Rob J Hyndman

Automatic algorithms for time series forecasting
Follow along using R

Requirements
Install the fpp package and its dependencies.
Motivation

1. Common in business to have over 1000 products that need forecasting at least monthly.
2. Forecasts are often required by people who are untrained in time series analysis.

Specifications

Automatic forecasting algorithms must:
- determine an appropriate time series model;
- estimate the parameters;
- compute the forecasts with prediction intervals.
Motivation

1. Common in business to have over 1000 products that need forecasting at least monthly.
2. Forecasts are often required by people who are untrained in time series analysis.

Specifications

Automatic forecasting algorithms must:

- determine an appropriate time series model;
- estimate the parameters;
- compute the forecasts with prediction intervals.
Example: Asian sheep

Numbers of sheep in Asia

<table>
<thead>
<tr>
<th>Year</th>
<th>Millions of Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>1960</td>
<td>250</td>
</tr>
<tr>
<td>1970</td>
<td>300</td>
</tr>
<tr>
<td>1980</td>
<td>350</td>
</tr>
<tr>
<td>1990</td>
<td>400</td>
</tr>
<tr>
<td>2000</td>
<td>450</td>
</tr>
<tr>
<td>2010</td>
<td>500</td>
</tr>
</tbody>
</table>
Example: Asian sheep

Automatic ETS forecasts

Year
millions of sheep
250 300 350 400 450 500 550
Example: Asian sheep

```r
library(fpp)
fit <- ets(livestock)
fcast <- forecast(fit)
plot(fcast)
```
Example: Cortecosteroid sales

Monthly cortecosteroid drug sales in Australia

<table>
<thead>
<tr>
<th>Year</th>
<th>Total scripts (millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1995</td>
<td>0.4</td>
</tr>
<tr>
<td>2000</td>
<td>0.6</td>
</tr>
<tr>
<td>2005</td>
<td>0.8</td>
</tr>
<tr>
<td>2010</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Year
Example: Corticosteroid sales

Forecasts from ARIMA(3,1,3)(0,1,1)[12]

<table>
<thead>
<tr>
<th>Year</th>
<th>Total scripts (millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1995</td>
<td>0.4</td>
</tr>
<tr>
<td>2000</td>
<td>0.6</td>
</tr>
<tr>
<td>2005</td>
<td>0.8</td>
</tr>
<tr>
<td>2010</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Total scripts (millions)

Year

Automatic algorithms for time series forecasting
Auto ARIMA

fit <- auto.arima(h02)
fcast <- forecast(fit)
plot(fcast)
1 Forecasting competitions
2 Exponential smoothing
3 ARIMA modelling
4 Automatic nonlinear forecasting?
5 Time series with complex seasonality
6 Recent developments
Accuracy of Forecasting: An Empirical Investigation

By Spyros Makridakis and Michèle Hibon

INSEAD—The European Institute of Business Administration

[Read before the Royal Statistical Society on Wednesday, December 13th, 1978, the President, Sir Claus Moser in the Chair]

Summary

In this study, the authors used 111 time series to examine the accuracy of various forecasting methods, particularly time-series methods. The study shows, at least for time series, why some methods achieve greater accuracy than others for different types of data. The authors offer some explanation of the seemingly conflicting conclusions of past empirical research on the accuracy of forecasting. One novel contribution of the paper is the development of regression equations expressing accuracy as a function of factors such as randomness, seasonality, trend-cycle and the number of data points describing the series. Surprisingly, the study shows that for these 111 series simpler methods perform well in comparison to the more complex and statistically sophisticated ARMA models.

Keywords: Forecasting; Time Series; Forecasting Accuracy

0. Introduction

The ultimate test of any forecast is whether or not it is capable of predicting future events accurately. Planning and decision making require accurate forecasts for effective planning.
Accuracy of Forecasting: An Empirical Investigation

By Spyros Makridakis and Michèle Hibon

INSEAD—The European Institute of Business Administration

SUMMARY

The authors used 111 time series to examine the accuracy of various time-series methods. The results showed that certain models achieve greater accuracy than others. The study also explores the seeming paradox of the accuracy of forecasting. On average, the accuracy of regression equations expressing level, trend, seasonality, and cycle is very high. Surprisingly, the study shows that there is only a small difference in accuracy between the more complex and the more straightforward models.

Keywords: forecasting; time series; forecasting accuracy

0. Introduction

The ultimate test of any forecasting method is whether or not it is capable of predicting future events accurately. Business and decision makers have a wide range of forecasts for any type of event, from the next quarter's sales to the long-term economic climate. The accuracy of these forecasts is crucial for effective decision making.
Makridakis and Hibon (1979)

This was the first large-scale empirical evaluation of time series forecasting methods.

- Highly controversial at the time.

- Difficulties:
  - How to measure forecast accuracy?
  - How to apply methods consistently and objectively?
  - How to explain unexpected results?

- Common thinking was that the more sophisticated mathematical models (ARIMA models at the time) were necessarily better.
- If results showed ARIMA models not best, it must be because analyst was unskilled.
As a result of this paper, researchers started to:

- consider how to automate forecasting methods;
- study what methods give the best forecasts;
- be aware of the dangers of over-fitting;
- treat forecasting as a different problem from time series analysis.

Makridakis & Hibon followed up with a new competition in 1982:

- 1001 series
- Anyone could submit forecasts (avoiding the charge of incompetence)
- Multiple forecast measures used.
As a result of this paper, researchers started to:

- consider how to automate forecasting methods;
- study what methods give the best forecasts;
- be aware of the dangers of over-fitting;
- treat forecasting as a different problem from time series analysis.

