Package: CvmortalityMult (via r-universe)

August 31, 2024

Type Package

Title Cross-Validation for Multi-Population Mortality Models

Version 1.0.7

Date 2024-08-31

Description Implementation of cross-validation method for testing the forecasting accuracy of several multi-population mortality models. The family of multi-population includes several multi-population mortality models proposed through the actuarial and demography literature. The package includes functions for fitting and forecast the mortality rates of several populations. Additionally, we include functions for testing the forecasting accuracy of different multi-population models. References. Atance, D., Debon, A., and Navarro, E. (2020) <doi:10.3390/math8091550>. Bergmeir, C. & Benitez, J.M. (2012) <doi:10.1016/j.ins.2011.12.028>. Debon, A., Montes, F., & Martinez-Ruiz, F. (2011) <doi:10.1007/s13385-011-0043-z>. Lee, R.D. & Carter, L.R. (1992) <doi:10.1080/01621459.1992.10475265>. Russolillo, M., Giordano, G., & Haberman, S. (2011) <doi:10.1080/03461231003611933>. Santolino, M. (2023) <doi:10.3390/risks11100170>.

License MIT + file LICENSE

URL https://github.com/davidAtance/CvmortalityMult

BugReports https://github.com/davidAtance/CvmortalityMult/issues

LazyLoad yes

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Depends R (>= 3.50)

Imports StMoMo, forecast, gnm (>= 1.1-2), tmap, sf, graphics

Encoding UTF-8

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

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

RemoteUrl https://github.com/davidatance/cvmortalitymult

RemoteRef HEAD

RemoteSha deed8a2a5931b5e71b1e3a19a24508b61a0c4d1b

Contents

---

fitLCmulti   *Function to fit multi-population mortality models*

Description

R function for fitting additive or multiplicative multi-population mortality model developed by: Debon et al. (2011) and Russolillo et al. (2011), respectively. These model follows the structure of the well-known Lee-Carter model (Lee and Carter, 1992) but including an additive or multiplicative parameter to capture the behavior of each population considered. This parameter seeks to capture the individual behavior of every population considered. It should be mentioned that this function is developed for fitting several populations. However, in case you only consider one population, the function will fit the single population version of the Lee-Carter model, the classical one.

Usage

```
fitLCmulti(
  model = c("additive", "multiplicative"),
  qxt,
  periods,
  ages,
  nPop,
  lxt = NULL
)
```

Arguments

model	multi-population mortality model chosen to fit the mortality rates c("additive", "multiplicative"). In case you do not provide any value, the function will apply the "additive" option.
qxt	mortality rates used to fit the additive multipopulation mortality model. This rates can be provided in matrix or in data.frame.
periods	periods considered in the fitting in a vector way c(minyear:maxyear).
ages	vector with the ages considered in the fitting. If the mortality rates provide from an abridged life tables, it is necessary to provide a vector with the ages, see the example.
nPop	number of population considered for fitting.
lxt	survivor function considered for every population, not necessary to provide.

Value

A list with class ″LCmulti″ including different components of the fitting process:

- ax parameter that captures the average shape of the mortality curve in all considered populations.

- bx parameter that explains the age effect x with respect to the general trend kt in the mortality rates of all considered populations.

- kt represent the national tendency of multi-mortality populations during the period.

- Ii gives an idea of the differences in the pattern of mortality in any region i with respect to Region 1.

- formula additive multi-population mortality formula used to fit the mortality rates.

- model provided the model selected in every case.

- data.used mortality rates used to fit the data.

- qxt.real real mortality rates.

- qxt.fitted fitted mortality rates using the additive multi-population mortality model.

- logit.qxt.fitted fitted mortality rates in logit way.

- Ages provided ages to fit the data.

- Periods provided periods to fit the periods.

- nPop provided number of populations to fit the periods.

References

Debon, A., Montes, F., and Martinez-Ruiz, F. (2011). Statistical methods to compare mortality for a group with non-divergent populations: an application to Spanish regions. European Actuarial Journal, 1, 291-308.

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

Russolillo, M., Giordano, G., & Haberman, S. (2011). Extending the Lee–Carter model: a three-way decomposition. Scandinavian Actuarial Journal, 2011(2), 96-117.

See Also

forecast.fitLCmulti, multipopulation_cv, multipopulation_loocv, plot.fitLCmulti

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#First, we present the data that we are going to use
SpainRegions
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)

library(gnm)
library(forecast)
#ADDITIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the additive multi-population mortality model
additive_Spainmales <- fitLCmulti(model = "additive",
                                  qxt = SpainRegions$qx_male,
                                  periods = c(1991:2020),
                                  ages = c(ages),
                                  nPop = 18,
                                  lxt = SpainRegions$lx_male)

additive_Spainmales

#If the user does not provide the model inside the function fitLCmult()
#the multi-population mortality model applied will be additive one.

