Package: StanMoMo (via r-universe)

October 29, 2026

Title Bayesian Mortality Modelling with 'Stan'

Version 1.2.0

Description Implementation of popular mortality models using the 'rstan' package, which provides the R interface to the 'Stan' C++ library for Bayesian estimation. The package supports well-known models proposed in the actuarial and demographic literature including the Lee-Carter (1992) <doi:10.1080/01621459.1992.10475265> and the Cairns-Blake-Dowd (2006) <doi:10.1111/j.1539-6975.2006.00195.x> models. By a simple call, the user inputs deaths and exposures and the package outputs the MCMC simulations for each parameter, the log likelihoods and predictions. Moreover, the package includes tools for model selection and Bayesian model averaging by leave future-out validation.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Biarch true

Depends R (>= 3.5.0)

Imports loo, methods, RcppParallel, Rcpp (>= 0.12.0), rstan (>= 2.26.0), rstantools (>= 2.0.0), tidyverse, dplyr, tibble, httr, bridgesampling, stats, utils, tidyselect, latex2exp, tidyr, ggplot2

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), rstan (>= 2.26.0), StanHeaders (>= 2.26.0), RcppParallel

SystemRequirements GNU make

Suggests knitr, rmarkdown, shinystan, Cairo

VignetteBuilder knitr

URL https://github.com/kabarigou/StanMoMo

BugReports https://github.com/kabarigou/StanMoMo/issues

NeedsCompilation yes

Author Karim Barigou [aut, cre] (<https://orcid.org/0000-0003-3389-9596>), Pierre-Olivier Goffard [aut] (<https://orcid.org/0000-0002-2667-6207>)

Maintainer Karim Barigou <karim290492@gmail.com>

Config/pak/sysreqs make

Repository https://cran.r-universe.dev

Date/Publication 2023-09-23 23:30:39 UTC

RemoteUrl https://github.com/cran/StanMoMo

RemoteRef HEAD

RemoteSha 01163f7dec3e1e99b25fc90d777299403879261b

Contents

StanMoMo-package 2
apc_stan 3
boxplot_post_dist 5
cbd_stan 6
compute_weights_BMA 8
extract_map 8
fit_mo_mo 9
FRMaleData 10
lc_stan 11
m6_stan 12
mortality_weights 15
rh_stan 16
sim_death_apc 18
sim_death_cbd 19
sim_death_lc 19
sim_death_m6 20
sim_death_mix_cbd_rh 20
sim_death_rh 21
sim_mortality_data 21

Index **23**

`StanMoMo-package` *The 'StanMoMo' package.*

Description

The **StanMoMo** package performs Bayesian Mortality Modeling with **Stan** for a variety of popular mortality models. The current package supports the Lee-Carter model, the Renshaw-Haberman model, the Age-Period-Cohort model, the Cairns-Blake-Dowd model and the M6 model. By a simple call, the user inputs deaths and exposures and the package outputs the MCMC simulations for each parameter, the log likelihoods and predictions. Moreover, the package includes tools for model selection and Bayesian model averaging by leave-future-out validation.

References

Stan Development Team (2020). RStan: the R interface to Stan. R package version 2.19.3. https://mc-stan.org

---

`apc_stan` *Bayesian Age-Period-Cohort model with 'Stan'*

Description

Fit and Forecast Bayesian APC model. The model can be fitted with a Poisson or Negative-Binomial distribution. The function outputs posteriors distributions for each parameter, predicted death rates and log-likelihoods.

Usage

```
apc_stan(
  death,
  exposure,
  forecast,
  validation = 0,
  family = c("poisson", "nb"),
  ...
)
```

Arguments

`death`	Matrix of deaths.
`exposure`	Matrix of exposures.
`forecast`	Number of years to forecast.
`validation`	Number of years for validation.
`family`	specifies the random component of the mortality model. `"Poisson"` assumes a Poisson model with log link and `"nb"` assumes a negative-binomial model with log link and overdispersion parameter ϕ.
`...`	Arguments passed to `rstan::sampling` (e.g. iter, chains).

Details

The created model is either a log-Poisson or a log-Negative-Binomial version of the APC model:

$$D_{x,t} \sim \mathcal{P}(\mu_{x,t} e_{x,t})$$

or

$$D_{x,t} \sim NB\left(\mu_{x,t} e_{x,t}, \phi\right)$$

with

$$\log \mu_{xt} = \alpha_x + \kappa_t + \gamma_{t-x}.$$

To ensure the identifiability of th model, we impose

$$\kappa_1 = 0, \gamma_1 = 0, \gamma_C = 0,$$

where C represents the most recent cohort in the data.

