MARC F. BELLEMARE

Agricultural and Applied Economics—Without Apology

'Metrics Monday: Least Squares Is But One Approach to Linear Regression

PUBLISHED APRIL 22, 2019

Last updated on April 21, 2019

One of the things I learned as an undergraduate at Montreal is the equivalence of ordinary least squares (OLS), maximum likelihood (ML), and the generalized method of moments (GMM) when it comes to linear regression.

This is something which I suspect a lot of people have lost track of in the wake of the Credibility Revolution, which has emphasized the use of linear methods.

(For instance, when I taught my causal inference with observational data class last semester, I remember showing my students a likelihood function and asking them whether they had covered maximum likelihood estimation in their first-year courses, drawing a number of blank stares.)

The idea is simple. When estimating the equation

(1) $y = bx + e$,

one has a choice of estimator, viz. OLS, ML, or GMM.

Intuitively,

1. OLS picks b so as to minimize the sum of squared residuals,
2. ML picks b so as to maximize the likelihood that the estimation sample is a random sample from a population of interest, and
3. GMM picks b by solving for what is known as a moment condition which, in the case of OLS, is such that $E(x'e) = 0$. That is, it chooses b to solve $E(x'(y - xb)) = 0$. Note that this is simply assuming that the regressors are uncorrelated with the errors, or an assumption of exogeneity.

If e is distributed normally, the OLS and ML estimators of b in equation 1 are identical. If the observations are independent and identically distributed (iid), the OLS and GMM estimators of b in equation 1 are equivalent.

Here is a bit of code that offers proof by Stata:

```
* Linear Regression Three Ways

clear
drop _all
set obs 1000
set seed 123456789
gen x = rnormal(0,1)
gen y = rnormal(5,1) + 10*x + rnormal(0,1)

* Ordinary Least Squares

reg y x

* Maximum Likelihood

capture program drop ols
program ols
  args lnf xb lnsigma
  local y "$ML_y1"
  quietly replace `lnf' = ln(normalden(`y', `xb',exp(`lnsigma')))
end

ml model lf ols (xb: y = x) (lnsigma:)
ml maximize

* Generalized Method of Moments

gmm (y - x*{beta} - {alpha}), instruments(x) vce(unadjusted)
```

In that code, I generate a variable x distributed N(0,1), and a variable y equal to a constant distributed N(5,1) added to 10 times variable x plus an error term e distributed N(0,1). Note that, by default, GMM estimates an IV-like setup; with OLS, this collapses to x serving as an instrument for itself.

The OLS, ML, and GMM estimators all yield the same point estimates of 5.014798 for the constant and 9.932382 for the slope coefficient. The standard errors in the example above are identical for the MLE and GMM cases, and they differ only very, very slightly for the OLS case.

A few things to note:

- MLE is a special case of GMM where a specific distribution is imposed on the data. In the ML example above, the normal distribution has been imposed on the data.
- Both GMM and MLE are iterative procedures, meaning that they start from a guess as to the value of b, then go on from there. In contrast, OLS does not guess, as its formula immediately solves for the value of b that minimizes the sum of squared residuals.

PUBLISHED IN **UNCATEGORIZED**

SUBSCRIBE

Join over 1020 subscribers in receiving new post notifications via email.

Your Email

Subscribe

SEARCH THIS SITE

ARCHIVES

Select Month