FLACSO
MÉXICO

Seminario Interinstitucional de Posgrado en Población
"El quehacer demográfico en la solución de los problemas nacionales"

Charla-Taller sobre uso de paquetes demogRáficos

Actividad virtual

Ponente: Ana Escoto
Facultad de Ciencias Políticas y Sociales, UNAM

1 de diciembre de 2023

12:00 hrs

Moderadora: Irene Casique
Centro Regional de Investigaciones Multidisciplinarias, UNAM

bit.ly/YoutubePPCPYS

{paquetes demogRáficos}

Paquetes y códigos para consultar información

Paquetes para análisis demográfico

Paquetes para presentación de información

{paquetes demogRáficos}: consulta población

- wppExplorer-package {wppExplorer}

- También están los paquetes

  - {wpp2022} * en desarrollo
  - {wpp2019}
  - {wpp2017}
  - {wpp2015}
  - {wpp2012}
  - {wpp2010}

{wppExplorer} (I)

```
library(tidyverse)
library(wppExplorer)

#Ejemplo

tfr <- wpp.indicator("fert") # pedir el indicador

tfr %>%
  dplyr::glimpse()
Rows: 7,050
Columns: 3
$ charcode <fct> AF, AL, DZ, AO, AG, AR, AM, AW, AU, AT, AZ, BS,
BH, BD, BB, B…
$ Year     <dbl> 1955, 1955, 1955, 1955, 1955, 1955, 1955, 1955,
1955, 1955, 1…
$ value    <dbl> 7.4500, 6.2300, 7.2780, 6.0000, 4.5000, 3.1540,
4.4940, 5.650…
```

{wppExplorer} (II)

Fitro de país

```
wpp.by.country(tfr,  'MX') %>% head()
     Year value
118  1955  6.75
353  1960  6.78
588  1965  6.75
823  1970  6.75
1058 1975  6.32
1293 1980  5.33
```

{wppExplorer} (III)

Fitro de año

```
wpp.by.year(tfr,  '2010') %>% head()
     charcode  value
2586       AF 6.4784
2587       AL 1.6400
2588       DZ 2.7240
2589       AO 6.3500
2590       AG 2.0000
2591       AR 2.3700
```

{wppExplorer} (IV)

`what` Name of indicator. Possible values are:

fert
: Total fertililty rate. Merges datasets `tfr` and `tfrprojMed`.

leF
: Female life expectancy. Merges datasets `e0F` and `e0Fproj`.

leM
: Male life expectancy. Merges datasets `e0M` and `e0Mproj`.

tpop
: Total population. Uses datasets with age- and sex-specific population counts and agregates over sexes and ages.

tpopF
: Total female population. Uses datasets `popF` and `popFprojMed` and aggregates over ages.

tpopM
: Total male population. Uses datasets `popM` and `popMprojMed` and aggregates over ages.

mig
: Total net migration. For **wpp2019** and **wpp2015**, the dataset `migration` is used. For **wpp2012** and **wpp2010** it aggregates datasets `migrationF` and `migrationM` over ages.

migrate
: Annual migration rate per thousand population. The denominator is approximated with the average population ($(P_t + P_{t-1})/2$).

{wppExplorer} (V)

migrate
: Annual migration rate per thousand population. The denominator is approximated with the average population ($(P_t + P_{t-1})/2$).

popagesex
: Population by sex and age. Uses datasets popM and popMprojMed. It requires two arguments in ..., namely `sexm=c("F", "M")` and
`agem=c("0-4", "5-9", ..., "95-99", "100+")`. The function aggregates population counts over the given sex and age groups.

mortagesex
: Mortality by sex and age. Uses datasets mxF and mxM. It requires two atguments in ..., namely `sex` which is either "F" or "M", and `age` which is one of ("0", "1", "5", "10", "15", "20", ... "95", "100+").

fertage
: Age-specific fertility rate. Uses datasets tfr and tfrprojMed which are merged together and dataset percentASFR to derive age-specific rates. It requires one argument in ..., namely `age` which is one of ("15-19", "20-24", ..., "45-49").

pfertage
: Percent age-specific fertility. Corresponds to the dataset percentASFR. Argument `agem` as defined above giving one or more age categories is required.

sexratio
: Sex ratio at birth. Corresponds to the dataset sexRatio.

medage
: Median age.

meanagechbear
: Mean age at childbearing.

meanageinchbearage
: Mean age of women in childbearing ages.

tdratio
: Total dependency ratio. Ratio of population of age 0 to 14 and 65+ to population of age 15-64.

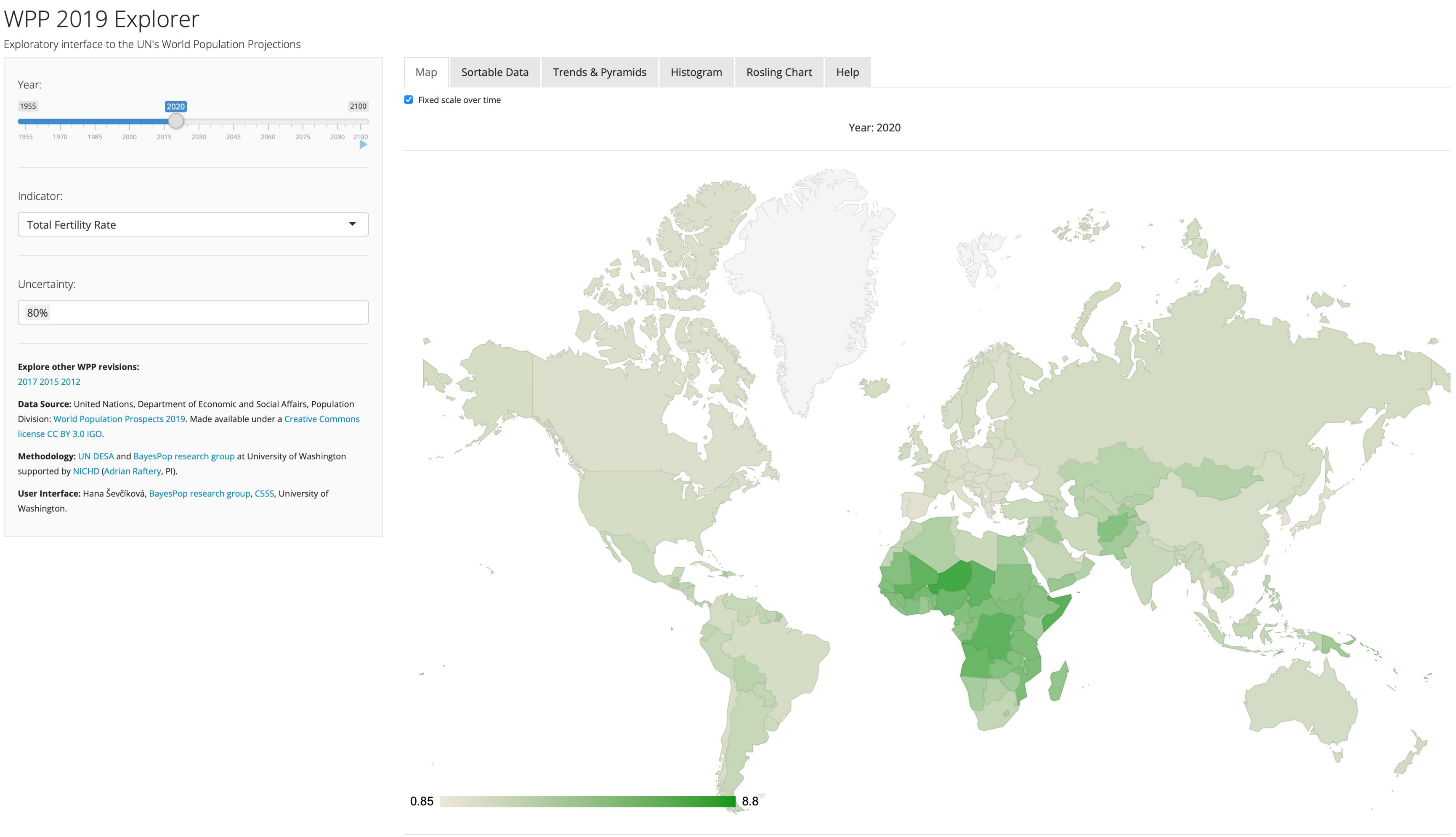
chdratio
: Child dependency ratio. Ratio of population of age 0 to 14 to population of age 15-64.

oadratio
: Old-age dependency ratio. Ratio of population of age 65+ to population of age 15-64.

psratio
: Potential support ratio. Inverse of old-age dependency ratio.

popgrowth
: Average annual population growth ($log(P_t/P_{t-1})/5$).

{wppExplorer} (VI)



https://bayespop.shinyapps.io/wpp2019explorer/

Definition of Projection Scenarios

In addition to the projection for the medium scenario based on probabilistic methods which provide 80 and 95 per cent prediction intervals, the *2022 Revision* includes ten different projection scenarios (see table below) that convey the sensitivity of the medium scenario projection to changes in the underlying assumptions, and to explore the implications of alternative future scenarios of population change. The results for these additional projection scenarios are available from the **Download Center** for the standard projection results in Excel file format as extra worksheets in each workbook file), or CSV file format. Probabilistic projection results are available as a separate set of files, see **Download > Probabilistic Projections** and **Figures**.

Medium scenario projection: in projecting future levels of fertility and mortality, probabilistic methods were used to reflect the uncertainty of the projections based on the historical variability of changes in each variable. The method takes into account the past experience of each country, while also reflecting uncertainty about future changes based on the past experience of other countries under similar conditions. The medium scenario projection corresponds to the median of several thousand distinct trajectories of each demographic component derived using the probabilistic model of the variability in changes over time. Prediction intervals reflect the spread in the distribution of outcomes across the projected trajectories and thus provide an assessment of the uncertainty inherent in the medium scenario projection.

Fertility scenarios: Five of those scenarios differ only with respect to the level of fertility, that is, they share the same assumptions made with respect to sex ratio at birth, mortality and international migration. The five fertility scenarios are: *low, medium, high, constant-fertility and instant-replacement-fertility*. A comparison of the results from these five scenarios allows an assessment of the effects that different fertility assumptions have on other demographic parameters. The high, low, constant-fertility and instant-replacement scenarios differ from the medium scenario only in the projected level of total fertility. In the high scenario, total fertility is projected to reach a fertility level that is 0.5 births above the total fertility in the medium scenario. In the low scenario, total fertility is projected to remain 0.5 births below the total fertility in the medium scenario. In the constant-fertility scenario, total fertility remains constant at the level estimated for 2022. In the instant-replacement scenario, fertility for each country is set to the level necessary to ensure a net reproduction rate of 1.0 starting in 2022. Fertility varies slightly over the projection period in such a way that the net reproduction rate always remains equal to one, thus ensuring the replacement of the population over the long run.

Mortality scenarios: A *constant-mortality scenario*, and a *"no change" scenario* (i.e., both fertility and mortality are kept constant) are available. The constant-mortality scenario uses the same fertility assumption (medium fertility), and international migration assumption as the medium scenario. Consequently, the results of the constant-mortality scenario can be compared with those of the medium scenario to assess the effect that changing mortality has on various population quantities.

Migration scenarios: similarly, the *zero-migration* scenario differs from the medium scenario only with respect to the underlying assumption regarding international migration. Therefore, the zero-migration scenario allows an assessment of the effect that non-zero net migration has on various population quantities. The *instant-replacement zero-migration scenario* combines the fertility assumption of the instant-replacement scenario and migration assumption from the zero-migration scenario. The *"no change" scenario* has the same assumption about international migration as the medium scenario but differs from the latter by having constant fertility and mortality. When compared to the medium scenario, therefore, its results shed light on the effects that changing fertility and mortality have on the results obtained.

The *momentum scenario* illustrates **the impact of age structure on long-term population change** PDF (United Nations, 2017). The scenario combines elements of three existing scenarios: the instant-replacement-fertility scenario, the constant-mortality scenario, and the zero-migration scenario.

