Lecture 6:
OLS Computation
Intro To Scraping

Big Data and Machine Learning for Applied Economics
Econ 4676

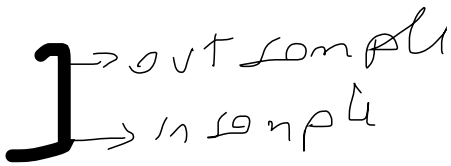
Ignacio Sarmiento-Barbieri

Universidad de los Andes

August 27, 2020

Recap

- What is Big Data?
- Quick Review of Statistical Properties
- Numerical Properties
- FWL
  - Fixed Effects
  - Leverage
  - Goodness of Fit



Agenda

1. Computation
   - Traditional Computation
   - Parallel vs Distributed
2. Web scraping
3. Further Readings

Motivation

- OLS workhorse

$$\hat{\beta} = (X'X)^{-1}X'y \tag{1}$$

- Involves inverting a $k \times k$ matrix $X'X$
- requires allocating $O(nk + k^2)$ if n is "big" we cannot store this in memory

Solving directly

```
beta<-solve(t(X)%*%X)%*%t(X)%*%y
```

may not be the smartest move

QR decomposition:

Most software use a QR decomposition:

Theorem If $A \in \mathbb{R}^{n \times k}$ then there exists an orthogonal $Q \in \mathbb{R}^{n \times n}$ and an upper triangular $R \in \mathbb{R}^{n \times k}$ so that $A = QR$

- Orthogonal Matrices:
  - Def: $Q'Q = QQ' = I$ and $Q' = Q^{-1}$
  - Prop: product of orthogonal is orthogonal, e.g $A'A = I$ and $B'B = I$ then $(AB)'(AB) = B'(A'A)B = B'B = I$
- **(Thin QR)** If $A \in \mathbb{R}^{n \times k}$ has full column rank then $A = Q_1 R_1$ the QR factorization is unique, where $Q_1 \in \mathbb{R}^{n \times k}$ and R is upper triangular with positive diagonal entries

Can use it these to get $\hat{\beta}$

$$(X'X)\hat{\beta} = X'y \tag{2}$$

$$(R'Q'QR)\hat{\beta} = R'Q'y \tag{3}$$

$$(R'R)\hat{\beta} = R'Q'y \tag{4}$$

$$R\hat{\beta} = Q'y \tag{5}$$

Solve by back substitution

QR decomposition:

$$X = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \quad y = \begin{pmatrix} 1 \\ 4 \\ 2 \end{pmatrix} \tag{6}$$

1. QR factorization X=QR

$$Q = \begin{bmatrix} -0.57 & -0.41 \\ -0.57 & -0.41 \\ -0.57 & 0.82 \end{bmatrix} \quad R = \begin{bmatrix} -1.73 & -4.04 \\ 0 & 0.81 \end{bmatrix} \tag{7}$$

2. Calculate $Q'y = [-4.04, -0.41]'$
3. Solve

$$\begin{bmatrix} -1.73 & -4.04 \\ 0 & 0.81 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} -4.04 \\ -0.41 \end{bmatrix} \tag{8}$$

Solution is $(3.5, -0.5)$

QR decomposition:

