Forecasting and Time Series Analysis

San Cannon

Rockhurst University

Week 4

Remember

Last week we discussed linear regression (single and multiple) and went over how to use it to capture trend and seasonality in data. Regression is used for both understandin time series components and forecasting.

Tonight we are going to go over methods for smoothing and transforming data to better improve forecasting. Some rely on understanding the components of the data.
Others will deal strictly with forecasting.

Exponential smoothing

We have seen that our simple forecasting models aren't particularly useful. They rely too heavily on a single measure (mean or most recent value) and don't adequately take into consideration other information.

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 but older information is still considered.

Let's look at some simple exponential smoothing (SES) models. These are appropriate only for data with no clear trend or seasonality. (How would you know?)

You've already seen this

Some of our simple forecasting models were special cases of weighted averages:

The naïve method assumes that the most current observation is the only important one and all previous observations provide no information for the future ($\hat{y}_{T+h|T} = y_T$, for $h = 1, 2 \ldots$). This can be thought of as a weighted average where all the weight is given to the last observation.

The mean method assumes that all observations are of equal importance and they are given equal weight when generating forecasts ($\hat{y}_{T+h|T} = \frac{1}{T}\sum_{t=1}^{T} y_t$, $h = 1, 2, \ldots$). Can we find a middle ground?

Basic SES model

Let's take a combination of the mean and the naive version:

$$\hat{y}_{t+1|t} = \alpha y_t + \alpha(1-\alpha)y_{t-1} + \alpha(1-\alpha)^2 y_{t-2} + \dots$$

Where α is the smoothing parameter - the larger the value for α, the faster the effect dies out.

Table with examples in your book: https://www.otexts.org/fpp/7/1

Weighted average

What about calculating some kind of average across one period ahead forecasts? What if we forecast tomorrow based on a combination of what we see today and what we thought today would be yesterday?

In math: $\hat{y}_{t+1|t} = \alpha y_t + (1-\alpha)\hat{y}_{t|t-1}$

So the forecast for a monthly series in October 2016 (16M10) would be a weighted average of the September value and what we thought the September value would be in October.:
$\hat{y}_{16M10|16M09} = \alpha y_{16M09} + (1-\alpha)\hat{y}_{16M09|16M08}$

Of course the forecast in for September that we made in August was a weighted average of the value we saw in August and what we thought the value for August would be in July:
$\hat{y}_{16M09|16M08} = \alpha y_{16M08} + (1-\alpha)\hat{y}_{16M08|16M07}$
etc.

But if you do all the algebra, you get the same equation as on the previous.

More details

The one-step-ahead forecast for time $T + 1$ is a weighted average of all the observations in the series $y_1 \dots y_T$ The rate at which the weights decrease is controlled by the parameter α

Note that the sum of the weights even for a small α will be approximately equal to one.

For any α between 0 and 1, the weights attached to lagged observations decrease exponentially. If α is small (i.e., close to 0), more weight is given to observations from the more distant past. If α is large (i.e., close to 1), more weight is given to the more recent observations.

At the extreme case where $\alpha = 1$, $y_{T+1|T} = y_T$ and forecasts are equal to the naive forecasts.

So how does it work?

First, we need to be aware of the initialization requirement: if you work through all the math (it's in the FPP book), you'll notice that there is dependence on the initial condition. All future forecasts are built on the very first forecast: the book denotes it by ℓ_0

In general, the weight attached to ℓ_0 is small.

However, in the case that α is small and/or the time series is relatively short (T is small), the weight may be large enough to have a noticeable effect on the resulting forecasts. Therefore, selecting suitable initial values can be quite important.

It is common to initialize things using the first observation in the series. There are arguments that it is better to treat the initial value as a parameter, along with α. This requires optimization because there is no way to solve the for the solution.

Optimization

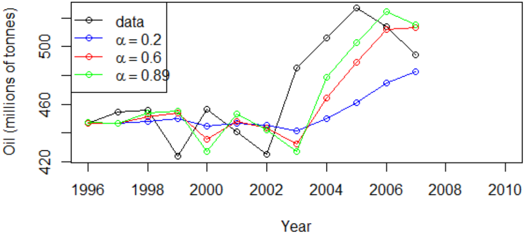
For every exponential smoothing method we also need to choose the value for the smoothing parameters. For simple exponential smoothing, there is only one smoothing parameterα.

