Package ‘BayesMortalityPlus’

May 9, 2023

Type Package

Title Bayesian Mortality Modelling

Version 0.1.0

Author Laboratorio de Matematica Aplicada (IM/UFRJ)

Maintainer Luiz Fernando Figueiredo <labmapackage@gmail.com>


License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0), utils, ggplot2, magrittr, dplyr

Imports MASS, progress, tidyr, scales

RoxygenNote 7.2.3

Collate 'Heatmap_default.R' 'PT.R' 'USA.R' 'blc.R' 'ffbs.R' 'dlm.R'
'gompertz_dlm.R' 'dlm_close.R' 'expectancy_blc.R'
'expectancy_default.R' 'fitted_dlm.R' 'fun_aux.R' 'qx_ci.R'
'expectancy_dlm.R' 'fitted_hp.R' 'expectancy_hp.R'
'fitted_blc.R' 'globals.R' 'heatmap_blc.R' 'heatmap_dlm.R'
'heatmap_hp.R' 'heatmap_list.R' 'hello.R'
'hp_poisson_reduced.R' 'hp_lognormal_reduced.R'
'hp_binomial_reduced.R' 'hp_poisson.R' 'hp_lognormal.R'

1
R topics documented:

'hp_binomial.R' 'hp.R' 'sir_gompertz.R' 'hp_close.R' 'hp_mix.R'
'improvement.R' 'kd_filter.R' 'kd_smoother.R' 'mean_blc.R'
'mean_blc_pred.R' 'plot_blc.R' 'plot_blc_pred.R' 'plot_chain.R'
'plot_dlm.R' 'plot_hp.R' 'plot_list.R' 'predict_blc.R'
'predict_dlm.R' 'print_blc.R' 'print_blc_pred.R' 'print_dlm.R'
'print_hp.R' 'quantile_blc.R' 'quantile_blc_pred.R'
'summary_dlm.R' 'summary_hp.R'

NeedsCompilation: no

Repository: CRAN

Date/Publication: 2023-05-09 09:30:06 UTC

R topics documented:

blc ......................................................... 3
dlm .......................................................... 5
dlm_close .................................................... 8
expectancy ................................................. 10
expectancy.BLC ............................................. 11
expectancy.DLM .............................................. 12
expectancy.HP .............................................. 13
fitted.BLC ................................................... 15
fitted.DLM ................................................... 16
fitted.HP ..................................................... 17
Heatmap ..................................................... 18
Heatmap.BLC ............................................... 18
Heatmap.DLM ............................................... 19
Heatmap.HP ................................................... 20
Heatmap.list ............................................... 22
hp ............................................................ 23
hp_close ..................................................... 26
hp_mix ....................................................... 28
improvement ............................................... 30
kd.filter ..................................................... 30
kd.smoother ................................................ 31
mean.BLC .................................................... 32
mean.PredBLC ............................................... 33
plot.BLC ..................................................... 34
plot.DLM ..................................................... 35
plot.HP ....................................................... 36
plot.list ..................................................... 38
plot.PredBLC ............................................... 40
plot_chain ................................................... 41
predict.BLC ................................................ 42
predict.DLM ................................................ 43
print.BLC .................................................... 44
print.DLM .................................................... 45
print.HP ..................................................... 46
Description

Performs Bayesian estimation of the Lee-Carter model considering different variances for each age.

Usage

```r
blc(Y, prior = NULL, init = NULL, numit = 2000, warmup = 1000)
```

Arguments

- `Y`: Log-mortality rates for each age.
- `prior`: A list containing the prior mean $m_0$ and the prior variance $C_0$.
- `init`: A list containing initial values for $\alpha$, $\beta$, $\phi_V$, $\phi_W$ and $\theta$.
- `numit`: The number of iterations.
- `warmup`: The number of initial iterations from the Gibbs sampler that should be discarded. Default considers half the number of iterations.

Details

Let $Y_{it}$ be the log mortality rate at age $i$ and time $t$. The Lee-Carter model is specified as follows:

$$Y_{it} = \alpha_i + \beta_i \kappa_t + \varepsilon_{it}, \ i = 1, \ldots, p \text{ and } t = 1, \ldots, T,$$

where $\alpha = (\alpha_1, \ldots, \alpha_p)'$ are the intercept of the model that represent the log-mortality rate mean in each age; $\beta = (\beta_1, \ldots, \beta_p)'$ are the coefficient regression that represent the speed of relative change in the log-mortality rate in each age. $\kappa = (\kappa_1, \ldots, \kappa_T)'$ are the state variable that represent the global relative change in log-mortality rate. Finally, $\varepsilon_{it} \sim N(0, \sigma_i^2)$ is the random error.

For the state variable $\kappa_t$ Lee and Carter (1992) proposed a random walk with drift to govern the dynamics over time:

$$\kappa_t = \kappa_{t-1} + \theta + \omega_t,$$

where $\theta$ is the drift parameter and $\omega_t$ is the random error of the random walk.

We implement the Bayesian Lee Carter (BLC) model, proposed by Pedroza (2006), to estimate the model. In this approach, we take advantage of the fact that the Bayesian Lee Carter can be specified as dynamic linear model, to estimate the state variables $\kappa_t$ through FFBS algorithm. To estimate the others parameters we use Gibbs sampler to sample from their respective posterior distribution.
Value

A BLC object.

- **alpha**: Posterior sample from alpha parameter.
- **beta**: Posterior sample from beta parameter.
- **phiv**: Posterior sample from phiv parameter. phiv is the precision of the random error of the Lee Carter model.
- **theta**: Posterior sample from theta.
- **phiw**: Posterior sample from phiw. phiw is the precision of the random error of the random walk.
- **kappa**: Sampling from the state variables.
- **Y**: Y Log-mortality rates for each age passed by the user to fit the model.
- **warmup**: The warmup of the algorithm specified by the user to fit the model.
- **numit**: The number of iterations specified by the user to fit the model.
- **m0**: The prior mean of kappa0.
- **C0**: The prior covariance matrix of kappa0.

References


See Also

- `fitted.BLC()`, `plot.BLC()`, `print.BLC()` and `predict.BLC()` for BLC methods to native R functions `fitted()`, `plot()`, `print()` and `predict()`.
- `expectancy.BLC()` and `Heatmap.BLC()` for BLC methods to compute and visualise the truncated life expectancy via `expectancy()` and `Heatmap()` functions.
- `improvement()` to compute the improvement of each age, based on the resulting chains of the beta parameter.

Examples

```
## Example of transforming the dataset to fit the function:

## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Calculating the mortality rates for the general population:
require(dplyr)
require(tidyr)
require(magrittr)
USA %>% mutate(qx = USA$Dx.Total/USA$Ex.Total) -> data
```
### dlm

**Dynamic Linear Model for mortality table graduation**

#### Description

This function fits a Dynamic Linear Model (DLM) for mortality data following a Bayesian framework using Forward Filtering Backward Sampling algorithm and Markov chain Monte Carlo Gibbs sampler to compute the posterior distribution. The response variable is the log of mortality rate, and it is modeled specifying the matrices $F_t$ and $G_t$ from the DLM equations. Furthermore, the discount factor is used to control the smoothness of the fitted model. By default, a linear growth model is specified.

#### Usage

```r
dlm(y, Ft = matrix(c(1, 0), nrow = 1), Gt = matrix(c(1, 0, 1, 1), 2),
    delta = 0.85, prior = list(m0 = rep(0, nrow(Gt)), C0 = diag(100, nrow(Gt))),
    prior.sig2 = list(a = 0.01, b = 0.01), M = 5000, bn = 3000, thin = 1,
    ages = 0:(length(y)-1))
```

#### Arguments

- **y**: Numeric vector of log mortality rates.  
- **Ft**: 1xp Matrix that specifies the observation equation, where $p$ is the number of parameters. By default, $F_t = \text{matrix}(c(1,0), \text{nrow} = 1)$.  
- **Gt**: pxp Matrix that specifies the system equations. By default, $G_t = \text{matrix}(c(1,0,1,1), 2)$.  
- **delta**: Positive number in $(0, 1)$ interval specifying the discount factor. A higher value of delta results in a higher smoothness of the fitted curve. By default, delta is '0.65'.
prior A list with the prior mean vector \((m_0)\) and covariance matrix \((C_0)\) of \(\theta_0\) (state vector at time (age) \(t = 0\)). By default mean of zeros and diagonal matrix with a common variance 100 is used. Each element of the list must be named accordingly with the parameter (m0 for mean vector and C0 for covariance matrix).

prior.sig2 A list with the prior parameters \((a, b)\) of Inverted Gamma distribution for \(\sigma^2\). Each element of the list must be named accordingly with the parameter (a for shape parameter and b for scale parameter).

