Package ‘ELT’

April 11, 2016

Type Package
Title Experience Life Tables
Version 1.6
Date 2016-04-10
Author Julien Tomas, Frederic Planchet, Wassim Youssef
Maintainer Wassim Youssef <Wassim.G.Youssef@gmail.com>
Description Build experience life tables.
License GPL (>= 2)
Depends R (>= 2.10.0), locfit, lattice, latticeExtra, xlsx
Encoding latin1
NeedsCompilation no
Repository CRAN
Date/Publication 2016-04-11 09:06:26

R topics documented:

ELT-package ......................................................... 2
AddReference ....................................................... 10
ComparisonMethods ............................................... 11
CompletionA ......................................................... 12
CompletionB ......................................................... 13
Dispersion ......................................................... 13
FctMethod1 ......................................................... 14
FctMethod2 ......................................................... 14
FctMethod3 ......................................................... 15
FctMethod4_1stPart ............................................... 15
FctMethod4_2ndPart ............................................... 16
Method1 .......................................................... 16
Method2 .......................................................... 17
Method3 .......................................................... 17
Method4A .......................................................... 18
Method4B .......................................................... 18
ELT-package

ELT - A package to build Experience Life Tables

Description

Collection of functions that can be used following a pre-established procedure to build and validate actuarial life tables.

Details

Package: ELT
Type: Package
Version: 1.6
Date: 2016-04-10
License: GNU
Depends: locfit,lattice,latticeExtra,xlsx

The package is meant to be used following a pre-established procedure.
See the reference for more info.

Please notice that the package includes the following internal functions:

These functions can be accessed with the prefix ELT::: using the following syntax: ELT::: [name of the function]. For example: ELT:::.GetHistory(). See technical note II1291-15 (http://www.ressources-actuarielles.net/gtmortalite) for the arguments and examples of the functions.
References

Tomas, J., Planchet, F., Prospective mortality tables and portfolio experience, Chapter 9 in Computational Actuarial Science, with R; Arthur Charpentier Editor, Chapman, 2014


Tomas, J., Planchet, F., Methodes de positionnement: Aspects Methodologiques, Institut des Actuaires, Rapport technique II1291-12, pp. 1-12, 2013


http://www.ressources-actuarielles.net/gtmortalite for data and exemple codes.

Examples

```r
## Not run:
data(MyPortfolio)
data(ReferenceMale)
data(ReferenceFemale)

## Initialize Age variables

AgeRange <- 30:90
AgeCrit <- 30:90
AgeRef <- 30:95

History <- ReadHistory(MyPortfolio = MyPortfolio, DateBegObs = "1996/01/01", DateEndObs = "2007/12/31", DateFormat ="

MyData <- AddReference(History = History, ReferenceMale = ReferenceMale, ReferenceFemale = ReferenceFemale)

## Execute method 1

OutputMethod1 <- Method1(MyData = MyData, AgeRange = AgeRange, Plot = T)

## Validate method 1 by the 1st level criteria
```
## Execute 1st level criteria.

ValidationLevel1Method1 <- ValidationLevel1(OutputMethod = OutputMethod1, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Plot = T, Excel = T)

## If the criterions corresponding to the 1st level are not satisfied, we can modify the age range used to compute the SMR and reexecute

## and

## ValidationLevel1Method1 <- ValidationLevel1(...).

## If the criterions corresponding to the 1st level are still not satisfied, we turn to the method 2, and it is useless to pursue the completion of the table and the validation.

## If the criterions are satisfied, we continue the validation with the criterions corresponding to the 2nd level.

## We can also turn to method 3 or 4 to improve the fit at a cost of a somewhat greater complexity.

## Validate method 1 by the 2nd level criterias

## Execute 2nd level criterions

ValidationLevel2Method1 <- ValidationLevel2(OutputMethod = OutputMethod1, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Excel = T)

## If the criterions corresponding to the 2nd level are not satisfied we turn to the method 2 and it is useless to pursue the completion of the table and the validation.

## If the criterions are satisfied, we continue the validation with the completion of the table and the criterions corresponding to the 3rd level.

## Completion Method 1

## Age range for the selection of the optimal starting age.

AgeRangeOptMale <- AgeRangeOptFemale <- c(80, 80)

## In theory, we could select the optimal starting age, however the optimal starting age can vary a lot with the calendar years leading to a relatively irregular surface. In practice, we select then a fixed age for the whole years.

