

# Package ‘NNMoMo’

May 7, 2026

**Type** Package

**Title** Neural Network Extension to 'StMoMo' for Lee-Carter Modeling

**Version** 0.1.0

**Description** Provides extensions to the 'StMoMo' package by incorporating neural network functionality for Lee-Carter and Poisson Lee-Carter mortality models. Includes tools for constructing mortality datasets from 'demogdata' objects and fitting neural network-based mortality models. Further analysis, such as plotting and forecasting, can be done with 'StMoMo' functions.

**License** GPL-3

**Depends** R (>= 4.1.0)

**Imports** demography, torch, luz, StMoMo

**Suggests** testthat, knitr, rmarkdown

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**LazyData** true

**NeedsCompilation** no

**Author** Daniel Herzog [aut, cre]

**Maintainer** Daniel Herzog <studium.daniel.herzog@gmail.com>

**Repository** CRAN

**Date/Publication** 2026-02-20 08:30:02 UTC

## Contents

fit.NNMoMo . . . . .	2
lcNN . . . . .	4
logLik.fitNNMoMo . . . . .	5
NNMoMoData . . . . .	5
NNMoMo_data_AUS . . . . .	7
NNMoMo_data_CAN . . . . .	8

NNMoMo_data_GBR . . . . .	8
NNMoMo_data_JPN . . . . .	9
NNMoMo_data_USA . . . . .	9
print.fitStMoMo_list . . . . .	10
print.NNMoMo . . . . .	10
print.NNMoMoData . . . . .	11
residuals.fitNNMoMo . . . . .	11
summary.fitStMoMo_list . . . . .	12
summary.NNMoMo . . . . .	12
summary.NNMoMoData . . . . .	13

## Index 14

---

fit.NNMoMo	<i>Fit a Neural Network Lee-Carter Model</i>
------------	--

---

### Description

Fit a neural network mortality model based on prepared NNMoMo and NNMoMoData objects. The function allows selection of specific ages and years for fitting, restriction to particular series (female", "male" or "both"), and control of the training epochs. The model is trained using torch and luz and returns a list of fitStMoMo-like objects.

### Usage

```
## S3 method for class 'NNMoMo'
fit(
  object,
  data,
  ages.fit = NULL,
  years.fit = NULL,
  series = c("both", "female", "male"),
  fitting.epochs = 2000,
  batch.size = 128,
  ...
)
```

### Arguments

object	An object of class NNMoMo specifying the model structure (e.g., embedding dimension, model type, activation function, loss function). For more information see <a href="#">lcNN</a> .
data	An object of class NNMoMoData containing mortality rates, population size, and relevant demographic features. See <a href="#">NNMoMoData</a> for details.
ages.fit	Optional numeric vector specifying which ages to include in the fitting process. By default, all available ages are used.
years.fit	Optional numeric vector specifying which years to include in the fitting process. By default, all available years are used.

series	Indicates whether to fit the model for "female", "male" or "both". Default and recommended is "both".
fitting.epochs	Positive integer specifying the number of training epochs for the neural network. Values much lower than 2000 are not recommended.
batch.size	Positive integer specifying the batch size when training the model. A batch size of 128 is recommended as it was found to work the smoothest.
...	Arguments to be passed to or from other methods.

### Details

Missing or zero values in the mortality rates are imputed using the average value at a certain age across all countries for that gender in that year. Data are processed for each combination of year and sex. Categorical variables such as country and sex are embedded via embedding layers.

### Value

An object of class `fitNNMoMo`, where each element is a `fitStMoMo`-like object containing the estimated parameters  $a_x$ ,  $b_x$ ,  $k_t$ , the underlying data, and other metadata from the model fitting. These objects can be further used with functions from the **StMoMo** package for analysis, plotting, or extracting fitted mortality rates and exposures.