World Population Prospects - Population Division - United Nations

{WDI} (I)

```
library(WDI)
WDIsearch('gender') %>% head()
```

```
                indicator
169            2.3_GIR.GPI
172            2.6_PCR.GPI
709      5.51.01.07.gender
1573 BI.EMP.PWRK.PB.FE.ZS
1575 BI.EMP.PWRK.PB.MA.ZS
1587 BI.EMP.TOTL.PB.FE.ZS
                                                                       name
169                        Gender parity index for gross intake ratio in grade 1
172                             Gender parity index for primary completion rate
709                                                             Gender equality
1573  Public sector employment, as a share of paid employment by gender: Female
1575    Public sector employment, as a share of paid employment by gender: Male
1587 Public sector employment, as a share of total employment by gender: Female
```

Metadata Glossary

Code	SG.VAW.REFU.ZS
Indicator Name	Women who believe a husband is justified in beating his wife when she refuses sex with him (%)
Long definition	Percentage of women ages 15-49 who believe a husband/partner is justified in hitting or beating his wife/partner when she refuses sex with him.
Source	Demographic and Health Surveys (DHS), Multiple Indicator Cluster Surveys (MICS), and other surveys
Topic	Violence
Periodicity	Annual
License URL	https://datacatalog.worldbank.org/public-licenses#cc-by
License Type	CC BY-4.0

Go to Data

{WDI} (II)

```
WDI(indicator='SG.VAW.REFU.ZS',
    country="all",
    start=1960,
    end=2020) %>%
  na.omit() %>%
  head()
```

```
         country iso2c iso3c year SG.VAW.REFU.ZS
2995 Afghanistan    AF   AFG 2015           33.4
3053     Albania    AL   ALB 2018            0.9
3062     Albania    AL   ALB 2009            8.9
3066     Albania    AL   ALB 2005            8.7
3299      Angola    AO   AGO 2016           11.5
3482     Armenia    AM   ARM 2016            0.8
```

{geodata}

Function	Description
`cmip6_world`, `cmip6_tile`	Downscaled and calibrated CMIP6 projected future climate data
`country_codes`	Country codes
`crop_calendar_sacks`	Crop calendar data by Sacks et al
`crop_monfreda`	Crop area and yield data for 175 crops by Monfreda et al.
`crop_spam`	MapSPAM crop data (area, yield, value)
`cropland`	Cropland density for the world derived from different sources (ESA, GLAD, QED)
`elevation_3s`, `elevation_30s`, `elevation_global`	Elevation data
`gadm`, `world`	Administrative boundaries for any country, or the entire world from GADM
`landcover`	Landcover data derived from ESA WorldCover
`footprint`	Human footprint data from the Last of the Wild project
`osm`	OpenStreetMap data by country (places and roads)
`population`	Population density Gridded Population of the World
`soil_af`	Chemical and physical soil properties data for Africa for different soil depths
`soil_af_elements`	Connect to or download chemical soil element concentration (for the 0-30 cm topsoil) data for Africa
`soil_af_water`	Physical soil properties data for Africa for water balance computation
`soil_af_isda`	Soil data for Africa derived from the iDSA data set
`soil_world_vsi`	Virtually connect to the global SoilGrids data
`soil_world`	Global soils data from SoilGrids
`sp_occurrence`	Species occurrence records from the Global Biodiversity Information Facility
`travel_time`	Travel time to and from cities and ports by Nelson et al.
`worldclim_global`, `worldclim_country`, `worldclim_tile`	WorldClim glocal climate data

International Journal of Climatology RMetS

Research Article

WorldClim 2: new 1-km spatial resolution climate surfaces for global land areas

Stephen E. Fick, Robert J. Hijmans

First published: 15 May 2017 | https://doi.org/10.1002/joc.5086 | Citations: 5,051

Read the full text › PDF TOOLS SHA

ABSTRACT

We created a new dataset of spatially interpolated monthly climate data for global land areas at a very high spatial resolution (approximately 1 km^2). We included monthly temperature (minimum, maximum and average), precipitation, solar radiation, vapour pressure and wind speed, aggregated across a target temporal range of 1970–2000, using data from between 9000 and 60 000 weather stations. Weather station data were interpolated using thin-plate splines with covariates including elevation, distance to the coast and three satellite-derived covariates: maximum and minimum land surface temperature as well as cloud cover, obtained with the MODIS satellite platform. Interpolation was done for 23 regions of varying size depending on station density. Satellite data improved prediction accuracy for temperature variables 5–15% (0.07–0.17 °C), particularly for areas with a low station density, although prediction error remained high in such regions for all climate variables. Contributions of satellite covariates were mostly negligible for the other variables, although their importance varied by region. In contrast to the common approach to use a single model formulatio for the entire world, we constructed the final product by selecting the best performing model for each region and variable. Global cross-validation correlations were ≥0.99 for temperature and humidity, 0.86 for precipitation and 0.76 for wind speed. The fact that most of our climate surface estimates were only marginally improved by use of satellite

{geodata} (I)

The {WorldClim} database

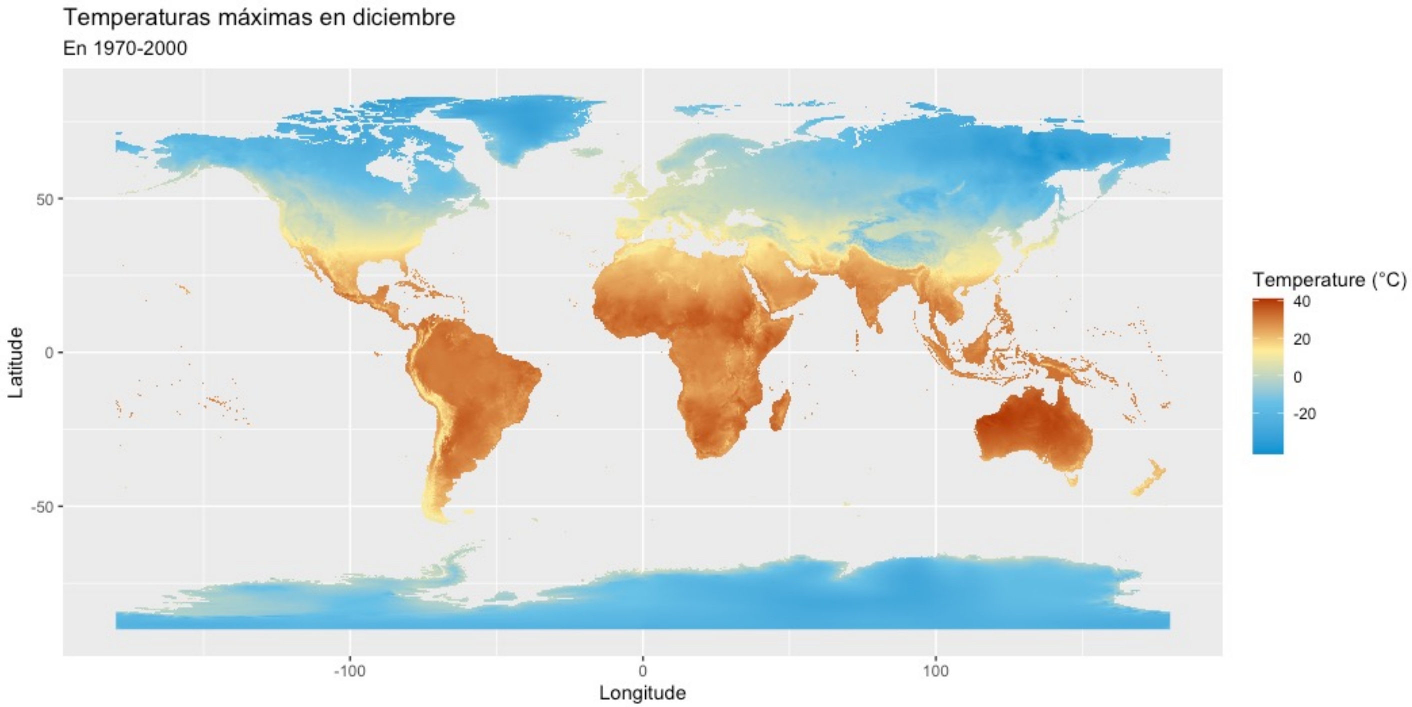
```
library(geodata)
tmax_data <- worldclim_global(var = "tmax",
                              res = 10,
                              path= getwd())

tmax_data
```

```
class       : SpatRaster
dimensions  : 1080, 2160, 12  (nrow, ncol, nlyr)
resolution  : 0.1666667, 0.1666667  (x, y)
extent      : -180, 180, -90, 90  (xmin, xmax, ymin, ymax)
coord. ref. : lon/lat WGS 84 (EPSG:4326)
sources     : wc2.1_10m_tmax_01.tif
              wc2.1_10m_tmax_02.tif
              wc2.1_10m_tmax_03.tif
              ... and 9 more source(s)
names       : wc2.1~ax_01, wc2.1~ax_02, wc2.1~ax_03, wc2.1~ax_04, wc2.1~ax_05, wc2.1~ax_06, ...
min values  :     -42.419,   -39.58325,   -53.40000,   -59.54875,    -59.8395,    -60.3600, ...
max values  :      42.157,    40.26450,    41.48825,    43.17525,     44.8155,     46.6155, ...
```

{geodata} (II)

¿Qué tanto calor hace en diciembre?

```
library(ggplot2)

# Converting the raster object into a
dataframe
tmax_data_may_df <-
as.data.frame(tmax_data$wc2.1_10m_tmax_
12, xy = TRUE, na.rm = TRUE)
rownames(tmax_data_may_df) <- c()

ggplot(
  data = tmax_data_may_df,
  aes(x = x, y = y)
) +
  geom_raster(aes(fill =
wc2.1_10m_tmax_12)) +
  labs(
    title = "Temperaturas máximas en
diciembre",
    subtitle = "En 1970-2000"
  ) +
  xlab("Longitude") +
  ylab("Latitude") +
  scale_fill_gradientn(
    name = "Temperature (°C)",
    colours = c("#0094D1", "#68C1E6",
"#FEED99", "#AF3301"),
    breaks = c(-20, 0, 20, 40)
  )
```



{geodata} (III)

También se puede descargar la “Gridded Population of the World” (Una base para densidad poblacional de CIESIN)

```
pop <- population(2020, res=10, path=getwd())
pop
class       : SpatRaster
dimensions  : 1080, 2160, 1  (nrow, ncol, nlyr)
resolution  : 0.1666667, 0.1666667  (x, y)
extent      : -180, 180, -90, 90  (xmin, xmax, ymin, ymax)
coord. ref. : lon/lat WGS 84 (EPSG:4326)
source      : gpw_v4_population_density_rev11_2020_10m.tif
name        : population_density
min value   :               0.00
max value   :           64552.77
```

Nos bajo un tiff con la información

{inegiR} + API INEGI (I)

El paquete inegiR permite revisar los datos del BIE

```
library(inegiR)
inegiR::inegi_series(serie=628194,# este el IPC
                     token = token) %>%
  head()
```

```
        date date_shortcut  values notes
1 2022-10-01           M10 125.276    NA
2 2022-09-01            M9 124.571    NA
3 2022-08-01            M8 123.803    NA
4 2022-07-01            M7 122.948    NA
5 2022-06-01            M6 122.044    NA
6 2022-05-01            M5 121.022    NA
```

https://www.inegi.org.mx/app/desarrolladores/generatoken/Usuarios/token_Verify

{inegiR} + API INEGI (II)

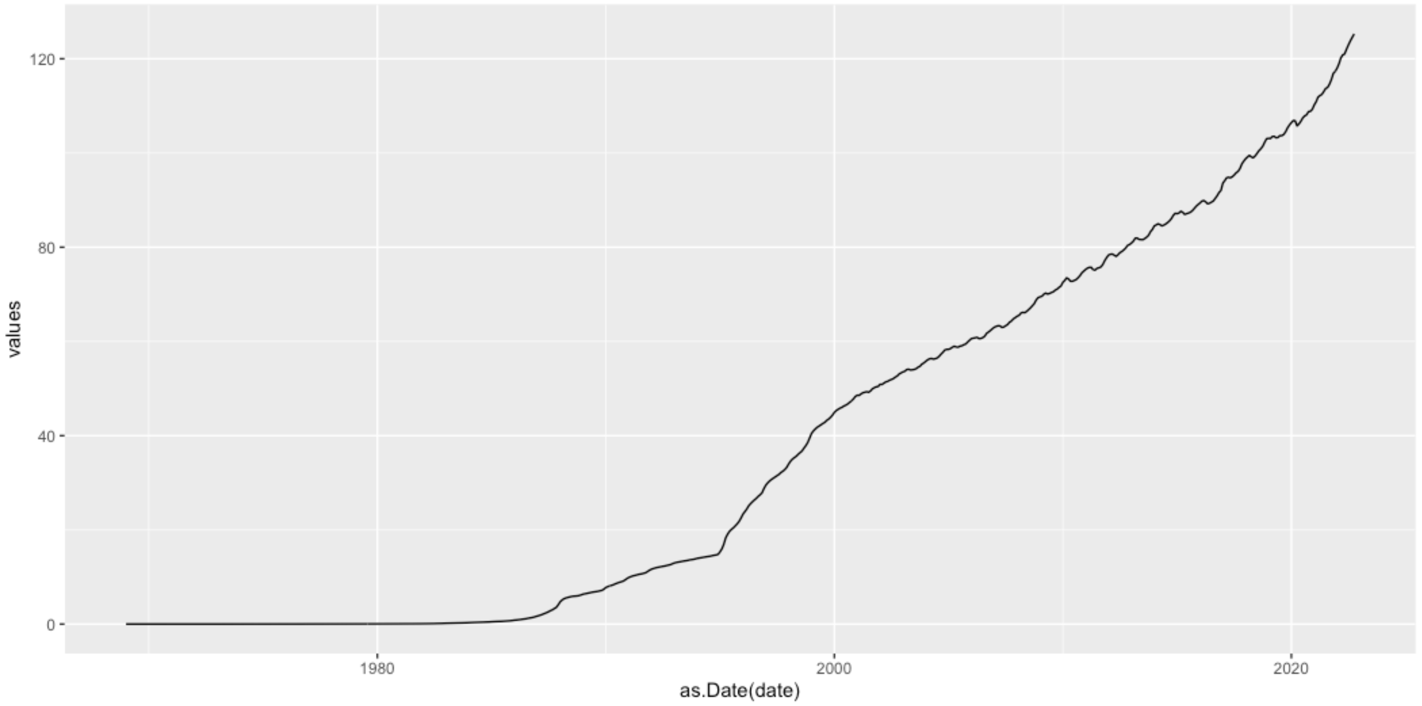
Guardamos esto como un objeto

```
# Podemos guardar esta consulta en un objeto y luego
graficar

consulta1<-inegi_series(serie=628194, # IPC mensual
                        token = token)

str(consulta1)
'data.frame':   646 obs. of  4 variables:
 $ date         : Date, format: "2022-10-01" "2022-09-01"
...
 $ date_shortcut: chr  "M10" "M9" "M8" "M7" ...
 $ values       : num  125 125 124 123 122 ...
 $ notes        : logi  NA NA NA NA NA NA ...
```

