The Two-Scale Realized Variance

1 Sampling and Observation as Before

We consider the efficient X evolving in continuous time as

$$dX_t = \sqrt{c_t} dW_t + dJ_t.$$

For a while we restrict $t \in [0, 1]$ but everything below holds day-by-day if $t \in [0, T]$. As before,

$$\Delta_i^n X = X_{i\Delta_n} - X_{(i-1)\Delta_n}$$

We allow for the possibility of measurement error (noise) by way of

$$Y_i^n = X_{i\Delta_n} + \chi_i$$

where χ_i reflects the noise, is taken to have mean zero, variance σ_χ^2, and is serially uncorrelated. If $\sigma_\chi^2 = 0$ then we are back in the case studied before, while if $\sigma_\chi^2 > 0$, then we are in the noisy case.

Keep in mind that the noisy Y_i^n represent the data at the highest frequency in the core (raw) data set. If the base data set contains only 1-min data, then Y_i^n means the 1-min log-price observations. If the base data set contains 15-second data, then the Y_i^n represent the 5-second log-price observations.

2 Coarse Sampling

All semester we have using coarse sampling as a way to circumvent the effects of the noise. Suppose, for the sake of example, the core (raw) data set has 30-second data but we decide to use 5-min data instead. We would form RV as

$$RV = (Y_{10}^n - Y_0^n)^2 + (Y_{20}^n - Y_{10}^n)^2 \ + \ \ldots \ + \ (Y_L^n - Y_{L-10}^n)^2$$

where L is the largest multiple of 10 such that $L \leq n$; i.e., $L = 10i_{end}$, $L = 10i_{end} \leq n$. Put more simply, we add up the sum of squared differences until we run out of data. Thankfully, MATLAB/Python can do that automatically without us needing to do the arithmetic to find the endpoints. Evidently, coarse sampling discards the data points $Y_1^n, \ldots, Y_9^n$, $Y_{11}^n, \ldots, Y_{19}^n$, etc. With coarse sampling, this discarding of data is just a price we pay to avoid the noise.

3 Subsampling and Two-Scale Realized Volatility

3.1 Subsamples

Many authors have noted that with coarse sampling we could actually form multiple RV-type estimators, by starting the sum of squares in RV using Y_1^n or Y_2^n, up to Y_9^n as the subtrahend. That is we can have the 10 different estimators

$$\begin{aligned}
RV_0 &= (Y_{10}^n - Y_0^n)^2 + (Y_{20}^n - Y_{10}^n)^2 \ + \ \dots \ + \ (Y_{L(0)}^n - Y_{L(0)-10}^n)^2 \\
RV_1 &= (Y_{11}^n - Y_1^n)^2 + (Y_{21}^n - Y_{11}^n)^2 \ + \ \dots \ + \ (Y_{L(1)}^n - Y_{L(1)-10}^n)^2 \\
RV_j &= \quad \vdots \qquad\qquad\qquad\qquad \vdots \qquad (Y_{L(j)}^n - Y_{L(j)-10}^n)^2 \\
RV_9 &= (Y_{19}^n - Y_9^n)^2 + (Y_{29}^n - Y_{19}^n)^2 \ + \ \dots \ + \ (Y_{L(9)}^n - Y_{L(9)-10}^n)^2
\end{aligned}$$

where $L(j)$ are as high as possible respecting the condition that we do not run out of data, i.e., $L(j) \leq n$. Then we can get a little more efficiency by averaging over the RV_j as in

$$RV^{\text{subave}} = \frac{1}{10} \sum_{j=0}^{9} RV_j.$$

The estimator RV^{subave} is still based on 5-min sampling, but it uses all of the data and therefore is a little more efficient that RV_0, which we have been using all semester.

