Package: wishartinference (via r-universe)

October 10, 2026

Type Package

Title Bayesian Inference for the Wishart Distribution Parameters

Version 0.1.0

Description Posterior inference for the shape parameter alpha and mean matrix mu in the model X_i ~ Wishart_p(2*alpha, Sigma), under both an improper prior and a proper Gamma/inverse-Wishart prior. The posterior mode is found via a Newton-within-EM algorithm and joint samples are drawn via rejection sampling.

License MIT + file LICENSE

Encoding UTF-8

Imports Rcpp

LinkingTo Rcpp, RcppArmadillo, BH

Suggests testthat (>= 3.1.5)

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

NeedsCompilation yes

Author Philip Everson [aut], Hanqi Shi [aut, cre]

Maintainer Hanqi Shi <hshi1@swarthmore.edu>

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

Date/Publication 2026-09-03 12:10:02 UTC

RemoteUrl https://github.com/cran/wishartinference

RemoteRef HEAD

RemoteSha 3ce31645153cdc82694b4c83288f5b60c9fdb066

Contents

gamma_inference 2
ldet 3
lfafun_gamma 4
lfafun_improper 5

lfafun_proper 5
lgammap_export 6
mode_alphaEM 7
mode_alphaEM_gamma 8
rejection_sampler 10
rejection_sampler_gamma 11
rwishart 12
wishart_hello 13
wishart_inference 13
wishart_stats 15

Index **16**

`gamma_inference` *Bayesian inference for the Wishart shape parameter, gamma (p = 1) case*

Description

Given `x_1,...,x_n ~ iid Gamma(alpha, alpha/mu)` – the p = 1 reduction of `X_i ~ Wishart_p(2*alpha, Sigma)` – computes the posterior of (alpha, mu) under either an improper or proper prior. Scalar (p = 1) analogue of `wishart_inference()`: takes a plain numeric vector instead of a p x p x n array, since each observation is already a positive scalar. There is no unifying dispatch with `wishart_inference()` – call this directly for scalar data.

Usage

```
gamma_inference(
  x,
  mu0 = 1,
  beta = 0,
  eta = 1,
  kappa = 0,
  nsamp = 10000L,
  tol = 1e-06,
  prnt = FALSE,
  max_em_iter = 1000L
)
```

Arguments

`x`	Numeric vector of n scalar observations, each > 0
`mu0`	Prior center for mu, must be > 0 (ignored if improper)
`beta`	Gamma prior shape, must be >= 0 (ignored if improper)
`eta`	Gamma prior rate, must be >= 0 (ignored if improper)
`kappa`	Prior strength on mu, must be >= 0 (ignored if improper)
`nsamp`	Number of posterior samples

tol	Convergence tolerance for the EM mode finder
prnt	If TRUE, print EM iterates
max_em_iter	Maximum number of EM iterations

Value

A list with results (alpha_samples, mu_samples, ahat, theoretical_acpt_rate, empirical_acpt_rate) and statistics (xbar, muhat, log_det_geometric_mean, cover_shape, cover_rate, elapsed_seconds). If a convergence failure is caught internally, returns list(error = "...") instead.

Examples

```
set.seed(1)
x <- rgamma(20, shape = 3, rate = 3)   # true alpha = 3, mu = 1

# improper prior (the default)
res <- gamma_inference(x, nsamp = 200)
res$results$ahat
quantile(res$results$alpha_samples, c(0.025, 0.975))

# proper prior
res2 <- gamma_inference(x, mu0 = 1, beta = 5, eta = 0.5,
                        kappa = 2, nsamp = 200)
res2$results$ahat
```

---

ldet *Log determinant of a positive definite matrix*

Description

Uses Armadillo's log_det for numerical stability.

Usage

```
ldet(X)
```

Arguments

X	Square positive definite matrix

Value

log|X|

Examples

```
ldet(diag(c(1, 2, 3)))   # log(6)

set.seed(1)
A <- crossprod(matrix(rnorm(9), 3, 3)) + diag(3)
ldet(A)
log(det(A))
```

`lfafun_gamma` *Log unnormalized posterior of alpha, gamma (p = 1) case*

Description

The p = 1 reduction of the Wishart model: X_i ~ Gamma(alpha, alpha/mu). Auto-detects improper vs. proper exactly as the p >= 2 functions do (beta = 0, eta = 1, kappa = 0 = improper), except kappa = 0 alone (with beta > 0) is also valid here – a proper prior on alpha with a flat/improper treatment of mu – unlike the p >= 2 backend, which requires kappa >= 1 whenever the prior isn't fully improper.