## Starting age for which the fitted probabilities of the death are replaced by the values obtained from the completion model.

BegAgeCompMale <- BegAgeCompFemale <- 85
## We check if the completion is smoothed with graphical diagnostics.

CompletionMethod1 <- CompletionA(OutputMethod = OutputMethod1, MyData = MyData, AgeRangeOptMale = AgeRangeOptMale, AgeRangeOptFemale = AgeRangeOptFemale, BegAgeCompMale = BegAgeCompMale, BegAgeCompFemale = BegAgeCompFemale, ShowPlot = T)

## If the completion is not satisfying, we modify the values AgeRangeOpt and BegAgeComp, and we repeat the previous script

CompletionA()

## If the completion is satisfying, we execute

FinalMethod1 <- CompletionB(ModCompletion = CompletionMethod1, OutputMethod = OutputMethod1, MyData = MyData, Plot = T, Excel = T)

## Validate method 1 by the 3rd level criteria

## Execute 3rd level criterions

ValidationLevel3Method1 <- ValidationLevel3(FinalMethod = FinalMethod1, MyData = MyData, Plot = T, Excel = T)

## Compute the coefficient of variation, confidence intervals and relative dispersion of the fitted periodic life expectancies

DispersionMethod1 <- Dispersion(FinalMethod = FinalMethod1, MyData = MyData, Plot = T, NbSim = 10)

## Execute method 2

OutputMethod2 <- Method2(MyData = MyData, AgeRange = AgeRange, Plot = T)

## Validate method 2 by the 1st level criteria

## Execute 1st level criteria.

ValidationLevel1Method2 <- ValidationLevel1(OutputMethod = OutputMethod2, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Plot = T, Excel = T)
## -- If the criterions corresponding to the 1st level are not satisfied, we turn to the method 3 and it is useless to pursue the completion of the table and the validation.
## -- If the criterions are satisfied, we continue the validation with the criterions corresponding to the 2nd level.
## -- We can also turn to method 4 to improve the fit at a cost of a somewhat greater complexity.

----------------------------------------------- ##
## Validate method 2 by the 2nd level criterias
----------------------------------------------- ##

## -- Execute 2nd level criterias

ValidationLevel2Method2 <- ValidationLevel2(OutputMethod = OutputMethod2, AgeCrit = AgeCrit, ValCrit = 0.05, MyData = MyData, Excel = T)

## -- If the criterions corresponding to the 2nd level are not satisfied we turn to the method 3 and it is useless to pursue the completion of the table and the validation.
## -- If the criterions are satisfied, we continue the validation with the completion of the table and the criterions corresponding to the 3rd level.

----------------------------------------------- ##
## Completion Method 2
----------------------------------------------- ##

## -- We check if the completion is smoothed with graphical diagnostics.

CompletionMethod2 <- CompletionA(OutputMethod = OutputMethod2, MyData = MyData, AgeRangeOptMale = AgeRangeOptMale, AgeRangeOptFemale = AgeRangeOptFemale, BegAgeCompMale = BegAgeCompMale, BegAgeCompFemale = BegAgeCompFemale, ShowPlot = T)

## -- If the completion is not satisfying, we modify the values of AgeRangeOpt and BegAgeComp, and we repeat the previous script
## -- If the completion is satisfying, we execute

FinalMethod2 <- CompletionB(ModCompletion = CompletionMethod2, OutputMethod = OutputMethod2, MyData = MyData, Plot = T, Excel = T)

----------------------------------------------- ##
## Validate method 2 by the 3rd level criterias
----------------------------------------------- ##

## -- Execute 3rd level criterias

ValidationLevel3Method2 <- ValidationLevel3(FinalMethod = FinalMethod2, MyData = MyData, Plot = T, Excel = T)

----------------------------------------------- ##
## Coef Varition, Conf int. and rel. disp. of fitted per. life exp.  ##

---

## Compute the coefficient of variation, confidence intervals and relative dispersion of the fitted periodic life expectancies

DispersionMethod2 <- Dispersion(FinalMethod = FinalMethod2, MyData = MyData, Plot = T, NbSim = 10)

## ------ Execute method 3  ##

OutputMethod3 <- Method3(MyData = MyData, AgeRange = AgeRange, Plot = T)

