & \cdot & \rho_{k-2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdot & \cdot & \cdot & 1 \end{bmatrix} \begin{bmatrix} \phi_{k1} \\ \phi_{k2} \\ \cdot \\ \cdot \\ \cdot \\ \phi_{kk} \end{bmatrix} \tag{2.16}$$

or

$$\boldsymbol{\rho_k} = \mathbf{P_k}\boldsymbol{\phi_k}.$$

where

$$\boldsymbol{\rho_k} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ . \\ . \\ . \\ \rho_k \end{bmatrix}, \mathbf{P_k} = \begin{bmatrix} 1 & \rho_1 & \rho_2 & . & . & . & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & . & . & . & \rho_{k-2} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & . & . & . & 1 \end{bmatrix}, \boldsymbol{\phi_k} = \begin{bmatrix} \phi_{k1} \\ \phi_{k2} \\ . \\ . \\ . \\ \phi_{kk} \end{bmatrix}$$

For each k, we compute the coefficients ϕ_{kk}. Solving the equations for $k = 1, 2, 3...$ successively, we obtain

For $k = 1$,

$$\phi_{11} = \rho_1. \tag{2.17}$$

For $k = 2$,

$$\phi_{22} = \frac{\begin{bmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{bmatrix}}{\begin{bmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{bmatrix}} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \tag{2.18}$$

For $k = 3$,

$$\phi_{33} = \frac{\begin{bmatrix} 1 & \rho_1 & \rho_1 \\ \rho_1 & 1 & \rho_2 \\ \rho_2 & \rho_1 & \rho_3 \end{bmatrix}}{\begin{bmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_1 \\ \rho_2 & \rho_1 & 1 \end{bmatrix}} \tag{2.19}$$

The quantity ϕ_{kk} is called the partial autocorrelation at lag k and can be defined as

$$\phi_{kk} = Corr(Y_t Y_{t-k} | Y_{t-1}, Y_{t-2}, ..., Y_{t-k+1}).$$

The partial autocorrelation between Y_t and Y_{t-k} is the correlation between Y_t and Y_{t-k} after removing the effect of the intermediate variables $Y_{t-1}, Y_{t-2}, ..., Y_{t-k+1}$.

In general the determinant in the numerator of Equations (2.17), (2.18) and (2.19) has the same elements as that in the denominator, but replacing the last column with $\boldsymbol{\rho_k} = (\rho_1, \rho_2, ...\rho_k)$.

2.5.1.3.5 PACF for AR(1) models

From Equation (2.6) we have

$\rho_k = \phi_1^k$ for $k = 0, 1, 2, 3, ...$

Hence, for $k = 1$, the first partial autocorrelation coefficient is

$$\phi_{11} = \rho_1 = \phi_1.$$

From (2.18) for $k = 2$, the second partial autocorrelation coefficient is

$$\phi_{22} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} = \frac{\phi_1^2 - \phi_1^2}{1 - \phi_1^2} = 0$$

.

Similarly, for $AR(1)$ we can show that $\phi_{kk} = 0$ for all $k > 1$. Hence, for $AR(1)$ process the partial autocorrelation is non-zero for lag 1 which is the order of the process, but is zero for lags beyond the order 1.

2.5.1.3.6 PACF for AR(2) model

Question 7: For $AR(2)$ process show that $\phi_{kk} = 0$ for all $k > 2$. Sketch the PACF of $AR(2)$ process.

2.5.1.3.7 PACF for AR(P) model

In general for $AR(p)$ process, the partial autocorrelation function ϕ_{kk} is non-zero for k less than or equal to p (the order of the process) and zero for all k greater than p. In other words, the partial autocorrelation function of a $AR(p)$ process has a cut-off after lag p.

2.5.2 Moving average (MA) models

We first derive the properties of $MA(1)$ and $MA(2)$ models and then give the results for the general $MA(q)$ model.

2.5.2.1 Properties of MA(1) model

The general form for $MA(1)$ model is

$$Y_t = \theta_0 + \theta_1 \epsilon_{t-1} + \epsilon_t \tag{2.20}$$

where θ_0 is a constant and ϵ_t is a white noise series.

2.5.2.1.1 Mean

Question 8: Show that $E(Y_t) = \theta_0$.

2.5.2.1.2 Variance

Question 9: Show that $Var(Y_t) = (1 + \theta_1^2)\sigma^2$.

We can see both mean and variance are time-invariant. MA models are finite linear combinations of a white noise sequence. Hence, MA processes are always weakly stationary.

2.5.2.1.3 Autocorrelation function of an MA(1) process

Method 1

To obtain the autocorrelation function of $MA(1)$, we first multiply both sides of Equation (2.20) by Y_{t-k} and take the expectation.

$$\begin{aligned} E[Y_t Y_{t-k}] &= E[\theta_0 Y_{t-k} + \theta_1 \epsilon_{t-1} Y_{t-k} + \epsilon_t Y_{t-k}] \\ &= \theta_0 E(Y_{t-k}) + \theta_1 E(\epsilon_{t-1} Y_{t-k}) + E(\epsilon_t Y_{t-k}) \end{aligned} \tag{2.21}$$

Using the independence between ϵ_t and Y_{t-k} (future error and past observation) $E(\epsilon_t Y_{t-k}) = 0$. Now we have

$$E[Y_t Y_{t-k}] = \theta_0^2 + \theta_1 E(\epsilon_{t-1} Y_{t-k}) \tag{2.22}$$

Now let's obtain an expression for $E[Y_t Y_{t-k}]$.

$$\begin{aligned} \gamma_k &= Cov(Y_t, Y_{t-k}) \\ &= E[(Y_t - \theta_0)(Y_{t-k} - \theta_0)] \\ &= E[Y_t Y_{t-k} - Y_t \theta_0 - \theta_0 Y_{t-k} + \theta_0^2] \\ &= E(Y_t Y_{t-k}) - \theta_0^2. \end{aligned} \tag{2.23}$$

Now we have

$$E(Y_t Y_{t-k}) = \gamma_k + \theta_0^2. \tag{2.24}$$

Using the Equations (2.22) and (2.24) we have

$$\gamma_k = \theta_0^2 - \theta_0^2 + \theta_1 E(\epsilon_{t-1} Y_{t-k}). \tag{2.25}$$

Now let's consider the case $k = 1$.

$$\gamma_1 = \theta_0^2 - \theta_0^2 + \theta_1 E(\epsilon_{t-1} Y_{t-1}) \tag{2.26}$$

Today's error and today's value are dependent. Hence, $E(\epsilon_{t-1} Y_{t-1}) \neq 0$. We first need to identify $E(\epsilon_{t-1} Y_{t-1})$.

$$E(\epsilon_{t-1} Y_{t-1}) = E(\theta_0 \epsilon_{t-1} + \theta_1 \epsilon_{t-2} \epsilon_{t-1} + \epsilon_{t-1}^2) \tag{2.27}$$

Since, $\{\epsilon_t\}$ is a white noise process $E(\epsilon_{t-1}) = 0$ and $E(\epsilon_{t-2} \epsilon_{t-1}) = 0$. Hence, we have

$$E(\epsilon_{t-1} Y_{t-1}) = E(\epsilon_{t-1}^2) = \sigma^2 \tag{2.28}$$

Substituting (2.28) in (2.26) we get

$$\gamma_1 = \theta_1 \sigma^2$$

.

Furthermore, $\gamma_0 = Var(Y_t) = (1 + \theta_1^2)\sigma^2$. Hence

$$\rho_1 = \frac{\gamma_1}{\gamma_0} = \frac{\theta_1}{1 + \theta_1^2}.$$

When $k = 2$, from Equation (2.26) and $E(\epsilon_{t-1} Y_{t-2}) = 0$ (future error and past observation) we get $\gamma_2 = 0$. Hence $\rho_2 = 0$. Similarly, we can show that

$$\gamma_k = \rho_k = 0$$

for all $k \geq 2$.

We can see that the ACF of $MA(1)$ process is zero, beyond the order of 1 of the process.

Method 2: By using the definition of covariance

$$\begin{aligned} \gamma_1 = Cov(Y_t, Y_{t-1}) &= Cov(\epsilon_t + \theta_1 \epsilon_{t-1} + \theta_0, \epsilon_{t-1} + \theta_1 \epsilon_{t-2} + \theta_0) \\ &= Cov(\theta_1 \epsilon_{t-1}, \epsilon_{t-1}) \\ &= \theta_1 \sigma^2. \end{aligned} \tag{2.29}$$

$$\begin{aligned} \gamma_2 = Cov(Y_t, Y_{t-2}) &= Cov(\epsilon_t + \theta_1 \epsilon_{t-1} + \theta_0, \epsilon_{t-2} + \theta_1 \epsilon_{t-3} + \theta_0) \\ &= 0. \end{aligned} \tag{2.30}$$

We have $\gamma_0 = \sigma^2(1 + \theta_1^2)$, (Using the variance).

Hence

$$\rho_1 = \frac{\gamma_1}{\gamma_0} = \frac{\theta_1}{1+\theta_1^2}.$$

Similarly we can show $\gamma_k = \rho_k = 0$ for all $k \geq 2$.

2.5.2.2 Properties of MA(2) model

An $MA(2)$ model is in the form

$$Y_t = \theta_0 + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + \epsilon_t \tag{2.31}$$

where θ_0 is a constant and ϵ_t is a white noise series.

2.5.2.2.1 Mean

Question 10: Show that $E(Y_t) = \theta_0$.

2.5.2.2.2 Variance

Question 11: Show that $Var(Y_t) = \sigma^2(1+\theta_1^2+\theta_2^2)$.

2.5.2.2.3 Autocorrelation function of an MA(2) process

Question 12: For $MA(2)$ process show that,

$$\rho_1 = \frac{\theta_1(1+\theta_2)}{1+\theta_1^2+\theta_2^2},$$

$$\rho_2 = \frac{\theta_2}{1+\theta_1^2+\theta_2^2},$$

and $\rho_k = 0$ for all $k \geq 3$.

2.5.2.3 Properties of MA(q) model

$$Y_t = \theta_0 + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + ... + \theta_q\epsilon_{t-q} + \epsilon_t \tag{2.32}$$

where θ_0 is a constant and ϵ_t is a white noise series.

2.5.2.3.1 Mean

Question 13: Show that the constant term of an MA model is the mean of the series (i.e. $E(Y_t) = \theta_0$).

2.5.2.3.2 Variance

Question 14: Show that the variance of an MA model is

$$Var(Y_t) = (1 + \theta_1^2 + \theta_2^2 + ... + \theta_q^2)\sigma^2.$$

2.5.2.3.3 Autocorrelation function of an MA(q) process

Question 15: Show that the autocorrelation function of a $MA(q)$ process is zero, beyond the order of q of the process. In other words, the autocorrelation function of a moving average process has a cutoff after lag q.

2.5.2.3.4 Partial autocorrelation function of an MA(q) process

The partial autocorrelation functions for $MA(q)$ models behave very much like the autocorrelation functions of $AR(p)$ models. The PACF of MA models decays exponentially to zero, rather like ACF for AR model.

2.5.3 Dual relation between AR and MA process

Dual relation 1

First we consider the relation AR(p) <–> MA(∞)

Let $AR(p)$ be a **stationary** AR model with order p. Then,

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

where $\epsilon_t \sim WN(0, \sigma^2)$.

Using the backshift operator we can write the $AR(p)$ model as

$$(1 - \phi_1 B - \phi_2 B^2 - ... - \phi_p B^P)Y_t = \epsilon_t.$$

Then

$$\phi(B)Y_t = \epsilon_t,$$

where $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - ... - \phi_p B^p$. Furthermore, Y_t can be written as infinite sum of previous ϵ's as below

$$Y_t = \phi^{-1}(B)\epsilon_t,$$

where $\phi(B)\psi(B) = 1$ and $\psi(B) = 1 + \Psi_1 B + \psi_2 B^2 + ...$ Then

$$Y_t = \psi(B)\epsilon_t.$$

This is a representation of $MA(\infty)$ process.

Next, we consider the relation MA(q) <–> AR(∞)

Let $MA(q)$ be **invertible** moving average process

$$Y_t = \epsilon_t + \theta_t\epsilon_{t-1} + \theta_2\epsilon_{t-2} + ... + \theta_p\epsilon_{t-q}.$$

Using the backshift operator we can write the $MA(q)$ process as

$$Y_t = (1 + \theta_1 B + \theta_2 B^2 - ... + \theta_q B^q)\epsilon_t.$$

Then,

$$Y_t = \theta(B)\epsilon_t,$$

where $\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + ... + \theta_1 B^q$. Hence, for an **invertible** moving average process, Y_t can be represented as a finite weighted sum of previous error terms, ϵ. Furthermore, since the process is invertible ϵ_t can be represented as an infinite weighted sum of previous Y's as below

$$\epsilon_t = \theta^{-1}(B)Y_t,$$

where $\pi(B)\theta(B) = 1$, and $\pi(B) = 1 + \pi_1 B + \pi B^2 + ...$ Hence,

$$\epsilon_t = \pi(B)Y_t.$$

This is an representation of a $AR(\infty)$ process.

Dual relation 2

An $MA(q)$ process has an ACF function that is zero beyond lag q and its PACF is decays exponentially to 0. Consequently, an $AR(p)$ process has an PACF that is zero beyond lag-p, but its ACF decays exponentially to 0.

Dual relation 3

For an $AR(p)$ process the roots of $\phi(B) = 0$ must lie outside the unit circle to satisfy the condition of stationarity. However, the parameters of the $AR(p)$ are not required to satisfy any conditions to ensure invertibility. Conversely, the parameters of the MA process are not required to satisfy any condition to ensure stationarity. However, to ensure the condition of invertibility, the roots of $\theta(B) = 0$ must lie outside the unit circle.

2.5.4 Autoregressive and Moving-average (ARMA) models

current value = linear combination of past values + linear combination of past error + current error

The $ARMA(p, q)$ can be written as

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + ... + \theta_q \epsilon_{t-q} + \epsilon_t,$$

where $\{\epsilon_t\}$ is a white noise process.

Using the back shift operator

$$\phi(B) Y_t = \theta(B) \epsilon_t,$$

where $\phi(.)$ and $\theta(.)$ are the pth and qth degree polynomials,

$$\phi(B) = 1 - \phi_1 \epsilon - ... - \phi_p \epsilon^p,$$

and

$$\theta(B) = 1 + \theta_1 \epsilon + ... + \theta_q \epsilon^q.$$

2.5.4.1 Stationary condition

Roots of

$$\phi(B) = 0$$

lie outside the unit circle.

2.5.4.2 Invertible condition

Roots of

$$\theta(B) = 0$$

lie outside the unit circle.

2.5.4.3 Autocorrelation function and Partial autocorrelation function

The ACF of an ARMA model exhibits a pattern similar to that of an AR model. The PACF of ARMA process behaves like the PACF of a MA process. Hence, the ACF and PACF are not informative in determining the order of an ARMA model.

2.6 Unit root tests

- Many financial time series are with trending behavior or nonstationarity in the mean.
- Two common trend removal or de-trending procedures
  - First differencing (appropriate for $I(1)$ time series).
  - time-trend regression (appropriate for trend stationary $I(0)$ time series).
- Unit root tests are statistical tests to determine the required order of differencing or whether it should be regressed on deterministic functions of time to render the data stationary.

2.6.1 Dickey-Fuller test

- Consider the model
$$\Delta y_t = c + \beta y_{t-1} + \epsilon_t$$
- Hypothesis to be tested $H_0 : \beta = 0$ and $H_1 : \beta < 0$
- Test statistics $= \frac{\hat{\beta}}{SE(\hat{\beta})}$

2.6.2 Augmented Dickey-Fuller test

- The Dickey-Fuller Unit Root Test is valid if the time series y_t is well characterized by an AR(1) model with white noise errors.
- Many financial time series have a more complicated dynamic structures
- The Augmented Dickey-Fuller (ADF) test allows for higher-order autoregressive processes by including Δy_{t-p} in the model.
- The number of lags included in the model should be just sufficient to remove any autocorrelation in errors.
- Consider the model:
$$\Delta y_t = c + \beta y_{t-1} + \alpha_1 \Delta y_{t-1} + \cdots + \alpha_p \Delta y_{t-p} + \epsilon_t$$
- Hypothesis to be tested $H_0 : \beta = 0$ and $H_1 : \beta < 0$
- ADF test: null hypothesis is that the data are non-stationary and non-seasonal.
- DF and ADF tests are not suitable when there is a deterministic trend

- Alternative tests:
  - Phillips-Perron Unit Root Tests
- The main difference between Phillips-Perron (PP) unit root tests and the ADF tests is in the way they deal with serial correlation and heteroskedasticity in the errors.

2.6.3 Stationarity Tests

- The ADF unit root test tests the null hypothesis that a time series y_t is $I(1)$.
- In contrast, Stationarity tests are for the null that y_t is $I(0)$.
- Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test is the most commonly used stationarity test that tests the null hypothesis that the data are stationary and non-seasonal.
- Other tests available for seasonal data

2.6.3.1 Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test

```
google_2018 %>%
  features(Close, unitroot_kpss)
```

```
## # A tibble: 1 x 3
##   Symbol kpss_stat kpss_pvalue
##   <chr>      <dbl>       <dbl>
## 1 GOOG       0.573      0.0252
```

```
google_2018 %>%
  mutate(diff_close = difference(Close)) %>%
  features(diff_close, unitroot_kpss)
```

```
## # A tibble: 1 x 3
##   Symbol kpss_stat kpss_pvalue
##   <chr>      <dbl>       <dbl>
## 1 GOOG      0.0955         0.1
```

```
google_2018 %>%
  features(Close, unitroot_ndiffs)
```

```
## # A tibble: 1 x 2
##   Symbol ndiffs
##   <chr>   <int>
## 1 GOOG        1
```

2.6.3.2 Automatically selecting differences

STL decomposition: $y_t = T_t + S_t + R_t$

Seasonal strength $F_s = \max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(S_t+R_t)}\right)$

If $F_s > 0.64$, do one seasonal difference.

```
usmelec %>% mutate(log_gen = log(Generation)) %>%
  features(log_gen, list(unitroot_nsdiffs, feat_stl))
```

```
## # A tibble: 1 x 10
##   nsdiffs trend_strength seasonal_streng~ seasonal_peak_y~
##     <int>          <dbl>            <dbl>            <dbl>
## 1       1          0.994            0.941                7
## # ... with 6 more variables: seasonal_trough_year <dbl>,
## #   spikiness <dbl>, linearity <dbl>, curvature <dbl>,
## #   stl_e_acf1 <dbl>, stl_e_acf10 <dbl>
```

```
usmelec %>% mutate(log_gen = log(Generation)) %>%
  features(log_gen, unitroot_nsdiffs)
```

```
## # A tibble: 1 x 1
##   nsdiffs
##     <int>
## 1       1
```

```
usmelec %>% mutate(d_log_gen = difference(log(Generation), 12)) %>%
  features(d_log_gen, unitroot_ndiffs)
```

```
## # A tibble: 1 x 1
##   ndiffs
##    <int>
## 1      1
```

2.7 ARIMA modelling in R

2.7.1 How does ARIMA() work?

A non-seasonal ARIMA process

$$\phi(B)(1-B)^d y_t = c + \theta(B)\varepsilon_t$$

Need to select appropriate orders: p, q, d

Hyndman and Khandakar (JSS, 2008) algorithm:

- Select no. differences d and D via KPSS test and seasonal strength measure.
- Select p, q by minimising AICc.
- Use step-wise search to traverse model space.

$$\text{AICc} = -2\log(L) + 2(p+q+k+1)\left[1 + \frac{(p+q+k+2)}{T-p-q-k-2}\right].$$

where L is the maximised likelihood fitted to the *differenced* data, $k = 1$ if $c \neq 0$ and $k = 0$ otherwise.

Step1: Select current model (with smallest AICc) from:
ARIMA$(2, d, 2)$
ARIMA$(0, d, 0)$
ARIMA$(1, d, 0)$
ARIMA$(0, d, 1)$

Step 2: Consider variations of current model:

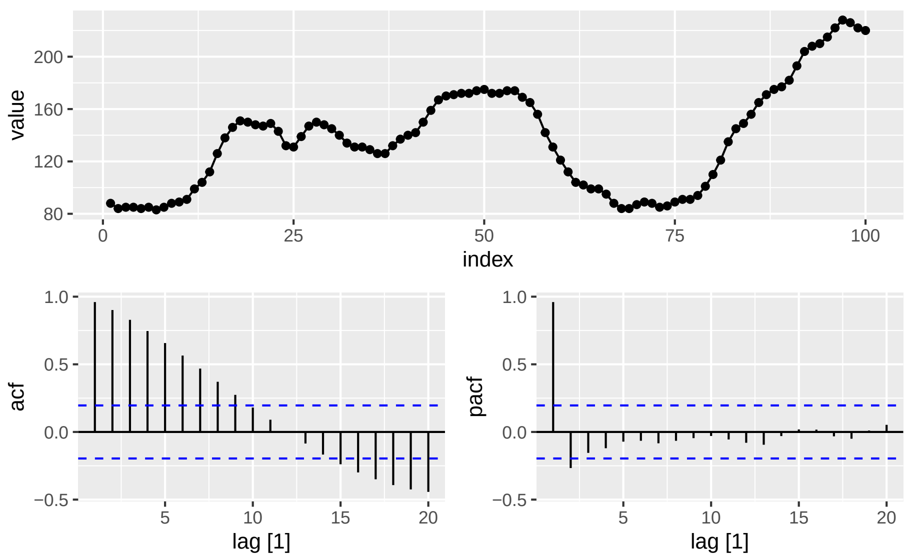
- vary one of p, q, from current model by ± 1;
- p, q both vary from current model by ± 1;
- Include/exclude c from current model.

Model with lowest AICc becomes current model.

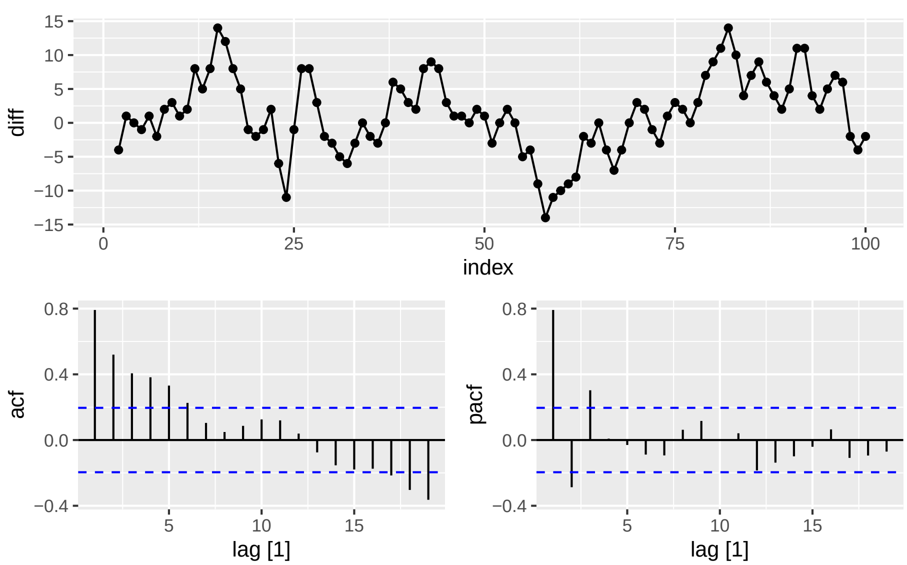
Repeat Step 2 until no lower AICc can be found.

2.7.2 Choosing your own model

```
web_usage <- as_tsibble(WWWusage)
web_usage %>% gg_tsdisplay(value, plot_type = 'partial')
```




```r
web_usage %>% mutate(diff = difference(value)) %>%
  gg_tsdisplay(diff, plot_type = 'partial')
```




```r
fit <- web_usage %>%
  model(arima = ARIMA(value ~ pdq(3, 1, 0)))
report(fit)
```

```
## Series: value
```

```
## Model: ARIMA(3,1,0)
##
## Coefficients:
##          ar1      ar2     ar3
##        1.151  -0.6612  0.3407
## s.e.  0.095   0.1353  0.0941
##
## sigma^2 estimated as 9.656:  log likelihood=-252
## AIC=512   AICc=512.4   BIC=522.4
```

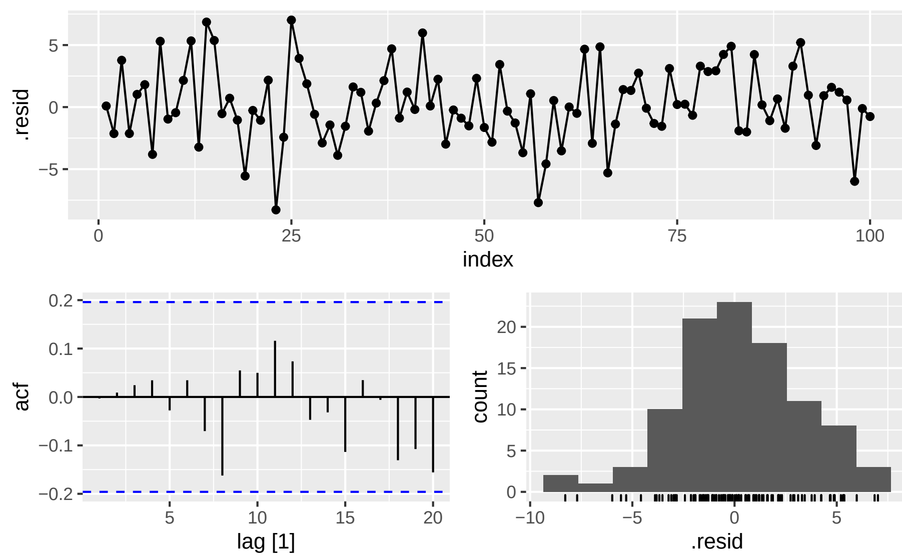
```
web_usage %>%
  model(ARIMA(value ~ pdq(d=1))) %>%
report()
```

```
## Series: value
## Model: ARIMA(1,1,1)
##
## Coefficients:
##          ar1     ma1
##       0.6504  0.5256
## s.e.  0.0842  0.0896
##
## sigma^2 estimated as 9.995:  log likelihood=-254.2
## AIC=514.3   AICc=514.5   BIC=522.1
```

```
web_usage %>%
  model(ARIMA(value ~ pdq(d=1),
    stepwise = FALSE, approximation = FALSE)) %>%
  report()
```

```
## Series: value
## Model: ARIMA(3,1,0)
##
## Coefficients:
##          ar1      ar2     ar3
##        1.151  -0.6612  0.3407
## s.e.  0.095   0.1353  0.0941
##
## sigma^2 estimated as 9.656:  log likelihood=-252
## AIC=512   AICc=512.4   BIC=522.4
```

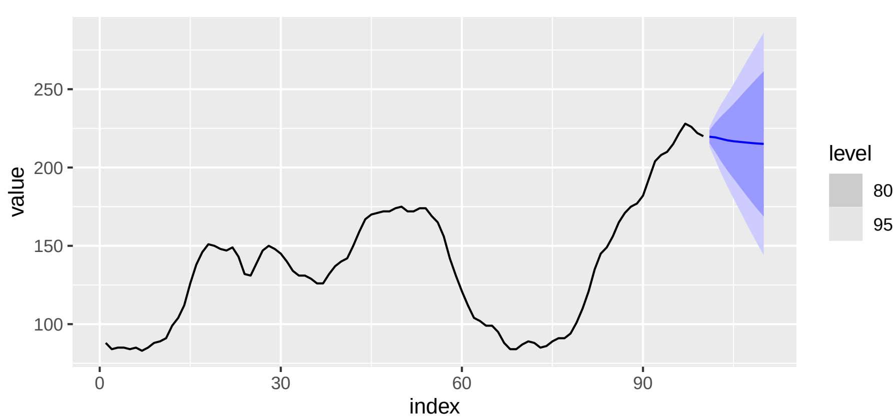
```
gg_tsresiduals(fit)
```




```r
augment(fit) %>%
  features(.resid, ljung_box, lag = 10, dof = 3)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 arima     4.49     0.722
```

```r
fit %>% forecast(h = 10) %>%
  autoplot(web_usage)
```



2.7.3 Modelling procedure with `ARIMA()`

1. Plot the data. Identify any unusual observations.
2. If necessary, transform the data (using a Box-Cox transformation) to stabilize the variance.
3. If the data are non-stationary: take first differences of the data until the data are stationary.
4. Examine the ACF/PACF: Is an AR(p) or MA(q) model appropriate?
5. Try your chosen model(s), and use the $AICc$ to search for a better model.
6. Check the residuals from your chosen model by plotting the ACF of the residuals, and doing a portmanteau test of the residuals. If they do not look like white noise, try a modified model.
7. Once the residuals look like white noise, calculate forecasts.

