Package ‘ilc’

February 20, 2015

Type Package

Title Lee-Carter Mortality Models using Iterative Fitting Algorithms

Version 1.0

Date 2014-11-19

Depends R (>= 3.0.3), demography, rainbow, forecast, date

Imports survival

Author Zoltan Butt, Steven Haberman and Han Lin Shang

Maintainer Zoltan Butt <Z.Butt@city.ac.uk>

Description Fitting a class of Lee-Carter mortality models using iterative fitting algorithm.

License GPL (>= 2)

LazyData yes

LazyLoad yes

LazyDataCompression xz

NeedsCompilation no

Repository CRAN

Date/Publication 2014-11-20 00:45:16

R topics documented:

`ilc-package` *Generalised Lee-Carter models using iterative fitting algorithms*

Description

The package implements a specialised iterative regression method for the analysis of age-period mortality based on a class of generalised Lee-Carter type modelling structure. Within the modelling framework of Renshaw and Haberman (2006), we use a Newton-Raphson iterative process to generate parameter estimates based on Poisson (or Gaussian) likelihood. In addition, we develop and implement here a stratified Lee-Carter model.

Details

The package contains methods for the analysis of a class of six different types of log-linear models in the GLM framework with Poisson or Gaussian errors that includes the basic LC model too. Also, this package includes tools for the fitting and analysis of the stratified LC model using an additional covariate (other than age and period). There are also made available some general diagnostic tools to analyse the data and the graduation results.

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

Maintainer: Zoltan Butt <Z.Butt@city.ac.uk>

References

Lee, R. and Carter, L. (1992), "Modelling and forecasting U.S. mortalit", *Journal of the American Statistical Association* **87**, 659-671.

Lee, L. (2000), "The Lee-Carter method for forecasting mortality, with various extensions and applications", *North American Actuarial Journal* **4**, 80-93.

Renshaw, A. E. and Haberman, S. (2003a), "Lee-Carter mortality forecasting: a parallel generalised linear modelling approach for England and Wales mortality projections", *Journal of the Royal Statistical Society, Series C*, **52**(1), 119-137.

Renshaw, A. E. and Haberman, S. (2003b), "Lee-Carter mortality forecasting with age specific enhancement", *Insurance: Mathematics and Economics*, **33**, 255-272.

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

Renshaw, A. E. and Haberman, S. (2008), "On simulation-based approaches to risk measurement in mortality with specific reference to Poisson Lee-Carter modelling", *Insurance: Mathematics and Economics*, **42**(2), 797-816.

Renshaw, A. E. and Haberman, S. (2009), "On age-period-cohort parametric mortality rate projections", *Insurance: Mathematics and Economics*, **45**(2), 255-270.

`coef.elca` *Extract extended Lee-Carter coefficients from an object of class elca*

Description

Extract extended Lee-Carter coefficients from an object of class elca

Usage

```
## S3 method for class 'elca'
coef(object, ...)
```

Arguments

`object`	Data object of class elca
`...`	Other arguments

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`print.elca`

coef.lca — *Extract Lee-Carter coefficients from an object of class lca*

Description

Extract Lee-Carter coefficients from an object of class lca

Usage

```
## S3 method for class 'lca'
coef(object, ...)
```

Arguments

`object`	Data object of class lca
`...`	Other arguments

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`coef.elca`

coef.rh — *Extract Lee-Carter coefficients from an object of class rh*

Description

Extract Lee-Carter coefficients from an object of class rh.

Usage

```
## S3 method for class 'rh'
coef(object, ...)
```

Arguments

`object`	Data object of class rh
`...`	Other arguments

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`print.rh`

`dd.cmi.pens` *Male mortality data of UK pensioners provided by Continuous Mortality Investigation UK*

Description

The dataset is made up by age- and time-specific mortality (hazard) rates and population (exposure) for male pensioners in the UK. Specifically, it covers individual ages between 50 - 108 and calendar years between 1983 - 2003.

Usage

```
data("dd.cmi.pens")
```

Format

Object of class `demogdata`

Source

Continuous Mortality Investigation (http://www.actuaries.org.uk/research-and-resources/pages/continuous-mortality-investigation)

See Also

`demogdata`

Examples

```
# print data summary:
dd.cmi.pens
#Mortality data for CMI
#    Series: male
#    Years: 1983 - 2003
#    Ages:  50 - 108
```

`dd.rfp` *Artificial (stratified) mortality experience (with Poisson error) for testing the SLC regression*

Description

It transforms a base age-period (2-dimensional) experience of mortality rates into artificially 'stratified' (3-dimensional) mortality rates by overlaying an extra random effect (i.e. other than age and period). Thus, it augments the log of the 2-dimensional mortality rates by an additive effect (with any number of levels) having Poisson distribution with means specified in the `rfp` argument of the function. It also randomises the base central exposures by a similar additive effect having a normal distribution with mean 0 and a constant age-specific standard deviation, which is calculated arbitrarily as the square root of the age-specific standard errors of the observed exposures. The latter adjustment is applied in order to further randomize the artificially created data. The purpose of the artificial data is to test the Stratified Lee-Carter regression method.