There are many cases where the α is chosen by domain knowledge or forecaster experience.

In the absence of those, a robust and objective way to obtain values for the unknown parameters included in any exponential smoothing method is to estimate them from the observed data. This generally involves minimizing the sum of squared errors (residuals).

Example: Oil

Let's look at a chart Annual oil production in Saudi Arabia, 1965-2010. Here's a chart of the raw data as well as the smoothed version with different values of α.



Forecasting past tomorrow

Simple exponential smoothing has a flat forecast function, and therefore for longer forecast horizons,

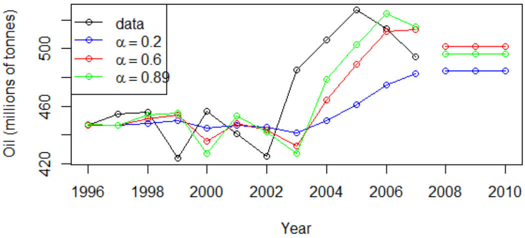
$\hat{y}_{T+h|T} = \hat{y}_{T+1|T} = \ell_T$ for $h = 2, 3, \ldots$

Remember this method is a combination of naïve and mean forecasts. When will this actually be helpful?

When the data have no trend or seasonal pattern.

Example: Oil with forecast

And the picture of the forecasts from the SES model?



Another approach:Holt's linear trend method

Extended simple exponential smoothing to allow forecasting of data with a trend. This method involves a forecast equation and two smoothing equations (one for the level and one for the trend):

Forecast equation	$\hat{y}_{t+h\|t} = \ell_t + hb_t$
Level equation	$\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1})$
Trend equation	$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$

where ℓ_t denotes an estimate of the level of the series at time t, b_t denotes an estimate of the trend (slope) of the series at time t, α is the smoothing parameter for the level, $0 \leq \alpha \leq 1$ and β^* is the smoothing parameter for the trend, $0 \leq \beta \leq 1$ (we denote this as β^* instead of β).

English please?

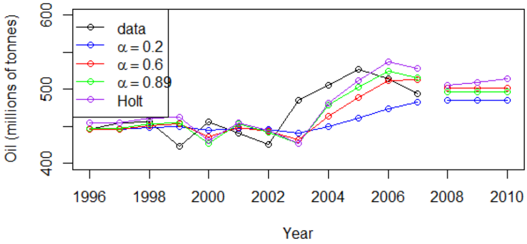
The forecast for $\hat{y}_{t+h|t}$ has two parts: the level ℓ_t and the product of the number of periods ahead (h) and the trend b

The level equation shows that ℓ_t is a weighted average of today's observation (y_t) and yesterday's forecast for today (the forecast equation for $t-1$ where $h=1$ is $\hat{y}_{t|t-1} = \ell_{t-1} + 1b_t - 1$)

The trend equation shows that b_t is a weighted average of the estimate at time t based on the difference between today's level ℓ_t and yesterday's level ℓ_{t-1} and yesterday's estimate of the trend b_{t-1}.

So what does this mean? It means that the forecast function isn't flat anymore but has a trend and the effect of that trend is dependent on how far out you are trying to forecast (h).

Picture please



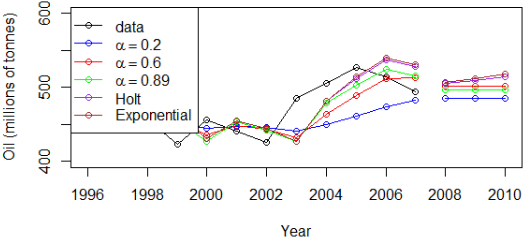
Up next: exponential trend method

This is a variation on Holt's method where the forecast depends on a multiplicative relationship rather than additive:

$$\begin{array}{ll} \text{Forecast equation} & \hat{y}_{t+h|t} = \ell_t b_t^h \\ \text{Level equation} & \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} b_{t-1}) \\ \text{Trend equation} & b_t = \beta^* \frac{\ell_t}{\ell_{t-1}} + (1-\beta^*) b_{t-1} \end{array}$$

These changes mean that the trend is exponential rather than linear so that forecasts now show a constant growth rate rather than a constant slope.