2.7.4 Automatic modelling procedure with `ARIMA()`

1. Plot the data. Identify any unusual observations.
2. If necessary, transform the data (using a Box-Cox transformation) to stabilize the variance.

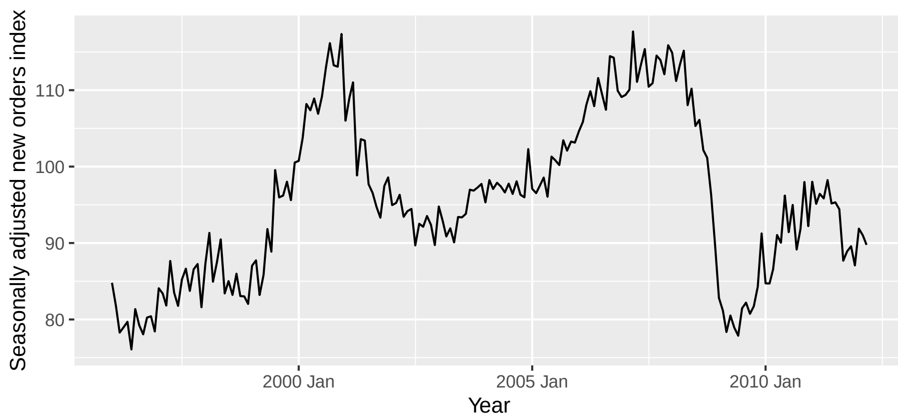
3. Use `ARIMA` to automatically select a model.

6. Check the residuals from your chosen model by plotting the ACF of the residuals, and doing a portmanteau test of the residuals. If they do not look like white noise, try a modified model.
7. Once the residuals look like white noise, calculate forecasts.

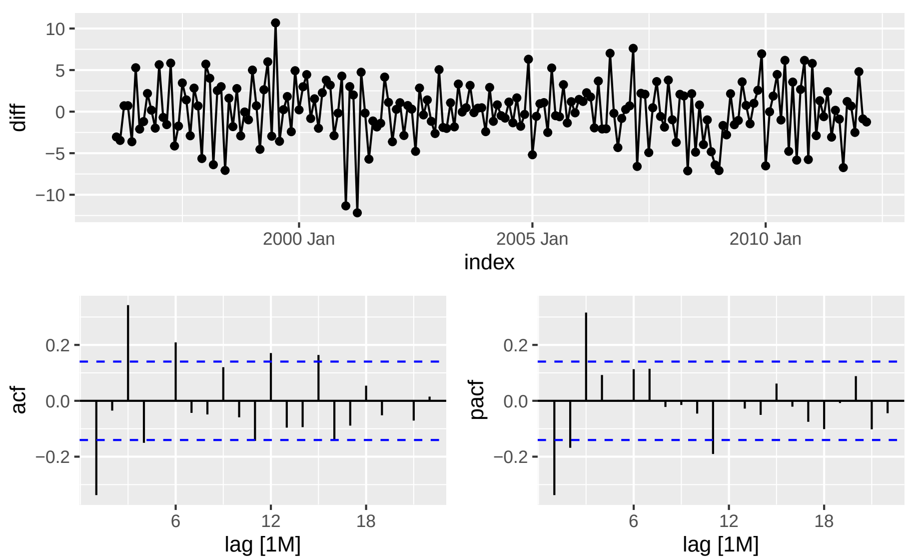
2.7.5 Example in R

Seasonally adjusted electrical equipment

```
elecequip <- as_tsibble(fpp2::elecequip)
dcmp <- elecequip %>%
  model(STL(value ~ season(window = "periodic"))) %>%
  components() %>% select(-.model)
dcmp %>% as_tsibble %>%
  autoplot(season_adjust) + xlab("Year") +
  ylab("Seasonally adjusted new orders index")
```




```
dcmp %>% mutate(diff = difference(season_adjust)) %>%
  gg_tsdisplay(diff, plot_type = 'partial')
```




```
fit <- dcmp %>%
  model(arima = ARIMA(season_adjust))
report(fit)
```

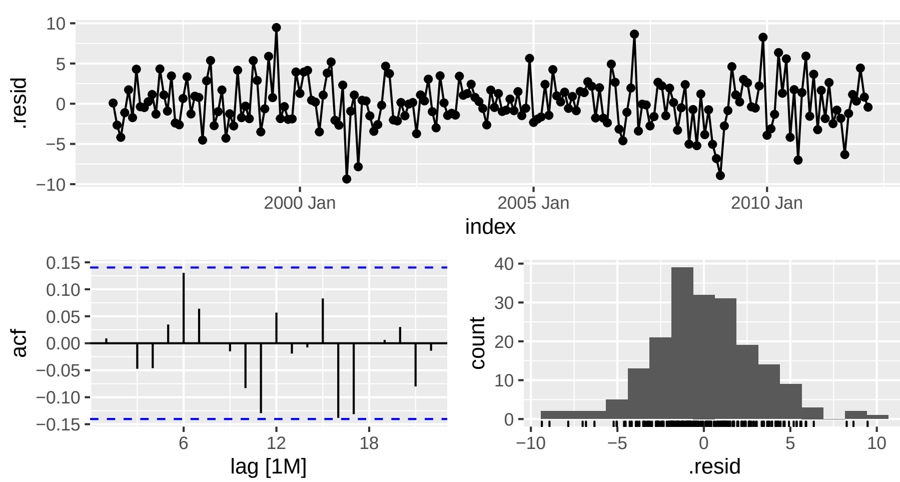
```
## Series: season_adjust
## Model: ARIMA(3,1,0)
##
## Coefficients:
##             ar1       ar2      ar3
```

```
##       -0.3418  -0.0426  0.3185
## s.e.   0.0681   0.0725  0.0682
##
## sigma^2 estimated as 9.639:  log likelihood=-493.8
## AIC=995.6   AICc=995.8   BIC=1009
```

```
fit <- dcmp %>%
  model(arima = ARIMA(season_adjust, approximation=FALSE))
report(fit)
```

```
## Series: season_adjust
## Model: ARIMA(3,1,1)
##
## Coefficients:
##          ar1     ar2     ar3      ma1
##       0.0044  0.0916  0.3698  -0.3921
## s.e.  0.2201  0.0984  0.0669   0.2426
##
## sigma^2 estimated as 9.577:  log likelihood=-492.7
## AIC=995.4   AICc=995.7   BIC=1012
```

```
gg_tsresiduals(fit)
```

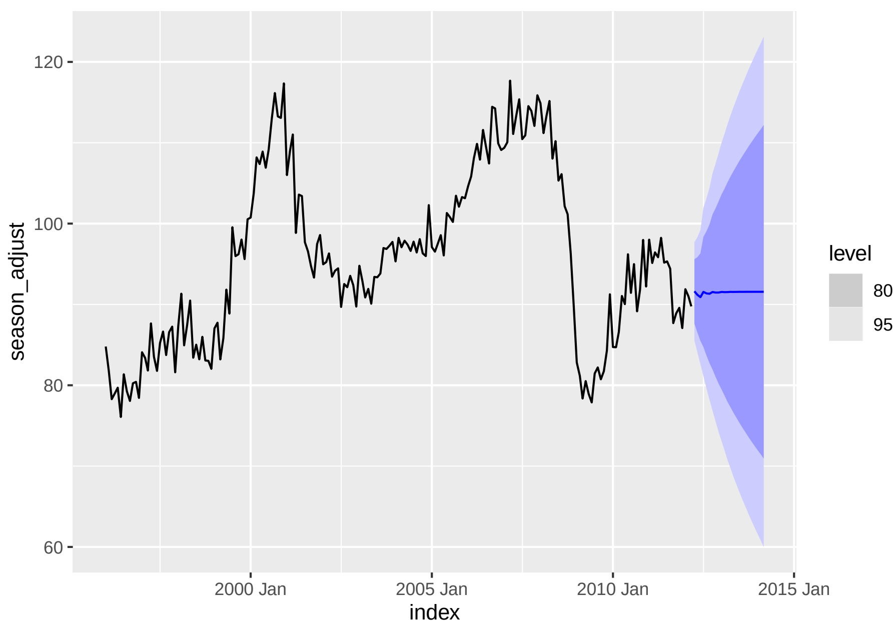



```
augment(fit) %>%
  features(.resid, ljung_box, lag = 24, dof = 4)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
```

```
##   <chr>     <dbl>     <dbl>
## 1 arima      24.0     0.241
```

```
fit %>% forecast %>% autoplot(dcmp)
```



2.8 Forecasting

2.8.1 Point forecasts

1. Rearrange ARIMA equation so y_t is on LHS.
2. Rewrite equation by replacing t by $T + h$.
3. On RHS, replace future observations by their forecasts, future errors by zero, and past errors by corresponding residuals.

Start with $h = 1$. Repeat for $h = 2, 3, \ldots$.

Example:

ARIMA(3,1,1) forecasts: Step 1

$$(1 - \phi_1 B - \phi_2 B^2 - \phi_3 B^3)(1 - B) y_t = (1 + \theta_1 B)\varepsilon_t,$$

$$[1-(1+\phi_1)B+(\phi_1-\phi_2)B^2+(\phi_2-\phi_3)B^3+\phi_3B^4]y_t=(1+\theta_1B)\varepsilon_t$$

$$y_t-(1+\phi_1)y_{t-1}+(\phi_1-\phi_2)y_{t-2}+(\phi_2-\phi_3)y_{t-3}+\phi_3y_{t-4}=\varepsilon_t+\theta_1\varepsilon_{t-1}.$$

$$y_t=(1+\phi_1)y_{t-1}-(\phi_1-\phi_2)y_{t-2}-(\phi_2-\phi_3)y_{t-3}-\phi_3y_{t-4}+\varepsilon_t+\theta_1\varepsilon_{t-1}.$$

2.8.1.1 Point forecasts (h=1)

$$y_t=(1+\phi_1)y_{t-1}-(\phi_1-\phi_2)y_{t-2}-(\phi_2-\phi_3)y_{t-3}-\phi_3y_{t-4}+\varepsilon_t+\theta_1\varepsilon_{t-1}.$$

ARIMA(3,1,1) forecasts: Step 2

$$y_{T+1}=(1+\phi_1)y_T-(\phi_1-\phi_2)y_{T-1}-(\phi_2-\phi_3)y_{T-2}-\phi_3y_{T-3}+\varepsilon_{T+1}+\theta_1\varepsilon_T.$$

ARIMA(3,1,1) forecasts: Step 3

$$\hat{y}_{T+1|T}=(1+\phi_1)y_T-(\phi_1-\phi_2)y_{T-1}-(\phi_2-\phi_3)y_{T-2}-\phi_3y_{T-3}+\theta_1e_T.$$

2.8.1.2 Point forecasts (h=2)

$$y_t=(1+\phi_1)y_{t-1}-(\phi_1-\phi_2)y_{t-2}-(\phi_2-\phi_3)y_{t-3}-\phi_3y_{t-4}+\varepsilon_t+\theta_1\varepsilon_{t-1}.$$

ARIMA(3,1,1) forecasts: Step 2

$$y_{T+2}=(1+\phi_1)y_{T+1}-(\phi_1-\phi_2)y_T-(\phi_2-\phi_3)y_{T-1}-\phi_3y_{T-2}+\varepsilon_{T+2}+\theta_1\varepsilon_{T+1}.$$

ARIMA(3,1,1) forecasts: Step 3

$$\hat{y}_{T+2|T}=(1+\phi_1)\hat{y}_{T+1|T}-(\phi_1-\phi_2)y_T-(\phi_2-\phi_3)y_{T-1}-\phi_3y_{T-2}.$$

2.8.2 Prediction intervals

95% prediction interval

$$\hat{y}_{T+h|T} \pm 1.96\sqrt{v_{T+h|T}}$$

where $v_{T+h|T}$ is estimated forecast variance.

- $v_{T+1|T} = \hat{\sigma}^2$ for all ARIMA models regardless of parameters and orders.
- Multi-step prediction intervals for $ARIMA(0, 0, q)$):

$$y_t = \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}.$$

$$\hat{\sigma}_h = \hat{\sigma}^2 \left[1 + \sum_{i=1}^{h-1} \theta_i^2\right], \qquad \text{for } h = 2, 3, \ldots.$$

- AR(1): Rewrite as MA(∞) and use above result.
- Other models beyond scope of this subject.
- Prediction intervals **increase in size with forecast horizon**.
- Prediction intervals can be difficult to calculate by hand
- Calculations assume residuals are **uncorrelated** and **normally distributed**.
- Prediction intervals tend to be too narrow.
  - the uncertainty in the parameter estimates has not been accounted for.
  - the ARIMA model assumes historical patterns will not change during the forecast period.
  - the ARIMA model assumes uncorrelated future errors.

2.9 References:

- Brockwell, P. J., Brockwell, P. J., Davis, R. A., & Davis, R. A. (2016). Introduction to time series and forecasting. springer.

- Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts.

- Zivot, E., & Wang, J. (2006). Unit root tests. Modeling Financial Time Series with S-PLUS®, 111-139.

Chapter 3

Exponential Smoothing

3.1 Introduction

3.1.1 Historical perspective

- Developed in the 1950s and 1960s as methods (algorithms) to produce point forecasts.
- Combine a "level", "trend" (slope) and "seasonal" component to describe a time series.
- The rate of change of the components are controlled by "smoothing parameters": α, β and γ respectively.
- Need to choose best values for the smoothing parameters (and initial states).
- Equivalent ETS state space models developed in the 1990s and 2000s.

3.1.2 Big idea: control the rate of change

α controls the flexibility of the **level**

- If $\alpha = 0$, the level never updates (mean)
- If $\alpha = 1$, the level updates completely (naive)

β controls the flexibility of the **trend**

- If $\beta = 0$, the trend is linear (regression trend)
- If $\beta = 1$, the trend updates every observation

γ controls the flexibility of the **seasonality**

- If $\gamma = 0$, the seasonality is fixed (seasonal means)
- If $\gamma = 1$, the seasonality updates completely (seasonal naive)

3.1.3 A model for levels, trends, and seasonalities

We want a model that captures the level (ℓ_t), trend (b_t) and seasonality (s_t).

How do we combine these elements?

- Additively?

$$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$$

- Multiplicatively?

$$y_t = \ell_{t-1} b_{t-1} s_{t-m}(1 + \varepsilon_t)$$

- Perhaps a mix of both?

$$y_t = (\ell_{t-1} + b_{t-1}) s_{t-m} + \varepsilon_t$$

How do the level, trend and seasonal components evolve over time?

General notation:

ETS: **E**xponen**T**ial **S**moothing

Error: Additive (`"A"`) or multiplicative (`"M"`)

Trend: None (`"N"`), additive (`"A"`), multiplicative (`"M"`), or damped (`"Ad"` or `"Md"`).

Seasonality: None (`"N"`), additive (`"A"`) or multiplicative (`"M"`)

3.2 Simple exponential smoothing

Time series $y_1, y_2, \dots, y_T$.

Random walk forecasts

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

Average forecasts

$$\hat{y}_{T+h|T} = \frac{1}{T}\sum_{t=1}^{T} y_t$$

- Want something in between these methods.
- Most recent data should have more weight.

Forecast equation

$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1-\alpha)y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + \cdots$$

where $0 \leq \alpha \leq 1$

	Weights assigned to observations for:			
Observation	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$
y_T	0.2	0.4	0.6	0.8
y_{T-1}	0.16	0.24	0.24	0.16
y_{T-2}	0.128	0.144	0.096	0.032
y_{T-3}	0.1024	0.0864	0.0384	0.0064
y_{T-4}	$(0.2)(0.8)^4$	$(0.4)(0.6)^4$	$(0.6)(0.4)^4$	$(0.8)(0.2)^4$
y_{T-5}	$(0.2)(0.8)^5$	$(0.4)(0.6)^5$	$(0.6)(0.4)^5$	$(0.8)(0.2)^5$

Component form

- Forecast equation $\hat{y}_{t+h|t} = \ell_t$
- Smoothing equation $\ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}$
- ℓ_t is the level (or the smoothed value) of the series at time t.
- $\hat{y}_{t+1|t} = \alpha y_t + (1-\alpha)\hat{y}_{t|t-1}$
  Iterate to get exponentially weighted moving average form.

Weighted average form

$$\hat{y}_{T+1|T} = \sum_{j=0}^{T-1} \alpha(1-\alpha)^j y_{T-j} + (1-\alpha)^T \ell_0$$

3.2.1 Optimising smoothing parameters

- Need to choose best values for α and ℓ_0.
  - Similarly to regression, choose optimal parameters by minimising SSE:

$$\text{SSE} = \sum_{t=1}^{T}(y_t - \hat{y}_{t|t-1})^2.$$

- Unlike regression there is no closed form solution — use numerical optimization.
- For Algerian Exports example:
  - $\hat{\alpha} = 0.8400$
  - $\hat{\ell}_0 = 39.54$

3.2.2 Models and methods

3.2.2.1 Methods

- Algorithms that return point forecasts.

3.2.2.2 Models

- Generate same point forecasts but can also generate forecast distributions.
- A stochastic (or random) data generating process that can generate an entire forecast distribution.
- Allow for "proper" model selection.

3.2.3 ETS(A,N,N): A model for SES

Component form

- Forecast equation: $\hat{y}_{t+h|t} = \ell_t$
- Smoothing equation: $\ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}$

Forecast error: $e_t = y_t - \hat{y}_{t|t-1} = y_t - \ell_{t-1}$

Error correction form

$$y_t = \ell_{t-1} + e_t$$
$$\ell_t = \ell_{t-1} + \alpha(y_t - \ell_{t-1})$$

$\ell_t = \ell_{t-1} + \alpha e_t$

Specify probability distribution for e_t, we assume $e_t = \varepsilon_t \sim \text{NID}(0, \sigma^2)$.

3.2.4 ETS(A,N,N)

- Measurement equation: $y_t = \ell_{t-1} + \varepsilon_t$
- State equation: $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$

where $\varepsilon_t \sim \text{NID}(0, \sigma^2)$.

- "innovations" or "single source of error" because equations have the same error process, ε_t.
  - Measurement equation: relationship between observations and states.
  - State equation(s): evolution of the state(s) through time.

3.2.5 ETS(M,N,N)

SES with multiplicative errors.

- Specify relative errors $\varepsilon_t = \frac{y_t - \hat{y}_{t|t-1}}{\hat{y}_{t|t-1}} \sim \text{NID}(0, \sigma^2)$
  - Substituting $\hat{y}_{t|t-1} = \ell_{t-1}$ gives:
    * $y_t = \ell_{t-1} + \ell_{t-1}\varepsilon_t$
    * $e_t = y_t - \hat{y}_{t|t-1} = \ell_{t-1}\varepsilon_t$
- Measurement equation: $y_t = \ell_{t-1}(1 + \varepsilon_t)$
- State equation: $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$
- Models with additive and multiplicative errors with the same parameters generate the same point forecasts but different prediction intervals.

3.2.6 ETS(A,N,N): Specifying the model

```
ETS(y ~ error("A") + trend("N") + season("N"))
```

By default, an optimal value for α and ℓ_0 is used.

α can be chosen manually in `trend()`.

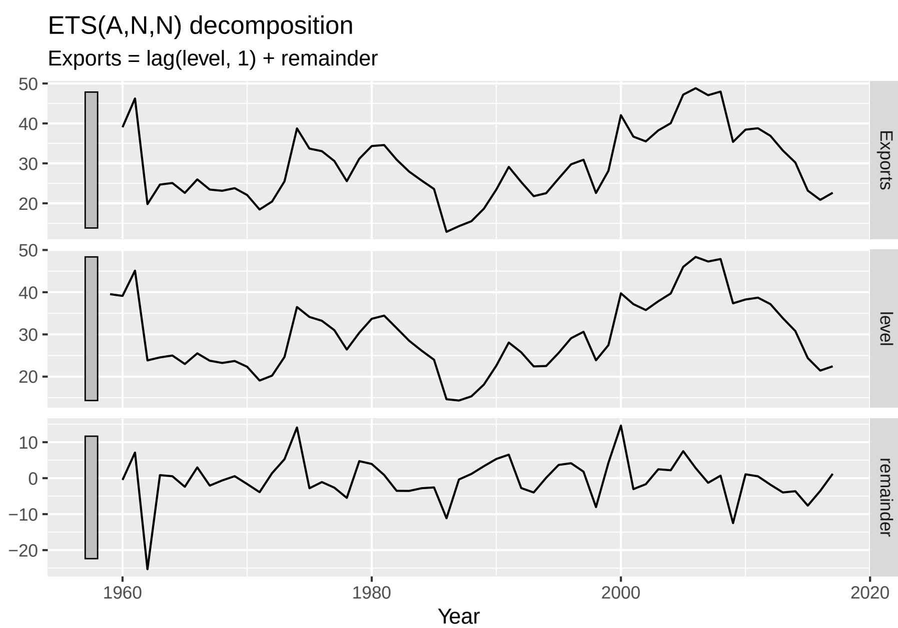
```
trend("N", alpha = 0.5)
trend("N", alpha_range = c(0.2, 0.8))
```

3.2.7 Example: Algerian Exports

```
algeria_economy <- global_economy %>%
  filter(Country == "Algeria")
fit <- algeria_economy %>%
  model(ANN = ETS(Exports ~ error("A") + trend("N") + season("N")))
report(fit)
```

```
## Series: Exports
## Model: ETS(A,N,N)
##   Smoothing parameters:
##     alpha = 0.84
##
##   Initial states:
##      l
##  39.54
##
##   sigma^2:  35.63
##
##   AIC  AICc   BIC
## 446.7 447.2 452.9
```

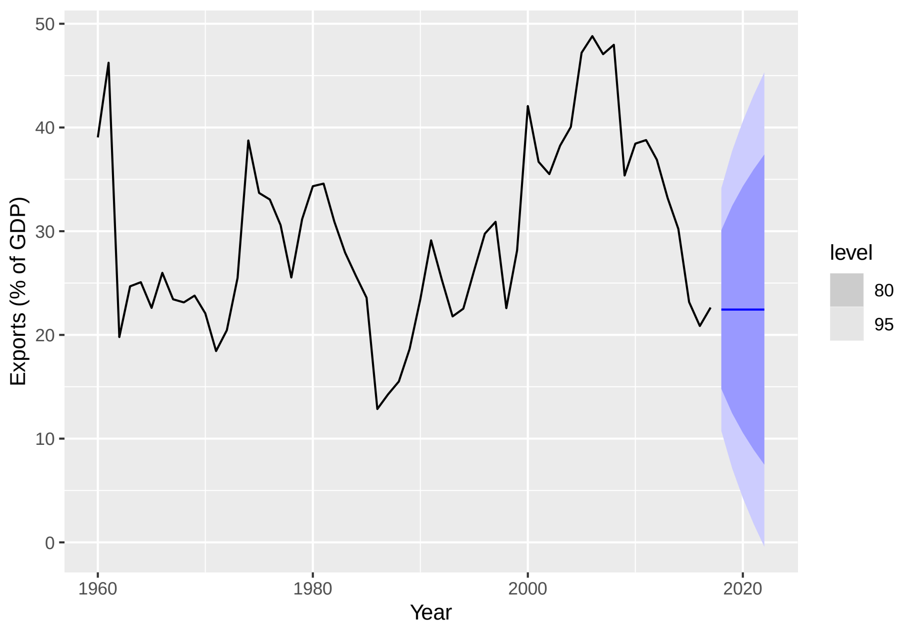
```
components(fit) %>% autoplot()
```




```r
components(fit) %>%
  left_join(fitted(fit), by = c("Country", ".model", "Year"))
```

```
## # A dable:                   59 x 7 [1Y]
## # Key:                       Country, .model [1]
## # ETS(A,N,N) Decomposition: Exports = lag(level, 1) +
## #   remainder
##    Country .model  Year Exports level remainder .fitted
##    <fct>   <chr>  <dbl>   <dbl> <dbl>     <dbl>   <dbl>
##  1 Algeria ANN     1959    NA    39.5    NA        NA
##  2 Algeria ANN     1960    39.0  39.1    -0.496    39.5
##  3 Algeria ANN     1961    46.2  45.1     7.12     39.1
##  4 Algeria ANN     1962    19.8  23.8   -25.3      45.1
##  5 Algeria ANN     1963    24.7  24.6     0.841    23.8
##  6 Algeria ANN     1964    25.1  25.0     0.534    24.6
##  7 Algeria ANN     1965    22.6  23.0    -2.39     25.0
##  8 Algeria ANN     1966    26.0  25.5     3.00     23.0
##  9 Algeria ANN     1967    23.4  23.8    -2.07     25.5
## 10 Algeria ANN     1968    23.1  23.2    -0.630    23.8
## # ... with 49 more rows
```

```r
fit %>%
  forecast(h = 5) %>%
  autoplot(algeria_economy) +
  ylab("Exports (% of GDP)") + xlab("Year")
```



3.3 Models with trend

3.3.1 Holt's linear trend

Component form

- Forecast $\hat{y}_{t+h|t} = \ell_t + hb_t$
- Level $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1})$
- Trend $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$
- Two smoothing parameters α and β^* $(0 \leq \alpha, \beta^* \leq 1)$.
- ℓ_t level: weighted average between y_t and one-step ahead forecast for time t, $(\ell_{t-1} + b_{t-1} = \hat{y}_{t|t-1})$
- b_t slope: weighted average of $(\ell_t - \ell_{t-1})$ and b_{t-1}, current and previous estimate of slope.
- Choose $\alpha, \beta^*, \ell_0, b_0$ to minimise SSE.

3.3.2 ETS(A,A,N)

Holt's linear method with additive errors.

- Assume $\varepsilon_t = y_t - \ell_{t-1} - b_{t-1} \sim \text{NID}(0, \sigma^2)$.