Usage

```
dd.rfp(ddata, rfp)
```

Arguments

`ddata`	mortality data object of class `demogdata`
`rfp`	vector of the means of the artificial additive effect, whereas the length of this argument determines the number of levels of the extra factor.

Details

Consider a cross-classified mortality experience observed over age (x) and period (t) made up of $k \times n$ data cells. This function will augment the observed data with Poisson distributed random 'noise' (reduction factors) corresponding to an extra covariate (g) having l levels with means specified in the `rfp` vector. That is, it creates an artificial data object made up by $k \times n \times l$ data cells containing the number of deaths corresponding to each subgroup of the stratified mortality experience.

Value

Multidimensional mortality data object of `rhdata` class with the following components:

`age`	vector of ages
`year`	vector of years
`covariates`	names of covariates
`deaths`	3-dimensional array of death counts
`pop`	3-dimensional array of exposure
`mu`	3-dimensional array of force of mortality
`label`	data label
`name`	data name

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

elca.rh

Examples

```
# vector of means of the additional effect (other than age and period):
rfp <- c(0.5, 1.2, -0.7, 2.5)
# create artificial stratified mortality experience of rhdata class:
rfp.cmi <- dd.rfp(dd.cmi.pens, rfp)
# print stratified rhdata data summary:
rfp.cmi
# plot the base level experience in the stratified rhdata
# a. central exposures:
matplot(rfp.cmi$age, rfp.cmi$pop[,,1], type='l', xlab='Age', ylab='Ec', main='Base Level')
# b. deaths:
matplot(rfp.cmi$age, rfp.cmi$deaths[,,1], type='l', xlab='Age', ylab='D', main='Base Level')
# c. log mortality rates:
matplot(rfp.cmi$age, log(rfp.cmi$mu[,,1]), type='l', xlab='Age', ylab='log(mu)', main='Base Level')
```

---

elca.rh *Extended (Stratified) Lee-Carter model (with a single extra parameter)*

Description

A purpose-built regression routine to fit the extended Lee-Carter model with an extra additive effect of an observable factor (other than age and period) on the log mortality mortality rates.

Usage

```
elca.rh(dat, year = dat$year, age = dat$age, dec.conv = 6,
error = c("poisson", "gaussian"),
restype = c("logrates", "rates", "deaths", "deviance"),
scale = F, interpolate = F, verbose = T, spar = NULL, ax.fix = NULL)
```

Arguments

dat	rhdata class multidimensional mortality data object
year	vector of years to be included in the regression (all available years by default)
age	vector of ages to be included in the regression (all available ages by default)
dec.conv	number of decimal places used to achieve convergence. The lower the value the faster the convergence of the fitting algorithm. Note: very high values could over fit the parameters.

error	type of error structure of the model choice (Poisson distribution of the errors by default)
restype	types of residuals, which also controls the type of the fitted value. Thus, in the cases of logrates and rates the function returns as fitted values the log and untransformed mortality rates, respectively. Likewise, the choices of deaths and deviance correspond to the fitted number of deaths
scale	logical, if TRUE, re-scale the interaction parameters so that the k_t has drift parameter equal to 1 (see also lca)
interpolate	logical, if TRUE, replace before regression all zero or missing values in the mortality rates of dat argument by interpolation across calendar years (see also smooth.demogdata)
verbose	logical, it controls the amount of process information
spar	numerical smoothing spline parameter in the interval (0,1] (with a recommended value of 0.6). If it is not NULL, the interaction effects (i.e. $\beta_x^{(0,1)}$) are smoothed out after the initial regression. Consequently, the period and/or cohort effects are adjusted (smoothed out) accordingly.
ax.fix	vector of constant age effect to be used in the model (e.g. the fitted values of a standard LC regression to the experience of a large population). If NULL the base ax values are estimated from dat

Details

This function models the number of deaths for a group within a generalised Lee-Carter framework with a Poisson or Gaussian error structure. The methodology quantifies the differences in the mortality experience of population subgroups differentiated by an additional measurable covariate (other than age and period). Additional covariate, for instance, could be related to geographical, socio-economic or race differences.

Value

An object of class elca with the following components:

lca	list of fitted lca model objects by the level of the extra factor
age	vector of fitted ages
year	vector of fitted years
ag	parameter estimates of the effects of the extra factor
ax	parameter estimates (or ax.fix) of (mean) age-specific mortality rates across the entire fitting period
bx	parameter estimates of age-specific interaction effect between age and period
kt	parameter estimates of year-specific period trend of mortality rates
adjust	type of error structure used in fitting (e.g. "poisson" or "gaussian")
label	data label
call	copy of the R call to the model
conv.iter	number of iterations used to reach convergence

mdev	mean deviance of total and base lack of fit (see also lca)
model	string expression of the fitted model
df	degree of freedom of the fitted GLM model

Author(s)

Z. Butt and S. Haberman and H. L. Shang

References

Li, N. and Lee, R. D. (2005), ‘Cohort mortality forecasts for a group of populations: an extension of the Lee-Carter method’, Demography, 42(3), 575-594. Renshaw and Haberman (2006), ‘A cohort-based extension to the Lee-Carter model for mortality reduction factors.’, Insurance: Mathematics and Economics, 38, 556-570.