For the priors, we assume that

$$\alpha_x \sim N(0, 100), \frac{1}{\phi} \sim Half - N(0, 1).$$

For the period term, similar to the LC model, we consider a random walk with drift:

$$\kappa_t = c + \kappa_{t-1} + \epsilon_t, \epsilon_t \sim N(0, \sigma^2)$$

with the following hyperparameters assumptions: $c \sim N(0, 10), \sigma \sim Exp(0.1)$.

For the cohort term, we consider a second order autoregressive process (AR(2)):

$$\gamma_c = \psi_1 \gamma_{c-1} + \psi_2 \gamma_{c-2} + \epsilon_t^{\gamma}, \quad \epsilon_t^{\gamma} \sim N(0, \sigma_\gamma).$$

To close the model specification, we impose some vague priors assumptions on the hyperparameters:

$$\psi_1, \psi_2 \sim N(0, 10), \quad \sigma_\gamma \sim Exp(0.1).$$

Value

An object of class `stanfit` returned by `rstan::sampling`

References

Cairns, A. J. G., Blake, D., Dowd, K., Coughlan, G. D., Epstein, D., Ong, A., & Balevich, I. (2009). A quantitative comparison of stochastic mortality models using data from England and Wales and the United States. North American Actuarial Journal, 13(1), 1-35.

Examples

```
#10-year forecasts for French data for ages 50-90 and years 1970-2017 with a log-Poisson model
ages.fit<-70:90
years.fit<-1990:2010
deathFR<-FRMaleData$Dxt[formatC(ages.fit),formatC(years.fit)]
exposureFR<-FRMaleData$Ext[formatC(ages.fit),formatC(years.fit)]
iterations<-50 # Toy example, consider at least 2000 iterations
fitAPC=apc_stan(death = deathFR,exposure=exposureFR, forecast = 5, family = "poisson",
iter=iterations,chains=1)
```

---

boxplot_post_dist *Boxplot for the posterior distribution*

Description

The function generates boxplots for the posterior distribution of the main parameters

Usage

```
boxplot_post_dist(stan_fit, parm_name, ages, years)
```

Arguments

stan_fit	stan fit object
parm_name	string to indicate the name of the parameter, to choose from c('a', 'b', 'g', 'k', 'k2', 'phi')
ages	range of ages
years	range of years

Value

Posterior distribution shown as boxplots

Examples

```
years <- 1990:2017
ages <- 50:90
cohorts <- sort(unique(as.vector(sapply(years, function(year) year - ages))))
death <- FRMaleData$Dxt[formatC(ages),formatC(years)]
exposure <- FRMaleData$Ext[formatC(ages),formatC(years)]
iterations<-50 # Toy example, consider at least 2000 iterations
stan_fit <- fit_mo_mo("m6", death , exposure, ages, 0, 5, "nb", 1, 4,
log_marg = FALSE,iter=iterations)
boxplot_post_dist(stan_fit, "k", ages, years)
```

```
boxplot_post_dist(stan_fit, "g", ages, years)
```

---

cbd_stan *Bayesian Cairns-Blake-Dowd (CBD) model with Stan*

Description

Fit and Forecast Bayesian CBD model. The model can be fitted with a Poisson or Negative-Binomial distribution. The function outputs posteriors distributions for each parameter, predicted death rates and log-likelihoods.

Usage

```
cbd_stan(
  death,
  exposure,
  age,
  forecast,
  validation = 0,
  family = c("poisson", "nb"),
  ...
)
```

Arguments

death	Matrix of deaths.
exposure	Matrix of exposures.
age	Vector of ages.
forecast	Number of years to forecast.
validation	Number of years for validation.
family	specifies the random component of the mortality model. "Poisson" assumes a Poisson model with log link and "nb" assumes a negative-binomial model with log link and overdispersion parameter ϕ.
...	Arguments passed to rstan::sampling (e.g. iter, chains).

Details

The created model is either a log-Poisson or a log-Negative-Binomial version of the CBD model:

$$D_{x,t} \sim \mathcal{P}(\mu_{x,t} e_{x,t})$$

or

$$D_{x,t} \sim NB\left(\mu_{x,t} e_{x,t}, \phi\right)$$

with

$$\log \mu_{xt} = \kappa_t^{(1)} + (x - \bar{x})\kappa_t^{(2)},$$

where $\bar{x}$ is the average age in the data.