{inegiR} + API INEGI (III)

Una fácil gráfica

```
consulta1 %>%
  ggplot()+
  aes(x=as.Date(date),
      y=values) +
  geom_line()
```



{paquetes demogRáficos}: otros paquetes con API

- Necesitan token y/o inscripción
- IPUMS internacional **{ipumsR}**
- Dataset Management for the Demographic and Health Survey (DHS) Data **{rdhs}**
- API Banxico **{siebanxicor}**

{paquetes demogRáficos}

Paquetes y códigos para consultar información

Paquetes para análisis demográfico

Paquetes para presentación de información

{apyramid} (I)

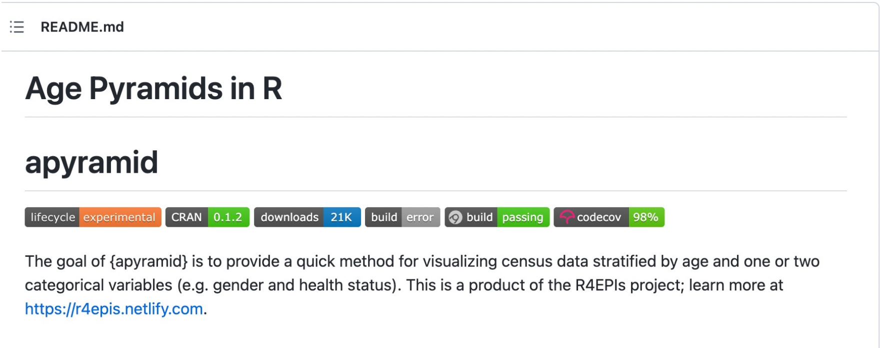
Una pirámide muy sencilla, primero descargo mi población, pero usando directamente wpp2019

```r
library(wpp2019)
data(popF)
data(popM)
```

Me salto un par de pasos hasta tener una base para México

```r
popmex2020 # en formato long
```

```
# A tibble: 42 × 8
   country_code name   age   anio  sex     poblacion  edad edad_factor
          <int> <chr>  <chr> <chr> <chr>       <dbl> <dbl> <fct>
 1          484 Mexico 0-4   2020  hombres   5605.       0 0
 2          484 Mexico 0-4   2020  mujeres   5354.       0 0
 3          484 Mexico 10-14 2020  hombres   5693.      10 10
 4          484 Mexico 10-14 2020  mujeres   5448.      10 10
 5          484 Mexico 100+  2020  hombres      4.72   100 100
 6          484 Mexico 100+  2020  mujeres      8.24   100 100
 7          484 Mexico 15-19 2020  hombres   5695.      15 15
 8          484 Mexico 15-19 2020  mujeres   5515.      15 15
 9          484 Mexico 20-24 2020  hombres   5505.      20 20
10          484 Mexico 20-24 2020  mujeres   5435.      20 20
# … with 32 more rows
```


README.md

Age Pyramids in R

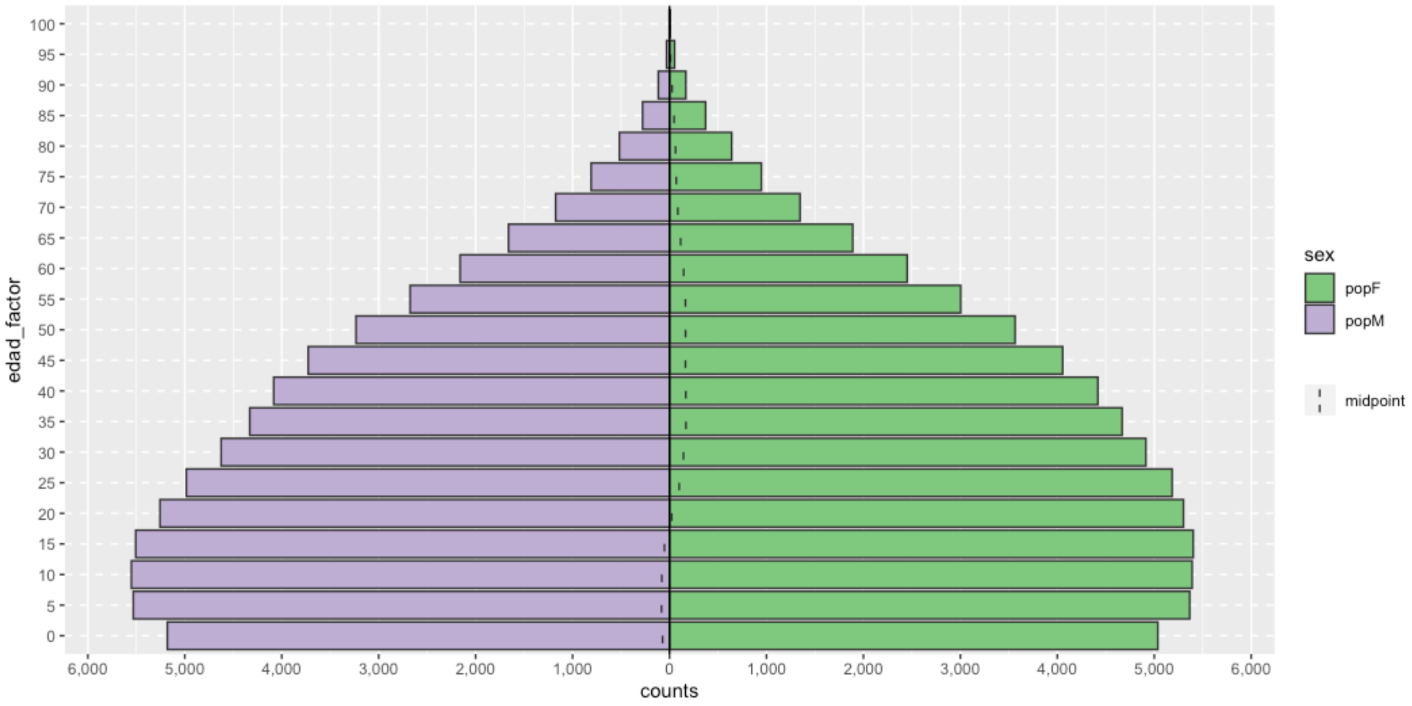
apyramid

lifecycle experimental | CRAN 0.1.2 | downloads 21K | build error | build passing | codecov 98%

The goal of {apyramid} is to provide a quick method for visualizing census data stratified by age and one or two categorical variables (e.g. gender and health status). This is a product of the R4EPIs project; learn more at https://r4epis.netlify.com.

{apyramid} (II)

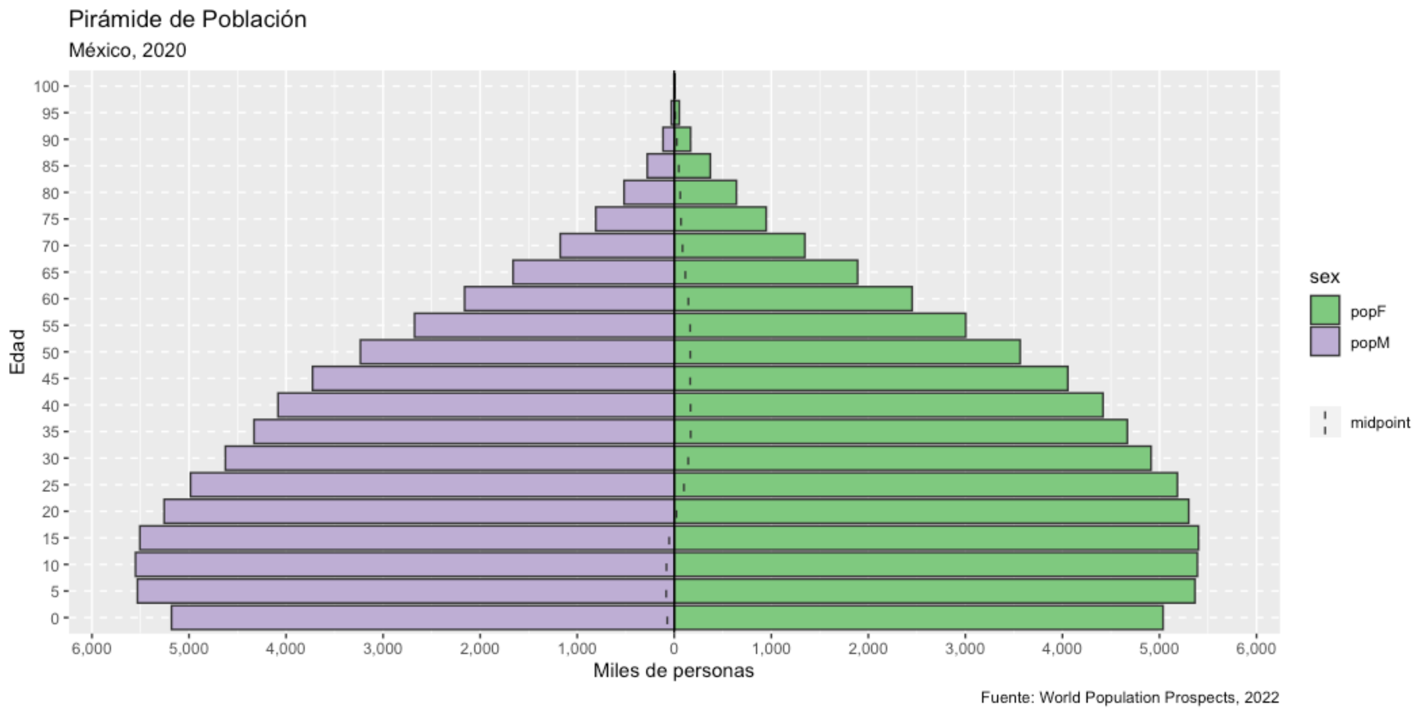
```
library(apyramid)

popmex2020 %>%
age_pyramid(edad_factor, # edad
              split_by = sex,
              count=poblacion
```



{apyramid} (III)

