Summer Intern Lectures
Time Series Analysis Team
June 2017

1 Kalman filter

1.1 Overview?

The Kalman filter is a technique that allows to recursively update one's beliefs of an unobserved (at least contamporaneously) variable. It uses Bayesian updating and historical data to predict a particular variable and potentially to forecast the state of said variable in the future.

Rocket example

The typical example used to give intuition of the Kalman filter is that of identifying a rocket. You do not know where the rocker at times t (we indicate this by x_t) is now but you have an idea, summarized in what is called a *prior distribution $p(x)$*. Now, suppose you get new information about the position of the rocket, for example you get a satellite recording of it and you know it is at y_t (the measurement itself will have some uncertainty). You ask yourself: where is the rocket at $t + 1$? Having more information to answer this question, you will use both your prior and the actual location of the rocket to determine y_{t+1}. Putting prior and data together will give you the *posterior distribution*. The posterior distribution becomes your prior on the position of the rocket of $t + 1$ and so on we can continue until the rocket hits something.

Here I steal from QuantEcon which displays this very well...
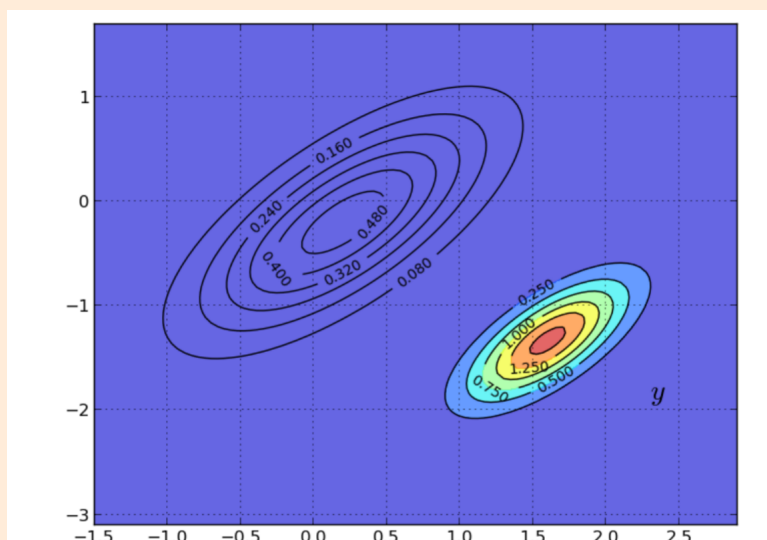
Notice, this relates very closely to the nowcast! As actual data comes out, we compare our *prior*, i.e. the forecast, to the release and then we update our model!

1.2 Methodology

Here we outline the steps to run the Kalman filter.

step 1 **Prior**

Before we start we will need to have a guess (or prior) on the rocket x. This will be summarized in a probability distribution $p(x)$.

step 2 **Filtering**

The second step is to update our prediction after having seen the data. We receive data about the state of the rocket that can be summarized as $p(y)$ (we shall call this the likelihood, responding tot he question "How *likely* are we to see the data?"). Both our initial prior and the data are uncertain so we take both of them into account to update our beliefs. The updating process is done very simply following Bayes' law:

$$p(x|y) = \frac{p(y|x) \cdot p(x)}{p(y)} \qquad (1)$$

Where we call $p(x|y)$ the *posterior* distribution which should be thought of as a weighted average between what the data suggests and your prior belifs. In particular, if the data is very noisy and your prior is very tight (i.e. you are fairly certain about the position of the rocket), the posterior will weigh your prior beliefs heavily and the data little, similarly, if the data is not noisy and your believs are flat (you are very uncertain about the position of the rocket a priori), the posterior will weigh the data more and the prior less.

This step is called the *filtering* step because it filters out the noise from the data and our prior when coming up with the posterior distribution.

step 3 **Forecasting**

If we want to forecast the position of the rocket in the next period or the one after that we need a forecasting model that relates the position of the rocket this period with the position of the rocket next period.

An example of such model would be the following:

$$x_{t+1} = f(x_t) + w_{t+1}, \qquad (2)$$

this law of motion (or transition equation) should look very familiar to those who have studied state-space models, in fact, in order to run the Kalman filter, you need to put your model in state-space format!

To get a prediction (often summarized as a whole predictive distribution) we simply put together our filtering distribution $p(y|x)$ (the posterior distribution from the filtering step) and our law of motion $f(x_t)$. The predictive distribution will be nothing other than:

$$p(x_{t+1}) = f(p(y|x)), \qquad (3)$$

(note this is not the cleanest way of writing this, but it conveys the idea).

1.3   Summary and recursion

Often we get multiple updates regarding the position of the rocket, at each one of these we are able to apply the two steps outlined above recursively:

```
for  t=1:T;
```

- Have a prior for the current period t, $p_t(x)$;

- Observe measurement y_t;

- Compute filtering distribution (posterior distribution) $p_t(x|y)$;

- Make prediction of x_{t+1}, $p_{t+1}(x)$ using the filtering distribution and the transition equation (this will be the prior for your next iteration).

```
end
```

- your initial prior $p(x_t)$