# BrainSpace: a toolbox for the analysis of macroscale gradients in neuroimaging and connectomics datasets

Reinder Vos de Wael[a,1], Oualid Benkarim[a,1], Casey Paquola[a], Sara Lariviere[a], Jessica Royer[a], Shahin Tavakol[a], Ting Xu[a,b], Seok-Jun Hong[a,b], Sofie L. Valk[d], Bratislav Misic[a], Michael P. Milham[a,b], Daniel S. Margulies[c], Jonathan Smallwood[e,1], Boris C. Bernhardt[a,1,*]

[a]*McConnell Brain Imaging Centre, Montreal Neurological Institute and Hospital, McGill University, Montreal, Canada*
[b]*Center for the Developing Brain, Child Mind Institute; New York, USA*
[c]*Frontlab, Institut du Cerveau et de la Moelle épinière, UPMC UMRS 1127, Inserm U 1127, CNRS UMR 7225, Paris, France*
[d]*Institute for Neuroscience and Medicine 7/Institute of Systems Neuroscience, Forschungszentrum Juelich - Heinrich Heine Universitaet Duesseldorf, Juelich, Germany.*
[e]*Department of Psychology, University of York, Heslington, England, United Kingdom*

## Abstract

Understanding how higher order cognitive function emerges from the underlying brain structure depends on quantifying how the behaviour of discrete regions are integrated within the broader cortical landscape. Recent work has established that this macroscale brain organization and function can be quantified in a compact manner through the use of multivariate machine learning approaches that identify manifolds often described as cortical gradients. By quantifying topographic principles of macroscale organization, cortical gradients lend an analytical framework to study structural and functional brain organization across species, throughout development and aging, and its perturbations in disease. More generally, its macroscale perspective on brain organization offers novel possibilities to investigate the complex relationships between brain structure, function, and cognition in a quantified manner. Here, we present a compact workflow and open-access toolbox that allows for (i) the identification of gradients (from structural or functional imaging data), (ii) their alignment (across subjects or modalities), and (iii) their visualization (in embedding or cortical space). Our toolbox also allows for controlled association studies between gradients with other brain-level features, adjusted with respect to several null models that account for spatial autocorrelation. The toolbox is implemented in both Python and Matlab, programming languages widely used by the neuroimaging and network neuroscience communities. Several use-case examples and validation experiments demonstrate the usage and consistency of our tools

[*] Corresponding author
*Email address:* `boris.bernhardt@mcgill.ca` (Boris C. Bernhardt)
[1] Authors contributed equally to this work.

for the analysis of functional and microstructural gradients across different spatial scales.

*Keywords:* gradients, manifolds, toolbox, repository, open science

## 1. Introduction

Over the last century, neuroanatomical studies in humans and non-human animals have highlighted two complementary features of neural organization. On the one hand, studies have demarcated structurally homogeneous areas with specific connectivity profiles, and ultimately distinct functional roles (e.g., Brodmann, 1909; von Economo and Koskinas, 1925; Palomero-Gallagher and Zilles, 2017; Flechsig, 1920; Glasser et al., 2016). In parallel, neuroanatomists have established that cortical organization may also be characterized in the form of smooth transitions at the system level, for example in terms of their histological properties and connectivity patterns (Von Bonin and Bailey, 1947; Bailey and Von Bonin, 1951; Goulas et al., 2019; Bailey and Von Bonin, 1951; Pandya et al., 2015). In a seminal attempt to synthesize an integrated view of the functions of the mammalian cortical landscape, Marsel Mesulam postulated a hierarchical sensory-fugal axis of cortical microstructural organization and connectivity (Mesulam, 1998). In this model, neural function is hypothesised to emerge not simply from modular neural systems, but through the complex interactions between regions. For example, peripheral neural systems such as primary sensory and motor regions, support functional interactions with the external world in a reasonably direct manner, while transmodal cortices, that take as input signals from regions towards the periphery, support increasingly abstract, and perceptually decoupled, cognitive operations (Mesulam, 1998).

Although much of the more recent work linking measures of neural processing (for example from functional magnetic resonance imaging, MRI) to cognition has focused on identifying discrete regions and modules and their specific functional roles (Eickhoff et al., 2018), recent conceptual and methodological developments have provided both the data and methods that allow macroscale brain features mapped to low dimensional manifold representations, also described as gradients (Margulies et al., 2016). Gradient analyses operating on connectivity data were applied to diffusion MRI tractography data in spefific brain regions (Cerliani et al., 2012; Bajada et al., 2017) as well as resting-state functional MRI connectivity maps (Margulies et al., 2016; Vos de Wael et al., 2018; Larivière et al., 2019; Tian and Zalesky, 2018; Marquand et al., 2017; Haak et al., 2018; Przeździk et al., 2019). Similar techniques have also been used to describe myelin-sensitive tissue measures and other morphological characteristics (Huntenburg et al., 2017; Paquola et al., 2019; Wagstyl et al., 2015). Other studies have used a similar approach to describe task based neural patterns either using meta-analytical co-activation mapping (Vos de Wael et al., 2018) or large-scale functional MRI task data sets (Shine et al., 2019). Gradients have been also successfully derived from non-imaging data that were registered

