Package: first90 (via r-universe)

August 10, 2024

Version 1.6.11

Title The first90 model

Description Implements the Shiny90 model for estimating progress towards the UNAIDS ``first 90'' target for HIV awareness of status in sub-Saharan Africa.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

ByteCompile true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Depends R (>= 2.10)

Imports mvtnorm, Matrix, fastmatch (>= 1.1), Rcpp, vroom

LinkingTo BH, Rcpp

Suggests testthat

URL https://github.com/mrc-ide/first90release, https://shiny90.unaids.org

BugReports https://github.com/mrc-ide/first90release/issues

Repository https://mrc-ide.r-universe.dev

RemoteUrl https://github.com/mrc-ide/first90release

RemoteRef master

RemoteSha d80ab65da2187e1ce2a3c11f5e2ee25de998f85a

Contents

add_ss_indices *Add incidences corresponding to state space*

Description

Add incidences corresponding to state space

Usage

```
add_ss_indices(dat, ss)
```

Arguments

dat	a data.frame with particular columns coded in particular ways
ss	state space definition (fp$ss)

Examples

```
## Not run:
data(survey_hts)
dat <- subset(survey_hts, survey_hts$country == "Malawi" & outcome == "evertest")
df <- add_ss_indices(dat, fp$ss)

## End(Not run)
```

artcov15to49 *ART coverage among age 15-49*

Description

ART coverage among age 15-49

Usage

```
artcov15to49(mod, sex = "both")
```

calc_asfr *Calculate ASFR from TFR and fertility distribution*

Description

Calculate ASFR from TFR and fertility distribution

Usage

```
calc_asfr(tfr, asfd)
```

Arguments

tfr	vector of annual TFR values
asfd	array of proportion of births by 5 year age group 15-49

Value

array of age-specific fertility rate by single-year of age 15-49.

`combine_inputs` *Combine PJNZ inputs*

Description

Combine PJNZ inputs

Usage

```
combine_inputs(lst)
```

Arguments

`lst`	a list of inputs, each returned from extract_pjnz()

`create_beers` *Beers coefficients to distribute from 5-year to single-year of age*

Description

Beers coefficients to distribute from 5-year to single-year of age

Usage

```
create_beers(n5yr)
```

`create_hts_param` *Create HIV testing rates from parameter inputs*

Description

Create HIV testing rates from parameter inputs

Usage

```
create_hts_param(theta, fp)
```

Details

This function creates two arrays `hts_rate` and `diagn_rate`. The `hts_rate` array summarizes HIV testing rates for the HIV negative population stratified by:

- HIV aggregated age groups (hAG: 1-9)
- sex (1 = male; 2 = female)
- testing history (1 = never tested; 2 = previously tested negative)
- year

Array `diagn_rate` summarizes HIV testing rates for the HIV positive population stratified by:

- CD4 stage (hDS: 1-7)
- HIV age group (hAG: 1-9)
- sex (1 = male; 2 = female)
- testing and treatment history (1 = never tested; 2 = previously tested negative; 3 = diagnosed, not treated; 4 = on ART)
- year

`diagnosed` *Proportion diagnosed among HIV+ by age, sex, or HIV status*

Description

This function calculates proportion diagnosed among the HIV positive population stratified by age group, sex, HIV status, and year.

Usage

```
diagnosed(mod, fp, df, VERSION = "C")
```

Arguments

`mod`	simulation model output
`fp`	simulation model parameter inputs
`df`	a data.frame with indices for prediction, see Details.

Details

Age groups are specified in terms of aggregate HIV age groups: 15-16, 17-19, 20-24, ..., 45-49, 50+. Another function could be added to handle other age groups if needed, with additional computational complexity.