#Once, we have fit the data, it is possible to see the ax, bx, kt, and Ii
#provided parameters for the fitting.
plot(additive_Spainmales)

#MULTIPLICATIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the multiplicative multi-population mortality model
multiplicative_Spainmales <- fitLCmulti(model = "multiplicative",
                                  qxt = SpainRegions$qx_male,
                                  periods = c(1991:2020),
                                  ages = c(ages),
                                  nPop = 18,
                                  lxt = SpainRegions$lx_male)


multiplicative_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, kt, and It
#provided parameters for the fitting.
plot(multiplicative_Spainmales)

#LEE-CARTER FOR SINGLE-POPULATION
#As we mentioned in the details of the function, if we only provide the data
#from one-population the function fitLCmulti()
```

```
#will fit the Lee-Carter model for single populations.
LC_Spainmales <- fitLCmulti(qxt = SpainNat$qx_male,
                            periods = c(1991:2020),
                            ages = ages,
                            nPop = 1)

LC_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, and kt
#parameters provided for the single version of the LC.
plot(LC_Spainmales)
```

---

forecast.fitLCmulti   *Function to forecast multi-population mortality model*

Description

R function for forecasting additive and multiplicative multi-population mortality model developed
by: Debon et al (2011) and Russolillo et al. (2011), respectively. This model follows the structure of
the well-known Lee-Carter model (Lee and Carter, 1992) but including an additive or multiplicative
parameter to capture the behavior of each population considered. This parameter seeks to capture
the individual behavior of every population considered. It should be mentioned that this function is
developed for fitting several populations. However, in case you only consider one population, the
function will fit the single population version of the Lee-Carter model, the classical one.

Usage

```
## S3 method for class 'fitLCmulti'
forecast(
  object,
  nahead,
  ktmethod = c("Arimapdq", "arima010"),
  kt_include.cte = TRUE,
  ...
)
```

Arguments

object	object "fitLCmulti" developed using function fitLCmulti(). With this object the function will determine the multi-population fitted with the function fitLCmulti().
nahead	number of periods ahead to forecast.
ktmethod	method used to forecast the value of kt Arima(p,d,q) or ARIMA(0,1,0); c("Arimapdq", "arima010").
kt_include.cte	if you want that kt include constant in the arima process.
...	other arguments for [iarima]().

Value

A list with class `"forLCmulti"` including different components of the forecasting process:

- `ax` parameter that captures the average shape of the mortality curve in all considered populations.
- `bx` parameter that explains the age effect x with respect to the general trend `kt` in the mortality rates of all considered populations.
- `arimakt` the arima selected for the `kt` time series.
- `kt.fitted` obtained values for the tendency behavior captured by `kt`.
- `kt.fut` projected values of `kt` for the nahead periods ahead.
- `kt.futintervals` arima selected and future values of `kt` with the different intervals, lower and upper, 80\
- `Ii` parameter that captures the differences in the pattern of mortality in any region i with respect to Region 1.
- `ktmethod` method selected to forecast the value of `kt` Arima(p,d,q) or ARIMA(0,1,0); c("Arimapdq", "arima010").
- `kt_include.cte` the decision regarding the inclusion of constant in the `kt` arima process.
- `formula` additive multi-population mortality formula used to fit the mortality rates.
- `model` provided the model selected in every case.
- `qxt.real` real mortality rates.
- `qxt.fitted` fitted mortality rates using the additive multi-population mortality model.
- `logit.qxt.fitted` fitted mortality rates in logit way estimated with the additive multi-population mortality model.
- `qxt.future` future mortality rates estimated with the additive multi-population mortality model.
- `logit.qxt.future` future mortality rates in logit way estimated with the additive multi-population mortality model.
- `nPop` provided number of populations to fit the periods.

References

Debon, A., Montes, F., & Martinez-Ruiz, F. (2011). Statistical methods to compare mortality for a group with non-divergent populations: an application to Spanish regions. European Actuarial Journal, 1, 291-308.

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

Russolillo, M., Giordano, G., & Haberman, S. (2011). Extending the Lee–Carter model: a three-way decomposition. Scandinavian Actuarial Journal, 2011(2), 96-117.

See Also

[fitLCmulti](), [plot.fitLCmulti](), [plot.forLCmulti](), [multipopulation_cv](), [multipopulation_loocv](), [iarima]()

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#First, we present the data that we are going to use
SpainRegions
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)

library(gnm)
library(forecast)
#ADDITIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the additive multi-population mortality model
additive_Spainmales <- fitLCmulti(model = "additive",
                                  qxt = SpainRegions$qx_male,
                                  periods = c(1991:2020),
                                  ages = c(ages),
                                  nPop = 18,
                                  lxt = SpainRegions$lx_male)

additive_Spainmales

#If the user does not provide the model inside the function fitLCmult()
#the multi-population mortality model applied will be additive one.

#Once, we have fit the data, it is possible to see the ax, bx, kt, and Ii
#provided parameters for the fitting.
plot(additive_Spainmales)

#Once, we have fit the data, it is possible to forecast the multipopulation
#mortality model several years ahead, for example 10, as follows:
fut_additive_Spainmales <- forecast(object = additive_Spainmales, nahead = 10,
                                    ktmethod = "Arimapdq", kt_include.cte = TRUE)

fut_additive_Spainmales
#Once the data have been adjusted, it is possible to display the fitted kt and
#its out-of-sample forecasting. In addition, the function shows
#the logit mortality adjusted in-sample and projected out-of-sample
#for the mean age of the data considered in all populations.
plot(fut_additive_Spainmales)

#MULTIPLICATIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the multiplicative multi-population mortality model
multiplicative_Spainmales <- fitLCmulti(model = "multiplicative",
                                  qxt = SpainRegions$qx_male,
                                  periods = c(1991:2020),
                                  ages = c(ages),
                                  nPop = 18,
                                  lxt = SpainRegions$lx_male)

multiplicative_Spainmales
```

```
#Once, we have fit the data, it is possible to see the ax, bx, kt, and It
#provided parameters for the fitting.
plot(multiplicative_Spainmales)

#Once, we have fit the data, it is possible to forecast the multipopulation
#mortality model several years ahead, for example 10, as follows:
fut_multi_Spainmales <- forecast(object = multiplicative_Spainmales, nahead = 10,
                                 ktmethod = "Arimapdq", kt_include.cte = TRUE)

fut_multi_Spainmales
#Once the data have been adjusted, it is possible to display the fitted kt and
#its out-of-sample forecasting. In addition, the function shows
#the logit mortality adjusted in-sample and projected out-of-sample
#for the mean age of the data considered in all populations.
plot(fut_multi_Spainmales)

#LEE-CARTER FOR SINGLE-POPULATION
#As we mentioned in the details of the function, if we only provide the data
#from one-population the function fitLCmulti()
#will fit the Lee-Carter model for single populations.
LC_Spainmales <- fitLCmulti(qxt = SpainNat$qx_male,
                            periods = c(1991:2020),
                            ages = ages,
                            nPop = 1)

LC_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, and kt
#parameters provided for the single version of the LC.
plot(LC_Spainmales)

#Once, we have fit the data, it is possible to forecast the multipopulation
#mortality model several years ahead, for example 10, as follows:
fut_LC_Spainmales <- forecast(object = LC_Spainmales, nahead = 10,
                              ktmethod = "Arimapdq", kt_include.cte = TRUE)

#Once the data have been adjusted, it is possible to display the fitted kt and
#its out-of-sample forecasting. In addition, the function shows
#the logit mortality adjusted in-sample and projected out-of-sample
#for the mean age of the data considered in all populations.
plot(fut_LC_Spainmales)
```

