6. ETS models
OTexts.com/fpp/7/
1. Exponential smoothing methods so far
2. Holt-Winters’ seasonal method
3. Taxonomy of exponential smoothing methods
4. Exponential smoothing state space models
Exponential smoothing methods

- Simple exponential smoothing: no trend. 
  \texttt{ses(x)}

- Holt’s method: linear trend. 
  \texttt{holt(x)}

- Exponential trend method. 
  \texttt{holt(x, exponential=TRUE)}

- Damped trend method. 
  \texttt{holt(x, damped=TRUE)}

- Damped exponential trend method. 
  \texttt{holt(x, damped=TRUE, exponential=TRUE)}
Exponential smoothing methods

- **Simple exponential smoothing: no trend.**
  \[ \text{ses}(x) \]

- **Holt’s method: linear trend.**
  \[ \text{holt}(x) \]

- **Exponential trend method.**
  \[ \text{holt}(x, \text{exponential} = \text{TRUE}) \]

- **Damped trend method.**
  \[ \text{holt}(x, \text{damped} = \text{TRUE}) \]

- **Damped exponential trend method.**
  \[ \text{holt}(x, \text{damped} = \text{TRUE}, \text{exponential} = \text{TRUE}) \]
Exponential smoothing methods

- Simple exponential smoothing: no trend.
  \( \text{ses}(x) \)

- Holt’s method: linear trend.
  \( \text{holt}(x) \)

- Exponential trend method.
  \( \text{holt}(x, \text{exponential}=\text{TRUE}) \)

- Damped trend method.
  \( \text{holt}(x, \text{damped}=\text{TRUE}) \)

- Damped exponential trend method.
  \( \text{holt}(x, \text{damped}=\text{TRUE}, \text{exponential}=\text{TRUE}) \)
Exponential smoothing methods

- Simple exponential smoothing: no trend.
  \[ \text{ses}(x) \]

- Holt’s method: linear trend.
  \[ \text{holt}(x) \]

- Exponential trend method.
  \[ \text{holt}(x, \text{exponential}=\text{TRUE}) \]

- Damped trend method.
  \[ \text{holt}(x, \text{damped}=\text{TRUE}) \]

- Damped exponential trend method.
  \[ \text{holt}(x, \text{damped}=\text{TRUE}, \text{exponential}=\text{TRUE}) \]
Exponential smoothing methods

- Simple exponential smoothing: no trend.  
  \texttt{ses(x)}

- Holt's method: linear trend.  
  \texttt{holt(x)}

- Exponential trend method.  
  \texttt{holt(x, exponential=TRUE)}

- Damped trend method.  
  \texttt{holt(x, damped=TRUE)}

- Damped exponential trend method.  
  \texttt{holt(x, damped=TRUE, exponential=TRUE)}
1. Exponential smoothing methods so far

2. Holt-Winters’ seasonal method

3. Taxonomy of exponential smoothing methods

4. Exponential smoothing state space models
Holt-Winters additive method

- Holt and Winters extended Holt’s method to capture seasonality.
- Three smoothing equations—one for the level, one for trend, and one for seasonality.
- Parameters: $0 \leq \alpha \leq 1$, $0 \leq \beta^* \leq 1$, $0 \leq \gamma \leq 1 - \alpha$ and $m = \text{period of seasonality}$.

\[
\hat{y}_{t+h|t} = \ell_t + h\beta_t + s_{t-m+h}^*
\]

\[
\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1})
\]

\[
b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1}
\]

\[
s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}
\]
Holt-Winters additive method

- Holt and Winters extended Holt’s method to capture seasonality.
- Three smoothing equations—one for the level, one for trend, and one for seasonality.
- Parameters: \(0 \leq \alpha \leq 1, \ 0 \leq \beta^* \leq 1, \ 0 \leq \gamma \leq 1 - \alpha\) and \(m = \) period of seasonality.

\[
\hat{y}_{t+h|t} = \ell_t + h b_t + s_{t-m+h_m} \\
\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1}) \\
b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1} \\
s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}
\]
Holt-Winters additive method

- Holt and Winters extended Holt’s method to capture seasonality.
- Three smoothing equations—one for the level, one for trend, and one for seasonality.
- Parameters: $0 \leq \alpha \leq 1$, $0 \leq \beta^* \leq 1$, $0 \leq \gamma \leq 1 - \alpha$ and $m = \text{period of seasonality}$.

\[
\hat{y}_{t+h|t} = \ell_t + h b_t + s_{t-m+h_m} \\
\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1}) \\
b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1} \\
s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}
\]
Holt and Winters extended Holt’s method to capture seasonality.