3.2 TSRV

Now suppose the Y_i^n are ultra-high frequency data, say 5-second data. We could sample coarsely to get 1-min data, a gap of 12 between observations, or 5-min data, a gap of 60 between observations, or in general a gap of k_n between observations:

$$\begin{aligned}
RV_0 &= (Y_{k_n}^n - Y_0^n)^2 + (Y_{2k_n}^n - Y_{k_n}^n)^2 \ + \ \dots \ + \ (Y_{L(0)}^n - Y_{L(0)-k_n}^n)^2 \\
RV_1 &= (Y_{1+k_n}^n - Y_1^n)^2 + (Y_{1+2k_n}^n - Y_{1+k_n}^n)^2 \ + \ \dots \ + \ (Y_{L(1)}^n - Y_{L(1)-k_n}^n)^2 \\
&\vdots \\
RV_j &= (Y_{j+k_n}^n - Y_j^n)^2 + (Y_{j+2k_n}^n - Y_{j+k_n}^n)^2 \ + \ \dots \ + \ (Y_{L(j)}^n - Y_{L(j)-k_n}^n)^2 \\
&\vdots \\
RV_{k_n-1} &= (Y_{2k_n-1}^n - Y_{k_n-1}^n)^2 + (Y_{3k_n-1}^n - Y_{2k_n-1}^n)^2 \ + \ \dots \ + \ (Y_{L(k_n-1)}^n - Y_{L(k_n-1)-k_n}^n)^2
\end{aligned}$$

When using the ultra-high frequency data we have to be more careful about the noise. We know the RV_j is upward biased on account of the noise and Ait-Sahalia and Jacod, 2014 characterize the bias obtaining:

$$\frac{1}{k_n} \sum_{j=0}^{k_n-1} RV_j = IV + \boxed{\frac{2n}{k_n}\sigma_\chi^2} + e$$

where $\frac{2n}{k_n}\sigma_\chi^2$ is the bias due to noise and e is a small mean-zero error term. Recall that σ_χ^2 is the variance of the noise as described at the top of this lecture.

In order to bias-correct the mean of the RV_j we somehow need to estimate σ_χ^2. But that estimate can be obtained, because at the ultra-high frequencies the price moves are totally dominated by the noise. That is

$$\mathbb{E}\left[\left(Y_j^n - Y_{j-1}^n\right)^2\right] \approx 2\sigma_\chi^2.$$

Then the estimator is

$$\hat{\sigma}_\chi^2 = \frac{1}{2n}\sum_{j=1}^{n}\left(Y_j^n - Y_{j-1}^n\right)^2$$

Putting all the pieces together we get the Two-Scale Realized Variance (TSRV) estimator

$$RV^{TSRV} = \frac{1}{k_n}\sum_{j=0}^{k_n-1} RV_j - \boxed{\frac{2n}{k_n}\hat{\sigma}_\chi^2}$$

where the estimate of the bias is subtracted off. This subtraction is sometimes called de-biasing. Note, there is nothing to prevent $RV^{TSRV} < 0$.

3.3 Choice of k_n and Other Considerations

So far this semester, we have been working with 5-min data. Given raw data at 5-seconds, the choice of 5-min data means choosing $k_n = 60$. A reasonable view is that essentially all the information from high frequency data is contained in the 5-min data. For the DOW stocks, choosing 1-min data might have been OK, but sampling any finer creates a lot of headaches due to noise that has to be corrected as in the TSRV estimator. Is the effort worth the time cost? Possibly not.

The view that 5-min data contains all the information is by no means generally accepted by all financial econometricians. However, in the end, many financial econometricians (including the first author of the textbook) often wind up using coarse sampling in their empirical work. The problem is that the correction for the microstructure noise needed in order to use all of the data gets so complicated that the small gain doing so delivers makes it hardly worthwhile. The increase in precision is not a lot:

$$\text{Asy. Std Deviation of RV } = \sqrt{2\int_0^1 \sigma_s^4\, ds} \quad \text{Asy. Std Deviation of TSRV } = \sqrt{\frac{4}{3}\int_0^1 \sigma_s^4\, ds}$$

The relative precision is $\sqrt{\frac{4}{3}/2} = \sqrt{\frac{2}{3}} = 0.81$, so there is a provable gain of 19% from using TSRV.