

SubnationalCRVS R Package: Demo

Jeremy Roth

Contents

1	Setup	1
2	View the Example Datasets Included with the SubnationalCRVS Package	2
3	Conduct Demographic Data Quality Assessment (DDQA)	3
3.1	Sex ratio: PlotSexRatios()	3
3.2	Age ratios: PlotAgeRatios()	6
3.3	Potential age heaping: PlotPotentialAgeHeaping()	8
3.4	Age heaping indices: PlotAgeHeapingScores()	10
4	DDM Estimation of Death Registration Completeness	12
4.1	Compute DDM estimates: EstimateDDM()	13
4.2	Plot DDM estimates: PlotDDM()	13

1 Setup

Since the SubnationalCRVS package is still under active development, it is hosted on my GitHub page (www.github.com/jroth-unfpa/SubnationalCRVS) rather than on CRAN. As a result, SubnationalCRVS cannot be installed with the usual `install.packages()` function; instead, we install SubnationalCRVS with the `install.github()` function from the devtools package. The dependency DemoTools is also not hosted on CRAN and must be installed with `install.github()`.

```
library(devtools)
install_github("timriffe/DemoTools") # installing the DemoTools dependency (not on CRAN)
install_github("jroth-unfpa/SubnationalCRVS") # installing the SubnationalCRVS package itself
```

Now we can load SubnationalCRVS, specify the name of the local folder in which we will save the plots, and create that local folder if it does not already exist.

```
library(SubnationalCRVS)
my_plots_dir <- "Plots/" # local folder where the plots should be saved
dir.create(my_plots_dir) # create the folder if it does not already exist
```

We will also load the dplyr package to customize the display of some of the tables returned by SubnationalCRVS.

```
library(dplyr)
```

2 View the Example Datasets Included with the Subnational-CRVS Package

The `SubnationalCRVS` package comes with two tabulations – disaggregated by age, sex, and province – created from publicly available datasets from Ecuador (LINKS), based on the country's 2001 Census, 2010 Census, and annual counts of registered deaths from 2001 through 2010.

```
head(ecuador_single_year_ages)
```

##	province_name	province_name_short	sex	age	pop1	pop2	date1	date2
## 1	Azuay	Azu	m	0	6086	6750	2001-11-25	2010-11-28
## 2	Azuay	Azu	m	1	6555	6984	2001-11-25	2010-11-28
## 3	Azuay	Azu	m	2	7232	7090	2001-11-25	2010-11-28
## 4	Azuay	Azu	m	3	7101	7095	2001-11-25	2010-11-28
## 5	Azuay	Azu	m	4	7083	6961	2001-11-25	2010-11-28
## 6	Azuay	Azu	m	5	6583	6895	2001-11-25	2010-11-28

The `ecuador_single_year_ages` dataset reports the estimated populations in Ecuador by single-year ages (0, 1, 2, ... in the `age` column) from both the 2001 Census (`pop1` column) and the 2010 Census (`pop2` column), separately for males and females (m and f in the `sex` column) and province (full name in the `province` column and abbreviated name in the `province_name_short` column).

```
head(ecuador_five_year_ages)
```

##	province_name	province_name_short	sex	age	pop1	pop2	deaths	date1
## 1	Azuay	Azu	m	0	34101	34886	772	2001-11-25
## 2	Azuay	Azu	m	10	34946	38125	223	2001-11-25
## 3	Azuay	Azu	m	15	32387	37611	416	2001-11-25
## 4	Azuay	Azu	m	20	25634	33665	480	2001-11-25
## 5	Azuay	Azu	m	25	18606	28376	475	2001-11-25
## 6	Azuay	Azu	m	30	16193	22026	456	2001-11-25
##	date2							
## 1	2010-11-28							
## 2	2010-11-28							
## 3	2010-11-28							
## 4	2010-11-28							
## 5	2010-11-28							
## 6	2010-11-28							

The `ecuador_five_year_ages` dataset has the same variables as `ecuador_single_year_ages` with two exceptions: (1) the `age` variable now represents five-year age groups (in the `age` column, with 0-4 coded as 0, 5-9 coded as 5, 10-14 coded as 10, etc.) instead of single-year ages; and (2) there is an additional column called `deaths` that reports the registered deaths collected between 2001 and 2010.

3 Conduct Demographic Data Quality Assessment (DDQA)

3.1 Sex ratio: PlotSexRatios()

One step in the demographic data quality assessment (DDQA) process is to use the `PlotSexRatios()` function in the `SubnationalCRVS` package to compute and plot sex ratios within each combination of province, sex, and single-year age for the 2001 and 2010 data stored in `ecuador_single_year_ages`.

