Package 'rsDCM'

November 14, 2026

Type Package

Title Robust and Sparse Dynamic Causal Modelling for Functional MRI

Version 0.2.0

Date 2026-11-08

Maintainer Godfred Arhin <arhin0122@gmail.com>

Description Fits robust and sparse group-level Dynamic Causal Models (DCMs) for functional magnetic resonance imaging data. The group method combines a Student-t likelihood with a Gaussian-spike and nonlocal product-moment slab prior, propagates first-level posterior covariance through block pre-whitening, fits empirical-Bayes modes by multistart generalized expectation-maximization with conditional maximization, and integrates posterior inclusion probabilities by parallel-tempering Monte Carlo (<doi:10.48550/arXiv.2609.06379>). The package also provides an R port of deterministic single-state DCM and Parametric Empirical Bayes routines from 'SPM25' (version 25.01.02), based on Friston et al. (2003) <doi:10.1016/S1053-8119(03)00202-7>.

License GPL-2

Encoding UTF-8

LazyData true

LazyDataCompression xz

Depends R (>= 4.0.0)

Imports Matrix, Rcpp, expm, MASS, methods, stats, utils

LinkingTo Rcpp

Suggests testthat (>= 3.0.0), knitr, rmarkdown, R.matlab, withr

URL https://kay202.github.io/rsDCM/

BugReports https://github.com/Kay202/rsDCM/issues

VignetteBuilder knitr

Config/testthat/edition 3

Config/roxygen2/version 7.3.3

NeedsCompilation yes

RoxygenNote 7.3.3

Author Godfred Arhin [aut, cre, trl],
Nilotpal Sanyal [aut],
SPM25 Authors [ctb, cph] (Original MATLAB SPM25 (v25.01.02) implementation; see LICENSE.note)

Repository CRAN

Date/Publication 2026-11-14 15:50:14 UTC

Contents

rsDCM-package . . . 2
dcm_estimate . . . 4
dcm_evidence . . . 5
dcm_fmri_priors . . . 6
dcm_fx_fmri . . . 7
dcm_gx_fmri . . . 7
dcm_int . . . 8
dcm_nlsi_GN . . . 9
dcm_peb_of_pebs . . . 10
dcm_peb_prepare . . . 11
dcm_peb_run . . . 12
narps_dcm . . . 12
rsdcm . . . 13
rsdcm_fit . . . 17
rsdcm_options . . . 18
toy_dcm . . . 19

Index **21**

`rsDCM-package` *rsDCM: Robust and Sparse Dynamic Causal Modelling for Functional MRI*

Description

The package provides a robust and sparse group-level extension of Parametric Empirical Bayes for Dynamic Causal Modelling (DCM), together with an R port of deterministic single-state fMRI DCM routines from SPM25.

Details

The proposed group method combines a Student-t likelihood on pre-whitened first-level posterior summaries with a Gaussian-spike and nonlocal product-moment (pMOM)-slab prior. Mode fitting uses a multistart generalized EM algorithm with conditional maximization. Posterior inclusion probabilities are integrated conditionally on fitted empirical-Bayes hyperparameters by parallel-tempering MCMC. Conditional Laplace standard deviations are also returned for the group coefficients.

Main functions

`rsdcm` Fit the robust and sparse group model.

`rsdcm_fit` Assemble group inputs from fitted DCMs.

`dcm_estimate` Invert a single-subject fMRI DCM.

`dcm_peb_run` Fit the Gaussian SPM-style PEB model.

Other functions

`rsdcm_options` Get or set package numerical options.

`dcm_nlsi_GN` Variational Laplace / Gauss-Newton inversion.

`dcm_int` Bilinear-system integrator (forward model).

`dcm_fmri_priors` Construct fMRI DCM priors.

`dcm_fx_fmri`, `dcm_gx_fmri` Neural-state and BOLD observation equations.

`dcm_bireduce`, `dcm_kernels` Bilinear reduction and Volterra kernels.

`dcm_evidence`, `dcm_log_evidence`, `dcm_log_evidence_reduce` Model evidence and Bayesian model reduction.

`dcm_peb_prepare`, `dcm_peb_run` Prepare and fit a second- or third-level PEB.

