3. Large Structured Data

Data Science for Economists — Summer 2025

Julian Hinz

Bielefeld University

Overview of topics

Trade data: Large data

- Firm level data
- Fitting PL distributions

Technical questions:

- Memory management with R
- Good practices with large data

FIRM TRADE DATA

“Countries don’t trade. Firms trade.”
Hallak and Levinsohn, 2005

Traditional trade theories

1. H-O model: countries trade because of different factor endowments
2. Ricardian model: countries trade because of different technologies

Data availability from late 1970s and early 1980s, provide evidence of unexpl. facts:

1. similar countries trade extensively
2. Intra industry trade is prominent:
   - Japan exports Toyota vehicles to Germany and imports Mercedes-Benz automobiles from Germany

Traditional trade theories

1. H-O model: countries trade because of different factor endowments
2. Ricardian model: countries trade because of different technologies

Data availability from late 1970s and early 1980s, provide evidence of unexpl. facts:

1. similar countries trade extensively
2. Intra industry trade is prominent:
   - Japan exports Toyota vehicles to Germany and imports Mercedes-Benz automobiles from Germany

Traditional trade theories

1. H-O model: countries trade because of different factor endowments
2. Ricardian model: countries trade because of different technologies

Data availability from late 1970s and early 1980s, provide evidence of unexpl. facts:

1. similar countries trade extensively
2. Intra industry trade is prominent:
   - Japan exports Toyota vehicles to Germany and imports Mercedes-Benz automobiles from Germany

Traditional trade theories

1. H-O model: countries trade because of different factor endowments
2. Ricardian model: countries trade because of different technologies

Data availability from late 1970s and early 1980s, provide evidence of unexpl. facts:

1. similar countries trade extensively
2. Intra industry trade is prominent:
   - Japan exports Toyota vehicles to Germany and imports Mercedes-Benz automobiles from Germany

Intra-industry Trade

Table VI.1. **Manufacturing intra-industry trade as a percentage of total manufacturing trade**

	1988-91	1992-95	1996-2000	Change
High and increasing intra-industry trade				
Czech Republic	n.a.	66.3	77.4	11.1
Slovak Republic	n.a.	69.8	76.0	6.2
Mexico	62.5	74.4	73.4	10.9
Hungary	54.9	64.3	72.1	17.2
Germany	67.1	72.0	72.0	5.0
United States	63.5	65.3	68.5	5.0
Poland	56.4	61.7	62.6	6.2
Portugal	52.4	56.3	61.3	8.9

OECD 2002

Firms in the New Trade Theories

Toyota and Mercedes-Benz offer different varieties of the same good

Krugman (1979 and 1980) introduces:

- **Monopolistic Competition:** firms produce differentiated products, this differentiation allows for a love of variety by consumers.
- **Economies of Scale:** Production under economies of scale permits firms to produce a wide variety of goods more cost effectively, leading to an increase in international trade.

Firms in the New Trade Theories

Implications

- The love of variety leads to increased trade volumes, with countries importing many different types of goods rather than producing them domestically.
- all firms participate in exporting

How frequent is exporting?

Exporting and Importing by U.S. Manufacturing Firms, 1997

NAICS industry	*Percent of all firms*	*Percent of firms that export*	*Percent of firms that import*	*Percent of firms that import & export*
311 Food Manufacturing	7	17	10	7
312 Beverage and Tobacco Product	1	28	19	13
313 Textile Mills	1	47	31	24
314 Textile Product Mills	2	19	13	9
315 Apparel Manufacturing	6	16	15	9
316 Leather and Allied Product	0	43	43	30
321 Wood Product Manufacturing	5	15	5	3
322 Paper Manufacturing	1	42	18	15
323 Printing and Related Support	13	10	3	2
324 Petroleum and Coal Products	0	32	17	14
325 Chemical Manufacturing	3	56	30	26
326 Plastics and Rubber Products	5	42	20	16
327 Nonmetallic Mineral Product	4	16	11	7
331 Primary Metal Manufacturing	1	51	23	21
332 Fabricated Metal Product	20	21	8	6
333 Machinery Manufacturing	9	47	22	19
334 Computer and Electronic Product	4	65	40	37
335 Electrical Equipment, Appliance	2	58	35	30
336 Transportation Equipment	3	40	22	18
337 Furniture and Related Product	6	13	8	5
339 Miscellaneous Manufacturing	7	31	19	15
Aggregate manufacturing	**100**	**27**	**14**	**11**