Makridakis & Hibon followed up with a new competition in 1982:

- 1001 series
- Anyone could submit forecasts (avoiding the charge of incompetence)
- Multiple forecast measures used.
The Accuracy of Extrapolation (Time Series) Methods: Results of a Forecasting Competition

S. MAKRIDAKIS  
inSEAD, Fontainebleau, France

A. ANDERSEN  
University of Sydney, Australia

R. CARBONE  
Université Laval, Quebec, Canada

R. FILDES  
Manchester Business School, Manchester, England

M. HIBON  
inSEAD, Fontainebleau, France

R. LEWANDOWSKI  
Marketing Systems, Essen, Germany

J. NEWTON  
E. PARZEN  
Texas A & M University, Texas, U.S.A.

R. WINKLER  
Indiana University, Bloomington, U.S.A.

ABSTRACT
In the last few decades many methods have become available for forecasting. As always, when alternatives exist, choices need to be made so that an appropriate forecasting method can be selected and used for the specific situation being considered. This paper reports the results of a forecasting competition that provides information to facilitate such choice. Seven experts in each of the 24 methods forecasted up to 1001 series for six up to eighteen time horizons. The results of the competition are presented in this paper whose purpose is to provide empirical evidence about differences found to exist among the various extrapolative (time series) methods used in the competition.
Main findings

1. Statistically sophisticated or complex methods do not necessarily provide more accurate forecasts than simpler ones.

2. The relative ranking of the performance of the various methods varies according to the accuracy measure being used.

3. The accuracy when various methods are being combined outperforms, on average, the individual methods being combined and does very well in comparison to other methods.

4. The accuracy of the various methods depends upon the length of the forecasting horizon involved.
The M3-Competition: results, conclusions and implications

Spyros Makridakis, Michèle Hibon*

INSEAD, Boulevard de Constance, 77305 Fontainebleau, France

Abstract

This paper describes the M3-Competition, the latest of the M-Competitions. It explains the reasons for conducting the competition and summarizes its results and conclusions. In addition, the paper compares such results/conclusions with those of the previous two M-Competitions as well as with those of other major empirical studies. Finally, the implications of these results and conclusions are considered, their consequences for both the theory and practice of forecasting are explored and directions for future research are contemplated. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Comparative methods — time series: univariate; Forecasting competitions; M-Competition; Forecasting methods, Forecasting accuracy
“The M3-Competition is a final attempt by the authors to settle the accuracy issue of various time series methods... The extension involves the inclusion of more methods/researchers (in particular in the areas of neural networks and expert systems) and more series.”

- 3003 series
- All data from business, demography, finance and economics.
- Series length between 14 and 126.
- Either non-seasonal, monthly or quarterly.
- All time series positive.
- M&H claimed that the M3-competition supported the findings of their earlier work.
- However, best performing methods far from “simple”.

Automatic algorithms for time series forecasting  
Forecasting competitions  
15
Makridakis and Hibon (2000)

Best methods:

<table>
<thead>
<tr>
<th>Theta</th>
</tr>
</thead>
<tbody>
<tr>
<td>A very confusing explanation.</td>
</tr>
<tr>
<td>Shown by Hyndman and Billah (2003) to be average of linear regression and simple exponential smoothing with drift, applied to seasonally adjusted data.</td>
</tr>
<tr>
<td>Later, the original authors claimed that their explanation was incorrect.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Forecast Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>A commercial software package with an unknown algorithm.</td>
</tr>
<tr>
<td>Known to fit either exponential smoothing or ARIMA models using BIC.</td>
</tr>
<tr>
<td>Method</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>Theta</td>
</tr>
<tr>
<td>ForecastPro</td>
</tr>
<tr>
<td>ForecastX</td>
</tr>
<tr>
<td>Automatic ANN</td>
</tr>
<tr>
<td>B-J automatic</td>
</tr>
</tbody>
</table>
### M3 results (recalculated)

<table>
<thead>
<tr>
<th>Method</th>
<th>MAPE</th>
<th>sMAPE</th>
<th>MASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theta</td>
<td>17.42</td>
<td>12.76</td>
<td>1.39</td>
</tr>
<tr>
<td>ForecastPro</td>
<td>18.00</td>
<td>13.06</td>
<td>1.47</td>
</tr>
<tr>
<td>ForecastX</td>
<td>17.35</td>
<td>13.09</td>
<td>1.42</td>
</tr>
</tbody>
</table>

- Calculations do not match published paper.
- Some contestants apparently submitted multiple entries but only best ones published.
Outline

1. Forecasting competitions
2. Exponential smoothing
3. ARIMA modelling
4. Automatic nonlinear forecasting?
5. Time series with complex seasonality
6. Recent developments
# 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>A,A</td>
<td>A,M</td>
</tr>
<tr>
<td>A_d (Additive damped)</td>
<td></td>
<td>A_d,N</td>
<td>A_d,A</td>
<td>A_d,M</td>
</tr>
<tr>
<td>M (Multiplicativd)</td>
<td></td>
<td>M,N</td>
<td>M,A</td>
<td>M,M</td>
</tr>
<tr>
<td>M_d (Multiplicativd damped)</td>
<td></td>
<td>M_d,N</td>
<td>M_d,A</td>
<td>M_d,M</td>
</tr>
</tbody>
</table>
### Seasonal 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>

**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&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
# Exponential smoothing methods

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
</tr>
<tr>
<td>N</td>
<td>N,N</td>
</tr>
<tr>
<td>A</td>
<td>A,N</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt;</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
<tr>
<td>M</td>
<td>M,N</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt;</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<sub>d</sub>,N: Additive damped trend 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</td>
</tr>
<tr>
<td>A (Additive)</td>
<td>N,A</td>
</tr>
<tr>
<td>Ad (Additive damped)</td>
<td>N,Ad</td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
</tr>
<tr>
<td>Md (Multiplicative damped)</td>
<td>M,Ad</td>
</tr>
</tbody>
</table>

- **N,N**: Simple exponential smoothing
- **A,N**: Holt’s linear method
- **Ad,N**: Additive damped trend method
- **M,N**: Exponential trend method

