Economic Forecasting

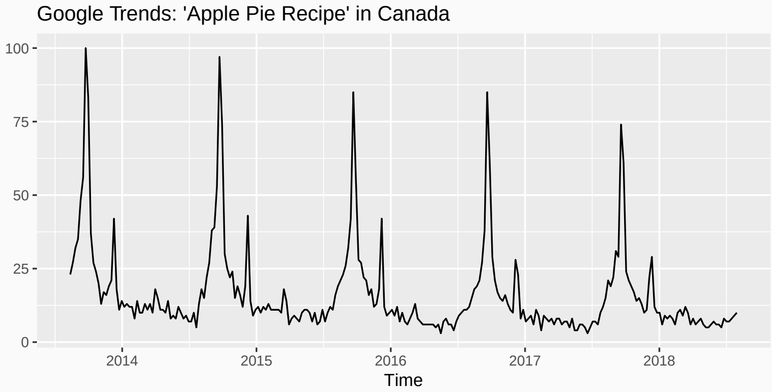
Seasonal ARIMA models

Sebastian Fossati

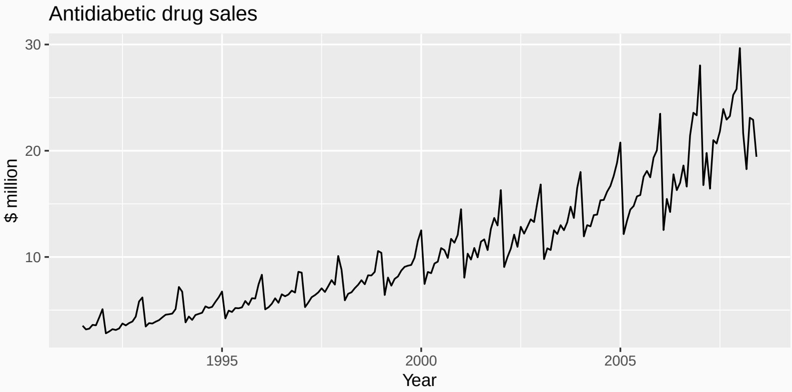
University of Alberta | E493 | 2023

Outline

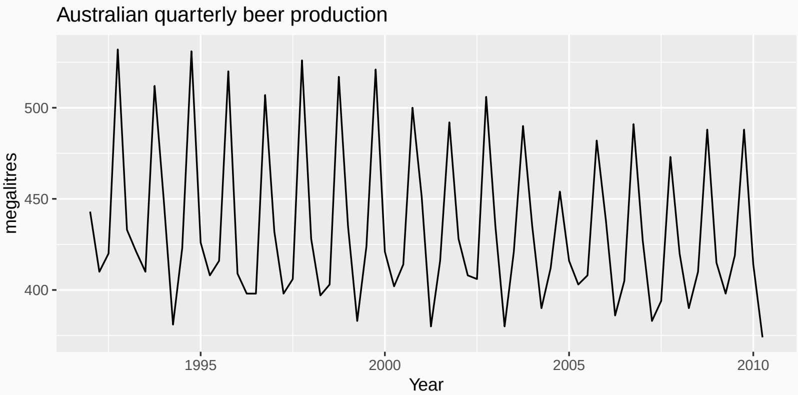
Seasonal data



Seasonal data

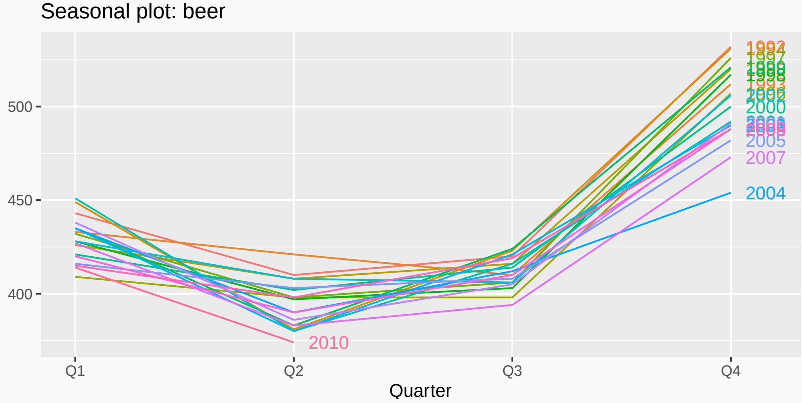


Seasonal data



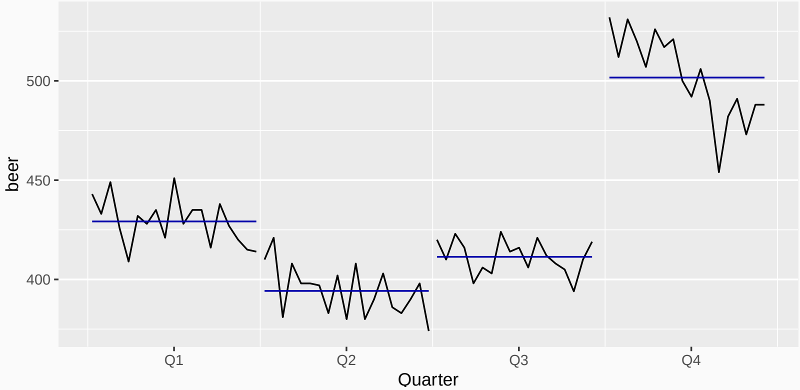
Seasonal data

```r
# season plot
ggseasonplot(beer, year.labels = TRUE)
```



Seasonal data

```r
# subseries plot
ggsubseriesplot(beer)
```



Time series patterns

Trend: pattern exists when there is a long-term increase or decrease in the data

Seasonal: pattern exists when a series is influenced by seasonal factors (eg, the quarter of the year, the month, or day of the week)

Cyclic: pattern exists when data exhibit rises and falls that are *not of fixed period* (duration usually of at least 2 years)

Seasonal or cyclic?

Differences between seasonal and cyclic patterns:

- seasonal pattern constant length; cyclic pattern variable length
- average length of cycle longer than length of seasonal pattern
- magnitude of cycle more variable than magnitude of seasonal pattern

