

# Package ‘BayesMortalityPlus’

August 2, 2025

**Type** Package

**Title** Bayesian Mortality Modelling

**Version** 1.0.0

**Description** Fit Bayesian graduation mortality using the Heligman-Pollard model, as seen in Heligman, L., & Pollard, J. H. (1980) <[doi:10.1017/S0020268100040257](https://doi.org/10.1017/S0020268100040257)> and Dellaportas, Petros, et al. (2001) <[doi:10.1111/1467-985X.00202](https://doi.org/10.1111/1467-985X.00202)>, and dynamic linear model (Campagnoli, P., Petris, G., and Petrone, S. (2009) <[doi:10.1007/b135794\\_2](https://doi.org/10.1007/b135794_2)>). While Heligman-Pollard has parameters with a straightforward interpretation yielding some rich analysis, the dynamic linear model provides a very flexible adjustment of the mortality curves by controlling the discount factor value. Closing methods for both Heligman-Pollard and dynamic linear model were also implemented according to Dodd, Erenkul, et al. (2018) <<https://www.jstor.org/stable/48547511>>. The Bayesian Lee-Carter model is also implemented to fit historical mortality tables time series to predict the mortality in the following years and to do improvement analysis, as seen in Lee, R. D., & Carter, L. R. (1992) <[doi:10.1080/01621459.1992.10475265](https://doi.org/10.1080/01621459.1992.10475265)> and Pedroza, C. (2006) <[doi:10.1093/biostatistics/kxj024](https://doi.org/10.1093/biostatistics/kxj024)>. Journal publication available at <[doi:10.18637/jss.v113.i09](https://doi.org/10.18637/jss.v113.i09)>.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.5.0)

**Imports** MASS, progress, tidyr, scales, utils, ggplot2, magrittr, dplyr, mvtnorm

**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' 'expectancy\_dlm.R' 'fun\_aux.R' 'fitted\_hp.R' 'expectancy\_hp.R' 'fitted\_blc.R' 'globals.R' 'heatmap\_blc.R' 'heatmap\_dlm.R' 'heatmap\_hp.R' 'heatmap\_list.R' 'hp\_poisson\_reduced.R' 'hp\_lognormal\_reduced.R' 'hp\_binomial\_reduced.R' 'hp\_poisson.R' 'hp\_lognormal.R' '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

**Author** Luiz F. V. Figueiredo [aut, cre] (ORCID:

<<https://orcid.org/0009-0007-9002-1997>>),

Lucas M. F. Silva [aut] (ORCID:

<<https://orcid.org/0009-0000-7452-3486>>),

Viviana G. R. Lobo [aut] (ORCID:

<<https://orcid.org/0000-0002-4076-8327>>),

Thais C. O. Fonseca [aut] (ORCID:

<<https://orcid.org/0000-0002-4943-3259>>),

Mariane B. Alves [aut] (ORCID: <<https://orcid.org/0000-0002-2489-9300>>)

**Maintainer** Luiz F. V. Figueiredo <labmapackage@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-07-10 21:40:17 UTC

## Contents

blc . . . . .	3
dlm . . . . .	5
dlm_close . . . . .	8
expectancy . . . . .	10
expectancy.BLC . . . . .	11
expectancy.DLM . . . . .	12
expectancy.HP . . . . .	14
fitted.BLC . . . . .	15
fitted.DLM . . . . .	16
fitted.HP . . . . .	17
Heatmap . . . . .	18
Heatmap.BLC . . . . .	19
Heatmap.DLM . . . . .	20
Heatmap.HP . . . . .	21
Heatmap.list . . . . .	22
hp . . . . .	24
hp_close . . . . .	27
hp_mix . . . . .	29
improvement . . . . .	31
mean.BLC . . . . .	31
mean.PredBLC . . . . .	32
plot.BLC . . . . .	33
plot.DLM . . . . .	34
plot.HP . . . . .	36
plot.list . . . . .	37

plot.PredBLC . . . . .	39
plot_chain . . . . .	40
predict.BLC . . . . .	41
predict.DLM . . . . .	42
print.BLC . . . . .	43
print.DLM . . . . .	44
print.HP . . . . .	45
PT . . . . .	46
quantile.BLC . . . . .	46
quantile.PredBLC . . . . .	47
summary.DLM . . . . .	48
summary.HP . . . . .	49
USA . . . . .	50

