Stata and R PIP clients

Accessing the World Bank's estimates of poverty and inequality

The `pip` Stata client (wrapper)

Installation

ssc

```
1 ssc install pip
```

Github

```
1 net install github, from("https://haghish.github.io/github/")
2 github install worldbank/pip
```

Installation

SSC

```
1 ssc install pip
```

Github

```
1 net install github, from("https://haghish.github.io/github/")
2 github install worldbank/pip, version(0.9.5.9003)
```

- Official website https://worldbank.github.io/pip/
- Report issues at https://github.com/worldbank/pip/issues

Country-level estimates

```
1 /* To query poverty at $2.15-a-day poverty line for
2 all countries in all survey years*/
3
4 pip,  clear
```

```
Version in use: 20230328_2017_01_02_PROD

  first observation
```

country_code	year	poverty_line	headcount	mean	median	welfare_type
AGO	2000	2.15	0.2141	7.3744	4.6645291	consumption

```
Click here to display how to cite
```

Country-level estimates

```
1 /* To query poverty at \$2.15-a-day poverty line
2 for Morocco in 2013 */
3
4 pip, country(mar) year(2013) clear
```

```
Version in use: 20230328_2017_01_02_PROD

 first observation
```

country_code	year	poverty_line	headcount	mean	median	welfare_type
MAR	2013	2.15	0.0144	10.5652	7.7121489	consumption

```
Click here to display how to cite
```

Country-level estimates

```
1 /* For extrapolated and interpolated data that
2 underpin the global and regional poverty numbers, use
3 `fillgaps` option */
4
5 pip, country(mar) year(2019) clear fillgaps
```

```
Version in use: 20230328_2017_01_02_PROD

 first observation
```

country_code	year	poverty_line	headcount	mean	welfare_type
MAR	2019	2.15	0.0067	12.4621	consumption

```
Click here to display how to cite
```

Global and regional poverty estimates

```
1  pip wb, clear
```

```
Version in use: 20230328_2017_01_02_PROD

 first observation
```

region_code	year	poverty_line	headcount	mean
WLD	2019	2.15	0.0851	18.0426

```
Click here to display how to cite
```

Global and regional poverty estimates

```
1 pip wb, clear
2
3 /* Query a particular region or global values with `region()` */
4
5 pip wb, clear region(wld)
6 pip wb, clear region(lac)
```

```
Version in use: 20230328_2017_01_02_PROD

_first observation
```

region_code	year	poverty_line	headcount	mean
LAC	2021	2.15	0.0469	17.3542

```
Click here to display how to cite
```

Poverty lines

```
1  /*Query poverty at other thresholds*/
2
3  pip, country(mar) year(2019) clear ///
4  fillgaps povline(6.85)
```

```
Version in use: 20230328_2017_01_02_PROD

 first observation
```

country_code	year	poverty_line	headcount	mean	welfare_type
MAR	2019	6.85	0.3144	12.4621	consumption

```
Click here to display how to cite
```

Poverty lines

```
1  /*Or multiple thresholds*/
2
3  pip, country(mar) year(2019) clear ///
4  fillgaps povline(2.15 3.65 6.85 10)
```

```
Version in use: 20230328_2017_01_02_PROD

 first 4 observations
```

country_code	year	poverty_line	headcount	mean	welfare_type
MAR	2019	10.00	0.5601	12.4621	consumption
MAR	2019	6.85	0.3144	12.4621	consumption
MAR	2019	3.65	0.0578	12.4621	consumption
MAR	2019	2.15	0.0067	12.4621	consumption

```
Click here to display how to cite
```

Data availability

```
1  pip info, clear
```

pip requires the packages below from SSC:
Version in use: 20230328_2017_01_02_PROD