Three smoothing equations—one for the level, one for trend, and one for seasonality.

Parameters: \( 0 \leq \alpha \leq 1, \ 0 \leq \beta^* \leq 1, \ 0 \leq \gamma \leq 1 - \alpha \) and \( m = \) period of seasonality.

\[
\hat{y}_{t+h|t} = l_t + hb_t + s_{t-m+h_m}
\]
\[
l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1})
\]
\[
b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1}
\]
\[
s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}
\]
Holt and Winters extended Holt’s method to capture seasonality.

Three smoothing equations—one for the level, one for trend, and one for seasonality.

Parameters: \(0 \leq \alpha \leq 1, \ 0 \leq \beta^* \leq 1, \ 0 \leq \gamma \leq 1 - \alpha\) and \(m = \text{period of seasonality.}\)

\[
\hat{y}_{t+h|t} = \ell_t + h b_t + s_{t-m+h_m^+}
\]
\[
\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1})
\]
\[
b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1}
\]
\[
s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}
\]
Holt-Winters additive method

- Holt and Winters extended Holt’s method to capture seasonality.
- Three smoothing equations—one for the level, one for trend, and one for seasonality.
- Parameters: \( 0 \leq \alpha \leq 1, \ 0 \leq \beta^* \leq 1, \ 0 \leq \gamma \leq 1 - \alpha \) and \( m = \) period of seasonality.

\[
\hat{y}_{t+h|t} = \ell_t + h b_t + s_{t-m+h^+_m}
\]

\[
\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1})
\]

\[
b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1}
\]

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

\[
h^+_m = \lceil (h - 1) \mod m \rceil + 1
\]
Holt-Winters multiplicative method

\[ \hat{y}_{t+h|t} = (l_t + h b_t) s_{t-m+h_m} \]

\[ l_t = \alpha \left( \frac{y_t}{s_{t-m}} \right) + (1 - \alpha)(l_{t-1} + b_{t-1}) \]

\[ b_t = \beta^* (l_t - l_{t-1}) + (1 - \beta^*) b_{t-1} \]

\[ s_t = \gamma \left( \frac{y_t}{(l_{t-1} + b_{t-1})} \right) + (1 - \gamma) s_{t-m} \]

- Most textbooks use \( s_t = \gamma \left( \frac{y_t}{l_t} \right) + (1 - \gamma) s_{t-m} \)
- We optimize for \( \alpha, \beta^*, \gamma, l_0, b_0, s_0, s_{-1}, \ldots, s_{1-m} \).
Holt-Winters multiplicative method

\[ \hat{y}_{t+h|t} = (\ell_t + h b_t) s_{t-m+h}^+ \]

\[ \ell_t = \alpha \left( \frac{y_t}{s_{t-m}} \right) + (1 - \alpha) (\ell_{t-1} + b_{t-1}) \]

\[ b_t = \beta^* (\ell_t - \ell_{t-1}) + (1 - \beta^*) b_{t-1} \]

\[ s_t = \gamma \left( \frac{y_t}{\ell_{t-1} + b_{t-1}} \right) + (1 - \gamma) s_{t-m} \]

- Most textbooks use \( s_t = \gamma \left( \frac{y_t}{\ell_t} \right) + (1 - \gamma) s_{t-m} \)
- We optimize for \( \alpha, \beta^*, \gamma, \ell_0, b_0, s_0, s_{-1}, \ldots, s_{1-m} \).
Damped Holt-Winters method

Damped Holt-Winters multiplicative method

\[ \hat{y}_{t+h|t} = [\ell_t + (1 + \phi + \phi^2 + \cdots + \phi^{h-1})b_t]s_{t-m+h_m^+} \]

\[ \ell_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)(\ell_{t-1} + \phi b_{t-1}) \]

\[ b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)\phi b_{t-1} \]

\[ s_t = \gamma[y_t/(\ell_{t-1} + \phi b_{t-1})] + (1 - \gamma)s_{t-m} \]

- This is often the single most accurate forecasting method for seasonal data.
A confusing array of methods?

- All these methods can be confusing!
- How to choose between them?
- The ETS framework provides an automatic way of selecting the best method.
- It was developed to solve the problem of automatically forecasting pharmaceutical sales across thousands of products.
All these methods can be confusing!

How to choose between them?

The ETS framework provides an automatic way of selecting the best method.

It was developed to solve the problem of automatically forecasting pharmaceutical sales across thousands of products.
All these methods can be confusing!

How to choose between them?

The ETS framework provides an automatic way of selecting the best method.

