# Tidy Time Series & Forecasting in R

7. Exponential smoothing

robjhyndman.com/workshop2020

# Outline

2

# Outline

3

# 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     E T S : **E**ponen**T**ial **S**moothing

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

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

# ETS models

General notation   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"`).

# ETS models

General notation     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

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

# 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

| | |
|---|---|
| Forecast equation | $\hat{y}_{T+h\mid T} = \ell_T$ |
| Measurement equation | $y_t = \ell_{t-1}(1 + \varepsilon_t)$ |
| State equation | $\ell_t = \ell_{t-1}(1 + \alpha \varepsilon_t)$ |

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

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

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

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.

# Outline

9

# Holt's linear trend

## Additive errors: ETS(A,A,N)

$$\text{Forecast equation} \quad \hat{y}_{T+h|T} = \ell_T + hb_T$$

$$\text{Measurement equation} \quad y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$$

$$\text{State equations} \quad \ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$$

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

# Holt's linear trend

## Additive errors: ETS(A,A,N)

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

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

## Multiplicative errors: ETS(M,A,N)

Forecast equation    $\hat{y}_{T+h|T} = \ell_T + hb_T$

Measurement equation    $y_t = (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t)$

State equations    $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$

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

# 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

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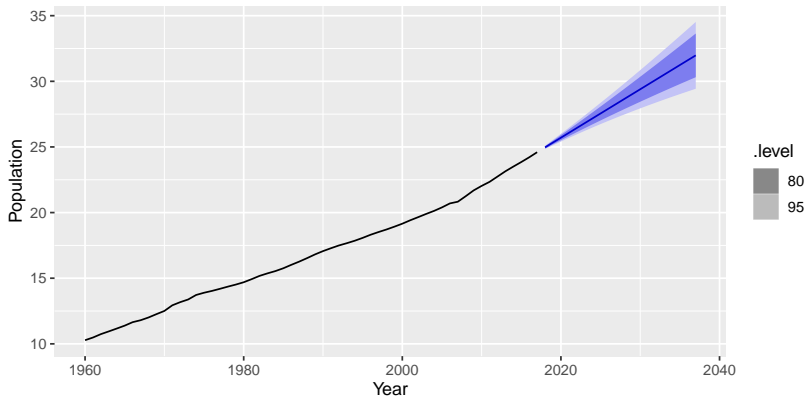


ETS(A,A,N) decomposition
Pop = lag(level, 1) + lag(slope, 1) + remainder
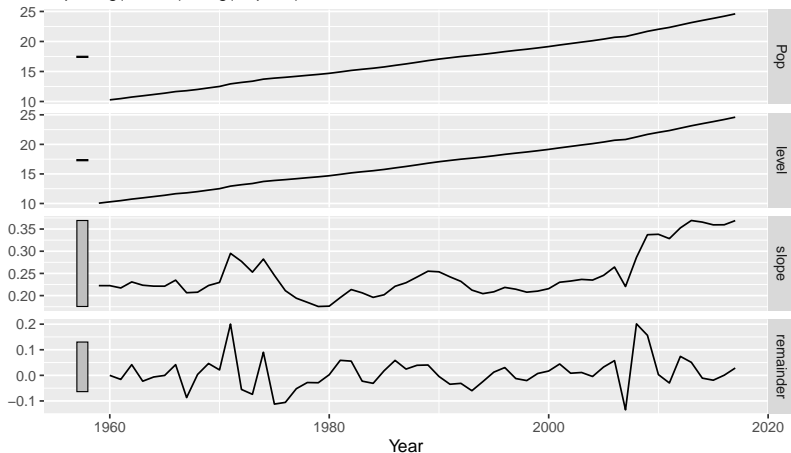
# Example: Australian population

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

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

## Additive errors

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

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

## Additive errors

$$\text{Forecast equation} \quad \hat{y}_{T+h|T} = \ell_T + (\phi + \cdots + \phi^{h-1})b_T$$

$$\text{Measurement equation} \quad y_t = (\ell_{t-1} + \phi b_{t-1}) + \varepsilon_t$$
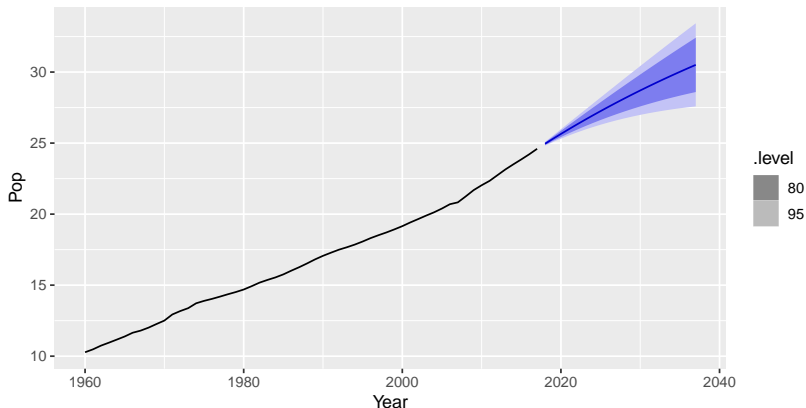
$$\text{State equations} \quad \ell_t = (\ell_{t-1} + \phi b_{t-1}) + \alpha\varepsilon_t$$

