MONASH University

MONASH BUSINESS SCHOOL

# Tidy Time Series & Forecasting in R

tsibble feasts fable

## 6. Automatic forecasting algorithms

robjhyndman.com/workshop2020

# Outline

# Outline

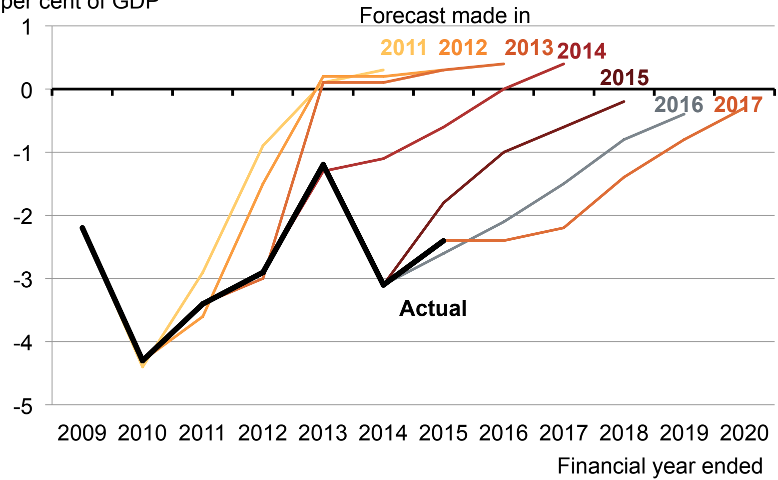
# Forecasting is difficult

**Commonwealth plans to drift back to surplus show the triumph of experience over hope**

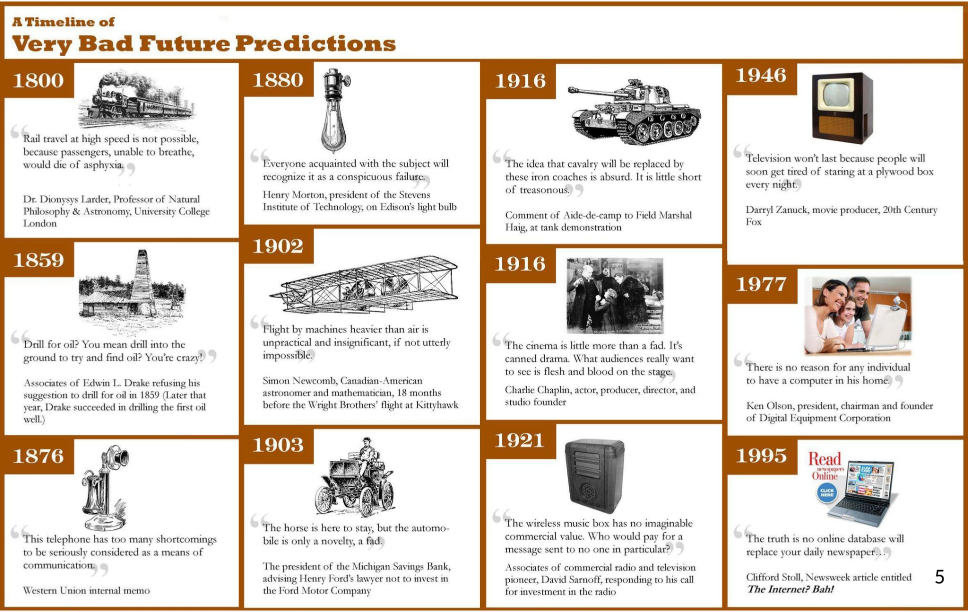
GRATTAN Institute

**Actual and forecast Commonwealth underlying cash balance**

per cent of GDP



# Forecasting is difficult



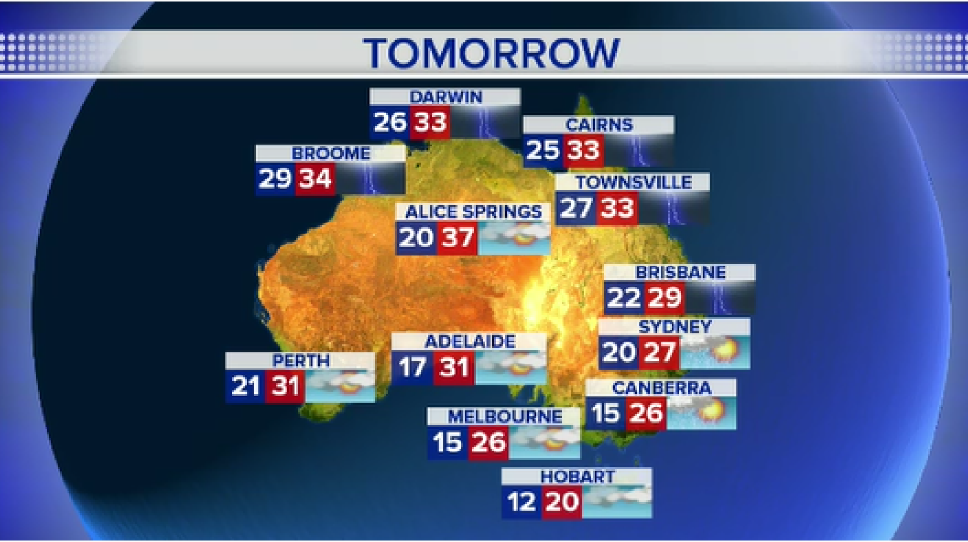
# What can we forecast?

# What can we forecast?

# What can we forecast?

# What can we forecast?

# What can we forecast?



# What can we forecast?

# What can we forecast?

# Which is easiest to forecast?

1. daily electricity demand in 3 days time
2. timing of next Halley’s comet appearance
3. time of sunrise this day next year
4. Google stock price tomorrow
5. Google stock price in 6 months time
6. maximum temperature tomorrow
7. exchange rate of $US/AUS next week
8. total sales of drugs in Australian pharmacies next month

# Which is easiest to forecast?

1. daily electricity demand in 3 days time
2. timing of next Halley’s comet appearance
3. time of sunrise this day next year
4. Google stock price tomorrow
5. Google stock price in 6 months time
6. maximum temperature tomorrow
7. exchange rate of $US/AUS next week
8. total sales of drugs in Australian pharmacies next month

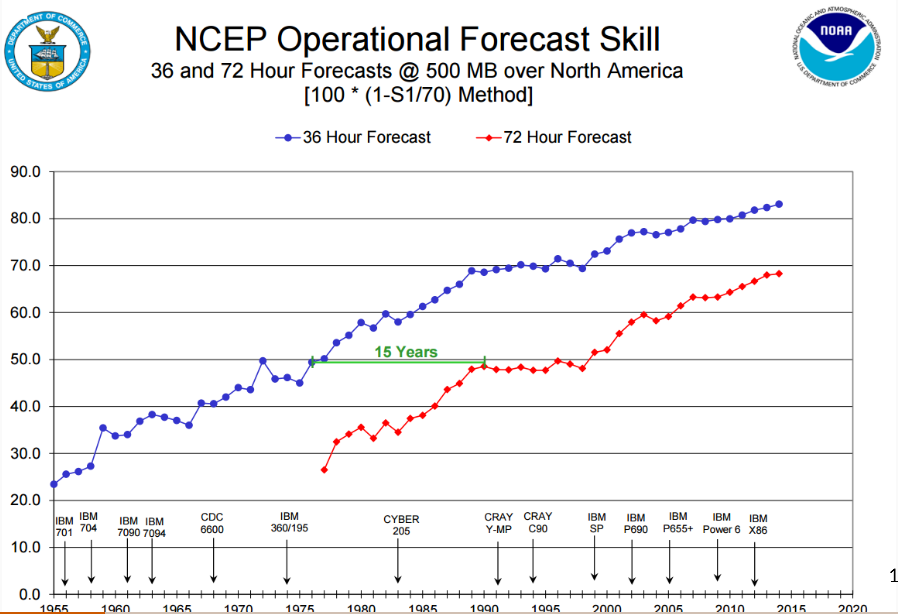
- how do we measure “easiest”?
- what makes something easy/difficult to forecast?