It was developed to solve the problem of automatically forecasting pharmaceutical sales across thousands of products.
All these methods can be confusing!
How to choose between them?
The ETS framework provides an automatic way of selecting the best method.
It was developed to solve the problem of automatically forecasting pharmaceutical sales across thousands of products.
1. Exponential smoothing methods so far
2. Holt-Winters’ seasonal method
3. Taxonomy of exponential smoothing methods
4. Exponential smoothing state space models
Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N (None)</td>
</tr>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A\textsubscript{d} (Additive damped)</td>
<td>A\textsubscript{d},N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M\textsubscript{d} (Multiplicative damped)</td>
<td>M\textsubscript{d},N</td>
</tr>
</tbody>
</table>
## Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N (None)</td>
</tr>
<tr>
<td>N (None)</td>
<td><strong>N,N</strong></td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
</tbody>
</table>

**(N,N):** Simple exponential smoothing
### Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N (None)</td>
</tr>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A_d (Additive damped)</td>
<td>A_d,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M_d (Multiplicative damped)</td>
<td>M_d,N</td>
</tr>
</tbody>
</table>

(N,N): Simple exponential smoothing

(A,N): Holt’s linear method
## Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
<th>N (None)</th>
<th>A (Additive)</th>
<th>M (Multiplicative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td></td>
<td>N,N</td>
<td>N,A</td>
<td>N,M</td>
</tr>
<tr>
<td>A (Additive)</td>
<td></td>
<td>A,N</td>
<td><strong>A,A</strong></td>
<td>A,M</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td></td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,A</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,M</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td></td>
<td>M,N</td>
<td>M,A</td>
<td>M,M</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td></td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,A</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,M</td>
</tr>
</tbody>
</table>

(N,N): Simple exponential smoothing  
(A,N): Holt’s linear method  
(A,A): Additive Holt-Winters’ method
## Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N (None)</td>
</tr>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
</tbody>
</table>

- **(N,N):** Simple exponential smoothing
- **(A,N):** Holt’s linear method
- **(A,A):** Additive Holt-Winters’ method
- **(A,M):** Multiplicative Holt-Winters’ method
### Exponential smoothing methods

#### Trend Component

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>Ad (Additive damped)</td>
<td>Ad,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>Md (Multiplicative damped)</td>
<td>Md,N</td>
</tr>
</tbody>
</table>

#### Seasonal Component

- **N (None)**
  - N,N
  - A,N
  - Ad,N
  - M,N
  - Md,N

- **A (Additive)**
  - N,A
  - A,A
  - Ad,A
  - M,A
  - Md,A

- **M (Multiplicative)**
  - N,M
  - A,M
  - Ad,M
  - M,M
  - Md,M

#### Explanation

- **(N,N):** Simple exponential smoothing
- **(A,N):** Holt’s linear method
- **(A,A):** Additive Holt-Winters’ method
- **(A,M):** Multiplicative Holt-Winters’ method
- **(Ad,M):** Damped multiplicative Holt-Winters’ method
### Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N, N,A, N,M</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N, A,A, A,M</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N, A&lt;sub&gt;d&lt;/sub&gt;,A, A&lt;sub&gt;d&lt;/sub&gt;,M</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N, M,A, M,M</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N, M&lt;sub&gt;d&lt;/sub&gt;,A, M&lt;sub&gt;d&lt;/sub&gt;,M</td>
</tr>
</tbody>
</table>

- **(N,N):** Simple exponential smoothing
- **(A,N):** Holt’s linear method
- **(A,A):** Additive Holt-Winters’ method
- **(A,M):** Multiplicative Holt-Winters’ method
- **(A<sub>d</sub>,M):** Damped multiplicative Holt-Winters’ method

There are 15 separate exponential smoothing methods.
**ses()** implements method \((N,N)\)

**holt()** implements methods \((A,N), (A_d,N), (M,N), (M_d,N)\)

**hw()** implements methods \((A,A), (A_d,A), (A,M), (A_d,M), (M,M), (M_d,M)\).
R functions

- `ses()` implements method (N,N)
- `holt()` implements methods (A,N), (A_d,N), (M,N), (M_d,N)
- `hw()` implements methods (A,A), (A_d,A), (A,M), (A_d,M), (M,M), (M_d,M).
ses() implements method (N,N)

holt() implements methods (A,N), (A_d,N), (M,N), (M_d,N)

hw() implements methods (A,A), (A_d,A), (A,M), (A_d,M), (M,M), (M_d,M).
Outline

1 Exponential smoothing methods so far
2 Holt-Winters’ seasonal method
3 Taxonomy of exponential smoothing methods
4 Exponential smoothing state space models
Until recently, there has been no stochastic modelling framework incorporating likelihood calculation, prediction intervals, etc.