Seasonal or cyclic?

Differences between seasonal and cyclic patterns:

- seasonal pattern constant length; cyclic pattern variable length
- average length of cycle longer than length of seasonal pattern
- magnitude of cycle more variable than magnitude of seasonal pattern

The timing of peaks and troughs is predictable with seasonal data, but unpredictable in the long term with cyclic data.

Outline

Time series decomposition

$$y_t = f(S_t, T_t, R_t)$$

$$\begin{aligned} \text{where} \quad y_t &= \quad \text{data at period } t \\ T_t &= \quad \text{trend-cycle component at period } t \\ S_t &= \quad \text{seasonal component at period } t \\ R_t &= \quad \text{remainder component at period } t \end{aligned}$$

Time series decomposition

$$y_t = f(S_t, T_t, R_t)$$

where	$y_t =$	data at period t
	$T_t =$	trend-cycle component at period t
	$S_t =$	seasonal component at period t
	$R_t =$	remainder component at period t

Additive decomposition: $y_t = S_t + T_t + R_t$.

Multiplicative decomposition: $y_t = S_t \times T_t \times R_t$.

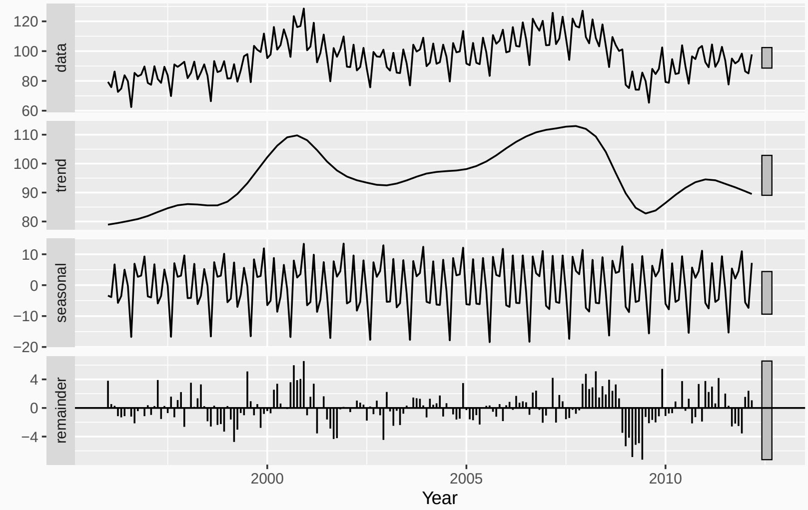
Time series decomposition

Remarks:

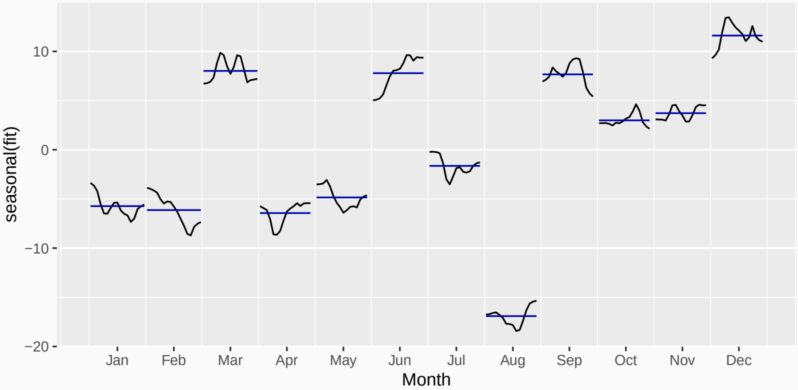
- additive model appropriate if magnitude of seasonal fluctuations does not vary with level
- if seasonal are proportional to level of series, then multiplicative model appropriate
- multiplicative decomposition more prevalent with economic series
- logs turn multiplicative relationship into an additive relationship:

$$y_t = S_t \times T_t \times E_t \quad \Rightarrow \quad \log y_t = \log S_t + \log T_t + \log R_t$$

Euro electrical equipment



Euro electrical equipment



Euro electrical equipment

Decompositions in R

To decompose a time series into seasonal, trend and irregular components we can use the `stl()` function.

Helper functions:

- `seasonal()` extracts the seasonal component
- `trendcycle()` extracts the trend-cycle component
- `remainder()` extracts the remainder component
- `seasadj()` returns the seasonally adjusted series

Seasonal adjustment

Useful by-product of decomposition: an easy way to calculate seasonally adjusted data.

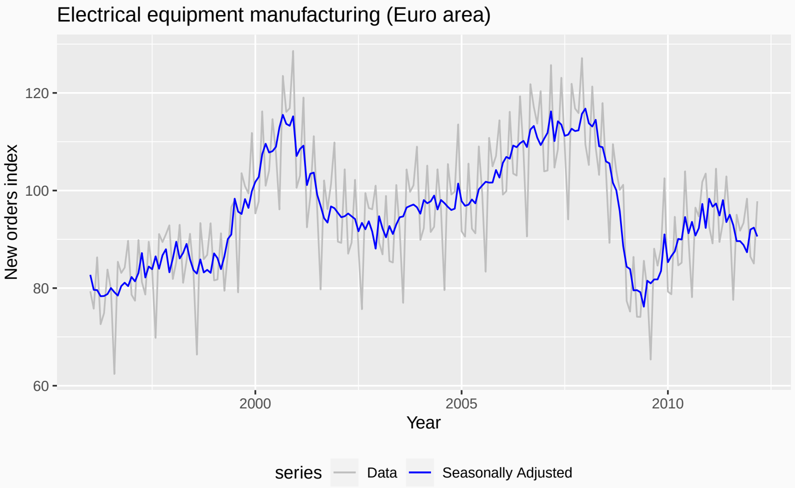
- additive decomposition: seasonally adjusted data given by

$$y_t - S_t = T_t + R_t$$

- multiplicative decomposition: seasonally adjusted data given by

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

Euro electrical equipment



Seasonal adjustment

Remarks:

- we use estimates of S based on past values to seasonally adjust a current value
- seasonally adjusted series reflect **remainders** as well as **trend**

History of time series decomposition

History:

- Classical method originated in 1920s
- Census II method introduced in 1957 (basis for X-11 method and X-12-ARIMA, X-13-ARIMA variants)
- STL method introduced in 1983
- TRAMO/SEATS introduced in 1990s

History of time series decomposition

History:

- Classical method originated in 1920s
- Census II method introduced in 1957 (basis for X-11 method and X-12-ARIMA, X-13-ARIMA variants)
- STL method introduced in 1983
- TRAMO/SEATS introduced in 1990s

