EXERCISE ON IV, LIML, ANGRIST-KRUEGER QOB

The Angrist-Krueger data set is available on the course web site as a .RData file and as a .txt file. The .RData form, akdataf.RData, has the data in the form of an R data frame. The variables, other than wage, are R "factors". This is convenient for using R `lm()` and `anova()` functions on the first item below. But for the IV and LIML calculations you will want to convert education to a numeric vector and quarter of birth to a set of dummy variables. The asciiqob.txt file has no labels, just the data, with each line containing, in order, log wage, education in years (0 to 20), year of birth (yob;30 to 39), place of birth (pob: 51 id numbers ranging from 1 to 56), and quarter of birth (qob: 1 to 4).

(1) Estimate an ordinary least squares regression of wage on education, yob (year of birth), and pob (place of birth), with each education level and year of birth a separate variable. Here we are ignoring endogeneity, but examining whether years of education enters linearly.
(2) Estimate, via instrumental variables, an equation relating wage to years of education (entered linearly), using the qob (quarter of birth) dummmies as instruments. Show the standard errors, as well as the point estimates, of the educationn coefficients.
(3) Write a program to evaluate the LIML likelihood with an improper prior proportional to the sum of the logs of the diagonal elements in d, in the singular value decomposition gamma = u d v′, where gamma is the matrix of coefficients in the regressions of endogenous right-hand-side variables on instruments. (In this single-x case, this is just the norm of the coefficient vector). Use it to find the posterior mode for the model you estimated with IV in 2. Then start an MCMC chain from this mode and iterate it until you think it has reasonably converged. Useful tools for checking convergence are available in the `coda` R package, which is also available in matlab/octave versions. With that package loaded, you can compute `effectivSize(mcmc(mcout))`, which should be 100 or more ideally at convergence for parameters of interest. You can also `plot(mcmc(mcout))`, which will give you "trace plots" of all the parameters. These should not show trends or slow oscilllations.
(4) Redo the IV and LIML estimations for a model in which now a high-school graduation dummy variable (equal to one when education is 12 or greater) is also included as a right-hand-side endogenous variable. For your MCMC results, in addition to the diagnostics you calculated before, generate a scatter

Date: September 20, 2018.

plot of the coefficients on the education and high school graduation dummies against each other.

We will discuss results in class next Tuesday, 9/25. You should have your work ready to present and on a usb stick. Feel free to calculate additional diagnostics or consider variations on the model. What policy conclusions are justified by the original Angrist-Krueger results, and do the results with a HS grad dummy cast doubt on those conclusions?

Code for the likelihood and to do the MCMC iterations is on the course web site. The optimization can be done with any optimization program. The one I've used is `csminwelNew`, which is available in R and (in an older version) in matlab/octave. That program can be installed as part of an R package that I put together for a previous time series course. It contains other programs as well, some of which will be useful in later parts of the course. The package is available at `\http://sims.princeton.edu/yftp/Times17/VARex/`. Download the zip file and extract it onto some location on your computer's file system. Then use R's **`install.packages`**`()` function to install it. See the R help for that function for how installation from a local directory is done.

The code on the web site, `ivlh()` forms the residual matrix at each iteration, then takes cross products. Since the number of observations is large, this is probably quite inefficient. I have code that I'm still testing that avoids this inefficiency that I'll post when I'm sure it works. What I have already posted is efficient enough for the exercises I've asked you to do so far. Of course if you write your own code from scratch, it would make sense to base it on the sample moments of the data, not the raw data.