Picture please



To infinity and beyond!

One potentially unappealing aspect of the Holt forecasts is that the trend continues infinitely into the future. Whether the slope or the growth is constant, there's nothing in this technique that accommodates the possibility of leveling off.

To achieve that we need to add a term to help to dampen the effect - that's another parameter with another Greek letter.

Not only do we have the smoothing parameters α (which applies to both SES and Holt functions) and β^* (which is found only in the Holt models), we now add ϕ which also lies between 0 and 1 as our dampening parameter.

Additive damped model

That means our additive model becomes:

Forecast equation	$\hat{y}_{t+h\|t} = \ell_t + (\phi + \phi^2 + \ldots + \phi^h) b_t$
Level equation	$\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$
Trend equation	$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$

The dampening parameter ϕ is applied to the trend component of the equations to help keep the trend from going off into space.

If $\phi = 1$, this is just Holt's original addiitve model. For $0 < \phi < 1$, the trend dampens at a particular rate so that it eventually becomes flat at some point in the future. The smaller the ϕ, the faster the flattening.

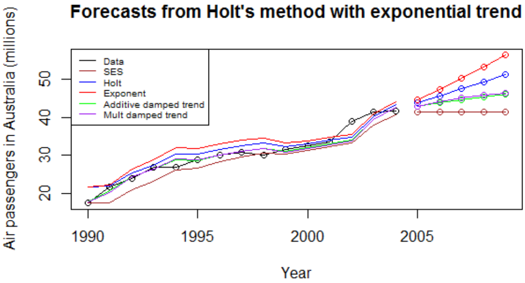
Multiplicative damped model

And the exponential model becomes:

$$\begin{array}{ll} \text{Forecast equation} & \hat{y}_{t+h|t} = \ell_t b^{(\phi+\phi^2+\ldots+\phi^h)} \\ \text{Level equation} & \ell_t = \alpha y_t + (1-\alpha)\ell_{t-1} b_{t-1}^{\phi} \\ \text{Trend equation} & b_t = \beta^* \frac{\ell_t}{\ell_{t-1}} + (1-\beta^*) b_{t-1}^{\phi} \end{array}$$

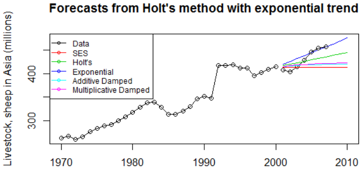
This method produces less conservative forecasts.

All together now

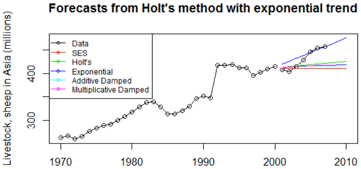


(Where $\alpha = 0.8, \beta = 0.2$)

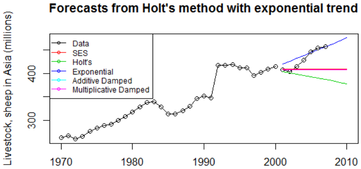
Changing the smoothing parameter



alpha = 0.8, beta = 0.2

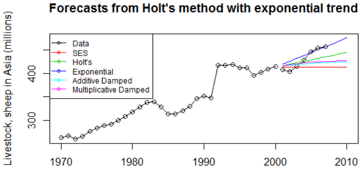


alpha = 0.6,beta = 0.2

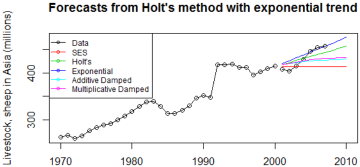


alpha = 0.4,beta = 0.2

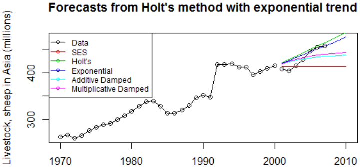
Changing the trend parameter



alpha = 0.8,beta = 0.3



alpha = 0.8,beta = 0.4



alpha = 0.8,beta = 0.6

But what about seasonality?

