Harvard Econometrics Math Camp
Worksheet #4

This session consists of two problems that I am asking you to solve by simulation in the programming language of your choice. The first problem concerns the intuition and assumptions behind some of the asymptotic tools we discussed today; in particular, the law of large numbers. The second problem is intended to get your hands dirty with matrix algebra (i.e. solving OLS with a computer by matrix inversion), but as a side-bonus, it's a nice constructive way to think about the central limit theorem and what the sampling distribution of an estimator looks like (though we will not appreciate this latter fact fully until Monday's discussion on statistical inference).

Problem 1: Sample Means and the LLN - This question is intended to make you think about the law of large numbers and when it applies. Please use the programming language of your choice to solve this problem - I'll use R.

1. Let $N = 100$ and and construct two $N \times 1$ vectors where each entry is an i.i.d. draw. from a standard normal distribution, $N(0,1)$. Call these vectors X_1 and X_2. Construct a new vector from these two vectors as $Y = X_1/X_2$; in other words, Y is the ratio of two independent standard normally-distributed variables.

2. Compute the sample means of X_1 and Y in this case with $N = 100$.

3. Repeat parts 1 and 2 with $N = 1000$, $N = 10000$, and $N = 100000$. What happens to the sample means of X_1 and Y? Is something weird happening? How does this relate to the law of large numbers?

4. Extra credit: Set $N = 100000$ and construct a new vector that is equal to the 'running sample mean' of X_1 and Y_1. Construct scatter plots of the running sample means of X_1 and Y as a function of n. Contrast the behavior of these figures and relate them to your discussion in (3).

Problem 2: Repeated Regression and the CLT

1. **Simulate some data**: The advantage of a simulation exercise is that you control (and therefore observe) the data generating process for your data. We will start by creating our data. Let $N = 50$. Generate an $N \times 1$ vector of ones, call it X_1. Generate an $N \times 1$ vector of independent draws from a standard normal distribution, call it X_2. Form an $N \times 2$ matrix, call it X, by concatenating these two column vectors; the first column of X is X_1 and the second column is X_2. Next, generate another $N \times 1$ vector of independent draws from a standard normal distribution, call it e. Let the "true" parameter vector $\beta = [1, 2]'$ (a column vector with 1 as the first entry and 2 as the second, and construct $Y = X\beta + e$.

2. **Generate OLS coefficients**: Consider the model $Y = X\beta + \epsilon$. Estimate the coefficients in this model using our simulated data and the matrix algebra characterization of the OLS estimator we described today. What are the estimated coefficients you obtain? What would you expect the OLS estimates to tend to as the simulated dataset size N grows large?

3. **Repeated simulation**: Refactor your code so that you run both steps above K times, where we initially will set K = 100. Make sure to re-draw the data in each simulation. You will want to save the estimated coefficients in each simulation as the rows of a $K \times 2$ matrix, call it B. Run this simulation exercise for $K = 100$ and $N = 50$ and examine a histogram of the estimated coefficients on X2 across simulations (i.e. the second element of the coefficient matrix). Now do the same thing, but with $K = 1000$ and $K = 10000$. What is happening to the distribution of estimates as you increase the number of simulations? Why do you think this is?

4. **Varying N**: What happens to the histogram from part 3 when you allow N to grow large as well?