Usage

```
lfafun_gamma(a, n, xbar, ldetxbarg, beta = 0, eta = 1, kappa = 0, mu0 = 1)
```

Arguments

`a`	Shape parameter alpha, must be > 0
`n`	Number of scalar observations
`xbar`	Sample mean of the observations
`ldetxbarg`	mean(log(x)) – scalar analogue of ldetxbarg
`beta`	Gamma prior shape, must be >= 0 (0 = improper)
`eta`	Gamma prior rate, must be >= 0
`kappa`	Prior strength on mu, must be >= 0 (0 = improper on mu)
`mu0`	Prior center for mu, must be > 0

Value

log f*(alpha), or -Inf if a <= 0

Examples

```
set.seed(1)
x <- rgamma(20, shape = 3, rate = 3)   # true alpha = 3, mu = 1

lfafun_gamma(3, 20, mean(x), mean(log(x)))

# proper prior on alpha
lfafun_gamma(3, 20, mean(x), mean(log(x)),
             beta = 5, eta = 0.5, kappa = 2, mu0 = 1)
```

lfafun_improper — *Log unnormalized marginal posterior of alpha under the improper prior*

Description

The improper prior p(alpha,mu) propto (alpha-(p-1)/2)^(-1) * |mu|^(-(p+1)/2), after marginalizing over mu, gives a closed-form log posterior for alpha. Returns `-Inf` when `a <= (p-1)/2`.

Usage

```
lfafun_improper(a, n, p, xbar, ldetxbarg)
```

Arguments

`a`	Shape parameter alpha, must be > (p-1)/2
`n`	Number of Wishart observations
`p`	Dimension
`xbar`	Sample mean matrix (p x p)
`ldetxbarg`	Log geometric mean of determinants (1/n)*sum log|X_i|

Value

log f*(alpha), or -Inf if alpha <= (p-1)/2

Examples

```
set.seed(1)
X  <- rwishart(10, 2, 6, diag(2))
st <- wishart_stats(X)

lfafun_improper(3, 10, 2, st$xbar, st$ldetxbarg)

# -Inf outside the domain alpha > (p-1)/2
lfafun_improper(0.2, 10, 2, st$xbar, st$ldetxbarg)
```

lfafun_proper — *Log unnormalized marginal posterior of alpha under the proper prior*

Description

The proper prior (alpha - (p-1)/2) ~ Gamma(beta, rate = beta*eta) and mu|alpha ~ inv-Wishart(2*kappa*alpha, 2*kappa*alpha*mu0), after marginalizing over mu, gives a closed-form log posterior for alpha. Returns `-Inf` outside the domain `alpha > (p-1)/2`.

Usage

```
lfafun_proper(a, n, p, ldet_muhat, ldetxbarg, ldet_mu0, beta, eta, kappa)
```

Arguments

a	Shape parameter alpha, must be > (p-1)/2
n	Number of Wishart observations
p	Dimension
ldet_muhat	log|muhat| where muhat = (n*xbar + kappa*mu0)/(n+kappa)
ldetxbarg	Log geometric mean of determinants (1/n)*sum log|X_i|
ldet_mu0	log|mu0|
beta	Gamma prior shape, must be >= 0
eta	Gamma prior rate parameter, must be >= 0
kappa	inv-Wishart prior strength, must be >= 1

Value

log f*(alpha), or -Inf if outside domain

Examples

```
set.seed(1)
n <- 10; p <- 2; kappa <- 1
X  <- rwishart(n, p, 6, diag(p))
st <- wishart_stats(X)

mu0   <- diag(2, p)
muhat <- (n * st$xbar + kappa * mu0) / (n + kappa)

lfafun_proper(3, n, p, ldet(muhat), st$ldetxbarg, ldet(mu0),
              beta = 5, eta = 0.5, kappa = kappa)
```

---

lgammap_export *Log of the multivariate Gamma function*

Description

Defined as log Gamma_p(a) = p*(p-1)/4 * log(pi) plus the sum over i = 1, ..., p of log Gamma(a - (i-1)/2). Required for the normalizing constant of the Wishart and inverse-Wishart distributions.

