The production process of the Global MPI

Nicolai Suppa

Stata UK conference (virtual)
London, UK
September 2021

CED
Centre d'Estudis Demogràfics

Outline

Motivation

Well-devised **workflow is vital** for any large-scale project.

Why sharing?

1. transparency and replication: how is the global MPI computed?
2. show & discuss workflow-related problems & solutions
3. share some experience and lessons & how to refine this process?

➜ aspects of the present workflow may be **relevant in other settings**, e.g.,

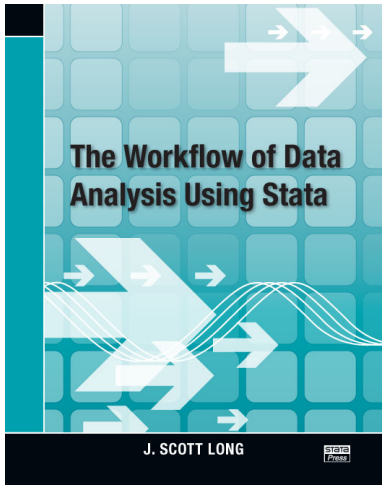
- other cross-country projects
- projects juggling with a plethora of estimates
- large scale projects where 'tiny' coding tweaks make a difference

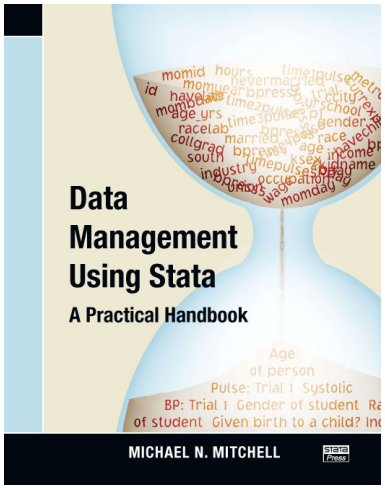
➜ general workflow questions receive **rather little attention**

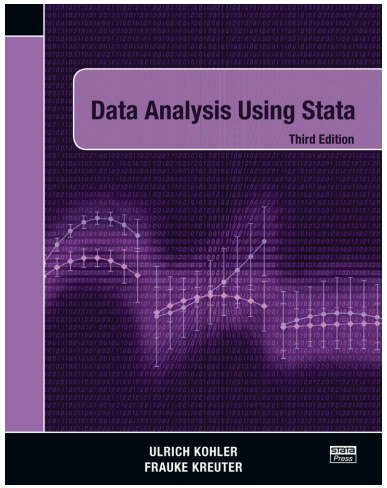
- hard to de-contextualise (typically project-specific)
- work-flow decisions may not be recognised as such
- alternative solutions make no real difference in practice

Related literature

Programming and workflows in Stata







The Stata Journal (2005)
5, Number 4, pp. 560–566

Suggestions on Stata programming style

Nicholas J. Cox
Durham University, UK
n.j.cox@durham.ac.uk

Abstract. Various suggestions are made on Stata programming style, under the headings of presentation, helpful Stata features, respect for datasets, speed and efficiency, reminders, and style in the large.

