

Package ‘GenHMM1d’

September 10, 2026

Type Package

Title Goodness-of-Fit for Zero-Inflated Univariate Hidden Markov Models

Version 0.2.8

Description Inference, goodness-of-fit tests, and predictions for continuous and discrete univariate Hidden Markov Models (HMM), including zero-inflated distributions. The goodness-of-fit test is based on a Cramer-von Mises statistic and uses parametric bootstrap to estimate the p-value. The description of the methodology is taken from Nasri et al (2020) <[doi:10.1029/2019WR025122](https://doi.org/10.1029/2019WR025122)>.

License GPL-3

Encoding UTF-8

Maintainer Bouchra R. Nasri <bouchra.nasri@umontreal.ca>

Depends R (>= 3.5.0), doParallel, parallel, foreach

Imports ggplot2, stats, matrixcalc, reshape2, rmutl, VaRES, VGAM, EnvStats, GLDEX, GeneralizedHyperbolic, actuar, extraDistr, sgt, skewt, sn, ssdtools, stabledist

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Bouchra R. Nasri [aut, cre, cph],
Mamadou Yamar Thioub [aut, cph],
Bruno N. Remillard [aut, cph]

Repository CRAN

Date/Publication 2026-09-10 15:00:02 UTC

Contents

CDF	2
distributions	3
ES	3
EstHMMGen	4
ForecastHMMCdf	6

ForecastHMMeta	7
ForecastHMMPdf	8
ForecastHMMVAR	9
GofHMMGen	10
graphEstim	12
GridSearchS0	12
PDF	13
QUANTILE	14
SimHMMGen	15
SimMarkovChain	15
Snd1	16
Index	17

CDF	<i>Cumulative distribution function</i>
-----	---

Description

This function computes the cumulative distribution function (cdf) of a univariate distribution

Usage

```
CDF(family, y, param, size = 0)
```

Arguments

family	distribution name; run the function distributions() for help
y	values at which the cdf is evaluated
param	parameters of the distribution; (1 x p)
size	additional parameter for some discrete distributions; run the command distributions() for help

Value

f	cdf
---	-----

distributions	<i>The names and descriptions of the univariate distributions</i>
---------------	---

Description

This function allows the users to find the details on the available distributions.

Usage

```
distributions()
```

Value

No returned value, allows the users to know the different distributions and parameters

ES	<i>Expected shortfall function</i>
----	------------------------------------

Description

This function computes the expected shortfall of an univariate distribution, excluding zero-inflated.

Usage

```
ES(p, param, family, size = 0, Nsim = 25000)
```

Arguments

p	value (1 x 1) at which the expected shortfall needs to be computed; between 0 and 1; (e.g 0.01, 0.05)
param	parameters of the distribution; (1 x p)
family	distribution name; run the function distributions() for help
size	additional parameter for some discrete distributions; run the command distributions() for help
Nsim	number of simulations

Value

es	expected shortfall
----	--------------------

Examples

```
family = "gaussian"

theta = c(-1.5, 1.7) ;
es = ES( 0.01, theta, family)
print('Expected shortfall : ')
print(es$es)
```

EstHMMGen

Estimation of univariate hidden Markov model

Description

This function estimates the parameters from a univariate hidden Markov model

Usage

```
EstHMMGen(
  y,
  ZI = 0,
  reg,
  family,
  start = 0,
  max_iter = 10000,
  eps = 1e-04,
  size = 0,
  theta0 = NULL,
  graph = FALSE
)
```

Arguments

y	observations; (n x 1)
ZI	1 if zero-inflated, 0 otherwise (default)
reg	number of regimes (including zero-inflated; must be > ZI)
family	distribution name; run the function distributions() for help
start	starting parameters for the estimation; (1 x p)
max_iter	maximum number of iterations of the EM algorithm; suggestion 10000
eps	precision (stopping criteria); suggestion 0.001.
size	additional parameter for some discrete distributions; run the command distributions() for help
theta0	initial parameters for each regimes; (r x p), default is NULL
graph	TRUE a graph, FALSE otherwise (default); only for continuous distributions