# Factors affecting forecastability

Something is easier to forecast if:

- we have a good understanding of the factors that contribute to it
- there is lots of data available;
- the forecasts cannot affect the thing we are trying to forecast.
- there is relatively low natural/unexplainable random variation.
- the future is somewhat similar to the past

# Improving forecasts



# Outline

# CASE STUDY 1: Paperware company

**Problem:** Want forecasts of each of hundreds of items. Series can be stationary, trended or seasonal. They currently have a large forecasting program written in-house but it doesn't seem to produce sensible forecasts. They want me to tell them what is wrong and fix it.

**Additional information**

- Program written in COBOL making numerical calculations limited. It is not possible to do any optimisation.
- Their programmer has little experience in numerical computing.
- They employ no statisticians and want

# CASE STUDY 1: Paperware company

## Methods currently used

- **A** 12 month average
- **C** 6 month average
- **E** straight line regression over last 12 months
- **G** straight line regression over last 6 months
- **H** average slope between last year’s and this year’s values. (Equivalent to differencing at lag 12 and taking mean.)
- **I** Same as H except over 6 months.
- **K** I couldn’t understand the explanation.

# CASE STUDY 2: PBS

# CASE STUDY 2: PBS

**The Pharmaceutical Benefits Scheme (PBS) is the Australian government drugs subsidy scheme.**

- Many drugs bought from pharmacies are subsidised to allow more equitable access to modern drugs.
- The cost to government is determined by the number and types of drugs purchased. Currently nearly 1% of GDP.
- The total cost is budgeted based on forecasts of drug usage.

# CASE STUDY 2: PBS

ABC News Online
AUSTRALIAN BROADCASTING CORPORATION

NewsRadio
Streaming audio news
LISTEN: WMP | Real

Select a Topic from the list below

Click "Refresh" or "Reload" on your browser for the latest edition.

This Bulletin: **Wed, May 30 2001 6:22 PM AEST**

POLITICS

- Top Stories
- Just In
- World
- Asia-Pacific
- Business
- Sport
- Arts
- Sci Tech
- Indigenous
- Weather
- Rural
- Local News
- Broadband

Search

SPECIALS

- Federal Election

## Opp demands drug price restriction after PBS budget blow-out

The Federal Opposition has called for tighter controls on drug prices after the Pharmaceutical Benefits Scheme (PBS) budget blew out by almost $800 million.

The money was spent on two new drugs including the controversial anti-smoking aid Zyban, which dropped in price from $220 to $22 after it was listed on the PBS.

the Public Record
For full election coverage

FEATURES

Public Record
Federal Election 2001

For a fresh perspective on the federal election, reach into ABC Online's campaign weblog, The Poll Vault.

Audio News Online

# CASE STUDY 2: PBS

- In 2001: \$4.5 billion budget, under-forecasted by \$800 million.
- Thousands of products. Seasonal demand.
- Subject to covert marketing, volatile products, uncontrollable expenditure.
- Although monthly data available for 10 years, data are aggregated to annual values, and only the first three years are used in estimating the forecasts.
- All forecasts being done with the `FORECAST` function in MS-Excel!

# CASE STUDY 3: Car fleet company

**Client:** One of Australia's largest car fleet companies

**Problem:** how to forecast resale value of vehicles? How should this affect leasing and sales policies?

# CASE STUDY 3: Car fleet company

**Client:** One of Australia's largest car fleet companies

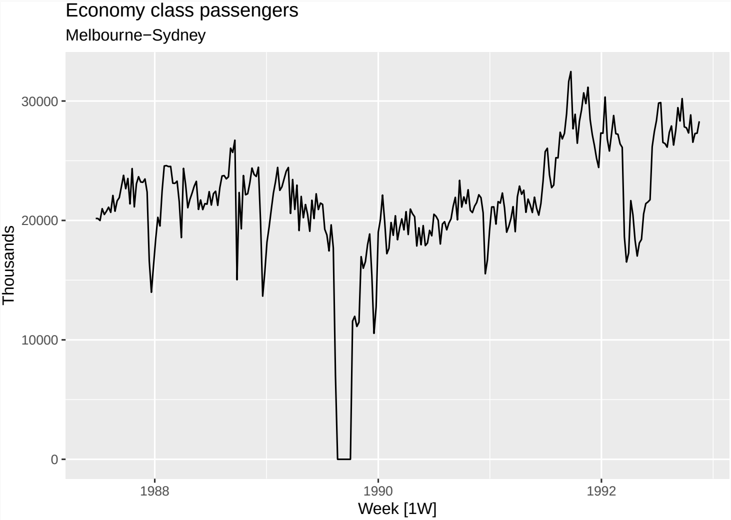
**Problem:** how to forecast resale value of vehicles? How should this affect leasing and sales policies?

**Additional information**

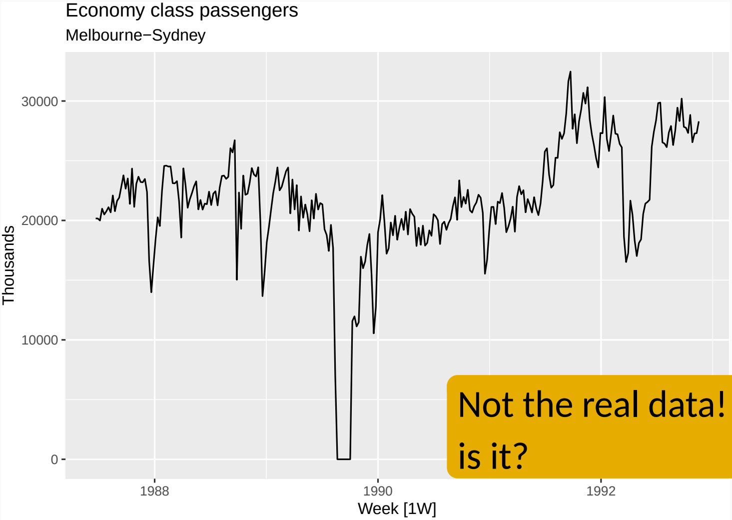
- They can provide a large amount of data on previous vehicles and their eventual resale values.
- The resale values are currently estimated by a group of specialists. They see me as a threat and do not cooperate.

# CASE STUDY 4: Airline

# CASE STUDY 4: Airline



# CASE STUDY 4: Airline




Not the real data! Or is it?

# CASE STUDY 4: Airline

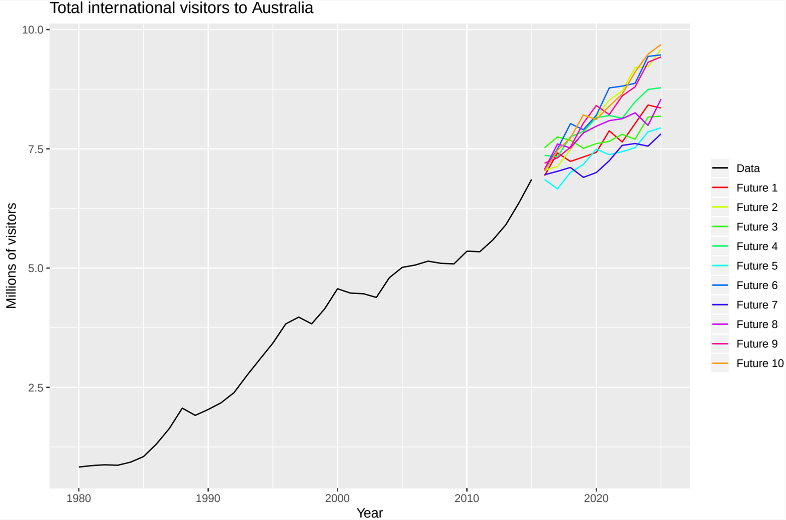
**Problem:** how to forecast passenger traffic on major routes?

### Additional information

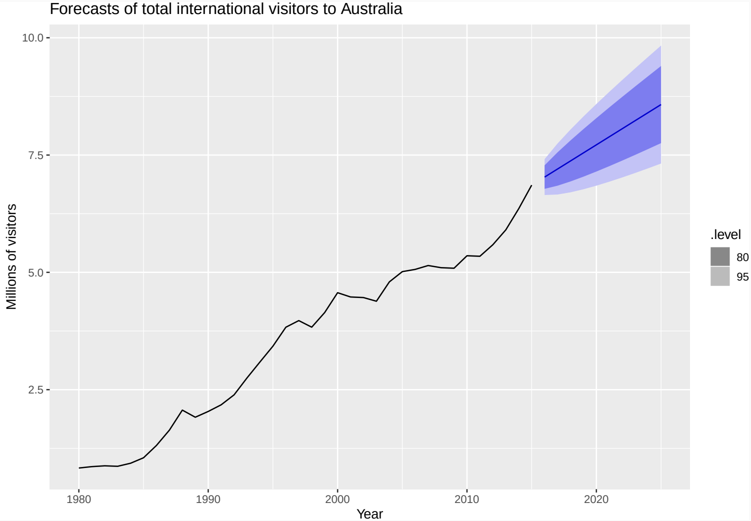
- They can provide a large amount of data on previous routes.
- Traffic is affected by school holidays, special events such as the Grand Prix, advertising campaigns, competition behaviour, etc.
- They have a highly capable team of people who are able to do most of the computing.