## ------ Execute 1st level criteria  ##

ValidationLevel1Method3 <- ValidationLevel1(OutputMethod = OutputMethod3, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Plot = T, Excel = T)

## ------ If the criterions corresponding to the 1st level are not satisfied, we turn to the method 4 and it is useless to pursue the completion of the table and the validation. If the criterions are satisfied, we continue the validation with the criterions corresponding to the 2nd level.  ##

ValidationLevel2Method3 <- ValidationLevel2(OutputMethod = OutputMethod3, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Excel = T)

## ------ If the criterions corresponding to the 2nd level are not satisfied, we turn to the method 4 and it is useless to pursue the completion of the table and the validation. If the criterions are satisfied, we continue the validation with the completion of the table and the criterions corresponding to the 3rd level.  ##

CompletionMethod3
# We check if the completion is smoothed with graphical diagnostics.

CompletionMethod3 <- CompletionA(OutputMethod = OutputMethod3, MyData = MyData, 
AgeRangeOptMale = AgeRangeOptMale, AgeRangeOptFemale = AgeRangeOptFemale, 
BegAgeCompMale = BegAgeCompMale, BegAgeCompFemale = BegAgeCompFemale, ShowPlot = T)

# If the completion is not satisfying, we modify the values AgeRangeOpt and BegAgeComp, and we repeat the previous script
# If the completion is satisfying, we execute

FinalMethod3 <- CompletionB(ModCompletion = CompletionMethod3, OutputMethod = OutputMethod3, 
MyData = MyData, Plot = T, Excel = T)

# Validate method 3 by the 3rd level criterias

ValidationLevel3Method3 <- ValidationLevel3(FinalMethod = FinalMethod3, MyData = MyData, 
Plot = T, Excel = T)

# Coef Variation, Conf int. and rel. disp. of fitted per. life exp.

DispersionMethod3 <- Dispersion(FinalMethod = FinalMethod3, MyData = MyData, Plot = T, NbSim = 10)

# Execute method 4

OutputMethod4PartOne <- Method4A(MyData = MyData, AgeRange = AgeRange, AgeCrit = AgeCrit, 
ShowPlot = T)

# Select the optimal smoothing parameters.

OutputMethod4 <- Method4B(PartOne, MyData = MyData, OptMale = c(1, 16), 
OptFemale = c(1, 14), Plot = T)
# Validate method 4 by the 1st level criteria

# Execute 1st level criteria.

ValidationLevel1Method4 <- ValidationLevel1(OutputMethod = OutputMethod4, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Plot = T, Excel = T)

# If the criterions corresponding to the 1st level are not satisfied, we turn to the method 4, and it is useless to pursue the completion of the table and the validation. If the criterions are satisfied, we continue the validation with the criterions corresponding to the 2nd level.

# Validate method 4 by the 2nd level criteria

# Execute 2nd level criterions

ValidationLevel2Method4 <- ValidationLevel2(OutputMethod = OutputMethod4, MyData = MyData, AgeCrit = AgeCrit, ValCrit = 0.05, Excel = T)

# If the criterions corresponding to the 2nd level are not satisfied, we turn to the method 4 and it is useless to pursue the completion of the table and the validation. If the criterions are satisfied, we continue the validation with the completion of the table and the criterions corresponding to the 3rd level.

# Completion Method 4

# We check if the completion is smoothed with graphical diagnostics.

CompletionMethod4 <- CompletionA(OutputMethod = OutputMethod4, MyData = MyData, AgeRangeOptMale = AgeRangeOptMale, AgeRangeOptFemale = AgeRangeOptFemale, BegAgeCompMale = BegAgeCompMale, BegAgeCompFemale = BegAgeCompFemale, ShowPlot = T)

# If the completion is not satisfying, we modify the values AgeRangeOpt and BegAgeComp, and we repeat the previous script CompletionA()

# If the completion is satisfying, we execute

FinalMethod4 <- CompletionB(ModCompletion = CompletionMethod4, OutputMethod = OutputMethod4, MyData = MyData, Plot = T, Excel = T)

# Validate method 4 by the 3rd level criteria

# Validate method 4 by the 4th level criteria

### Execute 3rd level criterions

```
ValidationLevel3Method4 <- ValidationLevel3(FinalMethod = FinalMethod4, MyData = MyData, Plot = T, Excel = T)
```

