Package ‘mig’

April 9, 2025

Type Package

Title Multivariate Inverse Gaussian Distribution

Version 2.0

Description Provides utilities for estimation for the multivariate inverse Gaussian distribution of Minami (2003) <doi:10.1081/STA-120025379>, including random vector generation and explicit estimators of the location vector and scale matrix. The package implements kernel density estimators discussed in Belzile, Desgagnes, Genest and Ouimet (2024) <doi:10.48550/arXiv.2209.04757> for smoothing multivariate data on half-spaces.

BugReports https://github.com/lbelzile/mig/issues

Imports statmod, TruncatedNormal (>= 2.3), Rcpp (>= 1.0.12)

Depends R (>= 2.10)

Suggests numDeriv, tinytest, knitr, rmarkdown, minqa

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

LazyData true

License MIT + file LICENSE

VignetteBuilder knitr

RoxygenNote 7.3.2

NeedsCompilation yes

Author Leo Belzile [aut, cre] (<https://orcid.org/0000-0002-9135-014X>),
Frederic Ouimet [aut] (<https://orcid.org/0000-0001-7933-5265>)

Maintainer Leo Belzile <belzilel@gmail.com>

Repository CRAN

Date/Publication 2025-04-08 22:50:02 UTC

Contents

an . . . 2
dmig . . . 2
fit_mig . . . 4
geomagnetic . . . 5
hsgauss_kdens . . . 5
kdens_bandwidth . . . 6
mig_kdens . . . 7
mle_truncgauss . . . 8
proj_hs . . . 8
tnorm_kdens . . . 9

Index **10**

`an` *Threshold selection for bandwidth*

Description

Automated thresholds selection for the robust likelihood cross validation. The cutoff is based on the covariance matrix of the sample data.

Usage

```
an(x)
```

Arguments

`x`	matrix of observations

Value

cutoff for robust selection

`dmig` *Multivariate inverse Gaussian distribution*

Description

The density of the MIG model is

$$f(\boldsymbol{x}+\boldsymbol{a}) = (2\pi)^{-d/2}\boldsymbol{\beta}^\top\boldsymbol{\xi}|\boldsymbol{\Omega}|^{-1/2}(\boldsymbol{\beta}^\top\boldsymbol{x})^{-(1+d/2)}\exp\left\{-\frac{(\boldsymbol{x}-\boldsymbol{\xi})^\top\boldsymbol{\Omega}^{-1}(\boldsymbol{x}-\boldsymbol{\xi})}{2\boldsymbol{\beta}^\top\boldsymbol{x}}\right\}$$

for points in the d-dimensional half-space $\{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{\beta}^\top(\boldsymbol{x}-\boldsymbol{a}) \geq 0\}$

Usage

```
dmig(x, xi, Omega, beta, shift, log = FALSE)

rmig(n, xi, Omega, beta, shift, method = c("invsim", "bm"), timeinc = 0.001)

pmig(q, xi, Omega, beta, log = FALSE, method = c("sov", "mc"), B = 10000L)
```

Arguments

x	n by d matrix of quantiles
xi	d vector of location parameters $\boldsymbol{\xi}$, giving the expected value
Omega	d by d positive definite scale matrix $\boldsymbol{\Omega}$
beta	d vector $\boldsymbol{\beta}$ defining the half-space through $\boldsymbol{\beta}^\top\boldsymbol{\xi} > 0$
shift	d translation for the half-space $\boldsymbol{a}$
log	logical; if TRUE, returns log probabilities
n	number of observations
method	string; one of inverse system (invsim, default), Brownian motion (bm)
timeinc	time increment for multivariate simulation algorithm based on the hitting time of Brownian motion, default to 1e-3.
q	n by d matrix of quantiles
B	number of Monte Carlo replications for the SOV estimator

Details

Observations are generated using the representation as the first hitting time of a hyperplane of a correlated Brownian motion.