### Examples

```
if (torch::torch_is_installed()) {
  # Example: fitting with random data, do not expect to get
  # any meaningful results.

  # creating example data for fitting:
  demography_obj <- demography::demogdata(
    data = matrix(runif(10*5), nrow = 10),
    pop = matrix(runif(10*5, 1000, 2000), nrow = 10),
    ages = 50:59,
    years = 2000:2004,
    type = "mortality",
    name = "male",
    label = "France"
  )
  nn_data <- NNMoMoData(demography_obj)

  # fitting the data in 10 epochs (in practice, many more epochs are needed):
  fit <- fit(object = lcNN(),
            data = nn_data,
            fitting.epochs = 10)

  plot(fit$France_male)
}
```

lcNN

*Create a Lee-Carter Model with a Neural Network***Description**

Utility function to initialize a NNMoMo object representing a neural network Lee-Carter model. Important settings of the neural network can be specified here.

**Usage**

```
lcNN(
  activation = c("linear", "tanh", "relu"),
  model_type = c("FCN", "LCN", "CNN"),
  loss_type = c("MSE", "Poisson"),
  q_e = 10,
  q_z1 = 10
)
```

**Arguments**

activation	Sets the activation function of the $k_t$ -subnet of the neural network. The first activation function is set by this parameter, the second is linear by default. "tanh" defines a tangens hyperbolicus activation function, "relu" is a Rectified Linear Unit function and "linear" sets a linear activation function.
model_type	Defines the feed-forward architecture of the neural network. "FCN" sets a fully connected network for the first layer of the $k_t$ -subnet, "CNN" sets a convolutional neural network layer for this subset and "LCN" defines a locally connected network layer (note that the "LCN" option has some stability issues and therefore the use of "FCN" or "CNN" is recommended).
loss_type	Defines the type of the loss function. As a default value, "MSE" (Mean Squared Error) is set. This function minimizes the squared difference between the observed and estimated mortality rates $m_{x,t}$ . "Poisson" evaluates the likelihood of the observed deaths.
q_e	Sets the dimension of the embedding vectors for country and sex speratly. The dimension of the full embedding vector is hence $2*q_e$ . A value between 5 and 15 is recommended.
q_z1	Sets the dimension of the first layer of the $k_t$ -subnet. A value between 5 and 50 is recommended.

**Value**

An object of class NNMoMo.

**Examples**

```
# Sets up the neural network with the recommended settings:
model_1 <- lcNN()

# Sets up a fully customized neural network:
model_2 <- lcNN(activation = "tanh",
               model_type = "CNN",
               loss_type = "Poisson",
               q_e = 5,
               q_z1 = 25)

# These models can be used for fitting.
```

---

logLik.fitNNMoMo	<i>Log-Likelihood Method for Neural Network Mortality Models</i>
------------------	--

---

**Description**

Computes the log-likelihood for objects of class `fitNNMoMo`. As this method has yet not been implemented, an error will be returned.

**Usage**

```
## S3 method for class 'fitNNMoMo'
logLik(object, ...)
```

**Arguments**

<code>object</code>	An object of class <code>fitNNMoMo</code> .
<code>...</code>	Further arguments passed to or from other methods.

**Value**

No value is returned. The function always raises an error.

---

NNMoMoData	<i>Create NNMoMoData Object from Demogdata Object or HMD Datasets</i>
------------	---

---

**Description**

Utility function to construct an object of class `NNMoMoData` from one or multiple mortality datasets of class `demogdata`. Each dataset is converted into a tabular representation with mortality rates and population counts by age, sex, and year, suitable for fitting a neural network using `fit.NNMoMo`.

**Usage**

```
NNMoMoData(...)
```

**Arguments**

... One or more objects of class `demogdata`. Each input must contain mortality rates and population data for both sexes ("female", "male"), structured as in the **demography** package. If no arguments are provided, the function will prompt the user for Human Mortality Database (HMD) login credentials (email and password) and automatically download a predefined list of countries using [hmd.mx](#). The list of countries is the same used by Scognamiglio, S. (2022). Calibrating the Lee-Carter and the Poisson Lee-Carter models via neural networks. *ASTIN Bulletin: The Journal of the IAA*, 52(2), 519-561. Cambridge University Press. [doi:10.1017/asb.2022.5](https://doi.org/10.1017/asb.2022.5)

**Details**

This function converts the age-by-year matrices of mortality rates and populations into long-format rows, one row per sex-year combination. Each row stores the full vector of rates and populations across all ages.

If any input is not of class `demogdata`, the function stops with an error.

