

Package: GaussianHMM1d (via r-universe)

September 3, 2024

Title Inference, Goodness-of-Fit and Forecast for Univariate Gaussian Hidden Markov Models

Version 1.1.1

Description Inference, goodness-of-fit test, and prediction densities and intervals for univariate Gaussian Hidden Markov Models (HMM). The goodness-of-fit 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 Chapter 10.2 of Remillard (2013) <[doi:10.1201/b14285](https://doi.org/10.1201/b14285)>.

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

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation yes

Author Bouchra R. Nasri [aut, cre, cph], Bruno N Remillard [aut, ctb, cph]

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

Date/Publication 2023-07-08 10:50:10 UTC

Repository <https://bouchranasri.r-universe.dev>

RemoteUrl <https://github.com/cran/GaussianHMM1d>

RemoteRef HEAD

RemoteSha 91d43c8da613fa83343e31d56bf8483f4f27384e

Contents

EstHMM1d	2
EstRegime	3
ForecastHMMeta	4
ForecastHMMPdf	5
GaussianMixtureCdf	6
GaussianMixtureInv	7

GaussianMixturePdf	7
GofHMM1d	8
Sim.HMM.Gaussian.1d	9
Sim.Markov.Chain	10
SimHMMGaussianInv	11
Sn	12

Index	13
--------------	-----------

EstHMM1d	<i>Estimation of a univariate Gaussian Hidden Markov Model (HMM)</i>
----------	--

Description

This function estimates parameters (μ , σ , Q) of a univariate Hidden Markov Model. It computes also the probability of being in each regime, given the past observations (η) and the whole series (λ). The conditional distribution given past observations is applied to obtain pseudo-observations W that should be uniformly distributed under the null hypothesis. A Cramér-von Mises test statistic is then computed.

Usage

```
EstHMM1d(y, reg, max_iter = 10000, eps = 1e-04)
```

Arguments

<code>y</code>	($n \times 1$) vector of data
<code>reg</code>	number of regimes
<code>max_iter</code>	maximum number of iterations of the EM algorithm; suggestion 10 000
<code>eps</code>	precision (stopping criteria); suggestion 0.0001.

Value

<code>mu</code>	estimated mean for each regime
<code>sigma</code>	estimated standard deviation for each regime
<code>Q</code>	($\text{reg} \times \text{reg}$) estimated transition matrix
<code>eta</code>	($n \times \text{reg}$) probabilities of being in regime k at time t given observations up to time t
<code>lambda</code>	($n \times \text{reg}$) probabilities of being in regime k at time t given all observations
<code>cvm</code>	Cramér-von Mises statistic for the goodness-of-fit test
<code>U</code>	Pseudo-observations that should be uniformly distributed under the null hypothesis of a Gaussian HMM
<code>LL</code>	Log-likelihood

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

References

Chapter 10.2 of B. Rémillard (2013). Statistical Methods for Financial Engineering, Chapman and Hall/CRC Financial Mathematics Series, Taylor & Francis.

Examples

```
Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2); mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05)
data <- Sim.HMM.Gaussian.1d(mu,sigma,Q,eta0=1,100)$x
est <- EstHMM1d(data, 2, max_iter=10000, eps=0.0001)
```

 EstRegime

Estimated Regimes for the univariate Gaussian HMM

Description

This function computes and plots the most likely regime for univariate Gaussian HMM using probabilities of being in regime k at time t given all observations (λ) and probabilities of being in regime k at time t given observations up to time t (η).

Usage

```
EstRegime(t, y, lambda, eta)
```

Arguments

<code>t</code>	($n \times 1$) vector of dates (years, ...); if no dates then $t=[1:\text{length}(y)]$
<code>y</code>	($n \times 1$) vector of data;
<code>lambda</code>	($n \times \text{reg}$) probabilities of being in regime k at time t given all observations;
<code>eta</code>	($n \times \text{reg}$) probabilities of being in regime k at time t given observations up to time t ;

Value

<code>A</code>	Estimated Regime using λ
<code>B</code>	Estimated Regime using η
<code>runsA</code>	Estimated number of runs using λ
<code>runsB</code>	Estimated number of runs using η
<code>pA</code>	Graph for the estimated regime for each observation using λ
<code>pB</code>	Graph for the estimated regime for each observation using η

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

References

Chapter 10.2 of B. Rémillard (2013). Statistical Methods for Financial Engineering, Chapman and Hall/CRC Financial Mathematics Series, Taylor & Francis.

Examples

```
Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2); mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05);
data <- Sim.HMM.Gaussian.1d(mu,sigma,Q,eta0=1,100)$x
t=c(1:100);
est <- EstHMM1d(data, 2)
EstRegime(t,data,est$lambda, est$eta)
```

ForecastHMMeta

Estimated probabilities of the regimes given new observations

Description

This function computes the estimated probabilities of the regimes for a Gaussian HMM given new observation after time n. it also computes the associated weight of the Gaussian mixtures that can be used for forecasted density, cdf, or quantile function.