`dcm_peb_design`, `dcm_peb_files`, `dcm_peb_load` Design-matrix and subject-PEB loading helpers.

Lower-level numerical helpers (matrix and vector utilities, numerical differentiation, matrix exponentials, and error-covariance bases) are also exported and documented; see, for example, `dcm_vec`, `dcm_inv`, `dcm_diff`, and `dcm_Ce`.

Datasets

`toy_dcm` Three-region single-subject DCM specification.

`narps_dcm` Corrected 48-subject NARPS DCM summaries.

Lower-level numerical helpers (matrix and vector utilities, numerical differentiation, matrix exponentials, and error-covariance bases) are also exported and documented; see, for example, `dcm_vec`, `dcm_inv`, `dcm_diff`, and `dcm_Ce`.

Acknowledgment

The single-subject DCM routines are a derivative work of the MATLAB SPM25 (version 25.01.02) toolbox, distributed under GPL-2 by the Wellcome Centre for Human Neuroimaging. See the `LICENSE.note` file in the package source for the list of ported routines.

Author(s)

Maintainer: Godfred Arhin <arhin0122@gmail.com> [translator] Authors:

- Godfred Arhin <arhin0122@gmail.com> [translator]
- Nilotpal Sanyal <nsanyal@utep.edu>

Other contributors:

- SPM25 Authors (Original MATLAB SPM25 (v25.01.02) implementation; see LICENSE.note) [contributor, copyright holder]

References

Arhin, G. and Sanyal, N. (2026). Robust and Sparse Group Dynamic Causal Modeling via Student-t Parametric Empirical Bayes and Nonlocal Priors. arXiv:2609.06379. doi:10.48550/arXiv.2609.06379

Friston, K. J., Harrison, L., and Penny, W. (2003). Dynamic causal modelling. NeuroImage, 19(4), 1273–1302. doi:10.1016/S10538119(03)002027

Friston, K.J., Mattout, J., Trujillo-Barreto, N., Ashburner, J., Penny, W. (2007). Variational free energy and the Laplace approximation. NeuroImage, 34(1), 220-234. doi:10.1016/j.neuroimage.2006.08.035

Botvinik-Nezer, R., Holzmeister, F., Camerer, C. F., et al. (2020). Variability in the analysis of a single neuroimaging dataset by many teams. Nature, 582(7810), 84-88. doi:10.1038/s41586020-23149

Valerio, D., Peres, A., Bergstrom, F., Seidel, P., Almeida, J. (2025). Neural and behavioral similarity-driven tuning curves for manipulable objects. Imaging Neuroscience, 3. doi:10.1162/imag_a_00482

See Also

Useful links:

- https://github.com/Kay202/rsDCM
- Report bugs at https://github.com/Kay202/rsDCM/issues

`dcm_estimate` *Estimate a Dynamic Causal Model for fMRI*

Description

Inverts a fully-specified DCM using variational Laplace inversion. This is the main user-facing entry point of the package. Mirrors SPM25's `dcm_estimate`. Progress is reported via `message()` and can be silenced with `suppressMessages()`.

Usage

```
dcm_estimate(P, save = FALSE)
```

Arguments

`P`	A DCM list, or a path to an `.RData`/`.rds` file containing one.
`save`	Logical. If `TRUE` and `P` is a file path, the estimated DCM is saved back to that path.

Value

The estimated DCM (a list) with posterior fields populated. The main fields of interest are `Ep` (posterior expectations, with `Ep$A`, `Ep$B`, `Ep$C` the connectivity estimates), `Cp` (posterior covariance), `Pp` (posterior probabilities), and `F` (the negative free energy, used for model comparison).

Supported model variants

This release implements only the **deterministic, single-state** fMRI DCM. The two-state (`options$two_state`), stochastic (`options$stochastic`) and spectral / cross-spectral-density (`options$induced`) variants are *not* yet implemented: switching any of them on causes `dcm_estimate` to stop with an informative error. These variants are planned for a future update. Non-linear DCM (a non-empty `d` array) is supported.