We must provide our tabulated data frame in the `data` argument of `PlotSexRatios()` and, additionally, we need to provide a few additional arguments to `PlotSexRatios()` to describe the variable names and values in the dataset:

- `name.disaggregations` is the name of variable representing the subnational disaggregation (apart from sex) in `data`. Here we specify `name.disaggregations="province_name"`.
- `name.sex` is the name of variable representing sex. Here we specify `name.sex="sex"`
- `name.age` is the name of variable representing age. Here we specify `name.age="age"`
- `name.date1` is the name of variable that provides the data of the earlier of the two time periods. Here we specify `name.date1="date1"` (the value of this variable, "2001-11-15" is the date of Ecuador's 2001 Census)
- `name.date2` is the name of variable that provides the data of the later of the two time periods. Here we specify `name.date2="date2"` (the value of this variable, "2010-11-28" is the date of Ecuador's 2010 Census)
- `name.population.year1` is the name of variable representing the population in the earlier of the two time periods represented in the dataset. Here we specify `name.population.year1="pop1"`
- `name.population.year2` is the name of variable representing the population in the later of the two time periods represented in the dataset. Here we specify `name.population.year2="pop2"`
- `name.male` is the name of value of the `name.sex` variable that represents males
- `name.female` is the name of value of the `name.sex` variable that represents females

In addition, we specify the option argument `plots.dir=my_plots_dir` so that the plots will save in a local folder `Plots`; the default behavior would be to save the plots in the working directory of the R script.

```
c_s <- ComputeSexRatios(data=ecuador_single_year_ages,
  name.disaggregations="province_name",
  name.males="m",
  name.females="f",
  name.age="age",
  name.sex="sex",
  name.date1="date1",
  name.date2="date2",
  name.population.year1="pop1",
  name.population.year2="pop2")

s <- PlotSexRatios(data=ecuador_single_year_ages,
  name.disaggregations="province_name",
  name.males="m",
  name.females="f",
  name.age="age",
```

```

name.sex="sex",
name.date1="date1",
name.date2="date2",
name.population.year1="pop1",
name.population.year2="pop2",
label.subnational.level="Province",
plots.dir=my_plots_dir)

```

The plots of sex ratios are saved in the `Plots/` sub-folder we specified with the argument `plots.dir=my_plots_dir`; `plots.dir` is an optional argument and, if we do not specify a value for it, the plots will be saved the working directory. We also specified `label.subnational.level="Province"` so that the disaggregations are labeled `Province` instead of the less clear `province_name`.

3.1.1 View sex ratios in combined plot

The sex ratios for all levels of subnational disaggregation are overlaid in the following “combined” plots separately for each data year.

```

knitr::include_graphics(path=paste0(my_plots_dir,
    "sex_ratios_combined_province_name_",
    Sys.Date(),
    ".pdf"))

```

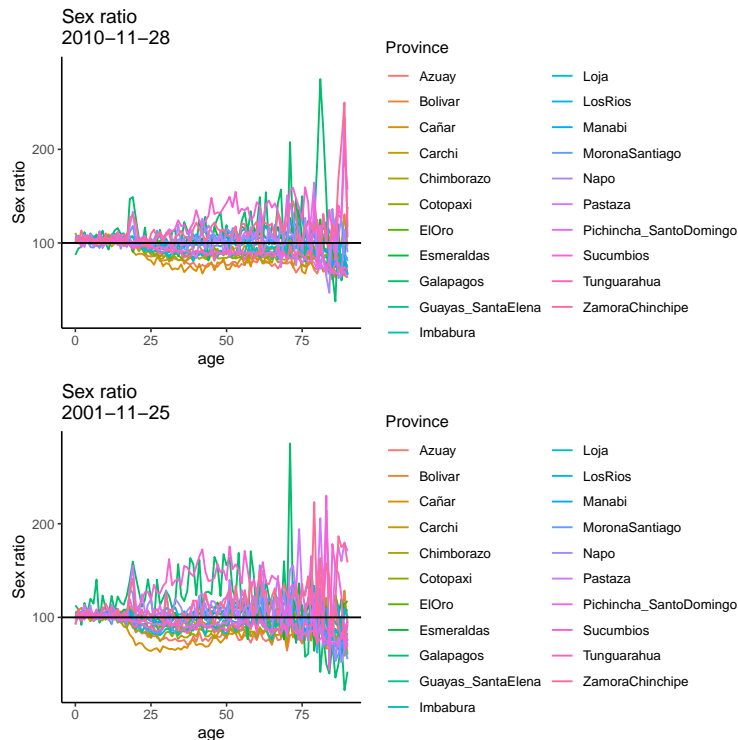


Figure 1: Sex ratios in Ecuador by province, combined plot

3.1.2 View sex ratios in disaggregated plots

Additionally, the sex ratios are plotted in separate figures for each level of subnational disaggregation in the following “disaggregated” plots

```
knitr::include_graphics(path=paste0(my_plots_dir,
                                     "sex_ratios_by_province_name_",
                                     Sys.Date(),
                                     ".pdf"))
```