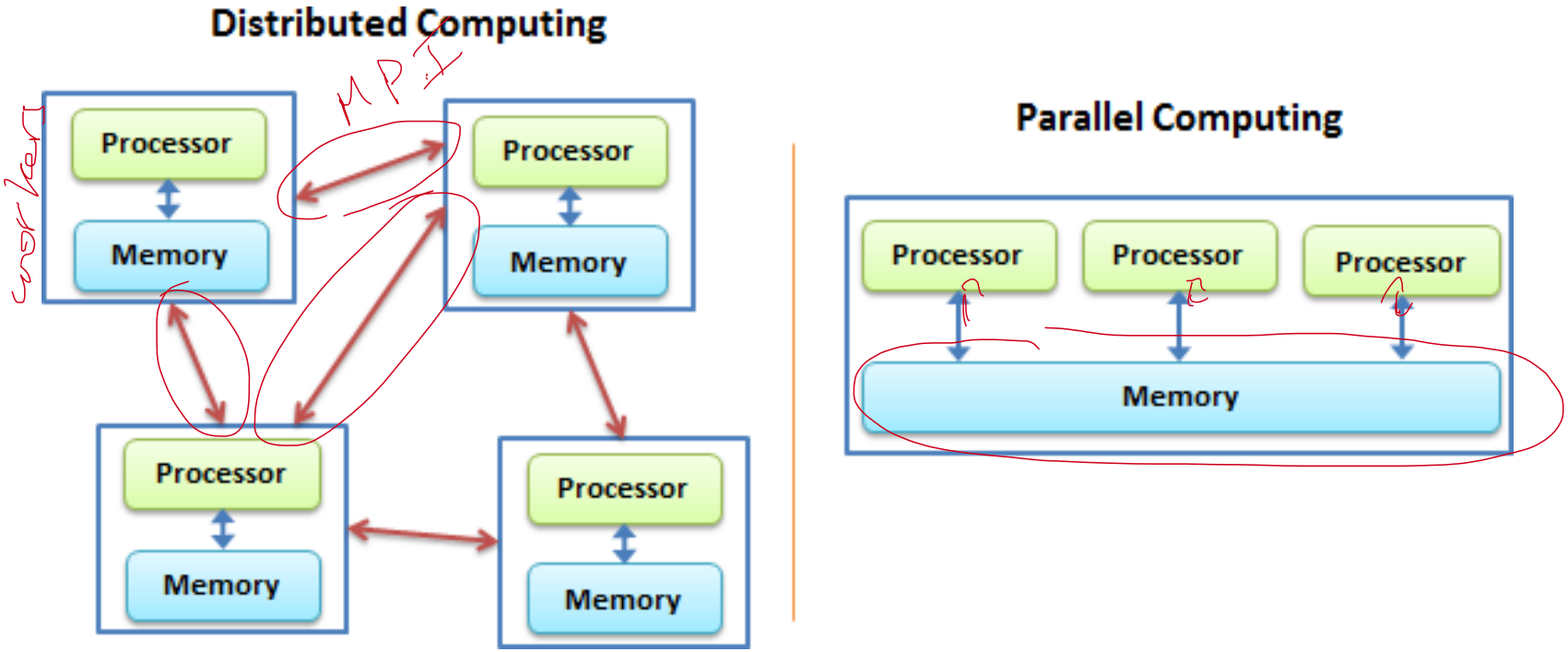
This is actually what R does under the hood

obj	list [12] (S3: lm)	List of length 12
coefficients	double [2]	-1.71e+08 3.01e+08
residuals	double [207607]	1.17e+09 -2.38e+08 -5.21e+08 -1.96e+08 -5.12e+07 -1.91e+08 ...
effects	double [207607]	-3.15e+11 2.10e+11 -5.24e+08 -1.98e+08 -5.34e+07 -1.93e+08 ...
rank	integer [1]	2
fitted.values	double [207607]	4.31e+08 4.31e+08 7.32e+08 4.31e+08 4.31e+08 4.31e+08 ...
assign	integer [2]	0 1
qr	list [5] (S3: qr)	List of length 5
df.residual	integer [1]	207605
xlevels	list [0]	List of length 0
call	language	lm(formula = price ~ bathrooms, data = dta0)
terms	formula	price ~ bathrooms
model	list [207607 x 2] (S3: data.fra	A data.frame with 207607 rows and 2 columns

Note that R's `lm` also returns many objects that have the same size as X and Y

Parallel vs Distributed

- An algorithm is parallel if it does many computations at once.
  - It needs to see all of the data
- It is distributed if you can work with subsets of data
  - `Stata-mp` is parallel. (license charges by core)
  - `R` and `Python` can be parallel **and** distributed