See Also

`dcm_nlsi_GN` for the underlying inversion, `dcm_fmri_priors` for the priors it constructs.

Examples

```
data(toy_dcm)
str(toy_dcm, max.level = 1)


# Full inversion of the bundled three-region model (takes about a minute).
# Progress reporting goes through message(), so it can be silenced.
fit <- suppressMessages(dcm_estimate(toy_dcm))
round(fit$Ep$A, 3)   # posterior connectivity estimates
round(fit$Pp$A, 3)   # posterior probability each connection is non-zero
fit$F                # negative free energy, for model comparison
```

`dcm_evidence` *Approximate model evidence (AIC, BIC)*

Description

AIC and BIC penalties for an estimated DCM, plus per-region cost terms. Mirrors SPM25's `spm_dcm_evidence`.

Usage

```
dcm_evidence(DCM)
```

Arguments

`DCM`	An estimated DCM.

Value

List with per-region cost, AIC penalty, BIC penalty, and overall AIC and BIC.

`dcm_fmri_priors` *Construct fMRI DCM priors*

Description

Builds the prior expectations `pE` and prior covariance `pC` for a deterministic fMRI DCM. Mirrors SPM25's `dcm_fmri_priors`.

Usage

```
dcm_fmri_priors(A, B, C, D, options = list())
```

Arguments

`A`	Connectivity adjacency matrix.
`B`	Modulatory adjacency array.
`C`	Driving-input adjacency matrix.
`D`	Non-linear adjacency array.
`options`	List of model options (`stochastic`, `induced`, `two_state`, `backwards`, `precision`, `decay`).

Value

List with `pE` (prior expectation), `x` (initial state template), and `pC` (prior covariance).

Model variants

Only the deterministic, single-state model is supported end-to-end in this release. Branches for the two-state (`options$two_state`) and spectral (`options$induced`) variants exist but are experimental and are not wired through `dcm_estimate`, which rejects them. They are planned for a future update.

See Also

`dcm_estimate`, which builds these priors for you.

Examples

```
# Priors for the bundled three-region (deterministic) model
data(toy_dcm)
pr <- dcm_fmri_priors(toy_dcm$a, toy_dcm$b, toy_dcm$c,
                      D = NULL, options = toy_dcm$options)
names(pr)
pr$pE$A          # prior expectation of the endogenous connections
dim(pr$x)        # 3 regions x 5 hemodynamic states
```

dcm_fx_fmri *fMRI neural and hemodynamic state equation*

Description

Computes dx/dt for the bilinear neural state equation coupled to the Buxton-Friston hemodynamic model. Mirrors SPM25's spm_fx_fmri.

Usage

```
dcm_fx_fmri(x, u, P, M)
```

Arguments

x	State matrix (rows = regions, cols = state variables).
u	Driving inputs at the current time.
P	List of model parameters (A, B, C, D, transit, decay, epsilon, ...).
M	Model structure list (optional).

Value

Matrix dx/dt of the same shape as x.

dcm_gx_fmri *fMRI BOLD observation equation*

Description

Computes the BOLD signal from the hemodynamic state variables. Mirrors SPM25's spm_gx_fmri.

Usage

```
dcm_gx_fmri(x, u, P, M)
```

Arguments

x	State matrix (rows = regions, cols = state variables).
u	Driving inputs at the current time.
P	List of model parameters (A, B, C, D, transit, decay, epsilon, ...).
M	Model structure list (optional).

Value

List with the BOLD prediction g and Jacobian dgdx.

dcm_int *Integrate a bilinear DCM*

Description

Forward-integrates the model defined by `M$f` (state equation) and `M$g` (observation equation) under the input `U`, returning the predicted observations at the requested sample points.

Usage

```
dcm_int(P, M, U)
```

Arguments

`P`	Parameter structure.
`M`	Model list. Required fields: `f`, `x`; optional `g`, `ns`, `delays`, `l`.
`U`	List with input matrix `u` and time step `dt`.

Details

`M$f` and `M$g` may be given as functions or as the names of functions. `M$n` must be the length of the flattened state (`length(dcm_vec(M$x))`, i.e. regions x hidden states), while `M$l` is the number of observed outputs (regions) and `M$m` the number of inputs.