See Also

dd.rfp, link{rhdata}

Examples

```
rfp <- c(0.5, 1.2, -0.7, 2.5)
rfp.cmi <- dd.rfp(dd.cmi.pens, rfp)
mod6e <- elca.rh(rfp.cmi, age=50:100, interp=TRUE, dec=3, verb=TRUE)
# display model summary and diagnostics:
mod6e; coef(mod6e)
```

---

extract.deaths *Miscellaneous utility functions for* demogdata *type mortality data*

Description

This function calculates and outputs the corresponding (observed) number of deaths from a demogdata type mortality data for a choice of ages and calendar years.

Usage

```
extract.deaths(data, ages = data$age, years = data$year, combine.upper = T,
fill.method = NULL, series = names(data$rate)[1])
```

Arguments

data	mortality data object of demogdata class
ages	vector of ages to extract
years	vector of years to extract
combine.upper	logical, if TRUE, ages above max(ages) will be grouped together

`fill.method` string value indicating the method to be used for correcting missing or 0 transition rates before estimating the number of deaths (e.g. one of "perks", "interpolate" or "mspline"). By default is set to NULL, which corresponds to no data correction.

`series` target series name (e.g. 'males') or index number (e.g. 1) of the data object to be extracted

Value

A 'fictive' `demogdata` class object in which the (mortality) `rate` component is replaced by the extracted number of deaths.

Note

When estimating the number of deaths (as the product between mortality rates and exposures), some assumptions will need to be made for the cases where the mortality rates are missing (NA). That is, it is not possible to estimate the number of deaths where the exposure (population) is zero because for those cases the corresponding hazard rate (mu) will most likely be NA in the dataset (unless mu was estimated by other means, like a moving average or smoothing, etc.). However, it is reasonable to assume that zero exposures correspond to no observed deaths, which is implemented here. Further, when a `fill.method` is specified, then the zero and the missing mortality rates are corrected before calculating the number of deaths.

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

`insp.dd`, `extract.ages`, `extract.years`

Examples

```
# 'observed' number of deaths (i.e. no data correction)
extract.deaths(dd.cmi.pens, ages=55:100)
# number of deaths with corrections using Perks mortality model
tmp.d <- extract.deaths(dd.cmi.pens, ages=55:100, fill='perks')
# Note: to further improve the plot the user can change the vertical
# axis label and/or main title (amongst other plotting parameters).
plot(tmp.d, transf=FALSE, ylab='Number of Deaths')    # change ylab
plot_dd(tmp.d, transf=FALSE, ylab='Number of Deaths', lpar=list(x.int=-0.2, y.int=0.9, cex=0.85))
plot_dd(tmp.d, y=1995:2003, transf=FALSE, lty=1:5, ylab='Number of Deaths',
  main=paste(tmp.d$lab, ": male (1995-2003)", sep='')) # change main title
```

`fill.rhdata` *Interpolate/Smooth an object of class rhdata*

Description

Interpolate/Smooth an object of class rhdata

Usage

```
fill.rhdata(data, method = c("mspline", "interpolate", "perks"), ...)
```

Arguments

`data`	Data object of class rh
`method`	Method of interpolation or smoothing
`...`	Other arguments

Value

An object does not have missing values

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

`fitted_plots` *Miscellaneous plotting functions for* `lca` *and* `lca.rh` *type regression objects*

Description

A diagnostic plot with two graphical regions showing the fitted log rates by the given ages and calendar years.

Usage

```
fitted_plots(lca.obj, file = paste("fit", deparse(substitute(lca.obj)),
 "ps", sep = "."), view = T, labs = T, col)
```

Arguments

lca.obj	an object of class `lca.rh`
file	an optional string value indicating the output postscript file name (i.e. with extension .ps). By default, it concatenates "fit." and the model object name (with extension ".ps"). If it is set to NULL, the plot will be sent instead to the active graphical window
view	logical, if TRUE (and `file` argument is not NULL) then Ghostview will be launched with the created .ps file
labs	logical, if TRUE, it adds age/years text labels to fitted curves
col	color palette to be used in the plot

Value

Diagnostic plots of fitted curves by age and by calendar year of a Lee-Carter model object.

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

`residual_plots`, `lca.rh`

Examples

```
mod6 <- lca.rh(dd.cmi.pens, mod='lc', max=110)
# send fitted plots with legends to 'fit.mod6.ps' file:
fitted_plots(mod6)
# send fitted plots without legends to active graphics window
fitted_plots(mod6, file=NULL, labs=FALSE)
```

---

flc.plot

Miscellaneous plotting functions for `lca` and `lca.rh` type regression objects. Plot of forecasted Lee-Carter model based on a fitted model object

---

Description

Plots the forecasted period effect and life expectancy of the fitted Lee-Carter model in a single graphical window.

Usage

```
flc.plot(lca.obj, at = 65, ...)
```

Arguments

`lca.obj`	an object of class `lca`
`at`	target age at which to calculate life expectancy
`...`	additional arguments to `forecast` function

Details

It makes use of a univariate ARIMA process (i.e. random walk with drift) in order to extrapolate the period effect k_t, which is illustrated by the fitted calendar years together with the corresponding forecasted life expectancy at the specified target age.