Details

#####

Value

theta	estimated parameters; (r x p)
Q	estimated transition matrix for the regimes; (r x r)
eta	conditional probabilities of being in regime k at time t given observations up to time t; (n x r)
lambda	conditional probabilities of being in regime k at time t given all observations; (n x r)
U	pseudo-observations that should be uniformly distributed under the null hypothesis
cvm	cramer-von-Mises statistic for goodness-of-fit
W	matrix of Rosenblatt transforms; (n x r)
LL	log-likelihood
nu	stationary distribution
AIC	Akaike information criterion
BIC	Bayesian information criterion
CAIC	consistent Akaike information criterion
AICcorrected	Akaike information criterion corrected
HQC	Hannan-Quinn information criterion
stats	empirical means and standard deviation of each regimes using lambda
pred_l	estimated regime using lambda
pred_e	estimated regime using eta
runs_l	estimated number of runs using lambda
runs_e	estimated number of runs using eta

Examples

```
family = "gaussian"
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2) ;
theta = matrix(c(-1.5, 1.7, 1, 1), 2, 2) ;
y = SimHMMGen(theta, Q=Q, family=family, n=100)$SimData
est = EstHMMGen(y, ZI=0, reg=2, family=family)
```

ForecastHMMCdf	<i>Forecasted cumulative distribution function of a univariate HMM at times $n+k1$, $n+k2$,...</i>
----------------	--

Description

This function computes the forecasted cumulative distribution function of a univariate HMM for multiple horizons, given observations up to time n

Usage

```
ForecastHMMCdf(
  x,
  ZI = 0,
  family,
  theta,
  Q,
  eta,
  size = 0,
  k = 1,
  graph = FALSE
)
```

Arguments

<code>x</code>	points at which the cdf function is computed
<code>ZI</code>	1 if zero-inflated, 0 otherwise (default)
<code>family</code>	distribution name; run the function <code>distributions()</code> for help
<code>theta</code>	parameters; ($r \times p$)
<code>Q</code>	probability transition matrix for the regimes; ($r \times r$)
<code>eta</code>	vector of the estimated probability of each regime at time n ; ($1 \times r$)
<code>size</code>	additional parameter for some discrete distributions; run the command <code>distributions()</code> for help
<code>k</code>	prediction times
<code>graph</code>	TRUE to produce plots (FALSE by default).

Value

<code>cdf</code>	values of the cdf function
------------------	----------------------------

Examples

```
family = "gaussian"
theta = matrix(c(-1.5, 1.7, 1, 1),2,2)
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2)
eta = c(0.96, 0.04)
x=seq(from=-6, to=6, by=0.1)
k=c(1,5,10,20)
cdf = ForecastHMMcdf(x, 0, family, theta, Q, eta, size=0, k, graph=TRUE)
```

ForecastHMMeta	<i>Predicted probabilities of regimes of a univariate HMM for a new observation</i>
----------------	---

Description

This function computes the predicted probabilities of the regimes for a new observation of a univariate HMM, given observations up to time n

Usage

```
ForecastHMMeta(ynew, ZI = 0, family, theta, Q, eta)
```

Arguments

ynew	new observations
ZI	1 if zero-inflated, 0 otherwise (default)
family	distribution name; run the function distributions() for help
theta	parameters; (r x p)
Q	probability transition matrix for the regimes; (r x r)
eta	vector of the estimated probability of each regime at time n; (1 x r)

Value

etanew	predicted probabilities of the regimes
--------	--

Examples

```
family = "gaussian"
theta = matrix(c(-1.5, 1.7, 1, 1),2,2)
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2)
eta = c(0.96, 0.04)
ForecastHMMeta(1.5, 0, family, theta, Q, eta)
```

ForecastHMMPdf	<i>Forecasted density function of a univariate HMM at time $n+k1$, $n+k2$, ...</i>
----------------	--

Description

This function computes the probability forecasted density function (with respect to Dirac(0)+Lesbesgue) of a univariate HMM for multiple horizons, given observations up to time n

Usage

```
ForecastHMMPdf(
  y,
  ZI = 0,
  family,
  theta,
  Q,
  eta,
  size = 0,
  k = 1,
  graph = FALSE
)
```

Arguments

<code>y</code>	points at which the pdf function is computed
<code>ZI</code>	1 if zero-inflated, 0 otherwise (default)
<code>family</code>	distribution name; run the function <code>distributions()</code> for help
<code>theta</code>	parameters; ($r \times p$)
<code>Q</code>	probability transition matrix for the regimes; ($r \times r$)
<code>eta</code>	vector of the estimated probability of each regime at time n ; ($1 \times r$)
<code>size</code>	additional parameter for some discrete distributions; run the command <code>distributions()</code> for help
<code>k</code>	prediction times (may be a vector of integers)
<code>graph</code>	TRUE to produce plots (FALSE is default)