Ord, Koehler & Snyder (JASA, 1997) and Hyndman, Koehler, Snyder and Grose (IJF, 2002) showed that all ES methods (including non-linear methods) are optimal forecasts from innovations state space models.

Hyndman et al. (2008) provides a comprehensive and up-to-date survey of area. The forecast package implements the state space framework.
Until recently, there has been no stochastic modelling framework incorporating likelihood calculation, prediction intervals, etc.

Ord, Koehler & Snyder (JASA, 1997) and Hyndman, Koehler, Snyder and Grose (IJF, 2002) showed that all ES methods (including non-linear methods) are optimal forecasts from innovations state space models.

Hyndman et al. (2008) provides a comprehensive and up-to-date survey of area.

The forecast package implements the state space framework.
Until recently, there has been no stochastic modelling framework incorporating likelihood calculation, prediction intervals, etc.

Ord, Koehler & Snyder (JASA, 1997) and Hyndman, Koehler, Snyder and Grose (IJF, 2002) showed that all ES methods (including non-linear methods) are optimal forecasts from innovations state space models.

Hyndman et al. (2008) provides a comprehensive and up-to-date survey of area.

The forecast package implements the state space framework.
Until recently, there has been no stochastic modelling framework incorporating likelihood calculation, prediction intervals, etc. Ord, Koehler & Snyder (JASA, 1997) and Hyndman, Koehler, Snyder and Grose (IJF, 2002) showed that all ES methods (including non-linear methods) are optimal forecasts from innovations state space models. Hyndman et al. (2008) provide a comprehensive and up-to-date survey of the area. The forecast package implements the state space framework.
Until recently, there has been no stochastic modelling framework incorporating likelihood calculation, prediction intervals, etc.

Ord, Koehler & Snyder (JASA, 1997) and Hyndman, Koehler, Snyder and Grose (IJF, 2002) showed that all ES methods (including non-linear methods) are optimal forecasts from innovations state space models.

Hyndman et al. (2008) provides a comprehensive and up-to-date survey of area.

The forecast package implements the state space framework.
Exponential smoothing

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>Ad (Additive damped)</td>
<td>Ad,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>Md (Multiplicative damped)</td>
<td>Md,N</td>
</tr>
</tbody>
</table>

General notation: \text{ETS}(\text{Error,Trend,Seasonal})
### Exponential smoothing

Exponential smoothing state space models

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>Ad (Additive damped)</td>
<td>Ad,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>Md (Multiplicative damped)</td>
<td>Md,N</td>
</tr>
</tbody>
</table>

**General notation**  **ETS***(Error,Trend,Seasonal)*
### Exponential smoothing

#### General notation

**ETS**\((Error,Trend,Seasonal)\)

**Exponential Smoothing**

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A(_d) (Additive damped)</td>
<td>A(_d),N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M(_d) (Multiplicative damped)</td>
<td>M(_d),N</td>
</tr>
<tr>
<td></td>
<td>N,A</td>
</tr>
<tr>
<td></td>
<td>A,A</td>
</tr>
<tr>
<td></td>
<td>A(_d),A</td>
</tr>
<tr>
<td></td>
<td>M,A</td>
</tr>
<tr>
<td></td>
<td>M(_d),A</td>
</tr>
<tr>
<td></td>
<td>N,M</td>
</tr>
<tr>
<td></td>
<td>A,M</td>
</tr>
<tr>
<td></td>
<td>A(_d),M</td>
</tr>
<tr>
<td></td>
<td>M,M</td>
</tr>
<tr>
<td></td>
<td>M(_d),M</td>
</tr>
</tbody>
</table>
Exponential smoothing

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
</tr>
<tr>
<td>(None)</td>
<td>N,N</td>
</tr>
<tr>
<td>(Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>(Additive damped)</td>
<td>A_d,N</td>
</tr>
<tr>
<td>(Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>(Multiplicative damped)</td>
<td>M_d,N</td>
</tr>
</tbody>
</table>

**General notation**  \textbf{ETS}(\textit{Error},\textit{Trend},\textit{Seasonal})

**Exponential Smoothing**

**ETS(A,N,N):** Simple exponential smoothing with additive errors
### Exponential smoothing

#### General notation

**ETS(\textit{Error, Trend, Seasonal})**

**Exponential Smoothing**

**ETS(A,A,N):** Holt’s linear method with additive errors

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(N)</td>
</tr>
<tr>
<td>(N) (None)</td>
<td>(N,N)</td>
</tr>
<tr>
<td>(A) (Additive)</td>
<td>(A,N)</td>
</tr>
<tr>
<td>(A_d) (Additive damped)</td>
<td>(A_d,N)</td>
</tr>
<tr>
<td>(M) (Multiplicative)</td>
<td>(M,N)</td>
</tr>
<tr>
<td>(M_d) (Multiplicative damped)</td>
<td>(M_d,N)</td>
</tr>
</tbody>
</table>
# Exponential smoothing