For the period terms, we consider a multivariate random walk with drift:

$$\boldsymbol{\kappa}_t = \boldsymbol{c} + \boldsymbol{\kappa}_{t-1} + \boldsymbol{\epsilon}_t^{\kappa}, \quad \boldsymbol{\kappa}_t = \begin{pmatrix} \kappa_t^{(1)} \\ \kappa_t^{(2)} \end{pmatrix}, \quad \boldsymbol{\epsilon}_t^{\kappa} \sim N(\mathbf{0}, \Sigma),$$

with normal priors: $\boldsymbol{c} \sim N(0, 10)$.

The variance-covariance matrix of the error term is defined by

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho_\Sigma \sigma_1 \sigma_2 \\ \rho_\Sigma \sigma_1 \sigma_Y & \sigma_2^2 \end{pmatrix}$$

where the variance coefficients have independent exponential priors: $\sigma_1, \sigma_2 \sim Exp(0.1)$ and the correlation parameter has a uniform prior: $\rho_\Sigma \sim U[-1, 1]$. As for the other models, the overdispersion parameter has a prior distribution given by

$$\frac{1}{\phi} \sim Half - N(0, 1).$$

Value

An object of class `stanfit` returned by `rstan::sampling`

References

Cairns, A. J. G., Blake, D., & Dowd, K. (2006). A Two-Factor Model for Stochastic Mortality with Parameter Uncertainty: Theory and Calibration. Journal of Risk and Insurance, 73(4), 687-718.

Examples

```
#10-year forecasts for French data for ages 50-90 and years 1970-2017 with a log-NB model
ages.fit<-50:90
years.fit<-1970:2017
deathFR<-FRMaleData$Dxt[formatC(ages.fit),formatC(years.fit)]
exposureFR<-FRMaleData$Ext[formatC(ages.fit),formatC(years.fit)]
iterations<-50 # Toy example, consider at least 2000 iterations
fitCBD=cbd_stan(death = deathFR,exposure=exposureFR, age=ages.fit, forecast = 10,
family = "poisson",iter=iterations,chains=1)
```

`compute_weights_BMA` *Compute the model evidence/marginal likelihood via bridge sampling (or via the harmonic mean estimator if bridge sampling fails)*

Description

Compute the model evidence/marginal likelihood via bridge sampling (or via the harmonic mean estimator if bridge sampling fails)

Usage

```
compute_weights_BMA(stan_fits, mortality_models)
```

Arguments

`stan_fits`	list of Stan model fits where the marginal likelihood was computed via bridge sampling
`mortality_models`	vector of mortality models names

Value

data frame with model evidence for BMA

`extract_map` *Function to get the a posterior means of the parameters based on a stanfit object*

Description

Function to get the a posterior means of the parameters based on a stanfit object

Usage

```
extract_map(stan_fit)
```

Arguments

`stan_fit`	a stanfit object

Value

named list with the point estimates of the parameters

`fit_mo_mo` *Wrapper function to fit and forecast mortality models*

Description

Wrapper function to fit and forecast mortality models

Usage

```
fit_mo_mo(
  mortality_model = "lc",
  death,
  exposure,
  ages = 50:90,
  validation = 0,
  forecast = 1,
  family = "nb",
  chains = 1,
  cores = 4,
  log_marg = FALSE,
  iter = 2000
)
```

Arguments

`mortality_model`	name of the mortality model
`death`	death matrix
`exposure`	exposure matrix
`ages`	vector of ages
`validation`	size of the validation set
`forecast`	number of calendar years to be forecast
`family`	underlying count distribution
`chains`	number of Markov chains
`cores`	number of cores used
`log_marg`	Do we compute the marginal likelihood or not?
`iter`	Length of the Markov chain trajectory

Value

a stanfit object

Examples

```
years <- 1990:2017
ages <- 50:90
cohorts <- sort(unique(as.vector(sapply(years, function(year) year - ages))))
death <- FRMaleData$Dxt[formatC(ages),formatC(years)]
exposure <- FRMaleData$Ext[formatC(ages),formatC(years)]
stan_fit <- fit_mo_mo("m6", death , exposure, ages, 0, 5, "nb", 1, 4,
log_marg = FALSE,iter=50)
boxplot_post_dist(stan_fit, "k", ages, years)
boxplot_post_dist(stan_fit, "g", ages, years)
```

`FRMaleData` *Deaths and Exposures Data of French Males*

Description

List containing deaths and exposures of French Male for ages 0-110 and years 1816-2017.