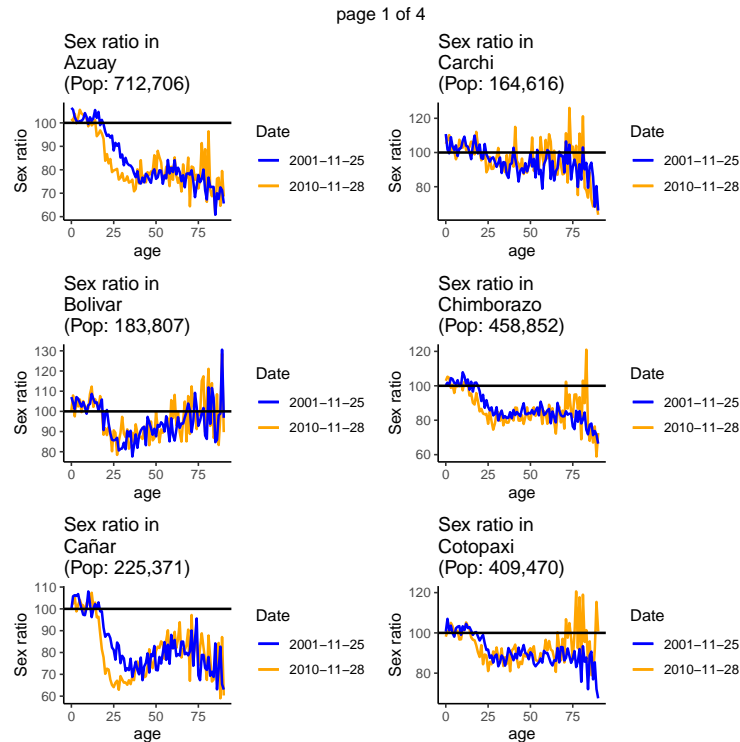


Figure 2: Sex ratios in Ecuador by province, disaggregated plots

3.1.3 View sex ratios in table

The object returned by `PlotSexRatios()` is a table that shows us the sex ratios for each combination of province, sex, and single-year age in the `sex_ratio_1` column (for the 2001 Census) and the `sex_ratio_2` column (for the 2010 Census).

```
s %>% select(province_name, age, pop1, pop2, sex_ratio_1, sex_ratio_2) %>%
  head()
```

##	province_name	age	pop1	pop2	sex_ratio_1	sex_ratio_2
## 1	Azuay	0	12073	13092	101.7	106.4
## 2	Azuay	1	13060	13596	100.8	105.6
## 3	Azuay	2	14195	14014	103.9	102.4
## 4	Azuay	3	14217	14124	99.8	100.9
## 5	Azuay	4	14012	13911	102.2	100.2
## 6	Azuay	5	12815	13736	105.6	100.8

3.2 Age ratios: PlotAgeRatios()

Another step in our demographic data quality assessment is using the `PlotAgeRatios()` function in the `SubnationalCRVS` package to compute and plot age ratios within each combination of province and sex for the 2001 and 2010 data stored in `ecuador_five_year_ages`.

```
a <- PlotAgeRatios(data=ecuador_five_year_ages,
  name.disaggregations="province_name",
  name.males="m",
  name.females="f",
  name.age="age",
  name.sex="sex",
  name.date1="date1",
  name.date2="date2",
  name.population.year1="pop1",
  name.population.year2="pop2",
  label.subnational.level="Province",
  plots.dir="Plots/")
```

The arguments we provided to `PlotAgeRatios()` are actually identical to those we specified for `PlotSexRatios()`, except now we are using the tabulation with five-year age groups (`ecuador_five_year_ages`) instead of the tabulation with single-year ages.

3.2.1 View age ratios in combined plot

The following “combined” plots, saved in the `Plots/` folder, show the age ratios for all levels of subnational disaggregation, separately for males and females in each data year.

```
knitr::include_graphics(path=paste0(my_plots_dir,
  "age_ratios_combined_province_name_",
  Sys.Date(),
  ".pdf"))
```