National Statistics Offices:

- US Census Bureau uses X-13-ARIMA-SEATS
- Statistics Canada uses X-12-ARIMA
- EuroStat use X-13-ARIMA-SEATS

Outline

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

Non-seasonal ARIMA models

ARIMA(1,1,1) model:

$$\begin{array}{ccccc} (1 - \phi_1 B) & (1 - B) y_t & = & c + (1 + \theta_1 B)\varepsilon_t \\ \uparrow & \uparrow & & \uparrow \\ \text{AR(1)} & \text{First} & & \text{MA(1)} \\ & \text{difference} & & \end{array}$$

Non-seasonal ARIMA models

ARIMA(1,1,1) model:

$$\underbrace{(1-\phi_1 B)}_{\text{AR(1)}} \quad \underbrace{(1-B)}_{\text{First difference}} y_t = c + \underbrace{(1+\theta_1 B)}_{\text{MA(1)}}\varepsilon_t$$

Written out:

$$\begin{aligned}
y_t &= c + y_{t-1} + \phi_1 y_{t-1} - \phi_1 y_{t-2} + \theta_1\varepsilon_{t-1} + \varepsilon_t \\
y_t - y_{t-1} &= c + \phi_1(y_{t-1} - y_{t-2}) + \theta_1\varepsilon_{t-1} + \varepsilon_t \\
y'_t &= c + \phi_1 y'_{t-1} + \theta_1\varepsilon_{t-1} + \varepsilon_t
\end{aligned}$$

Seasonal ARIMA models

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

where m = number of observations per year.

Seasonal ARIMA models

E.g., ARIMA$(1, 1, 1)(1, 1, 1)_4$ model (without constant)

Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)$_4$ model (without constant)

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

Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)$_4$ model (without constant)

$$\underbrace{(1-\phi_1 B)}_{\begin{pmatrix}\text{Non-seasonal}\\ \text{AR(1)}\end{pmatrix}}\underbrace{(1-\Phi_1 B^4)}_{\begin{pmatrix}\text{Seasonal}\\ \text{AR(1)}\end{pmatrix}}\underbrace{(1-B)}_{\begin{pmatrix}\text{Non-seasonal}\\ \text{difference}\end{pmatrix}}\underbrace{(1-B^4)}_{\begin{pmatrix}\text{Seasonal}\\ \text{difference}\end{pmatrix}}y_t = \underbrace{(1+\theta_1 B)}_{\begin{pmatrix}\text{Non-seasonal}\\ \text{MA(1)}\end{pmatrix}}\underbrace{(1+\Theta_1 B^4)}_{\begin{pmatrix}\text{Seasonal}\\ \text{MA(1)}\end{pmatrix}}\varepsilon_t.$$

Seasonal ARIMA models

E.g., 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:

$$\begin{aligned} y_t &= (1 + \phi_1)y_{t-1} - \phi_1 y_{t-2} + (1 + \Phi_1)y_{t-4} \\ &\quad - (1 + \phi_1 + \Phi_1 + \phi_1\Phi_1)y_{t-5} + (\phi_1 + \phi_1\Phi_1)y_{t-6} \\ &\quad - \Phi_1 y_{t-8} + (\Phi_1 + \phi_1\Phi_1)y_{t-9} - \phi_1\Phi_1 y_{t-10} \\ &\quad + \varepsilon_t + \theta_1\varepsilon_{t-1} + \Theta_1\varepsilon_{t-4} + \theta_1\Theta_1\varepsilon_{t-5}. \end{aligned}$$

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

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:

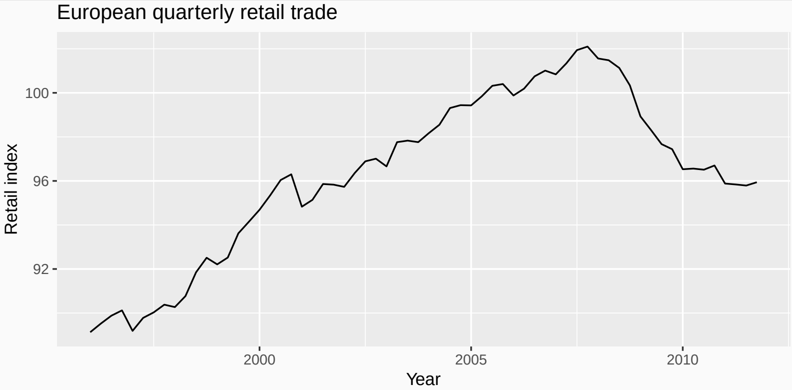
- a spike at lag 12 in the ACF but no other significant spikes
- the PACF will show exponential decay in the seasonal lags; that is, at lags 12, 24, 36, . . .