Usage

```
lgammap_export(a, p)
```

Arguments

a	Argument, must be > (p-1)/2
p	Dimension

Value

log Gamma_p(a)

Examples

```
lgammap_export(5, 3)

# equals the defining sum
a <- 5; p <- 3
p * (p - 1) / 4 * log(pi) + sum(lgamma(a - (0:(p - 1)) / 2))
```

---

mode_alphaEM *Find the posterior mode of alpha*

Description

Dispatches to the improper- or proper-prior EM mode finder depending on which parameters are supplied. The prior type is auto-detected: omit `mu0`, `beta`, `eta`, `kappa` for the improper prior; supply `mu0` and `beta >= 0`, `kappa >= 1` for the proper prior. Uses a Newton-within-EM algorithm.

Usage

```
mode_alphaEM(
  n,
  p,
  xbar,
  ldetxbarg,
  mu0 = NULL,
  beta = 0,
  eta = 1,
  kappa = 0,
  tol = 1e-06,
  prnt = FALSE,
  max_em_iter = 1000L,
  max_nr_iter = 100L
)
```

Arguments

n	Number of Wishart observations
p	Dimension
xbar	Sample mean matrix (p x p)
ldetxbarg	Log geometric mean of determinants
mu0	Prior center matrix (p x p); omit for improper prior
beta	Gamma prior shape, must be >= 0 (ignored if improper)
eta	Gamma prior rate, must be >= 0 (ignored if improper)
kappa	inv-Wishart prior strength, must be >= 1 (ignored if improper)
tol	Convergence tolerance
prnt	If TRUE, print EM iterates and bounds
max_em_iter	Maximum number of EM iterations
max_nr_iter	Maximum number of inner Newton-Raphson iterations

Value

A numeric vector `c(ahat, log f*(ahat))`

Examples

```
set.seed(1)
n <- 10; p <- 2
X  <- rwishart(n, p, 6, diag(p))   # true alpha = 3
st <- wishart_stats(X)

# improper prior: omit mu0, beta, eta, kappa
em <- mode_alphaEM(n, p, st$xbar, st$ldetxbarg)
em[1]   # posterior mode of alpha
em[2]   # log f*(ahat)

# proper prior: (alpha - (p-1)/2) ~ Gamma(beta, rate = beta*eta)
mode_alphaEM(n, p, st$xbar, st$ldetxbarg,
             mu0 = diag(2, p), beta = 5, eta = 0.5, kappa = 1)
```

---

mode_alphaEM_gamma *EM algorithm to find the posterior mode of alpha, gamma (p = 1) case*

Description

Scalar (p = 1) analogue of `mode_alphaEM()`. See the C++ doc comment for details on the bisection-style ascent used here and the n = 2, kappa = 0, beta = 0 structural boundary case.

Usage

```
mode_alphaEM_gamma(
  n,
  xbar,
  ldetxbarg,
  beta = 0,
  eta = 1,
  kappa = 0,
  mu0 = 1,
  tol = 1e-06,
  prnt = FALSE,
  max_em_iter = 1000L
)
```

Arguments

n	Number of scalar observations
xbar	Sample mean of the observations
ldetxbarg	mean(log(x))
beta	Gamma prior shape, must be >= 0 (0 = improper)
eta	Gamma prior rate, must be >= 0
kappa	Prior strength on mu, must be >= 0 (0 = improper on mu)
mu0	Prior center for mu, must be > 0
tol	Convergence tolerance
prnt	If TRUE, print EM iterates and bounds
max_em_iter	Maximum number of EM iterations

Value

A numeric vector `c(ahat, log f*(ahat))`

Examples

```
set.seed(1)
x <- rgamma(20, shape = 3, rate = 3)   # true alpha = 3

em <- mode_alphaEM_gamma(20, mean(x), mean(log(x)))
em[1]   # posterior mode of alpha

# proper prior
mode_alphaEM_gamma(20, mean(x), mean(log(x)),
                   beta = 5, eta = 0.5, kappa = 2, mu0 = 1)
```

`rejection_sampler` *Rejection sampler for the joint posterior of (alpha, mu)*

Description

Requires the posterior mode `ahat` and `mxlfa` = log f*(ahat) from `mode_alphaEM()`, plus the covering Gamma parameters `lambda` and `nu_star`. Prior type is auto-detected from parameters, exactly as in `mode_alphaEM()`.