Value

Numeric matrix of predicted observations (rows = samples).

See Also

`dcm_estimate`, which calls this as its forward model.

Examples

```
# Simulate the BOLD response of a two-region model to a boxcar input.
n <- 2L
pri <- dcm_fmri_priors(A = matrix(1, n, n),
                       B = array(0, c(n, n, 1)),
                       C = matrix(c(1, 0), n, 1),
                       D = array(0, c(n, n, 0)),
                       options = list())

U <- list(u = matrix(c(rep(1, 16), rep(0, 16)), ncol = 1), dt = 1)
M <- list(f = "dcm_fx_fmri", g = "dcm_gx_fmri", x = pri$x,
          m = ncol(U$u), n = length(pri$x), l = nrow(pri$x), ns = 32)

# The priors put C at zero, so start from them and switch on a driving
# input to region 1 and a connection from region 1 to region 2.
```

```
P <- pri$pE
P$C[1, 1] <- 1
P$A[2, 1] <- 0.4

y <- dcm_int(P, M, U)
dim(y)                  # 32 samples x 2 regions
round(y[seq(1, 32, 4), ], 3)
```

`dcm_nlsi_GN` *Variational Laplace inversion of a non-linear system*

Description

Performs Gauss-Newton optimisation of the variational free energy for a non-linear forward model with Gaussian priors. Mirrors SPM25's `dcm_nlsi_GN`.

Usage

```
dcm_nlsi_GN(M, U, Y)
```

Arguments

`M`	Model specification (list with `IS`/`f`/`g`, priors `pE`, `pC`, hyperpriors `hE`, `hC`, ...).
`U`	Input structure passed through to `M$IS`.
`Y`	Data (or list with `$y`, `$Q`, `$X0`, `$dt`).

Details

Most users should call `dcm_estimate` instead, which assembles `M`, `U` and `Y` from a DCM specification and calls this function. Use `dcm_nlsi_GN` directly only to invert a non-linear model that is not an fMRI DCM.

Progress is reported per Gauss-Newton iteration as `EM: (+) k F: ...`, where `(+)` marks an accepted step and `(-)` a rejected one. Set `M$noprint <- 1` to silence it.

Value

List with posterior expectation `Ep`, covariance `Cp`, log-precision estimate `Eh`, free energy `F`, and components.

See Also

`dcm_estimate` for the user-facing entry point.

Examples

```
# dcm_nlsi_GN is the Gauss-Newton optimiser that dcm_estimate() calls after
# assembling M, U and Y from a DCM specification. For a runnable inversion
# see the example in ?dcm_estimate; dcm_nlsi_GN returns the same posterior
# fields (Ep, Cp, Eh, F).
```

dcm_peb_of_pebs *Run a third-level PEB-of-PEBs over a directory of subject PEBs*

Description

Loads subject PEBs (`.rds` or `.mat`), builds the third-level design, runs the PEB-of-PEBs, and returns a tidy summary. Optionally saves the group PEB.

Usage

```
dcm_peb_of_pebs(
  peb_dir,
  subjects = NULL,
  covariates = NULL,
  Xnames = NULL,
  save_path = NULL,
  M = list(Q = "single", alpha = 1, beta = 16, maxit = 64),
  field = "all",
  verbose = TRUE
)
```

Arguments

`peb_dir`	Directory containing the subject PEB files.
`subjects`	Optional integer vector; restrict to these subject numbers.
`covariates`	Optional data frame of between-subject covariates, one row per subject in the order returned by `dcm_peb_files`. `NULL` gives an intercept-only design (the group mean).
`Xnames`	Optional custom design column names.
`save_path`	Optional path to save the group PEB to, as `.rds`. `NULL` (the default) does not write anything to disk.
`M`	Third-level model options passed to `dcm_peb_prepare`.
`field`	Parameter blocks to model; `"all"` is typical for PEB inputs.
`verbose`	Logical. Report progress via `message()`.

Details

Nothing is written to disk unless `save_path` is supplied.

Value

List with `group_PEB`, `save_path` (`NULL` if not saved), `subjects`, `X` (the design), and `summary` (a tidy data frame of parameter estimates).

