# Package 'nifti.pbcor'

July 31, 2024

**Type** Package

**Title** Parcel-Based Correlation Between NIfTI Images

**Version** 1.0

**Date** 2024-07-03

**Description** Estimate the correlation between two NIfTI images across random parcellations of the images (Fortea et al., under review). This approach overcomes the problems of both voxel-based correlations (neighbor voxels may be spatially dependent) and atlas-based correlations (the correlation may depend on the atlas used).

**License** Artistic License 2.0

**Depends** R (>= 2.10)

**Suggests** oro.nifti

**NeedsCompilation** no

**Author** Joaquim Radua [aut, cre] (<https://orcid.org/0000-0003-1240-5438>)

**Maintainer** Joaquim Radua <quimradua@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-07-31 10:20:16 UTC

## Contents



---

`nifti.pbcor` *Parcel-based correlation between two NIfTI objects*

---

### Description

Calculates the correlation between two NIfTI objects (e.g., brain maps) across random parcels.

## Usage

```
nifti.pbcor(nifti1, nifti2, mask_coords, n.parcels = 90, n.parcellations = 50,
            kmeans.iter.max = 30, kmeans.nstart = 1,
            kmeans.algorithm = "Hartigan-Wong", kmeans.trace = FALSE,
            cor.alternative = "two.sided", cor.method = "pearson",
            cor.exact = NULL, cor.conf.level = 0.95, cor.continuity = FALSE,
            verbose = TRUE)
```

## Arguments

| | |
|---|---|
| `nifti1` | an object of class `"nifti"` with 3D (e.g., the first brain map). |
| `nifti2` | an object of class `"nifti"` with 3D (e.g., the second brain map). |
| `mask_coords` | a mask matrix prepared by nifti.pbcor_mask. |
| `n.parcels` | the number of parcels to divide the mask into. The default is 90, approximately the minimum number of resels in the meta-analytic maps of gray matter differences between individuals with mental disorders and healthy controls in Fortea et al. (see below). |
| `n.parcellations` | the number of parcellations, i.e., how many times the mask will be parcellated. |
| `kmeans.iter.max` | (optional) argument `iter.max` passed to `kmeans` |
| `kmeans.nstart` | (optional) argument `nstart` passed to `kmeans` |
| `kmeans.algorithm` | (optional) argument `algorithm` passed to `kmeans` |
| `kmeans.trace` | (optional) argument `trace` passed to `kmeans` |
| `cor.alternative` | (optional) argument `alternative` passed to `cor.test` |
| `cor.method` | (optional) argument `method` passed to `cor.test` |
| `cor.exact` | (optional) argument `exact` passed to `cor.test` |
| `cor.conf.level` | (optional) argument `conf.level` passed to `cor.test` |
| `cor.continuity` | (optional) argument `continuity` passed to `cor.test` |
| `verbose` | (optional) logical, whether to print some messages during execution. |

## Details

This approach resolves the problems of voxel-based correlations, where contiguous voxels are non-independent, by randomly segmenting the mask (e.g., the brain) into parcels and calculating the Pearson correlation coefficient across these parcels. Additionally, it circumvents the limitations of atlas-based correlations, where estimates depend on the specific atlas used, by repeating the random parcellation and correlation multiple times and selecting the median estimate.

## Value

The parcel-based correlation between the two images across parcellations. The attribute "parcellations.cor.test" has the `cor.test` results obtained in the different random parcellations.

## Author(s)

Joaquim Radua

## References

Fortea et al., under review.

## See Also

nifti.pbcor_mask, readNIfTI

## Examples

```
library(oro.nifti)

# Path of the example files (with large voxels to ensure the example runs quickly)
mask_path = system.file("examples", "mask.nii.gz", package = "nifti.pbcor")
img1_path = system.file("examples", "img1.nii.gz", package = "nifti.pbcor")
img2_path = system.file("examples", "img2.nii.gz", package = "nifti.pbcor")

# Prepare the mask
mask = nifti.pbcor_mask(readNIfTI(mask_path))

# Conduct the parcel-based correlation
nifti.pbcor(readNIfTI(img1_path), readNIfTI(img2_path), mask)
```

---

# nifti.pbcor_mask *Prepare the mask for a parcel-based correlation between NIfTI images*

---

## Description

This function creates the mask required to conduct parcel-based correlations with nifti.pbcor.

## Usage

```
nifti.pbcor_mask(nifti, verbose = TRUE)
```

## Arguments

| | |
|---|---|
| nifti | an object of class "nifti" with 3D (e.g., the brain mask). |
| verbose | (optional) logical, whether to print some messages during execution. |

## Details

This function converts a NIfTI mask into the appropriate format for nifti.pbcor.

## Value

A matrix with the coordinates of the mask

## Author(s)

Joaquim Radua

## See Also

nifti.pbcor, readNIfTI

## Examples

```
library(oro.nifti)

# Path of the example files (with large voxels to ensure the example runs quickly)
mask_path = system.file("examples", "mask.nii.gz", package = "nifti.pbcor")
img1_path = system.file("examples", "img1.nii.gz", package = "nifti.pbcor")
img2_path = system.file("examples", "img2.nii.gz", package = "nifti.pbcor")

# Prepare the mask
mask = nifti.pbcor_mask(readNIfTI(mask_path))

# Conduct the parcel-based correlation
nifti.pbcor(readNIfTI(img1_path), readNIfTI(img2_path), mask)
```

# Index