MeasureAccuracy   *Measures of Accuracy*

Description

R function to estimate different measures of accuracy.

1. the sum of squared errors (SSE) for the mortality rates:

$$\sum_x \sum_t (qxt1 - qxt2)^2$$

where qxt1 is the real mortality rates `qxt_re`, and qxt2 is the adjusted mortality rates `qxt_aju`.

2. The mean squared errors (MSE) for the mortality rates:

$$\frac{1}{n} \sum_x \sum_t (qxt1 - qxt2)^2 = \frac{1}{n} SSE$$

where qxt1 is the real mortality rates `qxt_re`, and qxt2 is the adjusted mortality rates `qxt_aju`.

3. The mean absolute errors (MAE) for the mortality rates:

$$\frac{1}{n} \sum_x \sum_t |qx1 - qxt2|$$

. where qxt1 is the real mortality rates `qxt_re`, and qxt2 is the adjusted mortality rates `qxt_aju`.

4. The mean absolute percentage error (MAPE) for the mortality rates:

$$\frac{1}{n} \sum_x \sum_t \left| \frac{(qxt1 - qxt2)}{qxt2} \right|$$

where qxt1 is the real mortality rates `qxt_re`, and qxt2 is the adjusted mortality rates `qxt_aju`. You only have to provide the real value, the fitted or forecasted value for your mortality rates and the measure of accuracy chosen. However, the function is constructed to provide the real value and the fitted or forecasted value of your independent variable. These variables must have the same dimensions to be compared.

Usage

```
MeasureAccuracy(
  measure = c("SSE", "MSE", "MAE", "MAPE", "All"),
  qxt_re,
  qxt_aju,
  wxt
)
```

Arguments

measure	choose the non-penalized measure of accuracy that you want to use; c("SSE", "MSE", "MAE", "MAPE", "All"). Check the function. In case you do not provide any value, the function will apply the "SSE" as measure of forecasting accuracy.
qxt_re	real mortality rates used to check the goodness of fit measure.
qxt_aju	adjusted mortality rates using a specific mode.
wxt	weights of the mortality rates or data provided.

Value

An object with class ″MoA″ including the value of the measure of accuracy for the data provided.

References

Atance, D., Debón, A., & Navarro, E. (2020). A comparison of forecasting mortality models using resampling methods. Mathematics, 8(9), 1550.

See Also

fitLCmulti, forecast.fitLCmulti, multipopulation_cv, multipopulation_loocv.

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#To show how the function works, we need to provide fitted or forecasted data and the real data.
#In this case, we employ the following data of the library:
SpainRegions

library(gnm)
library(forecast)
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)
#In this case, we fit for males providing the lxt
multiplicative_Spainmales <- fitLCmulti(model = ″multiplicative″,
                                qxt = SpainRegions$qx_male,
                                periods = c(1991:2020),
                                ages = c(ages),
                                nPop = 18,
                                lxt = SpainRegions$lx_male)

multiplicative_Spainmales
plot(multiplicative_Spainmales)

#Once, we have the fitted data, we will obtain different measures of accuracy
#for the first population.
#We need to obtain wxt (weight of the mortality rates or data provided) using a
library(StMoMo)
wxt_1pob <- genWeightMat(ages = ages, years = c(1991:2020), clip = 0)

###########################
#SSE#
###########################
SSE_multSpmales <- MeasureAccuracy(measure = ″SSE″,
                        qxt_re = multiplicative_Spainmales$qxt.real$pob1,
                        qxt_aju = multiplicative_Spainmales$qxt.fitted$pob1,
                        wxt = wxt_1pob)
SSE_multSpmales
###########################
#MSE#
```