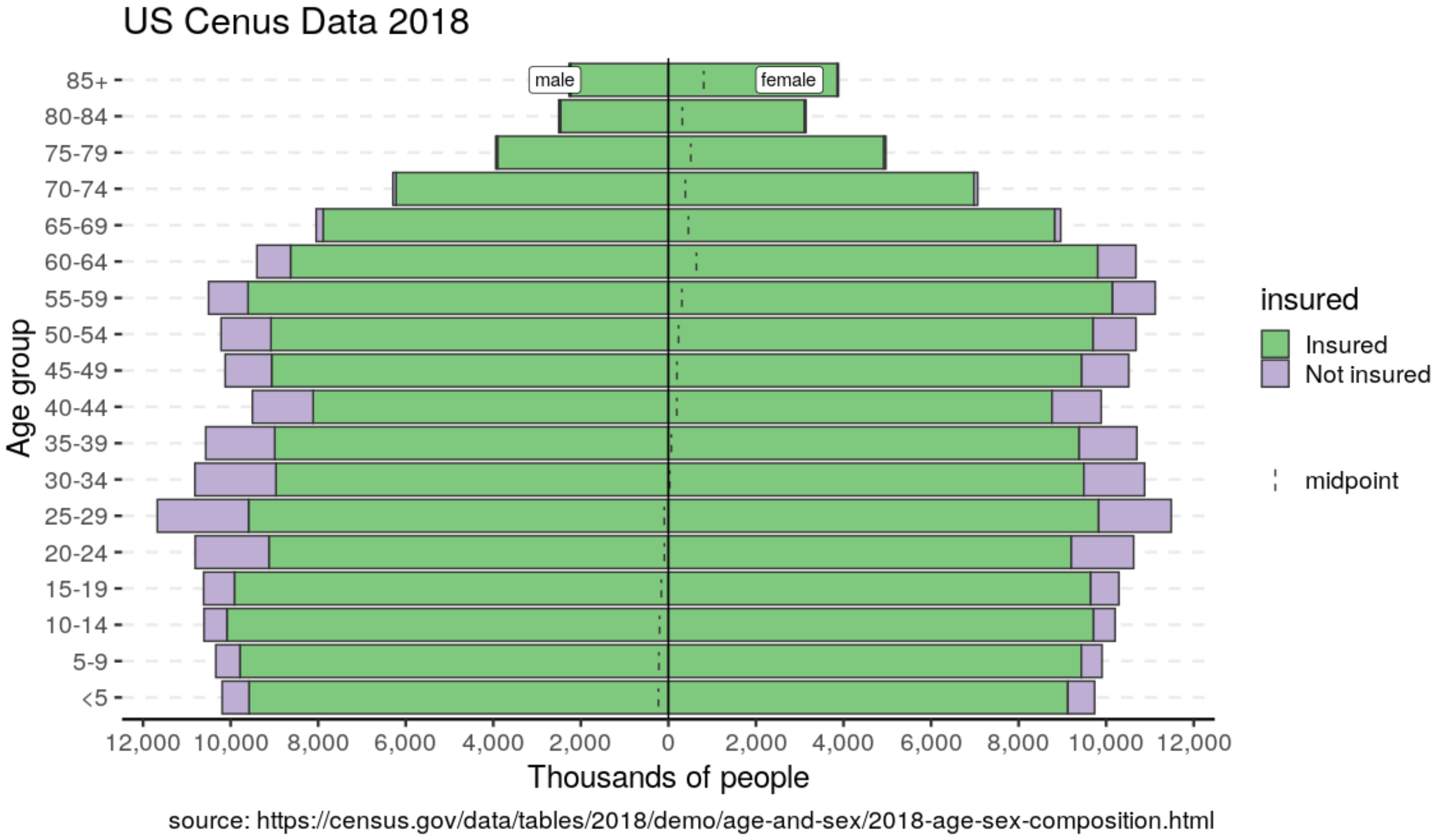
Una ventaja es que es compatible con {ggplot2}

```
popmex2020 %>%
age_pyramid(edad_factor, # edad
              split_by = sex,
              count=poblacion) + # sexo
  labs(title="Pirámide de Población",
       subtitle = "México, 2020",
       y ="Miles de personas", # ojo
       x = "Edad", # ojo
       caption = "Fuente: World Population Prospects, 2022" )
```

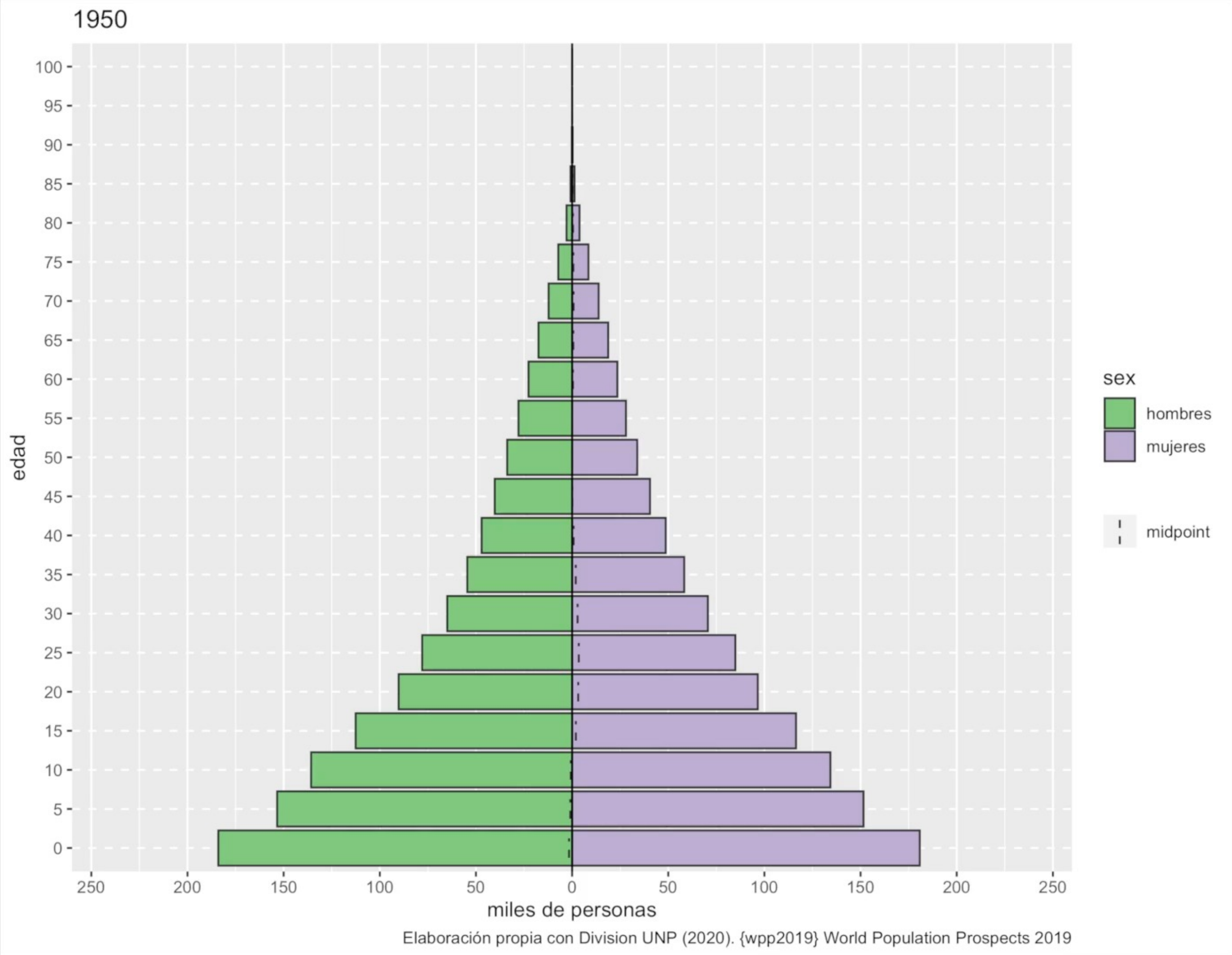


Hay una opción "stack_by" que permite hacer un stack si tenemos dos series

{apyramid} (IV)



Hay una opción "stack_by" que permite
hacer un stack si tenemos dos series


1950
edad
100
95
90
85
80
75
70
65
60
55
50
45
40
35
30
25
20
15
10
5
0
250 200 150 100 50 0 50 100 150 200 250
miles de personas
sex
hombres
mujeres
midpoint
Elaboración propia con Division UNP (2020). {wpp2019} World Population Prospects 2019

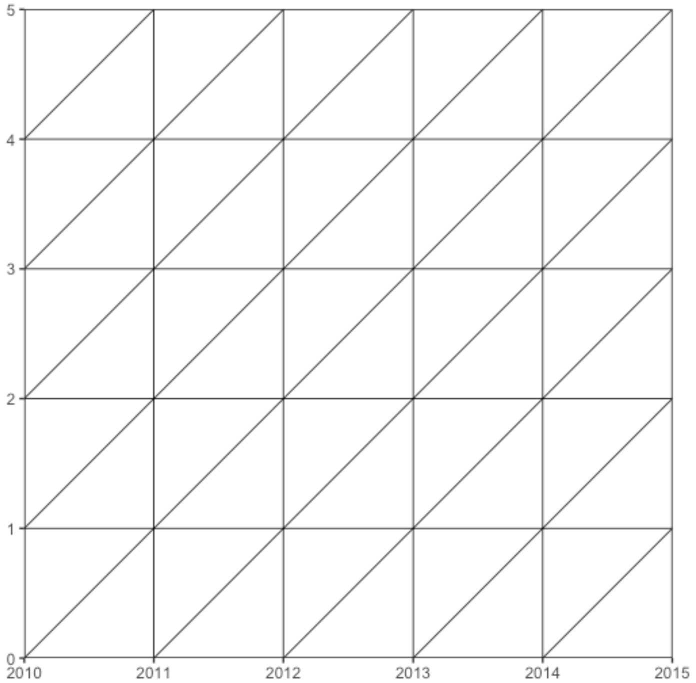
LexisPlotR (I)

Una herramienta muy útil son los diagramas de Lexis

```r
library(LexisPlotR)
# Dibuje una cuadrícula de Lexis desde el año 2010 hasta el año 2015,
# que representa las edades de 0 a 5 años.

lexis_grid(year_start = 2010,
           year_end = 2015,
           age_start = 0,
           age_end = 5)
```

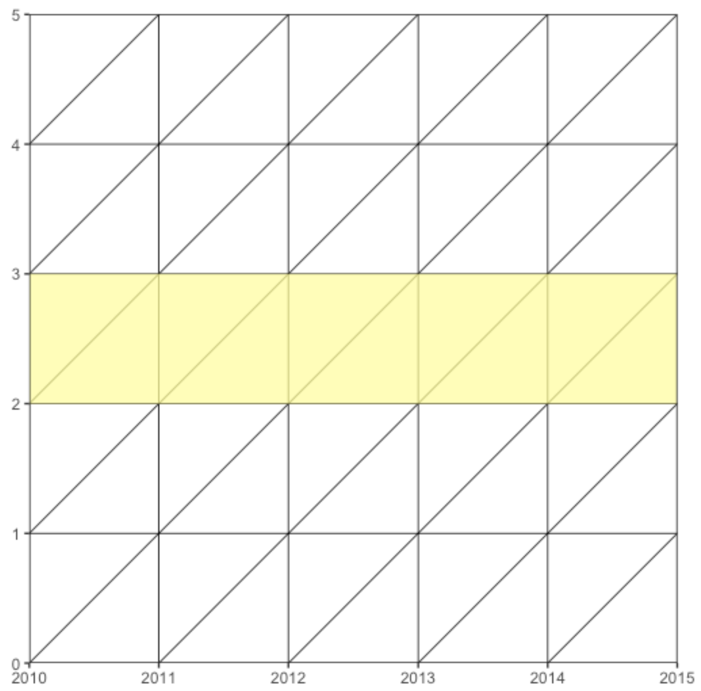
LexisPlotR

CRAN 0.4.0 | downloads 594/month | build passing | build passing



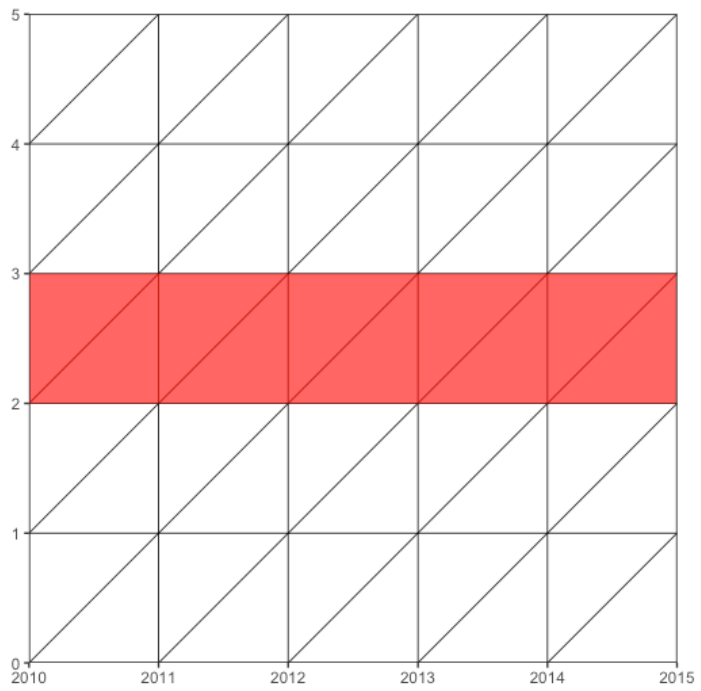
LexisPlotR(II)

```
lexis_grid(year_start = 2010,
           year_end = 2015,
           age_start = 0,
           age_end = 5) %>%
  lexis_age(age = 2)
```