### Automatic algorithms for time series forecasting

| Exponential smoothing | 19 |
# Exponential smoothing methods

## Trend Component

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

- **N,N**: Simple exponential smoothing
- **A,N**: Holt’s linear method
- **A_d,N**: Additive damped trend method
- **M,N**: Exponential trend method
- **M_d,N**: Multiplicative damped trend 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>N,N</td>
<td>N,A</td>
<td>N,M</td>
<td></td>
</tr>
<tr>
<td>A (Additive)</td>
<td>A,N</td>
<td>A,A</td>
<td>A,M</td>
<td></td>
</tr>
<tr>
<td>Ad (Additive damped)</td>
<td>Ad,N</td>
<td>Ad,A</td>
<td>Ad,M</td>
<td></td>
</tr>
<tr>
<td>M (Multiplicative)</td>
<td>M,N</td>
<td>M,A</td>
<td>M,M</td>
<td></td>
</tr>
<tr>
<td>Md (Multiplicative damped)</td>
<td>Md,N</td>
<td>Md,A</td>
<td>Md,M</td>
<td></td>
</tr>
</tbody>
</table>

- N,N: Simple exponential smoothing
- A,N: Holt’s linear method
- Ad,N: Additive damped trend method
- M,N: Exponential trend method
- Md,N: Multiplicative damped trend method
- A,A: Additive Holt-Winters’ method
## Exponential smoothing methods

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

- **N,N**: Simple exponential smoothing
- **A,N**: Holt’s linear method
- **A_d,N**: Additive damped trend method
- **M,N**: Exponential trend method
- **M_d,N**: Multiplicative damped trend method
- **A,A**: Additive Holt-Winters’ method
- **A,M**: Multiplicative 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_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>

There are 15 separate exp. smoothing methods.
There are 15 separate exp. smoothing methods. Each can have an additive or multiplicative error, giving 30 separate models.
### Exponential smoothing methods

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

- There are 15 separate exp. smoothing methods.
- Each can have an additive or multiplicative error, giving 30 separate models.
- Only 19 models are numerically stable.
### Exponential smoothing methods

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

- There are 15 separate exp. smoothing methods.
- Each can have an additive or multiplicative error, giving 30 separate models.
- Only 19 models are numerically stable.
- Multiplicative trend models give poor forecasts leaving 15 models.
### 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</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>
<tr>
<td></td>
<td>A, A</td>
</tr>
<tr>
<td></td>
<td>A&lt;sub&gt;d&lt;/sub&gt;, A</td>
</tr>
<tr>
<td></td>
<td>M, A</td>
</tr>
<tr>
<td></td>
<td>M&lt;sub&gt;d&lt;/sub&gt;, A</td>
</tr>
<tr>
<td></td>
<td>N, M</td>
</tr>
<tr>
<td></td>
<td>A, M</td>
</tr>
<tr>
<td></td>
<td>A&lt;sub&gt;d&lt;/sub&gt;, M</td>
</tr>
<tr>
<td></td>
<td>M, M</td>
</tr>
<tr>
<td></td>
<td>M&lt;sub&gt;d&lt;/sub&gt;, M</td>
</tr>
</tbody>
</table>

**General notation**

ETS : Exponential Smoothing

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

**Automatic algorithms for time series forecasting**

Exponential smoothing
# Exponential smoothing methods

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

**General notation**  
ETS : **Exponential Smoothing**

**Examples:**  
A,N,N: Simple exponential smoothing with additive errors  
A,N,A: Holt’s linear method with additive errors  
N,M,N: Multiplicative Holt-Winters’ method with multiplicative errors
### Exponential smoothing methods

#### Trend Component

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

#### Seasonal Component

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

<table>
<thead>
<tr>
<th>Trend Component</th>
<th>Seasonal Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>N,N</td>
</tr>
<tr>
<td>A</td>
<td>A,N</td>
</tr>
<tr>
<td>A&lt;sub&gt;d&lt;/sub&gt;</td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
<tr>
<td>M</td>
<td>M,N</td>
</tr>
<tr>
<td>M&lt;sub&gt;d&lt;/sub&gt;</td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,N</td>
</tr>
<tr>
<td></td>
<td>N,A</td>
</tr>
<tr>
<td></td>
<td>A,A</td>
</tr>
<tr>
<td></td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,A</td>
</tr>
<tr>
<td></td>
<td>M,A</td>
</tr>
<tr>
<td></td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,A</td>
</tr>
<tr>
<td></td>
<td>N,M</td>
</tr>
<tr>
<td></td>
<td>A,M</td>
</tr>
<tr>
<td></td>
<td>A&lt;sub&gt;d&lt;/sub&gt;,M</td>
</tr>
<tr>
<td></td>
<td>M,M</td>
</tr>
<tr>
<td></td>
<td>M&lt;sub&gt;d&lt;/sub&gt;,M</td>
</tr>
</tbody>
</table>

#### General notation

**ETS** : **Exponential Smoothing**

**Trend**

#### 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

---

**Automatic algorithms for time series forecasting**

**Exponential smoothing**
## 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>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>

### Seasonal Component

<table>
<thead>
<tr>
<th></th>
<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>
<td></td>
</tr>
<tr>
<td>A,N</td>
<td>A,A</td>
<td>A,M</td>
<td></td>
</tr>
<tr>
<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>
<td></td>
</tr>
<tr>
<td>M,N</td>
<td>M,A</td>
<td>M,M</td>
<td></td>
</tr>
<tr>
<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>
<td></td>
</tr>
</tbody>
</table>

### General notation

ETS : Exponential Smoothing

↑ ↖

**Trend Seasonal**

### 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
## Exponential smoothing methods

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

### General notation

\[ ETS : \text{Exponential Smoothing} \]

#### Error Trend Seasonal

### 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
### Exponential smoothing methods

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

**General notation**: \( \text{ETS} : \text{Exponential Smoothing} \)

**Error** **Trend** **Seasonal**

### 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
Exponential smoothing methods

Innovations state space models

- All ETS models can be written in innovations state space form (IJF, 2002).
- Additive and multiplicative versions give the same point forecasts but different prediction intervals.