Keywords: pr0018, Stata language, programming style

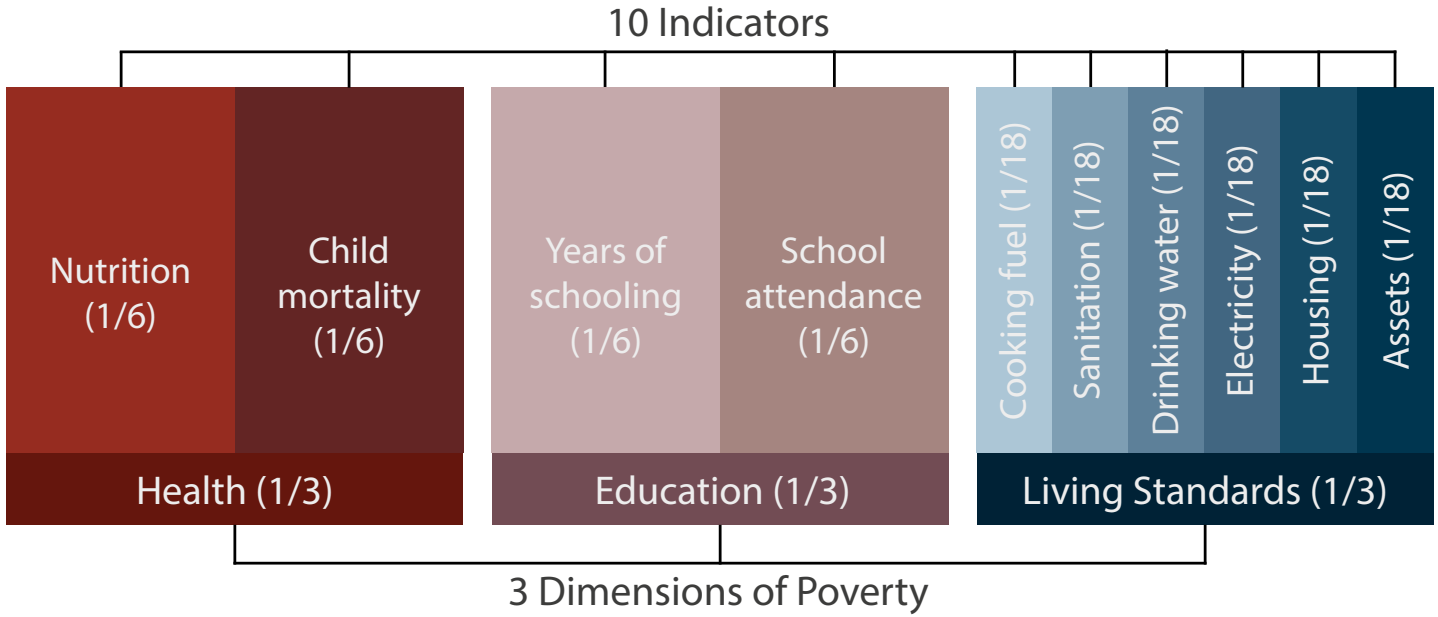
- lots of good advice!
- many workflow related problems live somewhere between general advice, best practices, and specific coding problems.

What is the global MPI?

➜ it is an **international measure of multidimensional poverty.**

Alkire and Foster (2011); Sen (1992); Alkire and Santos (2014); Alkire et al. (2020)

- available for 100+ countries (and 1200+ sub-national regions)
- developed and published by OPHI and UNDP (since 2010)



- two release types: 'global MPI' (CME) and 'changes over time' (COT); (time-harmonized indicators)

The global MPI

Computational aspects

- all figures are obtained from a **single survey** per country
- **numerous measures** are calculated for each country
  - headcount, intensity, adj. headcount, (un-) censored headcounts,...
- most numbers can be **disaggregated** by area, region, and age group
- parametric choices require **sensitivity checks** (e.g., weights, cutoffs)

➜ N: 5k–2.7m with $N_{med} \approx$ 50k; ➜ # of estimates $\approx$ 170k

Changes over time:

- 2–3 years for 80+ countries
- new type of estimate ('change')
- region variable may differ between release types (harmonization)