Available Surveys: Select a country or region

Countries

AGO	ALB	ARE	ARG	ARM	AUS	AUT	AZE
BDI	BEL	BEN	BFA	BGD	BGR	BIH	BLR
BLZ	BOL	BRA	BTN	BWA	CAF	CAN	CHE
CHL	CHN	CIV	CMR	COD	COG	COL	COM
CPV	CRI	CYP	CZE	DEU	DJI	DNK	DOM
DZA	ECU	EGY	ESP	EST	ETH	FIN	FJI
FRA	FSM	GAB	GBR	GEO	GHA	GIN	GMB
GNB	GRC	GTM	GUY	HND	HRV	HTI	HUN
IDN	IND	IRL	IRN	IRQ	ISL	ISR	ITA
JAM	JOR	JPN	KAZ	KEN	KGZ	KIR	KOR
LAO	LBN	LBR	LCA	LKA	LSO	LTU	LUX
LVA	MAR	MDA	MDG	MDV	MEX	MHL	MKD
MLI	MLT	MMR	MNE	MNG	MOZ	MRT	MUS
MWI	MYS	NAM	NER	NGA	NIC	NLD	NOR
NPL	NRU	PAK	PAN	PER	PHL	PNG	POL
PRT	PRY	PSE	ROU	RUS	RWA	SDN	SEN
SLB	SLE	SLV	SRB	SSD	STP	SUR	SVK
SVN	SWE	SWZ	SYC	SYR	TCD	TGO	THA
TJK	TKM	TLS	TON	TTO	TUN	TUR	TUV
TWN	TZA	UGA	UKR	URY	USA	UZB	VEN
VNM	VUT	WSM	XKX	YEM	ZAF	ZMB	ZWE

Regions

EAP ECA LAC MNA NAC SAR SSA
World Bank regions by year

Data availability

```
1  pip, country(mar) year(2019) clear
```

```
Warning: years selected for MAR do not match any survey year.
You could type pip_info, country(MAR) version(20230328_2017_01_02_PROD) clear to check availability.

the countries and years selected do not match any year available.
invalid syntax
r(197); t=0.40 17:34:04

. pip_info, country(MAR) version(20230328_2017_01_02_PROD) clear

Available Surveys for MAR
 Detailed information (browser)

Morocco-national
survey year
 1984 1990 1998 2000 2006 2013 All
```

Replicability

```
1  pip version, clear
```

	version
1.	20230328_2017_01_02_PROD
2.	20230328_2011_02_02_PROD
3.	20220909_2017_01_02_PROD
4.	20220909_2011_02_02_PROD

Towards distributional analysis

```
1  pip, country(mar) year(2019) clear fillgaps popshare(0.5)
```

```
Version in use: 20230328_2017_01_02_PROD

 first observation
```

country_code	year	poverty_line	headcount	mean	welfare_type
MAR	2019	9.10	0.5000	12.4621	consumption

```
Click here to display how to cite
```

Towards distributional analysis

```
1  numlist ".3(.1).8"
2  pip, country(mar) year(2019) clear fillgaps ///
3      popshare(`r(numlist)') n2disp(6)
```

```
Version in use: 20230328_2017_01_02_PROD

 first 6 observations
```

country_code	year	poverty_line	headcount	mean	welfare_type
MAR	2019	16.01	0.8000	12.4621	consumption
MAR	2019	12.82	0.6999	12.4621	consumption
MAR	2019	10.71	0.5999	12.4621	consumption
MAR	2019	9.10	0.5000	12.4621	consumption
MAR	2019	7.82	0.3999	12.4621	consumption
MAR	2019	6.70	0.3000	12.4621	consumption

```
Click here to display how to cite
```

Towards distributional analysis

```
1  numlist ".3(.1).8"
2  pip, country(mar) year(2019) clear fillgaps ///
3      popshare(`r(numlist)') n2disp(6)
```

```
Version in use: 20230328_2017_01_02_PROD

 first 6 observations
```

country_code	year	poverty_line	headcount	mean	welfare_type
MAR	2019	16.01	0.8000	12.4621	consumption
MAR	2019	12.82	0.6999	12.4621	consumption
MAR	2019	10.71	0.5999	12.4621	consumption
MAR	2019	9.10	0.5000	12.4621	consumption
MAR	2019	7.82	0.3999	12.4621	consumption
MAR	2019	6.70	0.3000	12.4621	consumption

```
Click here to display how to cite
```