```
###########################
MSE_multSpmales <- MeasureAccuracy(measure = "MSE",
                        qxt_re = multiplicative_Spainmales$qxt.real$pob1,
                        qxt_aju = multiplicative_Spainmales$qxt.fitted$pob1,
                        wxt = wxt_1pob)
MSE_multSpmales
###########################
#MAE#
###########################
MAE_multSpmales <- MeasureAccuracy(measure = "MSE",
                        qxt_re = multiplicative_Spainmales$qxt.real$pob1,
                        qxt_aju = multiplicative_Spainmales$qxt.fitted$pob1,
                        wxt = wxt_1pob)
MAE_multSpmales
###########################
#MAPE#
###########################
MAPE_multSpmales <- MeasureAccuracy(measure = "MSE",
                        qxt_re = multiplicative_Spainmales$qxt.real$pob1,
                        qxt_aju = multiplicative_Spainmales$qxt.fitted$pob1,
                        wxt = wxt_1pob)
MAPE_multSpmales
```

---

multipopulation_cv	*Function to apply cross-validation techniques for testing the forecasting accuracy of multi-population mortality models*

Description

R function for testing the accuracy out-of-sample of different multi-population mortality models, Additive (Debon et al., 2011) and Multiplicative (Russolillo et al., 2011). We provide a R function that employ the cross-validation techniques for data panel-time series (Atance et al. 2020) to test the forecasting accuracy. These techniques consist on split the database in two parts: training set (to run the model) and test set (to check the forecasting accuracy of the model). This procedure is repeated several times trying to check the forecasting accuracy in different ways. With this function, the user can provide its own mortality rates for different populations. The function will split the database chronologically (Bergmeir and Benitez, 2012) based on the nahead which consist on the length of the training set. We have include the following Figure 1 to understand how the R function works. " alt" alt It should be mentioned that this function is developed for cross-validation the forecasting accuracy of several populations. However, in case you only consider one population, the function will forecast the Lee-Carter model for one population. To test the forecasting accuracy of the selected model, the function provides five different measures: SSE, MSE, MAE, MAPE or All. Depending on how you want to check the forecasting accuracy of the model you could select one or other. In this case, the measures will be obtained using the mortality rates in the normal scale as recommended by Santolino (2023) against the log scale.

Usage

```
multipopulation_cv(
  qxt,
  model = c("additive", "multiplicative"),
  periods,
  ages,
  nPop,
  lxt = NULL,
  nahead,
  ktmethod = c("Arimapdq", "arima010"),
  kt_include.cte = TRUE,
  measures = c("SSE", "MSE", "MAE", "MAPE", "All")
)
```

Arguments

qxt	mortality rates used to fit the multi-population mortality models. This rates can be provided in matrix or in data.frame.
model	multi-population mortality model chosen to fit the mortality rates c("additive", "multiplicative"). In case you do not provide any value, the function will apply the "additive" option.
periods	periods considered in the fitting in a vector way c(minyear:maxyear).
ages	vector with the ages considered in the fitting. If the mortality rates provide from an abridged life tables, it is necessary to provide a vector with the ages, see the example.
nPop	number of population considered for fitting.
lxt	survivor function considered for every population, not necessary to provide.
nahead	is a vector specifying the number of periods to block in the blocked CV. The function operates by using the sum of the periods in nahead and three (the minimum number of years required to construct a time series), as the initial training set. This ensures that the first train set has sufficient observations to forecast the initial test set, which will be of length nahead.
ktmethod	method used to forecast the value of kt Arima(p,d,q) or ARIMA(0,1,0); c("Arimapdq", "arima010").
kt_include.cte	if you want that kt include constant in the arima process.
measures	choose the non-penalized measure of forecasting accuracy that you want to use; c("SSE", "MSE", "MAE", "MAPE", "All"). Check the function. In case you do not provide any value, the function will apply the "SSE" as measure of forecasting accuracy.

Value

An object of the class "MultiCv" including a list() with different components of the cross-validation process:

- ax parameter that captures the average shape of the mortality curve in all considered populations.

- bx parameter that explains the age effect x with respect to the general trend kt in the mortality rates of all considered populations.
- kt.fitted obtained values for the tendency behavior captured by kt .
- kt.future future values of kt for every iteration in the cross-validation.
- kt.arima the arima selected for each kt time series.
- Ii parameter that captures the differences in the pattern of mortality in any region i with respect to Region 1.
- formula multi-population mortality formula used to fit the mortality rates.
- model provided the model selected in every case.
- nPop provided number of populations to fit the periods.
- qxt.real real mortality rates.
- qxt.future future mortality rates estimated with the multi-population mortality model.
- logit.qxt.future future mortality rates in logit way estimated with the multi-population mortality model.
- meas_ages measure of forecasting accuracy through the ages of the study.
- meas_periodsfut measure of forecasting accuracy in every forecasting period(s) of the study.
- meas_pop measure of forecasting accuracy through the populations considered in the study.
- meas_total a global measure of forecasting accuracy through the ages, periods and populations of the study.

References

Atance, D., Debon, A., and Navarro, E. (2020). A comparison of forecasting mortality models using resampling methods. Mathematics 8(9): 1550.

Bergmeir, C. & Benitez, J.M. (2012) On the use of cross-validation for time series predictor evaluation. Information Sciences, 191, 192–213.

Debon, A., & Atance, D. (2022). Two multi-population mortality models: A comparison of the forecasting accuracy with resampling methods. in Contributions to Risk Analysis: Risk 2022. Fundacion Mapfre

Debon, A., Montes, F., & Martinez-Ruiz, F. (2011). Statistical methods to compare mortality for a group with non-divergent populations: an application to Spanish regions. European Actuarial Journal, 1, 291-308.

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

Russolillo, M., Giordano, G., & Haberman, S. (2011). Extending the Lee–Carter model: a three-way decomposition. Scandinavian Actuarial Journal, 96-117.

Santolino, M. (2023). Should Selection of the Optimum Stochastic Mortality Model Be Based on the Original or the Logarithmic Scale of the Mortality Rate?. Risks, 11(10), 170.

See Also

multipopulation_loocv, fitLCmulti, forecast.fitLCmulti, plot.fitLCmulti, plot.forLCmulti, MeasureAccuracy.

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#We present a cross-validation method for spanish male regions

ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40,
          45, 50, 55, 60, 65, 70, 75, 80, 85, 90)
library(gnm)
library(forecast)
#Let start with a simple nahead=5 CV method obtaining the SSE forecasting measure of accuracy
cv_Spainmales_addit <- multipopulation_cv(qxt = SpainRegions$qx_male,
                                          model = c("additive"),
                                          periods =  c(1991:2020), ages = c(ages),
                                          nPop = 18, lxt = SpainRegions$lx_male,
                                          nahead = 5,
                                          ktmethod = c("Arimapdq"),
                                          kt_include.cte = TRUE,
                                          measures = c("SSE"))
cv_Spainmales_addit

#Once, we have run the function we can check the result in different ways:
cv_Spainmales_addit$meas_ages
cv_Spainmales_addit$meas_periodsfut
cv_Spainmales_addit$meas_pop
cv_Spainmales_addit$meas_total
```