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

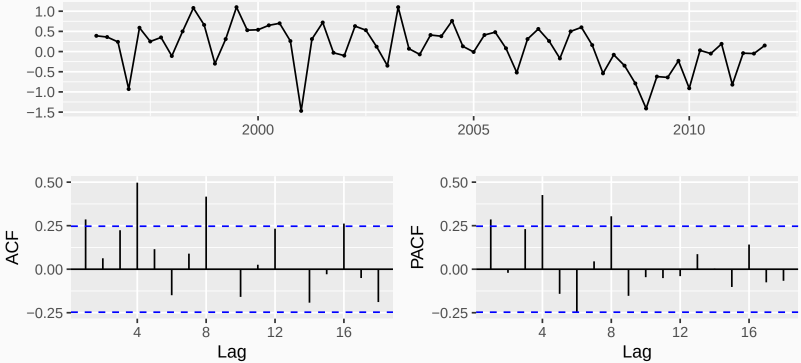
European quarterly retail trade

```r
# retail data
autoplot(euretail) + xlab("Year") + ylab("Retail index") +
  ggtitle("European quarterly retail trade")
```



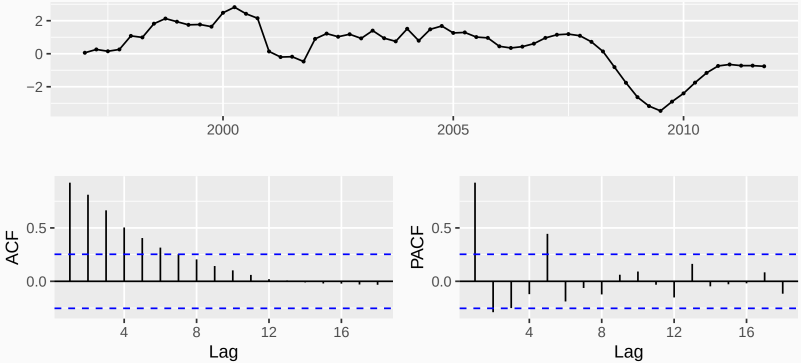
European quarterly retail trade

```r
# first diff
ggtsdisplay(diff(euretail, lag = 1))
```



European quarterly retail trade

```r
# seasonal diff
ggtsdisplay(diff(euretail, lag = 4))
```



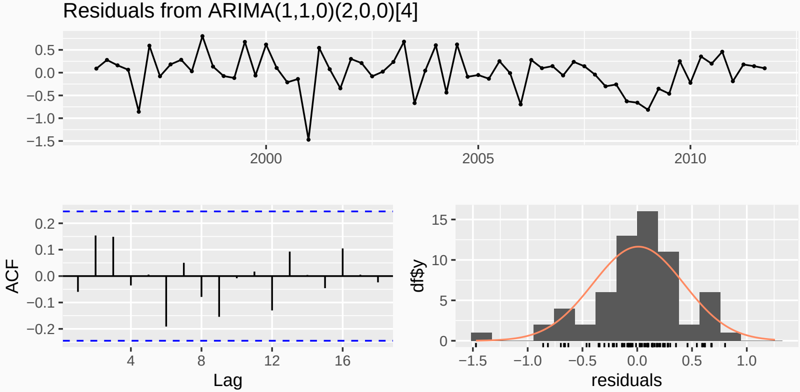
European quarterly retail trade

Remarks:

- $d = 1$ seems necessary (maybe $D = 1$ as well?)
- one significant spike at lag 1 in ACF and PACF suggests non-seasonal AR(1) or MA(1) component
- two significant spikes at lag 4 and 8 in PACF suggests seasonal AR(2) component
- initial candidate model: ARIMA$(1,1,0)(2,0,0)_4$

European quarterly retail trade

```r
fit1 <- Arima(euretail, order=c(1,1,0), seasonal=c(2,0,0))
checkresiduals(fit1, lag=24, test=FALSE)
```



European quarterly retail trade

```r
# residuals
checkresiduals(fit1, lag=24, plot=FALSE)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,1,0)(2,0,0)[4]
## Q* = 14, df = 21, p-value = 0.9
##
## Model df: 3.   Total lags used: 24
```

European quarterly retail trade

Models with lowest AICc values tend to give slightly better results than the other models.