## Index 51

blc *Lee-Carter Bayesian Estimation for mortality data*

### Description

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

### Usage

```
blc(Y, prior = NULL, init = NULL, M = 5000, bn = 4000, thin = 1)
```

### 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 $\theta$ .
M	The number of iterations. The default value is 5000.
bn	The number of initial iterations from the Gibbs sampler that should be discarded (burn-in). The default value is 4000.
thin	A Positive integer specifying the period for saving samples. The default value is 1.

### Details

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

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

where  $\alpha = (\alpha_1, \dots, \alpha_p)'$  are the intercept of the model that represent the log-mortality rate mean in each age;  $\beta = (\beta_1, \dots, \beta_p)'$  are the coefficient regression that represent the speed of relative change in the log-mortality rate in each age.  $\kappa = (\kappa_1, \dots, \kappa_T)'$  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.
bn	The warmup of the algorithm specified by the user to fit the model.
M	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

- Lee, R. D., & Carter, L. R. (1992). "Modeling and forecasting US mortality." *Journal of the American statistical association*, 87(419), 659-671.
- Pedroza, C. (2006). "A Bayesian forecasting model: predicting US male mortality." *Biostatistics*, 7(4), 530-550.
- Silva, L. M. F., Figueiredo, L. F. V., Lobo, V. G. R., Fonseca, T. C. O., & Alves, M. B. (2025). *Journal of Statistical Software*. BayesMortalityPlus: A Package in R for Bayesian Mortality Modeling. doi:10.18637/jss.v113.i09

### 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(mx = USA$Dx.Total/USA$Ex.Total) -> data

data %>%
  filter(Age %in% 18:80 & Year %in% 2000:2015) %>%
  mutate(logmx = log(mx)) %>%
  dplyr::select(Year, Age, logmx) %>%
  pivot_wider(names_from = Year, values_from = logmx) %>%
  dplyr::select(-Age) %>%
  as.matrix() %>%
  magrittr::set_rownames(18:80) -> Y

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 20)
print(fit)

## Viewing the results
plot(fit, ages = 18:80)
plot(fit, parameter = "beta", ages=18:80)
improvement(fit)
```

---

d1m

*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 to compute the posterior distribution. The response variable is the log of the mortality rate, and it is modeled specifying the matrices Ft and Gt 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

```
d1m(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, b = 0), M = 2000, 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, 'Ft = matrix(c(1,0), nrow = 1)'
Gt	pxp Matrix that specifies the system equations. By default, Gt = matrix(c(1,0,1,1), 2).
delta	Positive real value or real vector of the same length as y with values in the '(0, 1)' interval specifying the discount factor for each age. A higher value of delta results in a higher smoothness of the fitted curve. If a single value is defined, this same value is used for all ages. By default, delta is '0.85'.
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 that indicates the sampling size from the posterior distributions. The default value is 2000.
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  $v_t \sim N(0, \sigma^2)$  and  $w_t \sim N(0, \sigma^2 W_t)$ . We use the discount factors  $\delta$  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).  $\delta$  can be specified as a single value for all ages or as a vector in which each element is associated with an age.

A scheme described by (Petris et al, 2009) for conjugated inference is used. 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.
sig2	Posterior samples from $\sigma^2$ .
param	A list with the states parameters for filtering distribution (mt, Ct), predictive distribution (ft, Qt), smoothing distribution (as, Rs), and parameters of the posterior distribution for variance (alpha, beta).
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 <i>delta</i> value specified and priors informations.

## References

- Campagnoli, P., Petris, G., and Petrone, S. (2009). *Dynamic linear models with R*. Springer-Verlag New York.
- Silva, L. M. F., Figueiredo, L. F. V., Lobo, V. G. R., Fonseca, T. C. O., & Alves, M. B. (2025). *Journal of Statistical Software*. BayesMortalityPlus: A Package in R for Bayesian Mortality Modeling. doi:10.18637/jss.v113.i09

## 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\(\)](#).

[expectancy.DLM\(\)](#) and [Heatmap.DLM\(\)](#) for DLM methods to compute and visualise the truncated life expectancy via [expectancy\(\)](#) and [Heatmap\(\)](#) functions.