Value

Plot

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

`residual_plots`, `fitted_plots`, `fle.plot`, `forecast`, `life.expectancy`

Examples

```
mod6 <- lca.rh(dd.cmi.pens, mod='lc', interpolate=TRUE)
flc.plot(mod6, at=60, h=30, level=90)
```

`fle.plot` *Miscellaneous plotting functions for `lca` and `lca.rh` type regression objects. Plot of forecasted life expectancy based on a fitted Lee-Carter model object*

Description

Compute the historical and forecasted life expectancy from a fitted LC object and plot them in a single (overlay) figure.

Usage

```
fle.plot(lca.obj, at = 65, ...)
```

Arguments

`lca.obj`	an object of class `lca`
`at`	target age at which to calculate life expectancy
`...`	additional arguments to `forecast` function

Details

It makes use of the `life.expectancy` and `forecast` functions from the `demography` and `forecast` packages, respectively, in order to compute life expectancy at the specified target age.

Value

Plot

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

`residual_plots`, `fitted_plots`, `flc.plot`, `forecast`, `life.expectancy`

Examples

```
mod6 <- lca.rh(dd.cmi.pens, mod='lc', interpolate=TRUE)
fle.plot(mod6, at=60, h=30, level=90)
```

---

insp.dd *Miscellaneous utility functions for* `demogdata` *type mortality data*

Description

This function can extract any subset of the source data, such as the mortality (hazard) rates and population (exposure), by a given vector of ages and calendar years. Similarly, the function can output the observed number of deaths by age and calendar years, based on the source data sets included in the `demogdata` object.

Usage

```
insp.dd(data, what = c("rate", "pop", "deaths"), ages = data$age,
years = data$year, series = names(data$rate)[1])
```

Arguments

`data`	`demogdata` mortality data object
`what`	specifies the required type of data matrix to be extracted
`ages`	vector of ages to inspect in the data
`years`	vector of years to inspect in the data
`series`	target series name (e.g. 'males') or index number (e.g. 1) of the data object to be extracted

Details

A subset of mortality rates and population (exposures) can be directly inspected (i.e. extracted) from the corresponding component data matrices in the source `demogdata` object by using this function. To inspect the observed number of deaths by age and calendar years the function calls `extract.deaths` without data corrections.

Value

Returns a subset data matrix of the source data.

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`extract.deaths`, `extract.ages`, `extract.years`

Examples

```
# inspect mortality (hazard) rates:
insp.dd(dd.cmi.pens, 'rate', age=50:80, year=1985:1990)
# inspect exposure (population) values:
insp.dd(dd.cmi.pens, 'pop', age=50:80, year=1985:1990)
# inspect 'estimated' number of deaths
insp.dd(dd.cmi.pens, 'deaths', age=50:80, year=1985:1990)
```

`lca.dev.res` *Miscellaneous utility functions for* `lca` *and* `lca.rh` *type regression objects. Deviance residuals of the Lee-Carter model*

Description

A simple utility function to replace the original residuals (e.g. logratess, rates, deaths) of a LC fit with deviance residuals without the need to re-estimate the regression parameters. We note that the estimation of the parameters can be particularly slow in the case of the APC model.

Usage

```
lca.dev.res(lca.obj, pop, clip = 0)
```

Arguments

`lca.obj`	an object of class `lca`
`pop`	matrix of population data corresponding to the fitted mortality rates
`clip`	number of years to clip from start and end of cohort years

Details

The Lee-Carter regression object contains the type of residuals specified in the original function call, which might need to be changed for further analysis, but without actually re-running the entire iterative estimation process.

Value

An identical regression object as `lca.obj` containing the corresponding deviance residuals

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`lca.rh`

Examples

```
# original model object with 'lograte' residuals
mod6 <- lca.rh(dd.cmi.pens, mod="lc", error="gauss", max=110, interpolate=TRUE)
# adjusted model object with 'deviance' residuals:
dev6 <- lca.dev.res(mod6, insp.dd(dd.cmi.pens, "pop"))
```

---

lca.rh *A class of generalised Lee-Carter models*

Description

A purpose-built regression routine to fit any of the six variants of the class of Lee-Carter model structures using an iterative Newton-Raphson fitting method.

Usage

```
lca.rh(dat, year = dat$year, age = dat$age, series = 1, max.age = 100,
   dec.conv = 6, clip = 3, error = c("poisson", "gaussian"),
   model = c("m", "h0", "h1", "h2", "ac", "lc"),
     restype = c("logrates", "rates", "deaths", "deviance"), scale = F,
     interpolate = F, verbose = T, spar = NULL)
```

Arguments

`dat`	source data object of `demogdata` class
`year`	vector of years to be included in the regression (all available years by default)
`age`	vector of ages to be included in the regression (all available ages by default)
`series`	numerical index corresponding to the target series to be used from the source data

max.age highest age to be used in the regression

dec.conv number of decimal places used to achieve convergence. The lower the value the faster the convergence of the fitting algorithm.