Sources: Data are for 1997 and are for firms that appear in both the U.S. Census of Manufactures and the Linked-Longitudinal Firm Trade Transaction Database (LFTTD).
Notes: The first column of numbers summarizes the distribution of manufacturing firms across three-digit NAICS industries. Remaining columns report the percent of firms in each industry that export, import, and do both.

Bernard et al. (2007)

Stylized facts on exporters

Increasing availability since the 90s of firms/plants level data, showed:

- Exporting is extremely rare.
- Exporters are different than non exporters:
  - They are larger.
  - They are more productive.
  - They use factors differently.
  - They pay higher wages.
- Even among exporters a large heterogeneity persists...

Melitz Model

- Firm-level Productivity Heterogeneity
- Firm Selection and Market Entry
  - Selection mechanism based on productivity differences and fixed costs
  - Higher productivity firms more likely to enter/export
- Trade Liberalization and Trade Patterns:
  - Model predicts increase in exports after liberalization
  - More competition makes less productive firms exit, increasing average productivity
  - Trade flows driven by firm-level productivity differences
- Key Implications:
  - Firm dynamics and selection impact aggregate productivity
  - Market structure influenced by trade liberalization
  - Trade patterns driven by firm-level productivity heterogeneity

Trade theories vs stylized facts

Facts	"Old" trade theory	"New" trade theory	Heterogeneous firms model
	Ricardo (1817), Heckscher (1919), Ohlin (1933)	Krugman (1980)	Melitz (2003), Bernard et al. (2003)
Trade			
Interindustry trade	Yes	No	No
Intra-industry trade	No	Yes	Yes
Exporters and nonexporters within industries	No	Yes	Yes
Trade and productivity			
Exporters are more productive than nonexporters within industries	No	No	Yes

Facts	"Old" trade theory	"New" trade theory	Heterogeneous firms model
Trade and labor markets			
Net changes in employment across industries following trade liberalization	Yes	No	No
Simultaneous gross job creation and destruction within industries following trade liberalization	No	Yes	Yes
Trade liberalization affects relative factor rewards (income distribution)	Yes	No	No

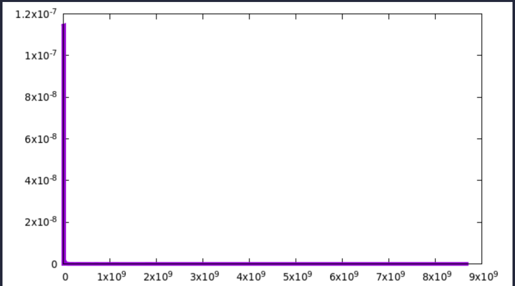
Firm heterogeneity

In New Trade Theory Trade patterns and welfare effects are driven by firm-level productivity heterogeneity. The Pareto Distribution has nice features:

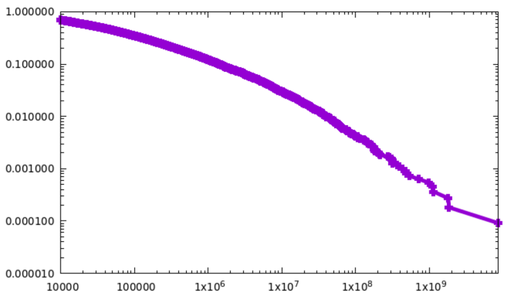
- Analytical Simplicity:
  - Closed-form CDF and PDF for easier mathematical manipulations.
- Policy Analysis and Comparative Statics:
  - Facilitates analysis of trade policy changes and their effects.
- Trade Elasticities and General Equilibrium Effects (Chaney 2008)
  - Derivation of extensive and intensive margin elasticities.
  - Provides insights into general equilibrium effects and distributional consequences.

Is this assumption on firm heterogeneity a good approximation of what we observe in real world?

