

Lecture 3: Generalized Method of Moments

Chris Conlon

February 14, 2021

NYU Stern

Common Questions/Extensions

Semiparametric Efficiency

In a famous paper, Chamberlain (1987) showed that GMM obtained the semi-parametric efficiency bound asymptotically. That means as our sample gets large and without making additional parametric assumptions (such as placing distributions on error terms) the efficient GMM provides the most efficient estimator.

My first stage estimates look ok, but my second stage estimates look like garbage

This means there is a problem with your weighting matrix.

- Make sure you are subtracting off the average $g_N(\theta)$ when computing the covariance.
- Another problem appears is when you take the inverse. There is a statistic known as the *condition number* which measures the ratio of the minimum and maximum eigenvalue of a matrix.
 - When these eigenvalues are close together then small errors in $A + \epsilon$ lead to small errors in $(A + \epsilon)^{-1}$.
 - Software sometimes reports either the condition number, or its inverse.
 - In an ideal world this would be ≈ 1 .
 - If the condition number is $10^{\pm 13}$ it approaches the numerical precision of your computer.
 - Then even tiny (sampling) errors can lead to nearly infinite weighting matrices.

What causes these problems?

- Usually this happens if the gradient of $Q_n(\theta)$ is not close to zero in a particular dimension
- or the average Jacobian $D(\theta) = \frac{1}{N} \sum_{i=1}^N \frac{\partial g(w_i; \theta)}{\partial \theta}$ is not close to zero in some dimension.

You are not an optimum of $Q_n(\theta)$!

My point estimates look okay, but my standard errors look crazy

- You have probably dropped a $\frac{1}{N}$ or $\frac{1}{\sqrt{N}}$ somewhere.
- You can ignore the N 's when obtaining point estimates because scaling $Q_n(\theta)$ or W_n does not affect the location of $\hat{\theta}$, but you need to be more careful in computing V_θ .

If 2 step GMM is good, is 3-step GMM better?

- Asymptotically the answer is no
- Even with a sub-optimal choice of weighting matrix, your first stage $\hat{\theta} \xrightarrow{P} \theta_0$ as $N \rightarrow \infty$.
- This means that in the limit, you always recover the efficient choice of \hat{W} .
- In finite sample, anything can happen, but numerous studies have found little to no improvement from considering a K -step GMM estimator.

What about Infinite-step GMM?

There isn't such a thing. But there is the **Continuously Updating GMM** estimator of Hansen Heaton and Yaron (1996). In this case we let

$$W(\theta) = \left(\frac{1}{N} \sum_{i=1}^N (g(w_i, \theta) - g_N(\theta))(g(w_i, \theta) - g_N(\theta))' \right)$$
$$Q_n(\theta)^{CUE} = g_N(\theta)' \left(\frac{1}{N} \sum_{i=1}^N (g(w_i, \theta) - g_N(\theta))(g(w_i, \theta) - g_N(\theta))' \right) g_N(\theta)$$

- Because the weighting matrix now changes with θ , even for linear models this becomes very difficult to estimate.
- The original GMM problem was a quadratic optimization problem. The CUE problem is no longer a convex optimization problem. Quasi-Newton approaches are no longer guaranteed to work.

Advanced Asymptotics

CUE has some additional advantages over GMM. We see this by examining the GMM FOC.

$$\hat{D}(\theta) W_n \hat{g}_N(\hat{\theta}) = 0$$

If we take an Edgeworth expansion we can see how the **asymptotic bias** term looks:

$$\frac{1}{N^2} \sum_{i=1}^N E[g(w_i, \theta) W_n g(w_i, \theta)]$$

As long as the variance is finite we can see that this bias disappears as $N \rightarrow \infty$. However it also tends to grow linearly with q the number of moment restrictions. This is often referred to as the *too many moments* or *many instruments* problem.

The challenge is that in general when we estimate $\hat{g}(\hat{\theta})$ and $\hat{D}(\hat{\theta})$ we construct them so that they are mechanically correlated with one another. It turns out the CUE estimator fixes this in exactly the right way. How is beyond the scope of this note (and course). If you are really interested you should look at Newey and Smith (2004).

Thanks!