Value

for dmig, the (log)-density

for rmig, an n vector if d=1 (univariate) or an n by d matrix if d > 1

an n vector of (log) probabilities

Author(s)

Frederic Ouimet (bm), Leo Belzile (invsim)

Leo Belzile

Examples

```
# Density evaluation
x <- rbind(c(1, 2), c(2,3), c(0,-1))
beta <- c(1, 0)
xi <- c(1, 1)
Omega <- matrix(c(2, -1, -1, 2), nrow = 2, ncol = 2)
```

```
dmig(x, xi = xi, Omega = Omega, beta = beta)
# Random number generation
d <- 5L
beta <- rexp(d)
xi <- rexp(d)
Omega <- matrix(0.5, d, d) + diag(d)
samp <- rmig(n = 1000, beta = beta, xi = xi, Omega = Omega)
stopifnot(isTRUE(all(samp %*% beta > 0)))
mle <- fit_mig(samp, beta = beta, method = "mle")
set.seed(1234)
d <- 2L
beta <- runif(d)
Omega <- rWishart(n = 1, df = 2*d, Sigma = matrix(0.5, d, d) + diag(d))[,,1]
xi <- rexp(d)
q <- mig::rmig(n = 10, beta = beta, Omega = Omega, xi = xi)
pmig(q, xi = xi, beta = beta, Omega = Omega)
```

`fit_mig` *Fit multivariate inverse Gaussian distribution*

Description

Fit multivariate inverse Gaussian distribution

Usage

```
fit_mig(x, beta, method = c("mle", "mom"), shift)
```

Arguments

`x`	`n` by `d` matrix of quantiles
`beta`	`d` vector $\boldsymbol{\beta}$ defining the half-space through $\boldsymbol{\beta}^\top\boldsymbol{\xi} > 0$
`method`	string, one of `mle` for maximum likelihood estimation, or `mom` for method of moments.
`shift`	`d` translation for the half-space $\boldsymbol{a}$

Value

a list with components:

- `xi`: estimate of the expectation or location vector
- `Omega`: estimate of the scale matrix

geomagnetic *Magnetic storms*

Description

Absolute magnitude of 373 geomagnetic storms lasting more than 48h with absolute magnitude (dst) larger than 100 in 1957-2014.

Format

a vector of size 373

Note

For a detailed article presenting the derivation of the Dst index, see `http://wdc.kugi.kyoto-u.ac.jp/dstdir/dst2/onDst`

Source

Aki Vehtari

References

World Data Center for Geomagnetism, Kyoto, M. Nose, T. Iyemori, M. Sugiura, T. Kamei (2015), *Geomagnetic Dst index*, doi:10.17593/14515-74000.

hsgauss_kdens *Gaussian kernel density estimator on half-space*

Description

Given a data matrix over a half-space defined by `beta`, compute an homeomorphism to $\mathbb{R}^d$ and perform kernel smoothing based on a Gaussian kernel density estimator, taking each turn an observation as location vector.

Usage

```
hsgauss_kdens(x, newdata, Sigma, beta, log = TRUE, ...)
```

Arguments

`x`	`n` by `d` matrix of quantiles
`newdata`	matrix of new observations at which to evaluated the kernel density
`Sigma`	scale matrix
`beta`	`d` vector $\boldsymbol{\beta}$ defining the half-space through $\boldsymbol{\beta}^\top \boldsymbol{\xi} > 0$
`log`	logical; if `TRUE`, returns log probabilities
`...`	additional arguments, currently ignored

Value

a vector containing the value of the kernel density at each of the newdata points

---

kdens_bandwidth *Optimal scale matrix for kernel density estimation*

Description

Given an n sample from a multivariate distribution on the half-space defined by $\{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{\beta}^\top \boldsymbol{x} > 0\}$, the function computes the bandwidth (type="isotropic") or scale matrix that minimizes the asymptotic mean integrated squared error away from the boundary. The latter depend on the true unknown density, which is replaced by the kernel density or a MIG distribution evaluated at the maximum likelihood estimator. The integral or the integrated squared error are obtained by Monte Carlo integration with N simulations

Usage

```
kdens_bandwidth(
  x,
  beta,
  shift,
  family = c("mig", "hsgauss", "tnorm"),
  method = c("amise", "lcv", "lscv", "rlcv"),
  type = c("isotropic", "diag", "full"),
  approx = c("kernel", "mig", "tnorm"),
  transformation = c("none", "scaling", "spherical"),
  N = 10000L,
  buffer = 0,
  maxiter = 2000L,
  ...
)
```

Arguments

x	an n by d matrix of observations
beta	d vector defining the half-space
shift	location vector for translating the half-space. If missing, defaults to zero
family	distribution for smoothing, either mig for multivariate inverse Gaussian, tnorm for truncated normal on the half-space and hsgauss for the Gaussian smoothing after suitable transformation.
method	estimation criterion, either amise for the expression that minimizes the asymptotic integrated squared error, lcv for likelihood (leave-one-out) cross-validation, lscv for least-square cross-validation or rlcv for robust cross validation of Wu (2019)