PIP Percentiles data

https://pip.worldbank.org

Auxiliary Data

```
1  pip tables, clear
```

```
Auxiliary tables available for 20230328_2017_01_02_PROD:
     01 | aux_versions
     02 | countries
     03 | country_coverage
     04 | country_list
     05 | cpi
     06 | decomposition
     07 | dictionary
     08 | framework
     09 | gdp
     10 | incgrp_coverage
     11 | indicators
     12 | interpolated_means
     13 | missing_data
     14 | national_poverty_lines
     15 | pce
     16 | pop
     17 | pop_region
     18 | poverty_lines
     19 | ppp
     20 | region_coverage
     21 | regions
     22 | spl
     23 | survey_means
```

Alternatively

```
1  pip tables, table(cpi) clear
2
3  pip tables, table(interpolated_means) clear
```

Help file

2. General troubleshooting (Go up to *Sections Menu*)

In case **pip** is not working correctly, try the following steps in order

1. Uninstall **pip** by typing **pip uninstall**
2. Execute **which pip**. If **pip** is still installed, delete all the **pip** files from wherever they are in your computer until the command above returns error. The idea is to leave no trace of **pip** in your computer.
3. Install **pip** again with the following code and check the version number. It should be the same as the most recent release

```
github install worldbank/pip
discard
which pip
```

4. Try to run it again and see if **pip** fails.
5. If it is still failing, open a new issue in the GitHub issues page, making sure you're adding all the necessary steps to reproduce the problem.
6. Once the issue is created, run the code below--making sure you replace the commented line--and send the test.log file, along with the issue number created in the previous step, to pip@worldbank.org.

```
log using "test.log", name(pip_test) text replace // this is in your cd
```

https://pip.worldbank.org

```
cret list
clear all
which pip
set tracedepth 4
set traceexpand on
set traceindent on
set tracenumber on
set trace on
/* the pip command that is failing. e.g., */
cap noi pip, region(EAP) year(last) clear
set trace off
log close pip_test
```

These styles are often rendered as color. In the Results window, on a white background,

decile9	decile9
decile10	decile10
ppp	ppp
is_interpolated	isinterpolated
distribution_type	usemicrodata
survey_coverage	coveragetype

These styles are often rendered as color. In the Results window, on a white background,

Parameters	Description
country(3-letter code)	List of *country code* (accepts multiples) or *all*. Default "*all*". Cannot be used with option *region()*
region(WB code)	List of *region code* (accepts multiple) or *all*. Default "*all*". Cannot be used with option *country()*
coverage(*string*)	Coverage level ("national", "urban", "rural", "all"). Default "all".
year(numlist|string)	List of years (accepts up to 10), or *all*, or *last*. Default "all".
povline(#)	List of poverty lines (in PPP specified, see option **ppp_year**(#)) to calculate poverty measures (accepts up to 5). Default is 2.15 and 2017 PPPs.
popshare(#)	List of quantiles. No default. Cannot be used with option **povline**
fillgaps	Loads country-level estimates (including extrapolations and interpolations) used to create regional and global aggregates.