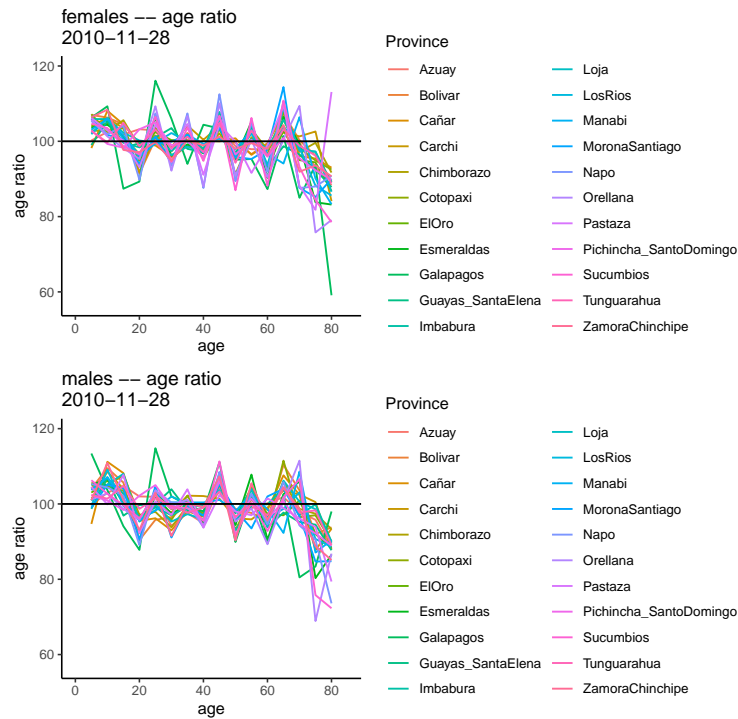


Figure 3: Age ratios in Ecuador by province, combined plot

3.2.2 View age ratios in disaggregated plots

In addition, `PlotAgeRatios()` also creates the following “disaggregated” plots, saved in the `Plots/` folder, where the age ratios for each level of disaggregation are shown in separate plots, with different sexes and data years overlaid within each plot.

```
knitr::include_graphics(path=paste0(my_plots_dir,
                                     "age_ratios_by_province_name_",
                                     Sys.Date(),
                                     ".pdf"))
```

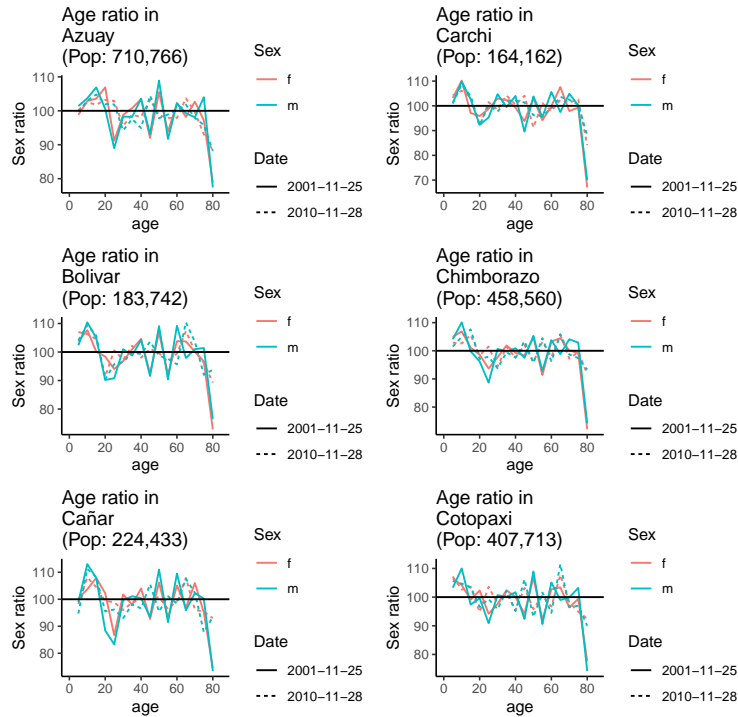


Figure 4: Age ratios in Ecuador by province, disaggregated plots

3.2.3 View age ratios in table

Just as the `PlotSexRatios()` function returns a table of disaggregated sex ratios, `PlotAgeRatios()` returns a table of disaggregated age ratios.

```
a %>% select(province_name, age, pop1, pop2, age_ratio_1, age_ratio_2) %>%
  head()
```

##	province_name	age	pop1	pop2	age_ratio_1	age_ratio_2
## 1	Azuay	0	33491	33876	NA	NA
## 2	Azuay	5	33817	35701	98.8	100.2
## 3	Azuay	10	34975	37366	102.9	102.5
## 4	Azuay	15	34181	37215	103.6	101.8
## 5	Azuay	20	31000	35753	106.9	103.2
## 6	Azuay	25	23844	32054	91.2	102.9

3.3 Potential age heaping: `PlotPotentialAgeHeaping()`

To give us a sense of whether “age-heaping” is occurring within the levels of disaggregation present in our `ecuador_single_year_ages` dataset, we turn to the `PlotPotentialAgeHeaping()` function and actually provide the same arguments we used in the `PlotSexRatios()` function.

```
PlotPotentialAgeHeaping(data=ecuador_single_year_ages,
  name.disaggregations="province_name",
  name.males="m",
```



```

name.females="f",
name.age="age",
name.sex="sex",
name.date1="date1",
name.date2="date2",
name.population.year1="pop1",
name.population.year2="pop2",
label.subnational.level="Province",
plots.dir="Plots/")