# Outline

# Sample futures



# Forecast intervals



# Statistical forecasting

- Thing to be forecast: a random variable, $y_t$.
- Forecast distribution: If $\mathcal{I}$ is all observations, then $y_t|\mathcal{I}$ means "the random variable $y_t$ given what we know in $\mathcal{I}$".
- The "point forecast" is the mean (or median) of $y_t|\mathcal{I}$
- The "forecast variance" is $\text{var}[y_t|\mathcal{I}]$
- A prediction interval or "interval forecast" is a range of values of $y_t$ with high probability.
- With time series, $y_{t|t-1} = y_t|\{y_1, y_2, \dots, y_{t-1}\}$.
- $\hat{y}_{T+h|T} = \text{E}[y_{T+h}|y_1, \dots, y_T]$ (an $h$-step forecast taking account of all observations up to time $T$).

# Outline

# 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”: $\alpha$, $\beta$ and $\gamma$ respectively.
- Need to choose best values for the smoothing parameters (and initial states).
- Equivalent ETS state space models developed in the 1990s and 2000s.

# A model for levels, trends, and seasonalities

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

**How do we combine these elements?**

# A model for levels, trends, and seasonalities

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

**How do we combine these elements?**

**Additively?**

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

# A model for levels, trends, and seasonalities

We want a model that captures the level ($\ell_t$), 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?

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

# A model for levels, trends, and seasonalities

We want a model that captures the level ($\ell_t$), 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?**

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

# A model for levels, trends, and seasonalities

We want a model that captures the level ($\ell_t$), 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?**

# ETS models

General notation

$$\begin{array}{ccccc} & E & T & S & : \textbf{E}\text{xponen}\textbf{T}\text{ial }\textbf{S}\text{moothing} \\ & \nearrow & \uparrow & \nwarrow & \\ \textbf{E}\text{rror} & & \textbf{T}\text{rend} & & \textbf{S}\text{eason} \end{array}$$

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

# ETS models

General notation $\quad$ E T S : **E**xponen**T**ial **S**moothing

$$\nearrow \quad \uparrow \quad \nwarrow$$

**E**rror **T**rend **S**eason

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

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

# ETS models

General notation $\quad$ E T S : **E**xponen**T**ial **S**moothing

**E**rror **T**rend **S**eason

**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"`)

# ETS(A,N,N): SES with additive errors

$$
\begin{aligned}
\text{Forecast equation} \qquad \hat{y}_{T+h|T} &= \ell_T \\
\text{Measurement equation} \qquad y_t &= \ell_{t-1} + \varepsilon_t \\
\text{State equation} \qquad \ell_t &= \ell_{t-1} + \alpha\varepsilon_t
\end{aligned}
$$

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

# ETS(A,N,N): SES with additive errors

| | |
|---|---|
| Forecast equation | $\hat{y}_{T+h\|T} = \ell_T$ |
| 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, $\varepsilon_t$.
- Measurement equation: relationship between observations and states.
- Transition/state equation(s): evolution of the state(s) through time.

# ETS(M,N,N): SES with multiplicative errors

$$\begin{aligned} \text{Forecast equation} \quad \hat{y}_{T+h|T} &= \ell_T \\ \text{Measurement equation} \quad y_t &= \ell_{t-1}(1 + \varepsilon_t) \\ \text{State equation} \quad \ell_t &= \ell_{t-1}(1 + \alpha\varepsilon_t) \end{aligned}$$

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

# ETS(M,N,N): SES with multiplicative errors

$$\begin{aligned}
\text{Forecast equation} \quad \hat{y}_{T+h|T} &= \ell_T \\
\text{Measurement equation} \quad y_t &= \ell_{t-1}(1 + \varepsilon_t) \\
\text{State equation} \quad \ell_t &= \ell_{t-1}(1 + \alpha\varepsilon_t)
\end{aligned}$$

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

- Models with additive and multiplicative errors with the same parameters generate the same point forecasts but different prediction intervals.

# ETS(A,A,N): Holt's linear trend

**Additive errors**

| | |
|---|---|
| Forecast equation | $\hat{y}_{T+h\|T} = \ell_T + hb_T$ |
| Measurement equation | $y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$ |
| State equations | $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ |
| | $b_t = b_{t-1} + \beta\varepsilon_t$ |

# ETS(A,A,N): Holt's linear trend

## Additive errors

$$
\begin{aligned}
\text{Forecast equation} \quad & \hat{y}_{T+h|T} = \ell_T + hb_T \\
\text{Measurement equation} \quad & y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t \\
\text{State equations} \quad & \ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t \\
& b_t = b_{t-1} + \beta\varepsilon_t
\end{aligned}
$$

## Multiplicative errors

$$
\begin{aligned}
\text{Forecast equation} \quad & \hat{y}_{T+h|T} = \ell_T + hb_T \\
\text{Measurement equation} \quad & y_t = (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t) \\
\text{State equations} \quad & \ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t) \\
& b_t = b_{t-1} + \beta\varepsilon_t
\end{aligned}
$$

# Example: Australian population

```
aus_economy <- global_economy %>% filter(Code == "AUS") %>%
  mutate(Pop = Population/1e6)
fit <- aus_economy %>% model(AAN = ETS(Pop))
report(fit)
```

```
## Series: Pop
## Model: ETS(A,A,N)
##   Smoothing parameters:
##     alpha = 1
##     beta  = 0.327
##
##   Initial states:
##     l     b
##  10.1 0.222
##
##   sigma^2:  0.0041
##
##   AIC  AICc   BIC
## -77.0 -75.8 -66.7
```

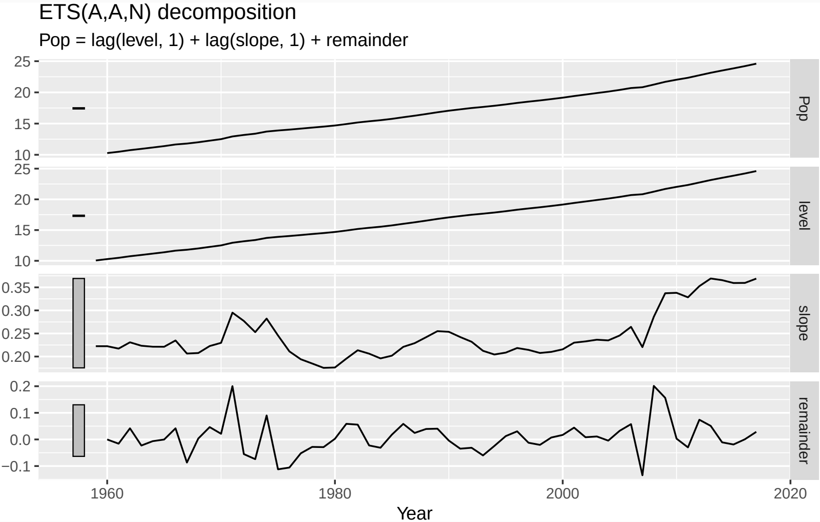
# Example: Australian population

```
components(fit)
```

```
## # A dable:                     59 x 7 [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
##     <fct>      <chr>  <dbl> <dbl> <dbl> <dbl>     <dbl>
##  1 Australia AAN     1959  NA    10.1 0.222 NA
##  2 Australia AAN     1960  10.3  10.3 0.222 -0.000145
##  3 Australia AAN     1961  10.5  10.5 0.217 -0.0159
##  4 Australia AAN     1962  10.7  10.7 0.231  0.0418
##  5 Australia AAN     1963  11.0  11.0 0.223 -0.0229
##  6 Australia AAN     1964  11.2  11.2 0.221 -0.00641
##  7 Australia AAN     1965  11.4  11.4 0.221 -0.000314
##  8 Australia AAN     1966  11.7  11.7 0.235  0.0418
##  9 Australia AAN     1967  11.8  11.8 0.206 -0.0869
## 10 Australia AAN     1968  12.0  12.0 0.208  0.00350
```