General notation

ETS : Exponential Smoothing

Error Trend Seasonal

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
ETS state space model

\[ x_{t-1} \rightarrow y_t \]

\[ \varepsilon_t \]

State space model

\[ x_t = (\text{level, slope, seasonal}) \]
ETS state space model

\[ x_t = (\text{level, slope, seasonal}) \]
ETS state space model

$x_{t-1}$ → $y_t$ → $x_t$ → $y_{t+1}$

State space model

$x_t = (\text{level, slope, seasonal})$
ETS state space model

\[ x_t = (\text{level}, \text{slope}, \text{seasonal}) \]
ETS state space model

\[ x_t = (\text{level, slope, seasonal}) \]
ETS state space model

\[ x_t = (\text{level}, \text{slope}, \text{seasonal}) \]
ETS state space model

State space model

\[ x_t = (\text{level, slope, seasonal}) \]
ETS state space model

\[ x_t = (\text{level}, \text{slope}, \text{seasonal}) \]

Automatic algorithms for time series forecasting
Exponential smoothing

21
ETS state space model

\[ \mathbf{x}_t = (\text{level}, \text{slope}, \text{seasonal}) \]
ETS state space model

\[ x_{t-1} \rightarrow y_t \rightarrow x_t \rightarrow y_{t+1} \rightarrow x_{t+1} \rightarrow y_{t+2} \rightarrow x_{t+2} \rightarrow y_{t+3} \rightarrow x_{t+3} \rightarrow y_{t+4} \]

State space model
\[ x_t = (\text{level, slope, seasonal}) \]

Estimation
Compute likelihood \( L \) from \( \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_T \).
Optimize \( L \) wrt model parameters.
ETS state space model

\[ x_t = (\text{level, slope, seasonal}) \]

**Estimation**

Compute likelihood \( L \) from \( \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_T \).
Optimize \( L \) wrt model parameters.

**Q: How to choose between the 15 useful ETS models?**
Cross-validation

Traditional evaluation

Training data

Test data

time
Cross-validation

Traditional evaluation

- Training data
- Test data

Standard cross-validation

- Time series data

Automatic algorithms for time series forecasting

Exponential smoothing

22
Cross-validation

Traditional evaluation

- Training data
- Test data

Standard cross-validation

Time series cross-validation

Automatic algorithms for time series forecasting

Exponential smoothing
Cross-validation

Traditional evaluation

Standard cross-validation

Time series cross-validation

Automatic algorithms for time series forecasting
Exponential smoothing
Cross-validation

Traditional evaluation

Standard cross-validation

Time series cross-validation

Also known as “Evaluation on a rolling forecast origin”
Akaike’s Information Criterion

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

where \( L \) is the likelihood and \( k \) is the number of estimated parameters in the model.

- This is a penalized likelihood approach.
- If \( L \) is Gaussian, then \( \text{AIC} \approx c + T \log \text{MSE} + 2k \)
  where \( c \) is a constant, \( \text{MSE} \) is from one-step forecasts on \text{training set}, and \( T \) is the length of the series.

Minimizing the Gaussian AIC is asymptotically equivalent (as \( T \to \infty \)) to minimizing \( \text{MSE} \) from one-step forecasts on \text{test set} via time series cross-validation.
Akaike’s Information Criterion

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

where \( L \) is the likelihood and \( k \) is the number of estimated parameters in the model.

- This is a \textit{penalized likelihood} approach.
- If \( L \) is Gaussian, then \( \text{AIC} \approx c + T \log \text{MSE} + 2k \)

where \( c \) is a constant, MSE is from one-step forecasts on \textit{training set}, and \( T \) is the length of the series.

Minimizing the Gaussian AIC is asymptotically equivalent (as \( T \to \infty \)) to minimizing MSE from one-step forecasts on \textit{test set} via time series cross-validation.
Akaike’s Information Criterion

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

where \( L \) is the likelihood and \( k \) is the number of estimated parameters in the model.

- This is a *penalized likelihood* approach.
- If \( L \) is Gaussian, then \( \text{AIC} \approx c + T \log \text{MSE} + 2k \)
  where \( c \) is a constant, \( \text{MSE} \) is from one-step forecasts on *training set*, and \( T \) is the length of the series.

Minimizing the Gaussian AIC is asymptotically equivalent (as \( T \to \infty \)) to minimizing MSE from one-step forecasts on *test set* via time series cross-validation.
Akaike’s Information Criterion

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

where \( L \) is the likelihood and \( k \) is the number of estimated parameters in the model.

- This is a *penalized likelihood* approach.
- If \( L \) is Gaussian, then \( \text{AIC} \approx c + T \log \text{MSE} + 2k \)

where \( c \) is a constant, \( \text{MSE} \) is from one-step forecasts on *training set*, and \( T \) is the length of the series.

Minimizing the Gaussian AIC is asymptotically equivalent (as \( T \to \infty \)) to minimizing MSE from one-step forecasts on *test set* via time series cross-validation.
Akaike’s Information Criterion

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

where \( L \) is the likelihood and \( k \) is the number of estimated parameters in the model.

- This is a penalized likelihood approach.
- If \( L \) is Gaussian, then \( \text{AIC} \approx c + T \log \text{MSE} + 2k \)

where \( c \) is a constant, MSE is from one-step forecasts on training set, and \( T \) is the length of the series.

Minimizing the Gaussian AIC is asymptotically equivalent (as \( T \to \infty \)) to minimizing MSE from one-step forecasts on test set via time series cross-validation.