The argument df must contain the following columns:

- haidx: HIV age group (1 = 15-16, 2 = 17-19, 3 = 20-24, ..., 8 = 45-49, 9 = 50+)
- sidx: sex (1 = male, 2 = female, 0 = both)
- yidx: year index
- hagspan: number of HIV age groups to span

Value

a vector

Examples

```
## Not run:
data(survey_hts)
dat <- subset(survey_hts, country == "Malawi" & outcome == "aware")
df <- add_ss_indices(dat, fp$ss)

df$pred <- diagnosed(mod, fp, df)

## End(Not run)
```

evertest *Proportion ever tested by age, sex, or HIV status*

Description

This function calculates proportion of ever tested among a population stratified by age group, sex, HIV status, and year.

Usage

```
evertest(mod, fp, df, VERSION = "C")
```

Arguments

mod	simulation model output
fp	simulation model parameter inputs
df	a data.frame with indices for prediction, see Details.

Details

Age groups are specified in terms of aggregate HIV age groups: 15-16, 17-19, 20-24, ..., 45-49, 50+. Another function could be added to handle other age groups if needed, with additional computational complexity.

The argument df must contain the following columns:

- haidx: HIV age group (1 = 15-16, 2 = 17-19, 3 = 20-24, ..., 8 = 45-49, 9 = 50+)
- sidx: sex (1 = male, 2 = female, 0 = both)
- hvidx: HIV status (1 = negative, 2 = positive, 0 = all)
- yidx: year index
- hagspan: number of HIV age groups to span

Value

a vector

Examples

```
## Not run:
data(survey_hts)
dat <- subset(survey_hts, country == "Malawi" & outcome == "evertest")
df <- add_ss_indices(dat, fp$ss)

df$pred <- evertest(mod, fp, df)

## End(Not run)
```

exists_dptag *Internal helper functions*

Description

Internal helper functions

Usage

```
exists_dptag(dp, tag, tagcol = 1)
```

extract_pjnz *Extract outputs from PJNZ needed for first90 model*

Description

Extract outputs from PJNZ needed for first90 model

Usage

```
extract_pjnz(pjnz = NULL, dp_file = NULL, pjn_file = NULL)
```

Arguments

pjnz	filepath to PJNZ file
dp_file	filepath to a .DP file
pjn_file	filepath to a .PJN file

Value

a list

get_dp_frr *Extract AIM module parameters*

Description

Extract AIM module parameters

Usage

```
get_dp_frr(dp, proj_years)
```

get_dp_srb *Get sex ratio at birth*

Description

Get sex ratio at birth

Usage

```
get_dp_srb(dp, proj_years)
```

get_dp_Sx *Get mortality probability (Sx) from Spectrum DP file*

Description

Get mortality probability (Sx) from Spectrum DP file

Usage

```
get_dp_Sx(dp, proj_years)
```

Details

This function extracts Sx for ages 0:79 and 80+. Spectrum calculates a separate Sx for age 80. The population projection model in EPP-ASM needs to be updated to handle this.

`get_dp_tfr` *Get age-specific fertility rate by single-year*

Description

Get age-specific fertility rate by single-year

Usage

```
get_dp_tfr(dp, proj_years)
```

`get_dp_totpop` *Extract arrays from Spectrum DP file*

Description

Extract arrays from Spectrum DP file

Usage

```
get_dp_totpop(dp, proj_years)
```

`get_pjn_country` *Get country name from parsed PJN*

Description

Get country name from parsed PJN

Usage

```
get_pjn_country(pjn)
```

Arguments

`pjn`	parsed PJN file

Details

`pjn` should be via `first90_read_csv_character(pjn_file)`

get_pjn_region *Get subnational region from parsed PJN*

Description

Get subnational region from parsed PJN

Usage

```
get_pjn_region(pjn)
```

Arguments

pjn	parsed PJN file

Details

`pjn` should be via `first90_read_csv_character(pjn_file)`

number_diagnoses *Number of new diagnoses by age and sex*

Description

Number of new diagnoses by age and sex

Usage

```
number_diagnoses(mod, fp, df, VERSION = "R")
```

Value

A data frame reporting new diagnoses in two ways:

1. Approximated from the testing rates and mid-year populations as in the function `number_tests()`.
2. Based on the number of new diagnoses recorded in each model time step.