M Positive integer indicating the number of iterations of the MCMC run.

bn Non-negative integer indicating the number of iteration to be discarded as the burn-in period.

thin Positive integer specifying the period for saving samples.

ages Numeric vector of the ages fitted. Default is ’0:(length(y)-1)’.

Details

Let \(Y_t\) be the log mortality rate at age \(t\). A DLM is specified as follows:

For \(t = 0\):

\[ \theta_0 \sim N_p(m_0, C_0) \]

Now, for \(t \geq 1\):

The observation equation:

\[ Y_t = F_t \theta_t + v_t \]

The system equation:

\[ \theta_t = G_t \theta_{t-1} + w_t \]

Where \(F_t\) and \(G_t\) are known matrices. \(v_t\) and \(w_t\) are independent random errors with \(v_t \sim N(0, V_t)\) and \(w_t \sim N(0, W_t)\). Assumes \(V_t = \sigma^2 V'\), constant for all ages \(t\) and \(V'\) known. We use the discount factor \(\delta\) to specify \(W_t\) as \(W_t = C_T (1 - \delta)/\delta\), where \(C_t\) is the conditional covariance matrix of \(\theta_t\). So, if \(\delta = 0\) there is no loss information as \(t\) increase (completely reducing the smoothness of the fitted curve).

To adjust the model adopted a scheme described by (Petris et al, 2009) that uses the FFBS algorithm and a Gibbs sampler step to sample from the parameters of the posterior distribution. For more details, see (Petris et al, 2009).

Value

A DLM class object.

\(\mu\) Posterior samples from \(\mu_t = F_t \theta_t\), for all \(t\).

\(\theta\) Posterior samples from \(\theta_t\), for all \(t\).

\(\sigma^2\) Posterior samples from \(\sigma^2\).

\(W\) Posterior samples from matrices \(W_t\).

\(info\) A list with some informations of the fitted model: the specification of \(F_t\) and \(G_t\) matrices, the data \(y\) and the ages, the discount factor \(\delta\) value specified and priors informations.
References


See Also

fitted.DLM(), predict.DLM(), plot.DLM(), print.DLM() and summary.DLM() for DLM methods to native R functions fitted(), plot(), print() and summary().

effectancy.DLM() and Heatmap.DLM() for DLM methods to compute and visualise the truncated life expectancy via expectancy() and Heatmap() functions.

dlm_close() for close methods to expand the life tables.

qx_ci() and plot_chain() to compute credible intervals and visualise the markov chains, respectively.

Examples

## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the 2010 male population ranging from 0 to 100 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:100
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]
qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fitting DLM
fit = dlm(y, M = 100, bn = 20, thin = 1)
print(fit)
summary(fit)

## Using other functions available in the package:
## plotting (See "?plot.DLM" in the BayesMortality package for more options):
plot(fit)

## qx estimation (See "?fitted.DLM" in the BayesMortality package for more options):
fitted(fit)

## chain's plot (See "?plot_chain" for more options):
plot_chain(fit, param = c("mu[0]", "mu[100]"))

## credible intervals (See "?qx_ci" for more options):
qx_ci(fit)
dlm_close

DLM: Fitting the advanced ages of the life tables.

Description

This function receives an object of the class DLM fitted by the dlm() function and fits a closing method to expand the life tables dataset to a maximum age argument inputed by the user. There are three closing methods available: 'linear', 'gompertz' and 'plateau'.

Usage

dlm_close(fit, method = c("linear", "gompertz", "plateau"),
  x0 = max(fit$info$ages), max_age = 120, k = 7,
  weights = seq(from = 0, to = 1, length.out = k),
  new_data = NULL)

Arguments

fit Object of the class DLM adjusted by the dlm() function.
method Character string specifying the closing method to be fitted, with them being: 'plateau', 'linear' or 'gompertz'.
x0 Integer with the starting age the closing method will be fitted from. Default is the last age fitted by the 'DLM' object.
max_age Integer with the maximum age the closing method will be fitted. Default age is '120'.
k Integer representing the size of the age-interval to be mixed with the 'linear' or 'gompertz' closing methods for a smooth graduation. If k = 0, no mixing will be made. Default: 7.
weights Vector of weights to be applied in the mixing of the life tables. Vector's size should be equal to k.
new_data Vector containing the log mortality rates in the period after the x0 input. This argument is necessary in the 'linear' and 'gompertz' closing methods.

Details

The three closing methods implemented by the function are: 1.'linear' method: Fits a linear regression starting at age x0 - k until the last age with data available 2.'gompertz' method: Used as the closing method of the 2010-2012 English Life Table No. 17, fits the Gompertz mortality law via SIR using the same available data as the 'linear' method. 3.'plateau' method: Keeps the death probability (qx) constant after the x0 argument.

Value

Returns a ClosedDLM class object with the predictive chains of the death probability (qx) from first fitted age to max_age argument, the data information utilized by the function and the closing method chosen.
References


See Also

fitted.DLM(), plot.DLM(), print.DLM() and summary.DLM() for ClosedDLM methods to native R functions fitted(), plot(), print() and summary().

expectancy.DLM() and Heatmap.DLM() for ClosedDLM methods to compute and visualise the truncated life expectancy via expectancy() and Heatmap() functions.

qx_ci() to compute credible intervals.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)
## Selecting the exposure and the death count of the year 2010, ranging from 0 to 90 years old:
USA2010 = USA[USA$Year == 2010,]
x = 0:100
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]
qx_t <- Dx/Ex
qx_t <- 1 - exp(-qx_t)
y <- log(qx_t)
fit <- dlm(y, M = 100, bn = 20, thin = 1)
## Applying the closing function with different methods:
close1 = dlm_close(fit, method = "plateau")
### Getting new data for the linear and gompertz methods:::
x2 = 101:110
Ex2 = USA2010$Ex.Male[x2+1]
Dx2 = USA2010$Dx.Male[x2+1]
qx2_t <- Dx2/Ex2
qx2_t <- 1 - exp(-qx2_t)
y2 <- log(qx2_t)
close2 = dlm_close(fit, method = "linear",
                  new_data = y2)
#### Using the other functions available in the package with the 'ClosedDLM' object:
## credible intervals (See "?qx_ci" for more options):
qx_ci(close1)
## qx estimation (See "?fitted" in the BayesMortalityPlus package for more options):
```

fitted(close2)

## life expectancy (See "?expectancy.DLM" for more options)
expectancy(close2, age = seq(0,120,by=20), graph = FALSE)

## plotting (See "?plot" in the BayesMortalityPlus package for more options):
plot(list(close1, close2, fit),
     colors = c("red4","seagreen", "blue"),
     labels = c("Plateau method","Linear method", "DLM fitted"),
     plotData = FALSE)

---

### Description

Generic function to expectancy method.

### Usage

```r
expectancy(x, ...)
```

### Arguments

- **x**: Object of one of these class: HP, DLM, BLC, ClosedHP, ClosedDLM, BLC, or PredBLC.
- **...**: Further arguments passed to or from other methods.

### Details

This function computes the life expectancy given by:

$$e_x = \sum t_p x$$

where:

$$t_p x = p_0 x p_1 x ... x p_x$$

### Value

A data.frame and (if graph = TRUE) a plot for HP, DLM, ClosedHP and ClosedDLM methods. A list that contains three vectors with the fitted values of life expectancy and the lower and upper limits of the credible intervals for each year used in fitted model or for the prediction, for BLC and PredBLC methods.

### See Also

`expectancy.HP()`, `expectancy.DLM()` and `expectancy.BLC()`.
expectancy.BLC  

Life Expectancy for fitted models and forecast

Description
Computes the fitted life expectancy for a specific age for each year in fit or prediction. It also calculates the limits of credible intervals.

Usage
```r
## S3 method for class 'BLC'
expectancy(x, at = NULL, cred = 0.95, ...)
```

Arguments
- `x`: A BLC or PredBLC object.
- `at`: A number that determines at which age the expectancy life is calculated based on the ages used in fit or prediction. For instance, `at = 0` is related to the first age used in fitted model.
- `cred`: A number that specifies the probability of the credible interval. Default is ‘0.95’.
- `...`: Further arguments passed to or from other methods.

Value
A list that contains three vectors with the fitted values of life expectancy and the lower and upper limits of the credible intervals for each year used in fitted model or for the prediction.

See Also
- `expectancy.HP()` and `expectancy.DLM()` for HP and DLM methods.
- `Heatmap.BLC()` for BLC method to drawing a Heatmap for the truncated life expectancy.

Examples
```r
## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Life expectancy for the years used in model fitted
expectancy(fit)