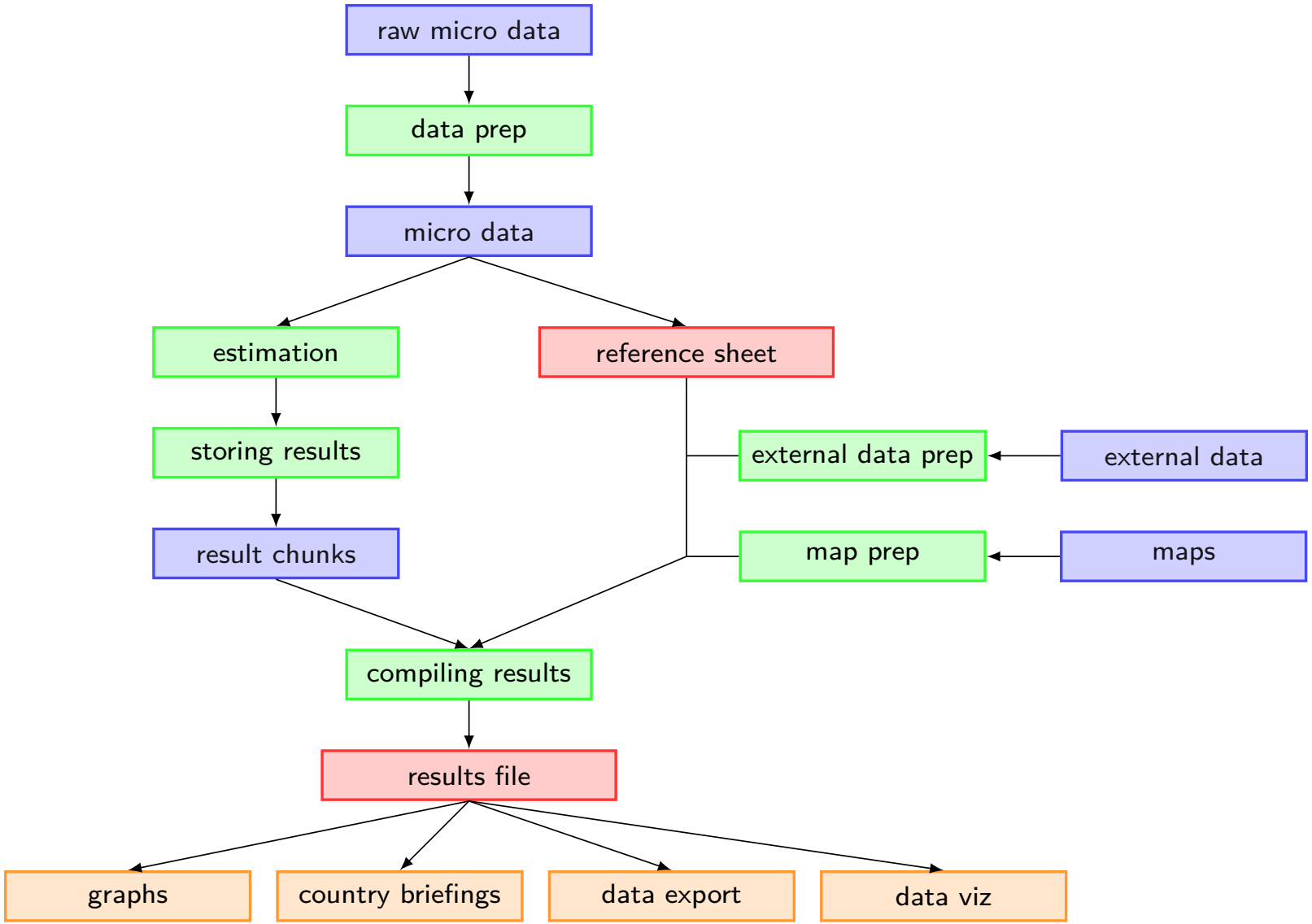
➜ # of level estimates $\approx$ 200k; # of change estimates $\approx$ 100k.

Desiderata

The 2018 revision

1. improve **efficiency** in general
   - estimation time and storage
2. ensure **replicability** and tractability
   - track down and fix errors
3. achieve **flexibility**
   - re-estimate selected countries or measures
4. low **maintenance costs**
   - Stata skills & feasible revisions
5. develop a more **widely applicable approach** to MPI-estimation and facilitate the provision of certain numbers (e.g., disaggregations, SE)
6. integrate estimation of **changes over time** into work flow (2021)