- Substituting into the error correction equations for Holt's linear method

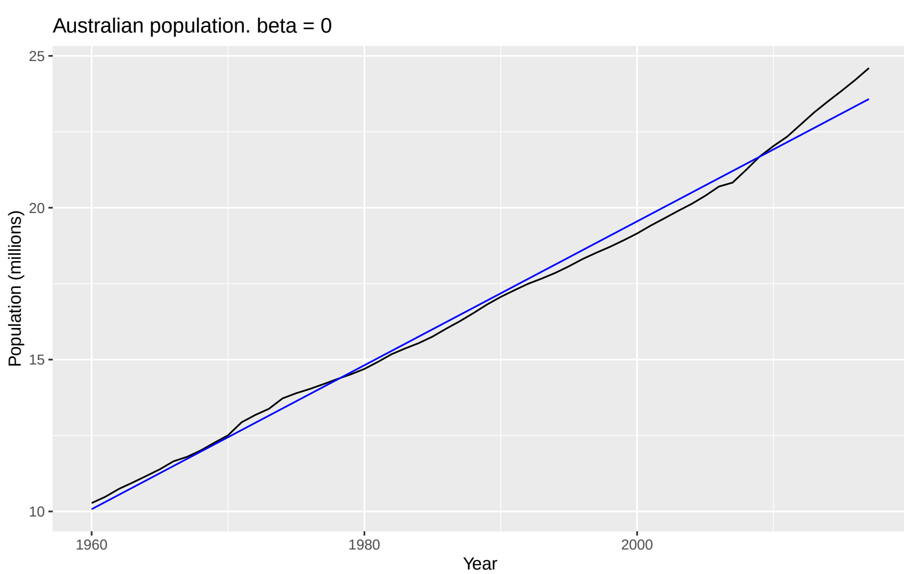
$$y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$$

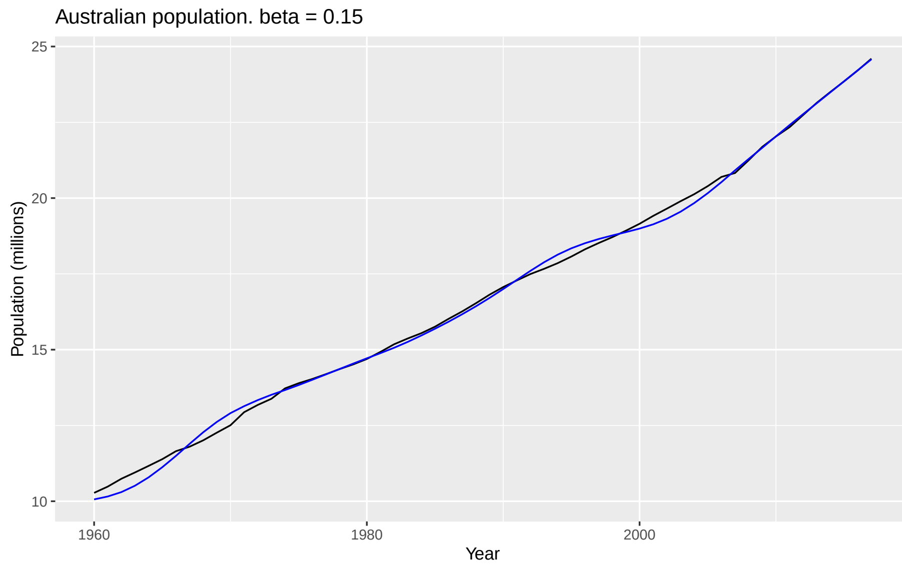
$$\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$$

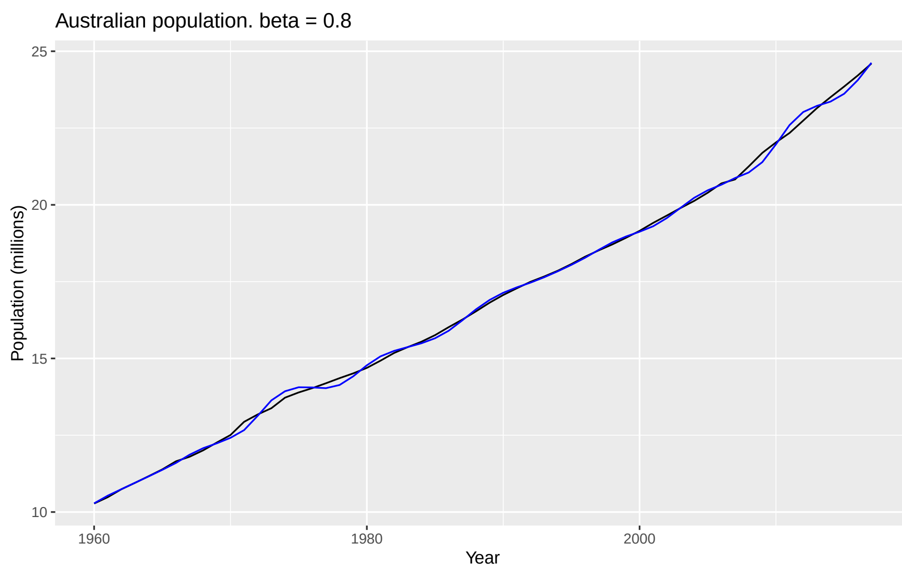
$$b_t = b_{t-1} + \alpha\beta^*\varepsilon_t$$

- For simplicity, set $\beta = \alpha\beta^*$.

3.3.3 Exponential smoothing: trend/slope







3.3.4 ETS(M,A,N)

Holt's linear method with multiplicative errors.

- Assume $\varepsilon_t = \frac{y_t - (\ell_{t-1} + b_{t-1})}{(\ell_{t-1} + b_{t-1})}$
- Following a similar approach as above, the innovations state space model underlying Holt's linear method with multiplicative errors is specified as

$$\begin{aligned}
y_t &= (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t) \\
\ell_t &= (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t) \\
b_t &= b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t
\end{aligned}$$

where again $\beta = \alpha\beta^*$ and $\varepsilon_t \sim \text{NID}(0, \sigma^2)$.

3.3.5 ETS(A,A,N): Specifying the model

```
ETS(y ~ error("A") + trend("A") + season("N"))
```

By default, optimal values for β and b_0 are used.

β can be chosen manually in `trend()`.

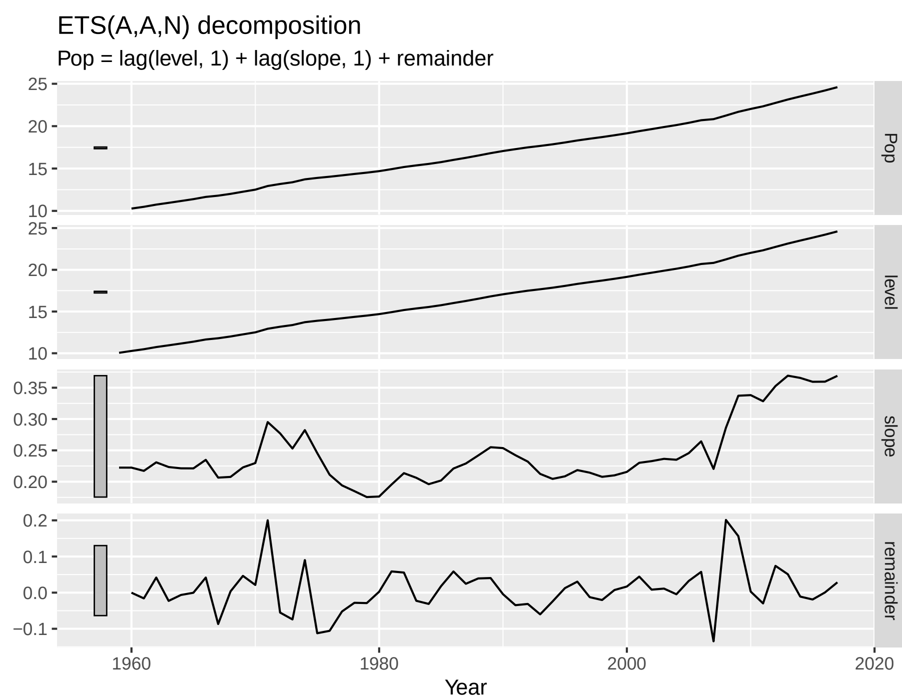
```
trend("A", beta = 0.004)
trend("A", beta_range = c(0, 0.1))
```

3.3.6 Example: Australian population

```
aus_economy <- global_economy %>% filter(Code == "AUS") %>%
  mutate(Pop = Population/1e6)
fit <- aus_economy %>%
  model(AAN = ETS(Pop ~ error("A") + trend("A") + season("N")))
report(fit)
```

```
## Series: Pop
## Model: ETS(A,A,N)
##   Smoothing parameters:
##     alpha = 0.9999
##     beta  = 0.3266
##
##   Initial states:
##      l      b
##  10.05 0.2225
##
##   sigma^2:  0.0041
##
##    AIC   AICc    BIC
## -76.99 -75.83 -66.68
```

```
components(fit) %>% autoplot()
```

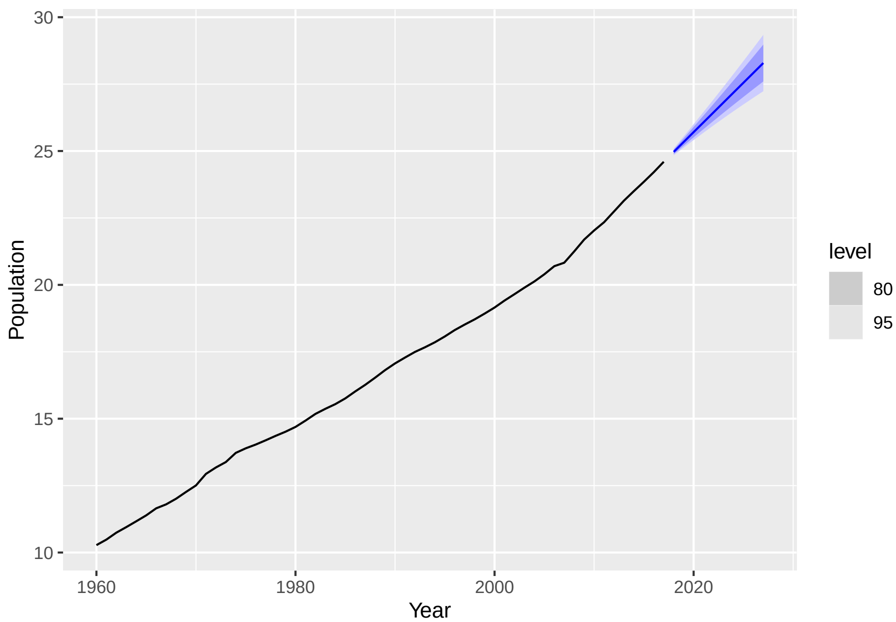



```r
components(fit) %>%
  left_join(fitted(fit), by = c("Country", ".model", "Year"))
```

```
## # A dable:                    59 x 8 [1Y]
## # Key:                        Country, .model [1]
## # ETS(A,A,N) Decomposition: Pop = lag(level, 1) +
## #   lag(slope, 1) + remainder
##    Country  .model  Year   Pop level slope remainder .fitted
##    <fct>    <chr>  <dbl> <dbl> <dbl> <dbl>     <dbl>   <dbl>
##  1 Austral~ AAN     1959  NA    10.1 0.222 NA           NA
##  2 Austral~ AAN     1960  10.3  10.3 0.222 -0.000145    10.3
##  3 Austral~ AAN     1961  10.5  10.5 0.217 -0.0159      10.5
##  4 Austral~ AAN     1962  10.7  10.7 0.231  0.0418      10.7
##  5 Austral~ AAN     1963  11.0  11.0 0.223 -0.0229      11.0
##  6 Austral~ AAN     1964  11.2  11.2 0.221 -0.00641     11.2
##  7 Austral~ AAN     1965  11.4  11.4 0.221 -0.000314    11.4
##  8 Austral~ AAN     1966  11.7  11.7 0.235  0.0418      11.6
##  9 Austral~ AAN     1967  11.8  11.8 0.206 -0.0869      11.9
## 10 Austral~ AAN     1968  12.0  12.0 0.208  0.00350     12.0
## # ... with 49 more rows
```

```r
fit %>%
  forecast(h = 10) %>%
```

```
autoplot(aus_economy) +
ylab("Population") + xlab("Year")
```



3.3.7 Damped trend method

Component form

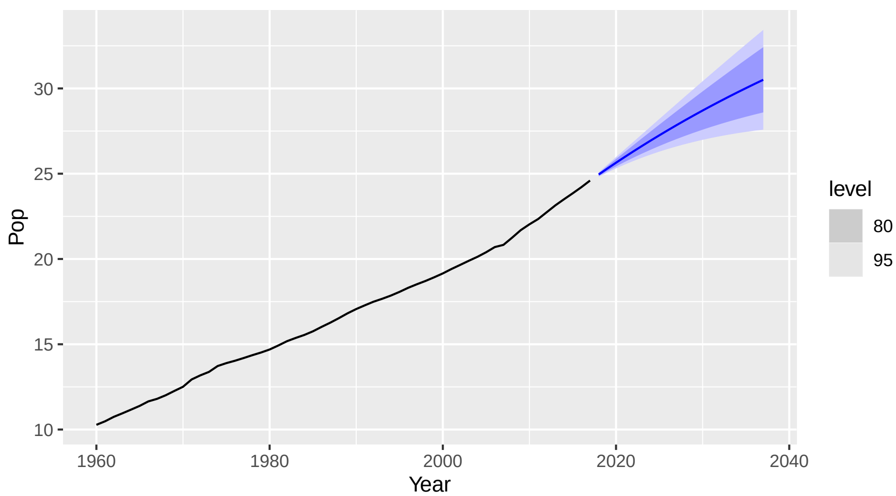
$$\hat{y}_{t+h|t} = \ell_t + (\phi + \phi^2 + \cdots + \phi^h)b_t$$
$$\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$$
$$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}.$$

- Damping parameter $0 < \phi < 1$.
- If $\phi = 1$, identical to Holt's linear trend.
- As $h \to \infty$, $\hat{y}_{T+h|T} \to \ell_T + \phi b_T/(1-\phi)$.
- Short-run forecasts trended, long-run forecasts constant.

3.3.8 Example: Australian population

- Write down the model for ETS(A,A_d,N)

```
aus_economy %>%
 model(holt = ETS(Pop ~ error("A") + trend("Ad") + season("N"))) %>%
 forecast(h = 20) %>%
 autoplot(aus_economy)
```




```
fit <- aus_economy %>%
  filter(Year <= 2010) %>%
  model(
    ses = ETS(Pop ~ error("A") + trend("N") + season("N")),
    holt = ETS(Pop ~ error("A") + trend("A") + season("N")),
    damped = ETS(Pop ~ error("A") + trend("Ad") + season("N"))
  )
```

```
tidy(fit)
accuracy(fit)
```

term	SES	Linear trend	Damped trend
alpha	1.00	1.00	1.00
beta^*		0.30	0.40
phi			0.98
l_0	10.28	10.05	10.04
b_0		0.22	0.25
Training RMSE	0.24	0.06	0.07
Test RMSE	1.63	0.15	0.21
Test MASE	6.18	0.55	0.75
Test MAPE	6.09	0.55	0.74
Test MAE	1.45	0.13	0.18

3.4 Models with seasonality

3.4.1 Holt-Winters additive method

Holt and Winters extended Holt's method to capture seasonality.

Component form

$$
\begin{aligned}
\hat{y}_{t+h|t} &= \ell_t + hb_t + s_{t+h-m(k+1)} \\
\ell_t &= \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}
\end{aligned}
$$

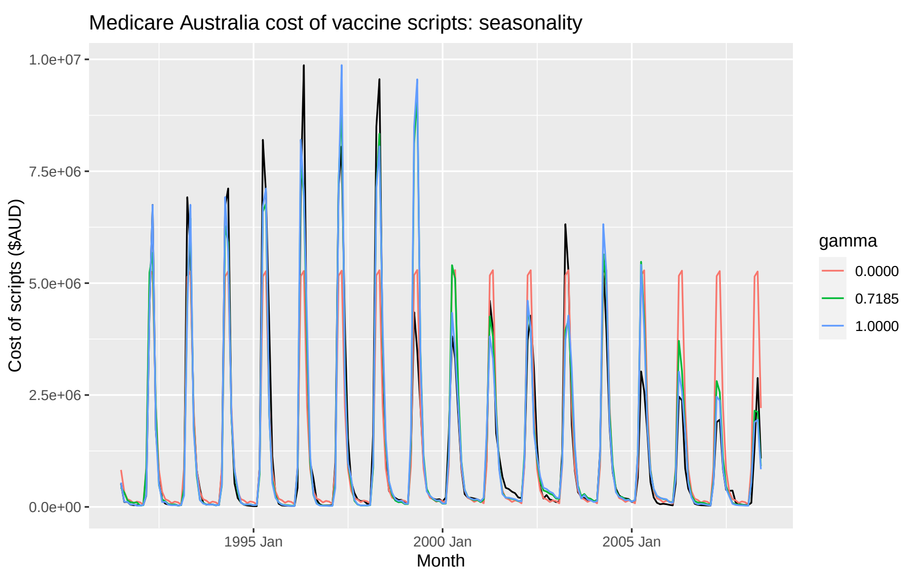
- $k =$ integer part of $(h-1)/m$. Ensures estimates from the final year are used for forecasting.
    - Parameters: $0 \le \alpha \le 1$, $0 \le \beta^* \le 1$, $0 \le \gamma \le 1-\alpha$ and $m =$ period of seasonality (e.g. $m = 4$ for quarterly data).
- Seasonal component is usually expressed as

$$s_t = \gamma^*(y_t - \ell_t) + (1-\gamma^*)s_{t-m}.$$

- Substitute in for ℓ_t:

$$s_t = \gamma^*(1-\alpha)(y_t - \ell_{t-1} - b_{t-1}) + [1-\gamma^*(1-\alpha)]s_{t-m}$$

- We set $\gamma = \gamma^*(1-\alpha)$.
- The usual parameter restriction is $0 \le \gamma^* \le 1$, which translates to $0 \le \gamma \le (1-\alpha)$.



3.4.2 ETS(A,A,A)

Holt-Winters additive method with additive errors.

- Forecast equation} $\hat{y}_{t+h|t} = \ell_t + hb_t + s_{t+h-m(k+1)}$
- Observation equation} $y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$
- State equations}

$$\begin{aligned} \ell_t &= \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t \\ s_t &= s_{t-m} + \gamma\varepsilon_t \end{aligned}$$

- Forecast errors: $\varepsilon_t = y_t - \hat{y}_{t|t-1}$
- k is integer part of $(h-1)/m$

Activity

- Write down the model for ETS(A,N,A)

3.4.3 Holt-Winters multiplicative method

For when seasonal variations are changing proportional to the level of the series.

Component form

$$\hat{y}t + ht = (\ell_t + hb_t)s_{t+h-m(k+1)}$$
$$\ell_t = \alpha\frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1})$$

b_{t} &=

$$\beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$$
$$s_t = \gamma\frac{y_t}{(\ell_{t-1} + b_{t-1})} + (1-\gamma)s_{t-m}$$

- k is integer part of $(h-1)/m$.
- With additive method s_t is in absolute terms: within each year $\sum_i s_i \approx 0$.
- With multiplicative method s_t is in relative terms: within each year $\sum_i s_i \approx m$.

3.4.4 ETS(M,A,M)

Holt-Winters multiplicative method with multiplicative errors.

- Forecast equation $\hat{y}_{t+h|t} = (\ell_t + hb_t)s_{t+h-m(k+1)}$
- Observation equation $y_t = (\ell_{t-1} + b_{t-1})s_{t-m}(1 + \varepsilon_t)$

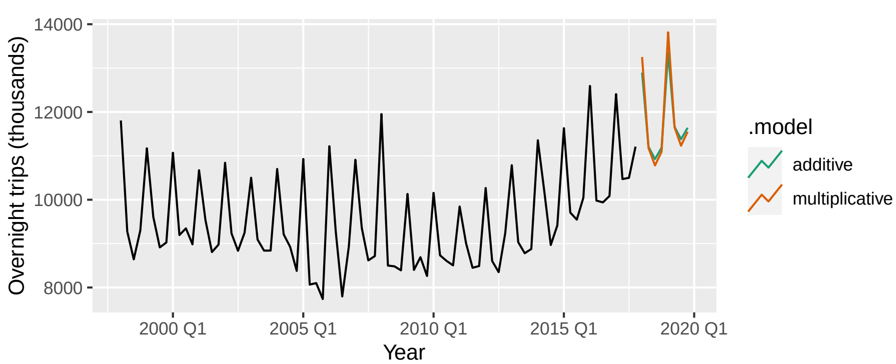
- State equations

$$\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$$
$$b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$$
$$s_t = s_{t-m}(1 + \gamma\varepsilon_t)$$

- Forecast errors: $\varepsilon_t = (y_t - \hat{y}_{t|t-1})/\hat{y}_{t|t-1}$
- k is integer part of $(h-1)/m$.

3.4.5 Example: Australian holiday tourism

```
aus_holidays <- tourism %>%
  filter(Purpose == "Holiday") %>%
  summarise(Trips = sum(Trips))
fit <- aus_holidays %>%
  model(
    additive = ETS(Trips ~ error("A") + trend("A") + season("A")),
    multiplicative = ETS(Trips ~ error("M") + trend("A") + season("M"))
  )
fc <- fit %>% forecast()
```

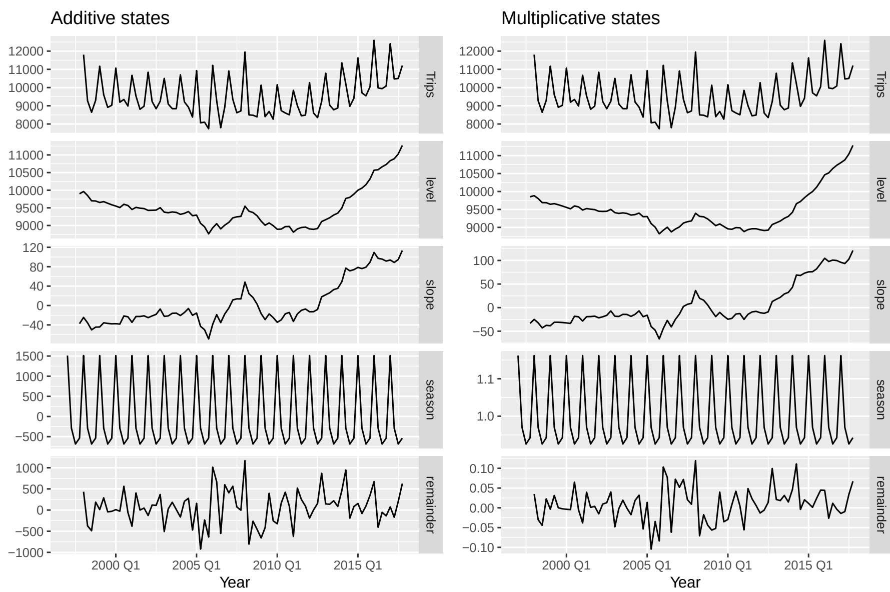


Estimated components

```
components(fit)
```

```
## # A dable:                              168 x 7 [1Q]
## # Key:                                  .model [2]
## # ETS(A,A,A) & ETS(M,A,M) Decomposition: Trips = lag(level,
## #   1) + lag(slope, 1) + lag(season, 4) + remainder
##    .model   Quarter  Trips level slope season remainder
##    <chr>      <qtr>  <dbl> <dbl> <dbl>  <dbl>     <dbl>
##  1 additive 1997 Q1    NA    NA    NA   1512.       NA
##  2 additive 1997 Q2    NA    NA    NA   -290.       NA
##  3 additive 1997 Q3    NA    NA    NA   -684.       NA
```

```
## 4 additive 1997 Q4    NA  9899. -37.4  -538.      NA
## 5 additive 1998 Q1 11806. 9964. -24.5  1512.     433.
## 6 additive 1998 Q2  9276. 9851. -35.6  -290.    -374.
## 7 additive 1998 Q3  8642. 9700. -50.2  -684.    -489.
## 8 additive 1998 Q4  9300. 9694. -44.6  -538.     188.
## 9 additive 1999 Q1 11172. 9652. -44.3  1512.      10.7
## 10 additive 1999 Q2  9608. 9676. -35.6  -290.     290.
## # ... with 158 more rows
```



3.4.6 Holt-Winters damped method

Often the single most accurate forecasting method for seasonal data:

$$\begin{aligned}
\hat{y}_{t+h|t} &= [\ell_t + (\phi + \phi^2 + \dots + \phi^h)b_t]s_{t+h-m(k+1)} \\
\ell_t &= \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1} \\
s_t &= \gamma \frac{y_t}{(\ell_{t-1} + \phi b_{t-1})} + (1-\gamma)s_{t-m}
\end{aligned}$$

3.5 Innovations state space models

3.5.1 Exponential smoothing methods

		Seasonal Component		
	Trend Component	N (None)	A (Additive)	M (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)

- (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): Damped multiplicative Holt-Winters' method

There are also multiplicative trend methods (not recommended).

3.5.2 ETS models

Additive Error

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

Multiplicative Error

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

3.5.3 Additive error models

Trend	Seasonal N	Seasonal A	Seasonal M
N	$y_t = \ell_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$	$y_t = \ell_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/\ell_{t-1}$
A	$y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + b_{t-1})$
A$_\mathbf{d}$	$y_t = \ell_{t-1} + \phi b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + \phi b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = \phi b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + \phi b_{t-1})$

3.5.4 Multiplicative error models

Trend	Seasonal N	Seasonal A	Seasonal M
N	$y_t = \ell_{t-1}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$	$y_t = (\ell_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \alpha(\ell_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
A	$y_t = (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
A$_\mathbf{d}$	$y_t = (\ell_{t-1} + \phi b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$

3.5.5 Estimating ETS models

- Smoothing parameters α, β, γ and ϕ, and the initial states ℓ_0, b_0, $s_0, s_{-1}, \dots, s_{-m+1}$ are estimated by maximising the "likelihood" = the probability of the data arising from the specified model.
- For models with additive errors equivalent to minimising SSE.
- For models with multiplicative errors, **not** equivalent to minimising SSE.

3.5.6 Innovations state space models

Let $\mathbf{x}_t = (\ell_t, b_t, s_t, s_{t-1}, \dots, s_{t-m+1})$ and $\varepsilon_t \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$.

$$
\begin{aligned}
y_t &= \underbrace{h(\mathbf{x}_{t-1})}_{\mu_t} + \underbrace{k(\mathbf{x}_{t-1})\varepsilon_t}_{e_t} \\
\mathbf{x}_t &= f(\mathbf{x}_{t-1}) + g(\mathbf{x}_{t-1})\varepsilon_t
\end{aligned}
$$

Additive errors

$$k(x) = 1. \qquad y_t = \mu_t + \varepsilon_t.$$

Multiplicative errors

$$k(\mathbf{x}_{t-1}) = \mu_t. \qquad y_t = \mu_t(1 + \varepsilon_t).$$
$$\varepsilon_t = (y_t - \mu_t)/\mu_t \text{ is relative error.}$$

3.5.7 Innovations state space models

Estimation

$$
\begin{aligned}
L^*(\ , \mathbf{x}_0) &= T \log\left(\sum_{t=1}^{T} \varepsilon_t^2\right) + 2\sum_{t=1}^{T} \log|k(\mathbf{x}_{t-1})| \\
&= -2\log(\text{Likelihood}) + \text{constant}
\end{aligned}
$$

- Estimate parameters $\ = (\alpha, \beta, \gamma, \phi)$ and initial states $\mathbf{x}_0 = (\ell_0, b_0, s_0, s_{-1}, \dots, s_{-m+1})$ by minimizing L^*.

3.5.8 Parameter restrictions

3.5.8.1 *Usual* region

- Traditional restrictions in the methods $0 < \alpha, \beta^*, \gamma^*, \phi < 1$ (equations interpreted as weighted averages).
- In models we set $\beta = \alpha\beta^*$ and $\gamma = (1-\alpha)\gamma^*$.
- Therefore $0 < \alpha < 1, \quad 0 < \beta < \alpha \quad$ and $0 < \gamma < 1 - \alpha$.
- $0.8 < \phi < 0.98$ — to prevent numerical difficulties.

3.5.8.2 *Admissible* region

- To prevent observations in the distant past having a continuing effect on current forecasts.
- Usually (but not always) less restrictive than the*traditional* region.
- For example for ETS(A,N,N):
  traditional $0 < \alpha < 1$ — *admissible* is $0 < \alpha < 2$.

3.5.9 Model selection

Akaike's Information Criterion

$$\text{AIC} = -2\log(\text{L}) + 2k$$

where L is the likelihood and k is the number of parameters initial states estimated in the model.

Corrected AIC

$$\text{AIC}_{\text{c}} = \text{AIC} + \frac{2(k+1)(k+2)}{T-k}$$

which is the AIC corrected (for small sample bias).

Bayesian Information Criterion

$$\text{BIC} = \text{AIC} + k(\log(T) - 2).$$

3.5.10 AIC and cross-validation

Minimizing the AIC assuming Gaussian residuals is asymptotically equivalent to minimizing one-step time series cross validation MSE.

3.5.11 Automatic forecasting

From Hyndman et al. (IJF, 2002):

- Apply each model that is appropriate to the data. Optimize parameters and initial values using MLE (or some other criterion).
- Select best method using AICc:
- Produce forecasts using best method.
- Obtain forecast intervals using underlying state space model.

Method performed very well in M3 competition.