## Life expectancy for the tenth and thirtieth age in the years used in model fitted (27 and 47 y.o.)
expectancy(fit, at = c(10,30))
```
**Description**

This function computes the life expectancy for each age for Dynamic Linear model.

**Usage**

```r
## S3 method for class 'DLM'
expectancy(
  x,
  age = seq(0, max(fit$info$ages), by = 10),
  graph = TRUE,
  max_age = 110,
  prob = 0.95,
  ...
)
```

**Arguments**

- `x` Object of the following classes: DLM or ClosedDLM.
- `age` Numeric vector specifying the ages to calculate the life expectancy. The default is a sequence (0, 10, 20, ...) until the last decade used in the fitted model.
- `graph` Logical value (TRUE ou FALSE). If TRUE, it also returns a plot. The default value is TRUE.
- `max_age` Positive number indicating the last age to be considered to compute the life expectancy (prediction will be considered to match the age interval if needed). This argument is only necessary with objects of the class DLM.
- `prob` A number specifying the probability of credible interval. The default value is 0.95.
- `...` Further arguments passed to or from other methods.

**Value**

A data.frame and (if graph = TRUE) a plot.

**See Also**

- `expectancy.HP()` and `expectancy.BLC()` for HP and BLC methods.
- `Heatmap.DLM()` and `Heatmap.list()` for DLM or list methods to drawing a Heatmap for the truncated life expectancy.
Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

# Example 1: --------------------------------
USA1990 = USA[USA$Year == 1990,]
Ex = USA1990$Ex.Total[1:111]
Dx = USA1990$Dx.Total[1:111]
qx_t <- Dx/Ex
qx_t <- 1 - exp(-qx_t)
y <- log(qx_t)
fit <- dlm(y, M = 100, bn = 20, thin = 1)
expectancy(fit)

# Example 2: -------------------------------
# Using some arguments:
expectancy(fit, age = c(0,20,30,60),
prob = 0.99, max_age = 90, graph = FALSE)
```

Description

This function computes the life expectancy for each age for Heligman-Pollard model.

Usage

```r
## S3 method for class 'HP'
expectancy(
  x,
  Ex = NULL,
  age = NULL,
  graph = TRUE,
  max_age = 110,
  prob = 0.95,
  ...
)
```
Arguments

  x          Object of the class HP or ClosedHP fitted by hp() or hp_close() functions.
  Ex         Numeric vector with the exposure by age. This argument is only necessary when
              using poisson and binomial models with objects of the class HP.
  age        Numeric vector specifying the ages to calculate the life expectancy. The default
              is a sequence (0, 10, 20,...) until the last decade used in the fitted model.
  graph      Logical value (TRUE ou FALSE). If TRUE, it returns a plot.
  max_age    Positive number indicating the last age to be considered to compute the life ex-
              pectancy (extrapolation will be considered to match the age interval if needed).
              This argument is only necessary with objects of the class HP.
  prob       A percentage specifying the probability of credible interval.
  ...        Further arguments passed to or from other methods.

Value

  A data.frame and (if graph = TRUE) a plot.

See Also

  expectancy.DLM() and expectancy.BLC() for DLM and BLC methods.
  Heatmap.HP() and Heatmap.list() for HP or list methods to drawing a Heatmap for the truncated life expectancy.

Examples

  ## Importing mortality data from the USA available on the Human Mortality Database (HMD):
  data(USA)

  # Example 1: -----------------------------
  USA1990 = USA[USA$Year == 1990,]
  Ex = USA1990$Ex.Total[1:91]
  Dx = USA1990$Dx.Total[1:91]
  x = 0:90
  fit <- hp(x, Ex, Dx, model = "binomial", M = 1000, bn = 0, thin = 10)
  expectancy(fit)

  # Example 2: -----------------------------
  # Using some arguments:
  Ex = USA1990$Ex.Total[1:106]
  expectancy(fit, Ex = Ex, age = c(0,20,30,60,105),
              max_age = 105, prob = 0.99, graph = FALSE)
Description

Computes the fitted values associated to each age and year based on the resulting chains from a fitted BLC model. In addition, this function also evaluates the values of lower and upper limits of the credible interval.

Usage

## S3 method for class 'BLC'
fitted(object, cred = 0.95, ...)

Arguments

object A BLC object, result of a call to blc() function.
cred A real number that indicates the probability of the credible interval.
... Other arguments.

Value

A list with the matrices of fitted values and lower and upper limits of the credible interval for each age and year.

See Also

fitted.HP() and fitted.DLM() for HP or DLM methods.

Examples

## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Log-mortality estimates for each age and year in model fitted
fitted(fit, cred = 0.95)
Description

This function computes the point estimations of the death probabilities (qx) of a mortality graduation returned by dlm() or dlm_close() functions.

Usage

```r
## S3 method for class 'DLM'
fitted(object, age = NULL, ...)
```

Arguments

- `object`: Object of the following classes: DLM or ClosedDLM.
- `age`: Vector with the ages to calculate the death probabilities (Optional). By default, all ages are considered.
- `...`: Other arguments.

Value

A data.frame object with the selected ages and the corresponding death probabilities.

See Also

- `fitted.HP()` and `fitted.BLC()` for HP or BLC methods.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the year 2000, ranging from 0 to 100 years old:
USA2000 = USA[USA$Year == 2000,]
x = 0:100
Ex = USA2000$Ex.Total[x+1]
Dx = USA2000$Dx.Total[x+1]
qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fitting dlm
fit = dlm(y, M = 100, bn = 20, thin = 1)

## Estimating the death probabilities (qx)
fitted(fit)
```
Description

This function computes the point estimations of the death probabilities (qx) of the HP or the ClosedHP class object fitted by the hp() or hp_close() functions.

Usage

```r
## S3 method for class 'HP'
fitted(object, age = NULL, ...)
```

Arguments

- `object`: Object of the class HP or ClosedHP adjusted by the hp() or hp_close() functions.
- `age`: Vector with the ages to calculate the death probabilities (Optional). By default, all ages are considered.
- `...`: Other arguments.

Value

A data.frame object with the selected ages and the corresponding death probabilities.

See Also

`fitted.BLC()` and `fitted.DLM()` for BLC or DLM methods.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and the death count of the year 2000, ranging from 0 to 90 years old:
USA2000 = USA[USA$Year == 2000,]
x = 0:90
Ex = USA2000$Ex.Total[x+1]
Dx = USA2000$Dx.Total[x+1]

## Fitting a simple model:
fit = hp(x = x, Ex = Ex, Dx = Dx, M = 5000, bn = 0, thin = 10)

## Estimating the death probabilities (qx)
fitted(fit)
fitted(fit, age = 0:110)
```
Generic Heatmap function

Description

Generic function to Heatmap method.

Usage

Heatmap(x, ...)

Arguments

x Object or list of objects of class HP, DLM, ClosedHP or ClosedDLM. Object of class BLC or PredBLC.
... Further arguments passed to or from other methods.

Value

A ggplot2 heatmap of the life expectancy.

See Also

Heatmap.HP(), Heatmap.DLM(), Heatmap.BLC() and Heatmap.list().

Draw a Heat Map for BLC or PredBLC fitted models

Description

Draws a Heat Map based on the life expectancy of a fitted BLC or PredBLC model.

Usage

## S3 method for class 'BLC'
Heatmap(x, x_lab = NULL, age = NULL, color = c("red", "white", "blue"), ...)

Arguments

x A BLC or PredBLC object, result of a call to blc() function or forecast via predict() function.
x_lab Description of the modelled object.
age Vector with the ages to plot the heatmap.
color Vector of colours used in the heatmap.
... Further arguments passed to or from other methods.
Value

A ggplot2 heatmap of the life expectancy.

See Also

Heatmap.HP() and Heatmap.DLM() for HP or DLM methods.

Examples

## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Heatmap:
Heatmap(fit, x_lab = 2000:2015, age = 18:80)

Heatmap.DLM

DLM: Heatmap for the life expectancy.

Description

This function plots a heatmap for the life expectancy of the fitted DLMs.

Usage

## S3 method for class 'DLM'
Heatmap(
x,
x_lab,
age = NULL,
max_age = 110,
color = c("red", "white", "blue"),
...
)

Arguments

x
Object or a list of objects of the class DLM or ClosedDLM returned by dlm() or dlm_close() functions.

x_lab
Description of the object 'fit'.

age
Vector with the ages to plot the heatmap.
Heatmap.HP

max_agePositivenumberindicatingthelastagetobeconsideredtocomputethelife
expectancy(predictionwillbeconsideredtomatchtheageintervalifneeded).
ThisargumentisonlynecessarywithobjectsoftheclassDLM.
colorVectorofcoloursusedintheheatmap.
...
Furtherargumentspassedtoorfromothermethods.

Value

A ggplot2 heatmap of the life expectancy.