- AICc of ARIMA$(1,1,0)(2,0,0)_4$ model is 79.65
- AICc of ARIMA$(0,1,1)(2,0,0)_4$ model is 82.32
- AICc of ARIMA$(1,1,1)(2,0,0)_4$ model is 80.01
- AICc of ARIMA$(2,1,0)(2,0,0)_4$ model is 79.61

European quarterly retail trade

Models with lowest AICc values tend to give slightly better results than the other models.

- AICc of ARIMA(1,1,0)(2,0,0)$_4$ model is 79.65
- AICc of ARIMA(1,1,0)(3,0,0)$_4$ model is 81.98
- AICc of ARIMA(1,1,0)(1,0,0)$_4$ model is 85.94
- AICc of ARIMA(1,1,0)(1,0,1)$_4$ model is 78

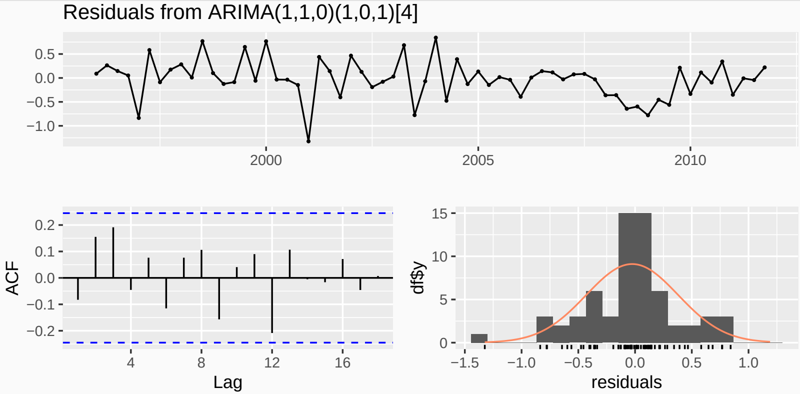
European quarterly retail trade

```r
# fit model
(fit2 <- Arima(euretail, order=c(1,1,0), seasonal=c(1,0,1)))
```

```
## Series: euretail
## ARIMA(1,1,0)(1,0,1)[4]
##
## Coefficients:
##          ar1    sar1     sma1
##        0.397  0.966  -0.737
## s.e.  0.127  0.044   0.161
##
## sigma^2 = 0.173:  log likelihood = -34.66
## AIC=77.31   AICc=78   BIC=85.88
```

European quarterly retail trade

```
# residuals
checkresiduals(fit2, lag=24, test=FALSE)
```



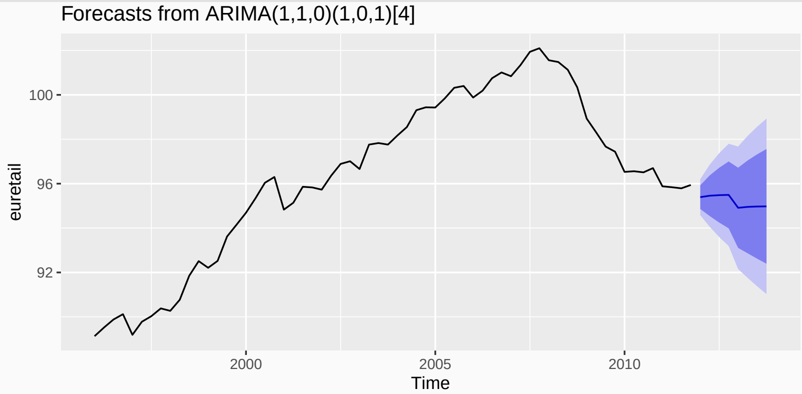
European quarterly retail trade

```r
# residuals
checkresiduals(fit2, lag=24, plot=FALSE)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,1,0)(1,0,1)[4]
## Q* = 16, df = 21, p-value = 0.8
##
## Model df: 3.   Total lags used: 24
```