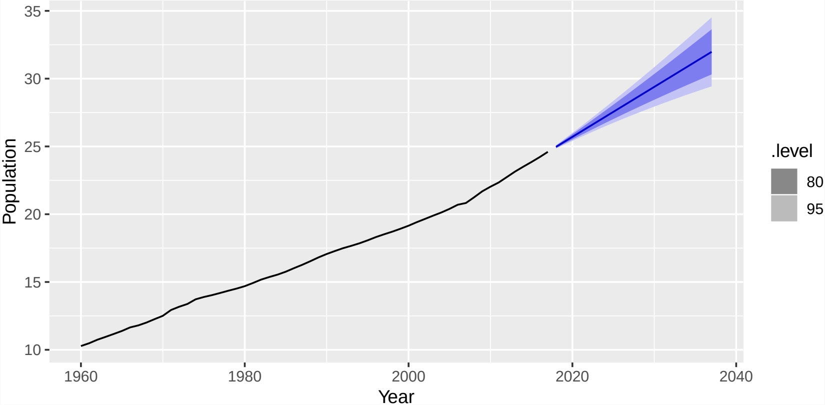
# Example: Australian population

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



# Example: Australian population

```
fit %>%
  forecast(h = 20) %>%
  autoplot(aus_economy) +
  ylab("Population") + xlab("Year")
```



# ETS(A,Ad,N): Damped trend method

## Additive errors

$$
\begin{aligned}
\text{Forecast equation} \quad & \hat{y}_{T+h|T} = \ell_T + (\phi + \cdots + \phi^{h-1})b_T \\
\text{Measurement equation} \quad & y_t = (\ell_{t-1} + \phi b_{t-1}) + \varepsilon_t \\
\text{State equations} \quad & \ell_t = (\ell_{t-1} + \phi b_{t-1}) + \alpha\varepsilon_t \\
& b_t = \phi b_{t-1} + \beta\varepsilon_t
\end{aligned}
$$

# ETS(A,Ad,N): Damped trend method

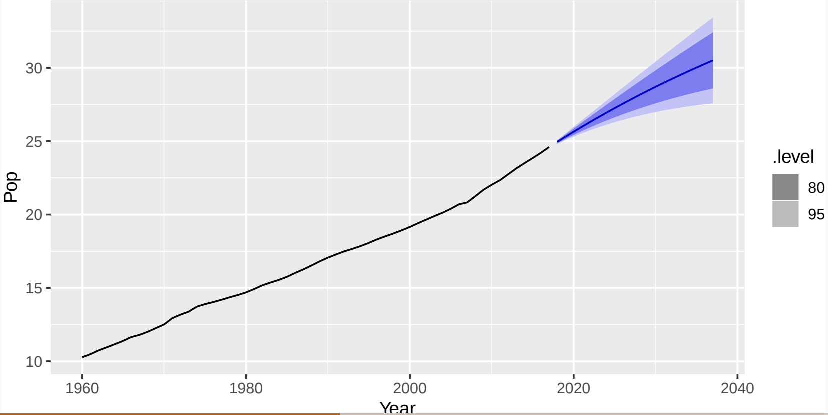
## Additive errors

$$
\begin{aligned}
\text{Forecast equation} \quad & \hat{y}_{T+h|T} = \ell_T + (\phi + \cdots + \phi^{h-1})b_T \\
\text{Measurement equation} \quad & y_t = (\ell_{t-1} + \phi b_{t-1}) + \varepsilon_t \\
\text{State equations} \quad & \ell_t = (\ell_{t-1} + \phi b_{t-1}) + \alpha\varepsilon_t \\
& b_t = \phi b_{t-1} + \beta\varepsilon_t
\end{aligned}
$$

- 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.

# Example: Australian population

```
aus_economy %>%
  model(holt = ETS(Pop ~ trend("Ad"))) %>%
  forecast(h = 20) %>%
  autoplot(aus_economy)
```



# Example: National populations

```r
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
```

# Example: National populations

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

```
## # A fable: 1,315 x 5 [1Y]
## # Key:     Country, .model [263]
##    Country     .model  Year   Pop .distribution
##    <fct>       <chr>  <dbl> <dbl> <dist>
##  1 Afghanistan ets     2018 36.4  N(36, 0.012)
##  2 Afghanistan ets     2019 37.3  N(37, 0.059)
##  3 Afghanistan ets     2020 38.2  N(38, 0.164)
##  4 Afghanistan ets     2021 39.0  N(39, 0.351)
##  5 Afghanistan ets     2022 39.9  N(40, 0.644)
##  6 Albania     ets     2018  2.87 N(2.9, 0.00012)
##  7 Albania     ets     2019  2.87 N(2.9, 0.00060)
##  8 Albania     ets     2020  2.87 N(2.9, 0.00169)
##  9 Albania     ets     2021  2.86 N(2.9, 0.00362)
```

# ETS(A,A,A): Holt-Winters additive method

$$
\begin{aligned}
\text{Forecast equation} \quad \hat{y}_{t+h|t} &= \ell_t + hb_t + s_{t+h-m(k+1)} \\
\text{Observation equation} \quad y_t &= \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t \\
\text{State equations} \quad \ell_t &= \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t \\
b_t &= b_{t-1} + \beta\varepsilon_t \\
s_t &= s_{t-m} + \gamma\varepsilon_t
\end{aligned}
$$

- $k$ = integer part of $(h-1)/m$.
- $\sum_i s_i \approx 0$.
- 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).

# ETS(M,A,M): Holt-Winters multiplicative method

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

- $k$ is integer part of $(h-1)/m$.
- $\sum_i s_i \approx m$.
- 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).

# 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 Australia    Holiday <ETS(A,N,A~
##  2 Adelaide Hills             South Australia    Holiday <ETS(A,A,N~
##  3 Alice Springs              Northern Territo~ Holiday <ETS(M,N,A~
##  4 Australia's Coral Coast    Western Australia Holiday <ETS(M,N,A~
##  5 Australia's Golden Outba~ Western Australia Holiday <ETS(M,N,M~
##  6 Australia's North West     Western Australia Holiday <ETS(A,N,A~
##  7 Australia's South West     Western Australia Holiday <ETS(M,N,M~
##  8 Ballarat                   Victoria           Holiday <ETS(M,N,A~
##  9 Barkly                     Northern Territo~ Holiday <ETS(A,N,A~
## 10 Barossa                    South Australia    Holiday <ETS(A,N,N~
```

# Example: Australian holiday tourism

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

```
## Series: Trips
## Model: ETS(M,N,A)
##   Smoothing parameters:
##     alpha = 0.157
##     gamma = 1e-04
##
##   Initial states:
##     l   s1   s2     s3     s4
##   142 -61 131 -42.2 -27.7
##
##   sigma^2:  0.0388
##
##   AIC  AICc   BIC
##   852   854   869
```

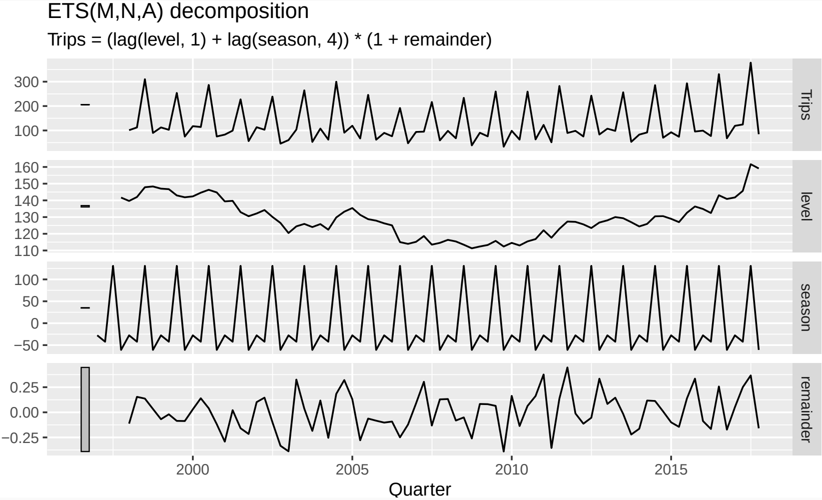
# Example: Australian holiday tourism