---

multipopulation_loocv   *Function to apply Leave-One-Out Cross-Validation (LOOCV) technique for testing the forecasting accuracy of multi-population mortality models*

Description

R function for testing the accuracy out-of-sample of different multi-population mortality models, Additive (Debon et al., 2011) and Multiplicative (Russolillo et al., 2011). We provide a R function that employ the leave-one-out cross-validation technique for data panel-time series (Atance et al. 2020) to test the forecasting accuracy of one-multipopulation mortality model. This technique consists on split the database in two parts: training set (to run the model) and test set (to check the forecasting accuracy of the model) with only one data (one period in this case). This procedure is repeated several times trying to check the forecasting accuracy in different ways enlarging the training set one period-ahead. With this function, the user can provide its own mortality rates for different populations. The function will split the database chronologically (Bergmeir and Benitez, 2012) based on the trainset1 which consist on the length of the first training set. We have include the following Figure 2 to understand how the R function works. " alt" alt It should be mentioned that this function is developed for testing the the forecasting accuracy of several populations using

leave-one-out cross-validation . However, in case you only consider one population, the function will forecast the Lee-Carter model for one population. To test the forecasting accuracy of the selected model, the function provides five different measures: SSE, MSE, MAE, MAPE or All. Depending on how you want to check the forecasting accuracy of the model you could select one or other. In this case, the measures will be obtained using the mortality rates in the normal scale as recommended by Santolino (2023) against the log scale.

Usage

```
multipopulation_loocv(
  qxt,
  model = c("additive", "multiplicative"),
  periods,
  ages,
  nPop,
  lxt = NULL,
  ktmethod = c("Arimapdq", "arima010"),
  kt_include.cte = TRUE,
  measures = c("SSE", "MSE", "MAE", "MAPE", "All"),
  trainset1
)
```

Arguments

qxt	mortality rates used to fit the multi-population mortality models. This rates can be provided in matrix or in data.frame.
model	multi-population mortality model chosen to fit the mortality rates c("additive", "multiplicative"). In case you do not provide any value, the function will apply the "additive" option.
periods	periods considered in the fitting in a vector way c(minyear:maxyear).
ages	vector with the ages considered in the fitting. If the mortality rates provide from an abridged life tables, it is necessary to provide a vector with the ages, see the example.
nPop	number of population considered for fitting.
lxt	survivor function considered for every population, not necessary to provide.
ktmethod	method used to forecast the value of kt Arima(p,d,q) or ARIMA(0,1,0); c("Arimapdq", "arima010").
kt_include.cte	if you want that kt include constant in the arima process.
measures	choose the non-penalized measure of forecasting accuracy that you want to use; c("SSE", "MSE", "MAE", "MAPE", "All"). Check the function. In case you do not provide any value, the function will apply the "SSE" as measure of forecasting accuracy.
trainset1	vector with the periods for the first training set. This value must be greater than 2 to meet the minimum time series size (Hyndman and Khandakar, 2008).

Value

A list with class "MultiCv" including different components of the cross-validation process:

- ax parameter that captures the average shape of the mortality curve in all considered populations.
- bx parameter that explains the age effect x with respect to the general trend kt in the mortality rates of all considered populations.
- kt.fitted obtained values for the tendency behavior captured by kt.
- kt.future future values of kt for every iteration in the cross-validation.
- kt.arimathe arima selected for each kt time series.
- Ii parameter that captures the differences in the pattern of mortality in any region i with respect to Region 1.
- formula multi-population mortality formula used to fit the mortality rates.
- nPop provided number of populations to fit the periods.
- qxt.real real mortality rates.
- qxt.future future mortality rates estimated with the multi-population mortality model.
- logit.qxt.future future mortality rates in logit way estimated with the multi-population mortality model.
- meas_ages measure of forecasting accuracy through the ages of the study.
- meas_periodsfut measure of forecasting accuracy in every forecasting period(s) of the study.
- meas_pop measure of forecasting accuracy through the populations considered in the study.
- meas_total a global measure of forecasting accuracy through the ages, periods and populations of the study.

References

Atance, D., Debon, A., and Navarro, E. (2020). A comparison of forecasting mortality models using resampling methods. Mathematics 8(9): 1550.

Bergmeir, C. & Benitez, J.M. (2012) On the use of cross-validation for time series predictor evaluation. Information Sciences, 191, 192–213.

Debon, A., & Atance, D. (2022). Two multi-population mortality models: A comparison of the forecasting accuracy with resampling methods. in Contributions to Risk Analysis: Risk 2022. Fundacion Mapfre

Debon, A., Montes, F., & Martinez-Ruiz, F. (2011). Statistical methods to compare mortality for a group with non-divergent populations: an application to Spanish regions. European Actuarial Journal, 1, 291-308.

Hyndman, R.J. & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. Journal of Statistical. Software, 26, 1–22.

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

Russolillo, M., Giordano, G., & Haberman, S. (2011). Extending the Lee–Carter model: a three-way decomposition. Scandinavian Actuarial Journal, 96-117.