See Also

`dcm_peb_prepare`, `dcm_peb_run`.

`dcm_peb_prepare` *Prepare a PEB (second- or third-level) model*

Description

Gathers the first-level posterior densities, selects the parameter indices implied by `field`, projects them onto a rank-reduced subspace, and builds the second-level priors, hyperpriors and precision components. Mirrors the preparation half of SPM25's `spm_dcm_peb`.

Usage

```
dcm_peb_prepare(P, M = list(), field = c("A", "B"))
```

Arguments

`P`	List of estimated DCMs (or, for a PEB-of-PEBs, a list of PEBs). Each element must carry `M$pE`, `M$pC`, `Ep`, `Cp` and `F`.
`M`	Second-level model specification. Recognised fields: `X` (design matrix, subjects x covariates), `W`, `Q` (covariance component option, see Details), `bE`, `bC`, `pC`, `hE`, `hC`, `alpha`, `beta`, `Xnames`, `maxit`. Defaults to an intercept-only design.
`field`	Character vector of parameter blocks to model (e.g. `c("A", "B")`), the string `"all"` for every named block, or numeric indices into `dcm_vec(M$pE)`.

Details

`M$Q` selects the form of the second-level precision components: `"single"` (one component, the default), `"fields"` (one per requested field), `"all"` (one per parameter), `"none"` (no components), or a list of numeric matrices for manual specification.

Value

A list ("prep") consumed by `dcm_peb_run`, holding the projected densities, design, priors, precision components and sizes.

See Also

`dcm_peb_run` to fit the prepared model.

dcm_peb_run — *Fit a prepared PEB model by variational Laplace*

Description

Runs the variational Laplace (Fisher scoring) loop on the structure built by dcm_peb_prepare, and returns the group-level PEB together with the first-level DCMs updated under the empirical priors. Mirrors the estimation half of SPM25's spm_dcm_peb.

Usage

```
dcm_peb_run(prep, verbose = TRUE)
```

Arguments

prep	The list returned by dcm_peb_prepare.
verbose	Logical. Report free energy per iteration via message(). Silence with suppressMessages() or verbose = FALSE.

Details

Progress is reported as VL Iter k: F=... dF=... [t=...], where t is the log step size of the Fisher scoring update. The loop stops when the step size collapses or the free-energy increase falls below 1e-4.

Value

List with two elements: PEB, the group-level result (Ep group parameter expectations, Cp their covariance, Eh/Ch log-precision estimates, F free energy, and the Pnames/Xnames/Snames labels), and P, the input DCMs with Ep, Cp, MpE, MpC and F updated under the empirical priors.

See Also

dcm_peb_prepare, dcm_peb_of_pebs.

narps_dcm — *Corrected subject-level DCM summaries for NARPS*

Description

Corrected within-subject PEB summaries for 48 participants in the openly shared mixed-gambles NARPS data. These summaries replace the preliminary package data used before the first-level input correction.

Usage

```
narps_dcm
```

Format

A list with:

eta A 48 by 22 matrix of posterior means.

Cp A named length-48 list of 22 by 22 posterior covariances.

parameter_names Names of the 22 parameters.

subject De-identified participant identifiers.

covariates Task version, gender, and age for each participant.

regions The four region labels in model order.

Details

Each participant has 22 parameters: 16 intrinsic connections from a four-region DCM and six task-modulatory connections. The regions are vmPFC, ventral striatum, amygdala, and anterior insula.

Source

Corrected DCM summaries derived from the NARPS data described by Botvinik-Nezer et al. (2020), Nature 582, 84–88. doi:10.1038/s4158602023149

Examples

```
data(narps_dcm)
dim(narps_dcm$eta)
length(narps_dcm$Cp)
table(narps_dcm$covariates$group)
```

`rsdcm` *Robust and sparse group-level DCM*

Description

Fits a group model to subject-level DCM posterior means and covariances. The likelihood is Student-t after block-covariance pre-whitening. Group coefficients receive a Gaussian-spike and nonlocal pMOM-slab prior.