Usage

```
FRMaleData
```

Format

A list of 2 matrices

Dxt Matrix of deaths, with 111 rows (ages) and 202 columns (years)

Ext Matrix of exposures, with 111 rows (ages) and 202 columns (years)

Source

Human Mortality Database https://www.mortality.org/

References

Human Mortality Database (2011). University of California, Berkeley (USA), and Max Planck Institute for Demographic Research (Germany). Available at https://www.mortality.org/

`lc_stan` *Bayesian Lee-Carter with Stan*

Description

Fit and Forecast Bayesian Lee-Carter model. The model can be fitted with a Poisson or Negative-Binomial distribution. The function outputs posteriors distributions for each parameter, predicted death rates and log-likelihoods.

Usage

```
lc_stan(
  death,
  exposure,
  forecast,
  validation = 0,
  family = c("poisson", "nb"),
  ...
)
```

Arguments

`death`	Matrix of deaths.
`exposure`	Matrix of exposures.
`forecast`	Number of years to forecast.
`validation`	Number of years for validation.
`family`	specifies the random component of the mortality model. `"Poisson"` assumes a Poisson model with log link and `"nb"` assumes a negative-binomial model with log link and overdispersion parameter ϕ.
`...`	Arguments passed to `rstan::sampling` (e.g. iter, chains).

Details

The created model is either a log-Poisson or a log-Negative-Binomial version of the Lee-Carter model:

$$D_{x,t} \sim \mathcal{P}(\mu_{x,t} e_{x,t})$$

or

$$D_{x,t} \sim NB\left(\mu_{x,t} e_{x,t}, \phi\right)$$

with

$$\log \mu_{xt} = \alpha_x + \beta_x \kappa_t.$$

To ensure the identifiability of th model, we impose

$$\sum_x \beta_x = 1, \kappa_1 = 0.$$

For the priors, the model chooses relatively wide priors:

$$\alpha_x \sim N(0, 100), \beta_x \sim Dir(1, \ldots, 1), \frac{1}{\phi} \sim Half - N(0, 1).$$

For the period term, we consider a first order autoregressive process (AR(1)) with linear trend:

$$\kappa_t = c + \kappa_{t-1} + \epsilon_t, \epsilon_t \sim N(0, \sigma^2)$$

with $c \sim N(0, 10), \sigma \sim Exp(0.1)$.

Value

An object of class `stanfit` returned by `rstan::sampling`.

References

Lee, R. D., & Carter, L. R. (1992). Modeling and forecasting U.S. mortality. Journal of the American Statistical Association, 87(419), 659-671.

Examples

```
#10-year forecasts for French data for ages 50-90 and years 1970-2017 with a log-Poisson model
ages.fit<-50:90
years.fit<-1970:2017
deathFR<-FRMaleData$Dxt[formatC(ages.fit),formatC(years.fit)]
exposureFR<-FRMaleData$Ext[formatC(ages.fit),formatC(years.fit)]
iterations<-50 # Toy example, consider at least 2000 iterations
fitLC=lc_stan(death = deathFR,exposure=exposureFR, forecast = 10,
family = "poisson",iter=iterations,chains=1)
```

`m6_stan` *Bayesian M6 model with Stan*

Description

Fit and Forecast Bayesian M6 model (CBD with cohort effect) introduced in Cairns et al (2009). The model can be fitted with a Poisson or Negative-Binomial distribution. The function outputs posteriors distributions for each parameter, predicted death rates and log-likelihoods.

Usage

```
m6_stan(
  death,
  exposure,
  forecast,
  age,
  validation = 0,
  family = c("poisson", "nb"),
  ...
)
```

Arguments

death	Matrix of deaths.
exposure	Matrix of exposures.
forecast	Number of years to forecast.
age	Vector of ages.
validation	Number of years for validation.
family	specifies the random component of the mortality model. "Poisson" assumes a Poisson model with log link and "nb" assumes a negative-binomial model with log link and overdispersion parameter ϕ.
...	Arguments passed to rstan::sampling (e.g. iter, chains).