Colombian firms' export value



Arithmetic scale



Log-log scale

A matter of scales

1. Arithmetic scale: histogram is highly right-skewed
    - the bulk of the distribution occurs for fairly small size (in terms of export value) but there's a small number of firms with a much higher than the typical value (origin of the long-tail)
2. Log-log scale: if we replot the same histogram with logarithmic horizontal and vertical axes the histogram follows quite closely a straight line.

What does it mean?

Power Laws

Let us define $p(x)dx$ as the fraction of firms with export value between x and $x + dx$. Then observing a straight line in a log-log scale means

$$\log p(x) = c - (\alpha + 1) \log x$$

where c and $-(\alpha + 1)$ represent the intercept and the slope of the line. If we take exponential on both sides we get

$$p(x) = Cx^{-(\alpha+1)}$$

Probability distributions with this functional form are said to follow a power law and $-(\alpha + 1)$ is the exponent of the PL

Power Laws

Let us define $p(x)dx$ as the fraction of firms with export value between x and $x + dx$. Then observing a straight line in a log-log scale means

$$\log p(x) = c - (\alpha + 1) \log x$$

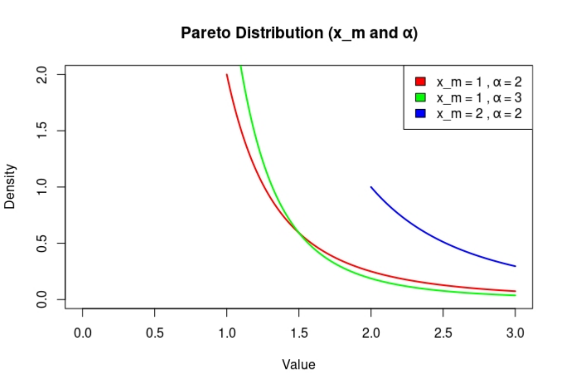
where c and $-(\alpha + 1)$ represent the intercept and the slope of the line. If we take exponential on both sides we get

$$p(x) = Cx^{-(\alpha+1)}$$

Probability distributions with this functional form are said to follow a power law and $-(\alpha + 1)$ is the exponent of the PL

Understanding Pareto Distribution Parameters

- **Scale Parameter (x_m:**
  - Minimum possible value or 'location' parameter.
  - Distribution begins at x_m and extends to infinity.
  - Must be a positive real number (x_m > 0).
- **Shape Parameter (α):**
  - Also known as the Pareto Index or 'shape' parameter.
  - Determines the shape of the distribution curve, particularly the 'tail'.
  - Must be a positive real number (α > 0).



```r
# Parameters
x_m_values <- c(1, 1, 2) # scale parameters
alpha_values <- c(2, 3, 2) # shape parameters
colors <- c("red", "green", "blue") # colors for different distributions

# Prepare plot
plot(0, 0, type="n", xlim=c(0, 3), ylim=c(0, 1),
     xlab="Value", ylab="Density",
     main="Pareto Distribution (x_m and α)")

# Loop over parameters
for (i in 1:length(x_m_values)) {
  x_m <- x_m_values[i]
  alpha <- alpha_values[i]

  # Generate distribution
  x_values <- seq(x_m, 3, by = 0.01)
  y_values <- dpareto(x_values, scale = x_m, shape = alpha)

  # Add to plot
  lines(x_values, y_values, col=colors[i], lwd=2)
}

# Add legend
legend("topright", legend=paste("x_m =", x_m_values, ", α =", alpha_values), fill=colors)
```

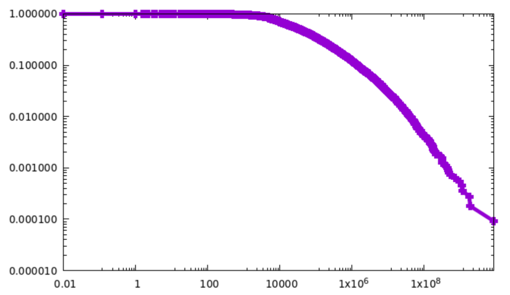
Estimating the parameters of a PL

Typically 3 methods to estimate Power law exponent from empirical data:

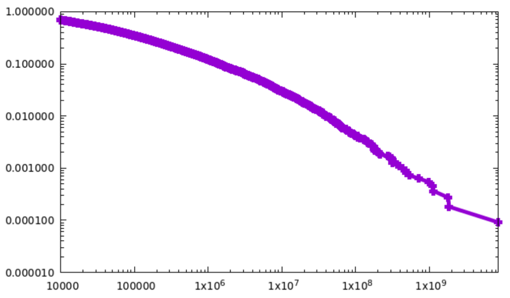
1. linear fit of the log-log plot of the empirical density (binned histogram);
2. linear fit of the log-log plot of the CCDF or rank-size;
3. maximum likelihood (ML).

Remarks. These estimation procedures are typically applied above a given threshold value

Full vs top sample



Full Sample



Only top 50%

Method 1 - Binning

Most used density estimator is the histogram, an estimate of the density formed by splitting the range of a variable X into equally spaced intervals and calculating the fraction of the sample in each interval.

Method 1 - Binning

Practically to build an histogram one has to set:

1. origin: x_0
2. width: h
3. bins: defined as $[x_0 + m \times h, x_0 + (m+1)h]$

where m can be positive integers.

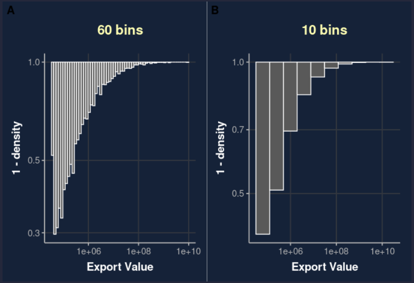
Given a sample $\{x_i, i = 1, ..., n\}$ the histogram $\hat{f}(x)$ is then defined as

$$\hat{f}(x) = \frac{1}{nh}(\#\text{of } x_i \text{ in the same bin as x}) \tag{1}$$

Method 1 - Practical Corner

Which bin width? The smaller the width, the more the # of bins

- the better the resolution of the frequency distribution
- the worse accuracy with which each value of f(x) is estimated



Method 2 - CCDF

An alternative (and more convenient) method to visualize and detect a PL behaviour is to plot the complementary cumulative distribution function (CCDF) on log-log scales.

Method 2 - CCDF

The CCDF P(x) is the fraction of firms that have export value equal or greater than x:

$$P(x) = \sum_{x_i \geq x} p(x_i) \tag{2}$$

Notice that, if p(x) = $Cx^{-(\alpha+1)}$ and $\alpha>2$, then:

$$P(x) = \sum_{x_i \geq x} p(x_i) = C \sum_{x_i \geq x} x^{-(\alpha+1)} \simeq C \int_x^\infty x^{-(\alpha+1)}\, \mathrm{d}x = \frac{C}{-\alpha} x^{-\alpha} \tag{3}$$

$\rightarrow p(x) \sim PL(-(\alpha+1))$ then the CCDF of the distribution $P(x) \sim PL(-\alpha)$

Hence, when plotted on log-log scales, the CCDF of a power law should appear as a straight line.

Method 2 - Practical Corner

The CCDF in a given point x is typically estimated as

$$P(x) = \frac{\#obs(x_i \geq x)}{n}$$

where we do not need any binning. If one observes that $P(x) \sim Cx^{-(\alpha)}$ then it should be reasonable to use OLS in

$$logP(x_i) = c-(\alpha)\,log(x_i) + \epsilon_i \tag{4}$$

Remark. x_i should be iid, not the case if we order to estimate the CCDF

Method 3 - Maximum Likelihood Estimation

As usual the statistical properties of a ML estimator depends on the validity of the underlying assumptions.

- if the true distribution of X is a Power law, the estimator performs quite well and it is not very sensitive to the sub-samples used for the estimates;

MLE for Pareto Distribution

- Likelihood Function:
  - Represents the probability of observing the data.
  - For i.i.d. observations $x_1, x_2, ..., x_n$ from a Pareto distribution:

$$L(\alpha, x_m) = f(x_1; \alpha, x_m) \cdot f(x_2; \alpha, x_m) \cdot ... \cdot f(x_n; \alpha, x_m)$$

- Log-Likelihood Function:
  - Simplifies calculations and improves numerical stability.
  - Take the natural logarithm of the likelihood function:

$$\log L(\alpha, x_m) = \log f(x_1; \alpha, x_m) + \log f(x_2; \alpha, x_m) + ... + \log f(x_n; \alpha, x_m)$$

- Maximizing the Log-Likelihood:
  - Numerical optimization techniques (e.g., gradient-based methods, Newton-Raphson) are used to find the maximum of the log-likelihood function.