```
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>
```

# Example: Australian holiday tourism

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



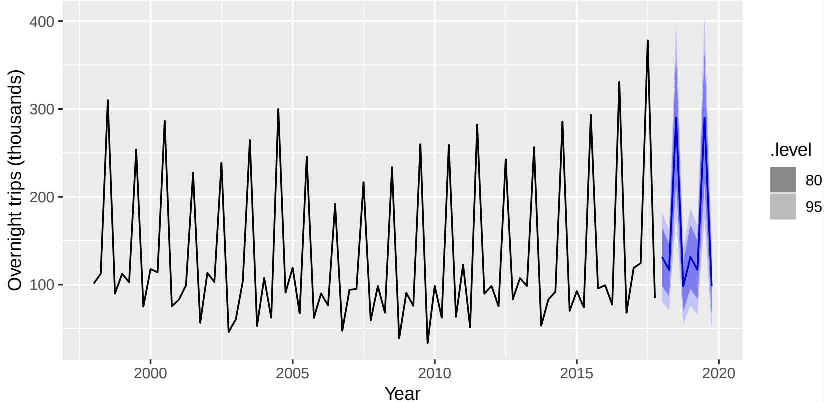
# Example: Australian holiday tourism

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

```
## # A fable: 608 x 7 [1Q]
## # Key:     Region, State, Purpose, .model [76]
##    Region    State     Purpose .model   Quarter Trips .distribution
##    <chr>     <chr>     <chr>   <chr>      <qtr> <dbl> <dist>
##  1 Adelaide  South A~ Holiday ets      2018 Q1 210.  N(210, 457)
##  2 Adelaide  South A~ Holiday ets      2018 Q2 173.  N(173, 473)
##  3 Adelaide  South A~ Holiday ets      2018 Q3 169.  N(169, 489)
##  4 Adelaide  South A~ Holiday ets      2018 Q4 186.  N(186, 505)
##  5 Adelaide  South A~ Holiday ets      2019 Q1 210.  N(210, 521)
##  6 Adelaide  South A~ Holiday ets      2019 Q2 173.  N(173, 537)
##  7 Adelaide  South A~ Holiday ets      2019 Q3 169.  N(169, 553)
##  8 Adelaide  South A~ Holiday ets      2019 Q4 186.  N(186, 569)
##  9 Adelaide~ South A~ Holiday ets      2018 Q1  19.4 N(19, 36)
## 10 Adelaide~ South A~ Holiday ets      2018 Q2  19.6 N(20, 36)
## # ... with 598 more rows
```

# Example: Australian holiday tourism

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



# Exponential smoothing 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 |

# Estimating ETS models

- Smoothing parameters $\alpha$, $\beta$, $\gamma$ and $\phi$, and the initial states $\ell_0, b_0, s_0, s_{-1}, \dots, s_{-m+1}$ 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.

# 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.

# 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).

# 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

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

# AIC and cross-validation

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

# Automatic forecasting

**From Hyndman et al. (IJF, 2002):**

1. Apply each model that is appropriate to the data. Optimize parameters and initial values using MLE.
2. Select best method using AICc.
3. Produce forecasts using best method.
4. Obtain forecast intervals using underlying state space model.

- Method performed very well in M3 competition.
- Used as a benchmark in the M4 competition.

# Outline

# Lab Session 7

Find an ETS model for the Gas data from `aus_production`.

- Why is multiplicative seasonality necessary here?
- Experiment with making the trend damped.

# Outline

# ARIMA models

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

# 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.

# Stationarity

**Definition**

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

# Stationarity

**Definition**

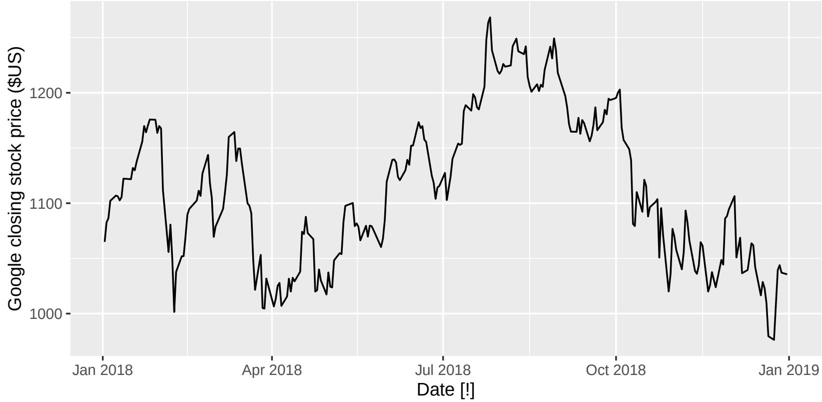
If $\{y_t\}$ is a stationary time series, then for all $s$, the distribution of $(y_t, \dots, y_{t+s})$ does not depend on $t$.

A **stationary series** is:

- roughly horizontal
- constant variance
- no patterns predictable in the long-term

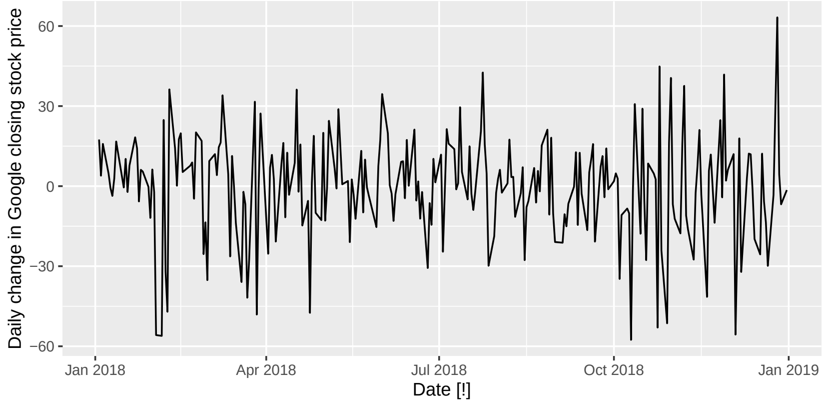
# Stationary?

```r
gafa_stock %>%
  filter(Symbol == "GOOG", year(Date) == 2018) %>%
  autoplot(Close) +
    ylab("Google closing stock price ($US)")
```



# Stationary?

```
gafa_stock %>%
  filter(Symbol == "GOOG", year(Date) == 2018) %>%
  autoplot(difference(Close)) +
    ylab("Daily change in Google closing stock price")
```



# Differencing

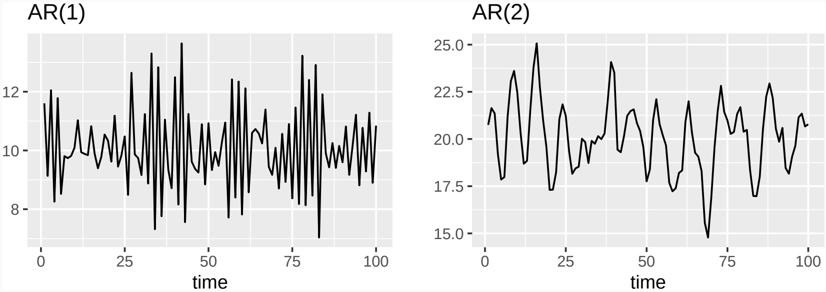
- Differencing helps to **stabilize the mean**.
- The differenced series is the *change* between each observation in the original series.
- Occasionally the differenced data will not appear stationary and it may be necessary to difference the data a second time.
- In practice, it is almost never necessary to go beyond second-order differences.

# 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 $\varepsilon_t$ is white noise. This is a multiple regression with **lagged values** of $y_t$ as predictors.

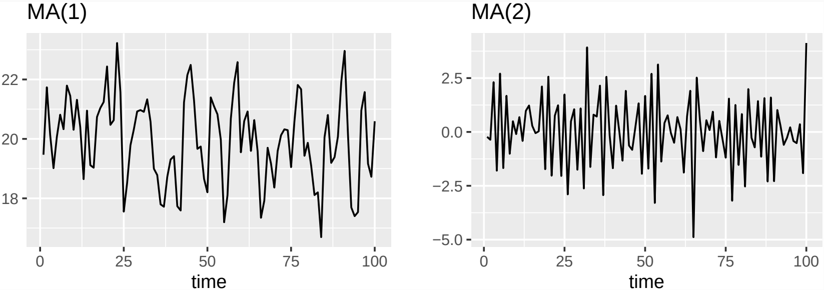



- Cyclic behaviour is possible when $p \geq 2$.

## 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 $\varepsilon_t$ is white noise. This is a multiple regression with **lagged *errors*** as predictors. *Don't confuse this with moving average smoothing!*



# ARIMA models

**Autoregressive Moving Average models:**

$$y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} \\ + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t.$$

# ARIMA models

**Autoregressive Moving Average models:**

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

- Predictors include both **lagged values of $y_t$ and lagged errors.**

# ARIMA models

**Autoregressive Moving Average models:**

$$y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t.$$

- Predictors include both **lagged values of $y_t$ and lagged errors.**

**Autoregressive Integrated Moving Average models**

- Combine ARMA model with **differencing**.
- $d$-differenced series follows an ARMA model.
- Need to choose $p$, $d$, $q$ and whether or not to include $c$.

# ARIMA 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,$q$)

# Example: National populations