European quarterly retail trade

```r
# residuals
autoplot(forecast(fit2))
```



European quarterly retail trade

Remarks:

- the textbook suggests the candidate model: ARIMA(0,1,3)$(0,1,1)_4$...
- AIC/AICc/BIC comparisons must have the same orders of differencing
- but **RMSE test set** comparisons can involve any models

European quarterly retail trade

```
# data and set up
n.end <- 2003.75 # 2003Q4
h.val <- 1
# loop
pred <- matrix(rep(NA,96),32,3)
for(i in 1:32){
  tmp0 <- 1996
  tmp1 <- n.end+(i-1)*.25
  tmp <- window(euretail,tmp0,tmp1)
  pred[i,1] <- window(euretail,tmp1+h.val*.25,tmp1+h.val*.25) # actual
  # estimate models
  fit1 <- Arima(tmp, order=c(1,1,0), seasonal=c(1,0,1))
  fit2 <- Arima(tmp, order=c(0,1,3), seasonal=c(0,1,1))
  # compute forecasts (h=1)
  pred[i,2] <- forecast(fit1, h=h.val)$mean[h.val]
  pred[i,3] <- forecast(fit2, h=h.val)$mean[h.val]
}
```

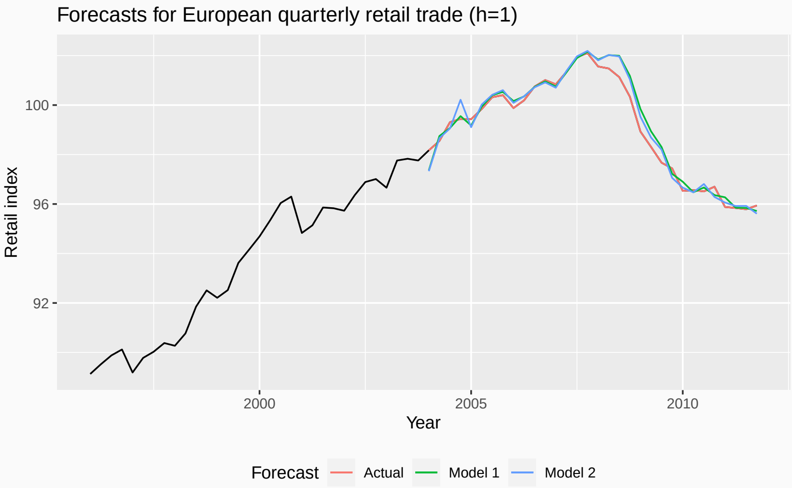
European quarterly retail trade

```
# compute rmse (h=1)
retailfit1 <- pred[,1] - pred[,2]
retailfit2 <- pred[,1] - pred[,3]
rmse1 <- sqrt(mean(retailfit1^2, na.rm=TRUE))
rmse2 <- sqrt(mean(retailfit2^2, na.rm=TRUE))
# display rmse
cbind(rmse1,rmse2)
```

```
##        rmse1  rmse2
## [1,] 0.3944 0.3795
```

- choose forecasting model with the smallest RMSE computed using time series cross-validation

European quarterly retail trade



European quarterly retail trade

```r
# data and set up
n.end <- 2003.75 # 2003Q4
h.val <- 4
# loop
pred <- matrix(rep(NA,87),29,3)
for(i in 1:29){
  tmp0 <- 1996
  tmp1 <- n.end+(i-1)*.25
  tmp <- window(euretail,tmp0,tmp1)
  pred[i,1] <- window(euretail,tmp1+h.val*.25,tmp1+h.val*.25) # actual
  # estimate models
  fit1 <- Arima(tmp, order=c(1,1,0), seasonal=c(1,0,1))
  fit2 <- Arima(tmp, order=c(0,1,3), seasonal=c(0,1,1))
  # compute forecasts (h=4)
  pred[i,2] <- forecast(fit1, h=h.val)$mean[h.val]
  pred[i,3] <- forecast(fit2, h=h.val)$mean[h.val]
}
```

