MONASH University

MONASH BUSINESS SCHOOL

# Forecasting: principles and practice

**Rob J Hyndman**

## 2.1 State space models

# Outline

# Methods V Models

## Exponential smoothing methods

- Algorithms that return point forecasts.

## Innovations state space models

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

# Methods V Models

## Exponential smoothing methods

- Algorithms that return point forecasts.

## Innovations state space models

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

# ETS models

- Each model has an *observation* equation and *transition* equations, one for each state (level, trend, seasonal), i.e., state space models.
- Two models for each method: one with additive and one with multiplicative errors, i.e., in total 18 models.
- ETS(Error,Trend,Seasonal):
  - Error = $\{A, M\}$
  - Trend = $\{N, A, A_d\}$
  - Seasonal = $\{N, A, M\}$.

# 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 |

# Exponential smoothing methods

| Trend Component | | Seasonal 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 |

**General notation** E T S : **E**xponen**T**ial **S**moothing

**E**rror **T**rend **S**easonal

**Examples:**

A,N,N: Simple exponential smoothing with additive errors
A,A,N: Holt's linear method with additive errors
M,A,M: Multiplicative Holt-Winters' method with multiplicative errors

There are 18 separate models in the ETS framework

# Exponential smoothing methods

| Trend Component | | Seasonal 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 |

**General notation** E T S : **E**xponen**T**ial **S**moothing

**E**rror **T**rend **S**easonal

**Examples:**

A,N,N: Simple exponential smoothing with additive errors
A,A,N: Holt's linear method with additive errors
M,A,M: Multiplicative Holt-Winters' method with multiplicative errors

**There are 18 separate models in the ETS framework**

# Exponential smoothing methods

| **Trend Component** | | **Seasonal 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 |

**General notation** E T S : **E**xponen**T**ial **S**moothing

**E**rror **T**rend **S**easonal

## Examples:

A,N,N: Simple exponential smoothing with additive errors
A,A,N: Holt's linear method with additive errors
M,A,M: Multiplicative Holt-Winters' method with multiplicative errors

**There are 18 separate models in the ETS framework**

# A model for SES

## Component form

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

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

## Error correction form

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

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

# A model for SES

## Component form

$$\begin{aligned} \text{Forecast equation} \qquad \hat{y}_{t+h|t} &= \ell_t \\ \text{Smoothing equation} \qquad \ell_t &= \alpha y_t + (1-\alpha)\ell_{t-1} \end{aligned}$$

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

## Error correction form

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

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

# A model for SES

## Component form

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

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

## Error correction form

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

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

# A model for SES

## Component form

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

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

## Error correction form

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

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

# 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 same error process, $\varepsilon_t$.
- Measurement equation: relationship between observations and states.
- Transition equation(s): evolution of the state(s) through time.

# 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.

# 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.

# 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.

# 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

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

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

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

# ETS(A,A,A)

Holt-Winters additive method with additive errors.

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

- Forecast errors: $\varepsilon_t = y_t - \hat{y}_{t|t-1}$
- $h_m^+ = \lfloor (h-1) \mod m \rfloor + 1.$