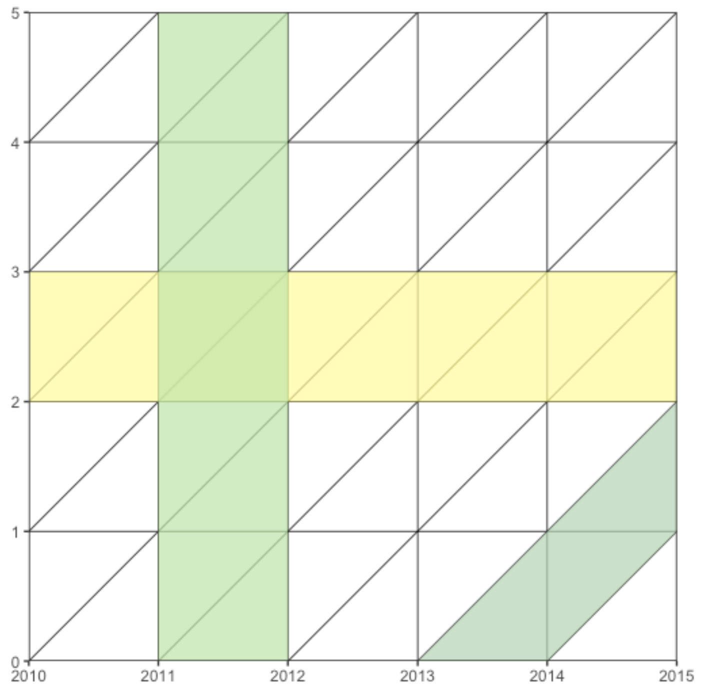
LexisPlotR (III)

```
lexis_grid(year_start = 2010,
           year_end = 2015,
           age_start = 0,
           age_end = 5) %>%
  lexis_age(age = 2,
            fill = "red")
```

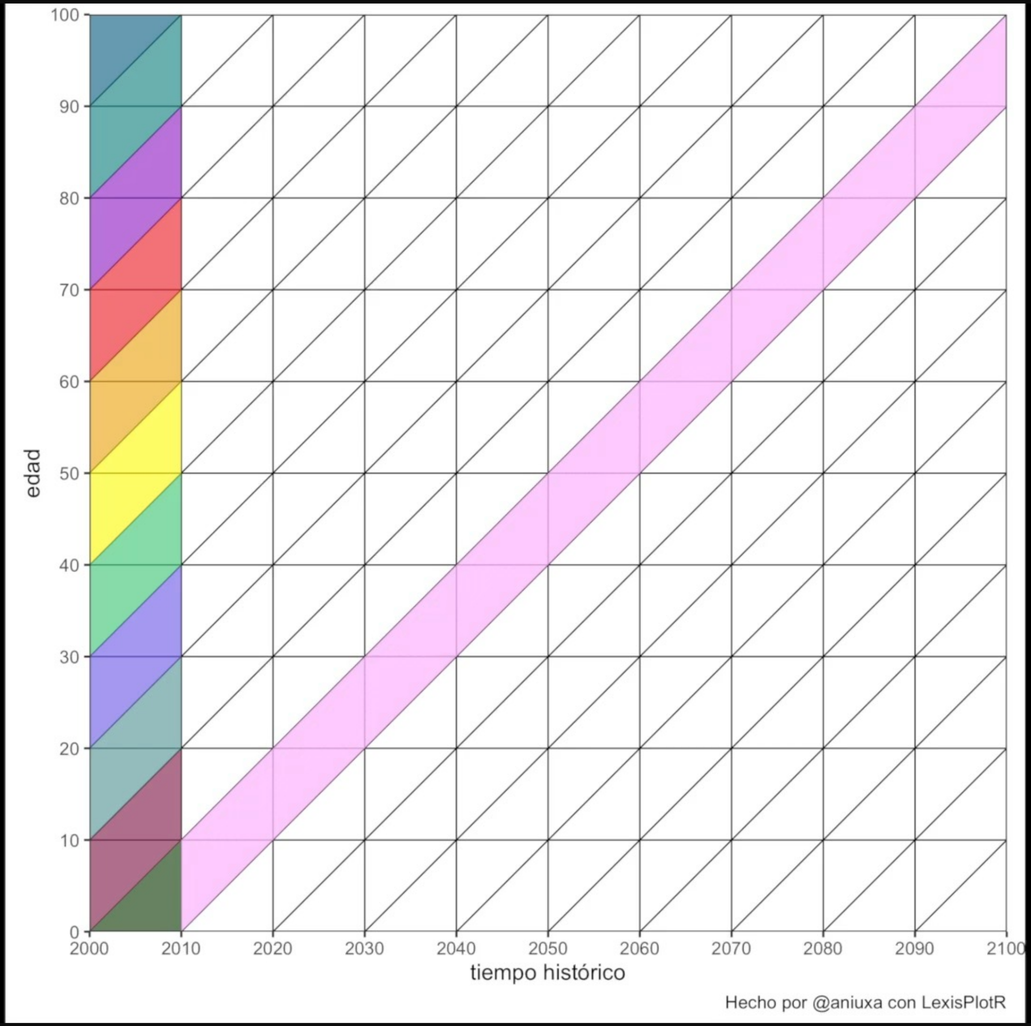


LexisPlotR (IV)

```
lexis_grid(year_start = 2010,
           year_end = 2015,
           age_start = 0,
           age_end = 5) %>%
  lexis_age(age = 2) %>%
  lexis_cohort(cohort = 2013) %>%
  lexis_year(year = 2011)
```



LexisPlotR (V)



{fmsb} (I)

- Este paquete tiene cosas muy interesantes. Es un paquete no sólo para demografía pero permite ajustar algunas funciones demográficas
- **Limitantes**: como que está en japonés :P
- Un ejemplo con el índice de Whipple, que mide la atracción digital. Necesitamos datos en edades singulares:

```
readxl::read_excel("sv1992.xlsx") %>%
  head() %>%
  janitor::clean_names()# datos de
ipums
```

```
# A tibble: 6 × 4
  age    male female unknown
  <chr> <dbl>  <dbl>   <dbl>
1 0      6093   6113       0
2 1      6089   5795       0
3 2      6805   6737       0
4 3      7028   6699       0
5 4      7294   6965       0
6 5      6628   6408       0
```

{fmsb} (II)

人口モデルに関連する関数群(Functions related to demographic models)

いくつかの人口モデルの関数定義を含む，人口分析に関連した関数群。生命表関連の関数群の中では，lxがl(0)=100000であることに注意。
Several functions related to demographic analyses, especially including the definitions of demographic models. Among lifetable functions, note that l(0) is 100000 for lx.

- Gompertz-Makehamの死亡モデル(GompertzMakeham, fitGM): GompertzMakeham()は３パラメータのGompertz-Makehamモデルの死力関数μ(x)を返す。
  The force of mortality μ(x) generated by the 3 parameters Gompertz-Makeham model.
  fitGM()はoptim()関数によりGompertz-Makehamモデルのqxへの当てはめの残差の平均平方和の平方根(RMSE)を最小にするパラメータを自動探索する
  Automatically find the parameters which minimize the root mean square error (RMSE) of fitting Gompertz-Makeham model into qx, using optim().
- Silerの死亡モデル(Siler, fitSiler): Siler()は５パラメータのSilerモデルのqxを返す。
  qx generated by the 5 parameters Siler model.
  fitSiler()はoptim()関数によりSilerモデルのqxへの当てはめの残差の平均平方和の平方根(RMSE)を最小にするパラメータを自動探索する。
- Dennyの死亡モデル(Denny, fitDenny): Denny()は３パラメータのDennyモデルのl(x)を返す。l(0)は100000であることに注意。
  Returns l(x) of Denny's model with 3 parameters. Note that l(0) is 100000.
  fitDenny()は，optim()関数によりDennyモデルのlxへの当てはめの残差の平均平方和の平方根(RMSE)を最小にするパラメータを自動探索する。
- Coale-McNeilの結婚モデル(CM, fitCM): CM()は３パラメータのCoale-McNeilモデルによる年齢別初婚確率と既婚率を返す。
  fitCM()はoptim()関数によりCoale-McNeilモデルの年齢別初婚確率または既婚率データへの当てはめの残差の平均平方和の平方根(RMSE)を最小にするパラメータを自動探索する。
- Coale-Trussellの出生モデル(CT, fitCT): CT()は２パラメータのCoale-Trussellモデルによる年齢別有配偶出生率(ASMFR)を返す。fitCT()はoptim()関数によりCoale-TrussellモデルのASMFRデータへの当てはめの残差の平均平方和の平方根(RMSE)を最小にするパラメータを自動探索する。
- Hadwigerの出生モデル(Hadwiger, fitHad): Hadwiger()は３パラメータのHadwigerモデルによる年齢別出生率(ASFR)を返す。fitHad()はoptim()関数によりHadwigerモデルのASFRデータへの当てはめの残差の平均平方和の平方根(RMSE)を最小にするパラメータを自動探索する。
- 生命表関数群(lifetable, ...): lifetable()はmxが与えられると生命表全体を計算しデータフレームとして返す。
  For given mx, calculating the whole lifetable and return it as a dataframe.
  dxtolx(), lxtoqx(), uxtoqx()など多様な変換関数を含む。
- Whippleの指標(WhipplesIndex): 年齢の誤報のうちAge-heapingと呼ばれる偏りを評価するための指標。1桁目が5や0で終わる切りの良い数字への偏りの程度を示す。
- 介護者比(CaretakerRatio): 50歳以上65歳未満の女性を主たる介護者，80歳以上の高齢者人口を介護される人口と考えて，主たる介護者人口に対する介護対象者の比を求めた値。
- 2つの人口構造の差異指数(IndexOfDissimilarity): 2つの人口構造の非類似度の指標として提案された，それぞれの年齢構成割合を求め，その差の絶対値を全年齢について合計してから0.5を掛けた値。
- ふくらみ指数(PEI): 黒田俊夫に提案された，人口構造データから計算される人口移動の指標。

fmsbパッケージマニュアル (sip21c.org)

{fmsb} (III)

```
library(fmsb)

sv1992 %>%
  filter(!age>64) %>%
  count(age, wt=male) %>%
  with(
    WhipplesIndex(n)
    )
```

```
$WI
[1] 127.154


$JUDGE
[1] "rough"
```

{demography}

Leyendo desde demography

Una vez que ya tenemos listos nuestros dos archivos de insumos, podemos leer esta información con el comando read.demogdata().

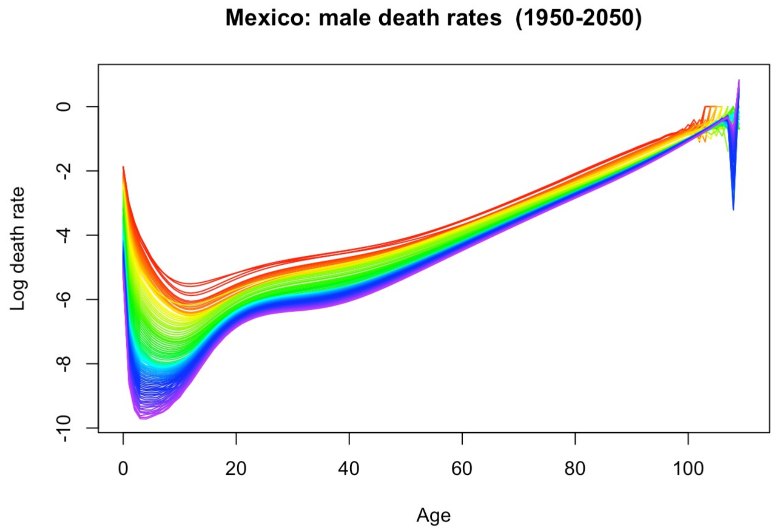
Revisemos los argumentos:

```
mex<-read.demogdata("mx.txt",
                    "exposure.txt",
                    type="mortality",
                    label="Mexico",
                    skip=0,
                    popskip = 0,
                    scale=1)
```

Una vez que ya tenemos este objeto, podemos utilizarlo fácilmente en distintos comandos del paquete. Revisaremos un par

Gráfico de las tasas de mortalidad por sexo

```
plot(mex,series="Male", plot.type="functions")
```