### Coef Varitation, Conf int. and rel. disp. of fitted per. life exp.

```
## Set the number of simulations

DispersionMethod4 <- Dispersion(FinalMethod = FinalMethod4, MyData = MyData, Plot = T, NbSim = 10)
```

### COMPARISON OF THE METHODS

```
Once we have fitted the data with a number of methods, we can compare them. In the following, we compare the fitted probabilities of death in original and log scale, the residuals, the fitted deaths as well as the coherence of the extrapolated mortality trends.
```

### You can change the color vector for comparison, color need to be in html format

```
Store the output into a list

ListOutputs <- list(OutputMethod1, OutputMethod2, OutputMethod3, OutputMethod4)
ListValidationLevel1 <- list(ValidationLevel1Method1, ValidationLevel1Method2, ValidationLevel1Method3, ValidationLevel1Method4)
ListValidationLevel2 <- list(ValidationLevel2Method1, ValidationLevel2Method2, ValidationLevel2Method3, ValidationLevel2Method4)
ListValidationLevel3 <- list(ValidationLevel3Method1, ValidationLevel3Method2, ValidationLevel3Method3, ValidationLevel3Method4)

ComparisonsMethodsLevels123 <- ComparisonMethods(ListOutputs, ListValidationLevel1, ListValidationLevel2, ListValidationLevel3, MyData = MyData, Plot = T, AgeCrit = AgeCrit)
```

### End(Not run)
ComparisonMethods

Description

This function imports reference tables.

Usage

AddReference(History, ReferenceMale = NULL, ReferenceFemale = NULL)

Arguments

History        History as returned by the ReadHistory function.
ReferenceMale  data.frame representing the reference table. See data(ReferenceMale) for the
               format.
ReferenceFemale data.frame representing the reference table. See data(ReferenceFemale) for the
                  format.

ComparisonMethods

ComparisonMethods function

Description

This function compares two or several methods using the three groups of criteria from the validation process.

Usage

ComparisonMethods(ListOutputs, ListValidationLevel1, ListValidationLevel2,
                   ListValidationLevel3, MyData = MyData, Plot = F,
                   ColorComp = c("#FF6590", "#309BBF", "#AD79FC", "#3CAB5F"),
                   LtyComp = rep(1, 4), AgeCrit)

Arguments

ListOutputs      For the comparisons of n methods, a list of n elements containing the returned
                 value of the functions Methodn().
ListValidationLevel1 For the comparisons of n methods, a list of n elements containing the returned
                     value of the function ValidationLevel1() for each of the n methods.
ListValidationLevel2 For the comparisons of n methods, a list of n elements containing the returned
                     value of the function ValidationLevel2() for each of the n methods.
ListValidationLevel3 For the comparisons of n methods, a list of n elements containing the returned
                     value of the function ValidationLevel3() for each of the n methods.
MyData           The list returned by the AddReference() function.
Plot

If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain PNG plots corresponding to the smoothed surface.

ColorComp

The color that will be used for the plots (HTML notation). For the comparisons of n methods, ColorComp is a vector of length n.

LtyComp

Vector of parameters (length n) for the lty plot parameter.

AgeCrit

Age range for the comparison of adjusted mortality and observed mortality.

---

CompletionA

**CompletionA function**

**Description**

This function executes the first part of table closure using Denuit and Goderniaux (2005)

**Usage**

CompletionA(OutputMethod, MyData, AgeRangeOptMale, AgeRangeOptFemale, 
BegAgeCompMale, BegAgeCompFemale, Color = MyData$Param$Color, 
ShowPlot = T)

**Arguments**

OutputMethod

The list returned by one of these functions: Method1(), Method2(), Method3() or Method4B().

MyData

The list returned by the AddReference() function.

AgeRangeOptMale

Age range from which the optimal starting age is selected for males

AgeRangeOptFemale

Age range from which the optimal starting age is selected for females

BegAgeCompMale

For ages after BegAgeCompMale, observed death probability is replaced by the model output.

BegAgeCompFemale

For ages after BegAgeCompFemale, observed death probability is replaced by the model output.

Color

The color that will be used for the plots (HTML notation).

ShowPlot

If true, create graphics comparing Before/After the completion create graphics of the completed surfaces.
CompletionB

CompletionB function

Description
This function executes the second part of table closure.