Details

The created model is either a log-Poisson or a log-Negative-Binomial version of the M6 model:

$$D_{x,t} \sim \mathcal{P}(\mu_{x,t} e_{x,t})$$

or

$$D_{x,t} \sim NB\left(\mu_{x,t} e_{x,t}, \phi\right)$$

with

$$\log \mu_{xt} = \kappa_t^{(1)} + (x - \bar{x})\kappa_t^{(2)} + \gamma_{t-x},$$

where $\bar{x}$ is the average age in the data.

To ensure the identifiability of th model, we impose

$$\gamma_1 = 0, \gamma_C = 0,$$

where C represents the most recent cohort in the data.

For the period terms, we consider a multivariate random walk with drift:

$$\boldsymbol{\kappa}_t = \boldsymbol{c} + \boldsymbol{\kappa}_{t-1} + \boldsymbol{\epsilon}_t^{\kappa}, \quad \boldsymbol{\kappa}_t = \begin{pmatrix} \kappa_t^{(1)} \\ \kappa_t^{(2)} \end{pmatrix}, \quad \boldsymbol{\epsilon}_t^{\kappa} \sim N\left(\mathbf{0}, \Sigma\right),$$

with normal priors: $\boldsymbol{c} \sim N(0, 10)$.

The variance-covariance matrix of the error term is defined by

$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho_\Sigma \sigma_1 \sigma_2 \\ \rho_\Sigma \sigma_1 \sigma_Y & \sigma_2^2 \end{pmatrix}$$

where the variance coefficients have independent exponential priors: $\sigma_1, \sigma_2 \sim Exp(0.1)$ and the correlation parameter has a uniform prior: $\rho_\Sigma \sim U[-1, 1]$. As for the other models, the overdispersion parameter has a prior distribution given by

$$\frac{1}{\phi} \sim Half - N(0, 1).$$

For the cohort term, we consider a second order autoregressive process (AR(2)):

$$\gamma_c = \psi_1 \gamma_{c-1} + \psi_2 \gamma_{c-2} + \epsilon_t^\gamma, \quad \epsilon_t^\gamma \sim N(0, \sigma_\gamma).$$

To close the model specification, we impose some vague priors assumptions on the hyperparameters:

$$\psi_1, \psi_2 \sim N(0, 10), \quad \sigma_\gamma \sim Exp(0.1).$$

Value

An object of class `stanfit` returned by `rstan::sampling`.

References

Cairns, A. J. G., Blake, D., Dowd, K., Coughlan, G. D., Epstein, D., Ong, A., & Balevich, I. (2009). A quantitative comparison of stochastic mortality models using data from England and Wales and the United States. North American Actuarial Journal, 13(1), 1-35.

Examples

```
#10-year forecasts for French data for ages 50-90 and years 1970-2017 with a log-Poisson model
ages.fit<-70:90
years.fit<-1990:2010
deathFR<-FRMaleData$Dxt[formatC(ages.fit),formatC(years.fit)]
exposureFR<-FRMaleData$Ext[formatC(ages.fit),formatC(years.fit)]
iterations<-50 # Toy example, consider at least 2000 iterations
fitM6=m6_stan(death = deathFR,exposure=exposureFR, age=ages.fit,forecast = 5,
family = "poisson",iter=iterations,chains=1)
```

`mortality_weights` *Model averaging/weighting via future-out stacking or pseudo-BMA weighting*

Description

Mortality Model averaging via stacking of predictive distributions and Pseudo-BMA weighting. Based on Yao et al. (2018) but adapted by Barigou et al. (2021) for mortality forecasting.

Usage

```
mortality_weights(X)
```

Arguments

`X`	A list of stanfit objects.

Details

Mortality model averaging via stacking of predictive distributions or pseudo-BMA weighting. Both approaches were proposed in Yao et al. (2018) based leave-one-out cross-validation which is not suited for forecasting. Barigou et al. (2021) adapted both approaches based on leave-future-out validation which is more appropriate for mortality forecasting.

The stacking method combines all models by maximizing the leave-future-out predictive density of the combination distribution. That is, it finds the optimal linear combining weights for maximizing the leave-future-out log score.

The pseudo-BMA method finds the relative weights proportional to the expected log predictive density of each model.

Similar to Yao et al. (2018), we recommend stacking for averaging predictive distributions as pseudo-BMA tends to select only one model.

Value

A matrix containing one weight for each model and each approach.