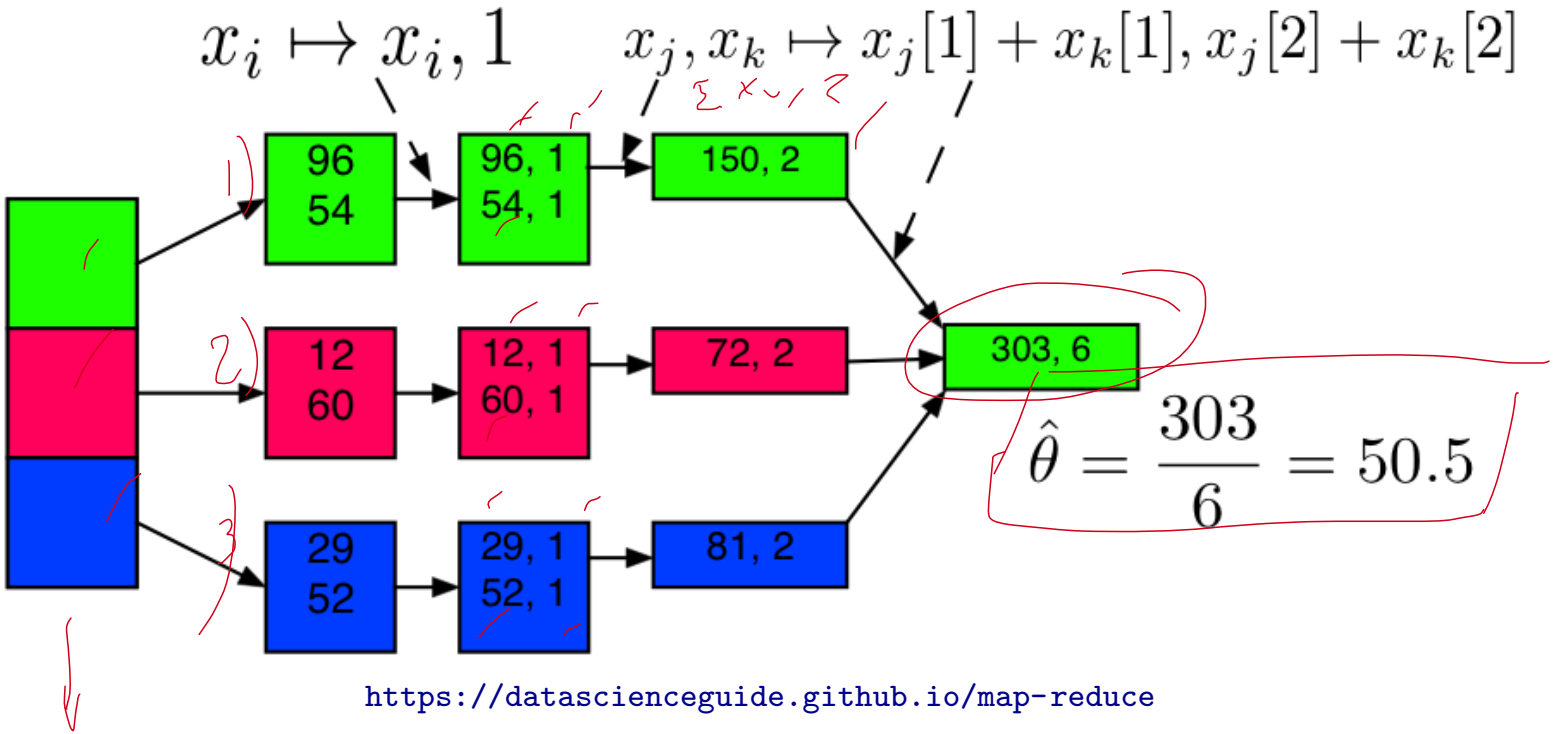


https://tinyurl.com/y3nzvkwh

Map Reduce

- Original Paper *MapReduce: Simplified data processing on large clusters (2004) Dean and Ghemawat*
- It is of the most popular frameworks
- Basic Idea:
  1. You need to be able to specify a key that indexes subgroups of data that can be analyzed in isolation.
  2. Map: Calculate and sort relevant statistics by key
  3. Partition and pipe the outcome of map so that outcomes with the same key end up on the same machine
  4. Reduce: Apply a summarization operation within the subgroup defined by each key.

Example: Mean by groups



https://datascienceguide.github.io/map-reduce

QR decomposition for block matrices

Idea on how to distribute OLS (Constantine & Gleich, 2011)

$$X_{8n\times k} = \begin{bmatrix} X^1_{2n\times k} \\ X^2_{2n\times k} \\ X^3_{2n\times k} \\ X^4_{2n\times k} \end{bmatrix} \tag{9}$$