```

3.3.1 View potential age heaping in combined plot

The following “combined” plots show us estimated population counts by single-year ages with different provinces represented with different overlaid colors, and separate plots for each sex and data year.

```

knitr::include_graphics(path=paste0(my_plots_dir,
                                     "potential_age_heaping_combined_province_name_",
                                     Sys.Date(),
                                     ".pdf"))

```

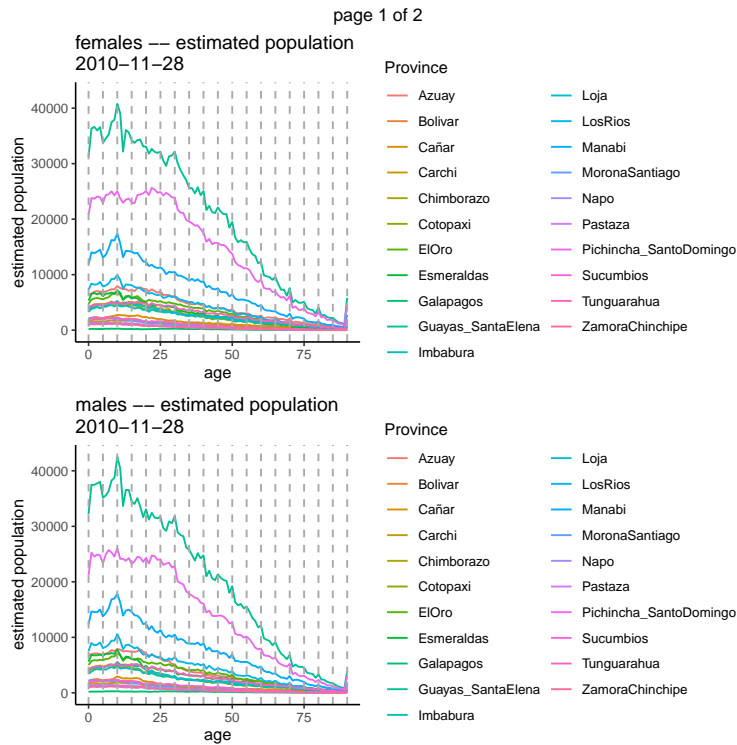


Figure 5: Population counts in Ecuador by single-year age, combined plot

3.3.2 View potential age heaping in disaggregated plots

`PlotPotentialAgeHeaping()` present separate plots of population counts for each province in Ecuador, with different sexes and data years overlaid within each plot.

```
knitr::include_graphics(path=paste0(my_plots_dir,
                                     "potential_age_heaping_by_province_name_",
                                     Sys.Date(),
                                     ".pdf"))
```