MLE for Pareto Distribution

- Estimating Coefficients:
  - The estimated values of the shape parameter (α) and the minimum value (x_m) are the maximum likelihood estimates for the Pareto distribution.
- Model Evaluation:
  - Assess the goodness of fit using statistical tests and visual comparisons (Q-Q plots, histograms) between the observed data and the estimated distribution.

References

Newman, M.E.J., 2005. Power laws, Pareto distributions and Zipf's law. Contemporary Physics, 46, pp. 323-351.

How large can be large?

Data that is processed in R has to be fully loaded into the RAM

$\rightarrow$ max size of data that you can process depends on the amount of free RAM available:

- Rule of thumb: free RAM = 2-3 $\times$ size of data

How much free RAM do I have?

- with WIN Powershell or CMD: ”C: systeminfo | find ”Available Physical Memory
- on Mac: ”top”
- on Linux: ”free -h”

Object size

```
object.size(my_data)
```

```
> chr_vect <- c("12","11","33")
> dbl_vect <- c(12,11,33)
> format(object.size(chr_vect), units = "auto")
[1] "248 bytes"
> format(object.size(dbl_vect), units = "auto")
[1] "80 bytes"
> memory.profile()
       NULL      symbol    pairlist     closure environment     promise    language     special     builtin        char
          1       37875     1250989       24936        7117       34670      338840          45         685      122872
    logical     integer      double     complex   character         ...         any        list  expression    bytecode
      36574      196015       18644          40      353876          11           0       91695           1       76205
externalptr     weakref         raw          S4
       7455        2137        2187        3573
```

Factors vs Characters

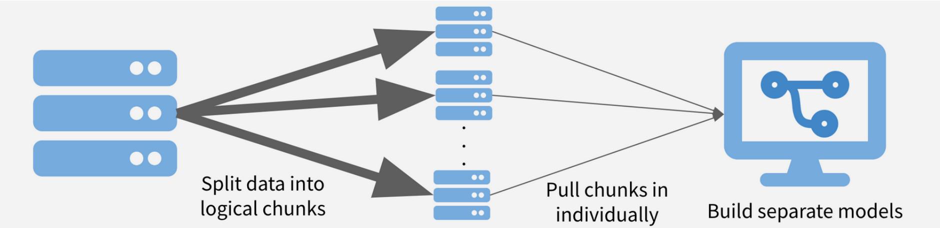
Encode variables efficiently (e.g., factor instead of character);

```
> gender <- c("female", "male", "other")
> format(object.size(gender), units = "auto")
[1] "272 bytes"
> format(object.size(as.factor(gender)), units = "auto")
[1] "672 bytes"
> gender <- rep(c("female", "male", "other"), 100)
> format(object.size(gender), units = "auto")
[1] "2.6 Kb"
> format(object.size(as.factor(gender)), units = "auto")
[1] "1.8 Kb"
```

A few other memory-management tips in R:

1. Sessions continue – and memory is occupied – until you log out.
2. Manage sessions efficiently by tidying the R session workspace:
   - Load only the data you need;
   - Remove redundant dataframe columns: dataframe$redundant <-NULL;
   - Remove `rm()` data objects from the workspace once you don't need them.
   - Force Garbage Collection `gc()` in loops (automatic gc is enough most of the time)