The basic workflow



The results file

Principle structure

- each estimate is an observation
- each estimate can be uniquely identified using auxiliary variables

  e.g., `ccty`, `measure`, `k`, `wgts`, `loa`, `indicator`, ...

```
. li ccty y sur loa measure b k wgts indi sp if inlist(k,33,.) & ccty == "IND" , noob sepby(k)
```

ccty	year	survey	loa	measure	b	k	wgts	indica~r	spec
IND	2015-2016	DHS	nat	A	43.94929	33	equal		GMPI
IND	2015-2016	DHS	nat	H	27.90772	33	equal		GMPI
IND	2015-2016	DHS	nat	M0	.1226525	33	equal		GMPI
IND	2015-2016	DHS	nat	hd	37.59741	.	.	d_nutr	GMPI
IND	2015-2016	DHS	nat	hd	2.68655	.	.	d_cm	GMPI
IND	2015-2016	DHS	nat	hd	13.86739	.	.	d_educ	GMPI
IND	2015-2016	DHS	nat	hd	6.396227	.	.	d_satt	GMPI
IND	2015-2016	DHS	nat	hd	58.47132	.	.	d_ckfl	GMPI
IND	2015-2016	DHS	nat	hd	51.96471	.	.	d_sani	GMPI
IND	2015-2016	DHS	nat	hd	14.59562	.	.	d_wtr	GMPI
IND	2015-2016	DHS	nat	hd	12.15246	.	.	d_elct	GMPI
IND	2015-2016	DHS	nat	hd	45.64144	.	.	d_hsg	GMPI
IND	2015-2016	DHS	nat	hd	13.9671	.	.	d_asst	GMPI

Advantages: single file, easy to explore and to extend

The reference sheet

- contains **survey-constant** information (country & region)
  - reduces data carried through estimation
  - allows parallel processing (estimation vs map prep)
  - simplifies some quality checks
  - facilitates running code selectively

```
. li ccty survey year cty region* fname *date if ccty == "BGD" , noob sep(0)
```

ccty	survey	year	cty	region	region_n~e	fname	fdate	adate
BGD	MICS	2019	Bangladesh	1	Barishal	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	2	Chattogram	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	3	Dhaka	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	4	Khulna	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	5	Mymensingh	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	6	Rajshahi	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	7	Rangpur	bgd_mics19.dta	29 Jun 2020	7 Jul 2020
BGD	MICS	2019	Bangladesh	8	Sylhet	bgd_mics19.dta	29 Jun 2020	7 Jul 2020

Tool: `refsh`

```
refsh using path2refsh, rebuild char(ccty survey year) ///
      id(ccty) region(region) path(path2microdata)
```

Estimation and storing

Options

1. Using (i) eststo, estadd, estwrite, estread (Jann, 2005, 2007)

```
eststo H‘k’_‘subg’: svy: mean I_‘k’ , over(‘subg’)
estadd loc measure "H" ...
```

(ii) `_coef_table` and (iii) `xsvmat` (Roger Newson)

```
. matlist r(table)'
```

	b	se	t	pvalue	ll	ul	df	crit	eform
d_cm	.0190303	.0007211	26.39225	1.3e-138	.0176165	.0204441	3092	1.960732	0

2. Using `frame post` (Stata 16)

```
svy: mean I_‘k’ , over(‘subg’)
...
frame post myframe (expr) (expr) ("H") (‘k’) ...
```

3. Using `collect`? (Stata 17)

Estimation and storing

The packaged approach

Tool: mpitb set, mpitb est

```
mpitb set, d1(d_cm d_nutr, name(hl)) d2(d_satt d_educ, ///
  name(ed)) d3(d_elct d_sani d_wtr d_hsg d_asst d_ckfl, ///
  name(ls)) name(gmpi_cme)

mpitb est , svy w(equal) n(gmpi_cme) me(all) aux(all) ///
  measuresdim(all) k(1 20 33 50) ts addmeta(ccty=‘cty’) ///
  levelsa(results/dta/ctys/‘cty’_main , replace)
```

Tools

- gafvars, mpi_setwgts, genwgts, addmetainfo,...

Result chunks

Single mega loop is dysfunctional!