Usage

```
ForecastHMMeta(ynew, mu, sigma, Q, eta)
```

Arguments

ynew	new observations (mx1);
mu	vector of means for each regime (r x 1);
sigma	vector of standard deviations for each regime (r x 1);
Q	transition probability matrix (r x r);
eta	vector of the estimated probability of each regime (r x 1) at time n;

Value

etanew	values of the estimated probabilities at times n+1 to n+m, using the new observations
w	weights of the mixtures for periods n+1 to n+m

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

References

Chapter 10.2 of B. Rémillard (2013). Statistical Methods for Financial Engineering, Chapman and Hall/CRC Financial Mathematics Series, Taylor & Francis.

Examples

```
mu <- c(-0.3, 0.7); sigma <- c(0.15, 0.05); Q <- matrix(c(0.8, 0.3, 0.2, 0.7), 2, 2); eta <- c(.1, .9);
x <- c(0.2, -0.1, 0.73)
out <- ForecastHMMPdf(x, mu, sigma, Q, eta)
```

ForecastHMMPdf

*Density function of a Gaussian HMM at time n+k***Description**

This function computes the density function of a Gaussian HMM at time n+k, given observation up to time n.

Usage

```
ForecastHMMPdf(x, mu, sigma, Q, eta, k)
```

Arguments

x	points at which the density function is computed (mx1);
mu	vector of means for each regime (r x 1);
sigma	vector of standard deviations for each regime (r x 1);
Q	transition probability matrix (r x r);
eta	vector of the estimated probability of each regime (r x 1) at time n;
k	time of prediction.

Value

f	values of the density function at time n+k
w	weights of the mixture

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

References

Chapter 10.2 of B. Rémillard (2013). Statistical Methods for Financial Engineering, Chapman and Hall/CRC Financial Mathematics Series, Taylor & Francis.

Examples

```
mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05); Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2) ;
eta <- c(.9,.1);
x <- seq(-1, 1, by = 0.01)
out <- ForecastHMMPdf(x,mu,sigma,Q,eta,3)
plot(x,out$f,type="l")
```

GaussianMixtureCdf *Distribution function of a mixture of Gaussian univariate distributions*

Description

This function computes the distribution function of a mixture of Gaussian univariate distributions

Usage

```
GaussianMixtureCdf(x, mu, sigma, w)
```

Arguments

x	Points at which the distribution function is computed (nx1);
mu	vector of means for each regime (r x 1);
sigma	vector of standard deviations for each regime (r x 1);
w	vector of the probability of each regime (r x r).

Value

F	values of the distribution function
---	-------------------------------------

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

Examples

```
mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05); w <-c(0.8, 0.2);
x <- seq(-1, 1, by = 0.01)
F <- GaussianMixtureCdf(x,mu,sigma,w)
plot(x,F,type="l")
```

GaussianMixtureInv *Inverse distribution function of a mixture of Gaussian univariate distributions*

Description

This function computes the inverse distribution function of a mixture of Gaussian univariate distributions

Usage

```
GaussianMixtureInv(p, mu, sigma, w)
```

Arguments

p Points in (0,1) at which the distribution function is computed (nx1);
mu vector of means for each regime (r x 1);
sigma vector of standard deviations for each regime (r x 1);
w vector of the probability of each regime (r x 1).

Value

q values of the quantile function

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

Examples

```
mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05); w <-c(0.8, 0.2);  
p <- seq(0.01, 0.99, by = 0.01)  
q <- GaussianMixtureInv(p,mu,sigma,w)  
plot(p,q,type="l")
```

GaussianMixturePdf *Density function of a mixture of Gaussian univariate distributions*

Description

This function computes the density function of a mixture of Gaussian univariate distributions

Usage

```
GaussianMixturePdf(x, mu, sigma, w)
```

Arguments

x	Points at which the density is computed (n x 1);
mu	vector of means for each regime (r x 1);
sigma	vector of standard deviations for each regime (r x 1);
w	vector of the probability of each regime (r x 1).

Value

f	Values of the distribution function
---	-------------------------------------

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

Examples

```
mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05); w <-c(0.8, 0.2);
x <- seq(-1, 1, by = 0.01)
f <- GaussianMixturePdf(x,mu,sigma,w)
plot(x,f,type="l")
```

GofHMM1d

Goodness-of-fit test of a univariate Gaussian Hidden Markov Model

Description

This function performs a goodness-of-fit test of a Gaussian HMM based on a Cramér-von Mises statistic using parametric bootstrap.