We've smoothed, we've trended but we haven't dealt with seasonality.

There's an algorithm for that too. Holt-Winters seasonal method has four parts: the forecast equation and three smoothing equations - one for the level ℓ_t, one for trend b_t, and one for the seasonal component denoted by s_t, with smoothing parameters α, β^* and γ. We use m to denote the period of the seasonality, i.e., the number of seasons in a year.

Like before, there are additive and multiplicative versions.

The additive method is preferred when the seasonal variations are roughly constant through the series

The multiplicative method is preferred when the seasonal variations are changing proportional to the level of the series.

Math for additive seasonal model

$$\begin{aligned}
\hat{y}_{t+h|t} &= \ell_t + hb_t + s_{t-m+h_m^+} \\
\ell_t &= \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}
\end{aligned}$$

where $h_m^+ = \lfloor (h-1) \mod m \rfloor + 1$, which ensures that the estimates of the seasonal indices used for forecasting come from the final year of the sample. (The notation $\lfloor u \rfloor$ means the largest integer not greater than u.)

So the forecast now has 3 pieces: the level, the trend and the seasonal part.

The details

The level equation shows a weighted average between the seasonally adjusted observation $(y_t - s_{t-m})$ and the non-seasonal forecast $(\ell_{t-1} + b_{t-1})$ for time t. The trend equation is identical to Holts linear method.

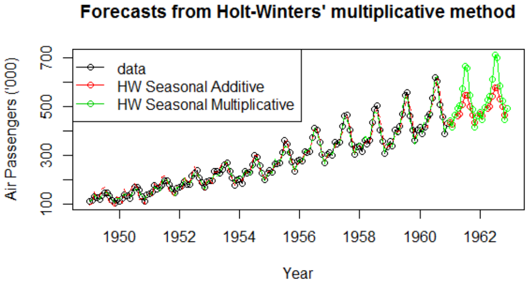
The seasonal equation shows a weighted average between the current seasonal index, $(y_t - \ell_{t-1} - b_{t-1})$, and the seasonal index of the same season last year (i.e., m time periods ago).

And of course there's a multiplicative version

$$
\begin{aligned}
\hat{y}_{t+h|t} &= (\ell_t + hb_t)s_{t-m+h_m^+}. \\
\ell_t &= \alpha\frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma\frac{y_t}{(\ell_{t-1}+b_{t-1})} + (1-\gamma)s_{t-m}
\end{aligned}
$$

which is too ugly to deal with.

Picture please

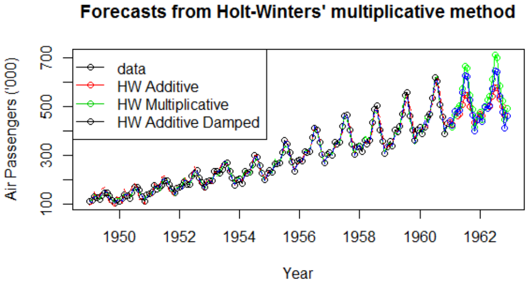


One final version

There is the ability to add a damping parameter to this model too. Just to show you the scary math:

$$
\begin{aligned}
\hat{y}_{t+h|t} &= [\ell_t + (\phi + \phi^2 + \cdots + \phi^h) b_t] s_{t-m+h_m^+}. \\
\ell_t &= \alpha(y_t / s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1} \\
s_t &= \gamma \frac{y_t}{(\ell_{t-1} + \phi b_{t-1})} + (1-\gamma) s_{t-m}
\end{aligned}
$$

Picture please



The point

There is a whole family of smoothing algorithms (15) that you can choose from depending on what characteristics of your data you need to model.

The FPP book has a "taxonomy" chart with all the Greek letters but here's a handy summary table:

	Seasonal Component		
Trend	N	A	M
Component	(None)	(Additive)	(Multiplicative)
N (None)	N,N	N, A	N, M
A (Additive)	A,N	A,A	A,M
A_d (Additive damped)	A_d,N	A_d,A	A_d,M
M (Multiplicative)	M,N	M,A	M,M
M_d (Multiplicative damped)	M_d,N	M_d,A	M_d,M

What does this mean for R code?

