Installation

This zip file contains source files and the primary data set for running the programs used to generate estimates and counterfactuals in Duflo, Hanna, and Ryan (2011), "Incentives Work: Getting Teachers to Come to School." The programs are written in Java SE using the Netbeans IDE. You will need the JAMA package for matrix operations, JFreeChart for plots, and the JSci package for statistical distributions. The code has a simple plug-in to use the Gridgain architecture if you have access to a cluster. I have commented out the KNITRO code, since it requires a paid license, but it is kept in the source code if you would like to use that for the optimization routines. Embedded in the optimization jar are the distribution files from the FPL Statistics Group (http://www1.fpl.fs.fed.us/optimization.html). Here are links (as of May 26, 2011) to the various files required to run these programs:

- Netbeans (www.netbeans.org)
- Java (http://www.oracle.com/technetwork/java/javase/downloads/index.html)
- Packages:
    - JAMA (http://math.nist.gov/javanumerics/jama/)
    - JSci (http://jsci.sourceforge.net/)
    - JFreeChart (http://www.jfree.org/jfreechart/)
    - GridGain (http://www.gridgain.com/)

Unzip the files into a directory. There are two subdirectories: src and data. When you start a Netbeans project with existing sources, point to the /src subdirectory as the location of the source files. You should make sure that you run the programs from the parent directory, as the programs look for the data files in /data. You will also have into include the two jar files (pmUtility.jar and optimization.jar) in the zip on your classpath, along with the packages listed above.

Running instructions

The primary entry point for running the program is CounterfactualMain.java. This program has boilerplate code for running every model that we reported (and a whole slew that we did not) and generating counterfactuals. Uncomment different lines in the file to run different models. Out of the box, the program will run Model V, which is a simple model with iid errors and common parameters across all agents. After minimizing the function, it will run a single counterfactual to compute the semi-elasticity of days worked with respect to the bonus cutoff. There are a series of Boolean switches in MainOptimalPolicy.java that control which counterfactuals are executed.