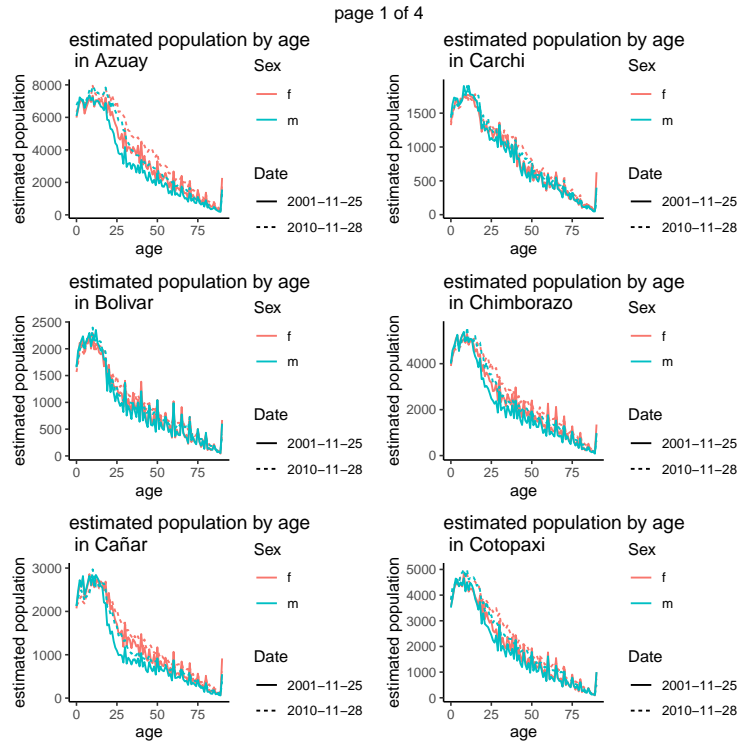


Figure 6: Population counts in Ecuador by single-year age, disaggregated plots

3.4 Age heaping indices: `PlotAgeHeapingScores()`

As a more concise summary of potential age-heaping suggested by the visualizations from `PlotPotentialAgeHeaping`, we now use the `PlotAgeHeapingScores` function with the same arguments we provided to the `PlotAgeRatios` function.

```
ageheaping <- PlotAgeHeapingScores(data=ecuador_single_year_ages,
                                   name.disaggregations="province_name",
                                   name.males="m",
                                   name.females="f",
                                   name.age="age",
                                   name.sex="sex",
                                   name.date1="date1",
                                   name.date2="date2",
                                   name.population.year1="pop1",
                                   name.population.year2="pop2",
                                   label.subnational.levels="Province",
                                   plots.dir="Plots/")
```

3.4.1 View age heaping indices in plots

`PlotAgeHeapingScores` plots the values of three age-heaping indices within each combination of province, sex, and data year: Roughness, Whipple, and Myers. The indices are computed with the `DemoTools::check_heaping_roughness`, `DemoTools::check_heaping_Whipple`, and `DemoTools::check_heaping_myers` functions, respectively, from the `DemoTools` package.

```
knitr::include_graphics(path=paste0(my_plots_dir,
                                     "age_heaping_scores_combined_province_name_",
                                     Sys.Date(),
                                     ".pdf"))
```

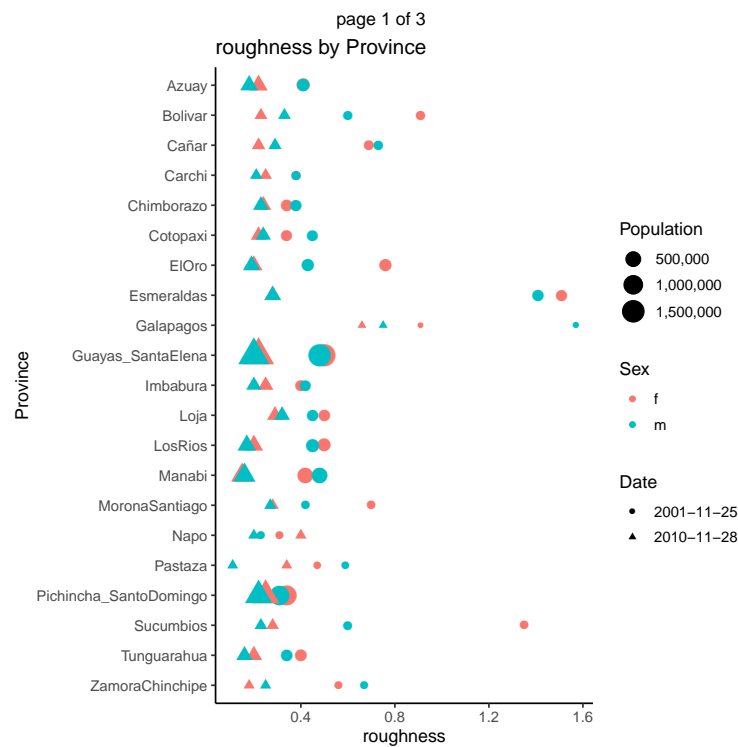


Figure 7: Age heaping indices in Ecuador by province

We can also view the age-heaping indices in the table returned by `PlotAgeHeapingScores`

3.4.2 View age heaping indices in table

```
head(ageheaping)
```

##	province_name	date	total_pop	sex	roughness	Whipple	Myers
## 1	Azuay	2001-11-25	319983	f	0.41	1.18	4.21
## 2	Bolivar	2001-11-25	86256	f	0.91	1.37	7.39
## 3	Cañar	2001-11-25	112041	f	0.69	1.22	4.89
## 4	Carchi	2001-11-25	77172	f	0.38	1.18	3.75
## 5	Chimborazo	2001-11-25	213106	f	0.34	1.25	5.44
## 6	Cotopaxi	2001-11-25	180328	f	0.34	1.27	5.99