Usage

```
rejection_sampler(
  ahat,
  mxlfa,
  lambda,
  nu_star,
  p,
  n,
  xbar,
  ldetxbarg,
  mu0 = NULL,
  beta = 0,
  eta = 1,
  kappa = 0,
  nsamp = 10000L
)
```

Arguments

`ahat`	Posterior mode from `mode_alphaEM()[1]`
`mxlfa`	log f*(ahat) from `mode_alphaEM()[2]`
`lambda`	Covering Gamma rate
`nu_star`	Covering Gamma shape (= ahat*lambda + 1)
`p`	Dimension
`n`	Number of Wishart observations
`xbar`	Sample mean matrix (p x p)
`ldetxbarg`	Log geometric mean of determinants
`mu0`	Prior center matrix; omit for improper prior
`beta`	Gamma prior shape; 0 = improper (with defaults below), must be >= 0 if proper
`eta`	Gamma prior rate; default 1.0, must be >= 0 if proper
`kappa`	inv-Wishart prior strength; 0 = improper, must be >= 1 if proper
`nsamp`	Number of posterior samples (default 10000)

Value

A list with alpha_sample, mu_sample, empirical_acpt_rate, theoretical_acpt_rate

Examples

```
set.seed(1)
n <- 10; p <- 2
X  <- rwishart(n, p, 6, diag(p))
st <- wishart_stats(X)
em <- mode_alphaEM(n, p, st$xbar, st$ldetxbarg)

# covering Gamma parameters under the improper prior
lambda  <- max(n * (ldet(st$xbar) - st$ldetxbarg), 1e-10)
nu_star <- em[1] * lambda + 1

samp <- rejection_sampler(em[1], em[2], lambda, nu_star,
                          p, n, st$xbar, st$ldetxbarg, nsamp = 100)
quantile(samp$alpha_sample, c(0.025, 0.5, 0.975))
samp$empirical_acpt_rate
```

---

rejection_sampler_gamma

Rejection sampler for the joint posterior of (alpha, mu), gamma (p = 1) case

---

Description

Scalar (p = 1) analogue of `rejection_sampler()`. mu | alpha, x is drawn from reciprocal-Gamma(nk_eff*alpha, alpha*(n*xbar+kappa*mu0)) – see the C++ doc comment for the derivation.

Usage

```
rejection_sampler_gamma(
  n,
  xbar,
  ldetxbarg,
  ahat,
  mxlfa,
  beta = 0,
  eta = 1,
  kappa = 0,
  mu0 = 1,
  nsamp = 10000L
)
```

Arguments

n	Number of scalar observations
xbar	Sample mean of the observations
ldetxbarg	mean(log(x))
ahat	Posterior mode from mode_alphaEM_gamma()[1]
mxlfa	log f*(ahat) from mode_alphaEM_gamma()[2]
beta	Gamma prior shape, must be >= 0 (0 = improper)
eta	Gamma prior rate, must be >= 0
kappa	Prior strength on mu, must be >= 0 (0 = improper on mu)
mu0	Prior center for mu, must be > 0
nsamp	Number of posterior samples (default 10000)

Value

A list with alpha_sample, mu_sample, empirical_acpt_rate, theoretical_acpt_rate

Examples

```
set.seed(1)
x  <- rgamma(20, shape = 3, rate = 3)
em <- mode_alphaEM_gamma(20, mean(x), mean(log(x)))

samp <- rejection_sampler_gamma(20, mean(x), mean(log(x)),
                                em[1], em[2], nsamp = 100)
quantile(samp$alpha_sample, c(0.025, 0.5, 0.975))
samp$empirical_acpt_rate
```

---

rwishart *Generate draws from a Wishart distribution*

Description

Uses the Bartlett decomposition: if L is the lower Cholesky factor of Sigma, then A = L * T satisfies W = A * A^T ~ Wishart_p(nu, Sigma), where T is lower triangular with T(i,i) ~ sqrt(Chi^2(nu - i)) and T(i,j) ~ N(0,1) for i > j.