See Also

Heatmap.BLC()andHeatmap.HP()forBLCorHPmethods.
Heatmap.list()tothelistmethod,addingmultipleobjectsinonesingleHeatmap.

Examples

## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

# Example 1: -------------------------------
USA2010 = USA[USA$Year == 2010,]
ExF = USA2010$Ex.Female[1:91]
DxF = USA2010$Dx.Female[1:91]
qx_tF <- DxF/ExF
qx_tF <- 1 - exp(-qx_tF)
yF <- log(qx_tF)
fitF <- dlm(yF, M = 100, bn = 20, thin = 1)
Heatmap(fitF, x_lab = "Female expec. 2010 USA", max_age = 90)
**Heatmap.HP**

**Usage**

```r
## S3 method for class 'HP'
Heatmap(
  x,
  x_lab,
  age = 0:90,
  max_age = 110,
  color = c("red", "white", "blue"),
  ...
)
```

**Arguments**

- `x`: Object or a list of objects of the class `HP` or `ClosedHP` returned by `hp()` or `close_hp()` functions.
- `x_lab`: Description of the object 'fit'.
- `age`: Vector with the ages to plot the heatmap.
- `max_age`: Positive number indicating the last age to be considered to compute the life expectancy (extrapolation will be considered to match the age interval if needed). This argument is only necessary with objects of the class `HP`.
- `color`: Vector of colours used in the heatmap.
- `...`: Further arguments passed to or from other methods.

**Value**

A ggplot2 heatmap of the life expectancy.

**See Also**

- `Heatmap.BLC()` and `Heatmap.DLM()` for BLC or DLM methods.
- `Heatmap.list()` to the list method, adding multiple objects in one single Heatmap.

**Examples**

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

# Example: -------------------------------
USA2010 = USA[USA$Year == 2010,]
ExF = USA2010$Ex.Female[1:91]
DxF = USA2010$Dx.Female[1:91]
x <- 0:90
fitF <- hp(x, ExF, DxF, model = "lognormal", M = 1000, bn = 0, thin = 10)

Heatmap(fitF, x_lab = "Female expec. 2010 USA")
```
Heatmap.list

Heatmap for the life expectancy.

Description
This function plots a heatmap for the life expectancy of the mortality graduations returned by hp(), dlm(), hp_close() or dlm_close() functions.

Usage
## S3 method for class 'list'
Heatmap(
  x,
  x_lab,
  age = NULL,
  max_age = NULL,
  color = c("red", "white", "blue"),
  ...
)

Arguments
- **x** List of objects of classes: HP, DLM, ClosedHP, or ClosedDLM.
- **x_lab** Description of the object 'fit'.
- **age** Vector with the ages to plot the heatmap.
- **max_age** Positive number indicating the last age to be considered to compute the life expectancy (methods for matching the age interval will be considered if needed). This argument is only necessary with objects of the class HP or DLM.
- **color** Vector with colours used in the heatmap.
- **...** Further arguments passed to or from other methods.

Value
A ggplot2 heatmap of the life expectancy.

See Also
Heatmap.HP(), Heatmap.DLM() and Heatmap.BLC() for drawing single Heatmaps.

Examples
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

# Example (HP): -------------------------------
## Selecting the data from 2010
USA2010 = USA[USA$Year == 2010,]

ExF = USA2010$Ex.Female[1:91]
DxF = USA2010$Dx.Female[1:91]
x <- 0:90

fitF <- hp(x, ExF, DxF, model = "lognormal", M = 1000, bn = 0, thin = 10)

ExM = USA2010$Ex.Male[1:91]
DxM = USA2010$Dx.Male[1:91]

fitM <- hp(x, ExM, DxM, model = "lognormal", M = 1000, bn = 0, thin = 10)

fits <- list(fitF = fitF, fitM = fitM)

Heatmap(fits, x_lab = c("Female 2010 USA","Male 2010 USA"),
        age = 15:85)

---

### hp

#### Bayesian HP

**Description**

This function fits the Heligman-Pollard (HP) model following a Bayesian framework using Markov chain Monte Carlo techniques to sample the posterior distribution. Three model options are available: The Binomial and the Poisson models consider nine parameters, whereas the Log-Normal model considers eight parameters for modelling the HP curve.

**Usage**

```r
hp(x, Ex, Dx, model = c("binomial", "lognormal", "poisson"),
    M = 50000, bn = round(M/2), thin = 10, m = rep(NA, 8), v = rep(NA, 8),
    inits = NULL, K = NULL, sigma2 = NULL, prop.control = NULL,
    reduced_model = FALSE)
```

**Arguments**

- `x` Numeric vector of the ages.
- `Ex` Numeric vector with the exposure by age.
- `Dx` Numeric vector with the death counts by age.
- `model` Character string specifying the model to be adopted. The options are: "binomial", "lognormal" or "poisson".
- `M` Positive integer indicating the number of iterations of the MCMC run.
- `bn` Non-negative integer indicating the number of iteration to be discarded as the burn-in period.
### Details

The binomial model assumes that $D_x$, the death count for the age $x$, has a binomial distribution: $\text{Bin}(E_x, qx)$, where $qx$ is probability of death in age $x$. The poisson model assumes that $D_x$ has a Poisson distribution: $\text{Po}(E_xqx)$. Both models consider the nine parameters HP curve, that was proposed by Heligman and Pollard (1980):

$$
HP_x = A(x+B)^C + De^{-E (\log(x/F))^2} + GHx/(1 + KGx)
$$

$$
qx = 1 - e(-HP_x)
$$

This approximation ensures that $qx$, which is a probability, is in the correct range.

The Log-Normal model assumes that the log odds of death $q_x/(1-q_x)$ has Normal distribution with a constant variance for all the ages. This model was proposed by Dellaportas et al.(2001) and they consider the eighth parameters HP curve as follows:

$$
\log(q_x/(1 - q_x)) = \log(A(x+B)^C + De^{-E (\log(x/F))^2} + GHx) + \varepsilon_x,
$$

where $\varepsilon_x$ has independent distributions Normal(0, $\sigma^2$) for all ages. More details of this model are available in Dellaportas, Smith e Stavropoulos (2001).

The reduced model does not consider the first term of the HP curve: $A(x+B)^C$. In this case, $A$, $B$ and $C$ are fixed as zero.

All parameters, with the exception of the extra parameter $K$ of the Binomial and the Poisson models that is fixed in value 1, are estimated by the MCMC methods. Gibbs sampling for $\sigma^2$ and Metropolis-Hastings for parameters $A$, $B$, $C$, $D$, $E$, $F$, $G$ and $H$. Informative prior distributions should help the method to converge quickly.

### Value

A `HP` class object.

- `summary` A table with summaries measures of the parameters.
- `post.samples` A list with the chains generated by MCMC.
A table with the data considered in fitted model.

info A list with some informations of the fitted model like prior distributions mean and variance, initial values.

References


See Also

fitted.HP(), plot.HP(), print.HP() and summary.HP() for HP methods to native R functions fitted(), plot(), print() and summary().

expectancy.HP() and Heatmap.HP() for HP methods to compute and visualise the truncated life expectancy via expectancy() and Heatmap() functions.

hp_close() for close methods to expand the life tables and hp_mix() for mixing life tables.

qx_ci() and plot_chain() to compute credible intervals and visualise the markov chains, respectively.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and death count of the 2010 male population ranging from 0 to 90 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:90
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]

## Fitting binomial model
fit = hp(x = x, Ex = Ex, Dx = Dx)
print(fit)
fit$summary
fit$info

## Fitting lognormal model
## Specifying number of iterations, burn-in, thinning and the initial value of sigma2
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "lognormal",
         M = 1000, bn = 0, thin = 10, sigma2 = 0.1)
summary(fit)

## Fitting poisson model
## Specifying the prior distribution parameters for B and E and fixing K as 0.
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "poisson",
         m = c(NA, 0.08, NA, NA, 9, NA, NA, NA),
         v = c(NA, 1e-4, NA, NA, 0.1, NA, NA, NA), K = 0)
```
summary(fit)

## Using other functions available in the package:
## plotting (See "?plot.HP" in the BayesMortalityPlus package for more options):
plot(fit)

## qx estimation (See "?fitted.HP" in the BayesMortalityPlus package for more options):
fitted(fit)

## chain's plot (See "?plot_chain" for more options):
plot_chain(fit)

## credible intervals (See "?qx_ci" for more options):
qx_ci(fit)

---

### hp_close

**HP Model: Fitting the advanced ages of the life tables.**

**Description**

This function receives an object of the class HP fitted by the hp() function and fits a closing method to expand the life tables dataset to a maximum age argument inputted by the user. There are four closing methods available: 'hp', 'plateau', 'linear', and 'gompertz'. The 'linear' method can only be applied with HP objects following the lognormal variant of the HP mortality law.