3.4.3 View Noubissi age-heaping indices: ComputeAgeHeapingScores()

We can also use the `ComputeAgeHeapingScores` function – which is called within `PlotAgeHeapingScores` – with the optional argument `Noubissi.display=TRUE` to view the Noubissi index computed for single-year ages ending with 0, 1, 2, ..., 9.

```
ageheaping_with_Noubissi <- ComputeAgeHeapingScores(data=ecuador_single_year_ages,
  name.disaggregations="province_name",
  name.males="m",
  name.females="f",
  name.age="age",
  name.sex="sex",
  name.date1="date1",
  name.date2="date2",
  name.population.year1="pop1",
  name.population.year2="pop2",
  Noubissi.display=TRUE)

head(ageheaping_with_Noubissi)
```

```
##      date sex province_name total_pop roughness Whipple Myers Noubissi_0
## 1 2001-11-25 f      Azuay      319983      0.41    1.18  4.21      1.18
## 2 2001-11-25 f    Bolivar      86256      0.91    1.37  7.39      1.35
## 3 2001-11-25 f    Cañar      112041      0.69    1.22  4.89      1.22
## 4 2001-11-25 f    Carchi      77172      0.38    1.18  3.75      1.15
## 5 2001-11-25 f Chimborazo    213106      0.34    1.25  5.44      1.23
## 6 2001-11-25 f    Cotopaxi    180328      0.34    1.27  5.99      1.25
##  Noubissi_1 Noubissi_2 Noubissi_3 Noubissi_4 Noubissi_5 Noubissi_6
## 1      0.91      1.02      0.98      0.95      1.08      0.98
## 2      0.84      1.03      0.91      0.86      1.22      0.97
## 3      0.90      1.02      0.98      0.91      1.11      0.98
## 4      0.91      1.05      0.96      0.92      1.11      0.98
## 5      0.88      1.01      0.97      0.89      1.12      0.98
## 6      0.88      1.02      0.95      0.90      1.18      0.98
##  Noubissi_7 Noubissi_8 Noubissi_9
## 1      0.94      1.04      0.86
## 2      0.92      1.03      0.75
## 3      0.95      1.03      0.80
## 4      0.96      1.04      0.86
## 5      0.94      1.05      0.82
## 6      0.91      1.06      0.80
```

4 DDM Estimation of Death Registration Completeness

The structure of the `ecuador_five_year_ages` dataset is inspired by the requirements for the `DDM::ddm` function from the `DDM` package, which uses established Death Distribution Methods (DDM) to estimate death registration completeness between two consecutive Censuses. As a result, the `SubnationalCRVS` package offers the `EstimateDDM` function as a simple wrapper to `DDM::ddm` to perform DDM estimation of death registration completeness.

4.1 Compute DDM estimates: EstimateDDM()

```
ddm_results <- EstimateDDM(data=ecuador_five_year_ages,  
  name.disaggregations="province_name",  
  name.age="age",  
  name.sex="sex",  
  name.males="m",  
  name.females="f",  
  name.date1="date1",  
  name.date2="date2",  
  name.population.year1="pop1",  
  name.population.year2="pop2",  
  name.deaths="deaths",  
  deaths.summed=TRUE)
```

```
## [1] "performing GGB-SEG estimation within each of 21 possible age ranges..."
```

We called `EstimateDDM` above using the same arguments as we used with `PlotAgeRatios`, with two additions:

- `name.deaths` provides the name of the variable represented the count of registered deaths between the two dates represented in `name.date1` and `name.date2`
- `deaths.summed`, which should be set to `TRUE` when the `name.deaths` variable represents the total number of registered deaths `name.date1` and `name.date2` and set to `FALSE` when the `name.deaths` variable represents the average number of registered deaths between the two dates.

4.2 Plot DDM estimates: PlotDDM()

We can plot the estimated death registration completeness (using the “hybrid” GGB-SEG method) with the `PlotDDM` function:

```
PlotDDM(ddm_results=ddm_results,  
  label.completeness="Estimated Completeness (%)",  
  label.subnational.levels="Province",  
  plots.dir="Plots/")
```

4.2.1 View estimates of death registration completeness in plot

```
knitr::include_graphics(path=paste0(my_plots_dir,  
  "ddm_point_estimates_combined_province_name_",  
  Sys.Date(),  
  ".pdf"))
```