Santolino, M. (2023). Should Selection of the Optimum Stochastic Mortality Model Be Based on the Original or the Logarithmic Scale of the Mortality Rate?. Risks, 11(10), 170.

See Also

multipopulation_cv, fitLCmulti, forecast.fitLCmulti, plot.fitLCmulti, plot.forLCmulti, MeasureAccuracy.

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#We present the leave-one-out cross-validation (LOOCV) method for spanish male regions
#The idea is to get the same results as in the short paper published in Risk Congress 2023
SpainRegions
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40,
          45, 50, 55, 60, 65, 70, 75, 80, 85, 90)
library(gnm)
library(forecast)
#Let start with a simple trainset1 = 10 CV method obtaining the SSE forecasting measure of accuracy
loocv_Spainmales_addit <- multipopulation_loocv(qxt = SpainRegions$qx_male,
                                       model = c("additive"),
                                       periods =  c(1991:2020), ages = c(ages),
                                       nPop = 18, lxt = SpainRegions$lx_male,
                                       ktmethod = c("Arimapdq"),
                                       kt_include.cte = TRUE,
                                       measures = c("SSE"),
                                       trainset1 = 10)
loocv_Spainmales_addit

#Once, we have run the function we can check the result in different ways:
loocv_Spainmales_addit$meas_ages
loocv_Spainmales_addit$meas_periodsfut
loocv_Spainmales_addit$meas_pop
loocv_Spainmales_addit$meas_total
```

---

plot.fitLCmulti	*Function to plot the parameters of the multi-population mortality models*

Description

R function to plot the parameters for the Additive (Debon et al., 2011) and Multiplicative (Russolillo et al., 2011) Multi-Population mortality model. It should be mentioned that in case that this function is developed for fitting several populations. However, in case you only consider one population, the function will fit the one-population Lee-Carter model (Lee and Carter, 1992).

Usage

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

Arguments

x   x developed using function `fitLCmulti()` which are objects of the `fitLCmulti` class.

...   additional arguments to show in the plot appearance.

Value

plot the different parameters for the multi-population mortality models ax, bx, kt and Ii. This function is valid for both approaches Additive and Multiplicative multi-population mortality models.

References

Debon, A., Montes, F., & Martinez-Ruiz, F. (2011). Statistical methods to compare mortality for a group with non-divergent populations: an application to Spanish regions. European Actuarial Journal, 1, 291-308.

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

Multi-population mortality model developed by: Russolillo, M., Giordano, G., & Haberman, S. (2011). Extending the Lee–Carter model: a three-way decomposition. Scandinavian Actuarial Journal, 2011(2), 96-117.

See Also

[fitLCmulti](), [forecast.fitLCmulti](), [plot.forLCmulti](), [multipopulation_cv](), [multipopulation_loocv]()

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#First, we present the data that we are going to use
SpainRegions
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)

library(gnm)
library(forecast)
#ADDITIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the additive multi-population mortality model
additive_Spainmales <- fitLCmulti(model = "additive",
                          qxt = SpainRegions$qx_male,
                          periods = c(1991:2020),
                          ages = c(ages),
                          nPop = 18,
                          lxt = SpainRegions$lx_male)

additive_Spainmales

#If the user does not provide the model inside the function fitLCmult()
```

```
#the multi-population mortality model applied will be additive one.

#Once, we have fit the data, it is possible to see the ax, bx, kt, and Ii
#provided parameters for the fitting.
plot(additive_Spainmales)

#MULTIPLICATIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the multiplicative multi-population mortality model
multiplicative_Spainmales <- fitLCmulti(model = "multiplicative",
                                qxt = SpainRegions$qx_male,
                                periods = c(1991:2020),
                                ages = c(ages),
                                nPop = 18,
                                lxt = SpainRegions$lx_male)

multiplicative_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, kt, and It
#provided parameters for the fitting.
plot(multiplicative_Spainmales)

#LEE-CARTER FOR SINGLE-POPULATION
#As we mentioned in the details of the function, if we only provide the data
#from one-population the function fitLCmulti()
#will fit the Lee-Carter model for single populations.
LC_Spainmales <- fitLCmulti(qxt = SpainNat$qx_male,
                                periods = c(1991:2020),
                                ages = ages,
                                nPop = 1)

LC_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, and kt
#parameters provided for the single version of the LC.
plot(LC_Spainmales)
```

---

plot.forLCmulti   *Function to plot the parameters of the multi-population mortality mod-*
                           els

Description

Function to plot different results of the forecasting process of multi-population mortality models, the additive (Debon et al., 2011) and the multiplicative (Russolillo et al., 2011), obtained using the forecast.fitLCmulti function which are xs of the forecastLCmulti class. In fact, the function will show the trend parameter kt fitted for the in-sample periods and its forecast results. Similarly, the behavior of the logit mortality rate for the mean in-sample age and the out-of-sample forecast

will be shown for all the populations considered. It should be mentioned that this function is developed for fitting several populations. However, in case you only consider one population, the function will show the single population version of the Lee-Carter model, the classical one.

Usage

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

Arguments

x   x developed using function `forecast.fitLCmulti()` which are objects of the
                `fortLCmulti` class.

...   additional arguments to show in the plot appearance.

Value

plot the trend parameter kt fitted for the in-sample periods and its forecast results for the multi-population mortality models. Similarly, the behavior of the logit mortality rate for the mean in-sample age and the out-of-sample forecast will be shown for all the populations considered.

References

Debon, A., Montes, F., & Martinez-Ruiz, F. (2011). Statistical methods to compare mortality for a group with non-divergent populations: an application to Spanish regions. European Actuarial Journal, 1, 291-308.