➜ need for a **cache** of previous estimates

(i) collect several estimates and save them in convenient **result chunks** to disk (e.g., along `ccty`, `loa`).

e.g., `BGD_main.dta`, `BGD_aux.dta`, `BGD_region.dta` ...

(ii) use dedicated folder for his (`results/dta/ctys`)

(iii) compile results: append all files found in that folder

```
clear
save results/dta/results_raw , replace emptyok
loc flist : dir "results/dta/ctys/" files "*.dta"
foreach f in `flist' {
  append using results/dta/ctys/`f' , nol
}
```

➜ need for convenient **control of loop** over countries

The main do-file

- **linearized workflow**: all other code can be run from here
- mainly for **interactive use** (re-estimation from scratch, too)
- sections
  1. reference sheet production
  2. certification scripts (microdata)
  3. estimation
  4. performance analysis
  5. compiling raw result files
  6. quality checks
  7. external data and map prep
  8. assemble results file
  9. deliverables: graphs, spreadsheets, country briefing, ...

Tool: `ctyselect` ➜ returns country codes in `r(ctylist)`

```
frame refsh : ctyselect ccty
frame refsh : ctyselect ccty, r(^A)
frame refsh : ctyselect ccty, s(BGD IND)
```

Certification scripts (for microdata)

Objective

- identify common sources for loop breaks (or worse) early on
- fail early & loud; reduce code complexity; easy to modify & extend

Application: cleaned microdata (possibly selective)

- variables are existing and numeric ... `conf numeric v ‘v’...`
- variables have valid values ... `assert inlist(‘v’,0,1)if !mi(‘v’)...`
- variables are not entirely missing

```
... qui count if !mi(‘v’)
    if ‘r(N)’ == 0 { ...
```

- data characteristics are not empty ... `assert "‘_dta[‘c’]’"!= ""...`

COT requires tests across datasets of a country:

- region coding plausible? missing indicators consistent?

Quality checks

Quality checks are implemented in **various stages**.

Automation may save lots of time, but manual screening remains essential.

1. Cross-check between different sources:
   (a) regular estimation vs rudimentary estimation in data prep
   (b) upcoming vs previous releases (where comparable)
   - Valuable commands in this context:
     - `assert float(b)== float(b_dp)if !mi(b_dp)`
     - `gen diff = abs(b - b_dp)> 1e-07 if !mi(b_dp)`

2. Timestamps

```
li ccty measure k b time* if k == 33 & loa == "nat" & ccty == "IND" , noob
```

ccty	measure	k	b	time	timedata
IND	H	33	27.90772	28 Apr 2020 02:48	27 Apr 2020 10:55

   - useful: `sum time* , f`

Graph and country brief production

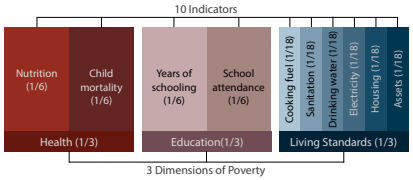
Oxford Poverty and Human Development Initiative (OPHI)
Oxford Department of International Development
Queen Elizabeth House, University of Oxford
www.ophi.org.uk

Global MPI Country Briefing 2018: India (South Asia)

The Global MPI

The global Multidimensional Poverty Index (MPI) was created using the multidimensional measurement method of Alkire and Foster (AF).[1] The global MPI is an index of acute multidimensional poverty that covers over 100 countries. It is computed using data from the most recent Demographic and Health Surveys (DHS), Multiple Indicator Cluster Surveys (MICS), Pan Arab Project for Family Health (PAPFAM) and national surveys. The MPI has three dimensions and 10 indicators as illustrated in figure 1. Each dimension is equally weighted, and each indicator within a dimension is also equally weighted.[2] Any person who fails to meet the deprivation cutoff is identified as deprived in that indicator. So the core information the MPI uses is the profile of deprivations each person experiences. Each deprivation indicator is defined in table A.1 of the appendix.