```r
fit <- global_economy %>%
  model(arima = ARIMA(Population))
fit
```

```
## # A mable: 263 x 2
## # Key:     Country [263]
##    Country             arima
##    <fct>               <model>
##  1 Afghanistan         <ARIMA(4,2,1)>
##  2 Albania             <ARIMA(0,2,2)>
##  3 Algeria             <ARIMA(2,2,2)>
##  4 American Samoa      <ARIMA(2,2,2)>
##  5 Andorra             <ARIMA(2,1,2) w/ drift>
##  6 Angola              <ARIMA(4,2,1)>
##  7 Antigua and Barbuda <ARIMA(2,1,2) w/ drift>
##  8 Arab World          <ARIMA(0,2,1)>
```

# Example: National populations

```
fit %>% filter(Country=="Australia") %>% report()
```

```
## Series: Population
## Model: ARIMA(0,2,1)
##
## Coefficients:
##           ma1
##        -0.661
## s.e.    0.107
##
## sigma^2 estimated as 4.063e+09:  log likelihood=-699
## AIC=1401   AICc=1402   BIC=1405
```

# Example: National populations

```r
fit %>% filter(Country=="Australia") %>% report()
```

```
## Series: Population
## Model: ARIMA(0,2,1)
##
## Coefficients:
##           ma1
##        -0.661
## s.e.    0.107
##
## sigma^2 estimated as 4.063e+09:  log likelihood=-699
## AIC=1401   AICc=1402   BIC=1405
```

$$y_t = 2y_{t-1} - y_{t-2} - 0.7\varepsilon_{t-1} + \varepsilon_t$$
$$\varepsilon_t \sim \text{NID}(0, 4 \times 10^9)$$

# Understanding ARIMA models

- If $c = 0$ and $d = 0$, the long-term forecasts will go to zero.
- If $c = 0$ and $d = 1$, the long-term forecasts will go to a non-zero constant.
- If $c = 0$ and $d = 2$, the long-term forecasts will follow a straight line.
- If $c \neq 0$ and $d = 0$, the long-term forecasts will go to the mean of the data.
- If $c \neq 0$ and $d = 1$, the long-term forecasts will follow a straight line.
- If $c \neq 0$ and $d = 2$, the long-term forecasts will follow a quadratic trend.

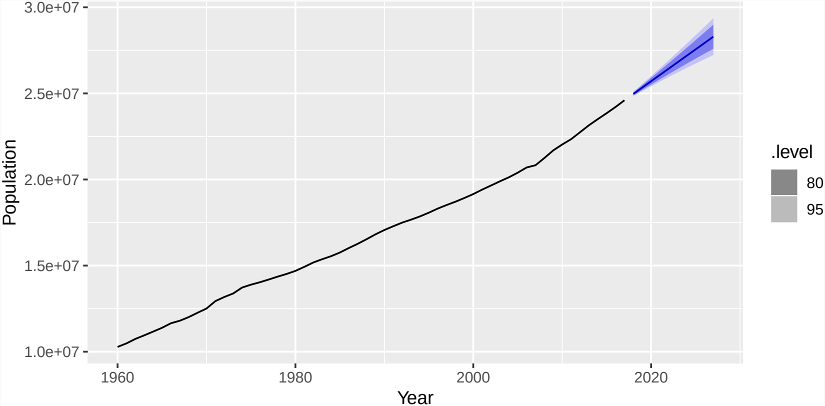
# Understanding ARIMA models

**Forecast variance and $d$**

- The higher the value of $d$, the more rapidly the prediction intervals increase in size.
- For $d = 0$, the long-term forecast standard deviation will go to the standard deviation of the historical data.

# Example: National populations