Usage
CompletionB(ModCompletion, OutputMethod, MyData, Color = MyData$Param$Color, Plot = F, Excel = F)

Arguments
- **ModCompletion**: Output of the function CompletionA().
- **OutputMethod**: The list returned by one of these functions: Method1(), Method2(), Method3() or Method4B().
- **MyData**: The list returned by the AddReference() function.
- **Color**: The color that will be used for the plots (HTML notation).
- **Plot**: If true, create graphics.
- **Excel**: If true, create Excel files.

Dispersion

Dispersion function

Description
This function allows to calculate confidence intervals for period life expectancies.

Usage
Dispersion(FinalMethod, MyData, NbSim, CompletionTable = T, Plot = F, Color = MyData$Param$Color)

Arguments
- **FinalMethod**: The list returned by the CompletionB() function.
- **MyData**: The list returned by the AddReference() function.
- **NbSim**: The number of simulations for the Dispersion.
- **CompletionTable**: If TRUE, apply completion.
- **Plot**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain png plots describing the validation analysis.
- **Color**: The color that will be used for the plots (HTML notation).
**FctMethod1**  
*FctMethod1 function*

**Description**

FctMethod1() is an alternative to Method1(). It allows to process the smoothing without using a "Data" object and by defining all the needed parameters independently.

**Usage**

FctMethod1(d, e, qref, x1, xR, t1, tR)

**Arguments**

- **d**: Number of deaths.
- **e**: Exposure to risk.
- **qref**: Mortality rates in Reference Table.
- **x1**: Age range used for calculation.
- **xR**: Age range of reference table.
- **t1**: Calendar years used for the calculation. It corresponds to the common years among observations and the reference table.
- **tR**: Calendar years of the reference.

---

**FctMethod2**  
*FctMethod2 function*

**Description**

FctMethod2() is an alternative to Method2(). It allows to process the smoothing without using a "Data" object and by defining all the needed parameters independently.

**Usage**

FctMethod2(d, e, qref, x1, x2, t1, t2)

**Arguments**

- **d**: Number of deaths.
- **e**: Exposure to risk.
- **qref**: Mortality rates in Reference Table.
- **x1**: Age range used for calculation.
- **x2**: Age range of reference table.
- **t1**: Calendar years used for the calculation. It corresponds to the common years among observations and the reference table.
- **t2**: Calendar years of the reference.
**FctMethod3**

**FctMethod3 function**

**Description**

FctMethod3() is an alternative to Method3(). It allows to process the smoothing without using a "Data" object and by defining all the needed parameters independently.

**Usage**

FctMethod3\(d, e, q_{ref}, x_1, x_2, t_1, t_2\)

**Arguments**

- \(d\) Number of deaths.
- \(e\) Exposure to risk.
- \(q_{ref}\) Mortality rates in Reference Table.
- \(x_1\) Age range used for calculation.
- \(x_2\) Age range of reference table.
- \(t_1\) Calendar years used for the calculation. It corresponds to the common years among observations and the reference table.
- \(t_2\) Calendar years of the reference.

**FctMethod4_1stPart**

**FctMethod4_1stPart function**

**Description**

FctMethod4_1stPart() is an alternative to Method4A(). It allows to process the smoothing without using a "Data" object and by defining all the needed parameters independently.

**Usage**

FctMethod4_1stPart\(d, e, q_{ref}, x_1, x_2, t_1\)

**Arguments**

- \(d\) Number of deaths.
- \(e\) Exposure to risk.
- \(q_{ref}\) Mortality rates in Reference Table.
- \(x_1\) Age range used for calculation.
- \(x_2\) Age range of reference table.
- \(t_1\) Calendar years used for the calculation. It corresponds to the common years among observations and the reference table.
**FctMethod4_2ndPart function**

**Description**

FctMethod4_2ndPart() is an alternative to Method4B(). It allows to process the smoothing without using a "Data" object and by defining all the needed parameters independently.

**Usage**

FctMethod4_2ndPart(d, e, qref, x1, x2, t1, t2, p.Opt, h.Opt)

**Arguments**

- **d**: Number of deaths.
- **e**: Exposure to risk.
- **qref**: Mortality rates in Reference Table.
- **x1**: Age range used for calculation.
- **x2**: Age range of reference table.
- **t1**: Calendar years used for the calculation. It corresponds to the common years among observations and the reference table.
- **t2**: Calendar years of the reference table.
- **h.Opt**: Window width.

