Package: phynd (via r-universe)

May 15, 2026

Type Package

Title Fast Design-Free Physiological Denoising

Version 0.0.0.9000

Description Fast, design-free physiological denoising inspired by PHYCAA+ using low-rank dynamic component extraction with optional DiCCA/DiPCA integration.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports methods, stats, utils, Rcpp, rsvd

LinkingTo Rcpp, RcppArmadillo

Suggests testthat (>= 3.0.0), dipca, neuroim2, jsonlite, knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

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

Date/Publication 2026-02-14 21:07:22 UTC

RemoteUrl https://github.com/bbuchsbaum/phynd

RemoteRef HEAD

RemoteSha 9a2c3522dfc1f267f6d2eb793073c11fb1c34fae

Contents

`benchmark_fast_phy_denoise`
Benchmark fast_phy_denoise across dimensions and SVD engines

Description

Benchmark fast_phy_denoise across dimensions and SVD engines

Usage

```
benchmark_fast_phy_denoise(
  grid_n_vox = c(4000L),
  grid_n_time = c(400L),
  svd_engines = c("svd", "rsvd"),
  reps = 1L,
  out_file = NULL,
  seed = 1L,
  ...
)
```

Arguments

`grid_n_vox`	Integer vector of voxel counts.
`grid_n_time`	Integer vector of time lengths.
`svd_engines`	Character vector of SVD engines to compare.
`reps`	Number of repetitions per configuration.
`out_file`	Optional CSV path for machine-readable results.
`seed`	Seed base.
`...`	Additional arguments passed to fast_phy_denoise.

Value

Data frame of benchmark results.

compcor_denoise *CompCor-style denoising (aCompCor / tCompCor)*

Description

CompCor-style denoising (aCompCor / tCompCor)

Usage

```
compcor_denoise(
  x,
  tr,
  mode = c("acompcorr", "tcompcorr"),
  n_comp = 5L,
  mask = NULL,
  nuisance_mask = NULL,
  wnn = NULL,
  wnn_thresh = 0.5,
  top_var_frac = 0.02,
  top_var_n = NULL,
  design = NULL,
  center_rows_first = TRUE,
  svd_engine = c("auto", "rsvd", "svd"),
  seed = NULL,
  return_diagnostics = TRUE
)
```

Arguments

x	Matrix-like data (voxels x time or time x voxels).
tr	Repetition time in seconds.
mode	One of "acompcorr" or "tcompcorr".
n_comp	Number of principal regressors to remove.
mask	Optional voxel mask (logical or indices) applied before processing.
nuisance_mask	Optional voxel mask for aCompCor ROI (logical or indices).
wnn	Optional wNN vector; used for aCompCor when nuisance_mask is not provided.
wnn_thresh	wNN threshold for aCompCor voxel selection.
top_var_frac	Fraction of highest-variance voxels for tCompCor.
top_var_n	Optional explicit number of voxels for tCompCor.
design	Optional design matrix (time x predictors) for guardrail orthogonalization.
center_rows_first	Whether to mean-center voxel time series first.
svd_engine	SVD backend for PCA ("auto", "rsvd", "svd").

seed	Optional random seed.
return_diagnostics	Return detailed diagnostics.

Details

This function provides an explicit reduced baseline that maps onto CompCor:

- mode = "acompcorr" reduces to an anatomical/noise-ROI CompCor variant: nuisance voxels are selected from nuisance_mask when provided; otherwise from wnn < wnn_thresh (or auto-derived wNN).
- mode = "tcompcorr" reduces to temporal CompCor: nuisance voxels are selected by highest temporal variance.

In both modes, principal nuisance regressors are extracted from the selected voxel set and projected out from all voxels in one linear projection step.

Value

A list with cleaned data, regressors, and diagnostics.

---

compute_design_free_qc

Compute design-free QC summary

Description

Compute design-free QC summary

Usage

```
compute_design_free_qc(x_raw, x_clean, wNN, component_table = NULL)
```

Arguments

x_raw	Raw data (matrix-like).
x_clean	Denoised data (matrix-like).
wNN	Non-neuronal weighting vector.
component_table	Optional component summary table from fast_phy_denoise.

Value

Named list of QC metrics.

compute_wNN_diff_energy

Compute fast wNN map using differenced energy

Description

Compute fast wNN map using differenced energy

Usage

```
compute_wNN_diff_energy(
  X_nt,
  nt_q = 0.74,
  nn_q = 0.95,
  threshold = c("percentile", "mixture")
)
```

Arguments

X_nt	Numeric matrix (voxels x time).
nt_q	Quantile for neuronal cutoff (default 0.74).
nn_q	Quantile for non-neuronal cutoff (default 0.95).
threshold	One of "percentile" or "mixture".

Value

A list containing wNN, delta, energy, and threshold diagnostics.

extract_caa_oneshot *One-shot CAA extraction in whitened PC subspace*

Description

One-shot CAA extraction in whitened PC subspace

Usage

```
extract_caa_oneshot(Z_kt, n_candidates = 30L)
```

Arguments

Z_kt	Numeric matrix with shape K x T (whitened reduced scores).
n_candidates	Number of candidates to return.

Value

A list with `scores` (T x L) and `predictability` (length L).

`fast_phy_denoise` *Fast design-free PHYCAA+-style denoising*

Description

Fast design-free PHYCAA+-style denoising

Usage

```
fast_phy_denoise(
  x,
  tr,
  design = NULL,
  mask = NULL,
  wnn = NULL,
  wnn_method = c("diff_energy"),
  wnn_threshold = c("percentile", "mixture"),
  delta_nt = 0.74,
  delta_nn = 0.95,
  extractor = c("caa", "dicca", "dipca"),
  lag_order = 1L,
  pca_rank = 100L,
  n_candidates = 30L,
  ratio_thresh = 1,
  pred_thresh = NULL,
  stability = c("none", "half"),
  stability_thresh = 0.3,
  max_passes = 1L,
  return_diagnostics = TRUE,
  svd_engine = c("auto", "rsvd", "svd"),
  use_cpp = TRUE,
  seed = NULL
)
```

Arguments

`x`	Matrix-like data. Default internal orientation is voxels x time.
`tr`	Repetition time in seconds.
`design`	Optional design matrix (time x predictors) used as guardrail.
`mask`	Optional voxel mask (logical or indices).
`wnn`	Optional precomputed wNN vector.
`wnn_method`	Currently supports "diff_energy".
`wnn_threshold`	"percentile" or "mixture".
`delta_nt`	Percentile for neuronal cutoff.
`delta_nn`	Percentile for non-neuronal cutoff.

extractor	One of "caa", "dicca", or "dipca".
lag_order	Dynamic lag order.
pca_rank	Low-rank dimension.
n_candidates	Max candidate components.
ratio_thresh	NN-vs-NT median R2 ratio threshold.
pred_thresh	Optional predictability threshold.
stability	"none" or "half" split-half filter.
stability_thresh	Correlation threshold for split-half stability.
max_passes	Number of denoising passes (1-2 recommended).
return_diagnostics	Whether to return diagnostics.
svd_engine	"auto", "rsvd", or "svd".
use_cpp	Use compiled Rcpp kernels when available.
seed	Optional random seed.

Value

A list with cleaned data, selected regressors, and diagnostics.

`write_qc_artifact` *Write QC artifact to file*

Description

Write QC artifact to file

Usage

```
write_qc_artifact(qc, file)
```

Arguments

qc	QC list from compute_design_free_qc().
file	Path ending in .rds, .csv, or .json.

Value

Invisibly returns file path.

Index