Automatic algorithms for time series forecasting

Exponential smoothing 23
Akaike’s Information Criterion

\[ AIC = -2 \log(L) + 2k \]

**Corrected AIC**

For small \( T \), AIC tends to over-fit. Bias-corrected version:

\[ AIC_c = AIC + \frac{2(k+1)(k+2)}{T-k} \]

- CV-MSE too time consuming for most automatic forecasting purposes. Also requires large \( T \).
- \( AIC_c \) asymptotically equivalent, can be used on small samples and is very fast to compute.
Akaike’s Information Criterion

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

Corrected AIC
For small \( T \), AIC tends to over-fit. Bias-corrected version:

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

- CV-MSE too time consuming for most automatic forecasting purposes. Also requires large \( T \).
- AICc asymptotically equivalent, can be used on small samples and is very fast to compute.
Akaike’s Information Criterion

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

**Corrected AIC**

For small \( T \), AIC tends to over-fit. Bias-corrected version:

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

- CV-MSE too time consuming for most automatic forecasting purposes. Also requires large \( T \).
- \( \text{AIC}_c \) asymptotically equivalent, can be used on small samples and is very fast to compute.
Based on Hyndman, Koehler, Snyder & Grose (IJF 2002):

- Apply each of 15 models that are appropriate to the data. Optimize parameters and initial values using MLE.
- Select best method using AICc.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.
Based on Hyndman, Koehler, Snyder & Grose (IJF 2002):

- Apply each of 15 models that are appropriate to the data. Optimize parameters and initial values using MLE.
- Select best method using AICc.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.
Based on Hyndman, Koehler, Snyder & Grose (IJF 2002):

- Apply each of 15 models that are appropriate to the data. Optimize parameters and initial values using MLE.
- Select best method using AICc.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.
ets algorithm in R

Based on Hyndman, Koehler, Snyder & Grose (IJF 2002):

- Apply each of 15 models that are appropriate to the data. Optimize parameters and initial values using MLE.
- Select best method using AICc.
- Produce forecasts using best method.
- Obtain prediction intervals using underlying state space model.
Exponential smoothing

Forecasts from ETS(M,A,N)

<table>
<thead>
<tr>
<th>Year</th>
<th>millions of sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>1960</td>
<td></td>
</tr>
<tr>
<td>1970</td>
<td></td>
</tr>
<tr>
<td>1980</td>
<td></td>
</tr>
<tr>
<td>1990</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td></td>
</tr>
</tbody>
</table>
Exponential smoothing

fit <- ets(livestock)
fcast <- forecast(fit)
plot(fcast)
Exponential smoothing

Forecasts from ETS(M,N,M)

Year

Total scripts (millions)

1995 2000 2005 2010

0.4 0.6 0.8 1.0 1.2 1.4 1.6

Automatic algorithms for time series forecasting
Exponential smoothing
Exponential smoothing

```
fit <- ets(h02)
fcast <- forecast(fit)
plot(fcast)
```
> fit
ETS(M,N,M)

Smoothing parameters:
  alpha = 0.4597
  gamma = 1e-04

Initial states:
  l  = 0.4501
  s  = 0.8628  0.8193  0.7648  0.7675  0.6946  1.2921
       1.3327  1.1833  1.1617  1.0899  1.0377  0.9937

sigma:  0.0675

AIC    AICc    BIC
-115.69960  -113.47738   -69.24592
## M3 comparisons

<table>
<thead>
<tr>
<th>Method</th>
<th>MAPE</th>
<th>sMAPE</th>
<th>MASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theta</td>
<td>17.42</td>
<td>12.76</td>
<td>1.39</td>
</tr>
<tr>
<td>ForecastPro</td>
<td>18.00</td>
<td>13.06</td>
<td>1.47</td>
</tr>
<tr>
<td>ForecastX</td>
<td>17.35</td>
<td>13.09</td>
<td>1.42</td>
</tr>
<tr>
<td>Automatic ANN</td>
<td>17.18</td>
<td>13.98</td>
<td>1.53</td>
</tr>
<tr>
<td>B-J automatic</td>
<td>19.13</td>
<td>13.72</td>
<td>1.54</td>
</tr>
<tr>
<td>ETS</td>
<td>17.38</td>
<td>13.13</td>
<td>1.43</td>
</tr>
</tbody>
</table>

**Automatic algorithms for time series forecasting**
7 Exponential smoothing

Exponential smoothing was proposed in the late 1950s (Brown 1959, Holt 1957 and Winters 1960 are key pioneering works) and has motivated some of the most successful forecasting methods. Forecasts produced using exponential smoothing methods are weighted averages of past observations, with the weights decaying exponentially as the observations get older. In other words, the more recent the observation the higher the associated weight. This framework generates reliable forecasts quickly and for a wide spectrum of time series which is a great advantage and of major importance to applications in industry.

This chapter is divided into two parts. In the first part we present in detail the mechanics of all exponential smoothing methods and their application in forecasting time series with various characteristics. This is key in understanding the intuition behind these methods. In this setting, selecting and using a forecasting method may appear to be somewhat ad-hoc. The
7 Exponential smoothing

Exponential smoothing was proposed in the late 1950s (Brown 1959, Holt 1957 and Winters 1960 are key pioneering works) and has motivated some of the most successful forecasting methods. Forecasts produced using exponential smoothing methods are weighted averages of past observations, with the more recent observations being given higher weights. In addition, the method has the flexibility to respond rapidly to changes in the underlying time series. The framework generates reliable forecasts quickly and is applicable to a wide spectrum of time series which is a great advantage and of major importance to applications in industry.

This chapter is divided into two parts. In the first part we present in detail the mechanics of all exponential smoothing methods and their application in forecasting time series with various characteristics. This is key in understanding the intuition behind these methods. In this setting, selecting and using a forecasting method may appear to be somewhat ad-hoc. The second part is more applications oriented.
Exponential smoothing was proposed in the late 1950s (Brown 1959 and Winters 1960 are key pioneering works) and has motivated a large number of successful forecasting methods. Forecasts produced using exponential smoothing methods are weighted averages of past observations, with the weights decaying exponentially as the observations get older. In other words, the more recent the observation the higher the associated weight. This framework generates reliable forecasts quickly and for a wide span of time series which is a great advantage and of major importance in industry.