Figure 1. Structure of the Global MPI



In the global MPI, a person is identified as multidimensionally poor or MPI poor if they are deprived in at least one third of the weighted MPI indicators. In other words, a person is MPI poor if the person's weighted deprivation score is equal to or higher than the poverty cutoff of 33.33%. Following the AF methodology, the MPI is calculated by multiplying the **incidence** of poverty (H) and the average **intensity** of poverty (A). More specifically, H is the proportion of the population that is multidimensionally poor, while A is the average proportion of dimensions in which poor people are deprived. So, $MPI = H \times A$, reflecting both the share of people in poverty and the degree to which they are deprived.

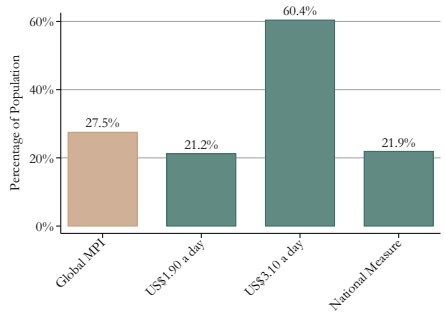
Table 1. Global MPI in India

Area	*MPI*	*H*	*A*	Vulnerable	Severe Poverty	Population Share
National	0.121	27.5%	43.9%	19.1%	8.6%	100.0%
Urban	0.039	9.0%	42.6%	13.7%	2.4%	32.7%
Rural	0.161	36.5%	44.1%	21.8%	11.6%	67.3%

Notes: Source: DHS year 2015-2016, own calculations.

[1]A formal explanation of the method is presented in Alkire and Foster (2011). An application of the method is presented in Alkire and Santos (2014).
[2]It should be noted that the AF method can be used with different indicators, weights and cutoffs to develop national MPIs that reflect the priorities of individual countries. National MPIs are more tailored to the context but cannot be compared.

Figure 2. Headcount Ratios by Poverty Measures



Notes: Source for global MPI: DHS, year 2015-2016, own calculations. Monetary poverty measures are the most recent estimates from World Bank (World Bank, 2018). Monetary poverty measure refer to 2011 ($1.90 a day), 2011 ($3.10 a day), and 2011 (national measure).

A headcount ratio is also estimated for two other ranges of poverty cutoffs. A person is identified as **vulnerable** to poverty if they are deprived in 20–33.33% of the weighted indicators. Concurrently, a person is identified as living in **severe poverty** if they are deprived in 50–100% of the weighted indicators. A summary of the global MPI statistics are presented in table 1 for national, rural and urban areas.

A brief methodological note is published following each round of global MPI update. For example, for the global MPI December 2018 update, please refer to Alkire **et al.** (2018). The note explains the methodological adjustments that were made while revising and standardizing indicators for over 100 countries. As such, it is useful to refer to the methodological notes with this country brief for specialized information on how the country survey data was managed.[3]

Poverty Headcount Ratios

Figure 2 compares the headcount ratios of the global MPI and monetary poverty measures. The height of the first bar of figure 2 shows the percentage of people who are MPI poor. The second and third bars represent the percentage of people who are poor according to the World Bank's $1.90 a day and $3.10 a day poverty line. The final bar denotes the percentage of people who are poor according to the national income or consumption and expenditure poverty measures.

[3]Previous methodological notes, published for each round of update, are made available on the OPHI website:http://ophi.org.uk/multidimensional-poverty-index/mpi-resources/.

- 1 for each country, 9–12 pages, up to 9 figures and 2 tables
- some countries lack section 'Subnational Analysis'

Graph and country brief production

- graphs for other countries or parameter choices are easy to obtain
- use (i) LaTeX-template, (ii) rely on LaTeX-variables, (iii) `ctyselect`

```
tempname lc
file open ‘lc’ using lc.tex , w t replace
file w ‘lc’ "\newcommand\ctyname{‘ctyname’}" _n ///
    "\newcommand\ctycode{‘ctycode’}" _n ///
    "\newcommand\calcyear{‘year’}" _n ///
    ...
file close ‘lc’
...
!pdflatex --interaction=nonstopmode --shell-escape
   \input{CB_template.tex}
!mv "CB_template.pdf" "pdfs/CB_‘ctycode’.pdf"
```

- Latex includes country-specific figures and omits entire section if needed.