References

Yao, Y., Vehtari, A., Simpson, D., & Gelman, A. (2018). Using stacking to average Bayesian predictive distributions (with discussion). Bayesian Analysis, 13(3), 917-1007.

Barigou K., Goffard P-O., Loisel S., Salhi Y. (2021). Bayesian model averaging for mortality forecasting using leave-future-out validation. arXiv preprint arXiv:2103.15434.

Examples

```
#10-year forecasts for French data for ages 50-90 and years 1970-2017 with a log-Poisson model
#where the 10 last years are held out for validation. We search for the model weights between
#the Lee-Carter model and the RH model (Lee-Carter with cohort effect).
ages.fit<-50:90
years.fit<-1970:2017
deathFR<-FRMaleData$Dxt[formatC(ages.fit),formatC(years.fit)]
exposureFR<-FRMaleData$Ext[formatC(ages.fit),formatC(years.fit)]
iterations<-1000 # Toy example, consider at least 2000 iterations
fitLC=lc_stan(death = deathFR,exposure=exposureFR, forecast = 10, validation=10,
family = "poisson",iter=iterations,chains=1)
fitRH=rh_stan(death = deathFR,exposure=exposureFR, forecast = 10, validation=10,
family = "poisson",iter=iterations,chains=1)
model_weights<-mortality_weights(list(fitLC,fitRH))
```

---

rh_stan *Bayesian Renshaw-Haberman model with Stan*

Description

Fit and Forecast Bayesian Renshaw-Haberman model (Lee-Carter with cohort effect) introduced in Renshaw and Haberman (2006). The model can be fitted with a Poisson or Negative-Binomial distribution. The function outputs posteriors distributions for each parameter, predicted death rates and log-likelihoods.

Usage

```
rh_stan(
  death,
  exposure,
  forecast,
  validation = 0,
  family = c("poisson", "nb"),
  ...
)
```

Arguments

death	Matrix of deaths.
exposure	Matrix of exposures.
forecast	Number of years to forecast.
validation	Number of years for validation.

`family`	specifies the random component of the mortality model. `"Poisson"` assumes a Poisson model with log link and `"nb"` assumes a negative-binomial model with log link and overdispersion parameter ϕ.
`...`	Arguments passed to `rstan::sampling` (e.g. iter, chains).

Details

The created model is either a log-Poisson or a log-Negative-Binomial version of the Renshaw-Haberman model:

$$D_{x,t} \sim \mathcal{P}(\mu_{x,t} e_{x,t})$$

or

$$D_{x,t} \sim NB\left(\mu_{x,t} e_{x,t}, \phi\right)$$

with

$$\log \mu_{xt} = \alpha_x + \beta_x \kappa_t + \gamma_{t-x}.$$

To ensure the identifiability of th model, we impose

$$\kappa_1 = 0, \gamma_1 = 0, \sum gamma_i = 0, \gamma_C = 0,$$

where C represents the most recent cohort in the data.

For the priors, the model chooses wide priors:

$$\alpha_x \sim N(0, 100), \beta_x \sim Dir(1, \ldots, 1), \frac{1}{\phi} \sim Half - N(0, 1).$$

For the period term, we consider the standard random walk with drift:

$$\kappa_t = c + \kappa_{t-1} + \epsilon_t, \epsilon_t \sim N(0, \sigma^2)$$

with $c \sim N(0, 10), \sigma \sim Exp(0.1)$.

For the cohort term, we consider a second order autoregressive process (AR(2)):

$$\gamma_c = \psi_1 \gamma_{c-1} + \psi_2 \gamma_{c-2} + \epsilon_t^{\gamma}, \quad \epsilon_t^{\gamma} \sim N(0, \sigma_\gamma).$$

To close the model specification, we impose some vague priors assumptions on the hyperparameters:

$$\psi_1, \psi_2 \sim N(0, 10), \quad \sigma_\gamma \sim Exp(0.1).$$

Value

An object of class `stanfit` returned by `rstan::sampling`.

References

Renshaw, A. E., & Haberman, S. (2006). A cohort-based extension to the Lee-Carter model for mortality reduction factors. Insurance: Mathematics and Economics, 38(3), 556-570.