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N</td>
<td>N,A</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
<td>A,A</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,A</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
<td>M,A</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,A</td>
</tr>
</tbody>
</table>

**General notation**  
\( \text{ETS}(\text{Error}, \text{Trend}, \text{Seasonal}) \)

**Exponential Smoothing**

**ETS(A,A,A):** Additive Holt-Winters’ method with additive errors
**Exponential smoothing**

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>(None) N, N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>(Additive) A, N</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;, N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>(Multiplicative) M</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;, N</td>
</tr>
</tbody>
</table>

**General notation**  
**ETS** *(Error, Trend, Seasonal)*  
**Exponential Smoothing**

**ETS**(M,A,M): Multiplicative Holt-Winters’ method with multiplicative errors
### Exponential smoothing

#### General notation

**ETS**(*Error*,*Trend*,*Seasonal*)

**Exponential Smoothing**

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N (None)</td>
</tr>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt; (Additive damped)</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt; (Multiplicative damped)</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
</tbody>
</table>

**ETS(A,A<sub>d</sub>,N):** Damped trend method with additive errors
## Exponential smoothing

There are 30 separate models in the ETS framework.

### General notation

ETS\((Error, Trend, Seasonal)\)

Exponential Smoothing

### Trend Component

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (None)</td>
<td>N,N</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
</tr>
<tr>
<td>A(_d) (Additive damped)</td>
<td>A(_d),N</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>M(_d) (Multiplicative damped)</td>
<td>M(_d),N</td>
</tr>
</tbody>
</table>

### Seasonal Component

<table>
<thead>
<tr>
<th>N (None)</th>
<th>A (Additive)</th>
<th>M (Multiplicative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N,N</td>
<td>N,A</td>
<td>N,M</td>
</tr>
<tr>
<td>A,N</td>
<td>A,A</td>
<td>A,M</td>
</tr>
<tr>
<td>A(_d),N</td>
<td>A(_d),A</td>
<td>A(_d),M</td>
</tr>
<tr>
<td>M,N</td>
<td>M,A</td>
<td>M,M</td>
</tr>
<tr>
<td>M(_d),N</td>
<td>M(_d),A</td>
<td>M(_d),M</td>
</tr>
</tbody>
</table>

**Error, Trend, Seasonal**
Innovations state space models

**SES**

\[ \hat{y}_{t+1|t} = l_t \]

\[ l_t = \alpha y_t + (1 - \alpha) l_{t-1} \]
Innovations state space models

**SES**

\[ \hat{y}_{t+1|t} = \ell_t \]
\[ \ell_t = \alpha y_t + (1 - \alpha) \ell_{t-1} \]

If \( \varepsilon_t = y_t - \hat{y}_{t-1|t} \)
\( \sim \text{NID}(0, \sigma^2) \), then

**ETS(A,N,N)**

\[ y_t = \ell_{t-1} + \varepsilon_t \]
\[ \ell_t = \alpha y_t + (1 - \alpha) \ell_{t-1} \]
\( = \ell_{t-1} + \alpha \varepsilon_t \)
Innovations state space models

**SES**

\[
\hat{y}_{t+1|t} = l_t \\
l_t = \alpha y_t + (1 - \alpha)l_{t-1}
\]

If \( \varepsilon_t = y_t - \hat{y}_{t-1|t} \)

\( \sim \text{NID}(0, \sigma^2) \), then

**ETS(A,N,N)**

\[
y_t = l_{t-1} + \varepsilon_t \\
l_t = \alpha y_t + (1 - \alpha)l_{t-1} \\
= l_{t-1} + \alpha \varepsilon_t
\]

**ETS(M,N,N)**

\[
y_t = l_{t-1}(1 + \varepsilon_t) \\
l_t = \alpha y_t + (1 - \alpha)l_{t-1} \\
= l_{t-1}(1 + \alpha \varepsilon_t)
\]
Innovations state space models

**SES**

\[ \hat{y}_{t+1|t} = \ell_t \]
\[ \ell_t = \alpha y_t + (1 - \alpha)\ell_{t-1} \]

If \( \varepsilon_t = y_t - \hat{y}_{t-1|t} \)
\[ \sim \text{NID}(0, \sigma^2) \], then