# Additive error models

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

# Multiplicative error models

| Trend | Seasonal N | Seasonal A | Seasonal M |
|---|---|---|---|
| **N** | $y_t = \ell_{t-1}(1+\varepsilon_t)$<br>$\ell_t = \ell_{t-1}(1+\alpha\varepsilon_t)$ | $y_t = (\ell_{t-1}+s_{t-m})(1+\varepsilon_t)$<br>$\ell_t = \ell_{t-1}+\alpha(\ell_{t-1}+s_{t-m})\varepsilon_t$<br>$s_t = s_{t-m}+\gamma(\ell_{t-1}+s_{t-m})\varepsilon_t$ | $y_t = \ell_{t-1}s_{t-m}(1+\varepsilon_t)$<br>$\ell_t = \ell_{t-1}(1+\alpha\varepsilon_t)$<br>$s_t = s_{t-m}(1+\gamma\varepsilon_t)$ |
| **A** | $y_t = (\ell_{t-1}+b_{t-1})(1+\varepsilon_t)$<br>$\ell_t = (\ell_{t-1}+b_{t-1})(1+\alpha\varepsilon_t)$<br>$b_t = b_{t-1}+\beta(\ell_{t-1}+b_{t-1})\varepsilon_t$ | $y_t = (\ell_{t-1}+b_{t-1}+s_{t-m})(1+\varepsilon_t)$<br>$\ell_t = \ell_{t-1}+b_{t-1}+\alpha(\ell_{t-1}+b_{t-1}+s_{t-m})\varepsilon_t$<br>$b_t = b_{t-1}+\beta(\ell_{t-1}+b_{t-1}+s_{t-m})\varepsilon_t$<br>$s_t = s_{t-m}+\gamma(\ell_{t-1}+b_{t-1}+s_{t-m})\varepsilon_t$ | $y_t = (\ell_{t-1}+b_{t-1})s_{t-m}(1+\varepsilon_t)$<br>$\ell_t = (\ell_{t-1}+b_{t-1})(1+\alpha\varepsilon_t)$<br>$b_t = b_{t-1}+\beta(\ell_{t-1}+b_{t-1})\varepsilon_t$<br>$s_t = s_{t-m}(1+\gamma\varepsilon_t)$ |
| $\mathbf{A_d}$ | $y_t = (\ell_{t-1}+\phi b_{t-1})(1+\varepsilon_t)$<br>$\ell_t = (\ell_{t-1}+\phi b_{t-1})(1+\alpha\varepsilon_t)$<br>$b_t = \phi b_{t-1}+\beta(\ell_{t-1}+\phi b_{t-1})\varepsilon_t$ | $y_t = (\ell_{t-1}+\phi b_{t-1}+s_{t-m})(1+\varepsilon_t)$<br>$\ell_t = \ell_{t-1}+\phi b_{t-1}+\alpha(\ell_{t-1}+\phi b_{t-1}+s_{t-m})\varepsilon_t$<br>$b_t = \phi b_{t-1}+\beta(\ell_{t-1}+\phi b_{t-1}+s_{t-m})\varepsilon_t$<br>$s_t = s_{t-m}+\gamma(\ell_{t-1}+\phi b_{t-1}+s_{t-m})\varepsilon_t$ | $y_t = (\ell_{t-1}+\phi b_{t-1})s_{t-m}(1+\varepsilon_t)$<br>$\ell_t = (\ell_{t-1}+\phi b_{t-1})(1+\alpha\varepsilon_t)$<br>$b_t = \phi b_{t-1}+\beta(\ell_{t-1}+\phi b_{t-1})\varepsilon_t$<br>$s_t = s_{t-m}(1+\gamma\varepsilon_t)$ |

# 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.
- We will estimate models with the `ets()` function in the forecast package.

# Innovations state space models

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

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

## Additive errors

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

## Multiplicative errors

$k(\boldsymbol{x}_{t-1}) = \mu_t. \qquad y_t = \mu_t(1 + \varepsilon_t).$

$\varepsilon_t = (y_t - \mu_t)/\mu_t$ is relative error.

# Innovations state space models

## Estimation

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

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

# Parameter restrictions

## *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.

## *Admissible* region

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

# Parameter restrictions

## *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.

## *Admissible* region

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

# 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

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

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

## Bayesian Information Criterion

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

# 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

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

# 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

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

# 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.

# 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.

# 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 |

# 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 $\text{E}(y_{t+h}|\boldsymbol{x}_t)$ unless trend and seasonality are both additive.
- Point forecasts for ETS(A,x,y) are identical to ETS(M,x,y) if the parameters are the same.

# Forecasting with ETS models

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

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

# Forecasting with ETS models

**Prediction intervals:** cannot be generated using the methods.

- The prediction intervals will differ between models with additive and multiplicative methods.
- 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.
- Options are available in R using the `forecast` function in the forecast package.

# Outline

# Example: drug sales

```r
ets(h02)
```

```
## ETS(M,Ad,M)
##
## Call:
##  ets(y = h02)
##
##   Smoothing parameters:
##     alpha = 0.2173
##     beta  = 2e-04
##     gamma = 1e-04
##     phi   = 0.9756
##
##   Initial states:
##     l = 0.3996
##     b = 0.0098
##     s=0.8675 0.8259 0.7591 0.7748 0.6945 1.2838
##            1.3366 1.1753 1.1545 1.0968 1.0482 0.983
##
##   sigma:  0.0647
##
##        AIC       AICc        BIC
## -123.21905 -119.52175  -63.49289
```

# Example: drug sales

```r
ets(h02, model="AAA", damped=FALSE)
```

```
## ETS(A,A,A)
##
## Call:
##  ets(y = h02, model = "AAA", damped = FALSE)
##
##   Smoothing parameters:
##     alpha = 0.1957
##     beta  = 1e-04
##     gamma = 0.4211
##
##   Initial states:
##     l = 0.4146
##     b = 0.0026
##     s=-0.1064 -0.1028 -0.1211 -0.1086 -0.161 0.2173
##            0.2306 0.0671 0.0667 0.0299 -0.0156 0.0038
##
##   sigma:  0.0538
##
##       AIC      AICc       BIC
## -73.33048 -70.04016 -16.92244
```