**Method1 function**

**Description**

This function fits the Qxt using method 1 (SMR method, see reference).

**Usage**

Method1(MyData, AgeRange, Plot = F, Color = MyData$Param$Color)

**Arguments**

- **MyData**: The list returned by the AddReference() function.
- **AgeRange**: Age range used for the calculation of the SMR.
- **Plot**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain png plots corresponding to the smoothed surface.
- **Color**: The color that will be used for the plots (HTML notation).
Method2

Method2 function

Description

This function fits the Qxt using method 2 (two parameters relational method, see reference).

Usage

Method2(MyData, AgeRange, Plot = F, Color = MyData$Param$Color)

Arguments

MyData The list returned by the AddReference() function.
AgeRange Age range used for the calculation of the parameters.
Plot If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contains png plots corresponding to the smoothed surface.
Color The color that will be used for the plots (HTML notation).

Method3

Method3 function

Description

This function fits the Qxt using method 3 (Poisson GLM, see reference).

Usage

Method3(MyData, AgeRange, Plot = F, Color = MyData$Param$Color)

Arguments

MyData The list returned by the AddReference() function.
AgeRange Age range used for the calculation of the parameters of the Poisson model.
Plot If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contains png plots corresponding to the smoothed surface.
Color The color that will be used for the plots (HTML notation).
Method4A  

Method4A function

Description
This function fits the Qxt using method 4 (first step) (non-parametric smoothing, see reference).

Usage
Method4A(MyData, AgeRange, AgeCrit, ShowPlot = F)

Arguments
- MyData: The list returned by the AddReference() function.
- AgeRange: Age range used for the construction of the life table.
- AgeCrit: Age range for the comparison of adjusted mortality and observed mortality.
- ShowPlot: AIC plots and plots allowing to judge about the fit.

Method4B  

Method4B function

Description
This function fits the Qxt using method 4 (second step) (non-parametric smoothing, see reference).

Usage
Method4B(PartOne, MyData, OptMale, OptFemale, Plot = F, ShowPlot = F, Color = MyData$Param$Color)

Arguments
- PartOne: The list returned by the Method4A() function.
- MyData: The list returned by the AddReference() function.
- OptMale: Optimal smoothing parameters, obtained from the graphics generated by Method4A() for the male population.
- OptFemale: Optimal smoothing parameters, obtained from the graphics generated by Method4A() for the female population.
- Plot: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain png plots corresponding to the smoothed surface.
- ShowPlot: If true, show plots.
- Color: The color that will be used for the plots (HTML notation).
MyPortfolio

MyPortfolio used for the exemple.

Description

Artificial Portfolio data exemple.

Usage

data(MyPortfolio)

Examples

data(MyPortfolio)

NoCompletion

NoCompletion function

Description

This function allows to keep the adjustment used by the locating method for high ages (for methods 1, 2 or 3).

Usage

NoCompletion(OutputMethod, MyData, Color = MyData$Param$Color, Plot = F, Excel = F)

Arguments

OutputMethod
The list returned by one of these functions : Method1(), Method2(), Method3() or Method4B().

MyData
The list returned by the AddReference() function.

Color
The color that will be used for the plots (HTML notation).

Plot
If TRUE, final mortality surfaces will be saved in Results/Graphics/FinalTables

Excel
If TRUE, final tables will be saved in Results/Excel/FinalTables.xlsx
Description

This function reads a data.frame and calculates exposure and number of deaths. This is the first function the user must call to build a mortality table.

Usage

```r
ReadHistory(MyPortfolio, DateBegObs, DateEndObs, DateFormat, Plot = F,
             Color = "#A4072E", Excel = F)
```

Arguments

- **MyPortfolio**: MyPortfolio is a data.frame of 6 columns as follows: 
  - Id : Id for the line ;
  - Gender : Male or Female ;
  - DateOfBirth : aaaa/mm/jj ;
  - DateIn : aaaa/mm/jj ;
  - DateOut : aaaa/mm/jj ;
  - Status : "other" or "deceased".
- **DateBegObs**: Date for the beginning of the observations.
- **DateEndObs**: Date for the end of the observations.
- **DateFormat**: Date format as expected by the as.Date R function.
- **Plot**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contains png plots corresponding to the smoothed surface.
- **Color**: The color that will be used for the plots (HTML notation).
- **Excel**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contains excel files corresponding to the smoothed surface.