If no inputs are provided, the function connects to the Human Mortality Database (HMD) using user-supplied credentials and downloads a predefined set of countries (AUS, AUT, BEL, BGR, BLR, CAN, CHE, CZE, DEUTE, DEUTW, DNK, ESP, EST, FIN, FRATNP, GBRTENW, GBR\_NIR, GBR\_SCO, GRC, HUN, IRL, ISL, ISR, ITA, JPN, LTU, LUX, LVA, NLD, NOR, NZL\_NM, POL, PRT, RUS, SVK, SVN, SWE, TWN, UKR, USA). It might take a few minutes to download all of them.

A `demogdata` object can also be created from scratch or imported manually from the Human Mortality Database using [hmd.mx](#).

The label of the `demogdata` object must be specified, as all subsequent operations rely on this label.

**Value**

An object of class `NNMoMoData` containing the following columns:

- `country`: Country label from the input `demogdata` object.
- `sex`: Sex of the population ("female" or "male").
- `year`: Calendar year.
- `rate_*`: Mortality rates by age (e.g., `rate_0`, `rate_1`, ...).
- `pop_*`: Population counts by age (e.g., `pop_0`, `pop_1`, ...).

**Examples**

```
# Example 1: Create from an existing demogdata object
demography_obj <- demography::demogdata(
  data = matrix(runif(10*5), nrow = 10),
  pop = matrix(runif(10*5, 1000, 2000), nrow = 10),
```

```
ages = 50:59,
years = 2000:2004,
type = "mortality",
name = "male",
label = "France"
)
nn_data <- NNMoMoData(demography_obj)
head(nn_data)

## Not run:
# Example 2: Automatic download from HMD if no arguments are provided
# (will prompt for email and password, requires HMD access)
nn_data_auto <- NNMoMoData()
head(nn_data_auto)

## End(Not run)
```

---

NNMoMo\_data\_AUS

*Australia Dataset*

---

## Description

Example dataset for Australia. Provided with the **NNMoMo** package.

## Usage

```
NNMoMo_data_AUS
```

## Format

An object of class `demogdata` of length 7.

## Details

Data obtained with `demography::hmd.mx()` about Australia. Downloaded in September 2025.

## Source

**NNMoMo** package

---

NNMoMo_data_CAN	<i>Canada Dataset</i>
-----------------	-----------------------

---

**Description**

Example dataset for Canada. Provided with the **NNMoMo** package.

**Usage**

```
NNMoMo_data_CAN
```

**Format**

An object of class `demogdata` of length 7.

**Details**

Data obtained with `demography::hmd.mx()` about Canada. Downloaded in September 2025.

**Source**

**NNMoMo** package

---

NNMoMo_data_GBR	<i>United Kingdom Dataset</i>
-----------------	-------------------------------

---

**Description**

Example dataset for the United Kingdom. Provided with the **NNMoMo** package.

**Usage**

```
NNMoMo_data_GBR
```

**Format**

An object of class `demogdata` of length 7.

**Details**

Data obtained with `demography::hmd.mx()` about the United Kingdom. Downloaded in September 2025.

**Source**

**NNMoMo** package

---

NNMoMo_data_JPN	<i>Japan Dataset</i>
-----------------	----------------------

---

**Description**

Example dataset for Japan. Provided with the **NNMoMo** package.

**Usage**

```
NNMoMo_data_JPN
```

**Format**

An object of class `demogdata` of length 7.

**Details**

Data obtained with `demography::hmd.mx()` about Japan. Downloaded in September 2025.

**Source**

**NNMoMo** package

---

NNMoMo_data_USA	<i>USA Dataset</i>
-----------------	--------------------

---

**Description**

Example dataset for the United States. Provided with the **NNMoMo** package.

**Usage**

```
NNMoMo_data_USA
```

**Format**

An object of class `demogdata` of length 7.

**Details**

Data obtained with `demography::hmd.mx()` about the USA. Downloaded in September 2025.

**Source**

**NNMoMo** package

---

`print.fitStMoMo_list` *Print Method for the Output of the Fitting Method*

---

**Description**

Method for printing the output of the fitting method. Gives fundamental information about the object.

**Usage**

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

**Arguments**

`x` An object of class `fitStMoMo_list`.  
`...` Further arguments passed to or from other methods.