Options	Description

R Client (wrapper)

Installation

```
# install.packages("devtools")
devtools::install_github("worldbank/pipr")
```

Webpage

https://worldbank.github.io/pipr/

Country-level estimates

```
 1 pipr::get_stats() |>
 2   select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 2,320 × 4
   country_code  year poverty_line   headcount
   <chr>        <dbl>        <dbl>       <dbl>
 1 AGO           2000         2.15 0.2140885
 2 AGO           2008         2.15 0.1463249
 3 AGO           2018         2.15 0.3112201
 4 ALB           1996         2.15 0.005348460
 5 ALB           2002         2.15 0.01092647
 6 ALB           2005         2.15 0.005910857
 7 ALB           2008         2.15 0.001999251
 8 ALB           2012         2.15 0.006206909
 9 ALB           2014         2.15 0.01023101
10 ALB           2015         2.15 0.001206211
# ℹ 2,310 more rows
```

Country-level estimates

```
 1 pipr::get_stats(country = "MAR") |>
 2   select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 6 × 4
  country_code  year poverty_line  headcount
  <chr>        <dbl>        <dbl>      <dbl>
1 MAR           1984         2.15 0.1319537
2 MAR           1990         2.15 0.04728802
3 MAR           1998         2.15 0.08375334
4 MAR           2000         2.15 0.07934032
5 MAR           2006         2.15 0.04242266
6 MAR           2013         2.15 0.01443110
```

Country-level estimates

```
 1 pipr::get_stats(country = c("COL", "MAR")) |>
 2   arrange(year) |>
 3   select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 33 × 4
   country_code  year poverty_line  headcount
   <chr>        <dbl>        <dbl>      <dbl>
 1 COL           1980         2.15 0.2075780
 2 MAR           1984         2.15 0.1319537
 3 COL           1988         2.15 0.1524133
 4 COL           1989         2.15 0.1416732
 5 MAR           1990         2.15 0.04728802
 6 COL           1991         2.15 0.1341064
 7 COL           1992         2.15 0.09557514
 8 COL           1996         2.15 0.1757369
 9 MAR           1998         2.15 0.08375334
10 COL           1999         2.15 0.2135428
# ℹ 23 more rows
```

Country-level estimates

```
pipr::get_stats(country = "COL", year = c(2012, 2018))  |>
  select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 2 × 4
  country_code  year poverty_line  headcount
  <chr>        <dbl>        <dbl>      <dbl>
1 COL           2012         2.15 0.06715117
2 COL           2018         2.15 0.04498261
```

Interporlated and Extrapolated values

```
 1 pipr::get_stats(country = "HTI")  |>
 2   select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 3 × 4
  country_code  year poverty_line headcount
  <chr>        <dbl>        <dbl>     <dbl>
1 HTI           2001         2.15 0.6783363
2 HTI           2012         2.15 0.2919219
3 HTI           2012         2.15 0.5787614
```

Interporlated and Extrapolated values

```
 1  pipr::get_stats(country = "HTI", fill_gaps = TRUE)    |>
 2    select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 41 × 4
   country_code  year poverty_line headcount
   <chr>        <dbl>        <dbl>     <dbl>
 1 HTI           1981         2.15 0.5127917
 2 HTI           1982         2.15 0.5320211
 3 HTI           1983         2.15 0.5377601
 4 HTI           1984         2.15 0.5445521
 5 HTI           1985         2.15 0.5504935
 6 HTI           1986         2.15 0.5627245
 7 HTI           1987         2.15 0.5717675
 8 HTI           1988         2.15 0.7630265
 9 HTI           1989         2.15 0.7783534
10 HTI           1990         2.15 0.7747321
# ℹ 31 more rows
```

Poverty line

```
 1 pipr::get_stats(country = "COL", povline = 3)  |>
 2   select(c(country_code, year, poverty_line, headcount))
```

```
# A tibble: 27 × 4
   country_code  year poverty_line headcount
   <chr>        <dbl>        <dbl>     <dbl>
 1 COL           1980            3 0.2994786
 2 COL           1988            3 0.2344281
 3 COL           1989            3 0.2310108
 4 COL           1991            3 0.2139244
 5 COL           1992            3 0.1753281
 6 COL           1996            3 0.2701443
 7 COL           1999            3 0.3089631
 8 COL           2000            3 0.2578116
 9 COL           2001            3 0.3186980
10 COL           2002            3 0.2312075
# ℹ 17 more rows
```