3.5.12 Example: National populations

```
fit <- global_economy %>%
  mutate(Pop = Population / 1e6) %>%
  model(ets = ETS(Pop))
fit
```

```
## # A mable: 263 x 2
## # Key:     Country [263]
##    Country                      ets
##    <fct>                    <model>
##  1 Afghanistan         <ETS(A,A,N)>
##  2 Albania             <ETS(M,A,N)>
##  3 Algeria             <ETS(M,A,N)>
##  4 American Samoa      <ETS(M,A,N)>
##  5 Andorra             <ETS(M,A,N)>
##  6 Angola              <ETS(M,A,N)>
##  7 Antigua and Barbuda <ETS(M,A,N)>
##  8 Arab World          <ETS(M,A,N)>
##  9 Argentina           <ETS(A,A,N)>
## 10 Armenia             <ETS(M,A,N)>
## # ... with 253 more rows
```

```
fit %>%
  forecast(h = 5)
```

```
## # A fable: 1,315 x 5 [1Y]
## # Key:     Country, .model [263]
##    Country     .model  Year             Pop .mean
##    <fct>       <chr>  <dbl>          <dist> <dbl>
##  1 Afghanistan ets     2018    N(36, 0.012) 36.4
##  2 Afghanistan ets     2019    N(37, 0.059) 37.3
##  3 Afghanistan ets     2020     N(38, 0.16) 38.2
##  4 Afghanistan ets     2021     N(39, 0.35) 39.0
##  5 Afghanistan ets     2022     N(40, 0.64) 39.9
##  6 Albania     ets     2018 N(2.9, 0.00012)  2.87
##  7 Albania     ets     2019   N(2.9, 6e-04)  2.87
##  8 Albania     ets     2020  N(2.9, 0.0017)  2.87
##  9 Albania     ets     2021  N(2.9, 0.0036)  2.86
## 10 Albania     ets     2022  N(2.9, 0.0066)  2.86
## # ... with 1,305 more rows
```

3.5.13 Example: Australian holiday tourism

```
holidays <- tourism %>%
  filter(Purpose == "Holiday")
fit <- holidays %>% model(ets = ETS(Trips))
fit
```

```
## # A mable: 76 x 4
## # Key:     Region, State, Purpose [76]
##    Region                    State           Purpose           ets
##    <chr>                     <chr>           <chr>         <model>
```

```
##  1 Adelaide              South Austral~ Holiday <ETS(A,N,A)>
##  2 Adelaide Hills        South Austral~ Holiday <ETS(A,A,N)>
##  3 Alice Springs         Northern Terr~ Holiday <ETS(M,N,A)>
##  4 Australia's Coral Co~ Western Austr~ Holiday <ETS(M,N,A)>
##  5 Australia's Golden O~ Western Austr~ Holiday <ETS(M,N,M)>
##  6 Australia's North We~ Western Austr~ Holiday <ETS(A,N,A)>
##  7 Australia's South We~ Western Austr~ Holiday <ETS(M,N,M)>
##  8 Ballarat              Victoria       Holiday <ETS(M,N,A)>
##  9 Barkly                Northern Terr~ Holiday <ETS(A,N,A)>
## 10 Barossa               South Austral~ Holiday <ETS(A,N,N)>
## # ... with 66 more rows
```

```
fit %>%
  filter(Region == "Snowy Mountains") %>%
  report()
```

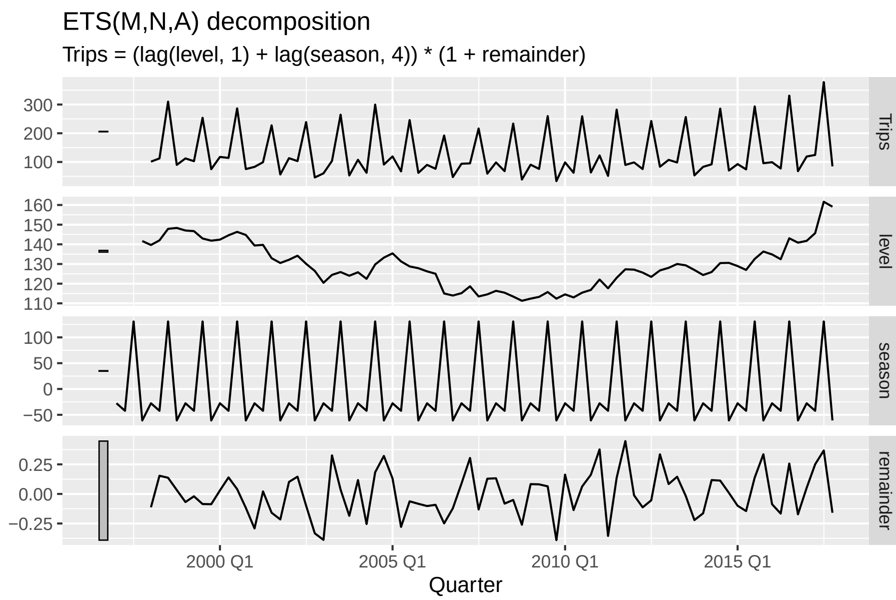
```
## Series: Trips
## Model: ETS(M,N,A)
##   Smoothing parameters:
##     alpha = 0.1571
##     gamma = 0.0001001
##
##   Initial states:
##      l     s1    s2     s3     s4
##  141.7 -60.96 130.9 -42.24 -27.66
##
##   sigma^2:  0.0388
##
##   AIC  AICc   BIC
## 852.0 853.6 868.7
```

```
fit %>%
  filter(Region == "Snowy Mountains") %>%
  components(fit)
```

```
## # A dable:                  84 x 9 [1Q]
## # Key:                      Region, State, Purpose, .model
## #   [1]
## # ETS(M,N,A) Decomposition: Trips = (lag(level, 1) +
## #   lag(season, 4)) * (1 + remainder)
##    Region State Purpose .model Quarter Trips level season
##    <chr>  <chr> <chr>   <chr>    <qtr> <dbl> <dbl>  <dbl>
##  1 Snowy~ New ~ Holiday ets    1997 Q1  NA      NA   -27.7
##  2 Snowy~ New ~ Holiday ets    1997 Q2  NA      NA   -42.2
##  3 Snowy~ New ~ Holiday ets    1997 Q3  NA      NA   131.
##  4 Snowy~ New ~ Holiday ets    1997 Q4  NA     142.  -61.0
##  5 Snowy~ New ~ Holiday ets    1998 Q1 101.    140.  -27.7
##  6 Snowy~ New ~ Holiday ets    1998 Q2 112.    142.  -42.2
```

```
##  7 Snowy~ New ~ Holiday ets    1998 Q3 310.    148.   131.
##  8 Snowy~ New ~ Holiday ets    1998 Q4  89.8   148.   -61.0
##  9 Snowy~ New ~ Holiday ets    1999 Q1 112.    147.   -27.7
## 10 Snowy~ New ~ Holiday ets    1999 Q2 103.    147.   -42.2
## # ... with 74 more rows, and 1 more variable:
## #   remainder <dbl>
```

```
fit %>%
  filter(Region == "Snowy Mountains") %>%
  components(fit) %>%
  autoplot()
```

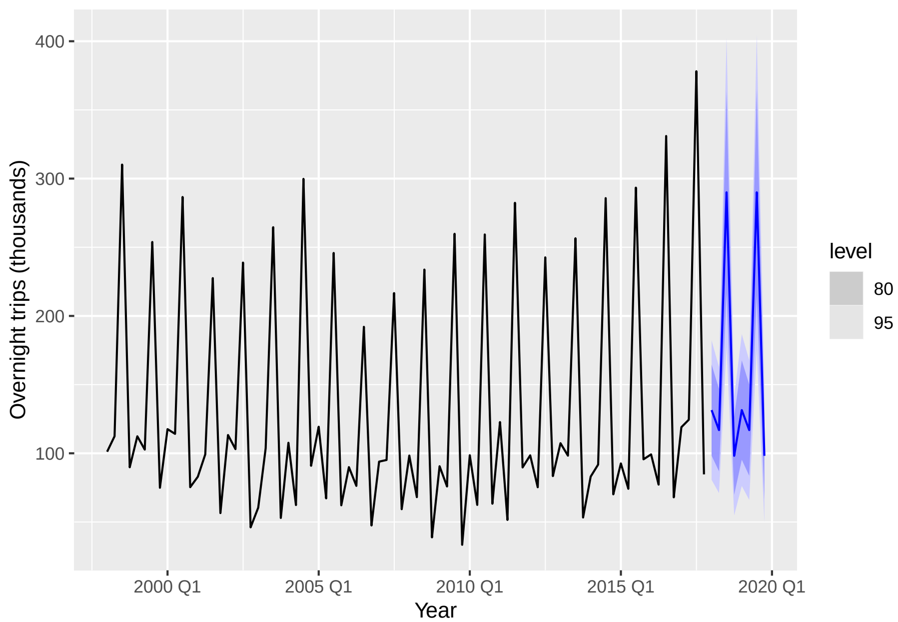



```
fit %>% forecast()
```

```
## # A fable: 608 x 7 [1Q]
## # Key:     Region, State, Purpose, .model [76]
##    Region   State   Purpose .model Quarter       Trips .mean
##    <chr>    <chr>   <chr>   <chr>    <qtr>      <dist> <dbl>
##  1 Adelaide South ~ Holiday ets    2018 Q1 N(210, 457) 210.
##  2 Adelaide South ~ Holiday ets    2018 Q2 N(173, 473) 173.
##  3 Adelaide South ~ Holiday ets    2018 Q3 N(169, 489) 169.
##  4 Adelaide South ~ Holiday ets    2018 Q4 N(186, 505) 186.
##  5 Adelaide South ~ Holiday ets    2019 Q1 N(210, 521) 210.
##  6 Adelaide South ~ Holiday ets    2019 Q2 N(173, 537) 173.
##  7 Adelaide South ~ Holiday ets    2019 Q3 N(169, 553) 169.
##  8 Adelaide South ~ Holiday ets    2019 Q4 N(186, 569) 186.
##  9 Adelaid~ South ~ Holiday ets    2018 Q1   N(19, 36)  19.4
```

```
## 10 Adelaid~ South ~ Holiday ets    2018 Q2   N(20, 36)  19.6
## # ... with 598 more rows
```

```
fit %>%
  forecast() %>%
  filter(Region == "Snowy Mountains") %>%
  autoplot(holidays) +
  xlab("Year") + ylab("Overnight trips (thousands)")
```



3.5.14 Some unstable models

- Some of the combinations of (Error, Trend, Seasonal) can lead to numerical difficulties; see equations with division by a state.
- These are: ETS(A,N,M), ETS(A,A,M), ETS(A,A_d,M).
- Models with multiplicative errors are useful for strictly positive data, but are not numerically stable with data containing zeros or negative values. In that case only the six fully additive models will be applied.

3.5.15 Exponential smoothing models

Additive Error

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

Multiplicative Error

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

3.5.16 Residuals

Response residuals

$$\hat{e}_t = y_t - \hat{y}_{t|t-1}$$

Innovation residuals

Additive error model:

$$\hat{\varepsilon}_t = y_t - \hat{y}_{t|t-1}$$

Multiplicative error model:

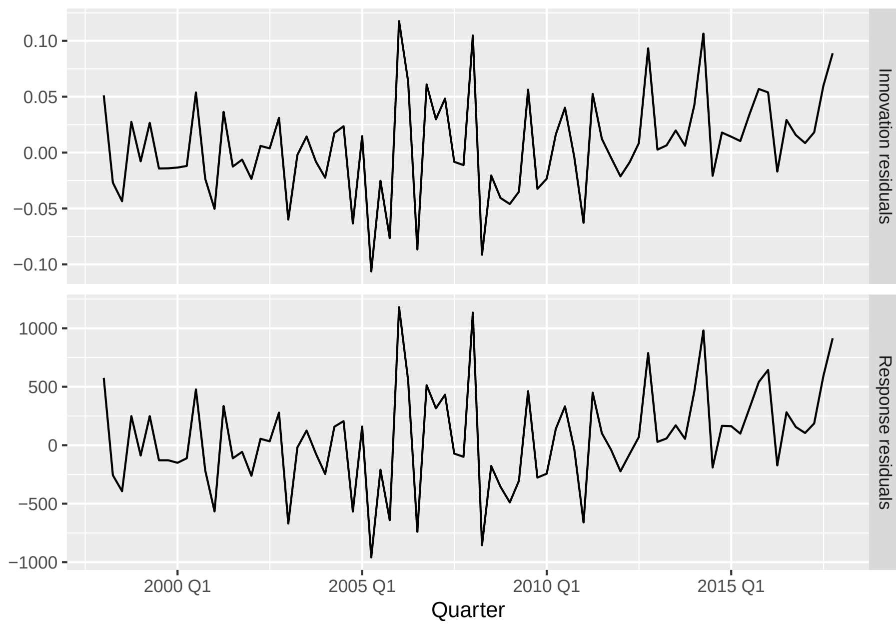
$$\hat{\varepsilon}_t = \frac{y_t - \hat{y}_{t|t-1}}{\hat{y}_{t|t-1}}$$

3.5.17 Example: Australian holiday tourism

```
aus_holidays <- tourism %>%
  filter(Purpose == "Holiday") %>%
  summarise(Trips = sum(Trips))
fit <- aus_holidays %>%
  model(ets = ETS(Trips)) %>%
  report()
```

```
## Series: Trips
## Model: ETS(M,N,M)
##   Smoothing parameters:
##     alpha = 0.3578
##     gamma = 0.0009686
##
##   Initial states:
##     l    s1     s2     s3    s4
##  9667 0.943 0.9268 0.9684 1.162
##
##   sigma^2:  0.0022
##
##  AIC AICc  BIC
## 1331 1333 1348
```

```r
residuals(fit)
residuals(fit, type = "response")
```



3.6 Forecasting with exponential smoothing

3.6.1 Forecasting with ETS models

Point forecasts: iterate the equations for $t = T+1, T+2, \ldots, T+h$ and set all $\varepsilon_t = 0$ for $t > T$.

- Not the same as $\mathrm{E}(y_{t+h}|\mathbf{x}_t)$ unless trend and seasonality are both additive.
- Point forecasts for ETS(A,,) are identical to ETS(M,,) if the parameters are the same.

3.6.2 Example: ETS(A,A,N)

$$
\begin{aligned}
y_{T+1} &= \ell_T + b_T + \varepsilon_{T+1} \\
\hat{y}_{T+1|T} &= \ell_T + b_T \\
y_{T+2} &= \ell_{T+1} + b_{T+1} + \varepsilon_{T+2} \\
&= (\ell_T + b_T + \alpha\varepsilon_{T+1}) + (b_T + \beta\varepsilon_{T+1}) + \varepsilon_{T+2} \\
\hat{y}_{T+2|T} &= \ell_T + 2b_T
\end{aligned}
$$

etc.

3.6.3 Example: ETS(M,A,N)

$$
\begin{aligned}
y_{T+1} &= (\ell_T + b_T)(1 + \varepsilon_{T+1}) \\
\hat{y}_{T+1|T} &= \ell_T + b_T. \\
y_{T+2} &= (\ell_{T+1} + b_{T+1})(1 + \varepsilon_{T+2}) \\
&= \{(\ell_T + b_T)(1 + \alpha\varepsilon_{T+1}) + [b_T + \beta(\ell_T + b_T)\varepsilon_{T+1}]\}(1 + \varepsilon_{T+2}) \\
\hat{y}_{T+2|T} &= \ell_T + 2b_T
\end{aligned}
$$

etc.

3.6.4 Forecasting with ETS models

Prediction intervals: can only generated using the models.

- The prediction intervals will differ between models with additive and multiplicative errors.
- Exact formulae for some models.
- More general to simulate future sample paths, conditional on the last estimate of the states, and to obtain prediction intervals from the percentiles of these simulated future paths.

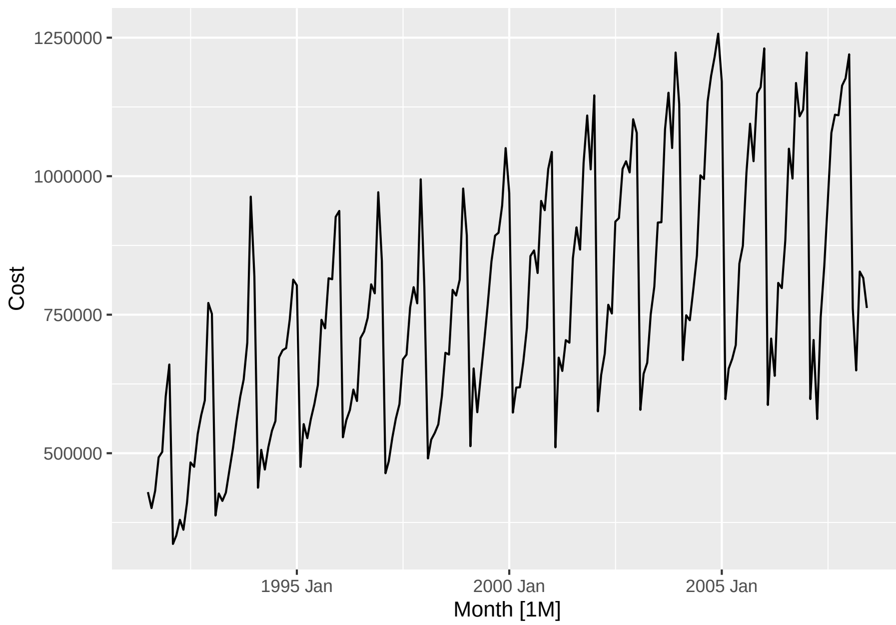
3.6.5 Prediction intervals

PI for most ETS models: $\hat{y}_{T+h|T} \pm c\sigma_h$, where c depends on coverage probability and σ_h is forecast standard deviation.

(A,N,N)	$\sigma_h = \sigma^2\big[1 + \alpha^2(h-1)\big]$
(A,A,N)	$\sigma_h = \sigma^2\Big[1 + (h-1)\big\{\alpha^2 + \alpha\beta h + \frac{1}{6}\beta^2 h(2h-1)\big\}\Big]$
(A,A$_d$,N)	$\sigma_h = \sigma^2\Big[1 + \alpha^2(h-1) + \frac{\beta\phi h}{(1-\phi)^2}\left\{2\alpha(1-\phi) + \beta\phi\right\} - \frac{\beta\phi(1-\phi^h)}{(1-\phi)^2(1-\phi^2)}\left\{2\alpha(1-\phi^2) + \beta\phi(1 + 2\phi - \phi^h)\right\}\Big]$
(A,N,A)	$\sigma_h = \sigma^2\Big[1 + \alpha^2(h-1) + \gamma k(2\alpha + \gamma)\Big]$
(A,A,A)	$\sigma_h = \sigma^2\Big[1 + (h-1)\big\{\alpha^2 + \alpha\beta h + \frac{1}{6}\beta^2 h(2h-1)\big\} + \gamma k\big\{2\alpha + \gamma + \beta m(k+1)\big\}\Big]$
(A,A$_d$,A)	$\sigma_h = \sigma^2\Big[1 + \alpha^2(h-1) + \frac{\beta\phi h}{(1-\phi)^2}\left\{2\alpha(1-\phi) + \beta\phi\right\} - \frac{\beta\phi(1-\phi^h)}{(1-\phi)^2(1-\phi^2)}\left\{2\alpha(1-\phi^2) + \beta\phi(1 + 2\phi - \phi^h)\right\} + \gamma k(2\alpha + \gamma) + \frac{2\beta\gamma\phi}{(1-\phi)(1-\phi^m)}\left\{k(1-\phi^m) - \phi^m(1-\phi^{mk})\right\}\Big]$

3.6.6 Example: Corticosteroid drug sales

```
h02 <- PBS %>%
  filter(ATC2 == "H02") %>%
  summarise(Cost = sum(Cost))
h02 %>%
  autoplot(Cost)
```




```
h02 %>%
  model(ETS(Cost)) %>%
  report()
```

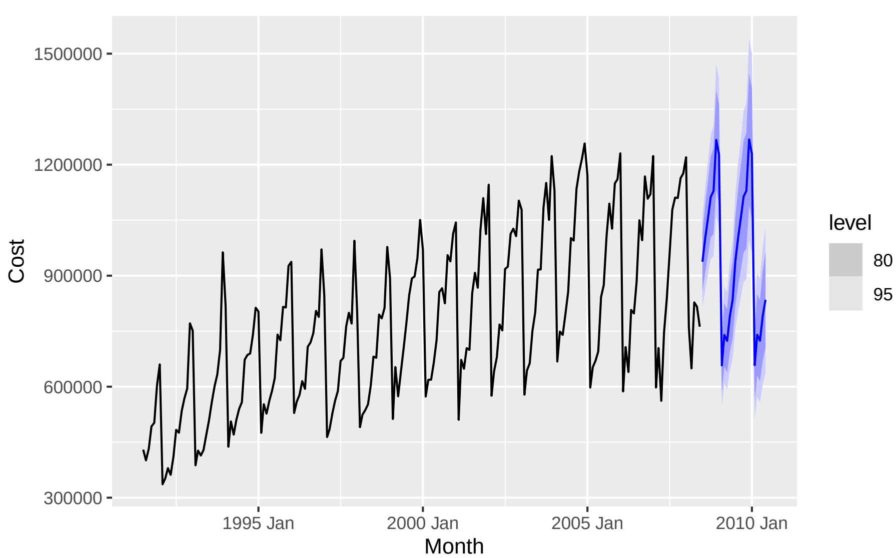
```
## Series: Cost
## Model: ETS(M,Ad,M)
##   Smoothing parameters:
##     alpha = 0.3071
##     beta  = 0.0001007
##     gamma = 0.0001007
##     phi   = 0.9775
##
##   Initial states:
##        l    b     s1    s2     s3     s4     s5    s6    s7
##   417269 8206 0.8717 0.826 0.7563 0.7733 0.6872 1.284 1.325
##     s8    s9   s10   s11    s12
##   1.18 1.164 1.105 1.048 0.9806
```

```
##
##   sigma^2:  0.0046
##
##  AIC AICc  BIC
## 5515 5519 5575
```

```
h02 %>%
  model(ETS(Cost ~ error("A") + trend("A") + season("A"))) %>%
  report()
```

```
## Series: Cost
## Model: ETS(A,A,A)
##   Smoothing parameters:
##     alpha = 0.1702
##     beta  = 0.006311
##     gamma = 0.4546
##
##   Initial states:
##       l    b     s1      s2      s3      s4      s5     s6
##  409706 9097 -99075 -136602 -191496 -174531 -241437 210644
##      s7     s8     s9    s10   s11    s12
##  244644 145368 130570 84458 39132 -11674
##
##   sigma^2:  3.499e+09
##
##  AIC AICc  BIC
## 5585 5589 5642
```

```
h02 %>% model(ETS(Cost)) %>% forecast() %>% autoplot(h02)
```

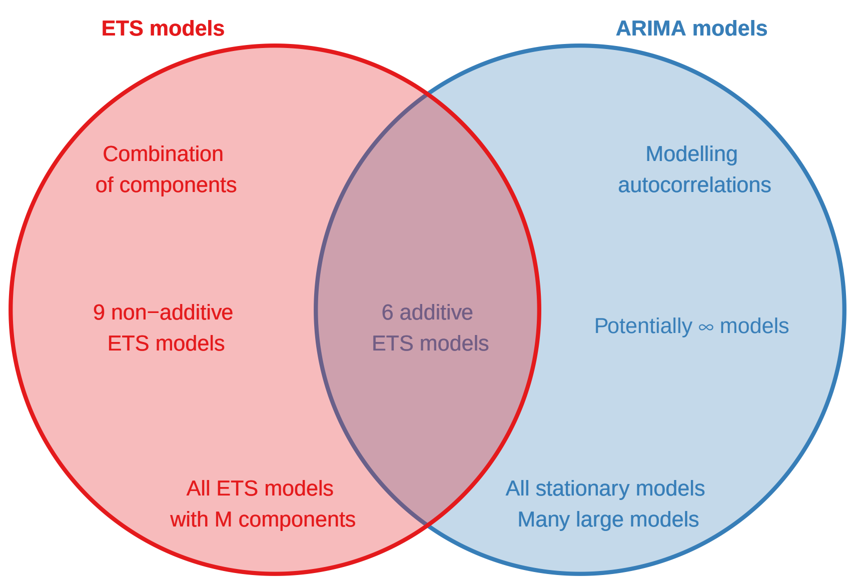


```
h02 %>%
  model(
    auto = ETS(Cost),
    AAA = ETS(Cost ~ error("A") + trend("A") + season("A"))
  ) %>%
  accuracy()
```

Model	ME	MAE	RMSE	MAPE	MASE
auto	2461	38649	51102	4.989	0.6376
AAA	-5780	43378	56784	6.048	0.7156

3.7 ARIMA vs ETS

- Myth that ARIMA models are more general than exponential smoothing.
- Linear exponential smoothing models all special cases of ARIMA models.
- Non-linear exponential smoothing models have no equivalent ARIMA counterparts.
- Many ARIMA models have no exponential smoothing counterparts.
- ETS models all non-stationary. Models with seasonality or non-damped trend (or both) have two unit roots; all other models have one unit root.



3.7.1 Equivalences

ETS model	**ARIMA model**	**Parameters**
ETS(A,N,N)	ARIMA(0,1,1)	$\theta_1 = \alpha - 1$
ETS(A,A,N)	ARIMA(0,2,2)	$\theta_1 = \alpha + \beta - 2$
		$\theta_2 = 1 - \alpha$
ETS(A,A_d,N)	ARIMA(1,1,2)	$\phi_1 = \phi$
		$\theta_1 = \alpha + \phi\beta - 1 - \phi$
		$\theta_2 = (1 - \alpha)\phi$
ETS(A,N,A)	ARIMA(0,0,m)(0,1,0)$_m$	
ETS(A,A,A)	ARIMA(0,1,$m+1$)(0,1,0)$_m$	
ETS(A,A_d,A)	ARIMA(1,0,$m+1$)(0,1,0)$_m$	

3.8 References:

- Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts.

Chapter 4

Volatility Models

This chapter is heavily based on Chapter 12 of Chatfield and Xing (2019) and Chapter 3 of Tsay (2010) .

4.1 Introduction

- Anything that is observed sequentially over time is a time series.

- **Financial time series** analysis focuses on the theory and practice of asset valuation over time.

- In finance, the data can be collected much more frequently – High frequency data.
- Many financial time series also exhibit changing variance and this can have important consequences in formulating financial decisions.

Example: Financial time series

- Typically, when we analyze assets, we look at the percentage change in prices or returns.

```
# Tidy financial analysis
library(tidyquant)

sp500 <- tq_get("^GSPC", from = "1995-01-04", to = "2021-02-25" )
print(sp500)
```

```
## # A tibble: 6,582 x 8
##    symbol date        open  high   low close volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>  <dbl>    <dbl>
```

```
##  1 ^GSPC  1995-01-04  459.  461.  458.  461. 3.20e8     461.
##  2 ^GSPC  1995-01-05  461.  461.  460.  460. 3.09e8     460.
##  3 ^GSPC  1995-01-06  460.  462.  459.  461. 3.08e8     461.
##  4 ^GSPC  1995-01-09  461.  462.  460.  461. 2.79e8     461.
##  5 ^GSPC  1995-01-10  461.  465.  461.  462. 3.52e8     462.
##  6 ^GSPC  1995-01-11  462.  464.  459.  462. 3.46e8     462.
##  7 ^GSPC  1995-01-12  462.  462.  461.  462. 3.13e8     462.
##  8 ^GSPC  1995-01-13  462.  466.  462.  466. 3.37e8     466.
##  9 ^GSPC  1995-01-16  466.  470.  466.  469. 3.16e8     469.
## 10 ^GSPC  1995-01-17  469.  470.  468.  470. 3.32e8     470.
## # ... with 6,572 more rows
```

```
# Convert each assets raw adjusted closing prices to returns
sp500_return <- sp500 %>%
  tq_transmute(select     = adjusted,
               mutate_fun = periodReturn,
               period     = "daily")

sp500_return %>%
  as_tsibble(index = date) %>%
  autoplot(daily.returns) +
  labs(x = "Day", y= "Daily return")
```

- The mean of the return series seems to be stable with an average return of approximately zero.
- The volatility of data changes over time.
- The focus of this chapter is to study some methods and econometric models for modeling the **volatility** (conditional standard deviation) of an asset return.
- These models are referred to as **conditional heteroscedastic models**.
- These models do not generally provide better point forecasts, but provides a better estimates of the (local) variance.
- As a result they allow to compute more reliable prediction intervals and therefore a better assessment of risk.
- Volatility models have many applications in economics and finance.
- This chapter discusses various types of **univariate volatility models**

4.2 Structure of a Model for Asset Returns

- Let, $\{P_T\}$, denotes a time series.
- Let, $\{Y_T\}$, denotes a derived series from which any trend and seasonal effects have been removed and linear (short-term correlations) effects may also have been removed.
    - Examples : Let $\{P_T\}$ be share price at the tth trading day.