Chunk and Pull



Chunk and Pull - Example

Download the Siren Data and store in project folder

- Siren is the French firm tax identifier
- Siren Data contains the stock of all French firms
- both active and inactive firms (since 1973), i.e. $>$ 23M firms

```
$ curl -O https://files.data.gouv.fr/insee-sirene/StockEtablissement_utf8.zip
$ unzip StockEtablissement_utf8.zip
$ zcat StockUniteLegale_utf8.csv | head -n 3
siren,statutDiffusionUniteLegale,unitePurgeeUniteLegale,dateCreationUniteLegale,sigleUniteLegale,sexeUniteLegale ..
000325175,O,,2000-09-26,,M,THIERRY, ...
001807254,O,,1972-05-01,,M,JACQUES-LUCIEN, ...

$ zcat StockUniteLegale_utf8.csv.gz | wc -l
23065462
```

Chunck and Pull in 3 Steps

1. split the data by year using the shell into smaller ”chucks”
    - use AWK for this, for Windows using WSL
    - AWK is compiled rather than interpreted language
2. write a function in R that compute the share of firms founded by a woman
3. pull together the output of each year to get a time series

Chunk and Pull - Step 1

1.A) Split the data into chunks, by year in which the company was founded (column $4) preserve info about the firm identifier ($1) and gender of the founder ($6)

```
#!/bin/sh
wd=temp
fname=StockUniteLegale_utf8.csv.gz
for year in {1990..2022}; do
    rm -rf ${wd}/yearly_data/SIREN_${year}.csv.gz
    echo "Working on year $year"
    echo "siren, gender_founder" >  ${wd}/yearly_data/SIREN_${year}.csv
    zcat $fname | awk -F ',' "{if(substr(\$4, 1, 4)==${year}) print \$1\",\"\$6}" >> ${wd}/yearly_data/SIREN_${year}.csv
    gzip -f  ${wd}/yearly_data/SIREN_${year}.csv
done
```

1.B) save the above in chunk_siren.sh and in the shell run `bash chunk_siren.sh`

Chunk and Pull - Step 1

```
$ ls temp/yearly_data
SIREN_1990.csv.gz  SIREN_1993.csv.gz  SIREN_1996.csv.gz  SIREN_1999.csv.gz
SIREN_2002.csv.gz  SIREN_2005.csv.gz  SIREN_2008.csv.gz  SIREN_2011.csv.gz
SIREN_2014.csv.gz  SIREN_2017.csv.gz  SIREN_2020.csv.gz SIREN_1991.csv.gz
SIREN_1994.csv.gz  SIREN_1997.csv.gz  SIREN_2000.csv.gz  SIREN_2003.csv.gz
SIREN_2006.csv.gz  SIREN_2009.csv.gz  SIREN_2012.csv.gz  SIREN_2015.csv.gz
SIREN_2018.csv.gz  SIREN_2021.csv.gz SIREN_1992.csv.gz  SIREN_1995.csv.gz
SIREN_1998.csv.gz  SIREN_2001.csv.gz  SIREN_2004.csv.gz  SIREN_2007.csv.gz
SIREN_2010.csv.gz  SIREN_2013.csv.gz  SIREN_2016.csv.gz  SIREN_2019.csv.gz
SIREN_2022.csv.gz
```

Chunk and Pull - Step 2

```r
compute_share_F <- function(dt) {
  year_dt <- as.numeric(gsub(".*?([0-9]+).*", "\\1", dt))
  print(paste0("Working on year ", year_dt, ""))
  read_csv(dt) %>% group_by(gender_founder) %>% mutate(freq=n()) %>%
    select(gender_founder, freq) %>% distinct() %>% mutate(year=year_dt) %>%
    spread(key=gender_founder, value=freq) %>% mutate(F_share=F/(F+M)) %>%
    select(F_share, year) %>%
    as.data.frame() %>% return()
}
```

Chunk and Pull - Step 3

```r
my_files <- list.files("~/Downloads/temp/yearly_data", full.names = TRUE)
pull_data <- map_df(my_files, compute_share_F)
```

Chunk and Pull - Output

```
pull_data %>% filter(!is.na(gender_founder)) %>%
  spread(key=gender_founder, value=freq) %>% mutate(F_share=F/(F+M)) %>%
  select(F_share, year)

     F_share year
1  0.3615124 1990
2  0.3656392 1991
3  0.3697042 1992
4  0.3677663 1993
...
29 0.4234902 2018
30 0.4144693 2019
31 0.4030988 2020
32 0.4207230 2021
```