

Options Pricing

Alun Meredith

ABSTRACT

1. NOISE FILTERING

Separate the noise from underlying trend through various models

1.1 Auto-regressive models

Auto-regression models the behaviour of stock price today as a function of the previous days plus some random noise:

$$y_t = \sum_{i=1}^{Order} \beta_i x_{t-i} + \epsilon_t \quad (1)$$

It is natural to think of stock price as auto-regressive; today's price is yesterday's price + returns. Although it is not clear what order the function would take and a stationary form clearly over-simplifies, for example volatility (size of the noise term) is known to not be stationary. There are extensions to this model such as GARCH models which account for varying volatility and ARIMA processes which include moving average effects.

An auto-regression model can easily be computed by minimising least squares with the pseudo-inverse for a set of time lagged features, such as for Order = 3 below:

$$\begin{bmatrix} x_1 & x_2 & x_3 \\ x_2 & x_3 & x_4 \\ \vdots & \vdots & \vdots \\ x_{n-3} & x_{n-2} & x_{n-1} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{Order} \end{bmatrix} = \begin{bmatrix} y_4 \\ y_5 \\ \vdots \\ y_n \end{bmatrix} \quad (2)$$

1.2 Kalman Filtering

Another method of filtering the noise of a signal is using Kalman filtering. This is a state based model with the advantage that it is updated iteratively. This means that upon receiving new data the model isn't recomputed but updated, a much more computationally efficient process.

The state space model is defined as an underlying value θ following an order 1 autoregressive function and an observed stock price as z which is the underlying value augmented by some noise. There are only two parameters which need to be defined; the noise of the observation V , estimated as the variance variance in residual from the AR model and the noise of the underlying value W which is tuned empirically using cross validation and RMSE.

$$z_t = \theta_t + v_t \quad v_t \sim N(0, V) \quad (3)$$

$$\theta_t = \theta_{t-1} + \omega_t \quad \omega_t \sim N(0, W) \quad (4)$$

$$\theta_0 \sim N(m_0, V) \quad (5)$$

This leads to the following set of iterative update equations:

Prediction

$$\hat{\theta}_{t|t-1} = \hat{\theta}_{t-1|t-1}$$

$$P_{t|t-1} = P_{t-1|t-1} + W$$

Correction

$$r_t = z_t - \hat{\theta}_{t|t-1}$$

$$S_t = P_{t|t-1} + V$$

$$K_t = P_{t|t-1} S_t^{-1}$$

$$\hat{\theta}_{t|t} = \hat{\theta}_{t|t-1} + K_t r_t$$

$$P_{t|t} = (1 - K_t H_t) P_{t|t-1}$$

We simulate an autoregressive function to test the Kalman filtering:

$$Y = 0.7Y_{t-1} + 0.3Y_{t-2} + \eta \quad \eta \sim N(0, 0.1) \quad (6)$$

Figures 2 and 1 describes the effect of Kalman smoothing. Figure 1 shows how the Kalman filtering algorithm estimates the underlying weights of the autoregressive model. It shows that there is high variance initially but quickly gets close to the correct weights but doesn't converge to them. Figure 2 shows the effect of the size of the evolution noise W . As the value decreases smoothing increases, this is due to a higher level of certainty P for the current prediction and therefore less weight associated to the correction.

Evaluating a the Kalman Smoothing and AR models for a variety of orders and W values we can find the optimal values for these parameters. Figure 3 shows that low values of W and order or high values of both W and order are best. This makes some sense as low W retains previous predictions more, if the underlying model was a high order autoregressive function then low order Kalman filtering could 'get access to' the higher order data by having low W . If the Kalman filtering is high enough order it can 'forget' about the earlier data by having high W . Order 1 systems effectively predict tomorrow's value equals today's value because no intercept was used in our model. Finally Kalman filtering never reaches the accuracy of RMSE but this is in some sense not fair as the AR model has seen all

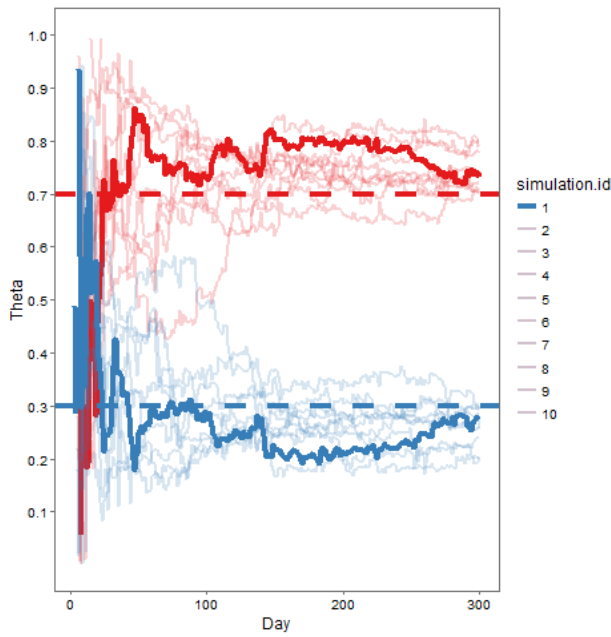


Figure 1: The prediction of underlying auto-regression weights from Kalman filtering over time for simulated data (6) ($W = 1e-4$)

the data and is fitting to a training set whereas the Kalman smoothing is predicting on unknown data at each step. A more accurate assessment of RMSE for AR could be done by computing the AR model iteratively for each t up to $t-1$ and then predicting the next value with it.