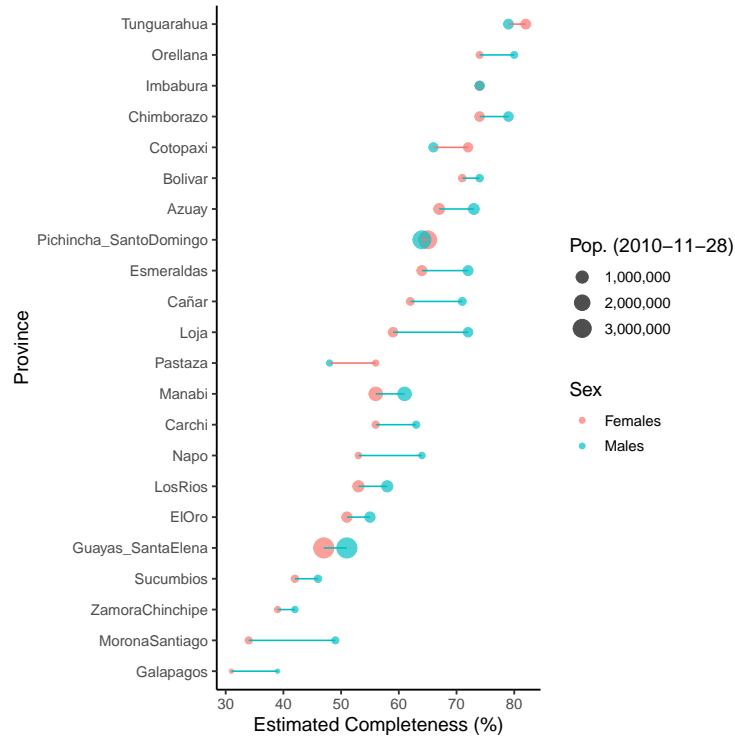


Figure 8: Point estimates of death registration completeness in Ecuador from 2001-2010, using the GGB-SEG method

4.2.2 View DDM point estimates in table

The `EstimateDDM` function also returns a list, in which the `ddm_estimates` object contains the estimated death registration completeness with the GGB-SEG approach

```
head(ddm_results$ggbseg_estimates)
```

##	cod	sex	ggbseg	lower_age_range	upper_age_range	total_pop1	total_pop2
## 1	Azuay	Females	0.67	15	60	599313	710766
## 2	Azuay	Males	0.73	15	65	599313	710766
## 3	Bolivar	Females	0.71	15	65	170696	183742
## 4	Bolivar	Males	0.74	15	65	170696	183742
## 5	Cañar	Females	0.62	15	65	206346	224433
## 6	Cañar	Males	0.71	15	65	206346	224433

4.2.3 View sensitivity of DDM point estimates in plot

Additionally, `PlotDDM` presents a visualization of the GGB-SEG estimates of death-registration completeness to the all permitted values of the age range (i.e. sensitivity) that is selected `DDM::ddm()` as part of the fitting procedure underlying its estimation.

```
knitr::include_graphics(path=paste0(my_plots_dir,
                                     "ddm_sensitivity_province_name_",
                                     Sys.Date(),
                                     ".pdf"))
```

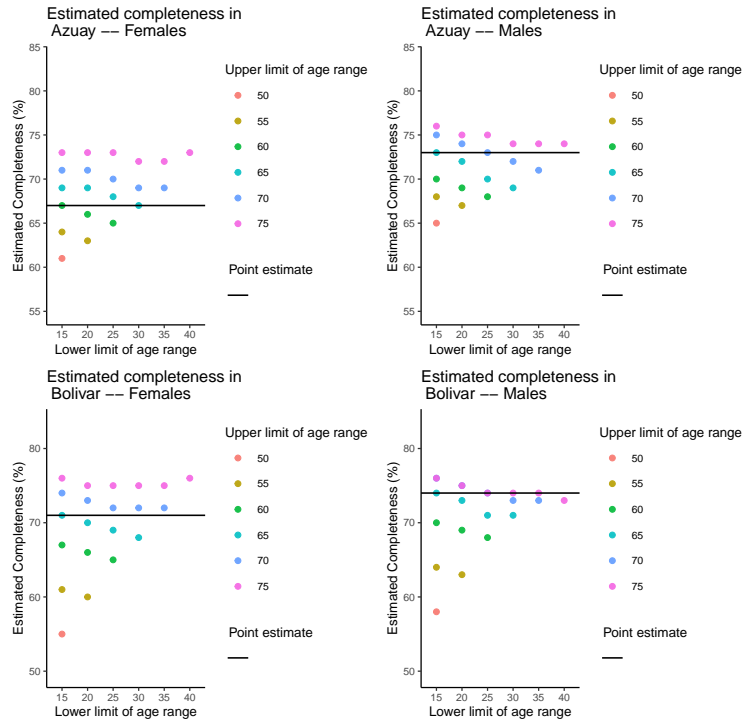


Figure 9: Sensitivity of point estimates of death registration completeness in Ecuador from 2001-2010 to choice of age-range parameter in the GGB-SEG method

4.2.4 View age-range sensitivity of DDM point estimates in table

The sensitivity estimates are also returned in table by `EstimateDDM`, in the `sensitivity_ddm_estimates` element of its list.

```
head(ddm_results$sensitivity_ggbseg_estimates, n=5)
```

##	cod	sex	ggbseg	lower_age_range	upper_age_range	total_pop1	total_pop2
## 1	Azuay	Females	0.61	15	50	599313	710766
## 2	Azuay	Females	0.64	15	55	599313	710766
## 3	Azuay	Females	0.63	20	55	599313	710766
## 4	Azuay	Females	0.67	15	60	599313	710766
## 5	Azuay	Females	0.66	20	60	599313	710766