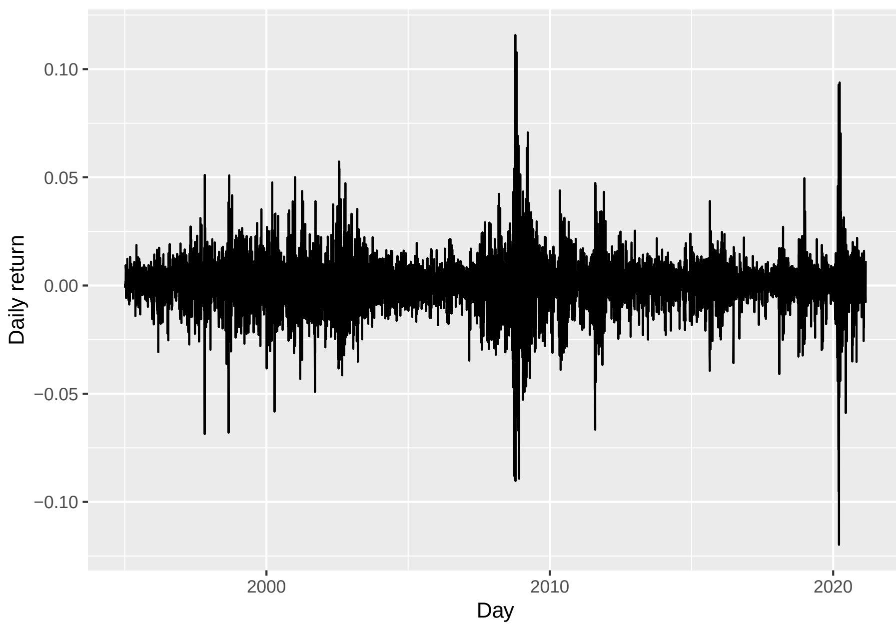


Figure 4.1: Daily returns of the adjusted closing prices of the Standard & Poor's 500 (S&P500) index from January 4, 1995 to February 25, 2021

$$Y_t = logP_t - logP_{t-1} \text{ or } Y_t = \frac{P_t - P_{t-1}}{P_{t-1}} \times 100\%$$

- This is often called the **return** or the **growth rate** of a series.

Example

- Let P_t be the adjusted closing prices of the S&P500 at the tth trading day.
- Let Y_t be the daily returns of the S&P500 Index at each day as shown in Figure 4.1.
- The basic idea in volatility modelling is that the return series $\{Y_t\}$ has very few serial correlations, but it is a dependent series.
- Consider the sample ACFs and PACFs of Y_t, $|Y_t|$ and Y_t^2 (Figure 4.2)

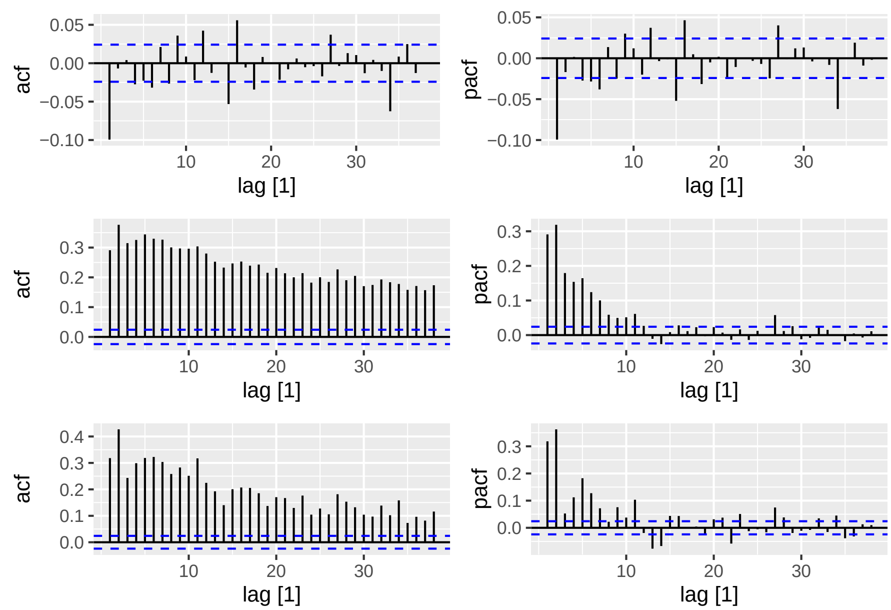


Figure 4.2: Sample ACF (left) and sample PACF (right) of various functions of the daily returns, Y_t, of adjusted closing prices of S&P500 Index from from January 4, 1995 to February 25, 2021. Top: Original series Y_t; Middle: Absolute value of Y_t; Bottom: Squared values of Y_t.

- Sample ACFs of the returns Y_t suggest no significant serial correlations except for small ones at lags 1, 3 and 5.
- However, the sample ACFs of $|Y_t|$ and Y_t^2, show strong dependence over all lags.
- Important feature: the returns may seem serially uncorrelated, but it is dependent.

[4] Prepared by Dr. Priyanga D. Talagala (Copyright 2021 Priyanga D. Talagala)

- This is a common observations for daily returns series

Volatility of a return series

- Let Y_t be the innovations in a linear time series model.
- Let X_t follow an $ARMA(p, q)$ model,

$$\phi(B)X_t = \theta(B)Y_t,$$

where $\phi(B)$ and $\theta(B)$ are polynomials of B with order p and q, respectively.

- Let $\mathcal{F}_t$, the set of observed data up to time t, (i.e. $\{X_1, X_2, \ldots, X_t\}$).
- Then the observation X_t can be written as

$$X_t = \mu_t + Y_t,$$

  where μ_t is the mean of X_t conditional on observed data $\mathcal{F}_{t-1}$,

$$\mu_t = E(X_t|\mathcal{F}_{t-1}) = \phi(B)X_t - (\theta(B) - 1)Y_t$$

and the innovation series Y_t has mean 0 and conditional variance

$$\sigma_t^2 = Var(X_t|\mathcal{F}_{t-1}) = Var(Y_t|\mathcal{F}_{t-1})$$

- The conditional heteroscedastic models of this chapter are concerned with the evolution of σ_t^2 over time.
- The model for μ_t is referred to as the mean equation for X_t and the model for σ_t^2 is the volatility equation for X_t.

4.3 Model Building

The process of building a volatility model for an asset return series consists of four main steps (Tsay, 2010):

1. Specify a mean equation by testing for serial dependence in the data and, if necessary, building an econometric model (e.g., an ARMA model) for the return series to remove any linear dependence.
2. Use the residuals of the mean equation to test for ARCH effects.
3. Specify a volatility model if ARCH effects are statistically significant, and perform a joint estimation of the mean and volatility equations.
4. Check the fitted model carefully and refine it if necessary.

4.4 Testing for ARCH Effect

- Let $Y_t = X_t - \mu_t$ be the residuals of the mean equation.
- The squared series Y_t^2 is then used to check for **conditional heteroscedasticity** (also known as the **ARCH effects**).
- Volatility clustering implies a strong autocorrelation in squared series.
- There are two statistical tests for conditional heteroscedasticity

  1. Method 1: Apply the usual Ljung–Box statistics $Q(m)$ to the Y_t^2 (McLeod and Li, 1983).

     – The null hypothesis is that the first m lags of ACF of the Y_t^2 series are zero.

  2. Method 2: The Lagrange multiplier test of Engle (1982).

4.5 Autoregressive Conditional Heteroskedastic (ARCH) Models

- ARCH model of Engle (1982) is the first model that provides a systematic framework for volatility modeling.
- Main idea:

  (a) Y_t is serially uncorrelated, but dependent,

  (b) the dependence of Y_t can be described by a simple quadratic function of its lagged values.

- To better describe the idea, we represents Y_t having a zero means in the form

$$Y_t = \sigma_t \epsilon_t, \tag{4.1}$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 Y_{t-1}^2 + \dots + \alpha_m Y_{t-m}^2, \tag{4.2}$$

where $\{\epsilon_t\}$ is a sequence of iid random variables with mean zero and variance 1.

- σ_t can be thought of as the local conditional standard deviation of the process.
- $\alpha_0 > 0$, and $\alpha_i \geq 0$ for $i > 0$.
- To ensure that the unconditional variance of Y_t is finite, coefficients α_i must satisfy some regularity conditions.
- The ϵ_t may follow the standard normal or a standardized Student-t or a generalized error distribution.

- A model for Y_t satisfying Equations 4.1 and 4.2 is called an **autoregressive conditional heteroskedastic model** of order m ($ARCH(m)$)
- Note that Equation 4.2 does not include an 'error' term and therefore does not define a stochastic process.

- The ARCH model allows for volatility clustering as the conditional variance σ_t^2 in Equation 4.2 depends on the lagged squared innovations Y_{t-1}^2. A large (positive or negative) residual at time $t-1$ implies that Y_{t-1}^2 is large and consequently the conditional variance σ_t^2 will be large as $\alpha_i \geq 0$. Thereby, large shocks tend to be followed by large shocks (in absolute terms) and small shocks tend to be followed by small shocks (in absolute terms). This is the feature we call **volatility clustering**.
- However, large variance does not necessarily produce a large **realization**. It only says that the **probability** of obtaining a large variate is greater than that of a smaller variance.

4.5.1 Example: Building an ARCH Model

- Let's apply the modeling procedure to build a simple ARCH model for the daily returns, X_t,, of adjusted closing prices of the S&P500 index.
- Figure 4.1 shows the daily return series from January 4, 1995 to February 25, 2021.
- The sample ACF and PACF of the squared returns in the bottom panel of Figure 4.2, show the existence of conditional heteroscedasticity.
- If an ARCH effect is found to be significant, one can use the PACF of Y_t^2 to determine the ARCH order.
- We consider an ARCH(2) model with the following specification for the daily return series

$$
\begin{aligned}
X_t &= \mu + Y_t, \\
Y_t &= \sigma_t \epsilon_t, \\
\sigma_t^2 &= \alpha_0 + \alpha_1 Y_{t-1}^2 + \alpha_2 Y_{t-2}^2.
\end{aligned}
$$

- We use the R function `garchFit` in the R package `fGarch` to estimate the model.

```
# install.packages("fGarch")
library(fGarch)
fit1 <- garchFit(~garch(2,0), data =sp500_return$daily.returns)
summary(fit1)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(2, 0), data = sp500_return$daily.returns)
##
## Mean and Variance Equation:
##  data ~ garch(2, 0)
```

```
## <environment: 0x7fc089b5a9f0>
##  [data = sp500_return$daily.returns]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1      alpha2
## 7.7804e-04  5.8828e-05  2.2118e-01  3.8733e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     7.780e-04   1.098e-04    7.087 1.37e-12 ***
## omega  5.883e-05   1.818e-06   32.366  < 2e-16 ***
## alpha1 2.212e-01   1.967e-02   11.245  < 2e-16 ***
## alpha2 3.873e-01   2.368e-02   16.354  < 2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  20748    normalized:  3.152
##
## Description:
##  Sat Feb 27 13:39:37 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  2657      0
##  Shapiro-Wilk Test  R    W      NA        NA
##  Ljung-Box Test     R    Q(10)  21.61     0.01721
##  Ljung-Box Test     R    Q(15)  34.45     0.002941
##  Ljung-Box Test     R    Q(20)  38.39     0.007933
##  Ljung-Box Test     R^2  Q(10)  362.7     0
##  Ljung-Box Test     R^2  Q(15)  546.8     0
##  Ljung-Box Test     R^2  Q(20)  713.2     0
##  LM Arch Test       R    TR^2   428.5     0
##
## Information Criterion Statistics:
##    AIC    BIC    SIC   HQIC
## -6.303 -6.299 -6.303 -6.302
```

Plot the result

```
sp500_return <- sp500_return %>%
  mutate(vol = volatility(fit1),
         resid.st = residuals(fit1, standardize=T)) %>%
  as_tsibble(index = date, regular = FALSE)

p1 <- sp500_return %>% autoplot(vol)
p2 <- sp500_return %>% autoplot(resid.st)
p3 <- sp500_return %>% ACF(resid.st) %>% autoplot()
p4 <- sp500_return %>% PACF(resid.st) %>% autoplot()
p5 <- sp500_return %>% ACF(resid.st^2) %>% autoplot()
p6 <- sp500_return %>% PACF(resid.st^2) %>% autoplot()
```

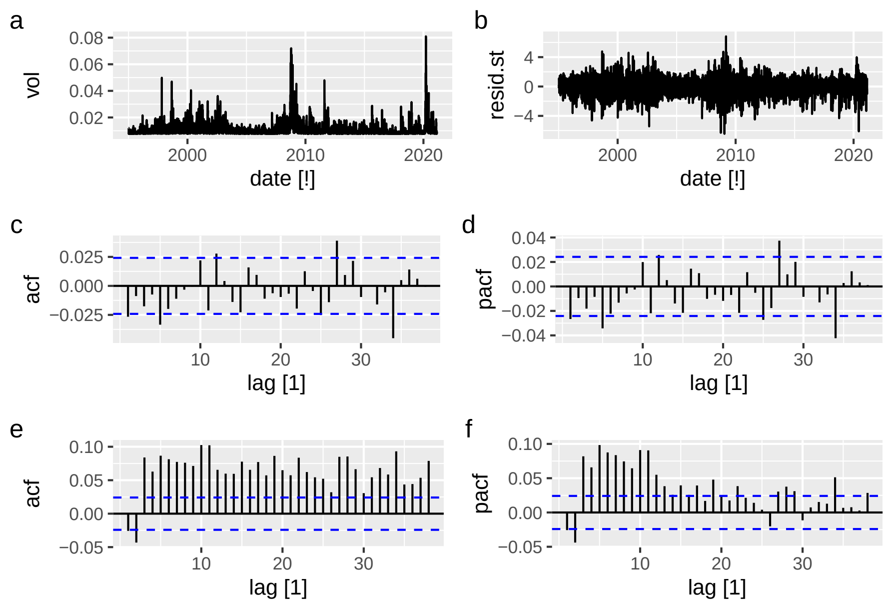


Figure 4.3: (a) Estimated volatility $\hat{\sigma_t}$, (b) the standardized residual $\hat{\epsilon_t}$, (c) sample ACF of $\hat{\epsilon_t}$, (d) sample PACF of $\hat{\epsilon_t}$, (e) sample ACF of $\hat{\epsilon_t^2}$, (d) sample PACF of $\hat{\epsilon_t^2}$, in the ARCH(2) model for daily returns of adjusted closing prices of the S&P500 Index from January 4, 1995 to February 25, 2021.

- Assuming that ϵ_t are iid, standard normal, we get the fitted model as follows:

$$X_t = 0.0007804_{(1.098e-04)} + Y_t,\ Y_t = \sigma_t \epsilon_t,$$

$$\sigma_t^2 = 0.00005883_{(1.818e-06)} + 0.2212_{(1.967e-02)} Y_{t-1}^2 + 0.3873_{(2.368e-02)} Y_{t-2}^2$$

where the standard errors of the parameters are given in the parentheses.

- Both the output and the plot show that the estimated residuals still have conditional heteroscedasticities.
- Therefore, the ARCH(2) model is not adequate.

4.6 Generalized ARCH (GARCH) Models

- Although the ARCH model is simple, it often requires many parameters to adequately describe the volatility process of an asset return.
- Bollerslev (1986) proposes a useful extension known as the generalized ARCH (GARCH) model.
- The ARCH model has been generalized to allow the variance to depend on past values of Y_t^2 as well as on past values of σ_t^2.
- A model for Y_t satisfying Equations 4.1 is said to follow a **generalized ARCH (or GARCH) model** of order (m, s) $(GARCH(m, s))$ when the local conditional variance is given by

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{m} \alpha_i Y_{t-i}^2 + \sum_{j=1}^{s} \beta_j \sigma_{t-j}^2, \tag{4.3}$$

where $\{\epsilon_t\}$ is a sequence of iid random variables with mean 0 and variance 1, $\alpha_0 > 0$, $\alpha_i \geq 0$, $\beta_j \geq 0$, and $\sum_{i=1}^{max(m,s)} (\alpha_i + \beta_i) < 1$.

- Here, $\alpha_i = 0$ for $i > m$ and $\beta_j = 0$ for $j > s$.
- The constraint on $\alpha_i + \beta_j$ implies that the unconditional variance of Y_t is finite, whereas its conditional variance σ_t^2 evolves over time.
- ϵ_t is often assumed to follow a standard normal or standardized Student-t distribution or generalized error distribution.
- Equation 4.3 reduces to a pure ARCH(m) model if $s = 0$ (i.e. GARCH$(m, 0)$).
- The α_i and β_j are known as ARCH and GARCH parameters, respectively.

To understand properties of GARCH models, it is informative to use the following representation.

- Let $\eta_t = Y_t^2 - \sigma_t^2$.

- Then $\sigma_t^2 = Y_t^2 - \eta_t$.
- By plugging $\sigma_{t-1}^2 = Y_{t-1}^2 - \eta_{t-1}$ $(i = 0, \dots, s)$ into Equation 4.3, we get the GARCH model as

$$Y_t^2 = \alpha_0 + \sum_{i=1}^{max(m,s)} (\alpha_i + \beta_i) Y_{t-i}^2 + \eta_t - \sum_{j=1}^{s} \beta_j \eta_{t-j}. \tag{4.4}$$

- It is easy to check that $\{\eta_t\}$ is a martingale difference series [i.e., $E(\eta_t) = 0$ and $cov(\eta_t, \eta_{t-j}) = 0$ for $j \geq 1$].
- However, $\{\eta_t\}$ in general is not an iid sequence.
- Equation 4.4 is an ARMA form for the squared series Y_t^2.
- Therefore, a GARCH model can be regarded as an application of the ARMA idea to the squared series Y_t^2.
- Therefore, the same invertibility and stationary assumptions of ARMA models apply to GARCh models.
- For example, Y_t is covariance stationary, if all roots of $1 - \sum_{j=1}^{max(m,s)} (\alpha_j + \beta_j) z^j = 0$ lie outside the unit circle.

GARCH$(1,1)$ model

- The $GARCH(1,1)$ model is often used to fit financial time series.
- The properties of GARCH models can easily be understood by focusing on the simplest $GARCH(1,1)$ model with

$$\sigma_t^2 = \alpha_0 + \alpha_1 Y_{t-1}^2 + \beta_1 \sigma_{t-1}^2,\ 0 \leq \alpha_1, \beta_1 \leq 1, (\alpha_1 + \beta_1) < 1.$$

- The model implies that a large Y_{t-1}^2 or σ_{t-1}^2 will lead to a large σ_t^2, which in turn will give rise to a large $Y_t^2 = \sigma_t^2 \epsilon_t^2$.
- This again generate the well-known behaviour of volatility clustering in financial time series.

4.6.1 Prediction of volatilities

- Both ARCH and GARCH models do not affect point forecasts of the original observed variables.
- Therefore, it is difficult to make a fair comparison of the forecasting abilities of different models for changing variance.
- Therefore, both the modeling aspect (understanding the changing structure of a series), and the assessment of risk, are more important than their ability to make point forecasts.
- Forecasts of a GARCH model are similar to those of an ARMA model.
- Consider the GARCH(1,1) model.
- Let the forecast origin is h.
- For 1-step ahead forecast, we have

$$\sigma^2_{h+1} = \alpha_0 + \alpha_1 Y_h^2 + \beta_1 \sigma_h^2$$

where Y_h and σ_h^2 are known at the time index h.

- Then, 1-step ahead forecast is

$$\sigma^2_h(1) = \alpha_0 + \alpha_1 Y_h^2 + \beta_1 \sigma_h^2.$$

- For multistep-ahead forecasts, we use $Y_t^2 = \sigma_t^2 \epsilon_t^2$.
- Then the volatility equation can be written as

$$\sigma^2_{t+1} = \alpha_0 + (\alpha_1 + \beta_1)\sigma_t^2 + \alpha_1 \sigma_t^2(\epsilon_t^2 - 1).$$

- When $t = h + 1$, we get,

$$\sigma^2_{h+2} = \alpha_0 + (\alpha_1 + \beta_1)\sigma_{h+1}^2 + \alpha_1 \sigma_{h+1}^2(\epsilon_{h+1}^2 - 1).$$

- Since $E(\epsilon^2_{h+1} - 1|\mathcal{F}_t) = 0$, the 2-step-ahead volatility forecast at the forecast origin h satisfies the equation

$$\sigma^2_h(2) = \alpha_0 + (\alpha_1 + \beta_1)\sigma_h^2(1).$$

- In general, we can write

$$\sigma^2_h(k) = \alpha_0 + (\alpha_1 + \beta_1)\sigma_h^2(k-1), \quad k > 1.$$

- This result is similar to an ARMA(1,1) model with AR polynomial $1-(\alpha_1+\beta_1)B$.

4.6.2 Example: Building a GARCH Model

- Let's apply the modeling procedure to build a $GARCH(1,1)$ model for the daily returns, X_t, of adjusted closing prices of the S&P500 index.
- Figure 4.1 shows the daily return series from January 4, 1995 to February 25, 2021.
- The sample ACF and PACF of the squared returns in the bottom panel of Figure 4.2, show the existence of conditional heteroscedasticity.

```
fit2 <- garchFit(~garch(1,1), data =sp500_return$daily.returns)
summary(fit2)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = sp500_return$daily.returns)
```

```
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x7ff02db5eac0>
##  [data = sp500_return$daily.returns]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 7.2933e-04  2.1055e-06  1.2134e-01  8.6479e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     7.293e-04   9.722e-05    7.502 6.28e-14 ***
## omega  2.106e-06   2.656e-07    7.927 2.22e-15 ***
## alpha1 1.213e-01   8.705e-03   13.940  < 2e-16 ***
## beta1  8.648e-01   8.885e-03   97.335  < 2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  21322    normalized:  3.239
##
## Description:
##  Sat Feb 27 16:30:14 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1195      0
##  Shapiro-Wilk Test  R    W      NA        NA
##  Ljung-Box Test     R    Q(10)  21.65     0.01699
##  Ljung-Box Test     R    Q(15)  30.7      0.009632
##  Ljung-Box Test     R    Q(20)  34.98     0.0202
##  Ljung-Box Test     R^2  Q(10)  12.97     0.2255
##  Ljung-Box Test     R^2  Q(15)  16.83     0.329
##  Ljung-Box Test     R^2  Q(20)  17.81     0.5999
##  LM Arch Test       R    TR^2   13.93     0.3055
##
## Information Criterion Statistics:
##    AIC    BIC    SIC   HQIC
## -6.478 -6.473 -6.478 -6.476
```

Plot the result

```r
sp500_return <- sp500_return %>%
  mutate(vol = volatility(fit2),
         resid.st = residuals(fit2, standardize=T)) %>%
  as_tsibble(index = date, regular = FALSE)

p1 <- sp500_return %>% autoplot(vol)
p2 <- sp500_return %>% autoplot(resid.st)
p3 <- sp500_return %>% ACF(resid.st) %>% autoplot()
p4 <- sp500_return %>% PACF(resid.st) %>% autoplot()
p5 <- sp500_return %>% ACF(resid.st^2) %>% autoplot()
p6 <- sp500_return %>% PACF(resid.st^2) %>% autoplot()
```

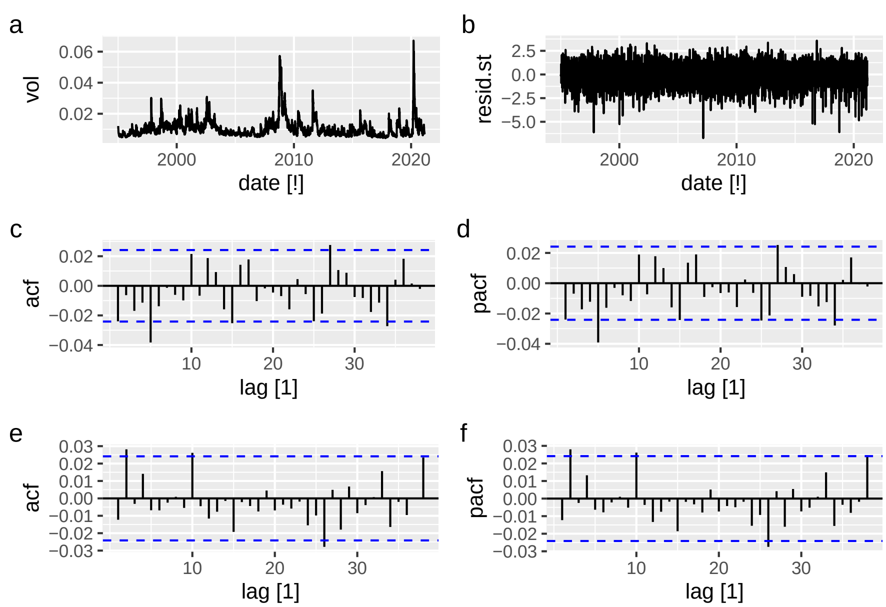
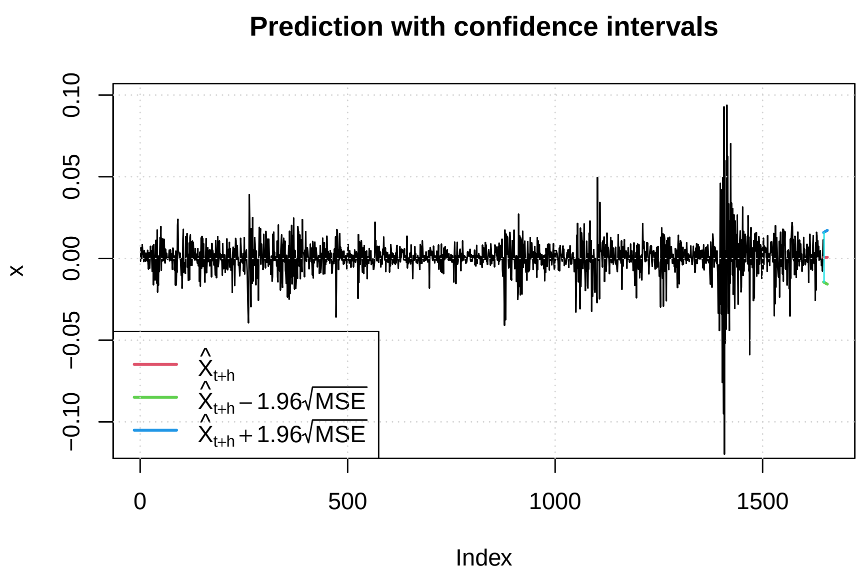


Figure 4.4: (a) Estimated volatility $\hat{\sigma}_t$, (b) the standardized residual $\hat{\epsilon}_t$, (c) sample ACF of $\hat{\epsilon}_t$, (d) sample PACF of $\hat{\epsilon}_t$, (e) sample ACF of $\hat{\epsilon_t^2}$, (d) sample PACF of $\hat{\epsilon_t^2}$, in the GARCH(1,1) model for daily returns of adjusted closing prices of the S&P500 Index from January 4, 1995 to February 25, 2021.

- Now the model is adequate as the standardized residuals in the fitted $GARCH(1,1)$ model show little conditional heteroscedasticity.

```r
## predict
predict(fit2, n.ahead = 10)
```

```
## predict with plotting
## 95% confidence level
predict(fit2, n.ahead = 10, plot=TRUE, conf=.95)
```



4.7 The ARMA-GARCH Models

- The dynamics of asset returns and their volatility can be modeled by combining GARCH models with the linear time series models discussed in Chapter 2.
- Let X_t follows an ARMA model with GARCH innovations.
- Then, it yields the following $ARMA(p, q) - GARCH(m, s)$ model for (X_t, σ_t) :

$$X_t = \mu + \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{j=1}^{q} \theta_j Y_{t-j} + Y_t, \quad Y_t = \sigma_t \epsilon_t,$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{m} \alpha_i Y_{t-i}^2 + \sum_{j=1}^{s} \beta_j \sigma_{t-j}^2.$$

- The ϵ_t are iid standard normal or Student-t random variables.
- The second equation can be replaced by other volatility models so that different aspects of volatilities can be characterized.