**Usage**

```r
hp_close(fit, method = c("hp", "plateau", "linear", "gompertz"),
x0 = max(fit$data$x), max_age = 120, k = 7,
weights = seq(from = 0, to = 1, length.out = 2*k+1),
new_Ex = NULL, new_Dx = NULL)
```

**Arguments**

- `fit`: Object of the class HP fitted by the hp() function
- `method`: Character string specifying the closing method to be fitted, with them being: 'hp', 'plateau', 'linear' or 'gompertz'.
- `x0`: Integer with the starting age the closing method will be fitted from. Default is the last age fitted by the 'HP' object.
- `max_age`: Integer with the maximum age the closing method will be fitted. Default age is '120'.
- `k`: Integer representing the size of the age interval to be mixed with the 'linear' or 'gompertz' closing methods for smooth graduation. If k = 0, no mixing will be applied.
weights Vector of weights to be applied in the mixing of the life tables. Vector’s size should be equal to $2^k+1$.

new_Ex Exposure in the period after the $x_0$ input. This argument is necessary for HP objects following the binomial and poisson distributions, as well as the 'linear' and 'gompertz' closing methods (optional).

new_Dx Vector containing the death counts in the period after the $x_0$ input. This argument is necessary in the 'linear' and 'gompertz' closing methods, being equal in length with the new_Ex argument.

Details

The four closing methods for life tables are: 1. 'hp' method: Expands the previously adjusted HP model until the max_age argument. 2. 'plateau' method: Keeps the death probability ($q_x$) constant after the $x_0$ argument. 3. 'linear' method: Fits a linear regression starting at age $x_0 - k$ until the last age with data available (lognormal only). 4. 'gompertz' method: Adopted as the closing method of the 2010-2012 English Life Table No. 17, fits the Gompertz mortality law via SIR using the same available data as the 'linear' method.

Value

Returns a ClosedHP class object with the predictive chains of the death probability ($q_x$) from first fitted age to max_age argument, the data utilized by the function and the closing method chosen.

References


See Also

fitted.HP(), plot.HP(), print.HP() and summary.HP() for ClosedHP methods to native R functions fitted(), plot(), print() and summary().

expectancy.HP() and Heatmap.HP() for ClosedHP methods to compute and visualise the truncated life expectancy via expectancy() and Heatmap() functions.

qx_ci() to compute credible intervals.

Examples

## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and the death count of the year 2010, ranging from 0 to 90 years old:
USA2010 = USA[USA$Year == 2010,]
x = 0:90
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]

## Fitting a lognormal HP model:
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "lognormal", M = 1000, bn = 0, thin = 10)

## Applying the closing function with different methods:
close1 = hp_close(fit, method = "hp", x0 = 90)
close2 = hp_close(fit, method = "plateau", x0 = 90)
close3 = hp_close(fit, method = "linear", x0 = 80,
    new_Ex = USA2010$Ex.Male[82:101],
    new_Dx = USA2010$Dx.Male[82:101])
close4 = hp_close(fit, method = "gompertz", x0 = 70,
    new_Ex = USA2010$Ex.Male[72:101],
    new_Dx = USA2010$Dx.Male[72:101],
    k = 5, max_age = 120)

#### Using the other functions available in the package with the 'ClosedHP' object:

## credible intervals (See "?qx_ci" for more options):
qx_ci(close1)

## qx estimation (See "?fitted.HP" in the BayesMortalityPlus package for more options):
fitted(close2)

## life expectancy (See "?expectancy.HP" for more options)
expectancy(close3, age = 0:110)

## plotting (See "?plot.HP" in the BayesMortalityPlus package for more options):
plot(close4)

g <- plot(list(close4, fit),
    colors = c("seagreen", "blue"),
    labels = c("Closed", "Model"))

# plotly::ggplotly(g)

---

**hp_mix**

**HP Model mixture**

### Description

This function mixes the fitted mortality table of the HP model with another mortality table provided by the user.

### Usage

```r
hp_mix (fit, mu_post, weights = NULL, mix_age,
        x0_prior, x0_post, max_age)
```
hp_mix

Arguments

- **fit**: Object of the class 'HP' fitted by the hp() function.
- **mu_post**: Vector with mortality rates considered in the mix.
- **weights**: Positive vector specifying the weights considered in the mix.
- **mix_age**: Positive vector specifying the age range in the mixture.
- **x0_prior**: Non-negative number indicating the initial age of the fitted HP model.
- **x0_post**: Non-negative number indicating the initial age of the mortality table provided by the user.
- **max_age**: Positive number indicating the final age in the mixture.

Value

Return the posterior distribution for qx.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and death count of the 2010 and 2013 male populations ranging from 0 to 90 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:90
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]

USA2013 = USA[USA$Year == 2013,]
Ex2 = USA2013$Ex.Male[x+1]
Dx2 = USA2013$Dx.Male[x+1]

## Fitting HP model for 2010 data and calculating the mortality rates of 2013
fit = hp(x = x, Ex = Ex, Dx = Dx,
         M = 1000, bn = 0, thin = 10)
tx_2013 = 1 - exp(-Dx2/Ex2)

## Mixing fitted model and mortality rates of 2013:
mix <- hp_mix(fit, tx_2013, x0_prior = 0, x0_post = 0, mix_age = c(50,90),
               max_age = 90)

## Obtaining the new estimated mortality table (after mixture):
qx_mix<- apply(mix$qx, 2, median, na.rm = TRUE)
qx_mix
```
improvement  

*Improvement for BLC fitted models*

**Description**

Calculates the improvement of each age, based on the resulting chains of the beta parameter from a fitted BLC model.

**Usage**

```r
improvement(obj, cred = 0.95)
```

**Arguments**

- `obj`  
  A BLC object, result of a call to `blc()` function.
- `cred`  
  A real number that represents the credibility level of the intervals.

**Value**

A dataframe with the improvement values of each age, as well as their credible intervals.

**Examples**

```r
## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Improvement:
improvement(fit)
improvement(fit, cred = 0.9) #90% credible intervals
```

---

kd.filter  

*Kalman Filter with Drift*

**Description**

Computes mean and variance of the distribution of the state, conditional on the covariances of observation and system errors, evolution matrices, drifts and the observations up to that point.

**Usage**

```r
kd.filter(y, m0, C0, V, W, Ft, Gt, a1, a2)
```
Arguments

\begin{itemize}
\item \texttt{y} \hspace{1cm} \text{Observed data as an m-by-N matrix where m is the dimension}
\item \texttt{m0} \hspace{1cm} \text{Initial mean as a p-vector}
\item \texttt{C0} \hspace{1cm} \text{Initial covariance p-by-p matrix}
\item \texttt{V} \hspace{1cm} \text{Covariance m-by-p matrix V referring to observation error}
\item \texttt{W} \hspace{1cm} \text{Covariance p-by-p matrix W referring to system error}
\item \texttt{Ft} \hspace{1cm} \text{Constant observation matrix as an m-by-p matrix}
\item \texttt{Gt} \hspace{1cm} \text{Constant evolution matrix as a p-by-p matrix}
\item \texttt{a1} \hspace{1cm} \text{Drift of the observational equation}
\item \texttt{a2} \hspace{1cm} \text{Drift of the system equation}
\end{itemize}

Value

A list containing the means \( \mathbf{s} \) and the covariances \( \mathbf{S} \)

Note

Reference: Petris et al, 2009, p.53

\hline
\texttt{kd.smoother} \hspace{1cm} \textit{Kalman Smoother with Drift}
\hline

Description

Computes mean and variance of the distribution of the state, conditional on the covariances of observation and system errors, equation matrices and all of the observations.

Usage

\texttt{kd.smoother(y, f, W, Gt, a2)}

Arguments

\begin{itemize}
\item \texttt{y} \hspace{1cm} \text{Observed data as an m-by-N matrix where m is the dimension}
\item \texttt{f} \hspace{1cm} \text{Kalman filtering results as returned by \texttt{k.filter} on the data}
\item \texttt{W} \hspace{1cm} \text{Covariance p-by-p matrix W referring to system error}
\item \texttt{Gt} \hspace{1cm} \text{Evolution matrix as a p-by-p matrix}
\item \texttt{a2} \hspace{1cm} \text{Drift of the system equation}
\end{itemize}

Value

A list containing the means \( \mathbf{s} \) and the covariances \( \mathbf{S} \)

Note

Reference: Petris et al, 2009, p.61
mean.BLC

Arithmetic Mean for BLC fitted models

Description

Calculates the means based on the resulting chains from a fitted BLC model.

Usage