$$b_t = \phi b_{t-1} + \beta\varepsilon_t$$

- 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

```
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)
```

# Outline

19

Consider the data set `fma::eggs`, the price of a dozen eggs in the United States from 1900-1993. Experiment with the various options in the 'ETS()" function to see how much the forecasts change with damped trend, or with a Box-Cox transformation. Try to develop an intuition of what each is doing to the forecasts. How can we stop the forecasts becoming negative?

[Hint: use `h=100` when forecasting, so you can clearly see the differences between the various options when plotting the forecasts.]

# Outline

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

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

- $k$ = integer part of $(h-1)/m$.
- $\sum_i s_i \approx 0$.
- Parameters: $0 \leq \alpha \leq 1$, $0 \leq \beta^* \leq 1$, $0 \leq \gamma \leq 1 - \alpha$ 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>
```
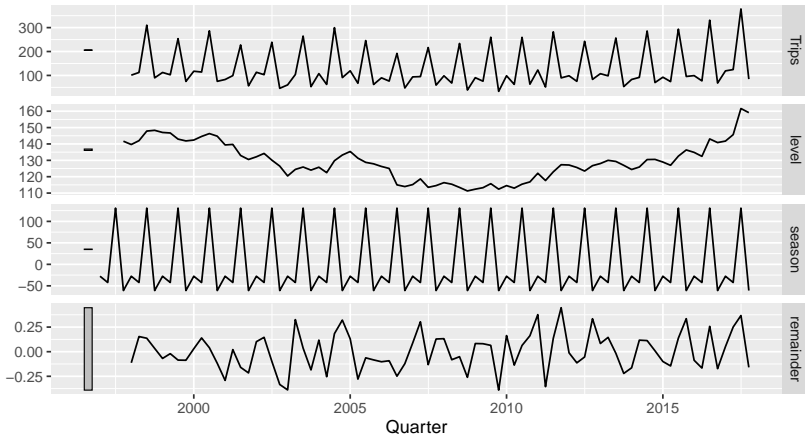
# Example: Australian holiday tourism

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



ETS(M,N,A) decomposition
Trips = (lag(level, 1) + lag(season, 4)) * (1 + remainder)

# Example: Australian holiday tourism

```
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)")
```
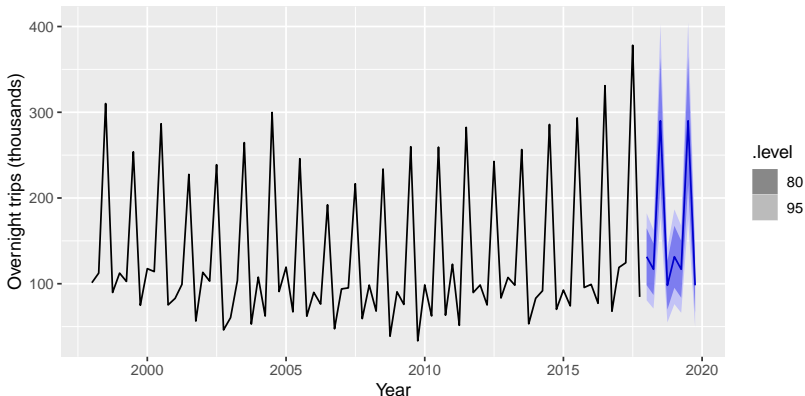
# Outline

# Exponential smoothing models

| Additive Error | Seasonal Component | | |
|---|---|---|---|
| **Trend** | N | A | M |
| **Component** | (None) | (Additive) | (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** | N | A | M |
| **Component** | (None) | (Additive) | (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 |

31

# Estimating ETS models

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

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

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

# Model selection

## Akaike's Information Criterion

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

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

## Corrected AIC

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

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

# Model selection

## Akaike's Information Criterion

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

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

## Corrected AIC

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

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

## Bayesian Information Criterion

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

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

36

## Lab Session 15

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

- Why is multiplicative seasonality necessary here?
- Experiment with making the trend damped.
  Does it improve the forecasts?