**Value**

Prints basic information about the `fitStMoMo_list` object to the console.

---

`print.NNMoMo` *Print Method for NNMoMo Objects*

---

**Description**

Method for printing the output of `lcNN`. Gives fundamental information about the object.

**Usage**

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

**Arguments**

`x` An object of class `NNMoMo`.  
`...` Further arguments passed to or from other methods.

**Value**

Prints basic information about the `NNMoMo` object to the console.

---

print.NNMoMoData      *Print Method for NNMoMoData Objects*

---

### Description

Method for printing the output of [NNMoMoData](#). Gives fundamental information about the object.

### Usage

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

### Arguments

x                      An object of class NNMoMoData.  
 ...                    Further arguments passed to or from other methods.

### Value

Prints basic information about the NNMoMoData object to the console.

---

residuals.fitNNMoMo      *Compute Residuals for NNMoMo Fitted Models*

---

### Description

This function computes the residuals for objects created by [fit.NNMoMo](#). It is analog to the [residuals.fitStMoMo](#) function for **StMoMo** but without the option to scale the residuals as nobs and npar are not provided by the neural network.

### Usage

```
## S3 method for class 'fitNNMoMo'
residuals(object, scale = FALSE, ...)
```

### Arguments

object                An object of class fitNNMoMo with the fitted parameters of a stochastic mortality model.  
 scale                 Logical, default FALSE. If TRUE, an error is thrown because the number of observations (nobs) and parameters (npar) cannot be calculated for fitNNMoMo objects.  
 ...                    Further arguments passed to or from other methods.

**Value**

An object of class `resStMoMo` with the residuals. This object has components:

- `residuals`: a matrix with the residuals.
- `ages`: ages corresponding to the rows in residuals.
- `years`: years corresponding to the columns in residuals.

---

```
summary.fitStMoMo_list
```

*Summary Method for the Output of the Fitting Method*

---

**Description**

Method for printing the summary of the fitting method. Gives detailed information about the object.

**Usage**

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

**Arguments**

<code>object</code>	An object of class <code>fitStMoMo_list</code> .
<code>...</code>	Further arguments passed to or from other methods.

**Value**

Displays a summary of the `fitStMoMo_list` object in the console.

---

```
summary.NNMoMo
```

*Summary Method for NNMoMo Objects*

---

**Description**

Method for printing the summary of `lcNN`. Gives detailed information about the object.

**Usage**

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

**Arguments**

<code>object</code>	An object of class <code>NNMoMo</code> .
<code>...</code>	Further arguments passed to or from other methods.

**Value**

Displays a summary of the NNMoMo object in the console.

---

summary.NNMoMoData      *Summary Method for NNMoMoData Objects*

---

**Description**

Method for printing the summary of [NNMoMoData](#). Gives detailed information about the object.

**Usage**

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

**Arguments**

object            An object of class NNMoMoData.  
...               Further arguments passed to or from other methods.

**Value**

Displays a summary of the NNMoMoData object in the console.

# Index

## \* datasets

NNMoMo\_data\_AUS, [7](#)

NNMoMo\_data\_CAN, [8](#)

NNMoMo\_data\_GBR, [8](#)

NNMoMo\_data\_JPN, [9](#)

NNMoMo\_data\_USA, [9](#)

`fit.NNMoMo`, [2](#), [5](#), [11](#)

`hmd.mx`, [6](#)

`lcNN`, [2](#), [4](#), [10](#), [12](#)

`logLik.fitNNMoMo`, [5](#)

NNMoMo\_data\_AUS, [7](#)

NNMoMo\_data\_CAN, [8](#)

NNMoMo\_data\_GBR, [8](#)

NNMoMo\_data\_JPN, [9](#)

NNMoMo\_data\_USA, [9](#)

NNMoMoData, [2](#), [5](#), [11](#), [13](#)

`print.fitStMoMo_list`, [10](#)

`print.NNMoMo`, [10](#)

`print.NNMoMoData`, [11](#)

`residuals.fitNNMoMo`, [11](#)

`residuals.fitStMoMo`, [11](#)

`summary.fitStMoMo_list`, [12](#)

`summary.NNMoMo`, [12](#)

`summary.NNMoMoData`, [13](#)