```r
## S3 method for class 'BLC'
mean(x, name, ...)  
```

Arguments

- `x`: A BLC object, result of a call to blc() function.
- `name`: A character with a parameter name of the BLC model that should be returned. It can be one of these: "alpha", "beta", "kappa", "phiv", "theta", "phiw".
- `...`: Further arguments passed to or from other methods.

Value

A vector with the mean values of the selected parameter.

See Also

- `mean.PredBLC()` for PredBLC object method.

Examples

```r
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

mean(fit, "kappa")
```
### mean.PredBLC

**Arithmetic Mean for Predictions**

**Description**

Calculates the means based on the resulting chains from a predicted year.

**Usage**

```r
## S3 method for class 'PredBLC'
mean(x, h, ...)   
```

**Arguments**

- `x`: A PredBLC object, result to the `pred()` function call on a BLC object.
- `h`: A positive integer specifying the year in the prediction horizon to be calculated.
- `...`: Further arguments passed to or from other methods.

**Value**

A vector with the mean values of the log-mortality chains.

**See Also**

- `mean.BLC()` for BLC object method.

**Examples**

```r
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Prediction for 2 years ahead
pred = predict(fit, h = 2)

mean(pred, 1)
mean(pred, 2)
```
plot.BLC

Plot of BLC fitted models

Description

This function plots the fitted log mortality values as well as the parameters values and credible intervals of the BLC fitted models.

Usage

```r
## S3 method for class 'BLC'
plot(x, parameter = "all", cred = 0.9, ages = NULL, ...)
```

Arguments

- `x`: A BLC or PredBLC object, result of a call to blc() function or forecast via predict() function.
- `parameter`: A character determines the parameter that will be plotted (BLC only). Default is "all" which means that all three parameters "alpha", "beta" and "kappa" will be plotted. It can also be "alpha", "beta", "kappa" or "fitted". The last one provides a plot with all the fitted tables.
- `cred`: A numeric value that indicates the probability for the credible interval. Default is 0.9.
- `ages`: A numeric vector that represents the ages used in the fitted BLC model. Default is NULL.
- `...`: Other arguments.

Value

A plot with the fitted log mortality or fitted values and credible intervals of the parameters.

See Also

plot.HP() and plot.DLM() for HP or DLM methods.

Examples

```r
## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Parameters' plot
plot(fit, parameter = "all")
plot(fit, parameter = "beta", cred = 0.95)
```
plot(fit, parameter = "alpha", ages = 18:80)
plot(fit, parameter = "kappa")

# Fitted mortality graduation
plot(fit, parameter = "fitted", ages = 18:80)

---

**plot.DLM**  

### Description

Function that returns a log-scale 
 ggplot of the DLM and ClosedDLM objects returned by dlm() and dlm_close() functions.

#### Usage

```r
## S3 method for class 'DLM'
plot(
x,
plotIC = TRUE,
plotData = TRUE,
labels = NULL,
colors = NULL,
linetype = NULL,
prob = 0.95,
age = NULL,
...
)
```

#### Arguments

- `x` Object of the class DLM or ClosedDLM returned by the dlm() or dlm_close() functions.
- `plotIC` Logical. If 'TRUE' (default), shows the predictive intervals.
- `plotData` Logical. If 'TRUE' (default), shows crude rate (black dots).
- `labels` Vector with the name of the curve label. (Optional).
- `colors` Vector with the color of the curve. (Optional).
- `linetype` Vector with the line type of the curve. (Optional).
- `prob` Coverage probability of the predictive intervals. Default is '0.95'.
- `age` Vector with the ages to plot the life table.
- `...` Other arguments.
Value

A `ggplot` object with fitted life table.

See Also

`plot.HP()`, `plot.BLC()` and `plot.PredBLC()` for HP, BLC or PredBLC methods.

`plot.list()` to the list method, adding multiple objects in one single plot.

`plot_chain()` to plot the chains generated by the MCMC algorithms for the HP and DLM objects.

Examples

```r
## Selecting the log mortality rate of the 1990 male population ranging from 0 to 100 years old
USA1990 = USA[USA$Year == 1990,]
x = 0:100
Ex = USA1990$Ex.Male[x+1]
Dx = USA1990$Dx.Male[x+1]
qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fitting DLM
fit = dlm(y, ages = 0:100, M = 100, bn = 20, thin = 1)

## Plotting the life tables:
plot(fit)

## Now we are starting from 20 years
fit2 = dlm(y[21:101], Ft = 1, Gt = 1, ages = 20:100, M = 100, bn = 20, thin = 1)

plot(fit2, plotIC = FALSE)

## To plot multiples life tables see ?plot.list
plot(list(fit, fit2), age = 20:100,
     plotData = FALSE,
     colors = c("red", "blue"),
     labels = c("1", "2"))
```

plot.HP

**HP Model: Plot the life table.**

Description

Function that returns a log-scale ggplot of HP and ClosedHP objects returned by the hp() and hp_close() functions.
### plot.HP

#### Usage

```r
## S3 method for class 'HP'
plot(
  x,
  age = NULL,
  Ex = NULL,
  plotIC = TRUE,
  plotData = TRUE,
  labels = NULL,
  colors = NULL,
  linetype = NULL,
  prob = 0.95,
  ...
)
```

#### Arguments

- `x`: Object of the class `HP` or `ClosedHP` returned by `hp()` or `hp_close()` functions.
- `age`: Vector with the ages to plot the life table.
- `Ex`: Vector with the exposures of the selected ages. Its length must be equal to the `age` vector. This argument is only necessary when using poisson and binomial `HP` models.
- `plotIC`: Logical. If `TRUE` (default), shows the predictive intervals.
- `plotData`: Logical. If `TRUE` (default), shows crude rate (black dots).
- `labels`: Vector with the name of the curve label. (Optional).
- `colors`: Vector with the color of the curve. (Optional).
- `linetype`: Vector with the line type of the curve. (Optional).
- `prob`: Coverage probability of the predictive intervals. Default is `0.95`.
- `...`: Other arguments.

#### Value

A `ggplot` object with fitted life table.

#### See Also

- `plot.DLM()`, `plot.BLC()` and `plot.PredBLC()` for DLM, BLC or PredBLC methods.
- `plot.list()` to the list method, adding multiple objects in one single plot.
- `plot_chain()` to plot the chains generated by the MCMC algorithms for the `HP` and DLM objects.

#### Examples

```r
## Selecting the exposure and the death count of the year 1990, ranging from 0 to 90 years old:
USA1990 = USA[USA$Year == 1990,]
x = 0:90
Ex = USA1990$Ex.Male[x+1]
```
Dx = USA1990$Dx.Male[x+1]

## Fitting the poisson and the lognormal model:
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "poisson",
        M = 2000, bn = 1000, thin = 1)
fit2 = hp(x = x, Ex = Ex, Dx = Dx, model = "lognormal",
        M = 2000, bn = 1000, thin = 1)

## Plot the life tables:
plot(fit)
plot(fit2, age = 0:110, plotIC = TRUE)

## To plot multiples life tables see ?plot.list
plot(list(fit, fit2),
     age = 0:110, Ex = USA1990$Ex.Male,
     plotIC = FALSE, colors = c("red", "blue"),
     labels = c("Poisson", "Lognormal"))

---

**plot.list**

Plot a set of life tables.

**Description**

Function that returns a log-scale 'ggplot' of the mortality graduation returned by hp(), dlm(), hp_close() or dlm_close() functions.

**Usage**

```r
## S3 method for class 'list'
plot(
    x,
    age = NULL,
    Ex = NULL,
    plotIC = TRUE,
    plotData = TRUE,
    labels = NULL,
    colors = NULL,
    linetype = NULL,
    prob = 0.95,
    ...
)
```

**Arguments**

- **x**
  - List of objects of the following classes: HP, DLM, ClosedHP or ClosedDLM.

- **age**
  - Vector with the ages to plot the life tables.
**Ex** Vector with the exposures of the selected ages. Its length must be equal to the age vector. This argument is only necessary when plotting poisson and binomial HP models.

**plotIC** Logical. If 'TRUE'(default), plots the predictive intervals.

**plotData** Logical. If 'TRUE' (default), plots the data used in the modelling as dots.

**labels** Description of the curve (Optional).

**colors** Vector of colours of the curves (Optional).

**linetype** Vector with the line type of the curve. (Optional).

**prob** Coverage probability of the predictive intervals. Default is '0.95'.

... Other arguments.

**Value**

A 'ggplot' object with fitted life tables.

**See Also**

*plot.DLM(), plot.HP(), plot.BLC() and plot.PredBLC() for single plots.*

**Examples**

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the 1990 male population ranging from 0 to 100 years old
USA1990 = USA[USA$Year == 1990,]
x = 0:90
Ex = USA1990$Ex.Male[x+1]
Dx = USA1990$Dx.Male[x+1]
qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fit poisson and lognormal HP model and DLM
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "poisson",
M = 2000, bn = 1000, thin = 1)
fit2 = dlm(y, M = 100, bn = 0, thin = 1)

## Plot multiples life tables
plot(list(fit, fit2),
    age = 0:110, Ex = USA1990$Ex.Male,
    plotIC = FALSE, colors = c("red", "blue"),
    labels = c("HP Poisson", "DLM"))

## Plot ClosedHP and ClosedDLM objects
close1 = hp_close(fit, method = "hp", x0 = 90)
close2 = dlm_close(fit2, method = "plateau")
```
plot(list(fit, fit2, close1, close2),
    plotIC = FALSE, colors = c("red", "blue", "green", "purple"),
    labels = c("HP", "DLM", "ClosedHP", "ClosedDLM"))

plot.PredBLC  Plot the log-mortality of a prediction

Description
This function plots the predicted log-mortality and the predict intervals of the log-mortality for a specific year in the prediction horizon.