`type`	string indicating whether to compute an isotropic model or estimate the optimal scale matrix via optimization
`approx`	string; distribution to approximate the true density function $f(x)$; either `kernel` for the kernel estimator evaluated at the sample points (except for `method="amise"`, which isn't supported), `mig` for multivariate inverse Gaussian with the method of moments or `tnorm` for the multivariate truncated Gaussian evaluated by maximum likelihood.
`transformation`	string for optional scaling of the data before computing the bandwidth. Either standardization to unit variance `scaling`, spherical transformation to unit variance and zero correlation (`spherical`), or `none` (default).
`N`	integer number of simulations for Monte Carlo integration
`buffer`	double indicating the buffer from the half-space
`maxiter`	integer; max number of iterations in the call to `optim`.
`...`	additional parameters, currently ignored

Value

a `d` by `d` scale matrix

References

Wu, X. (2019). Robust likelihood cross-validation for kernel density estimation. *Journal of Business & Economic Statistics*, 37(**4**), 761–770. doi:10.1080/07350015.2018.1424633 Bowman, A.W. (1984). An alternative method of cross-validation for the smoothing of density estimates, *Biometrika*, 71(**2**), 353–360. doi:10.1093/biomet/71.2.353 Rudemo, M. (1982). Empirical choice of histograms and kernel density estimators. *Scandinavian Journal of Statistics*, 9(**2**), 65–78. http://www.jstor.org/stable/4615859

`mig_kdens` *Multivariate inverse Gaussian kernel density estimator*

Description

Given a matrix of new observations, compute the density of the multivariate inverse Gaussian mixture defined by assigning equal weight to each component where $\boldsymbol{\xi}$ is the location parameter.

Usage

```
mig_kdens(x, newdata, Omega, beta, log = FALSE)
```

Arguments

`x`	`n` by `d` matrix of quantiles
`newdata`	matrix of new observations at which to evaluated the kernel density
`Omega`	`d` by `d` positive definite scale matrix $\boldsymbol{\Omega}$
`beta`	`d` vector $\boldsymbol{\beta}$ defining the half-space through $\boldsymbol{\beta}^\top\boldsymbol{\xi} > 0$
`log`	logical; if `TRUE`, returns log probabilities

Value

value of the (log)-density at `newdata`

`mle_truncgauss` *Maximum likelihood estimation of truncated Gaussian on half space*

Description

Given a data matrix and a vector of linear constraints for the half-space, we compute the maximum likelihood estimator of the location and scale matrix

Usage

```
mle_truncgauss(xdat, beta)
```

Arguments

`xdat`	matrix of observations
`beta`	vector of constraints defining the half-space

Value

a list with location vector `loc` and scale matrix `scale`

`proj_hs` *Orthogonal projection matrix onto the half-space*

Description

The orthogonal projection matrix P has unit determinant and transforms an `n` by `d` matrix by taking $x * P$. The components of the first column vector of the resulting matrix are strictly positive.

Usage

```
proj_hs(beta, inv = FALSE)
```

Arguments

`beta`	vector defining the half-space
`inv`	logical; if `TRUE`, return the inverse matrix

Value

a `d` by `d` orthogonal projection matrix tnorm_kdens *Truncated Gaussian kernel density estimator*

Description

Given a data matrix over a half-space defined by `beta`, compute the log density of the asymmetric truncated Gaussian kernel density estimator, taking in turn an observation as location vector.

Usage

```
tnorm_kdens(x, newdata, Sigma, beta, log = TRUE, ...)
```

Arguments

`x`	`n` by `d` matrix of quantiles
`newdata`	matrix of new observations at which to evaluated the kernel density
`Sigma`	scale matrix
`beta`	`d` vector $\boldsymbol{\beta}$ defining the half-space through $\boldsymbol{\beta}^\top \boldsymbol{\xi} > 0$
`log`	logical; if `TRUE`, returns log probabilities
`...`	additional arguments, currently ignored

Value

a vector containing the value of the kernel density at each of the `newdata` points

Index

an, 2

dmig, 2

fit_mig, 4

geomagnetic, 5

hsgauss_kdens, 5

kdens_bandwidth, 6

mig_kdens, 7
mle_truncgauss, 8

pmig *(dmig)*, 2
proj_hs, 8

rmig *(dmig)*, 2

tnorm_kdens, 9