Usage

```
GofHMM1d(y, reg, max_iter = 10000, eps = 1e-04, n_sample = 1000, n_cores)
```

Arguments

y	(n x 1) data vector
reg	number of regimes
max_iter	maximum number of iterations of the EM algorithm; suggestion 10 000
eps	eps (stopping criteria); suggestion 0.0001
n_sample	number of bootstrap samples; suggestion 1000
n_cores	number of cores to use in the parallel computing

Value

pvalue	pvalue of the Cramér-von Mises statistic in percent
mu	estimated mean for each regime
sigma	estimated standard deviation for each regime
Q	(reg x reg) estimated transition matrix
eta	(n x reg) conditional probabilities of being in regime k at time t given observations up to time t
lambda	(n x reg) probabilities of being in regime k at time t given all observations
cvm	Cramér-von Mises statistic for the goodness-of-fit test
W	Pseudo-observations that should be uniformly distributed under the null hypothesis of a Gaussian HMM
LL	Log-likelihood

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

References

Chapter 10.2 of B. Rémillard (2013). Statistical Methods for Financial Engineering, Chapman and Hall/CRC Financial Mathematics Series, Taylor & Francis.

Examples

```
Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2); mu <- c(-0.3,0.7) ; sigma <- c(0.15,0.05)
data <- Sim.HMM.Gaussian.1d(mu,sigma,Q,eta0=1,100)$x
gof <- GofHMM1d(data, 2, max_iter=10000, eps=0.0001, n_sample=100,n_cores=2)
```

Sim.HMM.Gaussian.1d *Simulation of a univariate Gaussian Hidden Markov Model (HMM)*

Description

This function simulates observations from a univariate Gaussian HMM

Usage

```
Sim.HMM.Gaussian.1d(mu, sigma, Q, eta0, n)
```

Arguments

mu	vector of means for each regime (r x 1);
sigma	vector of standard deviations for each regime (r x 1);
Q	Transition probability matrix (r x r);
eta0	Initial value for the regime;
n	number of simulated observations.

Value

x	Simulated Data
reg	Markov chain regimes

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

Examples

```
Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2) ; mu <- c(-0.3 ,0.7) ; sigma <- c(0.15,0.05);
sim <- Sim.HMM.Gaussian.1d(mu,sigma,Q,eta0=1,n=100)
```

Sim.Markov.Chain

Simulation of a finite Markov chain

Description

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

Usage

```
Sim.Markov.Chain(Q, n, eta0)
```

Arguments

Q	Transition probability matrix (r x r);
n	length of series;
eta0	inital value in 1,...,r.

Value

x	Simulated Markov chain
---	------------------------

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

Examples

```
Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2) ;
sim <- Sim.Markov.Chain(Q,eta0=1,n=100)
```

SimHMMGaussianInv *Simulation of a univariate Gaussian Hidden Markov Model (HMM)*

Description

Generates a univariate regime-switching random walk with Gaussian regimes starting from a given state η_0 , using the inverse method from noise u . Can be useful when generating multiple time series.

Usage

```
SimHMMGaussianInv(u, mu, sigma, Q, eta0)
```

Arguments

<code>u</code>	series of uniform i.i.d. series ($n \times 1$);
<code>mu</code>	vector of means for each regime ($r \times 1$);
<code>sigma</code>	vector of standard deviations for each regime ($r \times 1$);
<code>Q</code>	Transition probability matrix ($r \times r$);
<code>eta0</code>	Initial value for the regime;

Value

<code>x</code>	Simulated Data
<code>eta</code>	Probability of regimes

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

References

Nasri & Remillard (2019). Copula-based dynamic models for multivariate time series. *JMVA*, vol. 172, 107–121.

Examples

```
Q <- matrix(c(0.8, 0.3, 0.2, 0.7),2,2)
set.seed(1)
u <- runif(250)
mu <- c(-0.3, 0.7)
sigma <- c(0.15, 0.05);
eta0=1
x <- SimHMMGaussianInv(u,mu,sigma,Q,eta0)
```

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

Description

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

Usage

Sn(U)

Arguments

U vector of pseudos-observations (approximating uniform variates)

Value

Sn Cramér-von Mises statistic

Author(s)

Bouchra R Nasri and Bruno N Rémillard, January 31, 2019

Index

EstHMM1d, [2](#)
EstRegime, [3](#)

ForecastHMMeta, [4](#)
ForecastHMMPdf, [5](#)

GaussianMixtureCdf, [6](#)
GaussianMixtureInv, [7](#)
GaussianMixturePdf, [7](#)
GofHMM1d, [8](#)

Sim.HMM.Gaussian.1d, [9](#)
Sim.Markov.Chain, [10](#)
SimHMMGaussianInv, [11](#)
Sn, [12](#)