Usage
## S3 method for class 'PredBLC'
plot(x, h = NULL, cred = 0.95, plotIC = TRUE, age = NULL, ...)

Arguments
- **x**: A PredBLC object, result to the pred() function call on a BLC object.
- **h**: A numeric vector specifying the year(s) in the prediction horizon to be calculated.
- **cred**: A real number that represents the probability of the predict interval.
- **plotIC**: Logical. If TRUE (default), shows the predictive intervals.
- **age**: A numeric vector indicating the modelled ages. (Optional).
- **...**: Other arguments.

Value
A 'ggplot' object with the predicted mortality rates and their predict intervals.

See Also
plot.HP(), plot.DLM() and plot.BLC for HP, DLM or BLC methods.

Examples
## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

# Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

# Prediction for 10 years ahead
pred = predict(fit, h = 3)
## Plotting

```r
plot(pred, h = 1)
plot(pred, h = 3, cred = 0.9)
```

### Description

This function provides three options of plots for the chain generated by the MCMC algorithm in `hp()` and `dlm()` functions.

### Usage

```r
plot_chain(fit, param, type = c("trace", "acf", "density"))
```

### Arguments

- `fit`: Object of the classes `HP` or `DLM`.
- `param`: Character vector specifying the parameters to be plotted. It is used only when the class of `fit` object is `DLM`.
- `type`: Character string specifying the type of plot to be returned. There are three options: "trace" return a plot for the sample of the parameters; "acf" return a plot for the autocorrelation of the parameters; "density" return a plot for the posterior density of the parameters based on the samples generated by the MCMC method.

### Value

A plot of the chosen type of the selected parameter(s).

### Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the 2010 total population ranging from 0 to 90 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:90
Ex = USA2010$Ex.Total[x+1]
Dx = USA2010$Dx.Total[x+1]
y = log(1 - exp(-Dx/Ex))

## Fitting HP model
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "lognormal",
         m = c(NA, 0.08, rep(NA, 6)))
```
v = c(NA, 1e-4, rep(NA, 6)))

## Plotting all the available types of plot:
plot_chain(fit, type = "trace")
plot_chain(fit, type = "acf")
plot_chain(fit, type = "density")

## Fitting DLM
fit = dlm(y, M = 100, bn = 20, thin = 1)

plot_chain(fit, param = "sigma2", type = "trace")
plot_chain(fit, param = "mu[10]", type = "acf")

## Selecting all theta1 indexed with 1 in first digit
plot_chain(fit, param = "theta1[1", type = "density")

## Plotting all parameters indexed by age 10 and age 11
plot_chain(fit, param = c("[10]", "[11]"))

---

**predict.BLC**  
*Forecast for fitted BLC models*

**Description**

Calculates the means and variances of the forecast distributions based on the resulting chains from an estimation method.

**Usage**

```r
## S3 method for class 'BLC'
predict(object, h, ...)  
```

**Arguments**

- `object`  
  A BLC object that is result of a call to blc() function.

- `h`  
  The prediction horizon.

- `...`  
  Other arguments.

**Value**

A PredBLC object that contains a list with predicted values calculated from BLC object chains structured in an array.
predict.DLM

**Description**

Extrapolates the mortality curve fitted by DLM by calculating the median of death probability and the respective prediction interval.

**Usage**

```r
## S3 method for class 'DLM'
predict(object, h, prob = 0.95, ...)
```

**Arguments**

- `object` A DLM object that is result of a call to dlm() function.
- `h` The ages prediction horizon.
- `prob` Coverage probability of the predictive intervals.
- `...` Other arguments.

**Value**

A data.frame with the death probability prediction and prediction interval for the ages in the prediction horizon.

**See Also**

- `fitted.DLM()`
Examples

## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the year 2000, ranging from 0 to 100 years old:
USA2000 = USA[USA$Year == 2000,]
x = 0:100
Ex = USA2000$Ex.Total[x+1]
Dx = USA2000$Dx.Total[x+1]

qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fitting dlm
fit = dlm(y, M = 100, bn = 20, thin = 1)

## Extrapolating the death probabilities (qx)
predict(fit, h = 3, prob = 0.95)

print.BLC

Print Values for BLC fitted models

### Description

Print details from a fitted BLC model and returns it invisibly.

### Usage

```r
## S3 method for class 'BLC'
print(x, ...)
```

### Arguments

- `x` A BLC object, result of a call to blc() function.
- `...` Further arguments passed to or from other methods.

### Value

A character vector with the details of a fitted BLC model.

### See Also

`print.DLM()`, `print.HP()` and `print.PredBLC()` for DLM, HP or PredBLC methods.
print.DLM

Print Values for DLM fitted models

Description
Print details from a fitted DLM or ClosedDLM models and returns it invisibly.

Usage
### S3 method for class 'DLM'
print(x, ...)

Arguments
- x: A DLM or ClosedDLM object, result of a call to dlm() or dlm_close() function.
- ...: Further arguments passed to or from other methods.

Value
A character vector with the details of a fitted DLM or ClosedDLM model.

See Also
print.HP(), print.BLC() and print.PredBLC() for HP, BLC or PredBLC methods.

Examples
## Importing mortality data from USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the 2010 male population ranging from 0 to 100 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:100
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]
qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fitting DLM
fit = dlm(y, M = 100, bn = 20, thin = 1)
print(fit)
print.HP

Print Values for HP fitted models

Description

Print details from a fitted HP or ClosedHP models and returns it invisibly.

Usage

## S3 method for class 'HP'
print(x, ...)

Arguments

x
A HP or ClosedHP object, result of a call to hp() or hp_close() function.

...  
Further arguments passed to or from other methods.

Value

A character vector with the details of a fitted HP or ClosedHP model.

See Also

print.DLM(), print.BLC() and print.PredBLC() for DLM, BLC or PredBLC methods.

Examples

## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and death count of the 2010 male population ranging from 0 to 90 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:90
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]

## Fitting binomial model
fit = hp(x = x, Ex = Ex, Dx = Dx, M = 5000, bn = 0, thin = 10)
print(fit)
**print.PredBLC**

*Print Values for BLC prediction models*

**Description**

Print details from a fitted BLC prediction model and returns it invisibly.

**Usage**

```r
## S3 method for class 'PredBLC'
print(x, ...)
```

**Arguments**

- `x`: A `PredBLC` object, result to the `pred()` function call on a BLC object.
- `...`: Further arguments passed to or from other methods.

**Value**

A character vector with the details of a fitted `PredBLC` model.

**See Also**

`print.DLM()`, `print.HP()` and `print.BLC()` for DLM, HP or BLC methods.

**PT**

*Mortality Data from Portugal to be used as example*

**Description**

Matrix with the logarithm of the probability of death in Portugal’s population from 2000 until 2015. The ages vary from 18 to 80 years.

**Format**

A numeric matrix with 63 rows and 16 columns:

- **Row**: Ages available.
- **Column**: Years available.

**References**

Human Mortality Database. University of California, Berkeley (USA), and Max Planck Institute for Demographic Research (Germany). Available at www.mortality.org or www.humanmortality.de (Accessed: July 9th, 2021).
Sample Quantiles for BLC fitted models

Description

Compute the quantiles based on the resulting chains from a fitted BLC model.

Usage

```r
## S3 method for class 'BLC'
quantile(x, q, name, ...)
```

Arguments

- `x`: A BLC object, result of a call to `blc()` function.
- `q`: A real number that represents the probability of the quantiles.
- `name`: A character with a parameter name of the blc model that should be returned. It can be one of these: "alpha", "beta", "kappa", "phiv", "theta", "phiw".
- `...`: Further arguments passed to or from other methods.

Value

A data.frame with the quantiles of the selected parameter.

See Also

`quantile.PredBLC()` for PredBLC method.

Examples