**ETS(A,N,N)**

\[ y_t = \ell_{t-1} + \varepsilon_t \]
\[ \ell_t = \alpha y_t + (1 - \alpha)\ell_{t-1} \]
\[ = \ell_{t-1} + \alpha \varepsilon_t \]

**ETS(M,N,N)**

\[ y_t = \ell_{t-1}(1 + \varepsilon_t) \]
\[ \ell_t = \alpha y_t + (1 - \alpha)\ell_{t-1} \]
\[ = \ell_{t-1}(1 + \alpha \varepsilon_t) \]

All exponential smoothing methods can be written using analogous state space equations.
Innovations state space models

**Example:** Holt-Winters’ multiplicative seasonal method

**ETS(M,A,M)**

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

\[
\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)
\]

\[
b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t
\]

\[
s_t = s_{t-m}(1 + \gamma\varepsilon_t)
\]

where \( \beta = \alpha\beta^* \).
Innovations state space models

- All the methods can be written in this state space form.
- Prediction intervals can be obtained by simulating many future sample paths.
- For many models, the prediction intervals can be obtained analytically as well.
- Additive and multiplicative versions give the same point forecasts.
- Estimation is handled via maximizing the likelihood of the data given the model.
Innovations state space models

- All the methods can be written in this state space form.
- Prediction intervals can be obtained by simulating many future sample paths.
- For many models, the prediction intervals can be obtained analytically as well.
- Additive and multiplicative versions give the same point forecasts.
- Estimation is handled via maximizing the likelihood of the data given the model.
Innovations state space models

- All the methods can be written in this state space form.
- Prediction intervals can be obtained by simulating many future sample paths.
- For many models, the prediction intervals can be obtained analytically as well.
- Additive and multiplicative versions give the same point forecasts.
- Estimation is handled via maximizing the likelihood of the data given the model.
All the methods can be written in this state space form.

Prediction intervals can be obtained by simulating many future sample paths.

For many models, the prediction intervals can be obtained analytically as well.

Additive and multiplicative versions give the same point forecasts.

Estimation is handled via maximizing the likelihood of the data given the model.
All the methods can be written in this state space form.

Prediction intervals can be obtained by simulating many future sample paths.

For many models, the prediction intervals can be obtained analytically as well.

Additive and multiplicative versions give the same point forecasts.

Estimation is handled via maximizing the likelihood of the data given the model.
Akaike’s Information Criterion

\[ AIC = -2 \log(\text{Likelihood}) + 2p \]

where \( p \) is the number of estimated parameters in the model.

Minimizing the AIC gives the best model for prediction.

**AIC corrected (for small sample bias)**

\[ AIC_c = AIC + \frac{2(p + 1)(p + 2)}{n - p} \]

**Schwartz’ Bayesian IC**

\[ BIC = AIC + p(\log(n) - 2) \]
Akaike’s Information Criterion

\[ \text{AIC} = -2 \log(\text{Likelihood}) + 2p \]

where \( p \) is the number of estimated parameters in the model.

- **Minimizing** the AIC gives the best model for prediction.

**AIC corrected (for small sample bias)**

\[ \text{AIC}_C = \text{AIC} + \frac{2(p + 1)(p + 2)}{n - p} \]

**Schwartz’ Bayesian IC**

\[ \text{BIC} = \text{AIC} + p(\log(n) - 2) \]
Akaike’s Information Criterion

$$\text{AIC} = -2 \log(\text{Likelihood}) + 2p$$

where $p$ is the number of estimated parameters in the model.

- Minimizing the AIC gives the best model for prediction.

### AIC corrected (for small sample bias)

$$\text{AIC}_c = \text{AIC} + \frac{2(p + 1)(p + 2)}{n - p}$$

### Schwartz’ Bayesian IC

$$\text{BIC} = \text{AIC} + p(\log(n) - 2)$$
Akaike’s Information Criterion

AIC = \(-2 \log(\text{Likelihood}) + 2p\)

where \(p\) is the number of estimated parameters in the model.

- Minimizing the AIC gives the best model for prediction.

\[\text{AIC corrected (for small sample bias)}\]

\[\text{AIC}_{\text{C}} = \text{AIC} + \frac{2(p + 1)(p + 2)}{n - p}\]

Schwartz’ Bayesian IC

\[\text{BIC} = \text{AIC} + p(\log(n) - 2)\]
Akaike’s Information Criterion

\[ \text{AIC} = -2 \log(\text{Likelihood}) + 2p \]

where \( p \) is the number of estimated parameters in the model.