Usage

```
rwishart(n, p, nu, Sigma)
```

Arguments

n	Number of draws
p	Dimension of the matrix, must be >= 2
nu	Degrees of freedom, must be > p - 1
Sigma	Scale matrix (p x p), must be positive definite

Value

A cube of size (p, p, n) containing the Wishart draws

Examples

```
set.seed(1)
X <- rwishart(5, 2, 6, diag(2))
dim(X)
X[, , 1]
```

---

wishart_hello *Verify that the compiled library loaded correctly*

Description

A minimal smoke test, useful immediately after installation to confirm that the package's compiled code is available.

Usage

```
wishart_hello()
```

Value

Returns 0. Called for the confirmation message it prints.

Examples

```
wishart_hello()
```

---

wishart_inference *Bayesian inference for the Wishart shape parameter*

Description

Given `X_1,...,X_n ~ iid Wishart_p(2*alpha, Sigma)`, computes the posterior of (alpha, mu) where mu = alpha*Sigma, under either an improper or proper prior. Runs the full pipeline: sufficient statistics, EM mode finding, and rejection sampling.

Usage

```
wishart_inference(
  X,
  mu0 = NULL,
  beta = 0,
  eta = 1,
  kappa = 0,
  nsamp = 10000L,
  tol = 1e-06,
  prnt = FALSE,
  max_em_iter = 1000L,
  max_nr_iter = 100L
)
```

Arguments

X	Cube of n Wishart observations, dimensions (p, p, n)
mu0	Prior center matrix (p x p); omit for improper prior
beta	Gamma prior shape, must be >= 0 (ignored if improper)
eta	Gamma prior rate parameter, must be >= 0 (ignored if improper)
kappa	inv-Wishart prior strength, must be >= 1 (ignored if improper)
nsamp	Number of posterior samples
tol	Convergence tolerance for the EM mode finder
prnt	If TRUE, print EM iterates
max_em_iter	Maximum number of EM iterations
max_nr_iter	Maximum number of inner Newton-Raphson iterations

Value

A list with `results` (alpha_samples, mu_samples, ahat, theoretical_acpt_rate, empirical_acpt_rate) and `statistics` (xbar, muhat, log_det_geometric_mean, cover_shape, cover_rate, elapsed_seconds). If a convergence failure is caught internally, returns `list(error = "...")` instead.

Examples

```
set.seed(1)
X <- rwishart(10, 2, 6, diag(2))   # true alpha = 3

# improper prior (the default)
res <- wishart_inference(X, nsamp = 200)
res$results$ahat
quantile(res$results$alpha_samples, c(0.025, 0.975))

# proper prior
res2 <- wishart_inference(X, mu0 = diag(2, 2), beta = 5,
                          eta = 0.5, kappa = 1, nsamp = 200)
res2$results$ahat
```

wishart_stats *Compute sufficient statistics for Wishart observations*

Description

Returns the sample mean matrix xbar and the log geometric mean of determinants ldetxbarg, the sufficient statistics for (alpha, Sigma) under X_i ~ Wishart_p(2*alpha, Sigma).

Usage

```
wishart_stats(X)
```

Arguments

X	Cube of n Wishart draws, dimensions (p, p, n)

Value

A list with `xbar` (p x p sample mean matrix) and `ldetxbarg` (log geometric mean of determinants)

Examples

```
set.seed(1)
X  <- rwishart(10, 2, 6, diag(2))
st <- wishart_stats(X)
st$xbar
st$ldetxbarg

# ldetxbarg is a log geometric mean, so it sits below log|xbar|
st$ldetxbarg < ldet(st$xbar)
```

Index

gamma_inference, 2

ldet, 3
lfafun_gamma, 4
lfafun_improper, 5
lfafun_proper, 5
lgammap_export, 6

mode_alphaEM, 7
mode_alphaEM_gamma, 8

rejection_sampler, 10
rejection_sampler_gamma, 11
rwishart, 12

wishart_hello, 13
wishart_inference, 13
wishart_stats, 15