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

Multi-population mortality model developed by: Russolillo, M., Giordano, G., & Haberman, S. (2011). Extending the Lee–Carter model: a three-way decomposition. Scandinavian Actuarial Journal, 2011(2), 96-117.

See Also

[fitLCmulti](), [forecast.fitLCmulti](), [plot.fitLCmulti](), [multipopulation_cv](), [multipopulation_loocv]()

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#First, we present the data that we are going to use
SpainRegions
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)

library(gnm)
library(forecast)
#ADDITIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the additive multi-population mortality model
```

```
additive_Spainmales <- fitLCmulti(model = "additive",
                              qxt = SpainRegions$qx_male,
                              periods = c(1991:2020),
                              ages = c(ages),
                              nPop = 18,
                              lxt = SpainRegions$lx_male)

additive_Spainmales

#If the user does not provide the model inside the function fitLCmult()
#the multi-population mortality model applied will be additive one.

#Once, we have fit the data, it is possible to see the ax, bx, kt, and Ii
#provided parameters for the fitting.
plot(additive_Spainmales)

#Once, we have fit the data, it is possible to forecast the multipopulation
#mortality model several years ahead, for example 10, as follows:
fut_additive_Spainmales <- forecast(object = additive_Spainmales, nahead = 10,
                                ktmethod = "Arimapdq", kt_include.cte = TRUE)

fut_additive_Spainmales
#Once the data have been adjusted, it is possible to display the fitted kt and
#its out-of-sample forecasting. In addition, the function shows
#the logit mortality adjusted in-sample and projected out-of-sample
#for the mean age of the data considered in all populations.
plot(fut_additive_Spainmales)

#MULTIPLICATIVE MULTI-POPULATION MORTALITY MODEL
#In the case, the user wants to fit the multiplicative multi-population mortality model
multiplicative_Spainmales <- fitLCmulti(model = "multiplicative",
                              qxt = SpainRegions$qx_male,
                              periods = c(1991:2020),
                              ages = c(ages),
                              nPop = 18,
                              lxt = SpainRegions$lx_male)

multiplicative_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, kt, and It
#provided parameters for the fitting.
plot(multiplicative_Spainmales)

#Once, we have fit the data, it is possible to forecast the multipopulation
#mortality model several years ahead, for example 10, as follows:
fut_multi_Spainmales <- forecast(object = multiplicative_Spainmales, nahead = 10,
                              ktmethod = "Arimapdq", kt_include.cte = TRUE)

fut_multi_Spainmales
#Once the data have been adjusted, it is possible to display the fitted kt and
#its out-of-sample forecasting. In addition, the function shows
#the logit mortality adjusted in-sample and projected out-of-sample
#for the mean age of the data considered in all populations.
```

```
plot(fut_multi_Spainmales)

#LEE-CARTER FOR SINGLE-POPULATION
#As we mentioned in the details of the function, if we only provide the data
#from one-population the function fitLCmulti()
#will fit the Lee-Carter model for single populations.
LC_Spainmales <- fitLCmulti(qxt = SpainNat$qx_male,
                            periods = c(1991:2020),
                            ages = ages,
                            nPop = 1)

LC_Spainmales

#Once, we have fit the data, it is possible to see the ax, bx, and kt
#parameters provided for the single version of the LC.
plot(LC_Spainmales)

#Once, we have fit the data, it is possible to forecast the multipopulation
#mortality model several years ahead, for example 10, as follows:
fut_LC_Spainmales <- forecast(object = LC_Spainmales, nahead = 10,
                              ktmethod = "Arimapdq", kt_include.cte = TRUE)

#Once the data have been adjusted, it is possible to display the fitted kt and
#its out-of-sample forecasting. In addition, the function shows
#the logit mortality adjusted in-sample and projected out-of-sample
#for the mean age of the data considered in all populations.
plot(fut_LC_Spainmales)
```

---

regions	*regions*

Description

Data from the Spanish region of Spain which are provided to plot an indicator. This dataset contains a plot with the information of Spain regions (geometry and name of every region).

Usage

```
regions
```

Format

A data frame with 600 rows and 9 columns with class "SpainRegionsData" including the following information

- Codigo a vector containing the code of every region of Spain.
- Texto a vector containing the name of every region of Spain.

- Texto_Alt a vector containing the long name of every region of Spain.

- Ii a vector containing a possible value of one indicator to be shown.

- geometry the dimension of every region of Spain. This vector allows to plot the regions of Spain.

Value

a plot with the Spain regions colored by the indicator provided.

References

Spanish National Institute of Statistics (INE) (2023). Tablas de mortalidad, metodologia. Technical report, Instituto Nacional de Estadistica

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#In this case, we show the region dataset applying it to a multipopulation model.
#First, we present the dataset

regions

ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)

#Then, we fit the multiplicative model
library(gnm)
multiplicative_Spainmales <- fitLCmulti(model = "multiplicative",
                                        qxt = SpainRegions$qx_male,
                                        periods = c(1991:2020),
                                        ages = c(ages),
                                        nPop = 18,
                                        lxt = SpainRegions$lx_male)

multiplicative_Spainmales
library(sf)
#To show the values of the population indicator in the Spanish map.
SpainMap(regionvalue = multiplicative_Spainmales$Ii[2:18],
         main = c("Multiplicative for males"),
         name = c("Ii"))
```

---

residSVD2   *Function to extract the resid from SVD*

Description

This function uses the first d components of the singular value decomposition in order to approximate a vector of model residuals by a sum of d multiplicative terms, with the multiplicative structure determined by two specified factors follows the SVD function by Turner et al. (2023). For glm and gnm models from the gnm R-packages, the matrix entries are weighted working residuals. The primary use of residSVD is to generate good starting values for the parameters in Mult terms in models to be fitted using gnm. In this case, we modified the function in order to obtain good starting values for the multi-population mortality models.