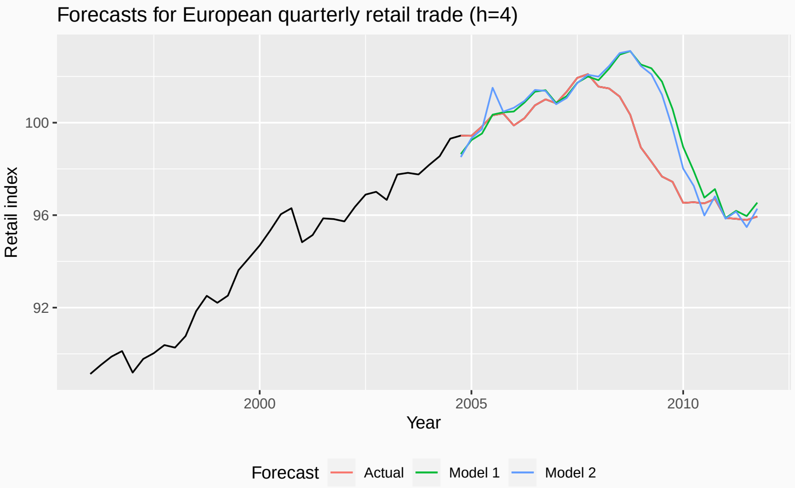
European quarterly retail trade

```r
# compute rmse (h=4)
retailfit1 <- pred[,1] - pred[,2]
retailfit2 <- pred[,1] - pred[,3]
rmse1 <- sqrt(mean(retailfit1^2, na.rm=TRUE))
rmse2 <- sqrt(mean(retailfit2^2, na.rm=TRUE))
# display rmse
cbind(rmse1,rmse2)
```

```
##      rmse1 rmse2
## [1,] 1.643 1.489
```

- choose forecasting model with the smallest RMSE computed using time series cross-validation

European quarterly retail trade



Outline

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 stepwise search to traverse model space

How does auto.arima() work?

Step1: Select 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 by ± 1
- p, q both vary by ± 1
- include/exclude c

Model with lowest AICc becomes current model.

Repeat Step 2 until no lower AICc can be found.

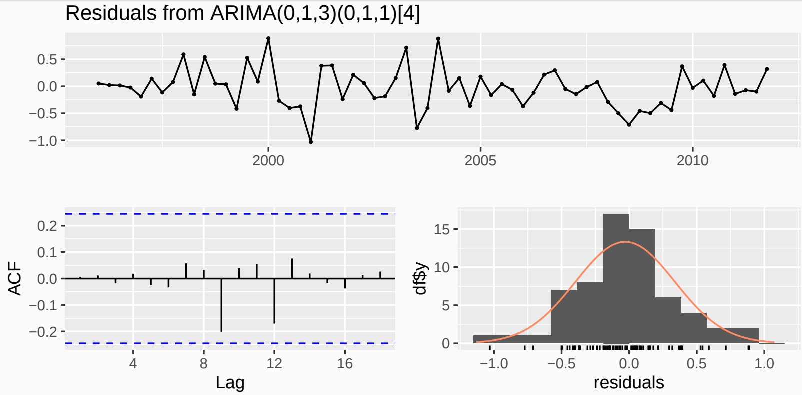
Choosing your own model

```r
# fit model
(fit <- auto.arima(euretail))
```

```
## Series: euretail
## ARIMA(0,1,3)(0,1,1)[4]
##
## Coefficients:
##          ma1    ma2    ma3     sma1
##        0.263  0.369  0.420  -0.664
## s.e.   0.124  0.126  0.129   0.155
##
## sigma^2 = 0.156:  log likelihood = -28.63
## AIC=67.26   AICc=68.39   BIC=77.65
```

Choosing your own model

```
# check residuals
checkresiduals(fit, test=FALSE)
```



Choosing your own model

```
# check residuals
checkresiduals(fit, plot=FALSE)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,3)(0,1,1)[4]
## Q* = 0.51, df = 4, p-value = 1
##
## Model df: 4.   Total lags used: 8
```

Modelling procedure with `auto.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 `auto.arima` to select a model
4. Check the residuals from your chosen model by plotting the ACF of the residuals, and doing a portmanteau test of the residuals
5. If they do not look like white noise, try a modified model
6. Once the residuals look like white noise, calculate forecasts