4.7.1 Example: Building an ARMA-GARCH Model

- Let's apply the modeling procedure to build an $ARMA(1,1) - GARCH(1,1)$ model for the daily returns, X_t, of adjusted closing prices of the S&P500 index

$$X_t = \mu + \phi X_{t-1} + \theta Y_{t-1} + Y_t,$$
$$Y_t = \sigma_t \epsilon_t,$$

$$\sigma_t^2 = \alpha_0 + \alpha Y_{t-1}^2 + \beta \sigma_{t-1}^2.$$

- Let ϵ_t are iid standard normal.

```
fit3 <- garchFit(~arma(1,1)+garch(1,1), data =sp500_return$daily.returns)
summary(fit3)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~arma(1, 1) + garch(1, 1), data = sp500_return$daily.returns)
##
## Mean and Variance Equation:
##  data ~ arma(1, 1) + garch(1, 1)
## <environment: 0x7fc15d07ec18>
##  [data = sp500_return$daily.returns]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##          mu          ar1          ma1        omega
##  8.7477e-05   8.8353e-01  -9.1463e-01   2.0642e-06
##      alpha1        beta1
##  1.1989e-01   8.6645e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## mu      8.748e-05   4.726e-05    1.851   0.0642 .
## ar1     8.835e-01   6.139e-02   14.392  < 2e-16 ***
## ma1    -9.146e-01   5.330e-02  -17.160  < 2e-16 ***
## omega   2.064e-06   2.603e-07    7.932 2.22e-15 ***
## alpha1  1.199e-01   8.565e-03   13.997  < 2e-16 ***
## beta1   8.664e-01   8.736e-03   99.186  < 2e-16 ***
```

```
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  21335    normalized:  3.241
##
## Description:
##  Sat Feb 27 20:47:45 2021 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1279      0
##  Shapiro-Wilk Test  R    W      NA        NA
##  Ljung-Box Test     R    Q(10)  13.06     0.2206
##  Ljung-Box Test     R    Q(15)  22.45     0.09663
##  Ljung-Box Test     R    Q(20)  28.44     0.09946
##  Ljung-Box Test     R^2  Q(10)  14.88     0.1364
##  Ljung-Box Test     R^2  Q(15)  18.44     0.2403
##  Ljung-Box Test     R^2  Q(20)  19.64     0.4806
##  LM Arch Test       R    TR^2   15.35     0.2229
##
## Information Criterion Statistics:
##    AIC    BIC    SIC   HQIC
## -6.481 -6.475 -6.481 -6.479
```

Plot the result

```
sp500_return <- sp500_return %>%
  mutate(vol = volatility(fit3),
         resid.st = residuals(fit3, standardize=T)) %>%
  as_tsibble(index = date, regular = FALSE)

p1 <- sp500_return %>% autoplot(vol)
p2 <- sp500_return %>% autoplot(resid.st)
p3 <- sp500_return %>% ACF(resid.st) %>% autoplot()
p4 <- sp500_return %>% PACF(resid.st) %>% autoplot()
p5 <- sp500_return %>% ACF(resid.st^2) %>% autoplot()
p6 <- sp500_return %>% PACF(resid.st^2) %>% autoplot()
```

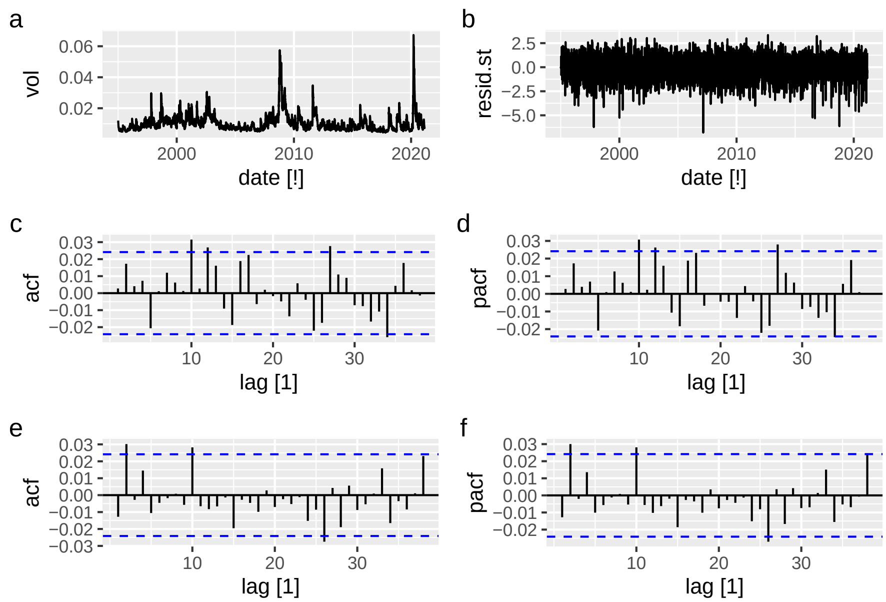


Figure 4.5: (a) Estimated volatility $\hat{\sigma_t}$, (b) the standardized residual $\hat{\epsilon_t}$, (c) sample ACF of $\hat{\epsilon_t}$, (d) sample PACF of $\hat{\epsilon_t}$, (e) sample ACF of $\widehat{\epsilon_t^2}$, (d) sample PACF of $\widehat{\epsilon_t^2}$, in the ARMA(1,1)-GARCH(1,1) model for daily returns of adjusted closing prices of the S&P500 Index from January 4, 1995 to February 25, 2021.

4.8 Other ARCH-Type Models

- There are many other types of ARCH models.
- It is important to use the context and any background theory, when choosing between different models for changing variance.

4.8.1 The Integrated GARCH (IGARCH) model

- The GARCH model becomes an IGARCH mode, if the AR polynomial of the GARCH representation in Equation 4.4 has a unit root.
- Similar to ARIMA models, in IGARCH models the impact of past squared shocks $\eta_{t-i} = Y_{t-i}^2 - \sigma_{t-i}^2$ for $i > 0$ on Y_t^2 is persistent.
- The IGARCH(1,1) model is of the form

$$
\begin{aligned}
Y_t &= \sigma_t \epsilon_t, \\
\sigma_t^2 &= \alpha + \beta\sigma_{t-1}^2 + (1-\beta)Y_{t-1}^2,
\end{aligned}
$$

where $\{\epsilon_t\}$ is a sequence of iid random variables with mean zero and variance 1 and $0 < \beta < 1$.

- More generally, model for Y_t satisfying Equations 4.1 is said to follow a **Integrated GARCH model** of order (m, s) $(IGARCH(m, s))$ when the local conditional variance is given by

$$
\sigma_t^2 = \alpha_0 + \sum_{i=1}^{m} \alpha_i Y_{t-i}^2 + \sum_{j=1}^{s} \beta_j \sigma_{t-j}^2, \tag{4.5}
$$

where $\{\epsilon_t\}$ is a sequence of iid random variables with mean 0 and variance 1, with $\sum_{i=1}^{m} \alpha_i + \sum_{j=1}^{s} \beta_j = 1$.

- Due to the above condition, the unconditional variance of Y_t, hence that of X_t , is not defined under $IGARCH(1,1)$ model.
- Under certain conditions, the IGARCH models are strictly stationary but not weakly stationary as they do not have the first two moments.
- When $\alpha 1 + \beta 1 = 1$,

$$
\sigma_h^2(k) = \sigma_h^2(1) + (k-1)\alpha_0, \quad k \geq 1,
$$

  where h is the forecast origin.

- As a result, the effect of $\sigma_h^2(1)$ on future volatilities is also persistent, and the volatility forecasts generates a straight line with slope α_0.
- When $\alpha_0 = 0$, the volatility forecasts are simply $\sigma_h^2(1)$ for all forecast horizons;

Example in R

```
# For univariate GARCH models)
library(rugarch)
spec3 <- ugarchspec(
  variance.model=list(model="iGARCH",
                      garchOrder=c(1,1)),
  mean.model=list(armaOrder=c(0,0),
                  include.mean=FALSE),
  distribution.model="norm",
  fixed.pars=list(omega=0))
fit <- ugarchfit(spec3,
       data = sp500_return$daily.returns)
summary(fit)
fit
plot(fit,which="all")
```

- Useful materials: The `rugarch` package vignette

4.9 Asymmetry in Volatility

- Some equity markets show asymmetry in their volatility clusters
- Asymmetric Volatility is when the volatility of a market or stock is higher when a market is in a downtrend and volatility tends to be lower in an uptrend.
- There may be a range of causes of asymmetric volatility, but factors such as leverage, panic selling, and serial correlation are often some of the drivers.
- The asymmetric GARCH models are employed to capture the asymmetric characteristics of volatility.

4.10 The exponential GARCH model

- In 1991, Nelson proposed the **Exponential GARCH** model to allow for asymmetric effects between positive and negative asset returns.
- An $EGARCH(h,k)$ model can be written as

$$\begin{aligned} Y_t &= \sigma_t \epsilon_t, \\ log(\sigma_t^2) &= \alpha_0 + \sum_{i=1}^{h} \beta_i log(\sigma_{t-i}^2) + \sum_{j=1}^{k} f_j(\epsilon_{t-j}) \end{aligned} \tag{4.6}$$

where the $\{\epsilon_t\}$ is a sequence of iid random variables with mean 0 and variance 1, and $f_j(\epsilon) = \alpha_j \epsilon + \gamma_j(|\epsilon| - E|\epsilon|)$

- Both ϵ_t and $|\epsilon_t| - E|\epsilon_t|$ are zero-mean iid sequences with continuous distributions.
- Therefore, $E[f_j(\epsilon_t)] = 0$

- The asymmetry of $f_j(\epsilon_t)$ can easily be seen by rewriting it as

$$f_j(\epsilon_t) = \begin{cases} (\alpha_j + \gamma_j)\epsilon_t - \gamma_j E|\epsilon_t|, & \text{if } \epsilon_t \geq 0 \\ (\alpha_j - \gamma_j)\epsilon_t - \gamma_j E|\epsilon_t|, & \text{if } \epsilon_t < 0 \end{cases}$$

- This shows the asymmetry of the volatility response to positive and negative returns.
- The model differs from the GARCH model in several ways
  - First, it uses logged conditional variance to relax the positiveness constraint of model coefficients.
  - Second, the use of $f_j(\epsilon_t)$ enables the model to respond asymmetrically to positive and negative lagged values of Y_t.
- Some additional properties of the EGARCH model can be found in Nelson (1991).

4.11 Stochastic Volatility Models

- The formulae for σ_t^2 in all GARCH type models are deterministic as there is no "error" term in either equation.
- An alternative approach to ARCH or GARCH models is to assume that σ_t follows a **Stochastic process**.
- This can be usually done by modelling $log(\sigma_t^2)$ or $log(\sigma_t)$ to ensure that σ_t^2 remains positive.
- Example: Let $log(\sigma_t^2) = h_t$ and h_t follows an AR process with an "error" component that is independent of the $\{\epsilon_t\}$ in the innovation series $\{Y_t\}$,

$$Y_t = \sigma_t \epsilon_t, \quad \sigma_t^2 = e^{h_t},$$

$$h_t = \phi_0 + \phi_1 h_{t-1} + \cdots + \phi_p h_{t-p} + \eta_t,$$

which has $AP(p)$ dynamics for $log\sigma_t^2$.

- These type of models are called **stochastic volatility** or **Stochastic variance models**.
- The ϵ_t and η_t are assumed to be independent normally distributed random variables with $\epsilon_t \sim N(0,1)$ and $\eta_t \sim N(0, \sigma^2)$
- In certain cases, it is more reasonable to assume that σ_t changes stochastically through time rather than deterministically.
  - Example: During a sudden change in volatility in financial market due to a special event such as a war involving oil producing countries

4.12 References:

- Chatfield, C., & Xing, H. (2019). The analysis of time series: an introduction with R. CRC press.
- Tsay, R. S. (2010). Analysis of Financial Time Series. John Wiley & Sons..

Chapter 5

Multivariate Time Series Modelling

This chapter is heavily based on Chapter 13 of Chatfield and Xing (2019) and Tsay (2013) .

5.1 Introduction

- Multivariate time series analysis considers observations taken simultaneously on two or more time series.
- Focus of multivariate time series analysis
  - Study the dynamic relationships between variables
  - Serial dependence **within** each series and the interdependence **between** series.
  - Improve the accuracy of prediction
- Challengers with Multivariate models
  - Model building process is more difficult for multivariate than univariate models
  - More variables to measure (More chance of mistakes in the data)
  - More parameters to estimate
  - Wider pool of candidate models
  - More vulnerable to specification than simpler univariate models.
  - Balance between parsimonious modelling and accurate identification.

5.2 The Cross-Correlation Function

- Cross-correlation function is a key tool in multivariate time series analysis.
- Let $\{\mathbf{X}_t\}$ is an *m-variate* multivariate process, where $\mathbf{X}_t^T = (X_{1t}, X_{2t}, \dots, X_{mt})$

Cross-covariance

- Let $\boldsymbol{\mu}_t$ be the vector of **mean** values of $\{\mathbf{X}_t\}$ at time t.
- Then its ith component is $\mu_{it} = E(X_{it})$.
- Let $\Gamma(t, t+k)$ be the **cross-covariance matrix** of $\mathbf{X}_t$ and $\mathbf{X}_{t+k}$ such that its (i, j)th element is the cross-covariance coefficient of X_{it} and $X_{j,t+k}$.
- A multivariate process is said to be **second-order stationary** if the mean and the cross-covariance matrices at different lags do not depend on time.
- Then $\boldsymbol{\mu}_t$ will be a constant (say $\boldsymbol{\mu}$) and $\Gamma(t, t+k)$ will be a function of the lag k only ($\Gamma(k)$).
- Then $\gamma_{ij}(k)$, the (i, j)th element of $(\Gamma(k))$ can be written as

$$\gamma_{ij}(k) = \mathrm{Cov}(X_{it}, X_{j,t+k}) = E[(X_{it} - \mu_i)(X_{j,t+k} - \mu_j)]$$

- In the stationary case, the set of cross-covariance matrices, $\Gamma(k)$ for $k = 0, \pm 1, \pm 2, \dots$, is known as **covariance matrix function**.
- Since

$$\gamma_{ij}(k) = \mathrm{Cov}(X_{it}, X_{j,t+k}) = \mathrm{Cov}(X_{j,t+k}, X_{it}) = \gamma_{ij}(-k),$$

we have

$$\Gamma(k) = \Gamma^T(-k), \quad k = 0, \pm 1, \pm 2, \dots .$$

- It is not an even function of lag.
- The diagonal terms, $\gamma_{ii}(k)$, are auto- rather than cross- covariances, and therefore have the property of being an even function of lags.

Cross-correlation

- Let $R(k)$ be the **cross-correlation matrix function** of the process.
- The $(i, j)$the element of $R(k)$ is given by

$$\rho_{ij}(k) = \mathrm{Corr}(X_{j,t+k}, X_{it}) = \gamma_{ij}(k)/\sigma_i\sigma_j$$

where σ_i is the standard deviations of X_{it} (this can also be expressed as $\sqrt{\gamma_{ii}(0)}$.)

- When $k > 0$, the correlation coefficient measures the linear dependence of $X_{j,t+k}$ on X_{it}, which occurs after time t.
- If $\rho_{ij}(k) \neq 0$ and $k > 0$, the series X_{it} leads the series X_{jt} at lag k.
- Furthermore, we can write

$$R(K) = R^T(-k), \qquad k = 0, \pm 1, \pm 2, \dots .$$

- Therefore, in practice, it is enough to consider the cross-correlation matrices $R(k)$ for $k > 0$.

Sample cross-correlation coefficient

- Let T be the total number of observations collected on the m variables over the same time period.
- Then the **sample cross-covariance** coefficient of X_i and X_j at lag k is given by

$$c_{ij}(k) = \begin{cases} \sum_{t=1}^{T-k}(x_{it} - \bar{x_i})(x_{j,t+k} - \bar{x_j})/T, & k = 0, 1, 2, \dots, (T-1) \\ \sum_{t=1-k}^{T}(x_{it} - \bar{x_i})(x_{j,t+k} - \bar{x_j})/T, & k = -1, -2, \dots, -(T-1). \end{cases}$$

- The **sample cross-correlation** coefficient of X_i and X_j at lag k is given by

$$\gamma_{ij}(k) = c_{ij}(k)/s_i s_j$$

where $s_i = \sqrt{c_{ii}(0)}$ is the sample standard deviation of observations on the ith variable.

Example

- Consider the daily returns of adjusted closing prices of the Standard & Poor's 500 (S&P500), the Dow Jones Industrial Average and the Nasdaq Composite indices from January 4, 1995 to February 25, 2021 (Figure 5.1).
- These three market indices characterize the performance of the U.S. stock market from different perspectives and therefore they should be highly correlated.

```r
# Tidy financial analysis
library(tidyquant)

#S&P 500 index
sp500 <- tq_get("^GSPC", from = "1995-01-04", to = "2021-02-25" )
print(sp500)
```

```
## # A tibble: 6,582 x 8
##    symbol date        open  high   low close volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>  <dbl>    <dbl>
##  1 ^GSPC  1995-01-04  459.  461.  458.  461. 3.20e8     461.
##  2 ^GSPC  1995-01-05  461.  461.  460.  460. 3.09e8     460.
##  3 ^GSPC  1995-01-06  460.  462.  459.  461. 3.08e8     461.
##  4 ^GSPC  1995-01-09  461.  462.  460.  461. 2.79e8     461.
##  5 ^GSPC  1995-01-10  461.  465.  461.  462. 3.52e8     462.
##  6 ^GSPC  1995-01-11  462.  464.  459.  462. 3.46e8     462.
##  7 ^GSPC  1995-01-12  462.  462.  461.  462. 3.13e8     462.
##  8 ^GSPC  1995-01-13  462.  466.  462.  466. 3.37e8     466.
##  9 ^GSPC  1995-01-16  466.  470.  466.  469. 3.16e8     469.
## 10 ^GSPC  1995-01-17  469.  470.  468.  470. 3.32e8     470.
## # ... with 6,572 more rows
```

```r
# The Dow Jones Industrial Average (DJIA)
dji<- tq_get("^DJI", from = "1995-01-04", to = "2021-02-25" )
print(dji)
```

```
## # A tibble: 6,582 x 8
##    symbol date        open  high   low close volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>  <dbl>    <dbl>
##  1 ^DJI   1995-01-04 3838. 3858. 3831. 3858. 272200    3858.
##  2 ^DJI   1995-01-05 3858. 3861. 3843. 3851. 258100    3851.
##  3 ^DJI   1995-01-06 3851. 3887. 3842. 3867. 302400    3867.
##  4 ^DJI   1995-01-09 3867. 3874. 3853. 3861. 208200    3861.
##  5 ^DJI   1995-01-10 3861. 3899. 3861. 3867. 282500    3867.
##  6 ^DJI   1995-01-11 3867. 3883. 3840. 3862. 281000    3862.
##  7 ^DJI   1995-01-12 3862  3864. 3851. 3859  237400    3859
##  8 ^DJI   1995-01-13 3859  3910. 3859  3908. 305800    3908.
##  9 ^DJI   1995-01-16 3908. 3937. 3907. 3932. 292500    3932.
## 10 ^DJI   1995-01-17 3932. 3935. 3916. 3931. 268800    3931.
## # ... with 6,572 more rows
```

```r
# The Nasdaq Composite
nasdaq<- tq_get("^IXIC", from = "1995-01-04", to = "2021-02-25" )
print(nasdaq)
```

```
## # A tibble: 6,582 x 8
##    symbol date        open  high   low close volume adjusted
##    <chr>  <date>     <dbl> <dbl> <dbl> <dbl>  <dbl>    <dbl>
##  1 ^IXIC  1995-01-04  745.  746.  740.  746. 2.90e8     746.
##  2 ^IXIC  1995-01-05  747.  748.  745.  746. 2.98e8     746.
##  3 ^IXIC  1995-01-06  746.  751.  746.  750. 3.13e8     750.
##  4 ^IXIC  1995-01-09  750.  753.  750.  752. 2.67e8     752.
##  5 ^IXIC  1995-01-10  754.  759.  754.  757. 3.54e8     757.
##  6 ^IXIC  1995-01-11  758.  760.  752.  756. 3.30e8     756.
##  7 ^IXIC  1995-01-12  756.  757.  755.  757. 3.02e8     757.
##  8 ^IXIC  1995-01-13  758.  762.  757.  762. 3.14e8     762.
##  9 ^IXIC  1995-01-16  762.  769.  762.  768. 3.01e8     768.
## 10 ^IXIC  1995-01-17  769.  772.  768.  772. 3.37e8     772.
## # ... with 6,572 more rows
```

```r
# Convert each assets raw adjusted closing prices to returns
sp500_return <- sp500 %>%
  tq_transmute(select     = adjusted,
               mutate_fun = periodReturn,
               period     = "daily")

dji_return <- dji %>%
  tq_transmute(select     = adjusted,
               mutate_fun = periodReturn,
```

```r
               period     = "daily")

nasdaq_return <- nasdaq %>%
  tq_transmute(select     = adjusted,
               mutate_fun = periodReturn,
               period     = "daily")

p1 <- sp500_return %>%
  as_tsibble(index = date) %>%
  autoplot(daily.returns) +
  labs(x = "Day", y= "S&P500")
p2 <- dji_return %>%
  as_tsibble(index = date) %>%
  autoplot(daily.returns) +
  labs(x = "Day", y= "Dow Jones")
p3 <- nasdaq_return %>%
  as_tsibble(index = date) %>%
  autoplot(daily.returns) +
  labs(x = "Day", y= "Nasdaq")

p1 / p2/ p3
```

A function of computing sample cross correlation

- We use the `ccm` function of the `MTS` package in R to obtain the cross-correlation plots for a dataset.

```r
library(MTS)
data <- full_join(sp500_return, dji_return, by= "date" )
data <- full_join(data, nasdaq_return, by ="date" )
colnames(data) <- c("date", "sp500", "dji", "nasdaq")

ret <- data %>%
   select(sp500, dji, nasdaq) %>%
   as.matrix()
MTS::MTSplot(ret)
```

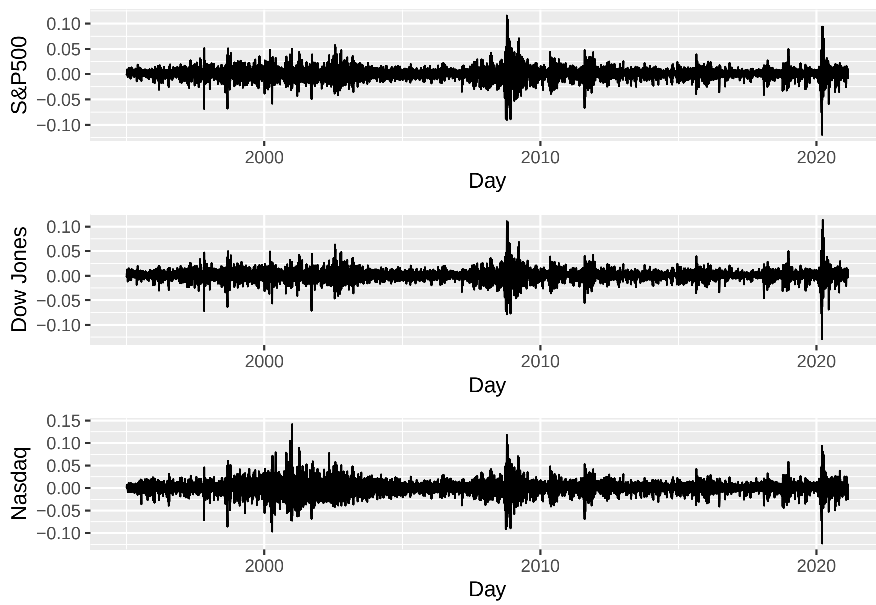
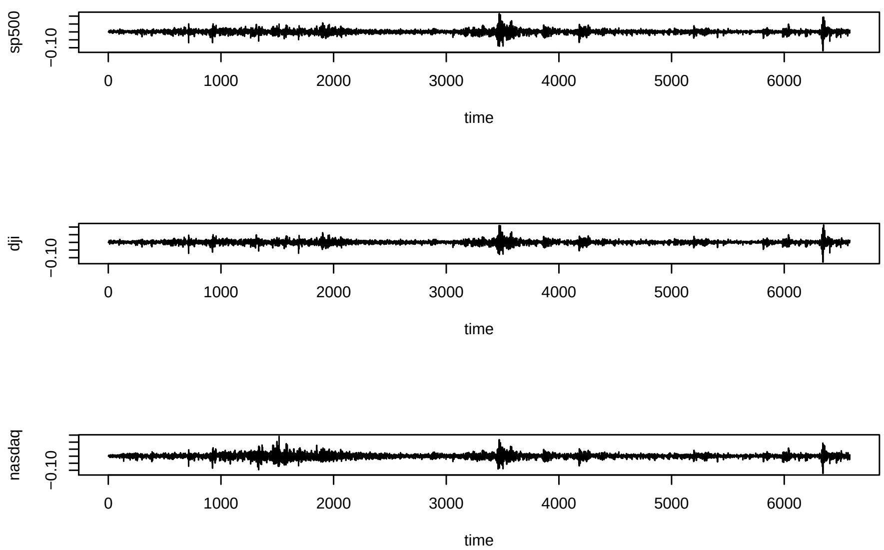
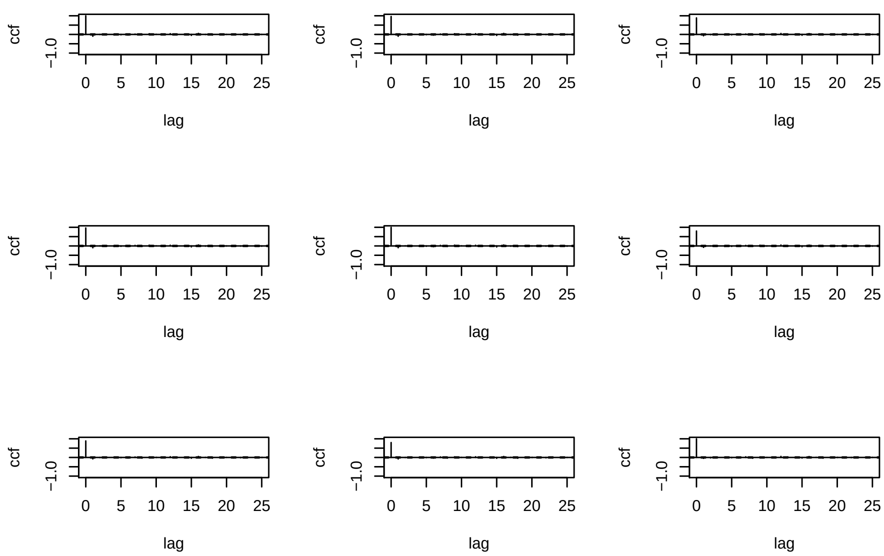
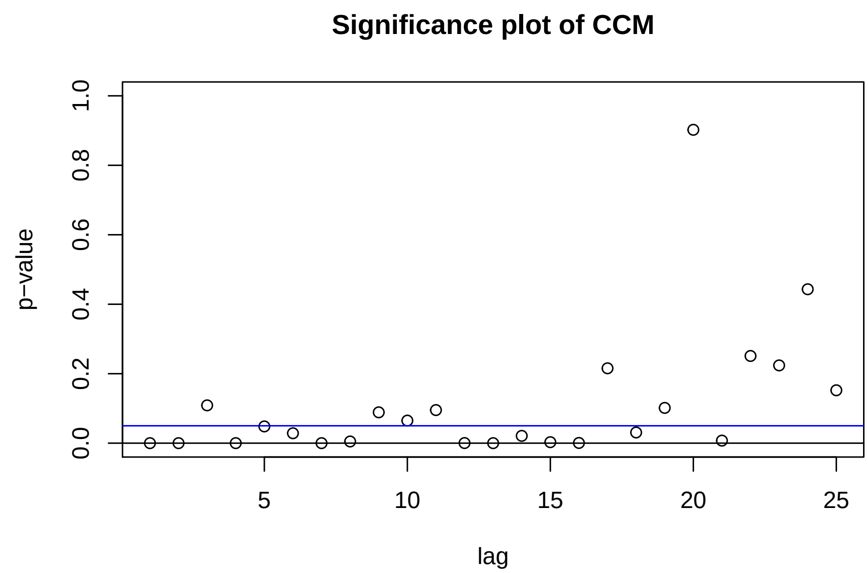


Figure 5.1: Daily returns of adjusted closing prices of the Standard & Poor's 500 (S&P500), the Dow Jones Indutrial Average and the Nasdaq Composite indices from January 4, 1995 to February 25, 2021




```r
MTS::ccm(ret, lags = 25)
```





- The concurrent interdependence of the three series are very strong.
- However, the lead-lag effect among the three series is relatively weak.

5.3 Vector Autoregressive Models

- The most commonly used multivariate time series model is the vector autoregressive (VAR) model.
- To study the properties of VAR(p) models, we start with the simple VAR(1) model.

5.3.1 VAR(1) models

- Consider m-variate multivariate process, $\{\mathbf{X}_t\}$, where $\mathbf{X}_t^T = (X_{1t}, X_{2t}, \dots, X_{mt})$.
- For simplicity, we restrict the attention to the case $m = 2$.
- For stationary series, we may, without loss of generality, assume the variables have been scaled to have zero mean.
- Now the model allows the values of X_{1t} and X_{2t} to depend linearly on the values of both series at time $(t-1)$.
- The resulting model for the two series then consist of two equations

$$\begin{cases} X_{1t} = \phi_{11}X_{1,t-1} + \phi_{12}X_{2,t-1} + \epsilon_{1t} \\ X_{2t} = \phi_{21}X_{1,t-1} + \phi_{22}X_{2,t-1} + \epsilon_{2t} \end{cases}$$

where $\{\phi_{ij}\}$ are constants.