This chapter is divided into two parts. In the first part we present the mechanics of all exponential smoothing methods and their application to forecasting time series with various characteristics. This is key in understanding the intuition behind these methods. In this setting we avoid and using a forecasting method may appear to be somewhat ad-hoc.
Exercise

1. Use `ets` to find the best ETS models for the following series: `ibmclose`, `eggs`, `bricksq`, `hsales`.

2. Try `ets` with `cangas` and `lynx`. What do you learn?

3. Can you find another series for which `ets` gives bad forecasts?
1 Forecasting competitions
2 Exponential smoothing
3 ARIMA modelling
4 Automatic nonlinear forecasting?
5 Time series with complex seasonality
6 Recent developments
ARIMA models

Inputs

$y_{t-1}$

$y_{t-2}$

$y_{t-3}$

Output

$y_t$
ARIMA models

Inputs

\( y_{t-1} \)
\( y_{t-2} \)
\( y_{t-3} \)
\( \varepsilon_t \)

Output

\( y_t \)

Autoregression (AR) model
ARIMA models

Inputs

$y_{t-1}$

$y_{t-2}$

$y_{t-3}$

Outputs

$y_t$

$\epsilon_t$

$\epsilon_{t-1}$

$\epsilon_{t-2}$

Autoregression moving average (ARMA) model
ARIMA models

Inputs

$y_{t-1}$

$y_{t-2}$

$y_{t-3}$

$\varepsilon_t$

$\varepsilon_{t-1}$

$\varepsilon_{t-2}$

Output

$y_t$

Autoregression moving average (ARMA) model

Estimation

Compute likelihood $L$ from $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_T$.

Use optimization algorithm to maximize $L$. 

Automatic algorithms for time series forecasting

ARIMA modelling

35
**ARIMA models**

**Inputs**
- $y_{t-1}$
- $y_{t-2}$
- $y_{t-3}$
- $\varepsilon_t$
- $\varepsilon_{t-1}$
- $\varepsilon_{t-2}$

**Output**
- $y_t$

**ARIMA model**
Autoregression moving average (ARMA) model applied to differences.

**Estimation**
- Compute likelihood $L$ from $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_T$.
- Use optimization algorithm to maximize $L$. 

Automatic algorithms for time series forecasting

ARIMA modelling 35
Automatic Time Series Forecasting: The forecast Package for R

Rob J. Hyndman
Monash University

Yeasmin Khandakar
Monash University

Abstract

Automatic forecasts of large numbers of univariate time series are often needed in business and government. We describe a package for automatic forecasting with a large range of models and algorithms.
Forecasts from ARIMA(0,1,0) with drift

Year
millions of sheep
250 300 350 400 450 500 550
Auto ARIMA

fit <- auto.arima(livestock)
fcast <- forecast(fit)
plot(fcast)
Auto ARIMA

Forecasts from ARIMA(3,1,3)(0,1,1)[12]

Year
Total scripts (millions)
1995 2000 2005 2010
0.4 0.6 0.8 1.0 1.2 1.4
fit <- auto.arima(h02)
fcast <- forecast(fit)
plot(fcast)
Auto ARIMA

> fit
Series: h02
ARIMA(3,1,3)(0,1,1)[12]

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>ar2</th>
<th>ar3</th>
<th>ma1</th>
<th>ma2</th>
<th>ma3</th>
<th>sma1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-0.3648</td>
<td>-0.0636</td>
<td>0.3568</td>
<td>-0.4850</td>
<td>0.0479</td>
<td>-0.353</td>
<td>-0.5931</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.2198</td>
<td>0.3293</td>
<td>0.1268</td>
<td>0.2227</td>
<td>0.2755</td>
<td>0.212</td>
<td>0.0651</td>
</tr>
</tbody>
</table>

sigma^2 estimated as 0.002706: log likelihood=290.25
AIC=-564.5  AICc=-563.71  BIC=-538.48
How does auto.arima() work?

A non-seasonal ARIMA process

\[
\phi(B)(1 - B)^d y_t = c + \theta(B) \varepsilon_t
\]

Need to select appropriate orders \( p, q, d \), and whether to include \( c \).

Algorithm choices driven by forecast accuracy.
How does auto.arima() work?

A non-seasonal ARIMA process

\[ \phi(B)(1 - B)^d y_t = c + \theta(B) \varepsilon_t \]

Need to select appropriate orders \( p, q, d \), and whether to include \( c \).

Hyndman & Khandakar (JSS, 2008) algorithm:

- Select no. differences \( d \) via KPSS unit root test.
- Select \( p, q, c \) by minimising AICc.
- Use stepwise search to traverse model space, starting with a simple model and considering nearby variants.

Algorithm choices driven by forecast accuracy.
How does auto.arima() work?

A non-seasonal ARIMA process

\[ \phi(B)(1 - B)^d y_t = c + \theta(B) \varepsilon_t \]

Need to select appropriate orders \( p, q, d \), and whether to include \( c \).

Hyndman & Khandakar (JSS, 2008) algorithm:
- Select no. differences \( d \) via KPSS unit root test.
- Select \( p, q, c \) by minimising AICc.
- Use stepwise search to traverse model space, starting with a simple model and considering nearby variants.

Algorithm choices driven by forecast accuracy.
How does auto.arima() work?

A seasonal ARIMA process

$$\Phi(B^m) \phi(B) (1 - B)^d (1 - B^m)^D y_t = c + \Theta(B^m) \theta(B) \epsilon_t$$

Need to select appropriate orders $p, q, d, P, Q, D$, and whether to include $c$.