# The `ets()` function

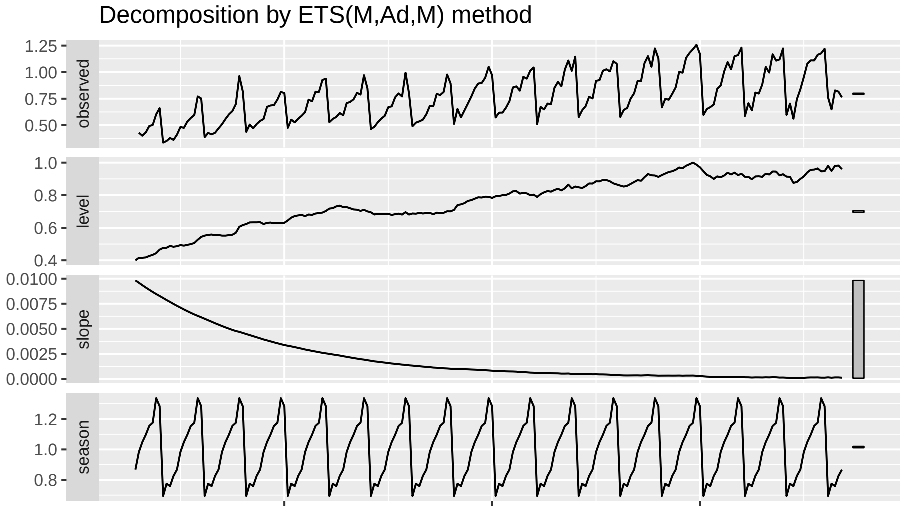
- Automatically chooses a model by default using the AIC, AICc or BIC.
- Can handle any combination of trend, seasonality and damping
- Produces prediction intervals for every model
- Ensures the parameters are admissible (equivalent to invertible)
- Produces an object of class “ets”.

# `ets` objects

- **Methods:** `coef()`, `autoplot()`, `plot()`, `summary()`, `residuals()`, `fitted()`, `simulate()` and `forecast()`
- `autoplot()` and `plot()` functions show time plots of the original time series along with the extracted components (level, growth and seasonal).

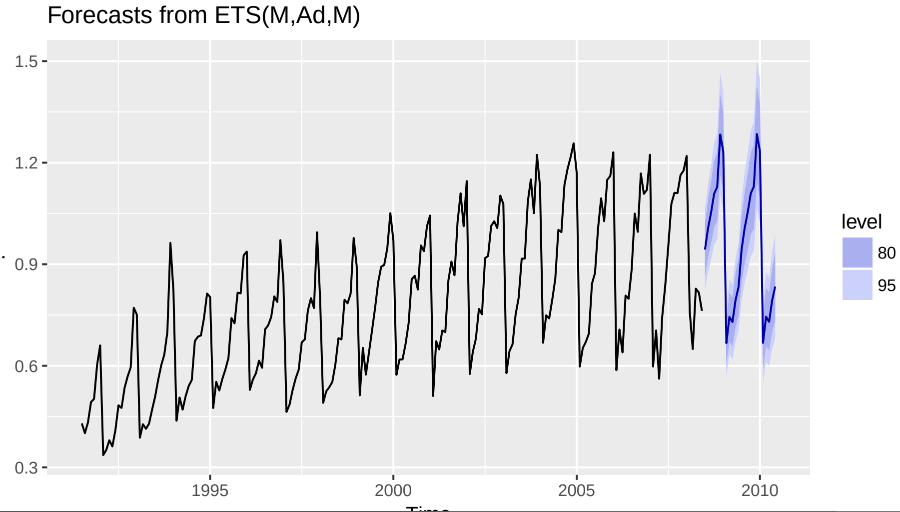
# Example: drug sales

```r
h02 %>% ets %>% autoplot
```



# Example: drug sales

```
h02 %>% ets %>% forecast %>% autoplot
```



# Example: drug sales

```r
h02 %>% ets %>% accuracy
```

```
##                       ME       RMSE        MAE       MPE     MAPE
## Training set 0.003353397 0.05144236 0.03919035 0.0437546 5.054949
##                   MASE         ACF1
## Training set 0.6465103 -0.007295816
```

```r
h02 %>% ets(model="AAA", damped=FALSE) %>% accuracy
```

```
##                       ME       RMSE        MAE        MPE     MAPE
## Training set 0.000396129 0.05381921 0.04038841 -0.3530507 5.375484
##                   MASE        ACF1
## Training set 0.6662743 -0.06641172
```