Examples

```
#10-year forecasts for French data for ages 50-90 and years 1970-2017 with a log-Poisson model
ages.fit<-70:90
years.fit<-1990:2010
deathFR<-FRMaleData$Dxt[formatC(ages.fit),formatC(years.fit)]
exposureFR<-FRMaleData$Ext[formatC(ages.fit),formatC(years.fit)]
iterations<-50 # Toy example, consider at least 2000 iterations
fitRH=rh_stan(death = deathFR,exposure=exposureFR, forecast = 5, family = "poisson",
iter=iterations,chains=1)
```

`sim_death_apc` *Simulation of death counts from the Age-Period-Cohort mortality model*

Description

Simulation of death counts from the Age-Period-Cohort mortality model

Usage

```
sim_death_apc(a, k, g, phi, years, ages, exposure)
```

Arguments

`a`	vector of age component
`k`	vector of period component
`g`	vector of cohort component
`phi`	dispersion parameter
`years`	vector of calendar years
`ages`	vectors of ages
`exposure`	matrix of exposure data

Value

matrix of death count

sim_death_cbd *Simulation of death counts from the CBD model*

Description

Simulation of death counts from the CBD model

Usage

```
sim_death_cbd(k, k2, phi, years, ages, exposure)
```

Arguments

k	first vector of period component
k2	second vector of period component
phi	dispersion parameter
years	vector of calendar years
ages	vectors of ages
exposure	matrix of exposure data

Value

matrix of death count

sim_death_lc *Simulation of death counts from the Lee-Carter mortality model*

Description

Simulation of death counts from the Lee-Carter mortality model

Usage

```
sim_death_lc(a, b, k, phi, exposure)
```

Arguments

a	vector of age component
b	vector of age/year component
k	vector of year component
phi	dispersion parameter
exposure	matrix of exposure data

Value

matrix of death count

sim_death_m6 *Simulation of death counts from the M6 model*

Description

Simulation of death counts from the M6 model

Usage

```
sim_death_m6(k, k2, g, phi, years, ages, exposure)
```

Arguments

k	first vector of period component
k2	second vector of period component
g	vector of cohort component
phi	dispersion parameter
years	vector of calendar years
ages	vectors of ages
exposure	matrix of exposure data

Value

matrix of death count

sim_death_mix_cbd_rh *Simulation of death counts from a hybrid model that averages the mortality rates from the cbd and rh models*

Description

Simulation of death counts from a hybrid model that averages the mortality rates from the cbd and rh models

Usage

```
sim_death_mix_cbd_rh(params_cbd, params_rh, years, ages, exposure, q)
```

Arguments

params_cbd	named list that contains the parameters of the cbd model
params_rh	named list that contains the parameters of the rh model
years	vector of calendar year
ages	vector of ages
exposure	matrix of exposure data
q	mixing parameter (0 <- rh, 1 <- cbd)

Value

matrix of death count

`sim_death_rh` *Simulation of death counts from the Renshaw-Haberman mortality model*

Description

Simulation of death counts from the Renshaw-Haberman mortality model

Usage

```
sim_death_rh(a, b, k, g, phi, years, ages, exposure)
```

Arguments

`a`	vector of age component
`b`	vector of age/year component
`k`	vector of period component
`g`	vector of cohort component
`phi`	dispersion parameter
`years`	vector of calendar years
`ages`	vectors of ages
`exposure`	matrix of exposure data

Value

matrix of death count

`sim_mortality_data` *Simulation of mortality data from various models*

Description

Simulation of mortality data from various models

Usage

```
sim_mortality_data(a, k, k2, b, g, phi, years, ages, exposure, mortality_model)
```

Arguments

a	vector of age component
k	first vector of time component
k2	second vector of time component
b	vector of age/time component
g	vector of cohort component
phi	dispersion parameter
years	vector of calendar year
ages	vector of ages
exposure	matrix of exposure
mortality_model	name of the mortality model that we simulate from

Value

matrix of death counts

Index

∗ **datasets**
FRMaleData, 10

apc_stan, 3

boxplot_post_dist, 5

cbd_stan, 6
compute_weights_BMA, 8

extract_map, 8

fit_mo_mo, 9
FRMaleData, 10

lc_stan, 11

m6_stan, 12
mortality_weights, 15

rh_stan, 16

sim_death_apc, 18
sim_death_cbd, 19
sim_death_lc, 19
sim_death_m6, 20
sim_death_mix_cbd_rh, 20
sim_death_rh, 21
sim_mortality_data, 21
StanMoMo *(*StanMoMo-package*)*, 2
StanMoMo-package, 2