ReferenceFemale

**ReferenceFemale used for the exemple.**

Description

This data corresponds to an adjusted version of the French national demographic projections INSEE 2060 for the female population.

Usage

```r
data(ReferenceFemale)
```

Examples

```r
data(ReferenceFemale)
```
Description

This data corresponds to an adjusted version of the French national demographic projections INSEE 2060 for the male population.

Usage

\texttt{data(ReferenceMale)}

Examples

\texttt{data(ReferenceMale)}

\begin{verbatim}
SurfacePlot xx zexpr mainexpr axis cc

Description

Allows to plot a surface.

Usage

\texttt{SurfacePlot(xx, zexpr, mainexpr, axis, cc)}

Arguments

\begin{verbatim}
xx data as matrix.
zexpr Title of z axis.
mainexpr Name for the graphic.
axis \texttt{c(min(abscissa), max(abscissa), min(ordinate), max(ordinate)).}
cc Color.
\end{verbatim}
ValidationLevel1

**ValidationLevel1 function**

**Description**
This function performs the first level of validation on the returned value of one of these functions: `Method1()`, `Method2()`, `Method3()` or `Method4B()`.

**Usage**
```
ValidationLevel1(OutputMethod, MyData, ValCrit, AgeCrit, Plot = F, 
Color = MyData$Param$Color, Excel = F)
```

**Arguments**
- **OutputMethod**: The list returned by one of these functions: `Method1()`, `Method2()`, `Method3()` or `Method4B()`.
- **MyData**: The list returned by the `AddReference()` function.
- **ValCrit**: Critical value for the comparison of adjusted mortality and observed mortality.
- **AgeCrit**: Age range for the comparison of adjusted mortality and observed mortality.
- **Plot**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain png plots describing the validation analysis.
- **Color**: The color that will be used for the plots (HTML notation).
- **Excel**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain excel files describing the validation analysis.

ValidationLevel2

**ValidationLevel2 function**

**Description**
This function performs the second level of validation on the returned value of one of these functions: `Method1()`, `Method2()`, `Method3()` or `Method4B()` (see reference).

**Usage**
```
ValidationLevel2(OutputMethod, MyData, ValCrit, AgeCrit, Excel = F)
```
Arguments

- **OutputMethod**: The list returned by one of these functions: Method1(), Method2(), Method3() or Method4B().
- **MyData**: The list returned by the AddReference() function.
- **ValCrit**: Critical value for the comparison of adjusted mortality and observed mortality.
- **AgeCrit**: Age range for the comparison of adjusted mortality and observed mortality.
- **Excel**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain excel files describing the validation analysis.

ValidationLevel3 validationLevel3 function

Description

This function performs the third level of validation on the returned value of one of these functions: Method1(), Method2(), Method3() or Method4B().

Usage

```
ValidationLevel3(FinalMethod, MyData, Plot = F, Color = MyData$Param$Color, Excel = F)
```

Arguments

- **FinalMethod**: The list returned by the CompletionB() function.
- **MyData**: The list returned by the AddReference() function.
- **Plot**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain png plots describing the validation analysis.
- **Color**: The color that will be used for the plots (HTML notation).
- **Excel**: If set to TRUE, a sub-directory will be created in the working directory. This sub-directory will contain excel files describing the validation analysis.
Index

*Topic datasets
   MyPortfolio, 19
   ReferenceFemale, 20
   ReferenceMale, 21

*Topic package
   ELT-package, 2

AddReference, 10
ComparisonMethods, 11
CompletionA, 12
CompletionB, 13

Dispersion, 13

ELT (ELT-package), 2
ELT-package, 2

FctMethod1, 14
FctMethod2, 14
FctMethod3, 15
FctMethod4_1stPart, 15
FctMethod4_2ndPart, 16

Method1, 16
Method2, 17
Method3, 17
Method4A, 18
Method4B, 18
MyPortfolio, 19

NoCompletion, 19

ReadHistory, 20
ReferenceFemale, 20
ReferenceMale, 21

SurfacePlot, 21

ValidationLevel1, 22
ValidationLevel2, 22
ValidationLevel3, 23