- Minimizing the AIC gives the best model for prediction.

AIC corrected (for small sample bias)

\[ \text{AIC}_c = \text{AIC} + \frac{2(p+1)(p+2)}{n-p} \]

Schwartz’ Bayesian IC

\[ \text{BIC} = \text{AIC} + p(\log(n) - 2) \]
Value of AIC/AICc/BIC given in the R output.

AIC does not have much meaning by itself. Only useful in comparison to AIC value for another model fitted to same data set.

Consider several models with AIC values close to the minimum.

A difference in AIC values of 2 or less is not regarded as substantial and you may choose the simpler but non-optimal model.

AIC can be negative.
Akaike’s Information Criterion

- Value of AIC/AICc/BIC given in the R output.
- AIC does not have much meaning by itself. Only useful in comparison to AIC value for another model fitted to same data set.
- Consider several models with AIC values close to the minimum.
- A difference in AIC values of 2 or less is not regarded as substantial and you may choose the simpler but non-optimal model.
- AIC can be negative.
Value of AIC/AICc/BIC given in the R output.

AIC does not have much meaning by itself. Only useful in comparison to AIC value for another model fitted to *same data set*.

Consider several models with AIC values close to the minimum.

A difference in AIC values of 2 or less is not regarded as substantial and you may choose the simpler but non-optimal model.

AIC can be negative.
Akaike’s Information Criterion

- Value of AIC/AICc/BIC given in the R output.
- AIC does not have much meaning by itself. Only useful in comparison to AIC value for another model fitted to same data set.
- Consider several models with AIC values close to the minimum.
- A difference in AIC values of 2 or less is not regarded as substantial and you may choose the simpler but non-optimal model.
- AIC can be negative.
Value of AIC/AICc/BIC given in the R output.

AIC does not have much meaning by itself. Only useful in comparison to AIC value for another model fitted to the same data set.

Consider several models with AIC values close to the minimum.

A difference in AIC values of 2 or less is not regarded as substantial and you may choose the simpler but non-optimal model.

AIC can be negative.
Exponential smoothing

From Hyndman et al. (2008):

- Apply each of 30 methods that are appropriate to the data. Estimate parameters and initial values using MLE.
- Select best method using AIC.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.

Method performed very well in M3 competition.
Exponential smoothing

From Hyndman et al. (2008):

- Apply each of 30 methods that are appropriate to the data. Estimate parameters and initial values using MLE.
- Select best method using AIC.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.

Method performed very well in M3 competition.
Exponential smoothing

From Hyndman et al. (2008):

- Apply each of 30 methods that are appropriate to the data. Estimate parameters and initial values using MLE.
- Select best method using AIC.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.

Method performed very well in M3 competition.
Exponential smoothing

From Hyndman et al. (2008):

- Apply each of 30 methods that are appropriate to the data. Estimate parameters and initial values using MLE.
- Select best method using AIC.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.

Method performed very well in M3 competition.
From Hyndman et al. (2008):

- Apply each of 30 methods that are appropriate to the data. Estimate parameters and initial values using MLE.
- Select best method using AIC.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.

Method performed very well in M3 competition.
Exponential smoothing

From Hyndman et al. (2008):

- Apply each of 30 methods that are appropriate to the data. Estimate parameters and initial values using MLE.
- Select best method using AIC.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.

Method performed very well in M3 competition.
fit <- ets(ausbeer)
fit2 <- ets(ausbeer, model="AAA", damped=FALSE)
fcast1 <- forecast(fit, h=20)
fcast2 <- forecast(fit2, h=20)

ets(y, model="ZZZ", damped=NULL, alpha=NULL,
    beta=NULL, gamma=NULL, phi=NULL,
    additive.only=FALSE,
    lower=c(rep(0.0001,3),0.80),
    upper=c(rep(0.9999,3),0.98),
    opt.crit=c("lik","amse","mse","sigma"), nmse=3,
    bounds=c("both","usual","admissible"),
    ic=c("aic","aicc","bic"), restrict=TRUE)
Exponential smoothing

```r
fit <- ets(ausbeer)
fit2 <- ets(ausbeer,model="AAA",damped=FALSE)
fcast1 <- forecast(fit, h=20)
fcast2 <- forecast(fit2, h=20)

es(y, model="ZZZ", damped=NULL, alpha=NULL,
   beta=NULL, gamma=NULL, phi=NULL,
   additive.only=FALSE,
   lower=c(rep(0.0001,3),0.80),
   upper=c(rep(0.9999,3),0.98),
   opt.crit=c("lik","amse","mse","sigma"), nmse=3,
   bounds=c("both","usual","admissible"),
   ic=c("aic","aicc","bic"), restrict=TRUE)
```
Exponential smoothing