clip number of marginal birth cohorts to exclude from the regression (i.e., give 0 weights). It is only applicable to the first 5 models (see below)

error type of error structure of the model choice (Poisson distribution of the errors by default)

model a character (see usage) or a numeric value (1-6) to specify the model choice

restype types of residuals, which also controls the type of the fitted value. Thus, in the cases of logratesand rates the function returns as fitted values the log and untransformed mortality rates, respectively. Likewise, the choices of deaths and deviance correspond to the fitted number of deaths

scale logical, if TRUE, re-scale the interaction parameters so that the k_t has drift parameter equal to 1 (see also lca)

interpolate logical, if TRUE, replace before regression all zero or missing values in the mortality rates of dat argument by interpolation across calendar years (see also smooth.demogdata)

verbose logical, if TRUE, the program prints out the updated deviance values along with the starting and final parameter estimates

spar numerical smoothing spline parameter in the interval (0,1] (with a recommended value of 0.6). If it is not NULL, the interaction effects (i.e. $\beta_x^{(0,1)}$) are smoothed out after the initial regression. Consequently, the period and/or cohort effects are adjusted (smoothed out) accordingly.

Details

Implements the modelling approach proposed in Renshaw and Haberman (2006), which extends the basic Lee-Carter model within the GLM framework. The function makes use of tailored iterative Newton-Raphson fitting algorithms to estimate the graduation parameters of the six variants within this class of extended Lee-Carter models.

Value

A Lee-Carter type fitted object with the following components:

label data label

age vector of fitted ages

year vector of fitted fitted years

<series> matrix of observed (source) mortality rates used for fitting. It is named the same way as the chosen series

ax parameter estimates of (mean) age-specific mortality rates across the entire fitting period

bx parameter estimates of age-specific interaction effect between age and period

kt parameter estimates of year-specific period trend of mortality rates

df	degree of freedom of the fitted GLM model
residuals	residuals of the fitted model in the form of a functional time series object
fitted	fitted values of the fitted model in the form of a functional time series object
varprop	percent of variance
y	source mortality data in the form of a functional time series object
mdev	mean deviance of total and base lack of fit (see also lca)
model	string expression of the fitted model
adjust	type of error structure (e.g. "poisson" or "gaussian")
call	copy of the R call to the model
conv.iter	number of iterations used to reach convergence
bx0	parameter estimates of age-specific interaction effect between age and cohort (only applies to the age-period-cohort model)
itx	parameter estimates of year-specific cohort trend of mortality rates (only applies to the age-period-cohort model)

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

References

Renshaw, A. E. and Haberman, S. (2003a), “Lee-Carter mortality forecasting: a parallel generalised linear modelling approach for England and Wales mortality projections", *Journal of the Royal Statistical Society, Series C*, **52**(1), 119-137.

Renshaw, A. E. and Haberman, S. (2003b), “Lee-Carter mortality forecasting with age specific enhancement", *Insurance: Mathematics and Economics*, **33**, 255-272.

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

Renshaw, A. E. and Haberman, S. (2008), “On simulation-based approaches to risk measurement in mortality with specific reference to Poisson Lee-Carter modelling", *Insurance: Mathematics and Economics*, **42**(2), 797-816.

Renshaw, A. E. and Haberman, S. (2009), “On age-period-cohort parametric mortality rate projections", *Insurance: Mathematics and Economics*, **45**(2), 255-270.

See Also

dd.rfp, elca.rh, lca

Examples

```
# standard LC model with Gaussian errors (corresponding to SVD graduation):
#   correct 0 or missing mortality rates before graduation
mod6g <- lca.rh(dd.cmi.pens, mod='lc', error='gauss', max=110, interpolate=TRUE)
# AP LC model with Poisson errors
mod6p <- lca.rh(dd.cmi.pens, mod='lc', error='pois', interpolate=TRUE)
```

```
# Model Summary, Coefficients and Plotting:
mod6p; coef(mod6p); plot(mod6p)

# Comparison with standard fitting method
# Standard LC model (with Gaussian errors) - SVD fit (demography package)
modlc <- lca(dd.cmi.pens, interp=TRUE, adjust='none')
# Gaussian (SVD) - Gaussian (iterative)
round(modlc$ax-mod6g$ax, 4)
round(modlc$bx-mod6g$bx, 4)
round(modlc$kt-mod6g$kt, 4)
# ---------------------------------------------------- #

# APC LC model fitted to restricted age range with 'deviance' residuals
#  the remaining 0/NA values reestimated:
# WARNING: for proper fit recommend dec=6, but it can lead to slow convergence!
mod1 <- lca.rh(dd.cmi.pens, age=60:100, mod='m', interpolate=TRUE, res='dev', dec=1)
```

`matflc.plot` *Miscellaneous plotting functions for* `lca` *and* `lca.rh` *type regression objects. Plot of forecasted Lee-Carter models based on a series of fitted model objects*

Description

Comparison plots of the forecasted period effect and life expectancy of a series of fitted Lee-Carter models

Usage

```
matflc.plot(lca.obj, lca.base, at = 65, label = NULL, ...)
```

Arguments

`lca.obj`	a list of fitted model objects of class `lca` (such as returned by `elca.rh` function)
`lca.base`	base fitted model object of class `lca` to be used in comparison
`at`	target age at which to calculate life expectancy
`label`	a data label
`...`	additional arguments to `forecast` function

Details

The function makes use of a univariate ARIMA process (i.e. random walk with drift) in order to extrapolate the period effects k_t of the model objects in `lca.obj`, which is illustrated by the calendar years together with the corresponding forecasted life expectancy for a given age.

