Exponential smoothing

William Mann

EMORY

GOIZUETA
BUSINESS
SCHOOL

Simple exponential smoothing

Simple exponential smoothing

Exponential smoothing is a family of weighted-average methods.
The simplest example is called **simple** exponential smoothing:

- First you make a forecast F_1 about the very first observation y_1.
- After you observe y_1, your forecast of F_2 is given by

$$F_2 = \alpha y_1 + (1 - \alpha) F_1$$

- Similarly, after you observe y_2, your next forecast is

$$F_3 = \alpha y_2 + (1 - \alpha) F_2$$

  and so forth.
- The value of α is called a smoothing parameter.

Why is it called exponential smoothing?

The procedure on the previous slide gives us this formula:

$$F_{t+1} = \alpha \times \left[y_t + (1-\alpha) y_{t-1} + (1-\alpha)^2 y_{t-2} + ... \right]$$

As time goes by, your initial forecast F_1 no longer matters.

There is always much greater weight given to recent observations rather than past observations, but at the same time, the process "remembers" a very long history of information about y_t.

This approach achieves a lot for the cost of only one parameter α.

Implementation: Choice of initial forecast

How to choose the initial forecast F_1? This should not be a big issue.

- Any different choice of F_1 will give you slightly different results, but it should not be a big issue in practice.
- If two reasonable choices yield very different results, something may be going wrong, or you may just need more data.
- The simplest choice is to "look ahead" and just use y_1.
  This is what we will do when implementing by hand.
- In this case the first forecast is slightly cheating, and we should ignore it when assessing the performance of the model.
- Software will often "backcast" from the early observations.
  Every common approach involves looking ahead in some way.

Implementation: Choice of smoothing parameter α

How to choose α? This is a more important issue.

- Notice that α is the amount that we change our forecast in response to a new observation. So actually a higher α makes our forecasts *less* smooth. (This is a bit confusing but is standard!)
- If the goal is just to visualize data, it's common to just try values of α until one looks reasonable.
- If the goal is to forecast, then α is typically chosen to minimize the sum of squared one-step forecast errors, $\sum_t (y_t - F_t)^2$.
- Although it looks much like a regression, there is no simple formula for this. Instead, search numerically over values of α.

See examples on Canvas.

Double exponential smoothing
(also called Holt's method)

Double exponential smoothing

When a series is trending in one direction, simple exponential smoothing will always generate forecasts that lag behind.

Double exponential smoothing keeps track of both the *level* of the series (as before), and also its *trend*.

The second part gives a sense of where the series is going. This improves forecast accuracy for data with slowly-evolving trends.

The formal framework is as follows:

$$\begin{aligned} L_t &= \alpha Y_t + (1-\alpha)(L_{t-1} + T_{t-1}) \\ T_t &= \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1} \\ F_{t+1} &= L_t + T_t \end{aligned}$$

(There is also a popular version with multiplicative trends, but it is easier and almost identical to just log-transform the data in the above approach.)

Double exponential smoothing: Implementation

How to set the initial values of the forecast components, L_1 and T_1?

- As before, we can set $L_1 = Y_1$. For T_1, a common choice is $T_1 = Y_2 - Y_1$, and this is what we will do in our examples.
- Most software follows an approach based on fitting a simple regression model to the early data.
- As with SES, this choice generally should not make a very big difference to your forecasts.

How to choose the parameters α and β?

- Again, just minimize the sum of squared one-step forecast errors.
- We just have to search over two parameters instead of one.

Triple exponential smoothing
(also called the Holt-Winters approach)

Triple exponential smoothing, aka Holt-Winters

Simple exponential smoothing captures the current *level* of a series.
Double exponential smoothing learns which way it's *trending*.
The other big component of most economic data is *seasonality*.

Neither simple nor double smoothing can capture seasonality,
but we can extend the method one more time to do so.

The result is called **triple exponential smoothing** or the
Holt-Winters method, or sometimes just **Winters method**.

This has been an extremely popular and effective forecasting method
for many decades now.

Triple exponential smoothing: Framework

$$
\begin{aligned}
L_t &= \alpha(Y_t - S_{t-p}) + (1-\alpha)(L_{t-1} + T_{t-1}) \\
T_t &= \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1} \\
S_t &= \delta(Y_t - L_t) + (1-\delta)S_{t-p} \\
F_t &= L_{t-1} + T_{t-1} + S_{t-p}
\end{aligned}
$$

We will not try to implement this by hand. The extra math, and the question of how exactly to initialize S_t, are tricky but not interesting.

The main takeaway is that this approach can capture rich behavior, and can quickly generate accurate forecasts in most practical settings, while estimating just three parameters (α, β, δ).

Again, there is also a version with *multiplicative* seasonality, in which we divide and multiply by S_t and S_{t-p}, instead of subtracting and adding. For some reason this version is very popular. In my experience the two versions give similar results.

Exponential smoothing: Summary

- Exponential-smoothing is a very popular forecasting approach.
  But notice that we derived it in a very *ad-hoc* way.
- As long as we are just forecasting, this may not be a problem.
- But there is less clear guidance for other goals, such as building prediction intervals, connecting with economic models, etc.
- These questions eventually require us to model the *process* that generates the data, as is usually done in statistics.
- From this point forward, we will develop such a framework.
  It is often called the ARMA framework, for reasons we will see.
- The point is not to generate better forecasts (it won't),
  but rather to give a clearer understanding of what we are doing.