Usage

```
rsdcm(
  eta_theta_y,
  C_theta_y_list,
  X_G,
  V_list,
  nu = 3,
  tau0 = 0.05,
  tau1 = 1,
  pi = 0.4,
  a0 = 2,
  b0 = 0.01,
  a1 = 2,
  b1 = 1,
  min_slab_spike_ratio = 1,
  max_iter = 200,
  tol = 1e-06,
  inner_sweeps = 3,
  verbose = TRUE,
  max_beta_step = 1,
  min_alpha = 1e-08,
  max_alpha = 100,
  whitening_method = c("precision_chol", "covariance_chol", "symmetric"),
  estimate_sigma2 = FALSE,
  sigma2_fixed = 1,
  beta_init = NULL,
  alpha_init = NULL,
  n_starts = 3L,
  start_jitter = 0.25,
  start_seed = NULL,
  objective_tol = 1e-10,
  gradient_tol = 1e-04,
  max_backtrack = 20L,
  pip_all_starts = FALSE,
  pip_mcmc_iter = 10000L,
  pip_mcmc_burn = 3000L,
  pip_mcmc_thin = 2L,
  pip_mcmc_chains = 4L,
  pip_mcmc_temperatures = 8L,
  pip_mcmc_min_inverse_temp = 0.05,
  pip_mcmc_swap_every = 2L,
  pip_mcmc_large_jump_prob = 0.1,
  pip_mcmc_seed = NULL
)
```

Arguments

`eta_theta_y`	Numeric `N` by `p` matrix of subject-level posterior means.

`C_theta_y_list` Length-`N` list of `p` by `p` posterior covariance matrices.

`X_G` Numeric `N` by `r` between-subject design matrix.

`V_list` List of symmetric `p` by `p` bases defining $\Sigma_b = \sum_k \alpha_k V_k$. The matrices must be linearly independent.

`nu` Positive Student-t degrees of freedom.

`tau0`, `tau1` Initial spike and slab standard deviations.

`pi` Prior slab probability in `(0, 1)`.

`a0`, `b0`, `a1`, `b1` Inverse-Gamma hyperparameters for the squared spike and slab scales.

`min_slab_spike_ratio`
Minimum allowed ratio `tau1/tau0`.

`max_iter` Maximum generalized-EM iterations per start.

`tol` Relative parameter and objective convergence tolerance.

`inner_sweeps` Number of coordinate-Newton sweeps in each conditional maximization of the group effects.

`verbose` Logical. Print fitting and MCMC diagnostics.

`max_beta_step` Maximum absolute coordinate-Newton step.

`min_alpha`, `max_alpha`
Bounds for covariance-component amplitudes.

`whitening_method`
Pre-whitening factorization. One of `"precision_chol"`, `"covariance_chol"`, or `"symmetric"`.

`estimate_sigma2`
Logical. Estimate the global residual scale. The default fixes it to avoid confounding with covariance-component scale.

`sigma2_fixed` Positive residual scale used when `estimate_sigma2 = FALSE`.

`beta_init`, `alpha_init`
Optional initial coefficient and covariance- component vectors.

`n_starts` Number of empirical-Bayes starts.

`start_jitter` Standard deviation multiplier for randomized starts.

`start_seed` Optional random seed for randomized starts.

`objective_tol` Numerical tolerance for monotonicity safeguards.

`gradient_tol` Maximum coefficient-score norm required for convergence.

`max_backtrack` Maximum backtracking halvings for a coefficient update.

`pip_all_starts` Logical. Compute integrated PIPs for every start rather than only the selected start.

`pip_mcmc_iter`, `pip_mcmc_burn`, `pip_mcmc_thin`
Parallel-tempering MCMC iteration, burn-in, and thinning settings.

`pip_mcmc_chains`
Number of independent temperature ladders.

`pip_mcmc_temperatures`
Number of replicas in each ladder.

`pip_mcmc_min_inverse_temp`
: Inverse temperature of the hottest replica.

`pip_mcmc_swap_every`
: Number of iterations between replica swaps.

`pip_mcmc_large_jump_prob`
: Probability of a slab-scale proposal.

`pip_mcmc_seed`
: Optional random seed for posterior PIP integration.