[dml\\_close\(\)](#) for close methods to expand the life tables.

[plot\\_chain\(\)](#) to 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]
y = log(Dx/Ex)

## Fitting DLM
fit = dlm(y)
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]"))

## Varying discount factor
fit2 = dlm(y, delta = c(rep(0.8, 36), rep(0.9, 65)))
plot(fit2)
```

---

dml\_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

```
dml_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

<code>fit</code>	Object of the class DLM adjusted by the <code>dlm()</code> function.
<code>method</code>	Character string specifying the closing method to be fitted, with them being: 'plateau', 'linear' or 'gompertz'.
<code>x0</code>	Integer with the starting age the closing method will be fitted from. Default is the last age fitted by the 'DLM' object.
<code>max_age</code>	Integer with the maximum age the closing method will be fitted. Default age is '120'.
<code>k</code>	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.
<code>weights</code>	Vector of weights of the closing method used in the mixture of the closing method and the fitted model made in the mixing age group. The vector's size should be equal to $2k+1$ . For a better understanding of this parameter and the mixture applied in this function, see Details.
<code>new_data</code>	Vector containing the log mortality rates of ages after the <code>x0</code> input. This is an optional argument used in the 'linear' and 'Gompertz' closing methods.



## Details

# There are three types of age groups when the closing method is applied: a group where only the fitted model (DLM) computes the death probabilities, followed by a group in which the death probabilities are a mix (or more precise a weighted mean) from the HP model and the closing method and followed by a group in which the death probabilities are computed just by the closing method. The mix is applied so the transition of the death probabilities of the ages between the fitted model and the closing method occurs smoothly.

The parameters 'x0' and 'k' define the mixing group age. The parameter 'x0' indicates the center age of the group. The parameter 'k' is the range of ages before 'x0' and after 'x0', so this group has a total of  $2k + 1$  age. Therefore, the parameter 'weights' must have a length size equal to  $2k + 1$ . In this case, the death probability is calculated as follows. Consider  $model_x$  and  $close_x$  as the death probability of the fitted model and closing method in the age  $x$ , respectively. Then, the resulting death probability of the mix is calculated as:

$$q_x = w_x model_x + (1 - w_x) close_x,$$

where  $w_x$  represents the weight of the closing method in the age  $x$ . This procedure is applied only in the linear and Gompertz methods.

The three closing methods implemented by the function are:

- 1.'linear' method: Fits a linear regression starting at age  $x_0 - k$  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 ( $q_x$ ) constant after the  $x_0$  argument.

## Value

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

## References

Dodd, Erenkul, Forster, Jonathan, Bijak, Jakub, & Smith, Peter 2018. "Smoothing mortality data: the English life table, 2010-12." *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 181(3), 717-735.

## 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.

## 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$Year == 2010,]
x = 0:100
Ex = USA2010$Ex.Male[x+1]
Dx = USA2010$Dx.Male[x+1]
y <- log(Dx/Ex)

fit <- dlm(y, M = 100)

## 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]
y2 <- log(Dx2/Ex2)

close2 = dlm_close(fit, method = "linear",
                  new_data = y2)

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

## 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)

```

---

expectancy

*Generic expectancy function*

---

### Description

Generic function to expectancy method.

### Usage

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:

$$t p_x = p_0 x p_1 x \dots x p_x$$

**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	<i>BLC: Life expectancy</i>
----------------	-----------------------------

---

**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**

```
## S3 method for class 'BLC'
expectancy(x, at = NULL, prob = 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.
- prob                  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**

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

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 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))
```

---

expectancy.DLM

*DLM: Life expectancy*

---

**Description**

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

**Usage**

```
## 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 or 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**

```
## 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]
y <- log(Dx/Ex)

fit <- dlm(y, M = 100)
expectancy(fit)

# Example 2: -----

# Using some arguments:

expectancy(fit, age = c(0,20,30,60),
prob = 0.99, max_age = 90, graph = FALSE)
```