Value

pdf	values of the pdf function
-----	----------------------------

Examples

```
family = "gaussian"
theta = matrix(c(-1.5, 1.7, 1, 1),2,2)
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2)
eta = c(0.06, 0.94)
x=seq(from=-6, to=6, by=0.1)
k=c(1,5,10,20)
pdf = ForecastHMPdf(x, 1, family, theta, Q, eta, k=k, graph=TRUE)
```

ForecastHMMVAR	<i>Value at risk (VAR) of a univariate HMM at time $n+k1$, $n+k2$, ...</i>
----------------	--

Description

This function computes the VAR of a univariate HMM for multiple horizons, given observations up to time n

Usage

```
ForecastHMMVAR(U, ZI = 0, family, theta, Q, eta, k = 1)
```

Arguments

U	values ($n \times 1$) between 0 and 1
ZI	1 if zero-inflated, 0 otherwise (default)
family	distribution name; run the function <code>distributions()</code> for help
theta	parameters; ($r \times p$)
Q	probability transition matrix for the regimes; ($r \times r$)
eta	vector of the estimated probability of each regime at time n ; ($1 \times r$)
k	prediction times (may be a vector of integers).

Value

var	values at risk ($1 \times \text{horizon}$)
-----	--

Examples

```
family = "gaussian"
theta = matrix(c(-1.5, 1.7, 1, 1),2,2)
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2)
eta = c(0.96, 0.04)
U=c(0.01,0.05)
k=c(1,2,3,4,5)
ForecastHMMVAR(U, 0, family, theta, Q, eta=eta,k)
```

GofHMMGen

Goodness-of-fit of univariate hidden Markov model

Description

This function performs a goodness-of-fit test for a univariate hidden Markov model

Usage

```
GofHMMGen(
  y,
  ZI = 0,
  reg,
  family,
  start = 0,
  max_iter = 10000,
  eps = 1e-04,
  size = 0,
  n_samples = 1000,
  n_cores = 1
)
```

Arguments

y	observations
ZI	1 if zero-inflated, 0 otherwise (default)
reg	number of regimes
family	distribution name; run the function distributions() for help
start	starting parameter for the estimation
max_iter	maximum number of iterations of the EM algorithm; suggestion 10000
eps	precision (stopping criteria); suggestion 0.0001.
size	additional parameter for some discrete distributions; run the command distributions() for help
n_samples	number of bootstrap samples; suggestion 1000
n_cores	number of cores to use in the parallel computing

Value

pvalue	pvalue of the Cramer-von Mises statistic in percent
theta	Estimated parameters; (r x p)
Q	estimated transition matrix; ; (r x r)
eta	(conditional probabilities of being in regime k at time t given observations up to time t; (n x r)
lambda	conditional probabilities of being in regime k at time t given all observations; (n x r)
U	pseudo-observations that should be uniformly distributed under the null hypothesis
cvm	Cramer-von-Mises statistic for goodness-of-fit
W	matrix of Rosenblatt transforms; (n x r)
LL	log-likelihood
nu	stationary distribution
AIC	Akaike information criterion
BIC	bayesian information criterion
CAIC	consistent Akaike information criterion
AICcorrected	Akaike information criterion corrected
HQC	Hannan-Quinn information criterion
stats	Empirical means and standard deviation of each regimes using lambda
pred_l	Estimated regime using lambda
pred_e	Estimated regime using eta
runs_l	Estimated number of runs using lambda
runs_e	Estimated number of runs using eta

Examples

```
## Not run:
family = "gaussian"
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2) ; theta = matrix(c(0, 1.7, 0, 1),2,2) ;
y = SimHMMGen(theta, size=0, Q, ZI=1, family, 50)$SimData
out=GofHMMGen(y,1,2,family,n_samples=10)

## End(Not run)
```

graphEstim	<i>Graphs</i>
------------	---------------

Description

This function shows the graphs resulting from the estimation of a HMM model

Usage

```
graphEstim(y, ZI = 0, reg, theta, family, pred_l, pred_e)
```

Arguments

y	observations
ZI	1 if zero-inflated, 0 otherwise (default)
reg	number of regimes
theta	estimated parameters; (r x p)
family	distribution name; run the function distributions() for help
pred_l	estimated regime using lambda
pred_e	estimated regime using eta

Value

No returned value; produces figures of interest for the HMM model

GridSearchS0	<i>Gridsearch</i>
--------------	-------------------

Description

This function performs a gridsearch to find a good starting value for the EM algorithm. A good starting value for the EM algorithm is one for which all observations have strictly positive density (the higher the better)