Details

Mode fitting uses multistart generalized expectation-maximization with conditional maximization. Integrated posterior inclusion probabilities (PIPs) are then computed by parallel-tempering MCMC while holding the empirical-Bayes hyperparameters fixed. The returned coefficient standard deviations use a conditional Laplace approximation at the selected mode.

The Student-t weights apply to residual coordinates defined by the selected whitening factor and the supplied parameter ordering. They are not invariant to arbitrary rotations of the whitened coordinates.

Value

A list containing empirical-Bayes modes `beta_mat`, conditional posterior means `beta_posterior_mean`, conditional Laplace standard deviations `beta_sd_approx`, integrated PIPs `inclusion`, PIP Monte Carlo errors and chain diagnostics, Student-t weights, fitted scales and covariance components, convergence traces, multistart diagnostics, whitening information, and fitted subject-level means.

References

Johnson, V. E. and Rossell, D. (2010). On the use of non-local prior densities in Bayesian hypothesis tests. Journal of the Royal Statistical Society: Series B, 72(2), 143–170.

See Also

`rsdcm_fit`, `dcm_peb_run`

Examples

```
set.seed(1)
N <- 12L
p <- 3L
X_G <- matrix(1, N, 1, dimnames = list(NULL, "intercept"))
eta <- matrix(rnorm(N * p, sd = 0.25), N, p)
eta[, 1] <- eta[, 1] + 0.7
colnames(eta) <- paste0("P", seq_len(p))
Cp <- replicate(N, diag(0.1, p), simplify = FALSE)
V <- lapply(seq_len(p), function(j) {
  z <- matrix(0, p, p)
  z[j, j] <- 1
  z
})
```

```
fit <- rsdcm(
  eta, Cp, X_G, V, n_starts = 1L, max_iter = 100L,
  pip_mcmc_iter = 1000L, pip_mcmc_burn = 400L,
  pip_mcmc_thin = 5L, pip_mcmc_chains = 2L,
  pip_mcmc_temperatures = 4L, verbose = FALSE
)
cbind(fit$beta_mat, PIP = fit$inclusion[, 1])
```

`rsdcm_fit` *Robust sparse group DCM from a list of fitted DCMs*

Description

Convenience wrapper around `rsdcm` that assembles its inputs from a list of estimated DCMs (as returned by `dcm_estimate`). For each subject it takes the posterior mean `Ep` and covariance `Cp` restricted to the requested parameter field, builds a between-subject design, and fits the robust + sparse group model.

Usage

```
rsdcm_fit(P, field = "A", covariates = NULL, X_G = NULL, V_list = NULL, ...)
```

Arguments

`P`	List of estimated DCMs; each must carry `Ep`, `Cp`, and `M$pE`/`M$pC` (or `options`) so the field indices can be resolved.
`field`	Parameter block(s) to model at the group level, named as in the DCM parameter structure `Ep`. For an fMRI DCM these are `"A"`, `"B"`, `"C"`, `"D"`, `"transit"`, `"decay"` and `"epsilon"`; pass one (e.g. `"A"`) or several (e.g. `c("A", "B")`). Case-sensitive: the connectivity blocks are uppercase (`"A"`, not `"a"`). Passed to `dcm_find_pC`.
`covariates`	Optional data frame of between-subject covariates, one row per subject; `NULL` gives an intercept-only design (the group mean). Ignored if `X_G` is supplied.
`X_G`	Optional group design matrix (N x r); overrides `covariates`.
`V_list`	Optional list of p x p variance-component bases; defaults to a per-parameter diagonal basis.
`...`	Further arguments passed to `rsdcm` (e.g. `nu`, `pi`, `verbose`).

Details

All subjects must share the same parameterisation, so the selected field indices are required to match across `P`.

Value

The list returned by `rsdcm`, with the group effects, conditional Laplace standard deviations, integrated PIPs, and diagnostics labelled by parameter and design columns, plus `param_index`.

See Also

rsdcm, dcm_estimate, dcm_peb_design.