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

- Select no. differences $d$ via KPSS unit root test.
- Select $D$ using OCSB unit root test.
- Select $p, q, P, Q, c$ by minimising AICc.
- Use stepwise search to traverse model space, starting with a simple model and considering nearby variants.
## M3 comparisons

<table>
<thead>
<tr>
<th>Method</th>
<th>MAPE</th>
<th>sMAPE</th>
<th>MASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theta</td>
<td>17.42</td>
<td>12.76</td>
<td>1.39</td>
</tr>
<tr>
<td>ForecastPro</td>
<td>18.00</td>
<td>13.06</td>
<td>1.47</td>
</tr>
<tr>
<td>B-J automatic</td>
<td>19.13</td>
<td>13.72</td>
<td>1.54</td>
</tr>
<tr>
<td>ETS</td>
<td>17.38</td>
<td>13.13</td>
<td>1.43</td>
</tr>
<tr>
<td>AutoARIMA</td>
<td>19.12</td>
<td>13.85</td>
<td>1.47</td>
</tr>
</tbody>
</table>
Exercise

1. Use `auto.arima` to find the best ARIMA models for the following series: `ibmclose, eggs, bricksq, hsales`.

2. Try `auto.arima` with `cangas` and `lynx`. What do you learn?

3. Can you find a series for which `auto.arima` gives bad forecasts?

4. How would you compare the ETS and ARIMA results?
Outline

1. Forecasting competitions
2. Exponential smoothing
3. ARIMA modelling
4. **Automatic nonlinear forecasting?**
5. Time series with complex seasonality
6. Recent developments
Automatic ANN in M3 competition did poorly.

Linear methods did best in the NN3 competition!

Very few machine learning methods get published in the IJF because authors cannot demonstrate their methods give better forecasts than linear benchmark methods, even on supposedly nonlinear data.

Some good recent work by Kourentzes and Crone on automated ANN for time series.

Watch this space!
Automatic ANN in M3 competition did poorly.

Linear methods did best in the NN3 competition!

Very few machine learning methods get published in the IJF because authors cannot demonstrate their methods give better forecasts than linear benchmark methods, even on supposedly nonlinear data.

Some good recent work by Kourentzes and Crone on automated ANN for time series.

Watch this space!
Automatic ANN in M3 competition did poorly.

Linear methods did best in the NN3 competition!

Very few machine learning methods get published in the IJF because authors cannot demonstrate their methods give better forecasts than linear benchmark methods, even on supposedly nonlinear data.

Some good recent work by Kourentzes and Crone on automated ANN for time series.

Watch this space!
Automatic nonlinear forecasting

- Automatic ANN in M3 competition did poorly.
- Linear methods did best in the NN3 competition!
- Very few machine learning methods get published in the IJF because authors cannot demonstrate their methods give better forecasts than linear benchmark methods, even on supposedly nonlinear data.
- Some good recent work by Kourentzes and Crone on automated ANN for time series.
- Watch this space!
Automatic nonlinear forecasting

- Automatic ANN in M3 competition did poorly.
- Linear methods did best in the NN3 competition!
- Very few machine learning methods get published in the IJF because authors cannot demonstrate their methods give better forecasts than linear benchmark methods, even on supposedly nonlinear data.
- Some good recent work by Kourentzses and Crone on automated ANN for time series.

Watch this space!
Outline

1. Forecasting competitions
2. Exponential smoothing
3. ARIMA modelling
4. Automatic nonlinear forecasting?
5. Time series with complex seasonality
6. Recent developments
Examples

US finished motor gasoline products

Thousands of barrels per day

Weeks


6500 7000 7500 8000 8500 9000 9500
Examples

Number of calls to large American bank (7am–9pm)

Number of call arrivals

3 March 17 March 31 March 14 April 28 April 12 May

5 minute intervals
Automatic algorithms for time series forecasting

Time series with complex seasonality

Examples

Turkish electricity demand

Electricity demand (GW)

Days


10 15 20 25
TBATS model

TBATS

Trigonometric terms for seasonality
Box-Cox transformations for heterogeneity
ARMA errors for short-term dynamics
Trend (possibly damped)
Seasonal (including multiple and non-integer periods)

**TBATS model**

\[ y_t = \text{observation at time } t \]

\[
y_t^{(\omega)} = \begin{cases} 
(y_t^{\omega} - 1)/\omega & \text{if } \omega \neq 0; \\
\log y_t & \text{if } \omega = 0.
\end{cases}
\]

\[
y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s_{t-m_i}^{(i)} + d_t
\]

\[
\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t
\]

\[
b_t = (1 - \phi) b + \phi b_{t-1} + \beta d_t
\]

\[
d_t = \sum_{i=1}^{p} \phi_i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t
\]

\[
s_t^{(i)} = \sum_{j=1}^{k_i} s_{j,t}^{(i)}
\]

\[
s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{* (i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t
\]

\[
s_{j,t} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{* (i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t
\]
TBATS model

\[ y_t = \text{observation at time } t \]

\[ y_t^{(\omega)} = \begin{cases} 
\frac{(y_t^\omega - 1)}{\omega} & \text{if } \omega \neq 0; \\
\log y_t & \text{if } \omega = 0.
\end{cases} \]

\[ y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s_{t-m_i}^{(i)} + d_t \]

\[ \ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t \]

\[ b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t \]

\[ d_t = \sum_{i=1}^{p} \phi_i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

\[ s_t^{(i)} = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \quad s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \]

\[ s_{j,t} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t \]
### TBATS model

\[ y_t = \text{observation at time } t \]

\[ y_t^{(\omega)} = \begin{cases} 
(y_t^{\omega} - 1)/\omega & \text{if } \omega \neq 0; \\
\log y_t & \text{if } \omega = 0.
\end{cases} \]

\[ y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s^{(i)}_{t-m_i} + d_t \]

\[ \ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t \]

\[ b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t \]

\[ d_t = \sum_{i=1}^{p} \phi_i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

\[ s^{(i)}_{t} = \sum_{j=1}^{k_i} s^{(i)}_{j,t} \]

\[ s^{(i)}_{j,t} = \begin{cases} 
\cos \lambda_{j}^{(i)} + s^{*}_{j,t-1} \sin \lambda_{j}^{(i)} + \gamma_1^{(i)} d_t & \text{if } \omega \neq 0; \\
-s^{(i)}_{j,t-1} \sin \lambda_{j}^{(i)} + s^{*}_{j,t-1} \cos \lambda_{j}^{(i)} + \gamma_2^{(i)} d_t
\end{cases} \]