> fit
ETS(M,Md,M)

Smoothing parameters:
   alpha = 0.1776
   beta  = 0.0454
   gamma = 0.1947
   phi   = 0.9549

Initial states:
   l = 263.8531
   b = 0.9997
   s = 1.1856 0.9109 0.8612 1.0423

sigma: 0.0356

AIC   AICc    BIC
2272.549 2273.444 2302.715
Exponential smoothing

> fit2
ETS(A,A,A)

Smoothing parameters:
  alpha = 0.2079
  beta  = 0.0304
  gamma = 0.2483

Initial states:
  l = 255.6559
  b = 0.5687
  s = 52.3841  -27.1061  -37.6758  12.3978

sigma: 15.9053

AIC   AICc    BIC
2312.768 2313.481 2339.583
**Exponential smoothing**

### 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`. 
**Exponential smoothing**

---

**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`.

---
**Exponential smoothing**

```r
define 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.
```
Exponential smoothing

**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() 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.
Exponential smoothing

**ets objects**

- **Methods**: `coef()`, `plot()`, `summary()`, `residuals()`, `fitted()`, `simulate()` and `forecast()`

- `plot()` function shows time plots of the original time series along with the extracted components (level, growth and seasonal).
Exponential smoothing

ets objects

- **Methods**: `coef()`, `plot()`, `summary()`, `residuals()`, `fitted()`, `simulate()` and `forecast()`

- `plot()` function shows time plots of the original time series along with the extracted components (level, growth and seasonal).
Decomposition by ETS(M,Md,M) method

plot(fit)
Goodness-of-fit

```r
> accuracy(fit)
    ME     RMSE    MAE     MPE    MAPE    MASE
0.17847 15.48781 11.77800 0.07204 2.81921 0.20705

> accuracy(fit2)
    ME     RMSE    MAE     MPE    MAPE    MASE
-0.11711 15.90526 12.18930 -0.03765 2.91255 0.21428
```
Forecast intervals

Forecasts from ETS(M,Md,M)

> plot(forecast(fit, level=c(50, 80, 95)))
Forecast intervals

> plot(forecast(fit, fan=TRUE))
Exponential smoothing

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

```r
> usfit <- ets(usnetelec[1:45])
> test <- ets(usnetelec[46:55], model = usfit)

> accuracy(test)

   ME  RMSE   MAE  MPE  MAPE   MASE
-3.354 58.03 43.854 -0.076 1.185  0.524

> accuracy(forecast(usfit,10), usnetelec[46:55])

   ME  RMSE   MAE  MPE  MAPE   MASE
 40.70 61.21 46.325 1.098 1.262  0.678
```
Unstable models

- ETS(M,M,A)
- ETS(M,M_{d},A)
- ETS(A,N,M)
- ETS(A,A,M)
- ETS(A,A_{d},M)
- ETS(A,M,N)
- ETS(A,M,N)
- ETS(A,M,A)
- ETS(A,M,A)
- ETS(A,M,M)
- ETS(A,M,M)
- ETS(A,M_{d},N)
- ETS(A,M_{d},N)
- ETS(A,M_{d},A)
- ETS(A,M_{d},A)
- ETS(A,M_{d},M)
- ETS(A,M_{d},M)
Unstable models

- ETS(M,M,A)
- ETS(M,M_d,A)
- ETS(A,N,M)
- ETS(A,A,M)
- ETS(A,A_d,M)
- ETS(A,M,N)
- ETS(A,M,A)
- ETS(A,M,M)
- ETS(A,M_d,N)
- ETS(A,M_d,A)
- ETS(A,M_d,M)

In practice, the models work fine for short- to medium-term forecasts provided the data are strictly positive.
ets(y, model="ZZZ", damped=NULL, alpha=NULL, beta=NULL, gamma=NULL, phi=NULL, additive.only=FALSE, lower=c(rep(0.0001,3),0.80), upper=c(rep(0.9999,3),0.98), opt.crit=c("lik","amse","mse","sigma"), nmse=3, bounds=c("both","usual","admissible"), ic=c("aic","aicc","bic"), restrict=TRUE)
The magic `forecast()` function

- `forecast` returns forecasts when applied to an `ets` object (or the output from many other time series models).

- If you use `forecast` directly on data, it will select an ETS model automatically and then return forecasts.
The magic forecast() function

- `forecast` returns forecasts when applied to an `ets` object (or the output from many other time series models).
- If you use `forecast` directly on data, it will select an ETS model automatically and then return forecasts.