```r
## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Parameters' median and quantiles 0.05, 0.95
quantile(fit, c(0.05, 0.5, 0.95), "alpha")
quantile(fit, c(0.05, 0.5, 0.95), "beta")
quantile(fit, c(0.05, 0.5, 0.95), "kappa")
quantile(fit, c(0.05, 0.5, 0.95), "phiv") ## random error precision
quantile(fit, c(0.05, 0.5, 0.95), "theta") ## drift parameter
quantile(fit, c(0.05, 0.5, 0.95), "phiw")
```
Sample Quantiles for Predictions

Description
Calculates the quantiles of log-mortality based on the resulting chains from a predicted year.

Usage
```r
## S3 method for class 'PredBLC'
quantile(x, q, h, ...)
```

Arguments
- `x`: A `PredBLC` object, result to the `pred()` function call on a `BLC` object.
- `q`: A real number that represents the probability of the quantiles.
- `h`: A positive integer specifying the year in the prediction horizon to be calculated.
- `...`: Further arguments passed to or from other methods.

Value
A data.frame with the quantiles of the selected parameter.

See Also
- `quantile.BLC()` for `BLC` method.

Examples
```r
## Importing log-mortality data from Portugal:
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, numit = 100, warmup = 20)

## Prediction for 2 years ahead
pred = predict(fit, h = 2)

## The log-mortality median for the first year of prediction
quantile(pred, q = 0.5, h = 1)

## The 0.1 and 0.9 quantiles for the first and second year of prediction
quantile(pred, q = c(0.1, 0.9), h = 1)
quantile(pred, q = c(0.1, 0.9), h = 2)
```
qx_ci

Predictive intervals for deaths probabilities.

Description
This function returns the predictive intervals for death probabilities of a mortality graduation provide by hp(), dlm(), hp_close() or dlm_close() functions.

Usage

\[
\text{qx_ci}(\text{fit}, \text{age} = \text{NULL}, \text{Ex} = \text{NULL}, \text{prob} = 0.95)
\]

Arguments

- **fit**: Object of the following classes: 'HP', 'DLM', 'ClosedHP', or 'ClosedDLM'.
- **age**: Vector with the ages to compute the predictive intervals.
- **Ex**: Vector with the exposures of the selected ages. Its length must be equal to the age vector. This argument is only necessary when using the Poisson and the Binomial distributions.
- **prob**: Coverage probability of the predictive intervals.

Value
A data.frame object with the selected ages and the respective predictive intervals of the death probabilities.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and the death count of the year 2000, ranging from 0 to 90 years old:
USA2000 = USA[USA$Year == 2000,]
x = 0:90
Ex = USA2000$Ex.Total[x+1]
Dx = USA2000$Dx.Total[x+1]

## Fitting poisson HP model and DLM:
fit = hp(x = x, Ex = Ex, Dx = Dx, model = "poisson",
M = 1000, bn = 0, thin = 10)

## predictive intervals
qx_ci(fit)
qx_ci(fit, age = 0:110, Ex = USA2000$Ex.Total, prob=0.99)
```
**Summary for DLM fitted models**

**Description**

Summarizes information from the parameters' markov chains of a fitted DLM or ClosedDLM model.

**Usage**

```r
## S3 method for class 'DLM'
summary(object, digits = 5, ...)
```

**Arguments**

- `object` A DLM or ClosedDLM object, result of a call to dlm() or dlm_close() function.
- `digits` An integer indicating the number of decimals places.
- `...` Further arguments passed to or from other methods.

**Value**

A data.frame object with the mean, standard deviation and 2.5%, 50% and 97.5% quantiles of a fitted DLM or ClosedDLM model.

**See Also**

`summary.HP()` for HP method.

**Examples**

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the log mortality rate of the 2010 male population ranging from 0 to 100 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:100
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]
qx_t = Dx/Ex
qx_t = 1 - exp(-qx_t)
y = log(qx_t)

## Fitting DLM
fit = dlm(y, M = 100, bn = 20, thin = 1)
summary(fit)
```
Summary for HP fitted models

Description

Summarizes information from the parameters' Markov chains of a fitted HP or ClosedHP model.

Usage

```r
## S3 method for class 'HP'
summary(object, ...)
```

Arguments

- `object` A HP or ClosedHP object, result of a call to `hp()` or `hp_close()` function.
- `...` Further arguments passed to or from other methods.

Value

A data.frame object with the mean, standard deviation and 2.5%, 50% and 97.5% quantiles of a fitted HP or ClosedHP model.

See Also

`summary.DLM()` for DLM method.

Examples

```r
## Importing mortality data from the USA available on the Human Mortality Database (HMD):
data(USA)

## Selecting the exposure and death count of the 2010 male population ranging from 0 to 90 years old
USA2010 = USA[USA$Year == 2010,]
x = 0:90
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]

## Fitting binomial model
fit = hp(x = x, Ex = Ex, Dx = Dx, M = 5000, bn = 0, thin = 10)
summary(fit)
```
Description

Data base with exposures and death counts of the population of the United States between the years 1933 and 2019.

Format

A data.frame with 9657 rows and 8 variables:

- **Year**  Years available.
- **Age**   Ages available.
- **Ex.Total** Exposure of the US population.
- **Dx.Total** Death count of the US population.
- **Ex.Male** Exposure of the US male population.
- **Dx.Male** Death count of the US male population.
- **Ex.Female** Exposure of the US female population.
- **Dx.Female** Death count of the US female population.

References

Human Mortality Database. University of California, Berkeley (USA), and Max Planck Institute for Demographic Research (Germany). Available at www.mortality.org or www.humanmortality.de (Accessed: August 9th, 2021).
**Index**

* data
  PT, 47
  USA, 53

blc, 3

dlm, 5
  dlm_close, 8
  dlm_close(), 7

expectancy, 10
  expectancy(), 4, 7, 9, 25, 27
  expectancy.BLC, 11
  expectancy.BLC(), 4, 10, 12, 14, 43
  expectancy.DLM, 12
  expectancy.DLM(), 7, 9–11, 14
  expectancy.HP, 13
  expectancy.HP(), 10–12, 25, 27

fitted(), 4, 7, 9, 25, 27, 43
  fitted.BLC, 15
  fitted.BLC(), 4, 16, 17, 43
  fitted.DLM, 16
  fitted.DLM(), 7, 9, 15, 17, 43
  fitted.HP, 17
  fitted.HP(), 15, 16, 25, 27

Heatmap, 18
  Heatmap(), 4, 7, 9, 25, 27
  Heatmap.BLC, 18, 43
  Heatmap.BLC(), 4, 11, 18, 20–22
  Heatmap.DLM, 19
  Heatmap.DLM(), 7, 9, 12, 18, 19, 21, 22
  Heatmap.HP, 20
  Heatmap.HP(), 14, 18–20, 22, 25, 27
  Heatmap.list, 22
  Heatmap.list(), 12, 14, 18, 20, 21

hp, 23
  hp_close, 26
  hp_close(), 25
  hp_mix, 28
  hp_mix(), 25

improvement, 30
  improvement(), 4

kd.filter, 30
  kd.smooother, 31

mean.BLC, 32
  mean.BLC(), 33
  mean.PredBLC, 33
  mean.PredBLC(), 32
  plot(), 4, 7, 9, 25, 27, 43
  plot.BLC, 34, 40
  plot.BLC(), 4, 36, 37, 39, 43
  plot.DLM, 35
  plot.DLM(), 7, 9, 34, 37, 39, 40
  plot.HP, 36
  plot.HP(), 25, 27, 34, 36, 39, 40
  plot.list, 38
  plot.list(), 36, 37
  plot.PredBLC, 40
  plot.PredBLC(), 36, 37, 39
  plot_chain, 41
  plot_chain(), 7, 25, 36, 37
  predict(), 4
  predict.BLC, 42
  predict.BLC(), 4
  predict.DLM, 43
  predict.DLM(), 7
  print(), 4, 7, 9, 25, 27, 43
  print.BLC, 44
  print.BLC(), 4, 43, 45–47
  print.DLM, 45
  print.DLM(), 7, 9, 44, 46, 47
  print.HP, 46
  print.HP(), 25, 27, 44, 45, 47
  print.PredBLC, 47
  print.PredBLC(), 44–46
INDEX

PT, 47
quantile.BLC, 48
quantile.BLC(), 49
quantile.PredBLC, 49
quantile.PredBLC(), 48
qx_ci, 50
qx_ci(), 7, 9, 25, 27
summary(), 7, 9, 25, 27
summary.DLM, 51
summary.DLM(), 7, 9, 52
summary.HP, 52
summary.HP(), 25, 27, 51
USA, 53