Package 'spFFBS'

July 15, 2026

Title Spatiotemporal Propagation for Multivariate Bayesian Dynamic Learning

Version 1.0-1

Maintainer Luca Presicce <l.presicce@campus.unimib.it>

Description Implementation of the Forward Filtering Backward Sampling (FFBS) algorithm with Dynamic Bayesian Predictive Stacking (DYNBPS) integration for multivariate spatiotemporal models, as introduced in ``Adaptive Markovian Spatiotemporal Transfer Learning in Multivariate Bayesian Modeling" (Presicce and Banerjee, 2026+) <doi:10.48550/arXiv.2602.08544>. This methodology enables efficient Bayesian multivariate spatiotemporal modeling, utilizing dynamic predictive stacking to improve inference across multivariate time series of spatial datasets. The core functions leverage 'C++' for high-performance computation, making the framework well-suited for large-scale spatiotemporal data analysis in parallel computing environments.

LinkingTo Rcpp, RcppArmadillo

Imports spBPS, Rcpp (>= 1.1.1), abind

Suggests doParallel, mniw, MBA, ggplot2, patchwork, reshape2, knitr, rmarkdown

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

VignetteBuilder knitr

URL https://lucapresicce.github.io/spFFBS/

NeedsCompilation yes

Author Luca Presicce [aut, cre] (ORCID: <https://orcid.org/0009-0005-7062-3523>)

Repository CRAN

Date/Publication 2026-07-15 19:20:02 UTC

Contents

make_prior 2
spFFBS 2

Index 5

`make_prior` *Convert a full prior covariance matrix to block form for spFFBS*

Description

Convert a full prior covariance matrix to block form for spFFBS

Usage

```
make_prior(m0, C0, nu0, Psi0, p)
```

Arguments

`m0`	(p+n) x q prior mean matrix.
`C0`	(p+n) x (p+n) prior covariance matrix (full).
`nu0`	Prior degrees of freedom (scalar).
`Psi0`	q x q prior scale matrix.
`p`	Number of regression coefficients (integer).

Value

A list with elements m, C_bb, C_bw, C_ww, nu, Psi ready to pass as `prior` to `spFFBS`.

`spFFBS` *spFFBS: Spatiotemporal Bayesian Pipeline (optimised)*

Description

Optimised wrapper for the spFFBS v2 C++ backend. Runs forward filtering, dynamic stacking weight computation, and optionally backward sampling, temporal forecasting, and spatial interpolation.

Usage

```
spFFBS(
  Y,
  X,
  D,
  grid = list(tau = NULL, phi = NULL),
  prior,
  G_beta = NULL,
  rho = 1,
  do_BS = FALSE,
```

```
  do_forecast = FALSE,
  do_spatial = FALSE,
  L = 200L,
  tnew = NULL,
  X_all = NULL,
  spatial = NULL,
  n_threads = 1L,
  verbose = TRUE
)
```

Arguments

Y	Response array, n x q x T.
X	Covariate array, n x p x T. Replaces P.
D	n x n spatial distance matrix.
grid	List with elements tau and phi (numeric vectors).
prior	Block-form prior list; see make_prior. Elements: m ((p+n)xq), C_bb (pxp), C_bw (pxn), C_ww (nxn), nu (scalar), Psi (qxq). If the old-style full C is supplied instead of C_bb/C_bw/C_ww, the wrapper auto-converts using make_prior().
G_beta	pxp upper-left block of the system matrix G. Default NULL = identity.
rho	Scalar multiplier for the lower-right nxn identity block of G. Default 1.
do_BS	Logical; run backward sampling? Default FALSE.
do_forecast	Logical; run temporal forecast? Default FALSE.
do_spatial	Logical; run spatial interpolation? Default FALSE.
L	Number of posterior samples. Default 200.
tnew	Forecast horizon (required when do_forecast = TRUE).
X_all	nxpx(T+tnew) covariate array for forecasting (required when do_forecast = TRUE).
spatial	List for spatial interpolation: list(crd, crdtilde, Xtilde, t). Xtilde should be a uxp matrix (covariates at prediction sites).
n_threads	Number of OpenMP threads. Default 1.
verbose	Logical; print progress? Default TRUE.

Value

A list with (depending on flags):

FF Forward filter output from spFF3_v2.

Wi nxJ dynamic weight matrix.

Wglobal Jx1 global (averaged) weight vector.

BS (if do_BS) T-list of nxqxL sample cubes.

forecast (if do_forecast) List with Y_pred array of dimensions nxqxLxT.

spatial (if do_spatial) Named list of L-lists of prediction matrices, one per requested time.

Differences from spFFBS (v1)

- G is now a *pxp matrix* (G_beta) plus a scalar rho, representing the block-diagonal system matrix $G = \mathrm{diag}(G_\beta,\, \rho I_n)$. Defaults to the identity (G_beta = NULL, rho = 1).
- X (nxpxT array of covariates) replaces the P cube. The observation matrix $P_t = [X_t \mid I_n]$ is formed implicitly inside C++.
- prior should be in block form; use make_prior() to convert a full C_0 matrix if needed.

Examples

```
set.seed(42)
n <- 50; T <- 10; p <- 2; q <- 2

coords <- matrix(runif(n * 2), ncol = 2)
D      <- as.matrix(dist(coords))
Y      <- array(rnorm(n * q * T), dim = c(n, q, T))
X      <- array(rnorm(n * p * T), dim = c(n, p, T))

prior  <- make_prior(
  m0   = matrix(0, n + p, q),
  C0   = diag(n + p),
  nu0  = 3,
  Psi0 = diag(q),
  p    = p
)

res <- spFFBS(
  Y = Y, X = X, D = D,
  grid  = list(tau = c(0.8, 0.9), phi = c(1, 2)),
  prior = prior,
  L = 50
)
```

Index

make_prior, 2, *3*

spFFBS, 2