Value

Plot

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

matfle.plot, flc.plot, elca.rh

Examples

```
rfp.cmi <- dd.rfp(dd.cmi.pens, c(0.5, 1.2, -0.7, 2.5))
mod6e <- elca.rh(rfp.cmi, age=50:70, interpolate=TRUE, dec=3)
# plot with original (fitted) base values
matflc.plot(mod6e$lca, label='RFP CMI')
# use a standard LC model fitting as base values
mod6 <- lca.rh(dd.cmi.pens, mod='lc', error='gauss', max.age = 70, interpolate=TRUE)
matflc.plot(mod6e$lca, mod6, label='RFP CMI')
```

---

matfle.plot — *Miscellaneous plotting functions for `lca` and `lca.rh` type regression objects. Plot of forecasted life expectancy based on a series of fitted Lee-Carter model objects*

Description

Compute the historical and forecasted life expectancy of a series of fitted Lee-Carter models and plot them in one comparative figure

Usage

```
matfle.plot(lca.obj, lca.base, at = 65, label = NULL, ...)
```

Arguments

`lca.obj`	a list of fitted model objects of class `lca` (such as returned by `elca.rh` function)
`lca.base`	base fitted model object of class `lca` to be used in comparison
`at`	target age at which to calculate life expectancy
`label`	a data label
`...`	additional arguments to `forecast` function

Details

It makes use of the `life.expectancy` and `forecast` functions from the `demography` and `forecast` packages, respectively, in order to compute life expectancy at the specified target age for each of the model objects in `lca.obj`.

Value

Plot

Author(s)

Z. Butt and S. Haberman and H. L. Shang

See Also

matflc.plot, fle.plot, elca.rh

Examples

```
rfp.cmi <- dd.rfp(dd.cmi.pens, c(0.5, 1.2, -0.7, 2.5))
mod6e <- elca.rh(rfp.cmi, age=50:100, interpolate=TRUE, dec=3)
# plot with original (fitted) base values
matfle.plot(mod6e$lca, label='RFP CMI')
# use a standard LC model fitting as base values
mod6 <- lca.rh(dd.cmi.pens, mod='lc', error='gauss', interpolate=TRUE)
matfle.plot(mod6e$lca, mod6, label='RFP CMI')
```

---

plot.elca *Plot an object of class elca*

Description

Plot an object of class elca

Usage

```
## S3 method for class 'elca'
plot(x, ...)
```

Arguments

x	An object of class elca
...	Other arguments

Value

A plot

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

print.elca

plot.rh *Plot an object of class rh*

Description

Plot an object of class rh

Usage

```
## S3 method for class 'rh'
plot(x, ...)
```

Arguments

x	An object of class rh
...	Other arguments

Value

A plot

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

print.rh

plot_coh_pars *Miscellaneous plotting functions for* `lca.rh` *type regression objects. Plot of the cohort effects of the generalised Lee-Carter model*

Description

This function plots the age- and time-specific patterns of the cohort effects (only) obtained from the fitting of a generalised Lee-Carter model.

Usage

```
plot_coh_pars(lca.obj)
```

Arguments

lca.obj	an object of class `lca.rh` (containing a generalised LC model with a cohort effect)

Value

A plot with two graphical regions showing the age- and time-specific cohort parameters (i.e. $beta_x^{(0)}$ and $iota_t$).

Author(s)

Z. Butt and S. Haberman and H. L. Shang

References

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

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

See Also

`plot.per.pars`, `lca.rh`

Examples

```
mod1 <- lca.rh(dd.cmi.pens, age=60:100, mod='m', interpolate=TRUE, res='dev', dec=1)
plot_coh_pars(mod1)
```

---

plot_dd — *Miscellaneous plotting functions for* `demogdata` *type mortality data. Versatile plotting tool with an optional legend.*

Description

Plot an object of class `demogdata`

Usage

```
plot_dd(dd.obj, year = dd.obj$year, col = rainbow(length(year), start = 0.1),
lpos = "UL", lpar = list(), ppar = T, ...)
```

Arguments

`dd.obj`	a mortality type data object of class `demogdata` (or number of deaths type data object returned by `extract.deaths`)
`year`	vector of years to be included in the plotting (all available years by default)
`col`	color palette to be used in the plot (by default, it uses a sequence of *rainbow* colors)

lpos — a text identifier (one of "UR", "LR", "UL", "LL", "UC", "LC", "CL", "CR"; whereas the abbreviation is made up by U/L/C=Upper/Lower/Center, L/R/C=Left/Right/Center) or a list containing the coordinates (e.g. x and y) of the upper left corner of the legend/object.

lpar — list of additional arguments to be passed on to `legend` (other than `legend`, `title`, `col` or `text.col`)

ppar — logical, if FALSE, ignores in the legend the plotting arguments `lty`, `lwd` and `pch` (i.e. in case they are given in `...`) and hence it creates a legend containing only text

... — additional plotting arguments that are passed on to both `plot` and `legend` functions (see `par`)

Value

Plot of mortality rates or number of deaths.