---

 expectancy.HP

*HP: Life expectancy*


---

### Description

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

### Usage

```
## 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 expectancy (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)
```

---

fitted.BLC

*BLC: Fitted death probabilities (qx)*


---

**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, prob = 0.95, ...)
```

**Arguments**

object	A BLC or PredBLC object, result of a call to blc() or predict() function.
prob	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, M = 100, bn = 20)

## Log-mortalities estimates for each age and year in model fitted
fitted(fit, prob = 0.95)
```

---

fitted.DLM

*DLM: Fitted death probabilities (qx)*

---

**Description**

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

**Usage**

```
## S3 method for class 'DLM'
fitted(object, age = NULL, prob = 0.95, ...)
```

**Arguments**

<code>object</code>	Object of the following classes: DLM or ClosedDLM.
<code>age</code>	Vector with the ages to calculate the death probabilities (Optional). By default, all ages are considered.
<code>prob</code>	Coverage probability of the predictive intervals.
<code>...</code>	Other arguments.

**Value**

A data.frame object with the selected ages and the corresponding estimates and predictive intervals of the death probabilities.



**See Also**

[fitted.HP\(\)](#) and [fitted.BLC\(\)](#) for HP or BLC methods.

**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]
y = log(Dx/Ex)

## Fitting dlm
fit = dlm(y, M = 100)

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

---

fitted.HP

*HP: Fitted death probabilities (qx)*

---

**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**

```
## S3 method for class 'HP'
fitted(object, age = NULL, Ex = NULL, prob = 0.95, ...)
```

**Arguments**

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

**Value**

A data.frame object with the selected ages and the corresponding estimates and predictive intervals of the death probabilities.

**See Also**

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

**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 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, Ex = USA2000$Ex.Total[0:110+1])
```

---

Heatmap

*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\(\)](#).

---

Heatmap.BLC

*BLC: Heatmap for the life expectancy*


---

**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, M = 100, bn = 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 = NULL,
  age = NULL,
  max_age = 110,
  color = c("red", "white", "blue"),
  ...
)
```

**Arguments**

<code>x</code>	Object or a list of objects of the class DLM or ClosedDLM returned by <code>dlm()</code> or <code>dml_close()</code> functions.
<code>x_lab</code>	Description of the object 'fit'.
<code>age</code>	Vector with the ages to plot the heatmap.
<code>max_age</code>	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.
<code>color</code>	Vector of colours used in the heatmap.
<code>...</code>	Further arguments passed to or from other methods.

**Value**

A ggplot2 heatmap of the life expectancy.

**See Also**

[Heatmap.BLC\(\)](#) and [Heatmap.HP\(\)](#) for BLC or HP methods.

[Heatmap.list\(\)](#) to the `list` method, adding multiple objects in one single Heatmap.

**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]
yF = log(DxF/ExF)

fitF <- dlm(yF, M = 100)

Heatmap(fitF, x_lab = "Female expec. 2010 USA", max_age = 90)
```

Heatmap.HP

*HP: Heatmap for the life expectancy***Description**

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

**Usage**

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

**Arguments**

<code>x</code>	Object or a list of objects of the class HP or ClosedHP returned by <code>hp()</code> or <code>close_hp()</code> functions.
<code>x_lab</code>	Description of the object 'fit'.
<code>age</code>	Vector with the ages to plot the heatmap.
<code>max_age</code>	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.
<code>color</code>	Vector of colours used in the heatmap.
<code>...</code>	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**

```
## 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 a set of life tables*

---

**Description**

This function plots a heatmap for the life expectancy of the mortality graduations returned by `hp()`, `d1m()`, `hp_close()` or `d1m_close()` functions.