```
fit %>% forecast(h=10) %>%
  filter(Country=="Australia") %>%
  autoplot(global_economy)
```



# How does ARIMA() work?

**Hyndman and Khandakar (JSS, 2008) algorithm:**

- Select no. differences $d$ via KPSS test.
- Select $p$, $q$ and inclusion of $c$ by minimising AICc.
- Use stepwise search to traverse model space.

# How does ARIMA() work?

**Hyndman and Khandakar (JSS, 2008) algorithm:**

- Select no. differences $d$ via KPSS test.
- Select $p$, $q$ and inclusion of $c$ by minimising AICc.
- Use stepwise 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.

# How does ARIMA() work?

**Hyndman and Khandakar (JSS, 2008) algorithm:**

- Select no. differences $d$ via KPSS test.
- Select $p$, $q$ and inclusion of $c$ by minimising AICc.
- Use stepwise 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.

Note: Can't compare AICc for different values of $d$.

# How does ARIMA() work?

**Step1:** Select current model (with smallest AICc) from:

ARIMA(2, $d$, 2)

ARIMA(0, $d$, 0)

ARIMA(1, $d$, 0)

ARIMA(0, $d$, 1)

# How does ARIMA() work?

**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 $\pm 1$;
- $p, q$ both vary from current model by $\pm 1$;
- Include/exclude $c$ from current model.

Model with lowest AICc becomes current model.

# How does ARIMA() work?

**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 $\pm 1$;
- $p, q$ both vary from current model by $\pm 1$;
- Include/exclude $c$ from current model.

Model with lowest AICc becomes current model.

**Repeat Step 2 until no lower AICc can be found.**

# Outline

# Lab Session 8

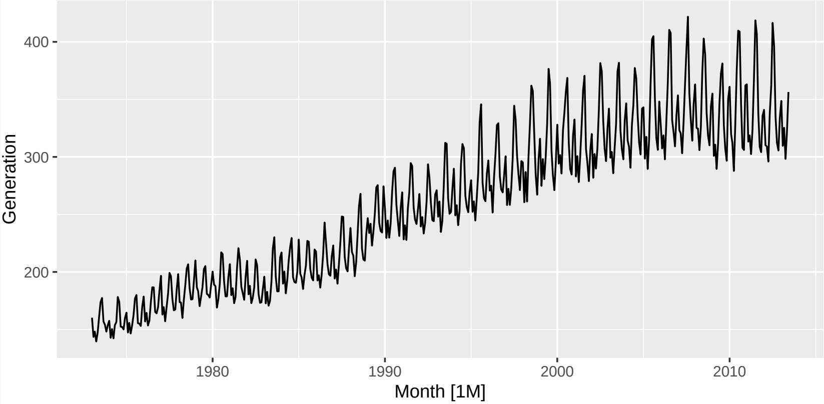
For the United States GDP data (from `global_economy`):

- Fit a suitable ARIMA model for the logged data.
- Produce forecasts of your fitted model. Do the forecasts look reasonable?

# Outline

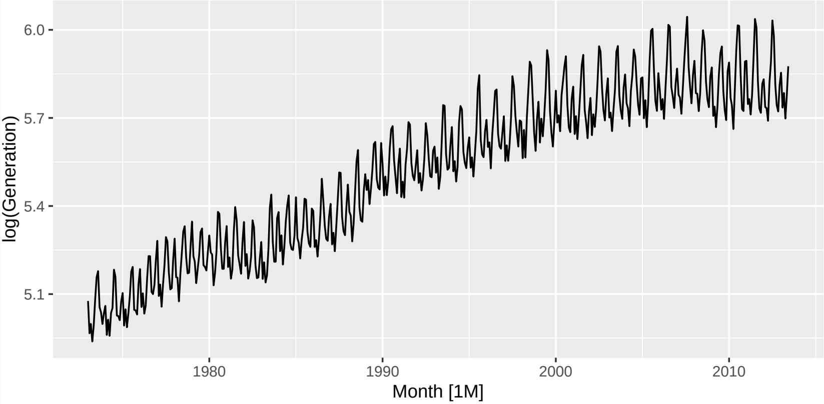
# Electricity production

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



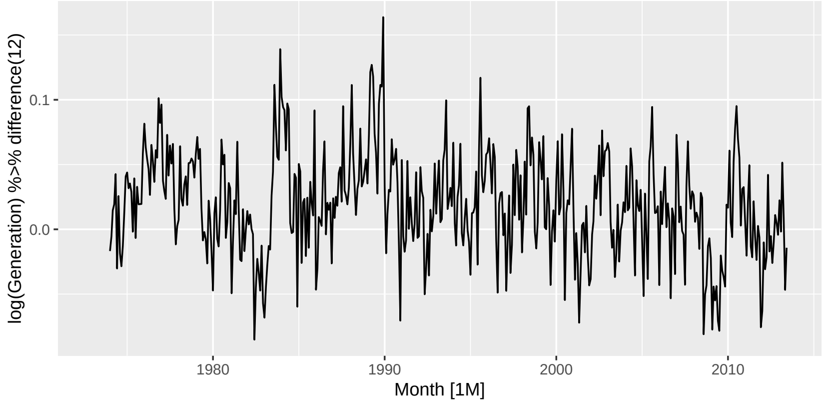
# Electricity production

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



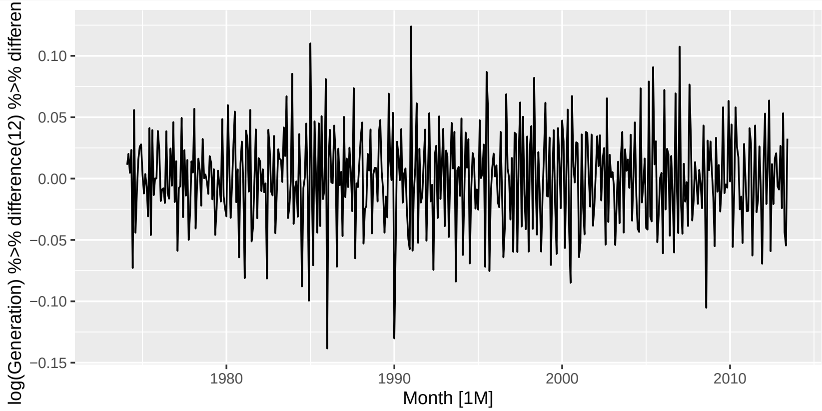
# Electricity production

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



# Electricity production

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



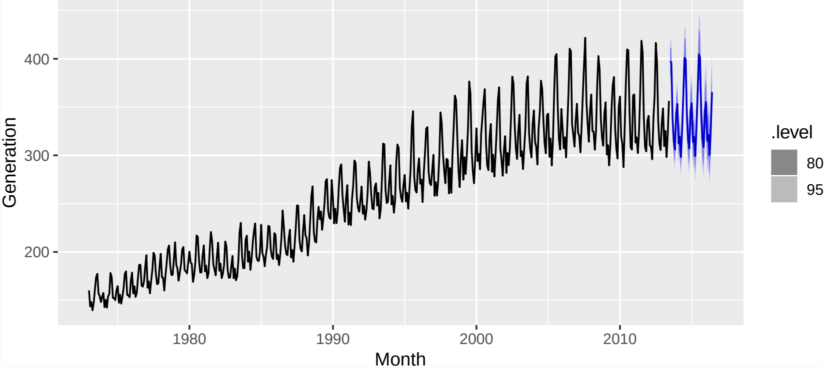
# Example: US electricity production

```r
usmelec %>%
  model(arima = ARIMA(log(Generation))) %>%
  report()
```

```
## Series: Generation
## Model: ARIMA(1,1,1)(2,1,1)[12]
## Transformation: log(.x)
##
## Coefficients:
##           ar1      ma1     sar1     sar2     sma1
##        0.4116  -0.8483   0.0100  -0.1017  -0.8204
## s.e.   0.0617   0.0348   0.0561   0.0529   0.0357
##
## sigma^2 estimated as 0.0006841:  log likelihood=1047
## AIC=-2082   AICc=-2082   BIC=-2057
```

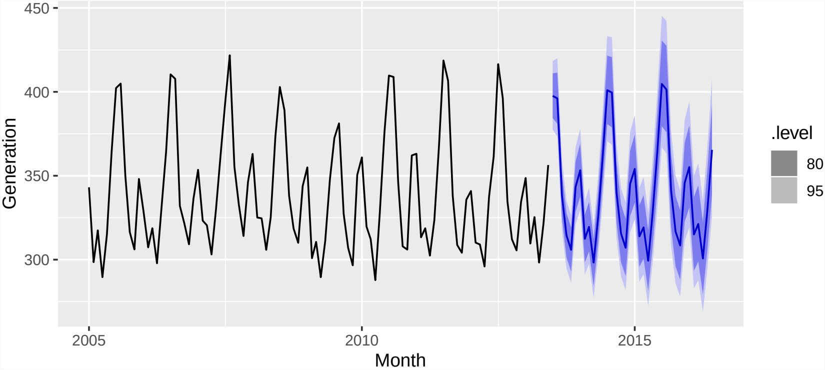
# Example: US electricity production

```
usmelec %>%
  model(arima = ARIMA(log(Generation))) %>%
  forecast(h="3 years") %>%
  autoplot(usmelec)
```



# Example: US electricity production

```
usmelec %>%
  model(arima = ARIMA(log(Generation))) %>%
  forecast(h="3 years") %>%
  autoplot(filter_index(usmelec, 2005 ~ .))
```



# 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 |

- $m$ = number of observations per year.
- $d$ first differences, $D$ seasonal differences
- $p$ AR lags, $q$ MA lags
- $P$ seasonal AR lags, $Q$ seasonal MA lags

Seasonal and non-seasonal terms combine multiplicatively

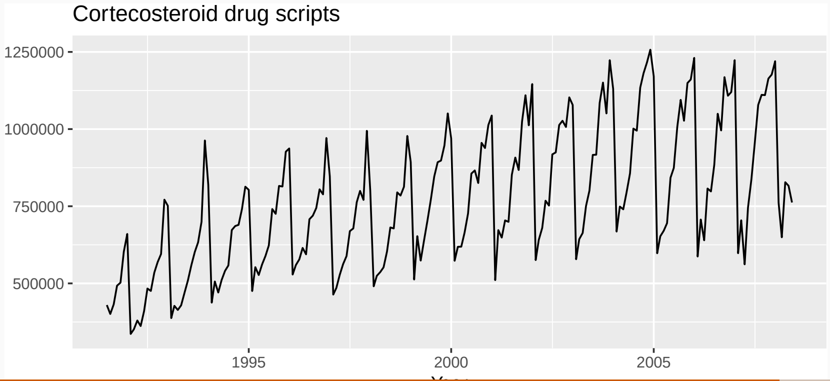
# Common ARIMA models

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 |

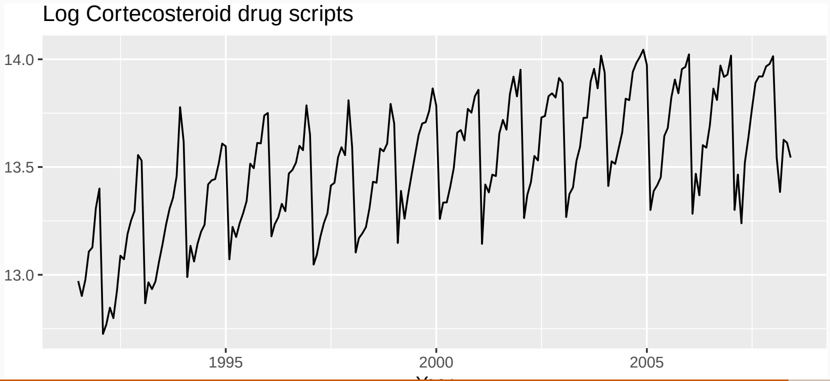
# Cortecosteroid drug sales