# The `ets()` function

`ets()` function also allows refitting model to new data set.

```
train <- window(h02, end=c(2004,12))
test <- window(h02, start=2005)
fit1 <- ets(train)
fit2 <- ets(test, model = fit1)
accuracy(fit2)
```

```
##                       ME       RMSE        MAE        MPE     MAPE
## Training set 0.002828719 0.0556371 0.04510416 -0.3111965 5.424523
##                   MASE       ACF1
## Training set 0.7094596 -0.3807942
```

```
accuracy(forecast(fit1,10), test)
```

```
##                        ME       RMSE        MAE        MPE      MAPE
## Training set  0.003385115 0.04466096 0.03278583   0.169324  4.332804
## Test set     -0.079220419 0.09420250 0.08237909 -10.353804 10.642337
##                   MASE        ACF1 Theil's U
## Training set 0.5560485 -0.01039592        NA
## Test set     1.3971516  0.01728834 0.6543488
```

# The ets() function in R

```
ets(y, model = "ZZZ", damped = NULL,
  additive.only = FALSE,
  lambda = NULL, biasadj = FALSE,
  lower = c(rep(1e-04, 3), 0.8),
  upper = c(rep(0.9999, 3), 0.98),
  opt.crit = c("lik", "amse", "mse", "sigma", "r
  nmse = 3,
  bounds = c("both", "usual", "admissible"),
  ic = c("aicc", "aic", "bic"),
  restrict = TRUE,
  allow.multiplicative.trend = FALSE, ...)
```

# The `ets()` function in R

- `y`
  The time series to be forecast.
- `model`
  use the ETS classification and notation: “N” for none, “A” for additive, “M” for multiplicative, or “Z” for automatic selection. Default `ZZZ` all components are selected using the information criterion.
- `damped`
- If `damped=TRUE`, then a damped trend will be used (either $A_d$ or $M_d$).
- `damped=FALSE`, then a non-damped trend will used.
- If `damped=NULL` (the default), then either a damped or a non-damped trend will be selected according to the information criterion chosen.

# The `ets()` function in R

- `additive.only`
  Only models with additive components will be considered if `additive.only=TRUE`. Otherwise all models will be considered.
- `lambda`
  Box-Cox transformation parameter. It will be ignored if `lambda=NULL` (the default value). Otherwise, the time series will be transformed before the model is estimated. When `lambda` is not `NULL`, `additive.only` is set to `TRUE`.
- `biadadj`
  Uses bias-adjustment when undoing Box-Cox transformation for fitted values.

# The `ets()` function in R

- `lower,upper` bounds for the parameter estimates of $\alpha$, $\beta^*$, $\gamma^*$ and $\phi$.
- `opt.crit=lik` (default) optimisation criterion used for estimation.
- `bounds` Constraints on the parameters.
  - *usual* region – `"bounds=usual"`;
  - *admissible* region – `"bounds=admissible"`;
  - `"bounds=both"` (the default) requires the parameters to satisfy both sets of constraints.
- `ic=aicc` (the default) information criterion to be used in selecting models.
- `restrict=TRUE` (the default) models that cause numerical difficulties are not considered in model selection.
- `allow.multiplicative.trend` allows models with a multiplicative trend.

# The `forecast()` function in R

```
forecast(object,
  h=ifelse(object$m>1, 2*object$m, 10),
  level=c(80,95), fan=FALSE,
  simulate=FALSE, bootstrap=FALSE,
  npaths=5000, PI=TRUE,
  lambda=object$lambda, biasadj=FALSE,...)
```

- `object`: the object returned by the `ets()` function.
- `h`: the number of periods to be forecast.
- `level`: the confidence level for the prediction intervals.
- `fan`: if `fan=TRUE`, suitable for fan plots.
- `simulate`: If `TRUE`, prediction intervals generated via simulation rather than analytic formulae. Even if `FALSE` simulation will be used if no algebraic formulae exist.

# The `forecast()` function in R

- `bootstrap`: If `bootstrap=TRUE` and `simulate=TRUE`, then simulated prediction intervals use re-sampled errors rather than normally distributed errors.
- `npaths`: The number of sample paths used in computing simulated prediction intervals.
- `PI`: If `PI=TRUE`, then prediction intervals are produced; otherwise only point forecasts are calculated. If `PI=FALSE`, then `level`, `fan`, `simulate`, `bootstrap` and `npaths` are all ignored.
- `lambda`: The Box-Cox transformation parameter. Ignored if `lambda=NULL`. Otherwise, forecasts are back-transformed via inverse Box-Cox transformation.
- `biasadj`: Apply bias adjustment after Box-Cox?

# Outline

# Lab Session 8