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

Jesse Tweedle

November 10, 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

Goal: study firm-firm trade

To understand

- supply chains and vertical integration
- intra-firm trade
- and more

Problem: we don't have firm-firm transaction data

But:

- we have lots of useful data
- and an idea of how to use it

Facts / needs

Facts: start with manufacturing

- $\approx$ 30k plants.
- $\approx$ 900m possible connections.
- $\approx$ 70 industries (IOIC)
- x goods (detailed confidential IOCC)

Needs

- Data that indicates relationship between plants
- Method to identify relationships
- Method to benchmark to make sure it all adds up

Normal approach: idea

To achieve this:

- Data that indicates relationship between plants
- Method to identify relationships

Do this:

- Use supply-use/input-output tables
- Assume every firm-firm relationship is the same as the industry IO relationship

Normal approach: problems

Implies way too many plant-plant relationships

- Use goods-only, square IO table, DC level
- $\approx 70^2 = 4900$ possible connections
- $\approx$ actual connections
- 50% density—half of the possible connections are given
- Gets worse using full table: 90% density of 50000+ possible connections

Normal approach: problems

Plant and IO data may not be consistent

- Relationship may not be consistent with plant data
- Plant-plant relationships may not be consistent with industry IO

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.