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`extract.deaths`, `plot.demogdata`, `legend`, `par`, `rainbow`

Examples

```
# plot log mortality rates with repositioned legend
plot_dd(dd.cmi.pens, xlim=c(40, 110), lpar=list(x.intersp=-0.2, y.intersp=0.9, cex=0.85))
# plot (untransformed) mortality rates with repositioned legend
plot_dd(dd.cmi.pens, age=60:95, lpar=list(x.intersp=-0.2, y.intersp=0.9, cex=0.85), transf=FALSE)
# plot a small subset of log mortality rates (calendar years: 1985 - 1995)
#      and add a line with the overall mean rates
plot_dd(dd.cmi.pens, lpos=list(x=0.85,y=0.55), year=1985:1995,
lpar=list(x.intersp=-0.1, y.intersp=0.95, cex=0.9))
lines(mean(dd.cmi.pens),lwd=2, lty=3, col='red')
# legend(coord('LC'), legend='mean rate', lwd=2, lty=3, col='red', text.col='red')
# plot number of (extracted) deaths:
tmp.d <- extract.deaths(dd.cmi.pens, ages=55:100, y=1995:2003)
plot_dd(tmp.d, transf=FALSE, lty=1:5, ylab='Number of Deaths',
main=paste(tmp.d$lab, ": male (1995-2003)", sep=''))
```

`plot_per_pars` — *Miscellaneous plotting functions for* `lca` *and* `lca.rh` *type regression objects. Plot of the period effects of the (generalised) Lee-Carter model*

Description

This function plots the age- and time-specific patterns of the period effects (only) obtained from the fitting of a generalised Lee-Carter model.

Usage

```
plot_per_pars(lca.obj)
```

Arguments

`lca.obj`	an object of class `lca`

Value

A plot with two graphical regions showing the age- and time-specific period parameters (i.e. $beta_x^{(1)}$ and $kappa_t$).

Author(s)

Z. Butt and S. Haberman and H. L. Shang

References

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

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

See Also

`plot_coh_pars`, `lca.rh`

Examples

```
mod6 <- lca.rh(dd.cmi.pens, mod='lc', interpolate=TRUE)
plot_per_pars(mod6)
```

---

print.elca *Print function for class elca*

Description

Print an object of class elca

Usage

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

Arguments

`x`	Data object of class elca
`...`	Other arguments

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`print.rh`

`print.rh` *Print function for class rh*

Description

Print an object of class rhdata

Usage

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

Arguments

`x`	Data object of class rh
`...`	Other arguments

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`print.rhdata`

`print.rhdata` *Print function for class rhdata*

Description

Print an object of class rhdata

Usage

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

Arguments

`x`	Data object of class rhdata
`...`	Other arguments

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

`print.rh`

`residual_plots` *Miscellaneous plotting functions for* `lca` *and* `lca.rh` *type regression objects.*

Description

A diagnostic plot with three graphical regions showing the residual values by the given ages, calendar years and (cohort) years of birth.

Usage

```
residual_plots(lca.obj, file = paste("res", deparse(substitute(lca.obj)),
"ps", sep = "."), view = T)
```

Arguments

`lca.obj`	an object of class `lca`
`file`	an optional string value indicating the output postscript file name (i.e. with extension .ps). By default, it concatenates "res." and the model object name (with extension ".ps"). If it is set to NULL, the plot will be sent instead to the active graphical window.
`view`	logical, if TRUE (and `file` argument is not NULL) then Ghostview will be launched with the created .ps file

Value

Diagnostic plot of the residual values of a Lee-Carter model object.

Author(s)

Zoltan Butt, Steven Haberman and Han Lin Shang

See Also

fitted_plots, lca.rh

Examples

```
mod6d <- lca.rh(dd.cmi.pens, mod='lc', restype='dev', interpolate=TRUE)
# send fitted plots with legends to 'res.mod6.ps' file:
residual_plots(mod6d)
# send fitted plots to active graphics window
residual_plots(mod6d, file=NULL)
```

---

rhdata *Data formatting utility for the extended (Stratified) LC model function*

Description

It creates rhdata class object suitable for fitting the extended SLC model using elca.rh iterative fitting method. Basically, it transforms a two-dimensional survival data into three-dimensional arrays of population (exposure) and mortality rates dependent on age, calendar time and additional covariate(s).

Usage

```
rhdata(dat, covar, xbreaks = 60:96, xlabels = 60:95,
ybreaks = mdy.date(1, 1, 1999:2008), ylabels = 1999:2007,
name = NULL, label = NULL)
```

Arguments

dat	data.frame containing individual observations of survival data along with values of additional covariate(s). Thus, the data set needs to contain the following named columns of individual survival records: - 'event' = binary value corresponding to the survival event (1 - fail/death or 0 - survive); - 'dob' = Julian date corresponding to the date of birth (or origin) of the survival time; - 'dev' = Julian date corresponding to date of event (or end) of the survival time. In addition, there should be at least one extra column corresponding to observations related to any additional covariate(s) (e.g. socio-economic factors).
covar	(partial) covariate name(s) or position number(s) in the dat data set. The covariate(s) must be of class 'factor'.