Usage

```
residSVD2(model, fac1, fac2, d = 1)
```

Arguments

model	object with na.action, residuals, and weights methods, e.g. objects inheriting from class "gnm".
fac1	first factor.
fac2	second factor.
d	integer, the number of multiplicative terms to use in the approximation.

Value

If d = 1, a numeric vector; otherwise a numeric matrix with d columns.

References

Turner, H., & Firth, D. (2023). Generalized nonlinear models in R: An overview of the gnm package. R package version 1.1-5. https://CRAN.R-project.org/package=gnm

See Also

fitLCmulti, forecast.fitLCmulti multipopulation_cv, multipopulation_loocv, plot.fitLCmulti

---

SpainMap   *Spain National map information*

Description

This data contains information to plot the percentiles plot in Spanish regions. Therefore, the users only have to provide a specific variable to show in regions of Spain.

Usage

```
SpainMap(regionvalue, main, name)
```

Arguments

regionvalue   vector with the values that you want to plot in percentiles in the Spain map.

main   the specific title of the map plot

name   the assigned name for the legend in map plot.

Value

a map from the regions of Spain colored with the variable provided by the user.

References

Spanish National Institute of Statistics (INE) (2023). Tablas de mortalidad, metodologia. Technical report, Instituto Nacional de Estadistica

Examples

```
name <- c("Ii")
main <- c("Multiplicative for males")
regionvalue <- c(0.131867619, -0.063994652,  0.088094096,
                 0.019685552,  0.064671498,   0.012212161,
                -0.088864474, -0.146079884, -0.017703536,
                 0.050376721,  0.052476852, -0.022871202,
                -0.093952332,  0.049266816, -0.101224890,
                 0.001481333, -0.078523511)
library(sf)

SpainMap(regionvalue, main, name)
```

SpainNat   *Spain National Mortality data*

Description

Data from the Spanish national of Spain from the Spanish National Institute of Statistics (INE) for both genders years 1991-2020 and abridged ages from 0 to 90. This dataset contains mortality rates for the total national population of Spain. Additionally, the dataset includes the number of people alive (lxt) for each age and period.

Usage

```
SpainNat
```

Format

A data frame with 600 rows and 9 columns with class "CVmortalityData" including the following information

- ccaa a vector containing all the regions of Spain. Indeed, the column takes the following information: Spain.

- years a vector containing the periods of the dataset from 1991 to 2020.

- ages a vector containing the abridged ages considered in the dataset, 0, <1, 1-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59, 60-64, 65-69, 70-74, 75-79, 80-84, 85-89, and 90-94.

- qx_male mortality rates for the males in the Spain Nation.

- qx_female mortality rates for the females in the Spain Nation.

- lx_male survivor function considered for the males of Spain Nation.

- lx_female survivor function considered for the females of Spain Nation.

- series information for the series of data provided.

- label the assigned tag to the data frame.

References

Spanish National Institute of Statistics (INE) (2023). Tablas de mortalidad, metodologia. Technical report, Instituto Nacional de Estadistica

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#In this case, we show the region dataset applying it to a multipopulation model.
#First, we present the dataset
SpainNat
#An example to how the additive multi-population model fits the data
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)
library(gnm)
LC_Spainmales <- fitLCmulti(qxt = SpainNat$qx_male,
                            periods = c(1991:2020),
                            ages = ages,
                            nPop = 1)

LC_Spainmales
```

SpainRegions   *Spain Regions Mortality data*

Description

Data from the Spanish region of Spain from the Spanish National Institute of Statistics (INE) for both genders years 1991-2020 and abridged ages from 0 to 90. This dataset contains mortality rates (qxt) from 18 different regions of Spain. Additionally, the dataset includes the number of people alive (lxt) for each age and period.

Usage

```
SpainRegions
```

Format

A data frame with 10800 rows and 9 columns with class "CVmortalityData" including the following information

- ccaa a vector containing all the regions of Spain. Indeed, the column takes the following information: Spain, Andalucia, Aragon, Asturias, Baleares, Canarias, Cantabria, Castillayla Mancha, CastillayLeon, Cataluna, ComunidadValenciana, Extremadura, Galicia, Madrid, Murcia, Navarra, PaisVasco, and LaRioja.

- years a vector containing the periods of the dataset from 1991 to 2020.

- ages a vector containing the abridged ages considered in the dataset, 0, <1, 1-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59, 60-64, 65-69, 70-74, 75-79, 80-84, 85-89, and 90-94.

- qx_male mortality rates for the males in every region of Spain including Nation data.

- qx_female mortality rates for the females in every region of Spain including Nation data.

- lx_male survivor function considered for the males in every region of Spain including Nation data.

- lx_female survivor function considered for the females in every region of Spain including Nation data.

- series information for the series of data provided.

- label the assigned tag to the data frame.

References

Spanish National Institute of Statistics (INE) (2023). Tablas de mortalidad, metodologia. Technical report, Instituto Nacional de Estadistica

Examples

```
#The example takes more than 5 seconds because it includes
#several fitting and forecasting process and hence all
#the process is included in donttest

#In this case, we show the region dataset applying it to a multipopulation model.
#First, we present the dataset
SpainRegions

#An example to how the additive multi-population model fits the data
ages <- c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90)

library(gnm)
multiplicative_Spainmales <- fitLCmulti(model = "multiplicative",
                                        qxt = SpainRegions$qx_male,
                                        periods = c(1991:2020),
                                        ages = c(ages),
                                        nPop = 18,
                                        lxt = SpainRegions$lx_male)

multiplicative_Spainmales
```

Index