**Usage**

```
## S3 method for class 'list'
Heatmap(
  x,
  x_lab = NULL,
  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 Heligman-Pollard curve for mortality table graduation*

---

### 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

```
hp(x, Ex, Dx, model = c("binomial", "lognormal", "poisson"),
  M = NULL, bn = NULL, 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. The default value is 50000 iterations. For the reduced model, the default is 30000 iterations.
bn	Non-negative integer indicating the number of iteration to be discarded as the burn-in period. The default value is half of the M value, rounded.
thin	Positive integer specifying the period for saving samples. The default value is 10.
m	Numeric vector with the mean of the prior distributions for (A, B, C, D, E, F, G, H).
v	Numeric vector with the variance of the prior distributions for (A, B, C, D, E, F, G, H).
inits	Numeric vector with the initial values for the parameters (A, B, C, D, E, F, G, H).
K	Number that specifies the extra parameter 'K' value for binomial and poisson models. It is considered only if model = "binomial" or model = "poisson". The default value is the optimal value.
sigma2	Positive number that specifies initial value of sigma2 in lognormal model. It is considered only if model = "lognormal".
prop.control	Positive number that is considered for tuning the acceptance rate of MCMC.
reduced_model	Logical value which determines if reduced model should be addressed. If 'TRUE' (default), the first term of the HP curve (infant mortality) is not considered.



## Details

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

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

$$q_x = 1 - e^{(-HP_x)}$$

This approximation ensures that  $q_x$ , 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:

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

where  $\epsilon_x$  has independent distributions  $\text{Normal}(0, \text{sigma}2)$  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 the estimated optimal value, are estimated by the MCMC methods. Gibbs sampling for  $\text{sigma}2$  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.
data	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

- Dellaportas, P., Smith, A. F., and Stavropoulos, P. (2001). "Bayesian Analysis of Mortality Data." *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 164 (2). Wiley Online Library: 275–291.
- Heligman, Larry, and John H Pollard. (1980). "The Age Pattern of Mortality." *Journal of the Institute of Actuaries (1886-1994)* 107 (1). JSTOR: 49–80.
- Silva, L. M. F., Figueiredo, L. F. V., Lobo, V. G. R., Fonseca, T. C. O., & Alves, M. B. (2025). *Journal of Statistical Software*. BayesMortalityPlus: A Package in R for Bayesian Mortality Modeling. doi:10.18637/jss.v113.i09

**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.

[plot\\_chain\(\)](#) to visualise the markov chains, respectively.

**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)
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)
```

**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**

```
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 of the closing method used in the mixture of the closing method and the fitted model made in the mixing age group. The vector's size should be equal to 2k+1. For a better understanding of this parameter and the mixture applied in this function, see Details.
new_Ex	Vector with exposure of ages after the x0 input. This is an optional argument used in the 'linear' and 'gompertz' closing methods. If this argument is specified, then new_Dx also needs to be.
new_Dx	Vector containing the death counts of the ages after the x0 input. This is also an optional argument used in the 'linear' and 'gompertz' closing methods. The length must be the same as new_Ex.

**Details**

There are three types of age groups when the closing method is applied: a group where only the HP-fitted model computes the death probabilities, followed by a group in which the death probabilities are a mix (or more precise a weighted mean) from the HP model and the closing method and

followed by a group in which the death probabilities are computed just by the closing method. The mix is applied so the transition of the death probabilities of the ages between the fitted model and the closing method occurs smoothly.

The parameters 'x0' and 'k' define the mixing group age. The parameter 'x0' indicates the center age of the group. The parameter 'k' is the range of ages before 'x0' and after 'x0', so this group has a total of  $2k + 1$  age. Therefore, the parameter 'weights' must have a length size equal to  $2k + 1$ . In this case, the death probability is calculated as follows. Consider  $model_x$  and  $close_x$  as the death probability of the fitted model and closing method in the age  $x$ , respectively. Then, the resulting death probability of the mix is calculated as:

$$q_x = w_x model_x + (1 - w_x) close_x,$$

where  $w_x$  represents the weight of the closing method in the age  $x$ . This computation is applied to all elements in the MCMC chain of the fitted model, resulting in a new chain of death probabilities. This procedure is applied only in the linear and Gompertz methods.

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 x0 argument.
- 3.'linear' method: Fits a linear regression starting at age x0 - 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

Dodd, Erengul, Forster, Jonathan, Bijak, Jakub, & Smith, Peter 2018. "Smoothing mortality data: the English life table, 2010-12." *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 181(3), 717-735.