QR to each block

$$X_{8n\times k} = \underbrace{\begin{bmatrix} Q^1_{2n\times k} & & & \\ & Q^2_{2n\times k} & & \\ & & Q^3_{2n\times k} & \\ & & & Q^4_{2n\times k} \end{bmatrix}}_{8n\times 4k} \underbrace{\begin{bmatrix} R^1_{k\times k} \\ R^2_{k\times k} \\ R^3_{k\times k} \\ R^4_{k\times k} \end{bmatrix}}_{4k\times k} \tag{10}$$

$$X_{8n\times k} = \underbrace{\underbrace{\begin{bmatrix} Q^1_{2n\times k} & & & \\ & Q^2_{2n\times k} & & \\ & & Q^3_{2n\times k} & \\ & & & Q^4_{2n\times k} \end{bmatrix}}_{8n\times 4k} \underbrace{\begin{matrix} Q_2 \\ \\ \\ \\ \end{matrix} \begin{matrix} R_2 \\ \\ \\ \\ \end{matrix}}_{4k\times k \;\; k\times k}}_{Q} \tag{11}$$

Spark

- The tools facilitating distributed computing are rapidly improving.
- One prominent system is Spark, that is quickly replacing MapReduce
- Seamlessly integration with `R` and `Python` and has it's own `MLlib`
  - E.g. Spark uses distributed version of stochastic gradient descent to compute OLS
- One of the key differences with MapReduce is how they load data
  - MapReduce has to read from and write to a disk
  - Spark loads it in-memory (can get 100x faster)

Motiviation Webscraping

Are Online and Offline Prices Similar? Evidence from Large Multi-Channel Retailers†

By ALBERTO CAVALLO*

AER

Online prices are increasingly used for measurement and research applications, yet little is known about their relation to prices collected offline, where most retail transactions take place. I conduct the first large-scale comparison of prices simultaneously collected from the websites and physical stores of 56 large multi-channel retailers in 10 countries. I find that price levels are identical about 72 percent of the time. Price changes are not synchronized but have similar frequencies and average sizes. These results have implications for national statistical offices, researchers using online data, and anyone interested in the effect of the Internet on retail prices. (*JEL* D22, L11, L81, O14)

Billion price project

Motiviation Webscraping

Decriminalizing Indoor Prostitution: Implications for Sexual Violence and Public Health

SCOTT CUNNINGHAM
Baylor University
and
MANISHA SHAH
University of California, Los Angeles & NBER

First version received November 2015*; Editorial decision August* 2017*; Accepted November* 2017 (*Eds.*)

Most governments in the world, including the U.S., prohibit sex work. Given these types of laws rarely change and are fairly uniform across regions, our knowledge about the impact of decriminalizing sex work is largely conjectural. We exploit the fact that a Rhode Island District Court judge unexpectedly decriminalized indoor sex work to provide causal estimates of the impact of decriminalization on the composition of the sex market, reported rape offences, and sexually transmitted infections. While decriminalization increases the size of the indoor sex market, reported rape offences fall by 30% and female gonorrhoea incidence declines by over 40%.

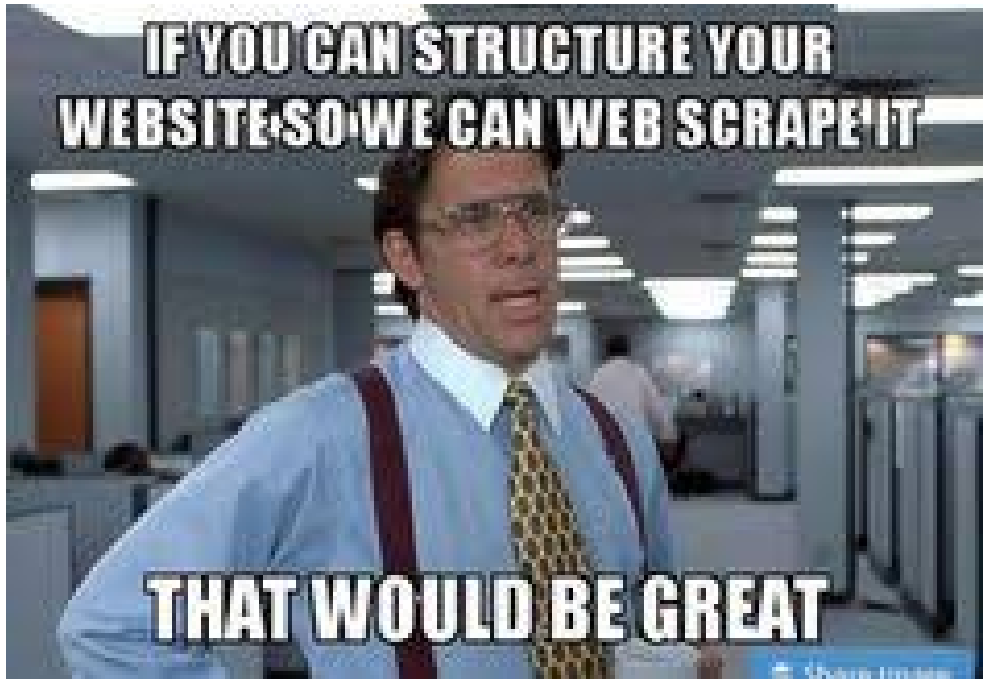
Key words: Regulation, Sex work, Public health, Crime.