```r
h02 <- PBS %>% filter(ATC2 == "H02") %>%
  summarise(Cost = sum(Cost))
h02 %>% autoplot(Cost) +
  xlab("Year") + ylab("") +
  ggtitle("Cortecosteroid drug scripts")
```



# Cortecosteroid drug sales

```
h02 <- PBS %>% filter(ATC2 == "H02") %>%
  summarise(Cost = sum(Cost))
h02 %>% autoplot(log(Cost)) +
  xlab("Year") + ylab("") +
  ggtitle("Log Cortecosteroid drug scripts")
```



# Cortecosteroid drug sales

```r
fit <- h02 %>%
  model(auto = ARIMA(log(Cost)))
report(fit)
```

```
## Series: Cost
## Model: ARIMA(2,1,0)(0,1,1)[12]
## Transformation: log(.x)
##
## Coefficients:
##            ar1      ar2     sma1
##        -0.8491  -0.4207  -0.6401
## s.e.    0.0712   0.0714   0.0694
##
## sigma^2 estimated as 0.004399:  log likelihood=245
## AIC=-483   AICc=-483   BIC=-470
```

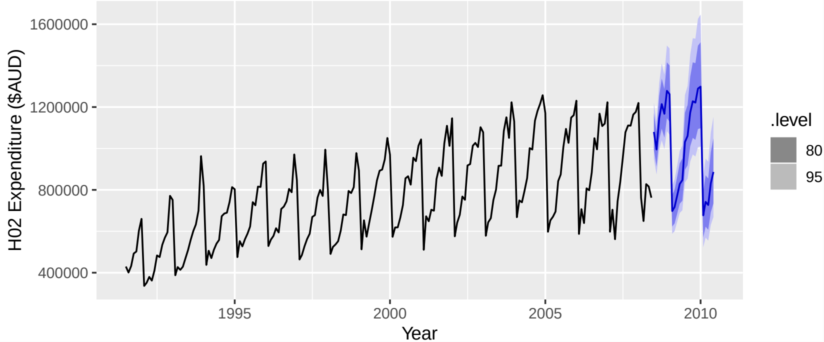
# Cortecosteroid drug sales

```
fit <- h02 %>%
  model(best = ARIMA(log(Cost), stepwise = FALSE,
                     approximation = FALSE,
                     order_constraint = p + q + P + Q <= 9))
report(fit)
```

```
## Series: Cost
## Model: ARIMA(4,1,1)(2,1,2)[12]
## Transformation: log(.x)
##
## Coefficients:
##           ar1    ar2    ar3     ar4     ma1   sar1    sar2
##       -0.0426  0.210  0.202  -0.227  -0.742  0.621  -0.383
## s.e.   0.2167  0.181  0.114   0.081   0.207  0.242   0.118
##         sma1   sma2
##       -1.202  0.496
## s.e.   0.249  0.214
##
## sigma^2 estimated as 0.004061:  log likelihood=254
## AIC=-489   AICc=-487   BIC=-456
```

# Cortecosteroid drug sales

```r
fit %>% forecast %>% autoplot(h02) +
  ylab("H02 Expenditure ($AUD)") + xlab("Year")
```



# Outline

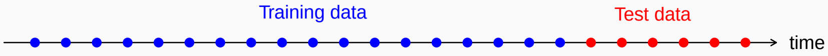
# Lab Session 9

For the Australian tourism data (from `tourism`):

- Fit a suitable ARIMA model for all data.
- Produce forecasts of your fitted models.
- Check the forecasts for the "Snowy Mountains" and "Melbourne" regions. Do they look reasonable?

# Outline

# Training and test sets




- A model which fits the training data well will not necessarily forecast well.
- Forecast accuracy is based only on the test set.

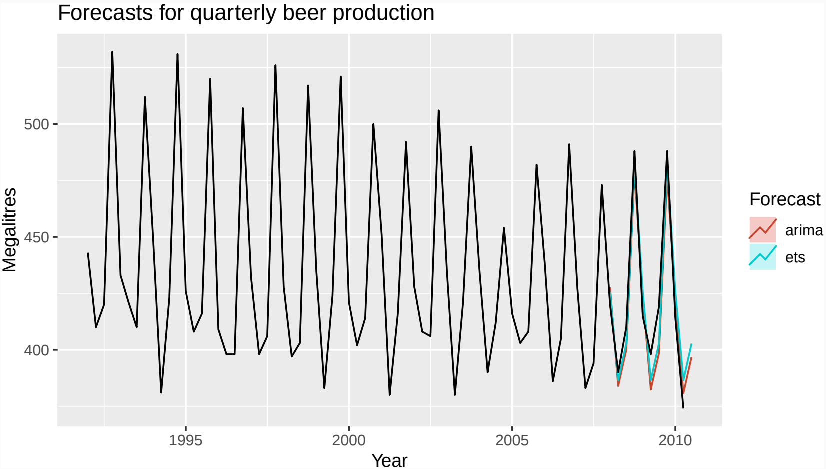
## Forecast errors

Forecast "error": the difference between an observed value and its forecast.

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T},$$

where the training data is given by $\{y_1, \dots, y_T\}$

# Measures of forecast accuracy



# Measures of forecast accuracy

$$
\begin{aligned}
y_{T+h} &= \text{$(T+h)$th observation, } h = 1, \dots, H \\
\hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\
e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T}
\end{aligned}
$$

$$\text{MAE} = \text{mean}(|e_{T+h}|)$$

$$\text{MSE} = \text{mean}(e^2_{T+h}) \qquad \text{RMSE} = \sqrt{\text{mean}(e^2_{T+h})}$$

$$\text{MAPE} = 100\text{mean}(|e_{T+h}|/|y_{T+h}|)$$

# Measures of forecast accuracy

$$
\begin{aligned}
y_{T+h} &= \text{$(T+h)$th observation, } h = 1, \dots, H \\
\hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\
e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T}
\end{aligned}
$$

$$
\begin{aligned}
\text{MAE} &= \text{mean}(|e_{T+h}|) \\
\text{MSE} &= \text{mean}(e_{T+h}^2) \qquad\qquad \text{RMSE} = \sqrt{\text{mean}(e_{T+h}^2)} \\
\text{MAPE} &= 100\text{mean}(|e_{T+h}|/|y_{T+h}|)
\end{aligned}
$$

- MAE, MSE, RMSE are all scale dependent.
- MAPE is scale independent but is only sensible if $y_t \gg 0$ for all $t$, and $y$ has a natural zero.

# Measures of forecast accuracy

## Mean Absolute Scaled Error

$$\text{MASE} = \text{mean}(|e_{T+h}|/Q)$$

where $Q$ is a stable measure of the scale of the time series $\{y_t\}$.

Proposed by Hyndman and Koehler (IJF, 2006).

For non-seasonal time series,

$$Q = (T-1)^{-1}\sum_{t=2}^{T}|y_t - y_{t-1}|$$

works well. Then MASE is equivalent to MAE relative to a naïve method.

# Measures of forecast accuracy

## Mean Absolute Scaled Error

$$\text{MASE} = \text{mean}(|e_{T+h}|/Q)$$

where $Q$ is a stable measure of the scale of the time series $\{y_t\}$.

Proposed by Hyndman and Koehler (IJF, 2006).

For seasonal time series,

$$Q = (T-m)^{-1} \sum_{t=m+1}^{T} |y_t - y_{t-m}|$$

works well. Then MASE is equivalent to MAE relative to a seasonal naïve method.

# Measures of forecast accuracy

```r
recent_production <- aus_production %>%
  filter(year(Quarter) >= 1992)
train <- recent_production %>% filter(year(Quarter) <= 2007)
beer_fit <- train %>%
  model(
    ets = ETS(Beer),
    arima = ARIMA(Beer)
  )
beer_fc <- forecast(beer_fit, h="4 years")
accuracy(beer_fc, aus_production)
```

```
## # A tibble: 2 x 9
##   .model .type    ME  RMSE   MAE   MPE  MAPE  MASE  ACF1
##   <chr>  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 arima  Test  4.18  11.2  10.4  0.940  2.47 0.657 0.145
## 2 ets    Test  0.854  9.80  8.99 0.151  2.18 0.568 0.207
```