- **Box-Cox transformation**
- **M seasonal periods**
TBATS model

\( y_t \) = observation at time \( t \)

\[ y_t^{(\omega)} = \begin{cases} \frac{(y_t^\omega - 1)}{\omega} & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases} \]

\[ y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s_{t-m_i}^{(i)} + d_t \]

\( \ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t \)

\( b_t = (1 - \phi) b + \phi b_{t-1} + \beta d_t \)

\[ d_t = \sum_{i=1}^{p} \phi_i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

\[ s_t^{(i)} = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \]

\[ s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \]

\[ s_{j,t}^{(i)} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t \]

Automatic algorithms for time series forecasting

Time series with complex seasonality

Box-Cox transformation

\( M \) seasonal periods

global and local trend
TBATS model

\[ y_t = \text{observation at time } t \]

\[ y_t^{(\omega)} = \begin{cases} 
\frac{(y_t^{\omega} - 1)}{\omega} & \text{if } \omega \neq 0; \\
\log y_t & \text{if } \omega = 0.
\end{cases} \]

\[ y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s_{t-m_i}^{(i)} + d_t \]

\[ \ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t \]

\[ b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t \]

\[ d_t = \sum_{i=1}^{p} \phi_i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

\[ s_t^{(i)} = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \]

Box-Cox transformation

\( M \) seasonal periods

global and local trend

ARMA error

Automatic algorithms for time series forecasting

Time series with complex seasonality
TBATS model

\[ y_t = \text{observation at time } t \]

\[ y_t^{(\omega)} = \begin{cases} \frac{(y_t^\omega - 1)}{\omega} & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases} \]

\[ y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s_{t-m_i}^{(i)} + d_t \]

\[ \ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t \]

\[ b_t = (1 - \phi) b + \phi b_{t-1} + \beta d_t \]

\[ d_t = \sum_{i=1}^{p} \phi i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

\[ s_t^{(i)} = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \]

\[ s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*^{(i)}} \sin \lambda_j^{(i)} + \gamma_{2}^{(i)} d_t \]
TBATS model

\[ y_t = \text{observation at time } t \]

\[ y_t(\omega) = \begin{cases} 
(y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\
\log & \text{if } \omega = 0 
\end{cases} \]

\[ y_t(\omega) = \ell_t - 1 + \phi b_t - 1 + M \sum_{i=1}^{s(i)} s(i) t - m_i + d_t \]

\[ \ell_t = \ell_{t-1} \]

\[ b_t = (1 - \phi) b_t \]

\[ d_t = P \sum_{i=1}^{k_i} s(i) \]

\[ s_t(i) = \sum_{j=1}^{k_i} s(j,t) \]

\[ s_{j,t} = s_{j,t-1} \cos \lambda_j^{(i)} + s_{j,t-1}^{*} \sin \lambda_j^{(i)} + \gamma^{(i)}_2 d_t \]
Examples

fit <- tbats(gasoline)
fcast <- forecast(fit)
plot(fcast)
Examples

```r
fit <- tbats(callcentre)
fcast <- forecast(fit)
plot(fcast)
```

Forecasts from TBATS(1, {3,1}, 0.987, {<169,5>, <845,3>})

5 minute intervals

Number of call arrivals

3 March 17 March 31 March 14 April 28 April 12 May 26 May 9 June
Examples

```r
fit <- tbats(turk)
fcast <- forecast(fit)
plot(fcast)
```

Forecasts from TBATS\((0, \{5,3\}, 0.997, \{<7,3>, <354.37,12>, <365.25,4>\})\)

Electricity demand (GW)

10 15 20 25

Automatic algorithms for time series forecasting

Time series with complex seasonality
Outline

1. Forecasting competitions
2. Exponential smoothing
3. ARIMA modelling
4. Automatic nonlinear forecasting?
5. Time series with complex seasonality
6. Recent developments
Further competitions

1. 2011 tourism forecasting competition.
2. Kaggle and other forecasting platforms.
3. GEFCom 2012: Point forecasting of electricity load and wind power.
Further competitions

1. 2011 tourism forecasting competition.
2. Kaggle and other forecasting platforms.
3. GEFCom 2012: Point forecasting of electricity load and wind power.
Further competitions

1. 2011 tourism forecasting competition.
2. Kaggle and other forecasting platforms.
3. GEFCom 2012: Point forecasting of electricity load and wind power.
Further competitions

1. 2011 tourism forecasting competition.
2. Kaggle and other forecasting platforms.
3. GEFCom 2012: Point forecasting of electricity load and wind power.
Forecasts about forecasting

1. Automatic algorithms will become more general — handling a wide variety of time series.

2. Model selection methods will take account of multi-step forecast accuracy as well as one-step forecast accuracy.

3. Automatic forecasting algorithms for multivariate time series will be developed.

4. Automatic forecasting algorithms that include covariate information will be developed.
Forecasts about forecasting

1. Automatic algorithms will become more general — handling a wide variety of time series.

2. Model selection methods will take account of multi-step forecast accuracy as well as one-step forecast accuracy.

3. Automatic forecasting algorithms for multivariate time series will be developed.

4. Automatic forecasting algorithms that include covariate information will be developed.
Forecasts about forecasting

1. Automatic algorithms will become more general — handling a wide variety of time series.

2. Model selection methods will take account of multi-step forecast accuracy as well as one-step forecast accuracy.

3. Automatic forecasting algorithms for multivariate time series will be developed.

4. Automatic forecasting algorithms that include covariate information will be developed.
Forecasts about forecasting

1. Automatic algorithms will become more general — handling a wide variety of time series.

2. Model selection methods will take account of multi-step forecast accuracy as well as one-step forecast accuracy.

3. Automatic forecasting algorithms for multivariate time series will be developed.

4. Automatic forecasting algorithms that include covariate information will be developed.