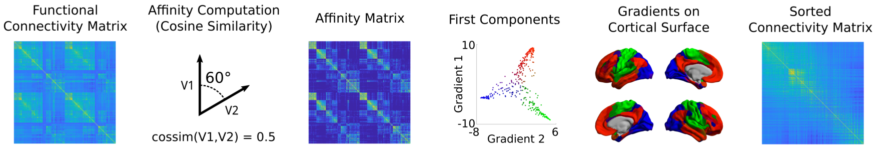
Functional Connectivity Matrix
Affinity Computation (Cosine Similarity)
V1
60°
V2
cossim(V1,V2) = 0.5
Affinity Matrix
First Components
10
Gradient 1
-10
-8
Gradient 2
6
Gradients on Cortical Surface
Sorted Connectivity Matrix

Figure 1: A typical gradient identification workflow. Starting from an input matrix (here, functional connectivity), we use a kernel function to build the affinity matrix (here capturing the connectivity of each seed region). This matrix is decomposed, often via linear rotations or non-linear manifold learning techniques into a set of principal eigenvectors describing axes of largest variance. The scores of each seed onto the first two axes are shown in the scatter plot, with colors denoting position in this 2D space. These colors may be projected back to the cortical surface and the scores can be used to sort the input connectome.

to stereotaxic space, including hippocampal post mortem gene expression information (Vogel et al., 2019) and 3D histology data (Paquola et al., 2019), to explore cellular and molecular signatures of neuroimaging and connectome measures. Core to these techniques is the computation of an affinity matrix that captures inter-area similarity of a given feature followed by the application of dimensionality reduction techniques to identify a gradual ordering of the input matrix in a lower dimensional manifold space (Figure 1).

The ability to describe the brain wide neural activity in a single manifold offers the possibility to understand how the integrated nature of neural processing gives rise to function and dysfunction. Adopting a macroscale perspective on cortical organization has already provided important insights into how cortex-wide patterns relate to cortical dynamics (Wang et al., 2019) and high level cognition (Sormaz et al., 2018; Murphy et al., 2018, 2019; Shine et al., 2019). Furthermore, several studies have leveraged gradients as an analytical framework to describe atypical macroscale organization across clinical conditions, for example, by showing perturbations in functional connectome gradients in autism (Hong et al., 2019) and schizophrenia (Tian et al., 2019). Finally, comparisons of gradients across different imaging modalities have highlighted the extent to which structure directly constrains functional measures (Paquola et al., 2019), while consideration of gradients across species has highlighted how evolution has shaped more integrative features of the cortical landscape (Goulas et al., 2019; Buckner and Krienen, 2013; Huntenburg et al., 2018; Xu et al., 2019; Buckner and Margulies, 2019; Fulcher et al., 2019).

The growth in our capacity to map whole brain cortical gradients, coupled with the promise of a better understanding of how structure gives rise to function, highlights the need for a set of tools that support the analysis of neural manifolds in a compact and reproducible manner. The goal of this paper is to present an open-access set of easy-to-use tools that allow the identification, visualization, and analysis of macroscale gradients of brain organization. We hope this will provide a method for calculating cortical manifolds that facilitates their use in future empirical work, allows comparison between studies, and allows for result replicability. To offer flexibility in implementation, we provide our toolbox in both Python and Matlab, two languages widely used in the neuroimaging and network neuroscience communities. Associated functions are freely available for download and complemented with an expandable online documentation. We anticipate that our toolbox will help researchers interested in studying gradients of cortical organization, and propel further work that establishes the overarching principles through which structural and functional organization of human and non-human brains gives rise to key aspects of human cognition.

## 2. Methodology

### *2.1. Data and code availability statement*

Our toolbox is openly available at: `http://github.com/MICA-MNI/BrainSpace`. The toolbox contains a parallel python and matlab implementation with closely-matched functionality. Along with the code, the toolbox contains several surface models, parcellations across multiple scales, and example data to reproduce the evaluations presented in this tutorial. Additionally, documentation of the proposed toolbox is available for both python and matlab implementations via `http://brainspace.readthedocs.io`.

### *2.2. Input data description*

Our toolbox requires a real input matrix. Let $X \in \mathbb{R}^{n \times p}$ be a matrix aggregating features of several seed regions. In other words, each seed is represented by a $p$-dimensional vector, $\mathbf{x}_i$, built based on the features of the $i$-th seed region, where $X(i,j) = \mathbf{x}_i^j$ denotes the $j$-th feature of the $i$-th seed. In many neuroimaging applications, $X$ may represent a connectivity metric (e.g., resting-state functional MRI connectivity or diffusion MRI tractography derived structural connectivity) between different seed and target brain regions. When seed and target regions are identical, the input matrix $X$ is square. Furthermore, if the connectivity measure used to build the matrix is non-directional, $X$ is also symmetric. If seeds and targets are different, for example when assessing connectivity patterns of a given region with the rest of the brain (Vos de Wael et al., 2018; Haak et al., 2018), we may have that $n \neq p$, resulting in a non-square matrix. The dimensions and symmetry properties of the input matrix $X$ may interact with the dimensionality reduction procedures presented in section 2.4. A simple strategy to make matrices symmetric and squared is to use kernel functions, which will be covered in the following section.