xbreaks	a sequence of age break points (including the starting and closing values) to be used for sub-grouping the input data set dat in order to calculate age-specific exposures and mortality rates. By default, it is set to 60:96 that corresponds to integer ages between 60 - 95.
xlabels	a sequence of age labels to be used for the sequence defined in xbreaks.
ybreaks	a sequence of year break points (as Julian calendar dates) to be used for sub-grouping the input data set dat in order to calculate year-specific exposures and mortality rates. By default, it is set to mdy.date(1, 1, 1999:2008) that corresponds to whole years between 1st of January of years 1999 - 2008.
ylabels	a sequence of year labels to be used for the sequence defined in ybreaks.
name	name of subset data series (e.g. male, female or total)
label	label (name) of overall data source (e.g. CMI)

Details

While the rhdata function can sub-group the input data by more than one additional covariates (possibly useful for other preliminary analysis), the fitting method implemented in elc.rh can only handle a single additional covariate. Also, currently, there are no generic methods to plot or to extract parts of the rhdata class object, but there are a few illustrations provided below how these might be carried out.

Value

List object defined as class rhdata made up by the following components:

year	vector of year labels
age	vector of age labels
covariates	vector of levels of the additional covariate
deaths	3-dimensional array of number of deaths (by age-year-covariate)
pop	3-dimensional array of population (exposure) (by age-year-covariate)
mu	3-dimensional array of central mortality rates (by age-year-covariate)
label	label (name) of overall data source
name	name of subset data series

Author(s)

Z. Butt and S. Haberman and H. L. Shang

References

Renshaw, A. E. and Haberman, S. (2003a), “Lee-Carter mortality forecasting: a parallel generalised linear modelling approach for England and Wales mortality projections", *Journal of the Royal Statistical Society, Series C*, **52**(1), 119-137.

Renshaw, A. E. and Haberman, S. (2003b), “Lee-Carter mortality forecasting with age specific enhancement", *Insurance: Mathematics and Economics*, **33**, 255-272.

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

Renshaw, A. E. and Haberman, S. (2008), “On simulation-based approaches to risk measurement in mortality with specific reference to Poisson Lee-Carter modelling", *Insurance: Mathematics and Economics*, **42**(2), 797-816.

Renshaw, A. E. and Haberman, S. (2009), “On age-period-cohort parametric mortality rate projections", *Insurance: Mathematics and Economics*, **45**(2), 255-270.

See Also

elca.rh, dd.rfp, demogdata, mdy.date

Examples

```
# See data set 'tab' provided in the ilc package
# names(tab)
# [1] "refno" "dob"   "dev"   "event" "cov1"  "cov2"
# Get multidimensional survival data:
mdat <- rhdata(tab, covar='cov2', xbreaks=60:96, xlabels=60:95,
  ybreaks=mdy.date(1,1,2000:2006), ylabels=2000:2005, name='M', label='CMI')
# Warning: although rhdata() can sort by more than a single parameter, for ex.
#   covar=c('cov1', 'cov2'), the SLC fitting only works at the moment with
#   a single extra covariate.

# print data summary:
mdat
#Multidimensional Mortality data for: MDat [M]
#Across covariates:
#         years: 2000 - 2005
#         ages:  60 - 95
#         cov2: 0, 1, 2, 3
# Graphical illustrations of mdat data levels (by the additional factor):
# plot of exposures:
matplot(mdat$age, mdat$pop[,,1], type='l', xlab='Age', ylab='Ec', main='Base Level')
matplot(mdat$age, mdat$pop[,,2], type='l', xlab='Age', ylab='Ec', main='Level 1')
# plot of deaths:
matplot(mdat$age, mdat$deaths[,,1], type='l', xlab='Age', ylab='D', main='Base Level')
matplot(mdat$age, mdat$deaths[,,2], type='l', xlab='Age', ylab='D', main='Level 1')
# plot of log mortality rates:
matplot(mdat$age, log(mdat$mu[,,1]), type='l', xlab='Age', ylab='log(mu)', main='Base Level')
matplot(mdat$age, log(mdat$mu[,,2]), type='l', xlab='Age', ylab='log(mu)', main='Level 1')
```

---

tab *Sample survival data with additional effects (other than age and time).*

Description

An artificial sample of individual observations of survival times along with two additional effects (fictive covariates).

Usage

```
tab
```

Format

Object of class `data.frame`

Details

The data set contains the following named columns of fictive individual survival records: - 'refno' = unique reference numbers; - 'dob' = Julian date corresponding to the date of birth (or origin) of the survival time; - 'dev' = Julian date corresponding to date of event (or end) of the survival time. - 'event' = binary value corresponding to the survival event (1 - fail/death or 0 - survive); - 'cov1' = first additional covariate with 13 levels; - 'cov1' = second additional covariate with 4 levels;

Source

NA

See Also

rhdata

Examples

```
# print out the first 10 observations:
tab[1:10,]
# sub-group by a single additional covariate and merge ages above 95:
mdat <- rhdata(tab, covar=c('cov2'), xbreaks=c(40:95, 105), xlabels=c(40:94, '95>'),
  ybreaks=mdy.date(1,1,2000:2007), ylabels=2000:2006, name='M', label='CMI')
mdat
```

Index