Using Machine Learning to Infer Unobserved Firm Networks—Part 2: benchmarking

Jesse Tweedle

November 9, 2016

Outline

1. Problem
2. Needs
3. Approach: separate data, benchmarking, methods steps, improve each one separately.
4. Data sources
5. Methods
6. Results

Problem

Have lots of data. Want to infer firm-firm trading relationships. But: want it to be internally consistent—a firm can't produce $10 million of output and only have customers that buy $1,000. Also need it to be externally consistent: if you add up all the firm-firm relationships, you should get back to industry accounts. More detail on this later.

Needs

Data: firm-firm (really location-location or establishment-establishment, we hope). We'll take anything that indicates a relationship between establishments. Method to infer links between establishments based on those data. Method to benchmark to the national accounts, and itself. Then look at results. Also want to seperate these things.

Approach

(1) Start with manufacturing. Get data + methods to work.
Check results. Refine each step.
Data: STF, IO, IPTF, ASM, etc.
Method: pick possible links using the data, then Lasso to
benchmark / solve system of equations.

Data sources (for now)

Data: STF, IO, IPTF, ASM, etc. Describe each one.

Methods

(1) Use IO, STF, ASM, IPTF to give any possible link between firms (e.g., an upper bound on links between firms), then a subset that we think is most likely (a lower bound on the set of links between firms).
(2) Benchmark to make expenditures between firms internally and externally consistent. Need to solve a huge, underdetermined system of equations, a big linear programming problem (tens-of-thousands of equations, hundreds of millions of parameters, maybe). Lasso is a good way.

Results

(1) it works, doesn't crash the server (for now). (2) it's relatively fast (x mins to solve manufacturing problem. (3) it works, relatively well, needs more refinement in input data to get it to work better—import/export registry, IO tables, final demand and such.