Here’s how the methods we discussed map to code options:

(N, N)	= simple exponential smoothing	⇒ ses()
(A, N)	= Holts linear method	⇒ holt()
(M, N)	= Exponential trend method	⇒ holt(exponential = TRUE)
(A_d, N)	= additive damped trend method	⇒ holt(damped = TRUE)
(M_d,N)	= multiplicative damped trend method	⇒ holt(damped = TRUE, exponential = TRUE)
(A, A)	= additive Holt-Winters method	⇒ hw(seasonal = ”additive”)
(A, M)	= multiplicative Holt-Winters method	⇒ hw(seasonal = ”mulitplicative”)
(A_d,M)	= Holt-Winters damped method	⇒ hw(seasonal = ”additive”, damped = TRUE)

Alternatively:

- ses() implements method (N,N)
- holt() implements methods (A,N), (Ad,N), (M,N), (Md,N)
- hw() implements methods (A,A), (Ad,A), (A,M), (Ad,M), (M,M), (Md,M)

State Space Model

Now we are using fancy terms. The algorithms in the table (ses, holt, hw) are methods to generate point estimates (individual fitted values for the forecast horizons). What they don't do is generate confidence intervals - we need to look at the statistical models behind the math.

Note the following line from the FPP book: *A statistical model is a stochastic (or random) data generating process that can produce an entire forecast distribution.*

We've seen this with the simple forecasting methods we've been playing with (mean, drift, etc.) Now we are going to put things in place for these exponential smoothing models.

Accounting for errors

Each of the methods we've seen so far have equations that involve the observed data as well as those to deal with various "states" (level, trend, seasonal) - this is where we get the term "state space model"

What we need to be able to do is account for where these models might miss the mark. Like the equations for the different states, we can model those errors as being additive or mutiplicative. This means that our 15 methods expands to 30 models.

So we have a third dimension (Error) which applies to the models and not the methods. So now our approach covers Error, Trend, and Seasonal so we'll use the terminology **ETS**.

Error	$\Rightarrow A, M$
Trend	$\Rightarrow N, A, A_d, M, M_d$
Seasonal	$\Rightarrow N, A, M$

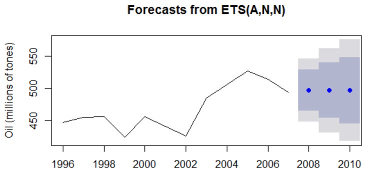
So how does this work?

ETS(A,N,N) : model with **A**dditive errors, **N**o trend, **N**o seasonality.
This is the SES model where we add an additive error component.

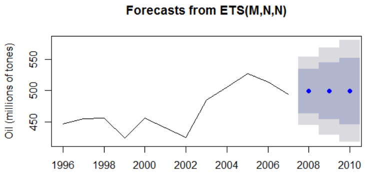
ETS(M,N,N) : model with **M**ultiplicative errors, **N**o trend, **N**o seasonality.
This is the SES model where we add an multiplicative error component.

They produce the same point forecast but different prediction intervals.

Example:



Forecasts from ETS(A,N,N)



Forecasts from ETS(M,N,N)

Hard to see but the error bands on the ETS(M,N,N) are wider.

Model selection

If we have 30 models, how do we know which one? The ETS framework will estimate all 30 and return a suggestion based on minimizing Akaike's Information Criteria (AIC).

The AIC measures of the relative quality of statistical models for a given set of data. It calculates goodness of fit as calculated by the likelihood function including a penalty for number of parameters because we know that more explanatory variables generally increases the model fit.

If we run the ETS function in R without specifying the E or the T or the S, it will choose that for us. We'll see that in the code shortly.

Summary

Topics for tonight:

- Introduction to smoothing including:
- 15 different exponential smoothing methods including ugly math that we won't need to go over again.
- High level discussion of state space models and the 30 options we have.
- Introduction to the ETS framework and model selection.

And now, we code.