JEL Codes: I18, J4, K42

Motiviation Webscraping

We also harvest data from an online review site called The Erotic Review. TER, a reputation website similar to Yelp.com, is one of the largest sex websites in the country and only covers indoor sex workers. Customers use it primarily to provide feedback on transactions with sex workers in a particular area. We collect approximately 90,000 records from TER database from 1999 to 2007 from all over the country. We identify Rhode Island-based sex workers by using phone number area codes. We primarily use the data to focus on the types of services provided, transaction prices, and provider race.

Webscraping basics



Webscraping basics

- How to get data, or ”content”, off the web and onto our computers.
- If you see it in your browser it exists somewhere
- To be “successful” one must have a working knowledge on:
  - how web pages display content (Hyper Text Markup Language or HTML)
  - where is the content “located”
    1. Server side
    2. Client side
  - The good news is that both server-side and client-side websites allow for web scraping

Caveat: ethical and legal limitations

- Just because you *can* scrape it, doesn't mean you *should*.
- Check `The Robots Exclusion Protocol` of a website, adding ``/robots.txt'' to the website's URL
  1. User-agent: the type of robots to which the section applies
  2. Disallow: directories/prefixes of the website not allowed to robots
  3. Allow: sections of the website allowed to robots
- `robots.txt` is de facto standard (see `http://www.robotstxt.org`)
- Also always check the terms and conditions and what they say about scraping
- Remember the immortal words of uncle Ben: "with great power comes great responsibility"

Server-side

- The website is ”static”, all the info is located in the HTML code that the host server sends
  - E.g. Wikipedia tables are already populated with all of the information - tables, numbers, dates, etc. - that we see in our browser.
- Challenges:
  - Finding the correct path CSS (or Xpath) ”selectors”.
  - Navigating dynamic webpages (e.g. ”Next page” and ”Show More” tabs).

Some useful tools

- CSS selectors:
  - SelectorGadget for Chrome
  - ScrapeMate for Firefox
  - Inspect Element
- Browsers: anything but explorer

Client-side

- The website contains an empty template of HTML and CSS.
  - E.g. It might contain a ”skeleton” table without any values.
- However, when we actually visit the page URL, our browser sends a *request* to the host server.
- If request is valid, then the server sends a response script, which our browser executes and uses to populate the HTML template with the specific information that we want.
- Challenges: Finding the ”API endpoints” can be tricky, since these are sometimes hidden from view.

Demo

photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/

Review & Next Steps

- Computation
- QR decomposition
- MapReduce and Spark
- Demo `Scraping`
- Message: web scraping involves as much art as it does science

- **Next Class:** MLE, Bayesian Stats.

- Questions? Questions about software?

Further Readings

- Constantine, P. G., & Gleich, D. F. (2011, June). Tall and skinny QR factorizations in MapReduce architectures. In Proceedings of the second international workshop on MapReduce and its applications (pp. 43-50).
- Dean, J., & Ghemawat, S. (2004). MapReduce: Simplified data processing on large clusters.
- Van Loan, C. F., Golub, G. H. (2012). Matrix Computations. United States: Johns Hopkins University Press.
- `Webscraping` tutorial from Prof. Grant McDermott.
- Web Scrapping slides from Fernandez Villaverde J., Guerrón P. & Zarruk Valencia, D.
- Wickham, H., & Wickham, M. H. (2016). Package 'rvest'. `https://cran.r-project.org/web/packages/rvest/rvest.pdf`.