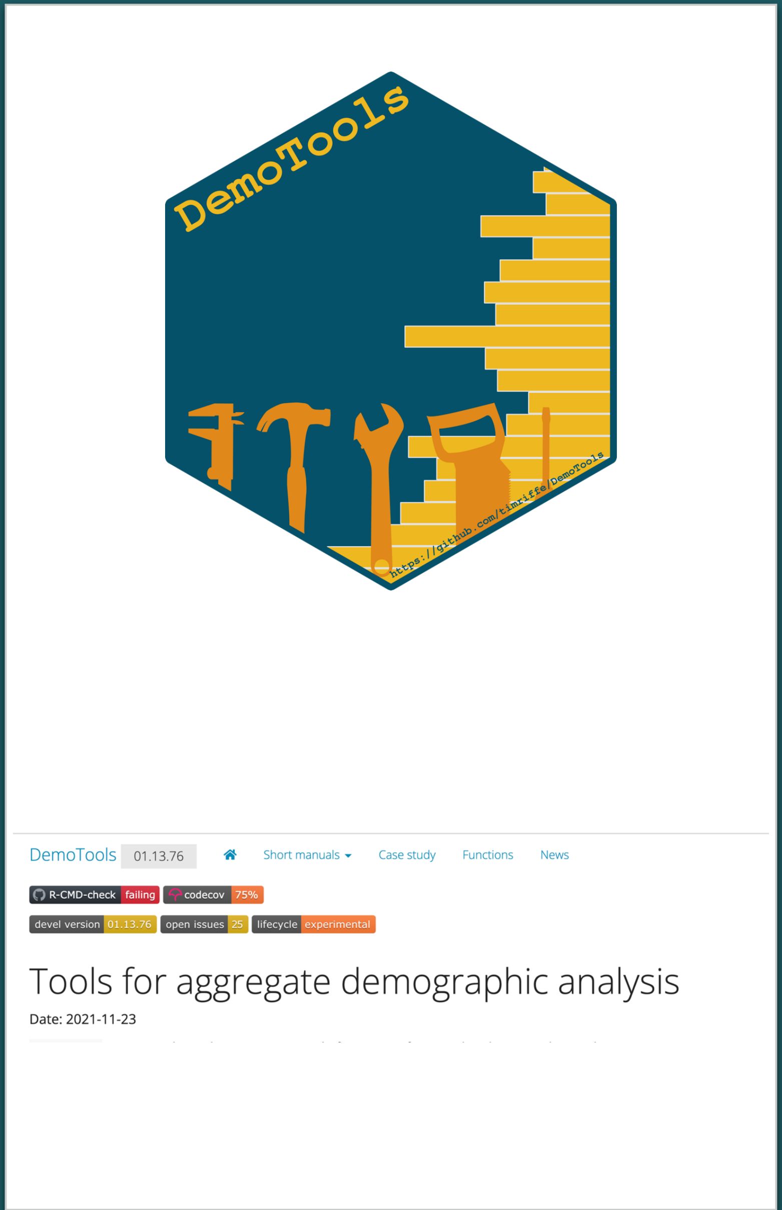
README.md

demography

The R package `demography` provides functions for demographic analysis including: lifetable calculations; Lee-Carter modelling; functional data analysis of mortality rates, fertility rates, net migration numbers; and stochastic population forecasting.

- Lo más complicado es pasar nuestros datos al formato del Human Mortality Database.

[Práctica 13. Aplicaciones relacionadas a la demografía (IV) (aniuxa.github.io)]



{DemoTools} (I)

- Es quizás el paquete más completo. Todavía no está en CRAN. Así que su instalación se debe hacer desde su versión en desarrollo y además intalar Stan

```
install.packages("rstan",
      repos =c("https://mc-stan.org/r-packages/",
      getOption("repos")))

remotes::install_github("timriffe/DemoTools")

library(DemoTools)
```

{DemoTools} (II)

Índices de atracción

```r
### Whipple

check_heaping_whipple(Value=sv1992$male,
                      Age= sv1992$age,
                      ageMin = 25,
                      ageMax = 60,
                      digit = c(0, 5))
```

```
[1] 1.27154
```

```r
### Noumbissi

check_heaping_noumbissi(sv1992$male,
                        Age=sv1992$age,
                        ageMin = 30,
                        ageMax = 60,
                        digit = 0)
```

```
[1] 1.411808
```

{DemoTools} (III)

Tablas de vida

Este paquete nos da la oportunidad de construir las tablas de vida con diferentes insumos, con **cualquiera** de las siguientes opciones:

- Vector de muertes y vector de Población media
- Vector de tasas de Mortalidad (nMx)
- Vector de cocientes de mortalidad (nqx)
- Vector de efectivos a edad exacta (lx)

{DemoTools} (IV)

Tablas de vida

```r
# Tabla abreviada ----

## Input: nMx ----

#Datos de México 2000
nMx <- c(0.025429618,
         0.000895531,
         0.000364678,
         0.000480071,
         0.000979976,
         0.001661119,
         0.002167313,
         0.002549786,
         0.00307099,
         0.003970018,
         0.005461053,
         0.007799417,
         0.011317907,
         0.016516166,
         0.024145341,
         0.035168272,
         0.051143602,
         0.074042144,
         0.136811785)
```

{DemoTools} (V)

Tablas de vida

```r
round(mx_lifetable2000, digits = 2)  %>%
  head()
```

	Age	AgeInt	nMx	nAx	nqx	lx	ndx	nLx	Sx	Tx	ex
0	0	1	0.03	0.11	0.02	100000.00	2486.76	97789.83	0.97	7214185	**72.14**
1	1	4	0.00	1.58	0.00	97513.24	348.55	389208.81	1.00	7116395	72.98
5	5	5	0.00	2.50	0.00	97164.69	177.01	485380.95	1.00	6727186	69.23
10	10	5	0.00	2.50	0.00	96987.69	232.53	484357.11	1.00	6241805	64.36
15	15	5	0.00	2.76	0.00	96755.16	473.05	482714.55	0.99	5757448	59.51
20	20	5	0.00	2.66	0.01	96282.11	796.59	479548.05	0.99	5274734	54.78

{DemoTools} (VI)

Otras funciones

¡No da tiempo para todas!

Este paquete puede producir tablas abreviadas, tablas singulares y extrapolar hasta 109 años

También hay funciones de suavizamiento de información.

{DemoTools} (VII)

Reference

Evaluate age structure

Indicators to evaluate the degreee of age heaping

`check_heaping_bachi()`	calculate Bachi's index of age heaping
`check_heaping_coale_li()`	Coale-Li age heaping index
`check_heaping_jdanov()`	Calculate Jdanov's old-age heaping index
`check_heaping_kannisto()`	Kannisto's age heaping index
`check_heaping_myers()`	Calculate Myer's blended index of age heaping
`check_heaping_noumbissi()`	calculate Noumbissi's digit heaping index
`check_heaping_roughness()`	Evaluate roughness of data in 5-year age groups
`check_heaping_sawtooth()`	Detect if heaping is worse on terminal digits 0s than on 5s
`check_heaping_spoorenberg()`	Spoorenberg's total modified Whipple index
`check_heaping_whipple()`	Calculate Whipple's index of age heaping

Evaluate consistency of age structures

Methods to assess the consistency of age structures

`ageRatioScore()`	Calculate the PAS age ratio score
`sexRatioScore()`	Calculate the PAS sex ratio score

Contents

[Function reference • DemoTools (timriffe.github.io)](https://timriffe.github.io)

{migest} (I)

migest

Tools for estimating, measuring and working with migration data.

See the pkgdown site for more information

Installation

You can install the released version of migest from CRAN with:

```
install.packages("migest")
```

And the development version from GitHub with:

```
# install.packages("devtools")
devtools::install_github("guyabel/migest")
```

{migest} (II)

¿Qué necesitamos?

```
library(migest)
library(tidyverse)
library(countrycode)
# download Abel and Cohen (2019) estimates

f <- read_csv("https://ndownloader.figshare.com/files/38016762", show_col_types = FALSE)
f
```

```
# A tibble: 307,833 × 9
   year0 orig  dest  sd_drop_neg sd_rev_neg mig_rate da_min_open da_min_closed
   <dbl> <chr> <chr>       <dbl>      <dbl>    <dbl>       <dbl>         <dbl>
 1  1990 BDI   BDI             0          0      0          0                0
 2  1990 COM   BDI             0          0      0          0                0
 3  1990 DJI   BDI             0          0      0          0                0
 4  1990 ERI   BDI             0          0      0          0                0
 5  1990 ETH   BDI             0          0      0          0                0
 6  1990 KEN   BDI            30         30     75.7       51.3            207.
 7  1990 MDG   BDI             0          0      0          0.03             0
 8  1990 MWI   BDI             0          0      0          0                0
 9  1990 MUS   BDI             0          0      0          0.06             0
10  1990 MYT   BDI             0          0      0          0                0
# ℹ 307,823 more rows
# ℹ 1 more variable: da_pb_closed <dbl>
```

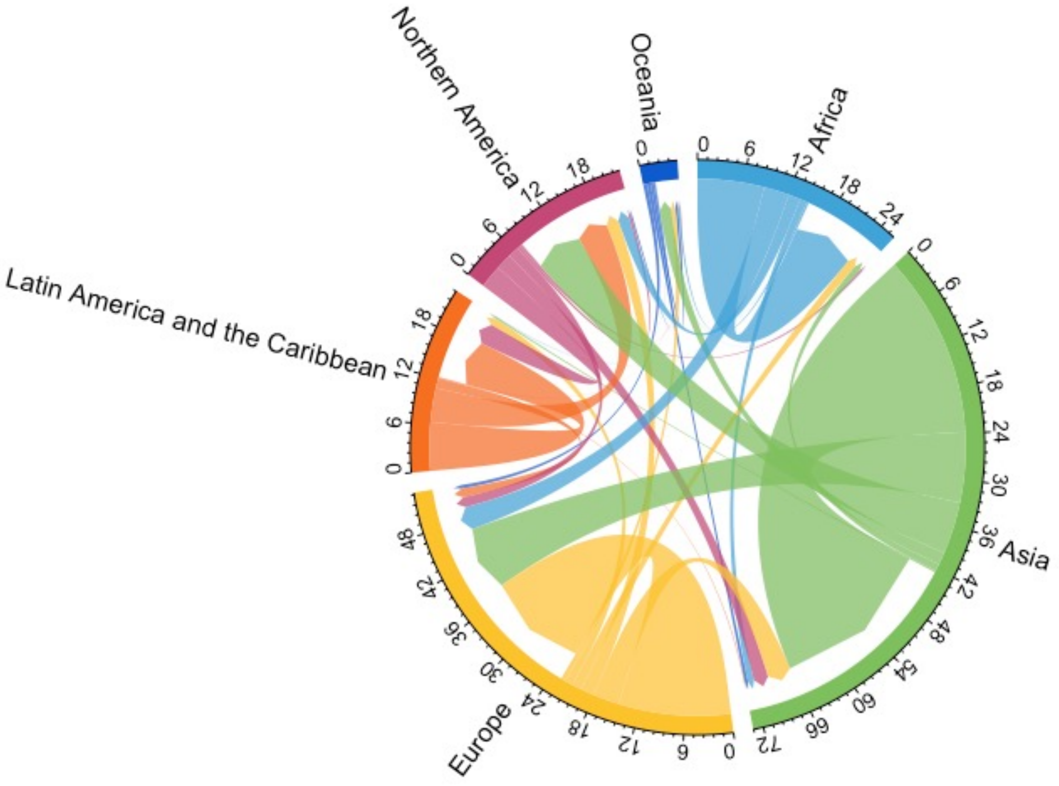
```
names(f)
```

```
[1] "year0"         "orig"          "dest"          "sd_drop_neg"
[5] "sd_rev_neg"    "mig_rate"      "da_min_open"   "da_min_closed"
[9] "da_pb_closed"
```

{migest} (III)

```
# El diccionario nos ayuda regionalizar la info

d <- f %>%
  mutate(
    orig = countrycode(sourcevar = orig, custom_dict = dict_ims,
                       origin = "iso3c", destination = "region")
    dest = countrycode(sourcevar = dest, custom_dict = dict_ims,
                       origin = "iso3c", destination = "region")
  ) %>%
  group_by(year0, orig, dest) %>%
  summarise_all(sum) %>%
  ungroup()



# 2015-2020 pseudo-Bayesian estimates for plotting
pb <- d %>%
  filter(year0 == 2015) %>% # elije un solo año
  mutate(flow = da_pb_closed/1e6) %>% # se tiene que hace "flujo
  select(orig, dest, flow) # solo necesitamos origen, destino y
pb
```