Multiple Poverty lines

```
povlines <- c(2, 4, 6, 10)
map_df(.x       = povlines,
       .f       = pipr::get_stats,
       country = "COL",
       year     = 2012) |>
  select(c(country_code, year, poverty_line, headcount))
```

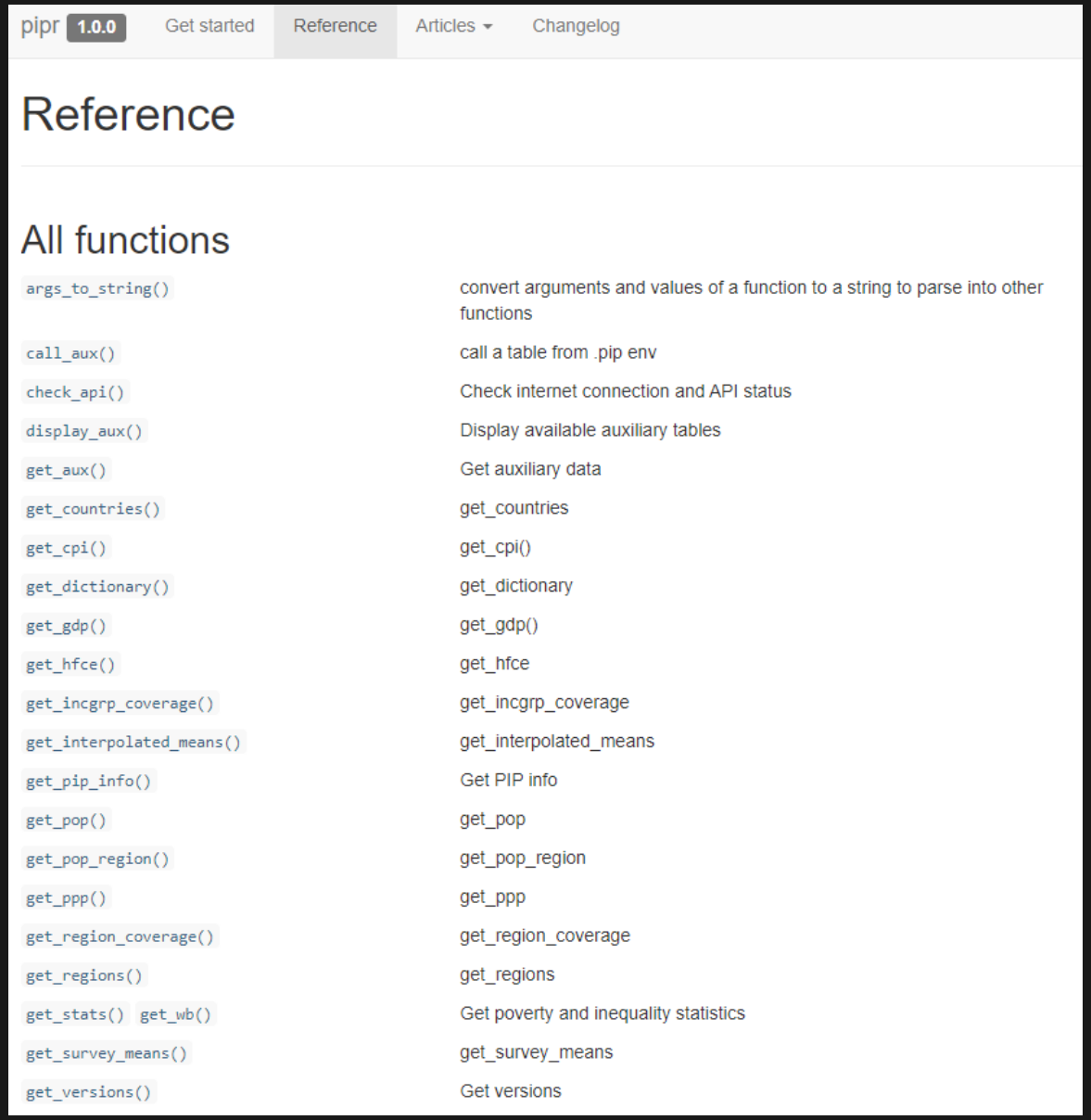
```
# A tibble: 4 × 4
  country_code  year poverty_line  headcount
  <chr>        <dbl>        <dbl>      <dbl>
1 COL           2012            2 0.05875268
2 COL           2012            4 0.1939661
3 COL           2012            6 0.3436242
4 COL           2012           10 0.5632525
```