Finally we can see the effects of the 3 best models (Kalman with low order and W , high order and high W and Auto-regression) in figure 4. It shows low order/ W kalman being very similar to the auto-regression model while high W /order Kalman while similar doesn't smooth as much.

2. LASSO REGULARISATION

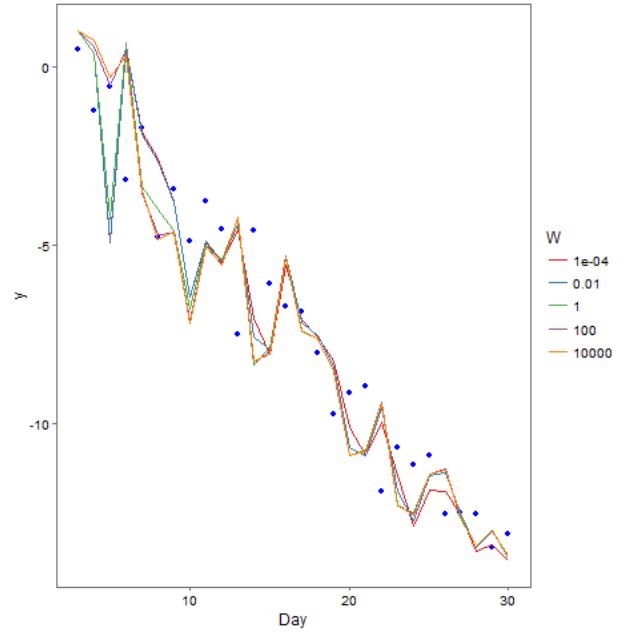


Figure 2: The Kalman smoothing vs. observed values for a variety of W values for simulated data (6)

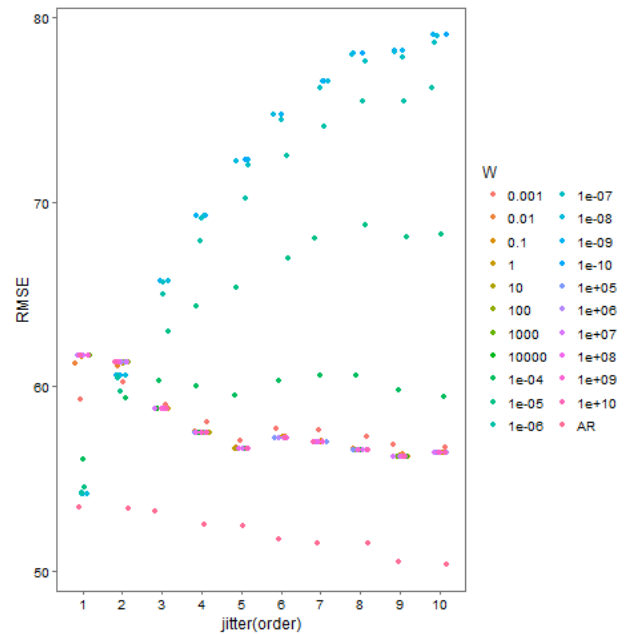


Figure 3: RMSE of Kalman smoothing and AutoRegression against Historic Data for a variety of orders and W values, x value is "jittered" to show overlapping points clearly

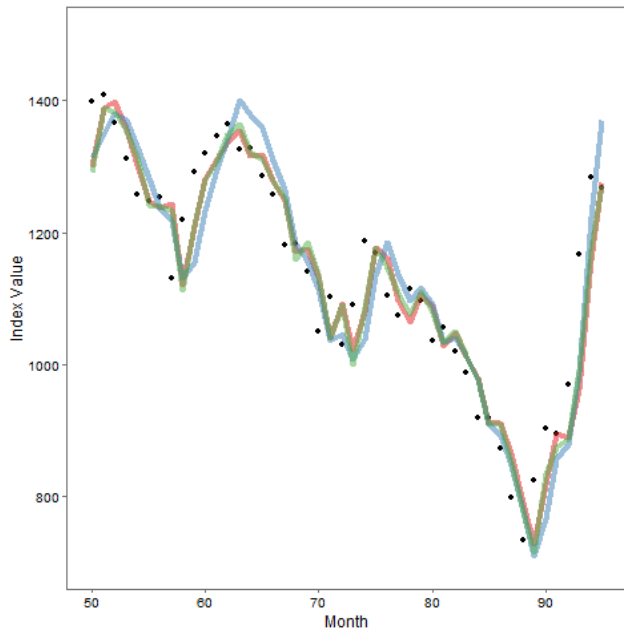


Figure 4: Effects of smoothing for: Kalman Order 1, $W 1e-7$ (red), Kalman Order 5, $W 1e5$ (blue) and Autoregression, Order 3 (green). On a subset of the data (Months 50-100, chosen to give Kalman filters enough time to train and capture shifts in the direction of the data)