- The two error terms, ϵ_{1t} and ϵ_{2t} are usually assumed to be white noise, but are often allowed to be correlated contemporaneously.
- Note: If coefficients, $\phi_{12} = \phi_{21} = 0$, then X_{1t} and X_{2t} are not dynamically correlated.
- If one of ϕ_{12} and ϕ_{21} is not zero, say $\phi_{12} = 0$ and $\phi_{21} \neq 0$, then X_{1t} does not depend on the lagged values of X_{2t}.
- Then the system of equations reduces to

$$\begin{cases} X_{1t} = \phi_{11}X_{1,t-1} + \epsilon_{1t} \\ X_{2t} = \phi_{21}X_{1,t-1} + \phi_{22}X_{2,t-1} + \epsilon_{2t}. \end{cases}$$

- This indicates that, while X_{2t} depends on the lagged value of X_{1t}, there is no feedback from X_{2t} to X_{1t}.
- That is, any causality goes only in one direction and therefore X_{1t} can be considered as the *input* and X_{2t} can be considered as the *output*.

Vector Form

- The system of equation can be written in vector form as

$$\mathbf{X}_t = \Phi\mathbf{X}_{t-1} + \boldsymbol{\epsilon}_t$$

where $\boldsymbol{\epsilon}_t^T = (\epsilon_{1t}, \epsilon_{2t})$ and

$$\Phi = \begin{pmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{pmatrix}.$$

- The above equation looks like and AR(1) model except that $\mathbf{X}_t$ and $\boldsymbol{\epsilon}_t$ are now vectors instead of scalars.
- Since $\mathbf{X}_t$ depends on $\mathbf{X}_{t-1}$, this model is called a **vector autoregressive model** of order 1 (VAR(1)).
- The above equation can also be expressed as

$$(I - \Phi B)\mathbf{X}_t = \boldsymbol{\epsilon}_t$$

where B is the backward shift operator, and I is the (2×2) identity matrix and ΦB represents the operator matrix

$$\begin{pmatrix} \phi_{11}B & \phi_{12}B \\ \phi_{21}B & \phi_{22}B \end{pmatrix}.$$

- The necessary and sufficient condition for the stationarity of $\mathbf{X}_t$ is that the roots of the determinant of $I - \Phi B$ lie outside the unit circle.

5.3.2 VAR(p) models

- The above VAR(1) model can be generalized from two to m variables and from first-order auto-regression to pth order.
- A VAR model of order p ($VAR(p)$) can be written in the form

$$\Phi(B)\mathbf{X}_t = \boldsymbol{\epsilon}_t$$

where $\mathbf{X}_t$ is a $(m \times 1)$ vector of observed variables, and Φ is a matrix polynomial of order p in the backward shift operator B such that

$$\Phi(B) = I - \Phi_1 B - \cdots - \Phi_p B^p,$$

where I is the $(m \times m)$ identity matrix and $\Phi_1, \Phi_2, \dots, \Phi_p$ are $(m \times m)$ matrices of parameters.

- Since we restrict attention to stationary processes, without loss of generality, we assume the variables have been scaled to have zero mean.
- The condition for stationarity is that the roots of the equation

$$\text{determinant}\{\Phi(x)\} = |I - \Phi_1 x - \Phi_2 x^2 - \cdots - \Phi_p x^p| = 0,$$

should lie outside the unit circle.

- Let $\boldsymbol{\epsilon}_t = (\epsilon_{1t}, \epsilon_{2t}, \dots, \epsilon_{mt})$ denote and $(m \times 1)$ vector of random variables.
- This multivariate time series is called a **multivariate white noise** if it is stationary with zero mean vector $\mathbf{0}$, and if the values of $\boldsymbol{\epsilon}_t$ at different times are uncorrelated.
- Then the $(m \times m)$ matrix of the cross-covariances of the elements of $\boldsymbol{\epsilon}_t$ with that of $\boldsymbol{\epsilon}_{t+j}$ is given by

$$\text{Cov}(\boldsymbol{\epsilon}_t, \boldsymbol{\epsilon}_{t+j}) \begin{cases} \Gamma_0 & j = 0 \\ 0_m & j \neq 0, \end{cases}$$

where Γ_0 denotes a $(m \times m)$ symmetric positive-definite matrix and 0_m denotes an $(m \times m)$ matrix of zeros.

- Therefore, each component of $\boldsymbol{\epsilon}_t$ behaves like univariate white noise.
- Further, Γ_0, the covariance matrix at lag zero, does not need to be diagonal, as an innovation at a particular time point could affect more than one measured variable at that time point.
- Therefore, we allow the components of $\boldsymbol{\epsilon}_t$ to be contemporaneously correlated.

5.3.3 Vector ARMA models

- As in the univariate case, the VAR models can also be generalized to include moving average (MA) terms as

$$\Phi(B)\mathbf{X}_t = \Theta(B)\boldsymbol{\epsilon}_t$$

where

$$\Theta(B) = I + \Theta_1 B - \dots - \Theta_q B^p,$$

is a matrix polynomial of order q in the backward shift operator B and $\Theta_1, \Theta_2, \dots, \Theta_q$ are $(m \times m)$ matrices of parameters.

- Then $\mathbf{X}_t$ follows a **vector ARMA** (VARMA) model of order (p, q).
- The necessary and sufficient condition for the stationarity of $\mathbf{X}_t$ is that the roots of the determinant of $I - \Phi B$ lie outside the unit circle.
- The condition for invertibility is that the roots of the equation

$$\text{determinant}\{\Theta(x)\} = |I + \Theta_1 x + \Theta_2 x^2 + \dots + \Theta_p x^q| = 0,$$

lies outside the unit circle.

5.3.4 Vector ARIMA models

- If $\Phi(B)$ includes a factor of the form $I(1-B)$, then the model is not stationary and deal with the first differences of the components of $\mathbf{X}_t$.
- Such a model is called a **vector ARIMA** (VARIMA) model.
- However, in practice, it may not be optimal to difference each component of $\mathbf{X}_t$ in the same way and should consider the possible presence of co-integration before differencing multivariate data.
- VARMA models can be generalized further by adding terms, involving additional exogenous variables to the right hand side of the equation and they are known as VARIMAX models.

5.4 Fitting VAR and VARMA models

- The process involves assessing the order p and q of the model, estimating the parameter matrices and estimating the variance-covariance matrix of the noise components.

5.4.1 Forecasting

- Forecasts can be computed for VAR, VARMA and VARIMA models by a natural extension of methods used for univariate ARIMA models.
- Minimum mean square error (MMSE) forecasts can be obtained by replacing

- future values of white noise with zeros
- future values of $\mathbf{X}_t$ with MMSE forecasts
- present and past values of $\mathbf{X}_t$ with the observed values
- present and past values of ϵ_t with the one step head forecast residuals.

Example- Analysis of macro-economic series

Here we consider the U.S. quarterly gross domestic product (gdp), the civilian unemployment rate (unrate) and consumer price index (cpi) for all urban consumers from the first quarter of 1948 to the third quarter of 2017.

```
data <- read.csv(here::here("data", "macrots.csv" ))
data$quarter <- as.Date(data$quarter)
data <- data %>%
  select(-X) %>%
  as_tsibble(index = "quarter")
p1<- data %>% autoplot(gdp) +
  xlab ("Quarter")
p2<- data %>% autoplot(unrate) +
  xlab ("Quarter")
p3<- data %>% autoplot(cpi)+
  xlab ("Quarter")
p1/p2/p3
```

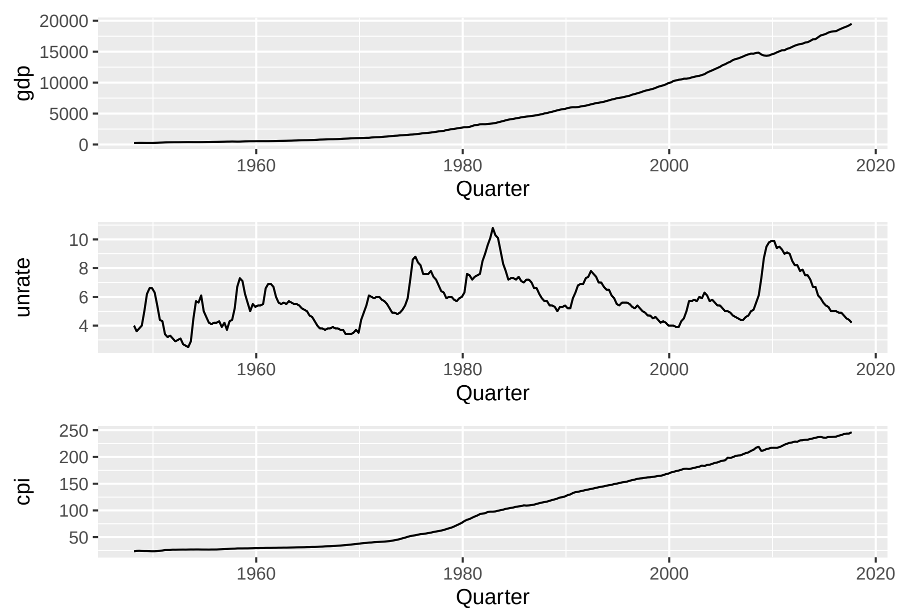


Figure 5.2: Time series plot of GDP, unemployment rate and CPI

- All the series display some level of nonstationarity (Figure 5.2).
- Therefore we transform the data by calculating the rate of change or difference for each series (Figure 5.3).

```
n<-nrow(data)
# Change rate of GDP
data$gdprate <- c(NA,diff(data$gdp)*100/data$gdp[1:(n-1)])
# The difference of unemployment rate
data$unemdiff <- c(NA,diff(data$unrate))
# Measure of inflation
data$cpirate <- c(NA,diff(data$cpi)*100/data$cpi[1:(n-1)])

p1<- data %>% autoplot(gdprate) +
  xlab ("Quarter")
p2<- data %>% autoplot(unemdiff) +
  xlab ("Quarter")
p3<- data %>% autoplot(cpirate)+
  xlab ("Quarter")
p1/p2/p3
```

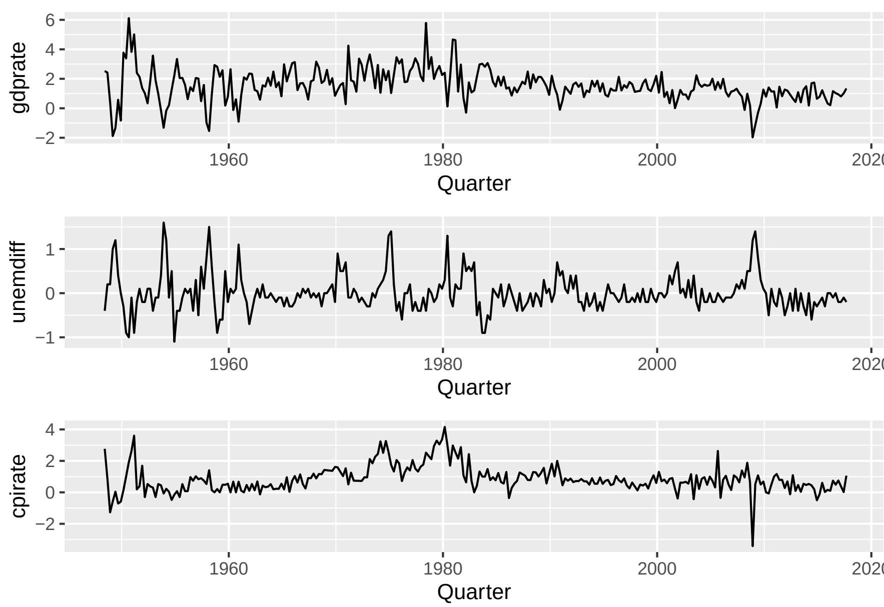


Figure 5.3: Time series plot of the difference series

- The scatterplot matrix in Figure 5.4 shows the cross-sectional dependence of the three series.

- Figure 5.4 shows a concurrent regression relationship between `gdprate` and `unemdiff`, `gdprate` and `cpirate`, respectively.

```
library(GGally)
ggpairs(data[,5:7])
```

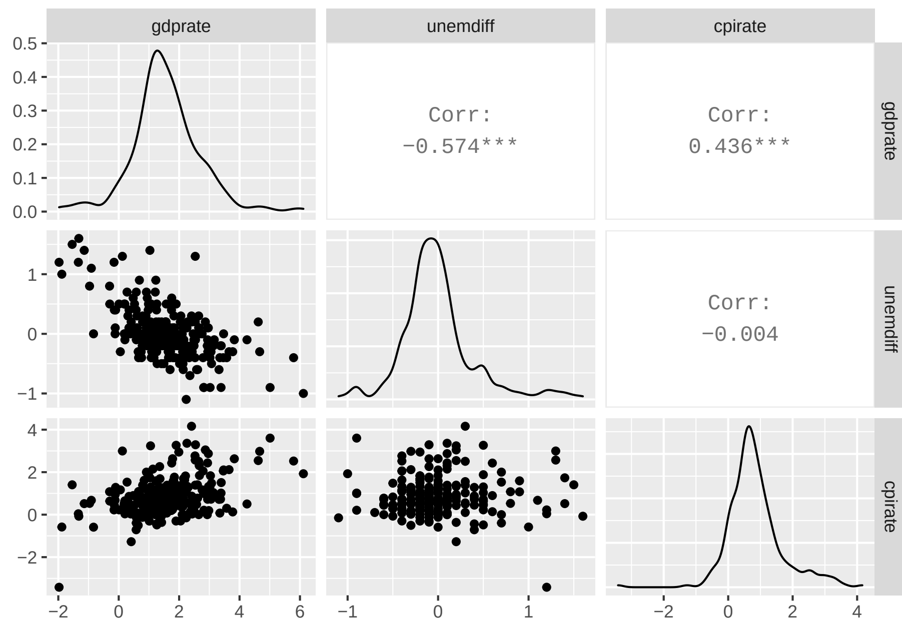
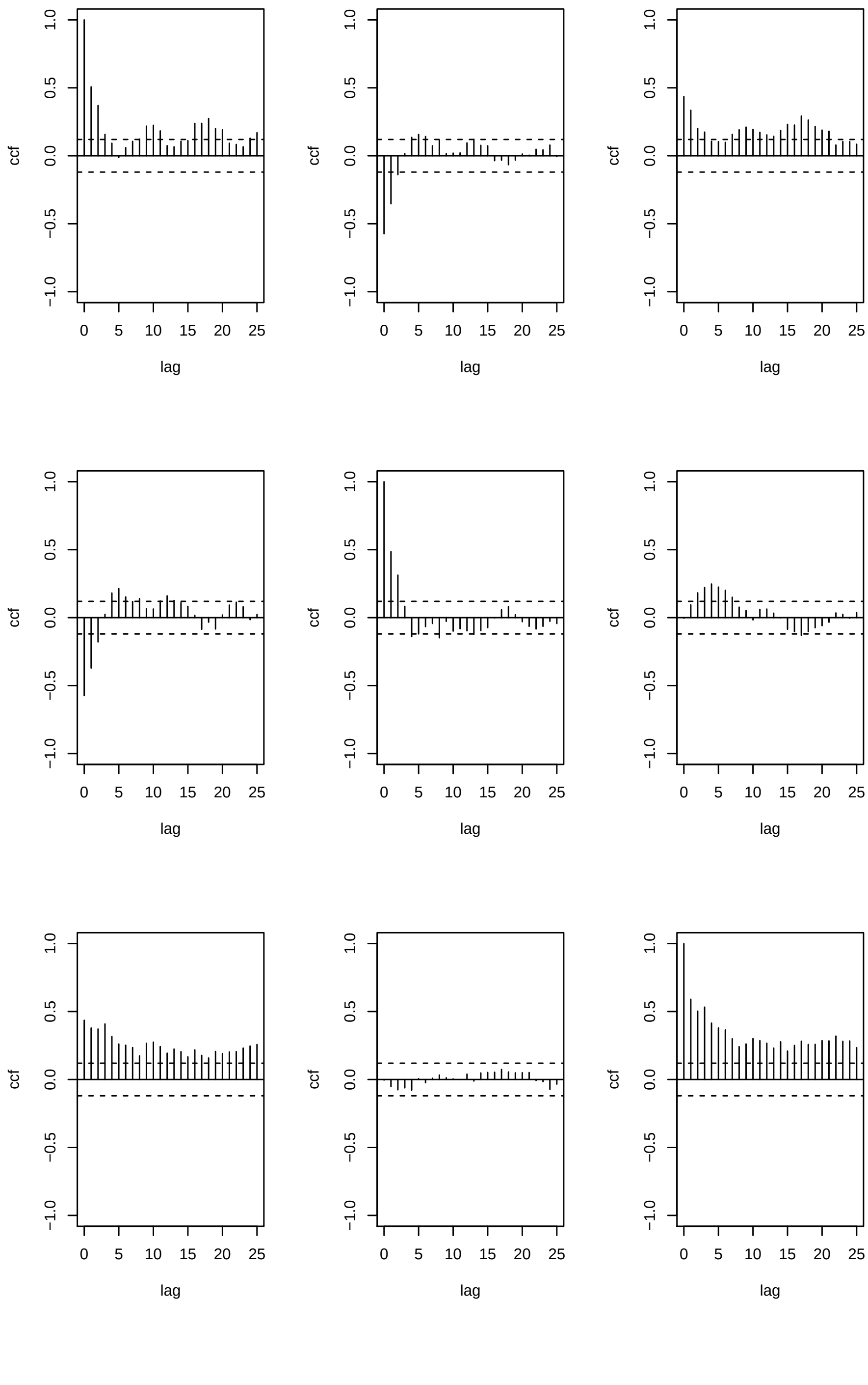
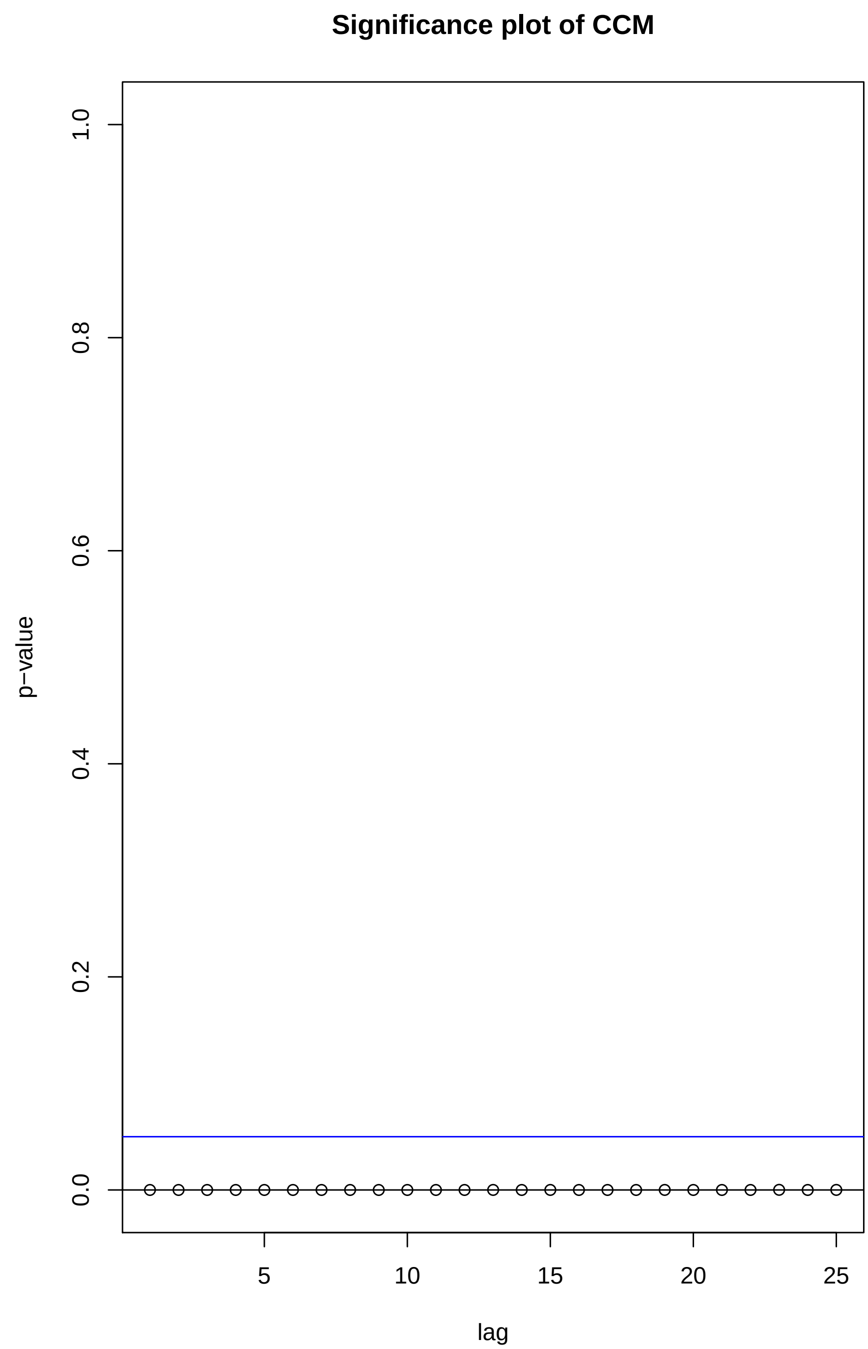


Figure 5.4: Scatterplot matrix

- Then we check the sample cross-correlations to see whether there are any lead or lag effects among the three series (Figure **??**.

```
data2 <- data %>%
  as_tibble() %>%
   select("gdprate", "unemdiff", "cpirate") %>%
   as.matrix()
data2 <- data2[-1,]
MTS::ccm(data2, lags = 25)
```


ccf
lag
1.0
0.5
0.0
−0.5
−1.0
0
5
10
15
20
25



- First, we consider a VAR(1) model for $\mathbf{X}_t$.

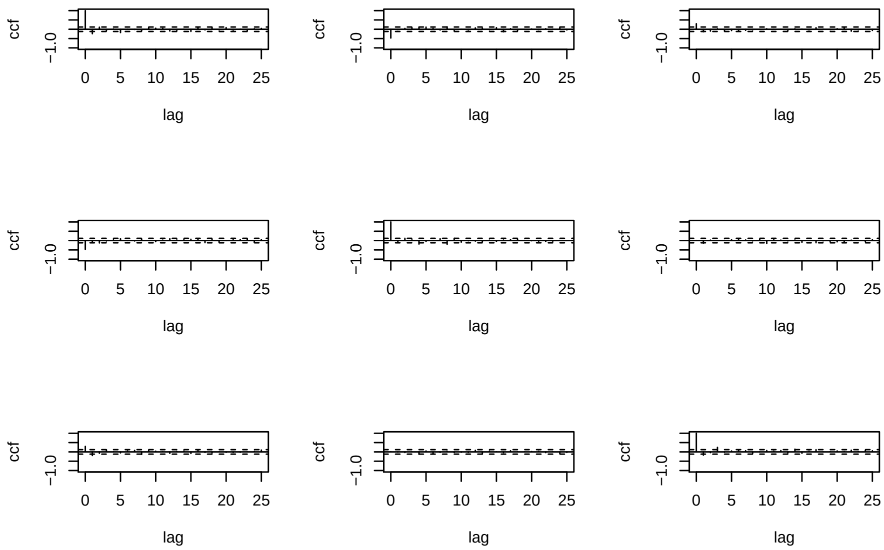
5.4. THE STRUCTURE OF VARMA MODELS

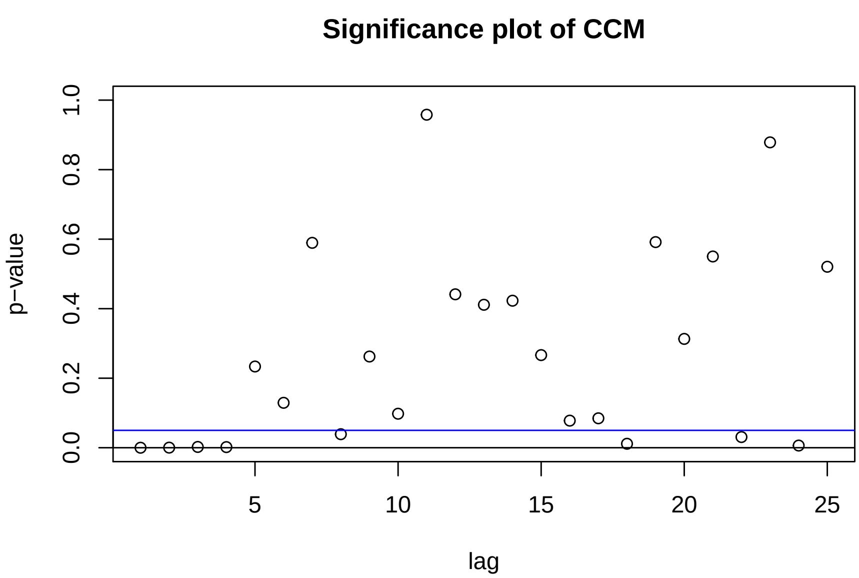
- We use the `VARMA` function in the `MTS` R package.

```r
var1_fit <- MTS::VARMA(data2, p=1, q=0, include.mean = FALSE, details = F)
```

```
## Number of parameters:  9
## initial estimates:  0.7327 0.1979 0.251 -0.077 0.3654 0.1182 0.2389 0.2681 0.5174
## Par. lower-bounds:  0.6185 -0.1504 0.0784 -0.1156 0.2477 0.0599 0.1651 0.0429 0.4058
## Par. upper-bounds:  0.8469 0.5463 0.4237 -0.0384 0.483 0.1765 0.3127 0.4933 0.629
## Final   Estimates:  0.7115 0.1508 0.2786 -0.06924 0.3811 0.1073 0.2309 0.2544 0.5149
##
## Coefficient(s):
##           Estimate  Std. Error  t value Pr(>|t|)
## gdprate    0.71155     0.05652   12.590  < 2e-16 ***
## unemdiff   0.15075     0.17319    0.870 0.384046
## cpirate    0.27860     0.08492    3.281 0.001035 **
## gdprate   -0.06924     0.01926   -3.595 0.000324 ***
## unemdiff   0.38114     0.05902    6.458 1.06e-10 ***
## cpirate    0.10733     0.02894    3.709 0.000208 ***
## gdprate    0.23086     0.03713    6.217 5.07e-10 ***
## unemdiff   0.25437     0.11379    2.235 0.025392 *
## cpirate    0.51490     0.05580    9.228  < 2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ---
## Estimates in matrix form:
## AR coefficient matrix
## AR( 1 )-matrix
##         [,1]  [,2]  [,3]
## [1,]  0.7115 0.151 0.279
## [2,] -0.0692 0.381 0.107
## [3,]  0.2309 0.254 0.515
##
## Residuals cov-matrix:
##         [,1]     [,2]    [,3]
## [1,]  1.0500 -0.16146 0.20992
## [2,] -0.1615  0.12194 0.00262
## [3,]  0.2099  0.00262 0.45331
## ----
## aic=  -3.138
## bic=  -3.02
```

```r
MTS::ccm(var1_fit$residuals, lags = 25)
```





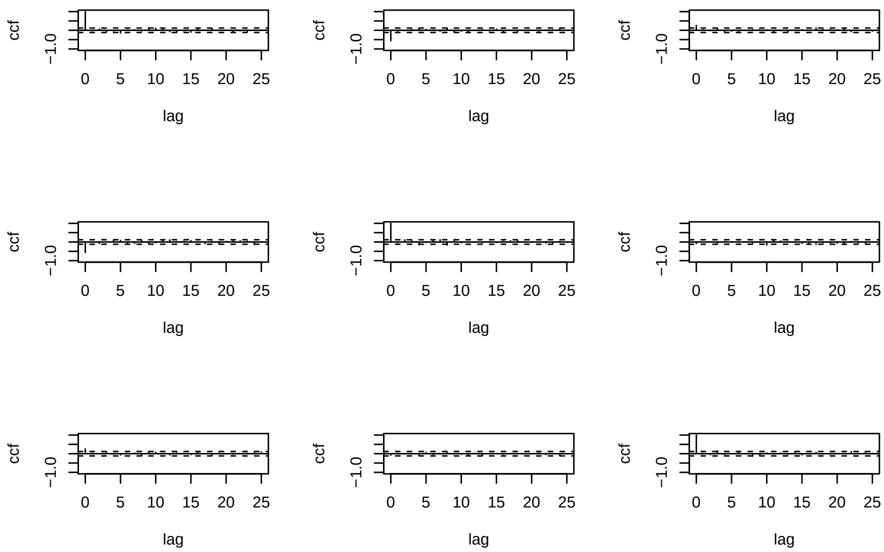
- We further consider fitting a VARMA model to the series $\mathbf{X}_t$.