## 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.

## 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:

## 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

```

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

```

**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**

```
## 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	<i>BLC: Improvement</i>
-------------	-------------------------

---

**Description**

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

**Usage**

```
improvement(obj, prob = 0.95)
```

**Arguments**

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

**Value**

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

**Examples**

```
## Importing log-mortality data from Portugal:  
data(PT)  
Y <- PT  
  
## Fitting the model  
fit = blc(Y = Y, M = 100, bn = 20)  
  
## Improvement:  
improvement(fit)  
improvement(fit, prob = 0.9) #90% credible intervals
```

---

mean.BLC	<i>BLC: Arithmetic mean</i>
----------	-----------------------------

---

**Description**

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

**Usage**

```
## 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**

```
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 20)

mean(fit, "kappa")
```

---

mean.PredBLC

*BLC: Arithmetic mean for predictions*

---

**Description**

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

**Usage**

```
## 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**

```
data(PT)
Y <- PT

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 20)

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

mean(pred, 1)
mean(pred, 2)
```

---

plot.BLC

*BLC: Plot the fitted values*


---

**Description**

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

**Usage**

```
## S3 method for class 'BLC'
plot(x, parameter = "all", prob = 0.9, age = NULL, year = NULL, ...)
```

**Arguments**

x	A BLC object, result of a call to blc() function.
parameter	A character determines the parameter that will be plotted. 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.
prob	A numeric value that indicates the probability for the credible interval. Default is '0.9'.
age	A numeric vector that represents the ages used in the fitted BLC model. Default is 'NULL'.
year	A numeric vector that represents the years 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**

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

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 20)

## Parameters' plot
plot(fit, parameter = "all")
plot(fit, parameter = "beta", prob = 0.95)
plot(fit, parameter = "alpha", age = 18:80)
plot(fit, parameter = "kappa")

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

---

plot.DLM

*DLM: Plot the life table*

---

**Description**

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

**Usage**

```
## 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 <code>dlm()</code> or <code>dlm_close()</code> 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**

```
## 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]
y = log(Dx/Ex)

## Fitting DLM
fit = dlm(y, ages = 0:100, M = 100)

## 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)

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: 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.

### Usage

```
## 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**

```
## 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()`, `dln()`, `hp_close()` or `dln_close()` functions.

**Usage**

```
## 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

```

## 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]
y = log(Dx/Ex)

## 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)

## 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

*BLC: Plot the log-mortality of a prediction*


---

## Description

This functions plot 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, prob = 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.
prob	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, M = 100, bn = 20)

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

## Plotting
plot(pred, h = 1)
plot(pred, h = 3, prob = 0.9)
```

---

plot\_chain

*Chain's plot*


---

**Description**

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

**Usage**

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

**Arguments**

<code>fit</code>	Object of the classes HP or DLM.
<code>param</code>	Character vector specifying the parameters to be plotted. It is used only when the class of fit object is DLM.
<code>type</code>	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**

```
## 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(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)

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

*BLC: Forecasting*

---

**Description**

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

**Usage**

```
## 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.

**See Also**

[fitted.BLC\(\)](#), [print.BLC\(\)](#), and [plot.PredBLC\(\)](#) for PredBLC methods to native R functions [fitted\(\)](#), [print\(\)](#), and [plot\(\)](#).

[expectancy.BLC\(\)](#) and [Heatmap.BLC](#) to compute and plot the life expectancy of the prediction(s).

**Examples**

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

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 20)

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

---

predict.DLM

*DLM: Prediction of death probability*

---

**Description**

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

**Usage**

```
## 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]

y = log(Dx/Ex)

## Fitting dlm
fit = dlm(y, M = 100)

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

---

print.BLC

*BLC: Print*


---

**Description**

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

**Usage**

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

**Arguments**

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

**Value**

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

**See Also**

[print.DLM\(\)](#) and [print.HP\(\)](#) for DLM or HP methods.

---

print.DLM

*DLM: Print*

---

**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\(\)](#) and [print.BLC\(\)](#) for HP or BLC methods.

**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]
y = log(Dx/Ex)
```

```
## Fitting DLM
fit = dlm(y, M = 100)
print(fit)
```