Global estimates

```
 1 pipr::get_wb()  |>
 2   arrange(year) |>
 3   select(c(region_code, year, poverty_line, headcount))
```

```
# A tibble: 347 × 4
   region_code  year poverty_line   headcount
   <chr>       <dbl>        <dbl>       <dbl>
 1 OHI          1981          1.9 0.005183592
 2 LAC          1981          1.9 0.1262799
 3 SAS          1981          1.9 0.4871674
 4 EAP          1981          1.9 0.7870897
 5 WLD          1981          1.9 0.3933288
 6 OHI          1982          1.9 0.005643049
 7 LAC          1982          1.9 0.1316380
 8 SAS          1982          1.9 0.4828071
 9 EAP          1982          1.9 0.7630886
10 WLD          1982          1.9 0.3875048
# ℹ 337 more rows
```

Many more functions



pipr 1.0.0 | Get started | Reference | Articles | Changelog

Reference

All functions

`args_to_string()`	convert arguments and values of a function to a string to parse into other functions
`call_aux()`	call a table from .pip env
`check_api()`	Check internet connection and API status
`display_aux()`	Display available auxiliary tables
`get_aux()`	Get auxiliary data
`get_countries()`	get_countries
`get_cpi()`	get_cpi()
`get_dictionary()`	get_dictionary
`get_gdp()`	get_gdp()
`get_hfce()`	get_hfce
`get_incgrp_coverage()`	get_incgrp_coverage
`get_interpolated_means()`	get_interpolated_means
`get_pip_info()`	Get PIP info
`get_pop()`	get_pop
`get_pop_region()`	get_pop_region
`get_ppp()`	get_ppp
`get_region_coverage()`	get_region_coverage
`get_regions()`	get_regions
`get_stats()` `get_wb()`	Get poverty and inequality statistics
`get_survey_means()`	get_survey_means
`get_versions()`	Get versions

https://pip.worldbank.org

Code | plot

```
 1  # Global poverty trends 1990-2019
 2  df <- get_wb() |>
 3    filter(year > 1989, region_code == "WLD") |>
 4    mutate(
 5      pop_in_poverty = round(pop_in_poverty / 1000000, 0),
 6      headcount = round(headcount, 3)
 7    )
 8
 9  headcount_col <- "#E69F00"
10
11  gr <- ggplot(df, aes(x = year)) +
12    geom_text(aes(label = headcount * 100,
13                  y     = headcount),
14              vjust   = 1,
215             nudge_y = -,
16              color   = headcount_col) +
17    geom_text(aes(label = pop_in_poverty,
18                  y     = pop_in_poverty / 5000),
```

Code | plot

```
# Number of poor by region
df <- get_wb() |>
  filter(year > 1989 & year < 2019) |>
  mutate(
    pop_in_poverty = round(pop_in_poverty / 1000000, 0),
    headcount = round(headcount, 3)
  )

regions <- df |>
  filter(!region_code %in% c("WLD", "AFE", "AFW")) |>
  mutate(
    region_name = fct_relevel(region_name,
                              c("Other high Income",
                                "Europe and Central Asia",
                                "Middle East and North Africa",
                                "Latin America and the Caribbean",
                                "East Asia and Pacific",
                                "South Asia",
```

Thanks.