The column `late_diagnoses` reports the number who are initiated to ART directly from the undiagnosed population.

The two approaches are largely for debugging purposes to understand the implications of the model choices.

number_retests *Calculate number adn proportion of retests conducted or number tested by year funtion based on number test in the outputs.R script*

Description

Calculate number adn proportion of retests conducted or number tested by year funtion based on number test in the outputs.R script

Usage

```
number_retests(mod, fp, df)
```

Arguments

`mod`	model output of class 'eppasm'
`fp`	parameter inputs (class 'specfp')
`df`	a data.frame with indices for prediction. See `evertest()` for more information.

Details

Number of tests (or number tested in past 12 months) are approximated by mid-year counts and annual testing rates within each stratum.

Value

a data.frame consisting of the number of tests, and population size corresponding to rows of df

number_tests *Calculate number of tests conducted or number tested by year*

Description

Calculate number of tests conducted or number tested by year

Usage

```
number_tests(mod, fp, df, tested12m = FALSE, VERSION = "C")
```

Arguments

`mod`	model output of class 'eppasm'
`fp`	parameter inputs (class 'specfp')
`df`	a data.frame with indices for prediction. See `evertest()` for more information.

Details

Number of tests (or number tested in past 12 months) are approximated by mid-year counts and annual testing rates within each stratum.

Value

a data.frame consisting of the number of tests, number tested in the past 12 months, and population size corresponding to rows of df

---

`prepare_inputs` *Create model inputs from Spectrum PJNZ*

Description

Create model inputs from Spectrum PJNZ

Usage

```
prepare_inputs(pjnzlist)
```

Arguments

`pjnzlist`	a vector of PJNZ file names to aggregate

Details

The aggregation makes a number of assumptions:

- Progression parameters are the same in all files, and values frome the first file are used.
- Special populations ART eligibility is the same in all files.

Examples

```
pjnzlist <- list.files("~/Documents/Data/Spectrum files/2018 final/SSA/",
  "CotedIvoire.*PJNZ$", full.names=TRUE, ignore.case=TRUE)
pjnzlist <- "~/Documents/Data/Spectrum files/2018 final/SSA/Malawi_2018_version_8.PJNZ"
```

`prepare_inputs_from_extracts` *Create model inputs from aggregated Spectrum PJNZ*

Description

Create model inputs from aggregated Spectrum PJNZ

Usage

```
prepare_inputs_from_extracts(pjnz_in)
```

Arguments

`pjnz_in`	a list of outputs from extract_pjnz

`select_hts` *Process survey data on hiv testing behaviors*

Description

Process survey data on hiv testing behaviors

Usage

```
select_hts(survey_hts, cnt, age_group)
```

`select_prgmdata` *Process programmatic data on number of tests*

Description

Process programmatic data on number of tests

Usage

```
select_prgmdata(prgm_dat, cnt, age_group)
```

`simmod` *Simulate EPP-ASM model*

Description

Simulate EPP-ASM model

Usage

```
simmod(fp, VERSION = "C")
```

`spectrum_output_table` *Table to write CSV outputs for Spectrum*

Description

Table to write CSV outputs for Spectrum

Usage

```
spectrum_output_table(mod, fp)
```

Details

Presently this returns point estimates for age 15+ population by sex:

- Number PLHIV,
- Ever tested among PLHIV
- Aware of HIV+ status
- On ART

PLHIV is mid-year estimate. All other outcomes are end of year estimate.

Value

a data.frame to write to CSV file for ingestion into Spectrum

Examples

```
## Not run:
write.csv(spectrum_output_table(mod, fp),
          "~/Downloads/Malawi-shiny90-example-output.csv",
           row.names = FALSE)

## End(Not run)
```

`svy_hts_interface` *Function to present the survey data on shinny interface*

Description

Function to present the survey data on shinny interface

Usage

```
svy_hts_interface(survey_hts, cnt, age_group = c("15-24", "25-34", "35-49"))
```

Index