---

print.HP

*HP: Print*


---

### 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\(\)](#) and [print.BLC\(\)](#) for DLM or BLC 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)
```

PT

*Mortality Data from Portugal to be used as example***Description**

Matrix with the logarithm of the mortality rates 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](http://www.mortality.org) or [www.humanmortality.de](http://www.humanmortality.de) (Accessed: July 9th, 2021).

quantile.BLC

*BLC: Sample quantiles***Description**

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

**Usage**

```
## 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**

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

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 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")
```

---

quantile.PredBLC

*BLC: Sample quantiles for predictions*

---

**Description**

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

**Usage**

```
## 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**

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

## Fitting the model
fit = blc(Y = Y, M = 100, bn = 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)
```

summary.DLM

*DLM: Summary***Description**

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

**Usage**

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

**Arguments**

object	A DLM or ClosedDLM object, result of a call to <code>d1m()</code> or <code>d1m_close()</code> 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**

```
## 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]
y = log(Dx/Ex)

## Fitting DLM
fit = dlm(y, M = 100)
summary(fit)
```

summary.HP

*HP: Summary***Description**

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

**Usage**

```
## 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

```
## 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)
```

---

USA

*Mortality Database from United States to be used as example*

---

## 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](http://www.mortality.org) or [www.humanmortality.de](http://www.humanmortality.de) (Accessed: August 9th, 2021).

# Index

## \* data

PT, 46  
USA, 50

b1c, 3

d1m, 5  
d1m\_close, 8  
d1m\_close(), 7

expectancy, 10  
expectancy(), 4, 7, 9, 26, 28  
expectancy.B1C, 11  
expectancy.B1C(), 4, 11, 13, 14, 42  
expectancy.D1M, 12  
expectancy.D1M(), 7, 9, 11, 12, 14  
expectancy.HP, 14  
expectancy.HP(), 11–13, 26, 28

fitted(), 4, 7, 9, 26, 28, 42  
fitted.B1C, 15  
fitted.B1C(), 4, 17, 18, 42  
fitted.D1M, 16  
fitted.D1M(), 7, 9, 16, 18, 43  
fitted.HP, 17  
fitted.HP(), 16, 17, 26, 28

Heatmap, 18  
Heatmap(), 4, 7, 9, 26, 28  
Heatmap.B1C, 19, 42  
Heatmap.B1C(), 4, 12, 19, 20, 22, 23  
Heatmap.D1M, 20  
Heatmap.D1M(), 7, 9, 13, 19, 22, 23  
Heatmap.HP, 21  
Heatmap.HP(), 14, 19, 20, 23, 26, 28  
Heatmap.list, 22  
Heatmap.list(), 13, 14, 19, 20, 22  
hp, 24  
hp\_close, 27  
hp\_close(), 26  
hp\_mix, 29

hp\_mix(), 26

improvement, 31  
improvement(), 4

mean.B1C, 31  
mean.B1C(), 33  
mean.PredB1C, 32  
mean.PredB1C(), 32

plot(), 4, 7, 9, 26, 28, 42  
plot.B1C, 33, 39  
plot.B1C(), 4, 35, 37, 38  
plot.D1M, 34  
plot.D1M(), 7, 9, 34, 37–39  
plot.HP, 36  
plot.HP(), 26, 28, 34, 35, 38, 39  
plot.list, 37  
plot.list(), 35, 37  
plot.PredB1C, 39  
plot.PredB1C(), 35, 37, 38, 42  
plot\_chain, 40  
plot\_chain(), 7, 26, 35, 37  
predict(), 4  
predict.B1C, 41  
predict.B1C(), 4  
predict.D1M, 42  
predict.D1M(), 7  
print(), 4, 7, 9, 26, 28, 42  
print.B1C, 43  
print.B1C(), 4, 42, 44, 45  
print.D1M, 44  
print.D1M(), 7, 9, 44, 45  
print.HP, 45  
print.HP(), 26, 28, 44  
PT, 46

quantile.B1C, 46  
quantile.B1C(), 47  
quantile.PredB1C, 47

`quantile.PredBLC()`, 47

`summary()`, 7, 9, 26, 28

`summary.DLM`, 48

`summary.DLM()`, 7, 9, 49

`summary.HP`, 49

`summary.HP()`, 26, 28, 48

USA, 50