Usage

```
GridSearchS0(family, y, params, size = 0, lbpdf = 0)
```

Arguments

family	distribution name; run the function distributions() for help
y	observations
params	list of six vectors named (p1, p2, p3, p4, p5, p6). Each corresponding to a parameter of the distribution (additionnal parameters will be ignored). For example : params = list(p1=c(0.5, 5, 0.5), p2=c(1, 5, 1), p3=c(0.1, 0.9, 0.1), p4=c(1,1,1), p5=c(1,1,1), p6=c(1,1,1)) where p1 is the grid of value for the first parameter.
size	additional parameter for some discrete distributions; run the command distributions() for help
lbpdf	minimal acceptable value of the density; (should be >= 0)

Value

goodStart	accepted parameter set
-----------	------------------------

Examples

```
family = "gaussian"

Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2) ;
theta = matrix(c(-1.5, 1.7, 1, 1), 2, 2) ;
sim = SimHMMGen(theta, size=0, Q, ZI=0, "gaussian", 50)$SimData ;
params = list(p1=c(-2, 2, 0.5), p2=c(1, 5, 1), p3=c(1, 1, 1), p4=c(1,1,1), p5=c(1,1,1), p6=c(1,1,1))
accepted_params = GridSearchS0(family, sim, params)
```

Description

This function computes the probability density function (pdf) of a univariate distribution

Usage

```
PDF(family, y, param, size = 0)
```

Arguments

family	distribution name; run the function distributions() for help
y	observations
param	parameters of the distribution; (1 x p)
size	additional parameter for some discrete distributions; run the command distributions() for help

Value

f pdf

QUANTILE

Quantile function

Description

This function computes the quantile function of a univariate distribution, excluding zero-inflated.

Usage

```
QUANTILE(p, param, family, size = 0)
```

Arguments

p	values at which the quantile needs to be computed; between 0 and 1; (e.g 0.01, 0.05)
param	parameters of the distribution; (1 x p)
family	distribution name; run the function distributions() for help
size	additional parameter for some discrete distributions; run the command distributions() for help

Value

q quantile/VAR

Examples

```
family = "gaussian"

theta = matrix(c(-1.5, 1.7),1,2) ;
quantile = QUANTILE(0.01, theta, family)
print('Quantile : ')
print(quantile)
```

SimHMMGen	<i>Simulation of univariate hidden Markov model</i>
-----------	---

Description

This function simulates observation from a univariate hidden Markov model

Usage

```
SimHMMGen(theta, size = 0, Q, ZI = 0, family, n)
```

Arguments

theta	parameters; (r x p)
size	additional parameter for some discrete distributions; run the command distributions() for help
Q	transition probability matrix for regimes; (r x r)
ZI	1 if zero-inflated, 0 otherwise (default)
family	distribution name; run the function distributions() for help
n	number of simulated observations

Value

SimData	Simulated data
MC	Simulated Markov chain

Examples

```
family = "gaussian"
Q = matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2) ;
theta = matrix(c(0, 1.7, 0, 10),2,2) ;
out = SimHMMGen(theta, Q=Q, ZI=1,family=family, n=50)
y=out$SimData
```

SimMarkovChain	<i>Markov chain simulation</i>
----------------	--------------------------------

Description

This function generates a Markov chain $X(1), \dots, X(n)$ with transition matrix Q , starting from a state eta0 or the uniform distribution on $1, \dots, r$.

Usage

```
SimMarkovChain(Q, n, eta0)
```

Arguments

Q	transition probability matrix
n	number of simulated vectors
eta0	initial value in 1,...,r.

Value

x	Generated Markov chain
---	------------------------

Snd1	<i>Cramer-von Mises statistic for the goodness-of-fit test of the null hypothesis of a univariate uniform distribution over [0,1]</i>
------	---

Description

This function computes the Cramer-von Mises statistic S_n for goodness-of-fit of the null hypothesis of a univariate uniform distribution over $[0,1]$

Usage

Snd1(U)

Arguments

U	vector of pseudos-observations (approximating uniform)
---	--

Value

sta	Cramer-von Mises statistic
-----	----------------------------

Index

CDF, [2](#)

distributions, [3](#)

ES, [3](#)

EstHMMGen, [4](#)

ForecastHMMCdf, [6](#)

ForecastHMMeta, [7](#)

ForecastHMPdf, [8](#)

ForecastHMMVAR, [9](#)

GofHMMGen, [10](#)

graphEstim, [12](#)

GridSearchS0, [12](#)

PDF, [13](#)

QUANTILE, [14](#)

SimHMMGen, [15](#)

SimMarkovChain, [15](#)

Snd1, [16](#)