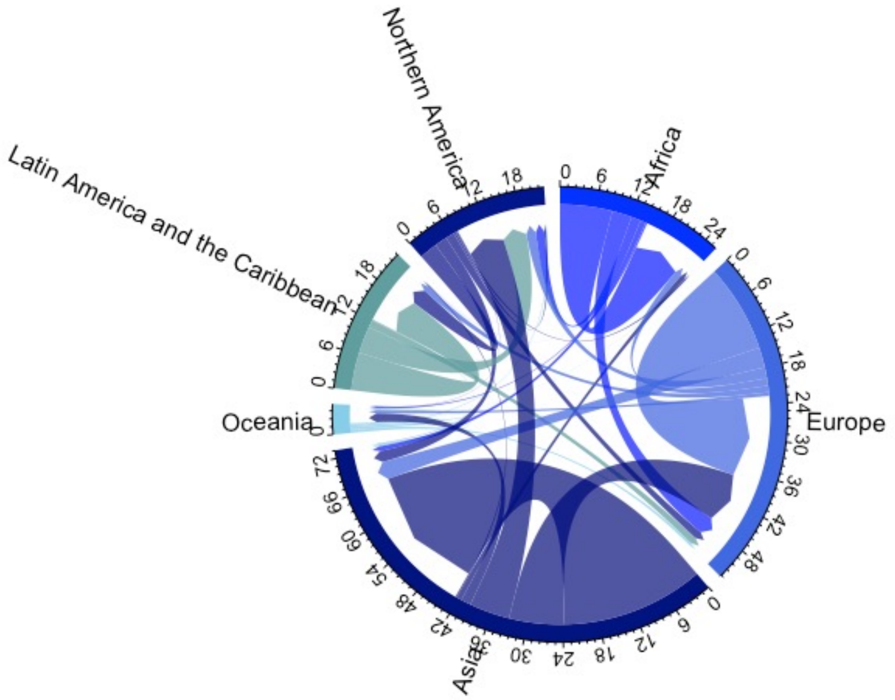
```
mig_chord(x = pb)
```



{migest} (III)

Se puede personalizar

```
# pass arguments to circlize::chordDiagramFromDataFrame

mig_chord(x = pb,
          # order of regions
          order = unique(pb$orig)[c(1, 3, 2, 6, 4, 5)],
          # spacing for labels
          preAllocateTracks = list(track.height = 0.3),
          # colours
          grid.col = c("blue", "royalblue", "navyblue",
"skyblue", "cadetblue", "darkblue")
)
```



{migest} (IV)

migest 2.0.4 | Reference | Changelog

Co

Reference

Migration data summaries

`sum_bilat()`	Summary of bilateral flows, counter-flow and net migration flow
`sum_region()` `sum_country()`	Summary of regional in-, out-, turnover and net-migration totals from an origin-destination migration flow matrix or data frame.
`sum_lump()`	Sum and lump together small flows into a "other" category
`sum_expand()`	Sum bilateral data to include aggregate bilateral totals for origin and destination meta areas
`multi_comp()`	Multiplicative component description of origin-destination migration flow tables
`multi_comp2()`	Multiplicative component descriptions of origin-destination flow tables based on total reference coding system.

Visualisation

`mig_chord()`	Chord diagram for directional origin-destination data
`str_wrap_n()`	Wrap character string to fit a target number of lines

Flows from stocks estimation methods

`ffs_demo()`	Estimation of bilateral migrant flows from bilateral migrant stocks using demographic accounting approaches
`ffs_diff()`	Estimation of bilateral migrant flows from bilateral migrant stocks using stock differencing approaches
`ffs_rates()`	Estimation of bilateral migrant flows from bilateral migrant stocks using rates approaches

{migest} (III)

¿Qué necesitamos?

```
library(migest)
library(tidyverse)
library(countrycode)

# download Abel and Cohen (2019) estimates
f <- read_csv("https://ndownloader.figshare.com/files/38016762", show_col_types = FALSE)
f
```

```
# A tibble: 307,833 × 9
   year0 orig  dest  sd_drop_neg sd_rev_neg mig_rate da_min_open da_min_closed
   <dbl> <chr> <chr>       <dbl>      <dbl>    <dbl>       <dbl>         <dbl>
 1  1990 BDI   BDI             0          0        0           0             0
```

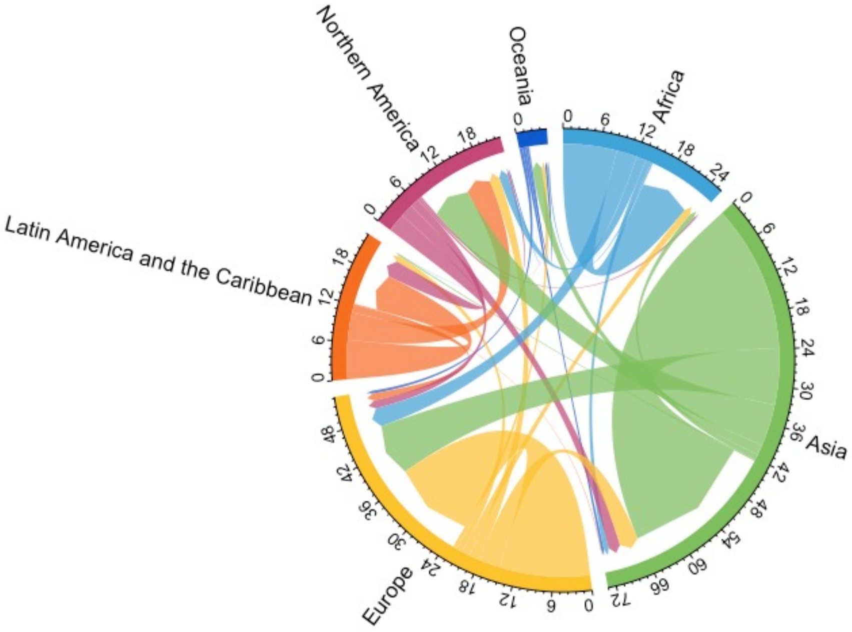
```
names(f)
```

```
[1] "year0"         "orig"          "dest"          "sd_drop_neg"
[5] "sd_rev_neg"    "mig_rate"      "da_min_open"   "da_min_closed"
[9] "da_pb_closed"
```

{migest} (III)

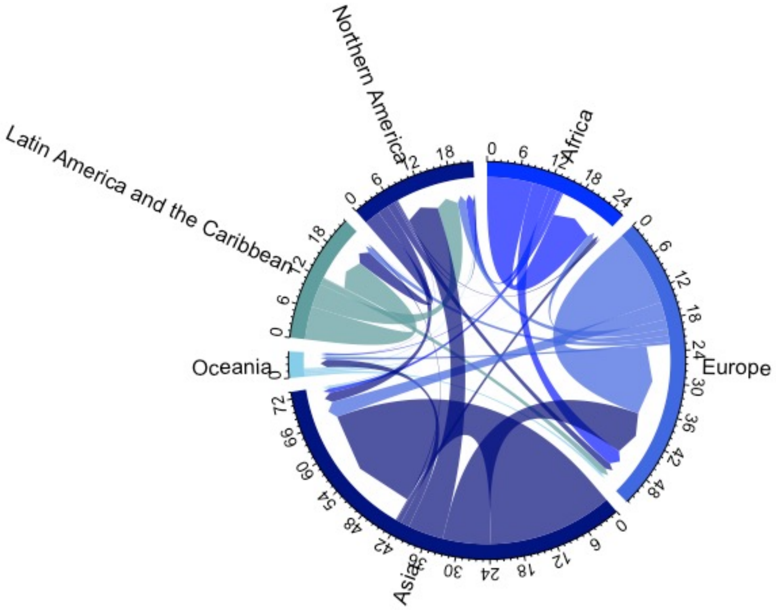
```r
# El diccionario nos ayuda regionalizar la info

d <- f %>%
  mutate(
    orig = countrycode(sourcevar = orig, custom_dict = dict_ims,
                       origin = "iso3c", destination = "region"),
    dest = countrycode(sourcevar = dest, custom_dict = dict_ims,
                       origin = "iso3c", destination = "region")
  ) %>%
  group_by(year0, orig, dest) %>%
  summarise_all(sum) %>%
  ungroup()
d
```

```
# A tibble: 216 × 9

   year0 orig   dest   sd_drop_neg sd_rev_neg mig_rate da_min_open da_min_closed

   <dbl> <chr>  <chr>        <dbl>      <dbl>    <dbl>       <dbl>         <dbl>

 1  1990 Africa Africa     4297155    7845806  5.47e6    6872677.      7728373.

 2  1990 Africa Asia        240464     258816  7.24e5     283708.       554047.

 3  1990 Africa Europe      555826     664496  1.91e6     830461.      2190967.

 4  1990 Africa Latin…        1505       2709  7.81e3       9043.        56747.

 5  1990 Africa North…      289058     301706  2.23e5     321650.       783334.

 6  1990 Africa Ocean…       21550      23570  6.59e4      30186.       165598.

 7  1990 Asia   Africa       94088     158903  2.00e5     102036.        93577.

 8  1990 Asia   Asia       3616112    8617460  1.44e7    6969956.     10337980.

 9  1990 Asia   Europe     1496141    2322839  5.48e6    2851352.      4214903.

10  1990 Asia   Latin…       14316      14343  1.07e5      20177.       136270.

# ℹ 206 more rows

# ℹ 1 more variable: da_pb_closed <dbl>

# A tibble: 36 × 3

   orig   dest                              flow

   <chr>  <chr>                            <dbl>

 1 Africa Africa                          8.69

 2 Africa Asia                            0.896

 3 Africa Europe                          3.31

 4 Africa Latin America and the Caribbean 0.0361

 5 Africa Northern America                1.59

 6 Africa Oceania                         0.264
```



{migest} (III)

Se puede personalizar



HMDHFDplus

build passing | CRAN 1.9.18 | open issues 10 | License GPL v3

There is an `R` package inside this repository called `HMDHFDplus`.

This package contains some code migrated over from the `DemogBerkeley` package, also hosted on github, as well as some new code. The `HMDHFDplus` package only contains functions for reading data into R. The `R` code is also more oranized here. Currently there are functions for reading in HMD, JMD, CHMD, HFD, and HFC code, and there are plans to implement HLD code as well, once that database is done reorganizing.

README.md

demography

The R package `demography` provides functions for demographic analysis including: lifetable calculations; Lee-Carter modelling; functional data analysis of mortality rates, fertility rates, net migration numbers; and stochastic population forecasting.

README.md

This package provides various indices, like Crude Migration Rate, different Gini indices or the Coefficient of Variation among others, to show the (un)equality of migration.

Installation

Most recent stable version can be installed directly from CRAN:

```
install.packages('migration.indices')
```

Package 'fmsb'

March 1, 2022

sion 0.7.3

e 2022-03-01

e Functions for Medical Statistics Book with some Demographic Data

README.md

Age Pyramids in R

apyramid

lifecycle experimental | CRAN 0.1.2 | downloads 21K | build error | build passing | codecov 98%

The goal of {apyramid} is to provide a quick method for visualizing census data stratified by age and one or two categorical variables (e.g. gender and health status). This is a product of the R4EPIs project; learn more at https://r4epis.netlify.com.

Códigos propios
Códigos de Germán Rodríguez
(https://data.princeton.edu/eco572)

WP 13.2
16 October 2013

UNITED NATIONS STATISTICAL COMMISSION and ECONOMIC COMMISSION FOR EUROPE

STATISTICAL OFFICE OF THE EUROPEAN UNION (EUROSTAT)

Joint Eurostat/UNECE Work Session on Demographic Projections
organised in cooperation with Istat
(29-31 October 2013, Rome, Italy)

Item 13 – Bayesian approaches

Bayesian Probabilistic Population Projections: Do It Yourself

Hana Sevcikova, University of Washington
Adrian E. Raftery, University of Washington and University College Dublin
Patrick Gerland, United Nations

Software

bayesTFR bayesLife bayesLifeHIV bayesPop wppExplorer & data MortCast

All packages described below are available on CRAN. Development versions are available from GitHub.

Softwares | BayesPop (washington.edu)

rsocsim - Socsim R package

Socsim ('Social Simulator') is an open source stochastic microsimulation platform used to produce synthetic populations with plausible kinship structures using demographic rates as input. We stand on the shoulders of giants: Socsim was originally developed for Unix at UC Berkeley [1-2], where it has been mantained for decades. The current release of `rsocsim` aims to be OS-agnostic and, for the most part, back-compatible with the original Socsim distribution (https://lab.demog.berkeley.edu/socsim/).

`rsocsim` is still under development. You might find some bugs or unexpected behavior, the API might change without warnings in the next weeks. If you encounter an error or bug, we are happy to hear from you in the issues.

DemoKin

`DemoKin` uses matrix demographic methods to compute expected (average) kin counts from demographic rates under a range of scenarios and assumptions. The package is an R-language implementation of Caswell (2019, 2020, 2022), and Caswell and Song (2021). It draws on previous theoretical development by Goodman, Keyfitz and Pullum (1974).

{paquetes demogRáficos}

Paquetes y códigos para consultar información

Paquetes para análisis demográfico

Paquetes para presentación de información

Paquetes para presentar resultados

- **shiny**: aplicaciones web interactivas con R.
- **R Markdown**: Informes reproducibles y automatizados.
- **Quarto**: Informes reproducibles y automatizados.

{mxmaps} (I)

Instalación

Este paquete es un paquete muy sencillo para hacer mapas, desarrollado por Diego Valle Jones.