COT-induced changes

Countries are now observed in several different years.

1. dedicated **reference sheet** for this release branch (HOT)

```
. li ccty survey year t T fname if ccty == "SEN" , noobs
```

ccty	survey	year	t	T	fname
SEN	DHS	2005	1	3	sen_dhs05.dta
SEN	DHS	2017	2	3	sen_dhs17.dta
SEN	DHS	2019	3	3	sen_dhs19.dta

2. **new variables** in results file: `flavour` and `ctype`
3. dedicated **results file for changes**:

```
. li ccty b measure t0 t1 ctype year_t? survey_t? if ccty == "SEN" , noob
```

ccty	b	measure	t0	t1	ctype	year_t0	year_t1	survey~0	survey~1
SEN	XX	H	1	2	abs	2005	2017	DHS	DHS
SEN	XX	H	1	2	rel	2005	2017	DHS	DHS
SEN	XX	H	2	3	abs	2017	2019	DHS	DHS
SEN	XX	H	2	3	rel	2017	2019	DHS	DHS

Lessons

- a sensible workflow has many benefits
  - often simpler and **cleaner code**, less programming needed (e.g., missing indicators)
  - may allow sensible **packaging** of the code (e.g., `ctyselect`)
  - principle-based workflow **simplifies documentation**
  - well-defined production stages encourage **division of work**
- key insights to identify this workflow
  1. clarify the objective of the project
  2. 'Data dominates. If you've chosen the right data structures and organized things well, the algorithms will almost always be self-evident. [...]' (Rob Pike rule 5)
- it was **not trivial** to develop a sensible work flow
  - required lots of discussion, experimentation and time
  - simple coding decisions may prove to determine the workflow

Open issues

- public and internal **documentation**
  - gitlab wiki? (Stata help files, desktop companion), paper, ...
- finalize **COT integration**
  - management of different versions of time-harmonized indicators
  - naming conventions, graphs, various tweaks, ...
- integrate **version control** (git) more rigorously
- finalize & release underlying **MPI toolbox**
- which other aspects could be interesting for a wider audience?
  - ancient coding decisions, which turned out to be problematic?
  - difficult trade-offs faced during revision?
  - contextual factors?

Questions, comments, and suggestions are always welcome under

nsuppa@ced.uab.es

nicolaisuppa

@nicolaisuppa | @CEDemografia | @ophi_oxford

References

Alkire, S. and Foster, J. (2011). Counting and multidimensional poverty measurement. *Journal of Public Economics*, 95(7-8):476–487.

Alkire, S., Kanagaratnam, U., and Suppa, N. (2020). The global multidimensional poverty index (MPI): 2020. OPHI MPI Methodological Notes 49, Oxford Poverty and Human Development Initiative, University of Oxford.

Alkire, S. and Santos, M. E. (2014). Measuring acute poverty in the developing world: Robustness and scope of the multidimensional poverty index. *World Development*, 59:251–274.

Cox, N. (2005). Suggestions on stata programming style. *The Stata Journal*, 5(4):560–566.

Jann, B. (2005). Making regression tables from stored estimates. *The Stata Journal*, 5(3):288–308.

Jann, B. (2007). Making regression tables simplified. *The Stata Journal*, 7(2):227–244.

Kohler, U. and Kreuter, F. (2012). *Data Analysis Using Stata, Third Edition*. Stata Press.

Long, J. S. (2008). *The Workflow of Data Analysis Using Stata*. Stata Press.

Mitchell, M. N. (2010). *Data Management Using Stata*. Stata Press.

Sen, A. K. (1992). *Inequality Reexamined*. Russell Sage Foundation book. Russell Sage Foundation, New York, 3 edition.