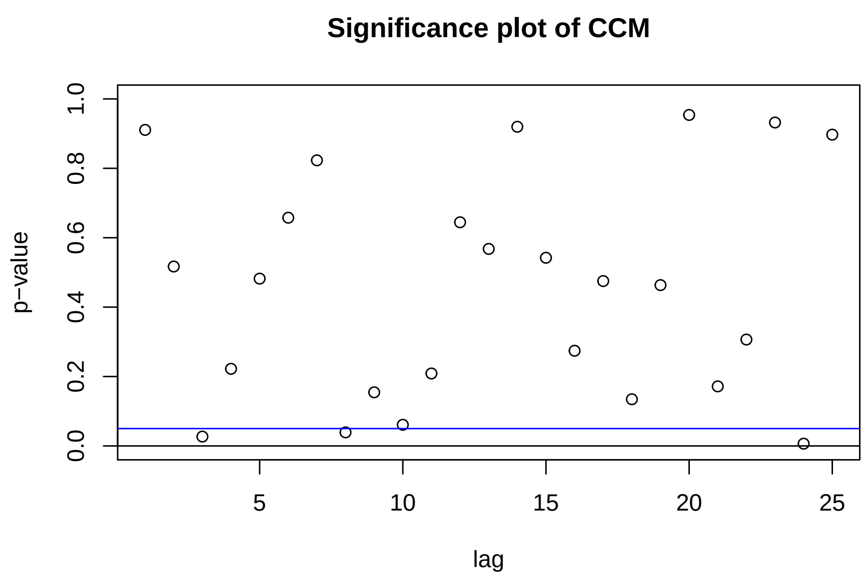
```r
#VARMA(1,1)
varma11_fit <- MTS::VARMA(data2, p=1, q=1, include.mean = FALSE, details = F)
```

```
## Number of parameters:  18
## initial estimates:  1.167 1.089 -0.3236 -0.1116 0.4118 0.1927 0.1568 0.118 0.7276 -0.899 -1.604 0.4924 0
## Par. lower-bounds:  0.9623 0.4671 -0.6495 -0.1855 0.1865 0.0748 0.0196 -0.3005 0.5085 -1.158 -2.351 0.11
## Par. upper-bounds:  1.371 1.712 0.0023 -0.0377 0.6371 0.3107 0.2941 0.5364 0.9467 -0.64 -0.8567 0.8731 0
## Final   Estimates:  1.332 1.373 -0.5913 -0.129 0.3544 0.2297 0.1203 0.1012 0.7905 -1.019 -1.758 0.7369 0
##
## Coefficient(s):
##           Estimate  Std. Error  t value Pr(>|t|)
## gdprate    1.33178     0.11647   11.434  < 2e-16 ***
## unemdiff   1.37330     0.36526    3.760 0.000170 ***
## cpirate   -0.59130     0.19045   -3.105 0.001904 **
## gdprate   -0.12900     0.04815   -2.679 0.007388 **
## unemdiff   0.35437     0.13210    2.683 0.007304 **
## cpirate    0.22966     0.07780    2.952 0.003160 **
## gdprate    0.12026     0.04966    2.422 0.015455 *
## unemdiff   0.10122     0.16779    0.603 0.546348
## cpirate    0.79055     0.08226    9.611  < 2e-16 ***
##           -1.01903     0.13581   -7.503 6.24e-14 ***
##           -1.75792     0.43938   -4.001 6.31e-05 ***
##            0.73690     0.22074    3.338 0.000843 ***
##            0.02455     0.05988    0.410 0.681827
##           -0.08370     0.14973   -0.559 0.576135
##           -0.19928     0.08924   -2.233 0.025548 *
##           -0.10262     0.06640   -1.545 0.122233
##           -0.26379     0.23668   -1.115 0.265059
##           -0.47173     0.10349   -4.558 5.16e-06 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ---
## Estimates in matrix form:
## AR coefficient matrix
## AR( 1 )-matrix
##        [,1]  [,2]   [,3]
## [1,]  1.332 1.373 -0.591
## [2,] -0.129 0.354  0.230
## [3,]  0.120 0.101  0.791
## MA coefficient matrix
## MA( 1 )-matrix
##         [,1]   [,2]   [,3]
## [1,]  1.0190 1.7579 -0.737
## [2,] -0.0245 0.0837  0.199
## [3,]  0.1026 0.2638  0.472
##
## Residuals cov-matrix:
##         [,1]     [,2]     [,3]
## [1,]  0.8230 -0.17354  0.14418
## [2,] -0.1735  0.11549 -0.01724
## [3,]  0.1442 -0.01724  0.38361
## ----
```

```
## aic=  -3.637
## bic=  -3.402
```

```r
MTS::ccm(varma11_fit$residuals, lags = 25)
```





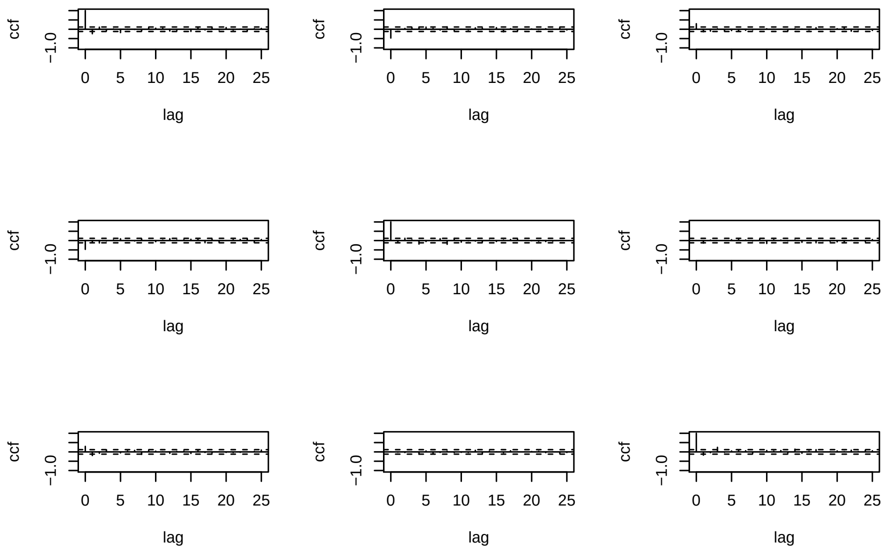
5.4.2 Order Selection

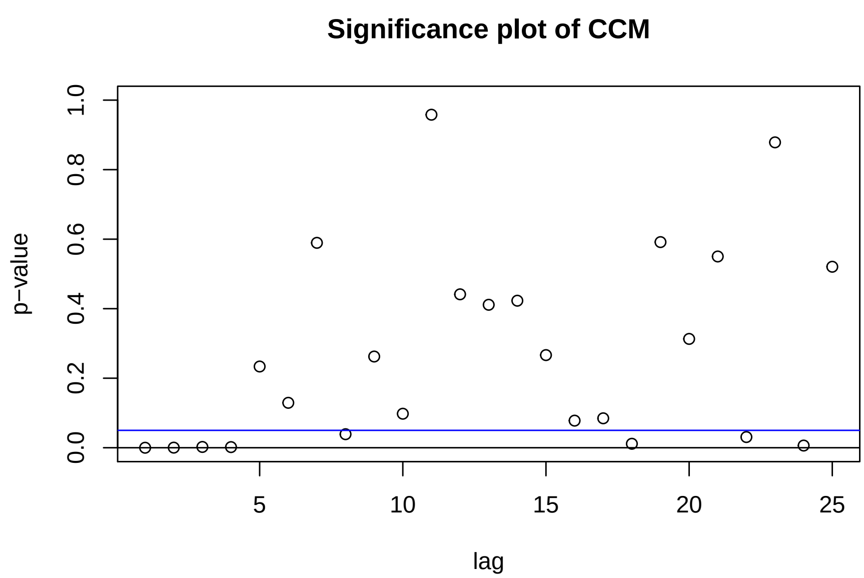
If you want some help selecting p and q

```
varma_auto_fit <- MTS::VARMA(data2, include.mean = FALSE, details = F)
```

```
## Number of parameters:  9
## initial estimates:  0.7327 0.1979 0.251 -0.077 0.3654 0.1182 0.2389 0.2681 0.5174
## Par. lower-bounds:  0.6185 -0.1504 0.0784 -0.1156 0.2477 0.0599 0.1651 0.0429 0.4058
## Par. upper-bounds:  0.8469 0.5463 0.4237 -0.0384 0.483 0.1765 0.3127 0.4933 0.629
## Final   Estimates:  0.7115 0.1508 0.2786 -0.06924 0.3811 0.1073 0.2309 0.2544 0.5149
##
## Coefficient(s):
##           Estimate  Std. Error  t value Pr(>|t|)
## gdprate    0.71155     0.05652   12.590  < 2e-16 ***
## unemdiff   0.15075     0.17319    0.870 0.384046
## cpirate    0.27860     0.08492    3.281 0.001035 **
## gdprate   -0.06924     0.01926   -3.595 0.000324 ***
## unemdiff   0.38114     0.05902    6.458 1.06e-10 ***
## cpirate    0.10733     0.02894    3.709 0.000208 ***
## gdprate    0.23086     0.03713    6.217 5.07e-10 ***
## unemdiff   0.25437     0.11379    2.235 0.025392 *
## cpirate    0.51490     0.05580    9.228  < 2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ---
## Estimates in matrix form:
## AR coefficient matrix
## AR( 1 )-matrix
##         [,1]  [,2]  [,3]
## [1,]  0.7115 0.151 0.279
## [2,] -0.0692 0.381 0.107
## [3,]  0.2309 0.254 0.515
##
## Residuals cov-matrix:
##         [,1]     [,2]    [,3]
## [1,]  1.0500 -0.16146 0.20992
## [2,] -0.1615  0.12194 0.00262
## [3,]  0.2099  0.00262 0.45331
## ----
## aic=  -3.138
## bic=  -3.02
```

```
MTS::ccm(varma_auto_fit$residuals, lags = 25)
```


ccf
−1.0
0 5 10 15 20 25
lag


Significance plot of CCM
p−value
0.0 0.2 0.4 0.6 0.8 1.0
5 10 15 20 25
lag

Generating a VARMA Process

```
set.seed(1234)
p1 <- matrix(c(0.2,-0.6,0.3,1,1),2,2)
sig <- matrix(c(4,0.8,0.8,1),2,2)
th1 <- matrix(c(-0.5,0,0,-0.6),2,2)
m1 <- VARMAsim(1000, arlags = c(1), malags = c(1), phi = p1,theta = th1, sigma = sig)
zt <- m1$series
head(zt)
```

```
##         [,1]     [,2]
## [1,] -2.7473 -2.33168
## [2,] -3.5389 -1.84248
## [3,] -0.9314 -0.04607
## [4,]  2.8290 -0.01222
## [5,]  0.8131 -2.33142
## [6,] -3.9268 -3.60356
```

```
varma_auto_fit <- MTS::VARMA(zt, include.mean = FALSE, details = F)
```

```
## Number of parameters:  4
## initial estimates:  0.4422 0.2961 -0.5508 1.038
## Par. lower-bounds:  0.3916 0.2557 -0.5769 1.017
## Par. upper-bounds:  0.4927 0.3364 -0.5248 1.059
## Final   Estimates:  0.4418 0.2976 -0.5501 1.039
##
## Coefficient(s):
##       Estimate  Std. Error  t value Pr(>|t|)
## [1,]   0.44176     0.02525    17.50   <2e-16 ***
## [2,]   0.29755     0.02018    14.74   <2e-16 ***
## [3,]  -0.55007     0.01299   -42.35   <2e-16 ***
## [4,]   1.03859     0.01038   100.04   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ---
## Estimates in matrix form:
## AR coefficient matrix
## AR( 1 )-matrix
##        [,1]  [,2]
## [1,]  0.442 0.298
## [2,] -0.550 1.039
##
## Residuals cov-matrix:
##        [,1]   [,2]
## [1,] 4.7286 0.9683
## [2,] 0.9683 1.2513
## ----
```

```
## aic=  1.613
## bic=  1.633
```

```r
VARorder(zt)
```

```
## selected order: aic =  6
## selected order: bic =  3
## selected order: hq =  3
## Summary table:
##        p   AIC   BIC    HQ      M(p) p-value
##  [1,]  0 4.642 4.642 4.642    0.0000  0.0000
##  [2,]  1 1.617 1.637 1.624 2982.6949  0.0000
##  [3,]  2 1.326 1.365 1.341  293.4415  0.0000
##  [4,]  3 1.257 1.316 1.279   75.6926  0.0000
##  [5,]  4 1.250 1.328 1.280   14.4993  0.0059
##  [6,]  5 1.253 1.351 1.290    4.5802  0.3331
##  [7,]  6 1.243 1.361 1.288   17.4699  0.0016
##  [8,]  7 1.244 1.381 1.296    6.9209  0.1401
##  [9,]  8 1.244 1.401 1.304    7.4485  0.1140
## [10,]  9 1.249 1.426 1.316    3.2249  0.5209
## [11,] 10 1.254 1.450 1.328    3.5016  0.4776
## [12,] 11 1.259 1.475 1.341    2.4851  0.6473
## [13,] 12 1.266 1.501 1.355    1.1209  0.8909
## [14,] 13 1.273 1.528 1.370    0.6265  0.9601
```

5.5 Granger Causality Tests

- Let F_t be the available information at time t
- Let $F_{-i,t}$ be F_t where all information regarding the the ith components, X_{it} removed.
- Consider the bivariate VAR(1) model

$$\begin{pmatrix} X_{1t} \\ X_{2t} \end{pmatrix} = \begin{pmatrix} \phi_{10} \\ \phi_{20} \end{pmatrix} + \begin{pmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{pmatrix} \begin{pmatrix} X_{1,t-1} \\ X_{2,t-1} \end{pmatrix} + \begin{pmatrix} \epsilon_{1t} \\ \epsilon_{2t} \end{pmatrix}.$$

- Consider the h-step ahead forecast $X_t(h)$ based on F_t and the associated forecast error $e_t(h)$.
- Let $X_{j,t+h}|F_{-i,t}$ be the h-step ahead prediction of $X_{j,t+h}$ based on $F_{-i,t}$.
- Let $e_{j,t+h}|F_{-i,t}$ be the associated forecast error where $i \neq j$.
- We say that X_{1t} causes X_{2t}, if the bivariate forecast for X_{2t} is more accurate than its univariate forecast.
- That is, X_{1t} causes X_{2t} if

$$Var[e_{2t}(h)] < Var[e_{2,t+h}|F_{-1,t}].$$

5.5.1 Test for Granger causality

- VAR models can be used to investigate lead-lag behaviour.
- The bivariate VAR(p) model can be expressed as

$$x_t = c_1 + \sum_{i=1}^{p} \alpha_{1i} x_{t-i} + \sum_{i=1}^{p} \beta_{1i} y_{t-i} + \epsilon_{1t}$$

$$y_t = c_2 + \sum_{i=1}^{p} \alpha_{2i} x_{t-i} + \sum_{i=1}^{p} \beta_{2i} y_{t-i} + \epsilon_{2t}$$

- The test for Granger causality from x to y is an F-test for the joint significance of $\alpha_{21}, \dots, \alpha_{2p}$, in an OLS regression.
- Similarly, Granger causality from y to x is an F-test for the joint significance of $\beta_{11}, \dots, \beta_{1p}$.

Test for Granger Causality in R

```
data3 <-  data %>% as_tibble() %>%
  select("gdprate", "cpirate") %>%
  as.matrix()
data3 <- ts(data = data3, start=c(1948,1), frequency = 4 )
head(data3)
```

```
##      gdprate  cpirate
## [1,]      NA       NA
## [2,]  2.5262  2.76596
## [3,]  2.4185  0.86957
## [4,]  0.4147 -1.27258
## [5,] -1.8834 -0.58212
## [6,] -1.3357  0.04182
```

```
library(lmtest)
grangertest(gdprate ~ cpirate, order = 3, data = data3)
```

```
## Granger causality test
##
## Model 1: gdprate ~ Lags(gdprate, 1:3) + Lags(cpirate, 1:3)
## Model 2: gdprate ~ Lags(gdprate, 1:3)
##   Res.Df Df    F Pr(>F)
## 1    268
## 2    271 -3 2.35  0.072 .
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(cpirate ~ gdprate, order = 3, data = data3)
```

```
## Granger causality test
##
## Model 1: cpirate ~ Lags(cpirate, 1:3) + Lags(gdprate, 1:3)
## Model 2: cpirate ~ Lags(cpirate, 1:3)
##   Res.Df Df    F Pr(>F)
## 1    268
## 2    271 -3 4.69 0.0033 **
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## alternative ways of specifying the same test
grangertest(data3, order = 3)
```

```
## Granger causality test
##
## Model 1: cpirate ~ Lags(cpirate, 1:3) + Lags(gdprate, 1:3)
## Model 2: cpirate ~ Lags(cpirate, 1:3)
##   Res.Df Df    F Pr(>F)
## 1    268
## 2    271 -3 4.69 0.0033 **
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(data3[, 1], data3[, 2], order = 3)
```

```
## Granger causality test
##
## Model 1: data3[, 2] ~ Lags(data3[, 2], 1:3) + Lags(data3[, 1], 1:3)
## Model 2: data3[, 2] ~ Lags(data3[, 2], 1:3)
##   Res.Df Df    F Pr(>F)
## 1    268
## 2    271 -3 4.69 0.0033 **
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Using `MTS` package

Perform VAR(p) and constrained VAR(p) estimations to test the Granger causality. It uses likelihood ratio and asymptotic chi-square.

```
data3 <-  data %>% as_tibble() %>%
  select("gdprate", "cpirate") %>%
  drop_na() %>%
  as.matrix()
MTS::GrangerTest(data3,p=3,include.mean=T,locInput=c(1))
```

```
## Number of targeted zero parameters:  3
## Chi-square test for Granger Causality and p-value:  7.065 0.06986
## Constant term:
## Estimates:  0.7174 0.004604
## Std.Error:  0.1149 0.0751
## AR coefficient matrix
## AR( 1 )-matrix
##        [,1]  [,2]
## [1,] 0.4498 0.000
## [2,] 0.0924 0.315
## standard error
##        [,1]  [,2]
## [1,] 0.0600 0.000
```

```
## [2,] 0.0407 0.059
## AR( 2 )-matrix
##        [,1]  [,2]
## [1,] 0.2053 0.000
## [2,] 0.0198 0.117
## standard error
##        [,1]   [,2]
## [1,] 0.0647 0.0000
## [2,] 0.0429 0.0617
## AR( 3 )-matrix
##         [,1]  [,2]
## [1,] -0.1135 0.000
## [2,]  0.0679 0.236
## standard error
##       [,1]   [,2]
## [1,] 0.060 0.0000
## [2,] 0.041 0.0575
##
## Residuals cov-mtx:
##        [,1]   [,2]
## [1,] 0.8791 0.1577
## [2,] 0.1577 0.3619
##
## det(SSE) =  0.2932
## AIC =  -1.162
## BIC =  -1.045
## HQ  =  -1.115
```

- Different researches show different results about the causality between economic growth and inflation.
- Fisher (1993) shows that causality goes from inflation to economic growth.
- In contrast to Fisher's findings, Umaru and Zubariu (2011) found that Nigeria's GDP causes inflation and not inflation causing GDP using Granger causality test.
- Studies also show that the causality relation can be different in the short run and in the long run.
- A study conducted by Datta and Chanda(2011) on Malaysia shows that causality exist between inflation and economic growth in the short run and direction of causality is from inflation to economic growth but in the long run economic growth causes inflation.

5.6 Cointegration

This chapter is heavily based on Stock and Watson (2015).

- Two or more time series with stochastic trends can move together very closely over the long run.

- Two or more time series that have a common stochastic trend are said to be **cointegrated**.

Definition (Stock and Watson (2015))

Suppose X_t and Y_t are integrated of order one. If, for some coefficient θ, $Y_t - \theta X_t$ is integrated of order zero, then X_t and Y_t are said to be **cointegrated**. The coefficient θ is called the **cointegrating coefficient**.

That is, X_t and Y_t are $I(1)$ and if there is a θ such that $Y_t - \theta X_t$ is $I(0)$, X_t and Y_t are cointegrated

- Put differently, cointegration of X_t and Y_t means that X_t and Y_t have the same or a common stochastic trend and this trend can be eliminated by taking a specific difference of the series such that the resulting series is stationary.
- R functions for cointegration analysis are implemented in the package `urca`

Example

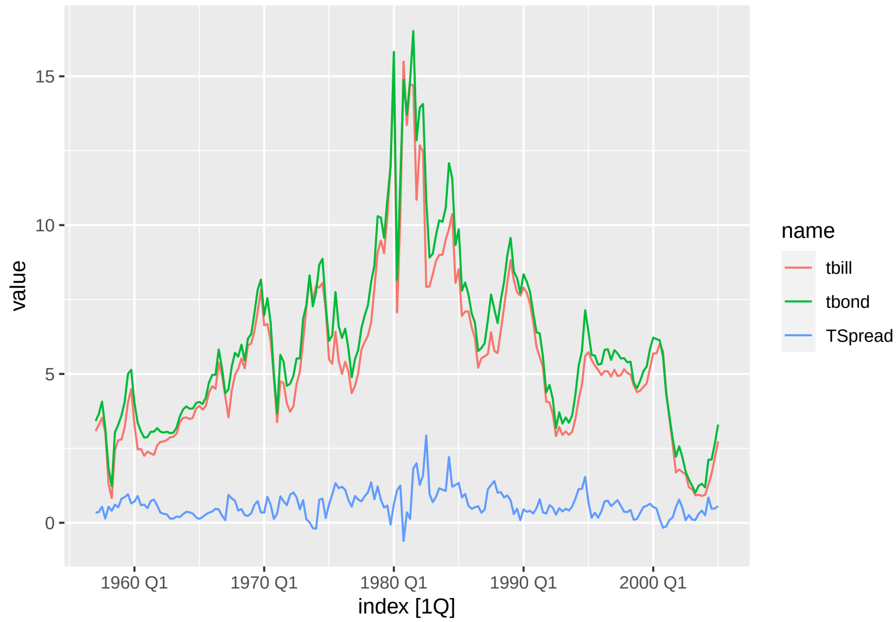
Consider the the relation between 3-month treasury bills, U.S. 10-years treasury bonds and the spread in their interest rates.

- Interest rates on long-term and short term treasury bonds are closely linked to macroeconomic conditions.
- While interest rates on both types of bonds have the same long-run tendencies, they behave quite differently in the short run.
- The difference in interest rates of two bonds with distinct maturity is called the term spread.

```
# AER package Provides functions, data sets,
# examples, and demos for Applied Econometrics
library(AER)
# A quarterly multiple time series from 1947(1) to 2004(4) with 2 variables, GDP and T-bill
data("USMacroSW")

# tbill: 3-months Treasury bills interest rate
# tbond: 10-years Treasury bonds interest rate
USMacroSW <- USMacroSW %>% as_tsibble(pivot_longer = FALSE) %>%
  select(index, tbill, tbond) %>%
  mutate(TSpread = tbond - tbill) %>%
  pivot_longer(cols = tbill:TSpread)

p <- USMacroSW %>% autoplot(value)
print(p)
```



- The figure suggests that long-term and short-term interest rates are cointegrated as both interest series seem to have the same long-run behavior.
- They share a common stochastic trend.
- The term spread, which is obtained by taking the difference between long-term and short-term interest rates, seems to be stationary.

5.6.1 Testing for Cointegration

- There are three ways to decide whether two variables can plausibly be modeled as cointegrated
  - use expert knowledge and economic theory
  - graph the series and check whether they appear to have a common stochastic trend
  - perform statistical tests for cointegration
- The unit root testing procedures discussed so far can be extended to test for cointegration.
- The insight on which these tests are based is that if two series Y_t and X_t are cointegrated, the series obtained by taking the difference $Y_t - \theta X_t$ must be stationary. - If the series are not cointegrated, $Y_t - \theta X_t$ is nonstationary $[I(1)]$.
- The hypothesis that Y_t and X_t are not cointegrated $[i.e.\ \ Y_t - \theta X_t$ is $I(1)$ $]$ therefore can be tested by testing the null hypothesis that $Y_t - \theta X_t$ has a unit root.
- If the hypothesis is rejected, then Y_t and X_t can be modeled as cointegrated.
- The details of this test depend on whether the cointegrating coefficient θ is known.

5.6.1.1 Testing for cointegration when θ is known

- In some situations expert knowledge or economic theory can be used to suggest values of θ.
- When θ is known, the Dickey-Fuller and DF-GLS unit root tests can be used to test for cointegration by first constructing the series $z_t = Y_t - \theta X_t$ and then testing the null hypothesis that z_t has a unit autoregressive root.

5.6.1.2 Testing for cointegration when θ is unknown

- If θ is unknown, it must be estimated before the unit root test can be applied.
- This can be done by first estimating the cointegrating coefficient θ by OLS estimation of the regression

$$Y_t = \alpha + \theta X_t + z_t.$$

- In the second step, a Dickey-Fuller t-test (with an intercept but no time trend) is used to test for a unit root in the residual from this regression, $\hat{z}_t$.
- This two-step procedure is called the Engle-Granger Augmented Dickey-Fuller test for cointegration or **EG-ADF test**.

Extensions to Multiple Cointegrated Variables

- If there are three variables, Y_t, X_{1t} and X_{2t}, each of which is $I(1)$, then they are cointegrated with cointegrating coefficients θ_1 and θ_2 if $Y_t - \theta_1 X_{1t} - \theta_2 X_{2t}$ is stationary.
- When there are three or more variables, there can be multiple cointegrating relationships.
- The EG-ADF procedure for testing a single cointegrating relationship among multiple variables is the same as for the case of two variables, except that the regression equation is modified so that both X_{1t} and X_{2t} are regressors.
- Tests for multiple cointegrating relationships can be performed using the system methods, such as Johansen's (1988) method.

5.6.2 Vector Error Correction Models

- In the previous chapters, we eliminated the stochastic trend in an $I(1)$ series Y_t by computing its first difference, ΔY_t.
- The problem created by stochastic trends were then avoided by using ΔY_t instead of Y_t in time series regression.
- However, if X_t and Y_t are cointegrated, there is another way to eliminate the common trend from the difference.
- Because the term $Y_t - \theta X_t$ is stationary, it too can be used in regression analysis.
- In fact, if X_t and Y_t are cointegrated, the first differences of X_t and Y_t can be modeled using a VAR, augmented by including $Y_{t-1} - \theta X_{t-1}$ as an additional regressor:

$$\Delta Y_t = \beta_{10} + \beta_{11}\Delta Y_{t-1} + \cdots + \beta_{1p}\Delta Y_{t-p} + \gamma 11\Delta X_{t-1} + \cdots + \gamma_{1p}\Delta X_{t-p} + \alpha_1(Y_{t-1} - \theta X_{t-1}) + u_{1t}$$

$$\Delta X_t = \beta_{20} + \beta_{21}\Delta Y_{t-1} + \cdots + \beta_{2p}\Delta Y_{t-p} + \gamma 21\Delta X_{t-1} + \cdots + \gamma_{2p}\Delta X_{t-p} + \alpha_2(Y_{t-1} - \theta X_{t-1}) + u_{2t}$$

- The term $Y_t - \theta X_t$ is called the **error correction term**.
- The combined model in the above two equations is called a **vector error correction model (VECM)**.
- In a VECM, past values of $Y_t - \theta X_t$, help to predict future values of ΔY_t and/or ΔX_t.

5.7 References:

- Chatfield, C., & Xing, H. (2019). The analysis of time series: an introduction with R. CRC press.
- Kleiber, C., & Zeileis, A. (2008). Applied econometrics with R. Springer Science & Business Media.
- Stock, J. H., & Watson, M. W. (2015). Introduction to econometrics.
- Tsay, R. S. (2013). Multivariate time series analysis: with R and financial applications. John Wiley & Sons.

Bibliography

Brockwell, P. J., Brockwell, P. J., Davis, R. A., and Davis, R. A. (2016). *Introduction to time series and forecasting*. Springer.

Chatfield, C. and Xing, H. (2019). *The analysis of time series: an introduction with R*. CRC press.

Stock, J. H. and Watson, M. W. (2015). *Introduction to econometrics*.

Tsay, R. S. (2010). *Analysis of Financial Time Series*. John Wiley & Sons.

Tsay, R. S. (2013). *Multivariate time series analysis: with R and financial applications*. John Wiley & Sons.