Examples

```
# rsdcm_fit() consumes a list of fitted DCMs (each from dcm_estimate), then
# fits the group model on a chosen field:
#   fits <- lapply(dcm_files, function(f) dcm_estimate(readRDS(f)))
#   grp  <- rsdcm_fit(fits, field = "A", covariates = my_covariates)
# See ?rsdcm for a self-contained group-level example.
```

`rsdcm_options` *Get or set rsDCM runtime options*

Description

Read or modify package-internal numerical options. The option `GLOBAL_DX` controls the finite-difference step used by `dcm_diff`.

Usage

```
rsdcm_options(...)
```

Arguments

`...`	Named options to set. With no arguments, returns all options.

Value

The current option list, invisibly when setting values.

Examples

```
old <- rsdcm_options()$GLOBAL_DX
rsdcm_options(GLOBAL_DX = 1e-4)
rsdcm_options(GLOBAL_DX = old)
```

`toy_dcm` *Example DCM for fMRI inversion*

Description

A three-region Dynamic Causal Model used in the documentation, vignette, and tests. It is an input specification (no posterior fields), ready to invert with `dcm_estimate`.

Usage

```
toy_dcm
```

Format

A list with the standard SPM-style DCM fields:

a 3 x 3 binary matrix of endogenous connections to estimate.

b 3 x 3 x 1 binary array of modulatory connections.

c 3 x 1 binary matrix of driving inputs; here the input enters region 3 only.

Y Observed BOLD data. `Y$y` is the 482 x 3 time-series matrix (rows = scans, columns = regions); `Y$dt` is the sampling interval in seconds; `Y$X0` is the confound design; `Y$Q` is the error-covariance component list; `Y$name` holds the region names.

U Input design. `U$u` is the 482 x 1 input matrix at the microtime resolution; `U$dt` is the microtime step; `U$name` holds the input names.

n, v Number of regions (3) and number of scans (482).

TE Echo time in seconds.

name Name of the model.

xY Region (VOI) structure carried over from the source DCM.

options Estimation options: `nonlinear`, `two_state`, `stochastic`, `centre`, `mode`, `maxit`, `induced`, `maxnodes`, `hE`, `hC`.

Details

The model has 3 regions and 482 scans, with a single driving input entering region 3 and a single modulatory input. All entries of the a matrix are enabled, so every directed connection between the three regions is estimated.

Source

Derived from the fMRI data of Valerio et al. (2025), "Neural and behavioral similarity-driven tuning curves for manipulable objects", Imaging Neuroscience, 3, doi:10.1162/imag_a_00482, via a Dynamic Causal Modelling exercise carried out with that data during package development. The build script that prepared the shipped object is in `data-raw/make-toy-dcm.R`. See the package `CITATION` for the full author list.

Examples

```
data(toy_dcm)
str(toy_dcm, max.level = 1)
# Invert it with dcm_estimate(toy_dcm); see ?dcm_estimate for a runnable
# (donttest) inversion example.
```

Index

∗ **datasets**
narps_dcm, 12
toy_dcm, 19

dcm_bireduce, *3*
dcm_Ce, *3*
dcm_diff, *3*, *18*
dcm_estimate, *3*, 4, *6*, *8, 9*, *17–19*
dcm_evidence, *3*, 5
dcm_find_pC, *17*
dcm_fmri_priors, *3*, *5*, 6
dcm_fx_fmri, *3*, 7
dcm_gx_fmri, *3*, 7
dcm_int, *3*, 8
dcm_inv, *3*
dcm_kernels, *3*
dcm_log_evidence, *3*
dcm_log_evidence_reduce, *3*
dcm_nlsi_GN, *3*, *5*, 9
dcm_peb_design, *3*, *18*
dcm_peb_files, *3*, *10*
dcm_peb_load, *3*
dcm_peb_of_pebs, 10, *12*
dcm_peb_prepare, *3*, *10*, 11, *12*
dcm_peb_run, *3*, *10, 11*, 12, *16*
dcm_vec, *3*

narps_dcm, *3*, 12

rsDCM *(*rsDCM-package*)*, 2
rsdcm, *3*, 13, *17, 18*
rsDCM-package, 2
rsdcm_fit, *3*, *16*, 17
rsdcm_options, *3*, 18

toy_dcm, *3*, 19