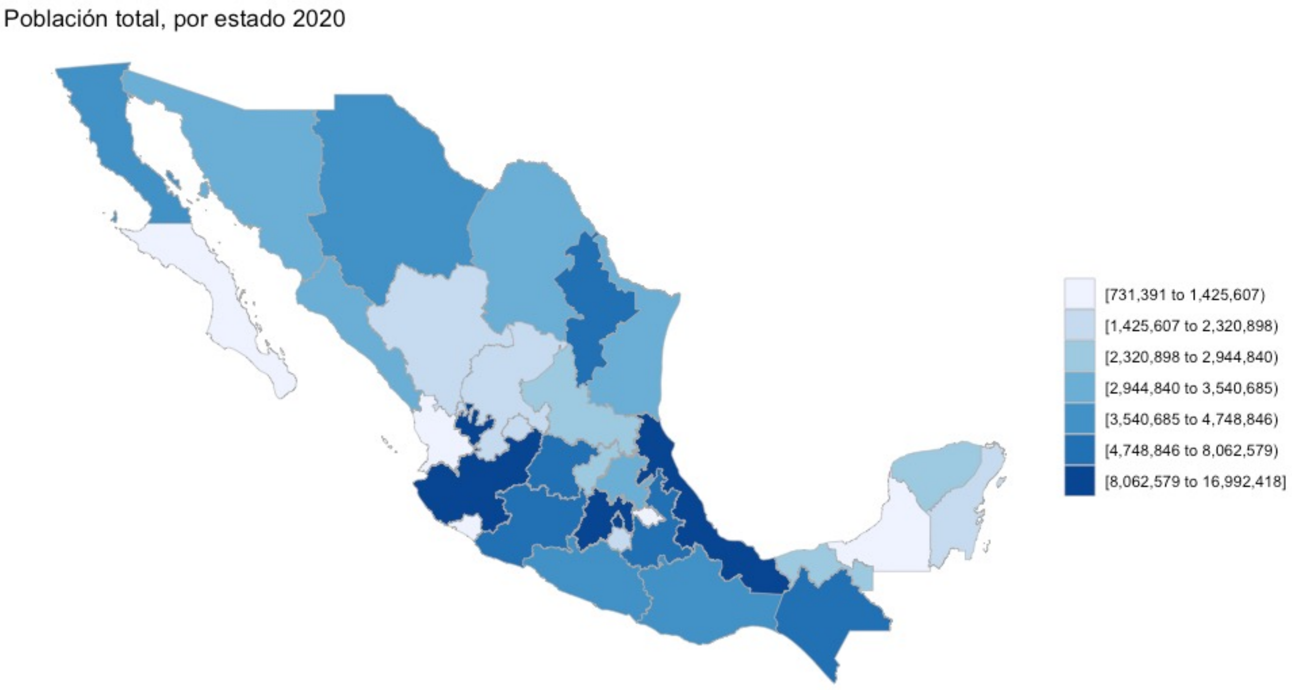
Permite hacer mapas a nivel municipal y estatal de México.

Se instala también desde su versión en desarrollo

```
remotes::install_github("diegovalle/mxmaps")
library(mxmaps)
```

Este paquete ya viene con algunos datos de población de los censos mexicanos.

{mxmaps} (II)

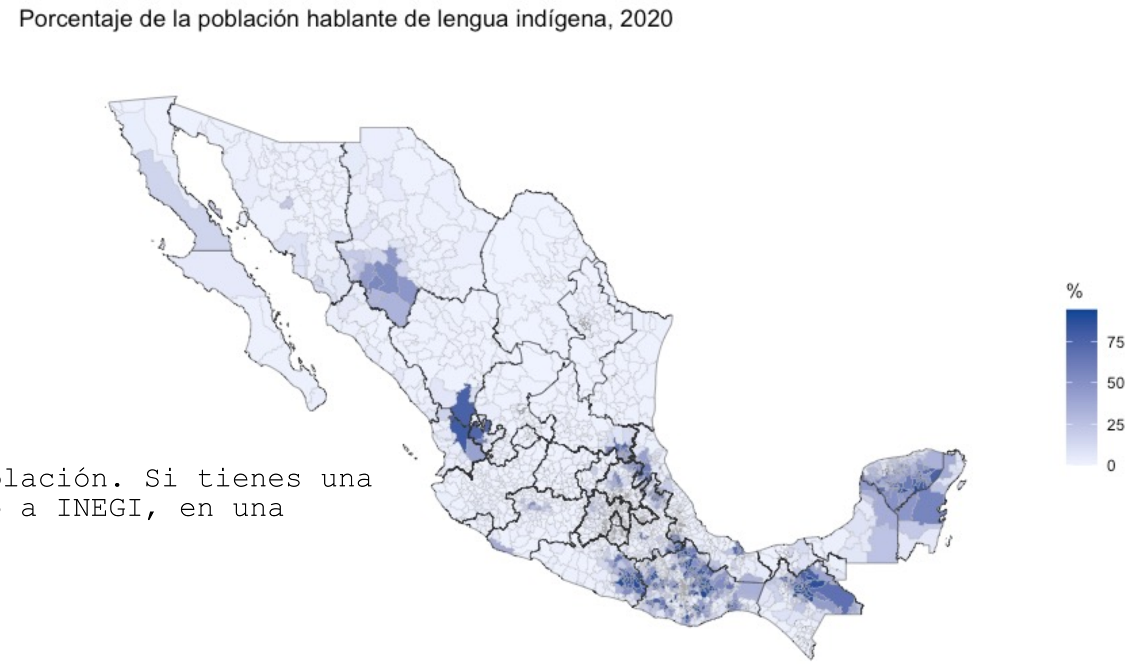


Nivel estatal

El primer ejemplo utiliza con los datos a nivel es
población. Si tienes una base con la codificación
los estados de acuerdo a INEGI, en una variable qu
región, no habrá problema

```
data("df_mexstate_2020")

df_mxstate_2020 %>% head()
  region          state_name  state_name_official state_abbr
1     01      Aguascalientes       Aguascalientes        AGS
2     02     Baja California      Baja California         BC
3     03 Baja California Sur  Baja California Sur        BCS
4     04            Campeche             Campeche       CAMP
5     05            Coahuila Coahuila de Zaragoza       COAH
6     06              Colima               Colima        COL


df_mxstate_2020$value <- df_mxstate_2020$pop

mxstate_choropleth(df_mxstate_2020,
                   title = "Población total, por estado 2020")
```

{mxmaps} (III)



Nivel municipal

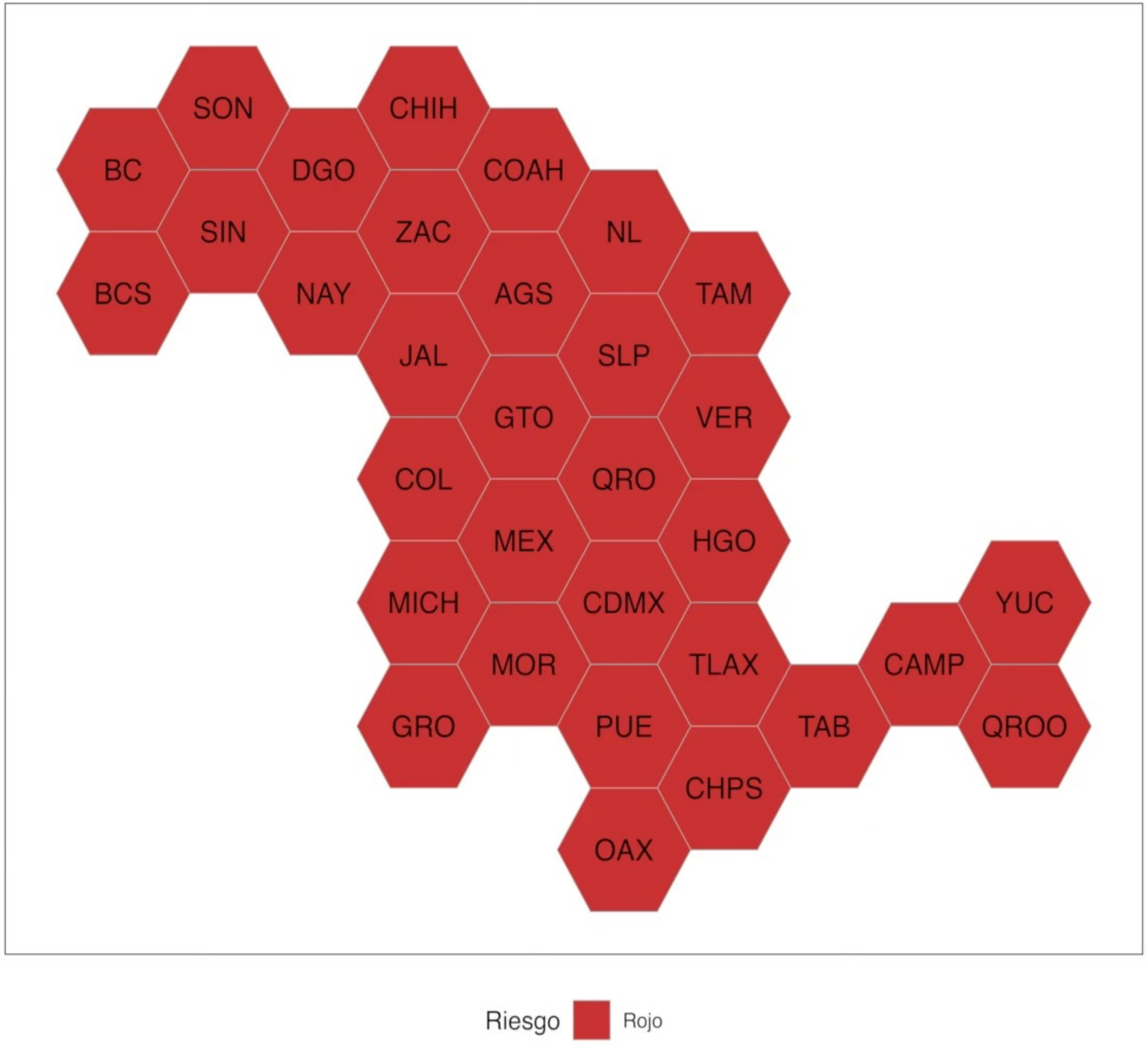
Este segundo ejemplo utiliza con los datos a nivel estatal de población. Si tienes una base con la codificación de los datos en los municipio de acuerdo a INEGI, en una variable que se llama región, no habrá problema

```
data("df_mxmunicipio_2020")
df_mxmunicipio_2020 %>% head()
```

```
# A tibble: 6 × 17
  state_code munic…¹ region state…² state…³ state…⁴ state…⁵ munic…⁶  year    pop
  <chr>      <chr>   <chr>  <chr>   <chr>   <chr>   <chr>   <chr>   <dbl>  <int>
1 01         001     01001  Aguasc… Aguasc… AGS     Ags.    Aguasc…  2020 948990
2 01         002     01002  Aguasc… Aguasc… AGS     Ags.    Asient…  2020  51536
3 01         003     01003  Aguasc… Aguasc… AGS     Ags.    Calvil…  2020  58250
4 01         004     01004  Aguasc… Aguasc… AGS     Ags.    Cosío    2020  17000
5 01         005     01005  Aguasc… Aguasc… AGS     Ags.    Jesús …  2020 129929
6 01         006     01006  Aguasc… Aguasc… AGS     Ags.    Pabell…  2020  47646
```

```
df_mxmunicipio_2020$value <-  df_mxmunicipio_2020$indigenous_language /
df_mxmunicipio_2020$pop * 100

mxmunicipio_choropleth(df_mxmunicipio_2020,
                       num_colors = 1,
                       title = "Porcentaje de la población hablante de lengua indígena, 2020",
                       legend = "%")
```

Semáforo 6-2020



Elaboración propia con la información del paquete {semaforos} (Gruson, 2021/2022)
usando el paquete {mxmaps} (Valle Jones, 2022)

{códigos de R}

Códigos propios
Códigos de Germán Rodríguez
(https://data.princeton.edu/eco572)

This **Applied Demography Toolbox** is a list of applied demography computer programs, packages, scripts, spreadsheets, databases, and reference texts. If you use, share, or reproduce information or ideas from the linked files, please cite the respective source. If you have questions, comments, or additions for this site, please email me (Eddie Hunsinger) at edyhsgr@protonmail.com. Acknowledgments.

Applied
Demography
Toolbox

Filter by title or attribute text. Asterisks identify items maintained here.

Topic attributes (all) | Technology attributes (all)

[Applied Demography Toolbox (applieddemogtoolbox.github.io)](https://applieddemogtoolbox.github.io)

Una versión anterior

Gracias

ana.escoto@politicas.unam.mx