### *2.3. Affinities and kernel functions*

Since we are interested in studying the relationships between the seed regions in terms of their features (e.g., connectivity with target regions), our toolbox provides several kernel functions to compute the relationship between every pair of seed regions and derive a non-negative square symmetric affinity matrix $A \in \mathbb{R}^{n \times n}$, where $A(i,j) = A(j,i)$ denotes the similarity or affinity between seeds $i$ and $j$. Moreover, a square symmetric matrix is a requirement for the next step in our framework (i.e., dimensionality reduction). Note that when the input matrix $X$ is already square and symmetric (e.g., seed and target regions are the same), there may be no need to derive the affinity matrix and $X$ can be used directly to perform dimensionality reduction. Accordingly, BrainSpace provides the option of skipping this step and using the input matrix as the affinity.

There are numerous approaches to build affinity matrices. Our toolbox currently implements the following kernels: Gaussian, cosine similarity, normalized angle similarity, Pearson's correlation coefficient, and Spearman rank order correlations. For simplicity, let $\mathbf{x} = \mathbf{x}_i$ and $\mathbf{y} = \mathbf{x}_j$, these kernels can be expressed as follows:

1. Gaussian kernel:
$$A(i,j) = e^{-(\gamma\|\mathbf{x}-\mathbf{y}\|^2)},$$
where $\gamma$ is the inverse kernel width and $\|\cdot\|$ denotes the $l_2$-norm.
2. Cosine similarity:
$$A(i,j) = cossim(\mathbf{x},\mathbf{y}) = \frac{\mathbf{x}\mathbf{y}^T}{\|\mathbf{x}\|\|\mathbf{y}\|},$$
where $cossim(\cdot,\cdot)$ is the cosine similarity function and $T$ stands for transpose.
3. Normalized angle similarity:
$$A(i,j) = 1 - \frac{\cos^{-1}(cossim(\mathbf{x},\mathbf{y}))}{\pi}.$$
4. Pearson correlation:
$$A(i,j) = \rho(\mathbf{x},\mathbf{y}) = cossim(\mathbf{x}-\bar{\mathbf{x}},\mathbf{y}-\bar{\mathbf{y}}),$$
where $\rho$ is the Pearson correlation coefficient, and $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ denote the means of $\mathbf{x}$ and $\mathbf{y}$, respectively.
5. Spearman rank order correlation:
$$A(i,j) = \rho(\mathbf{r}_x,\mathbf{r}_y),$$
where $\mathbf{r}_x$ and $\mathbf{r}_y$ denote the ranks of $\mathbf{x}$ and $\mathbf{y}$, respectively.

While not being exhaustive, the first version of BrainSpace thus includes commonly used kernels in the gradient literature and additional ones for experimentation. To our knowledge, no gradient paper has used Pearson or Spearman correlation. Note that if $X$ is already row-wise demeaned, Pearson correlation amounts to cosine similarity. The Gaussian kernel is widely used in the machine learning community (for example in the context of Laplacian eigenmaps and support vector machines), which provides a simple approach to convert Euclidean distances between our seeds into similarities. Cosine similarity, [example application (Margulies et al., 2016)], computes the angle between our feature vectors to describe their similarity. It is important to note that cosine similarity ranges between -1 and 1, with negative correlations to be transformed to non-negative values. This motivated the inclusion of the normalized angle kernel, [example application: (Vos de Wael et al., 2018)], as it was developed to circumvent negative similarities by transforming the similarities to $\left[0,1\right]$, with 1 denoting identical angles, and 0 opposing angles. Still, cosine similarity and Pearson and Spearman correlation coefficients may produce negative values (i.e., $\left[-1,1\right]$) if the feature vectors are negatively correlated. A simple approach to deal with this issue is to set negative values to zero. Additionally, BrainSpace provides an option for column-wise thresholding of the input matrix as done previously (Margulies et al., 2016; Vos de Wael et al., 2018; Paquola et al., 2019; Hong et al., 2019). Each vector of the input matrix is thresholded at a given sparsity (e.g., by keeping the weights of the top 10% entries for each region). This procedure ensures that only strong, and potentially less noisy, connections contribute to the gradient.

In addition to the aforementioned kernels, BrainSpace provides the option of skipping this step and using the input matrix as an affinity matrix as well as the option to provide a custom kernel.

### *2.4. Dimensionality reduction*

Recall that in the input matrix, each seed $\mathbf{x}_i$ is defined by a $p$-dimensional feature vector, where $p$ may denote hundreds of parcels or thousands of vertices/voxels. The aim of dimensionality reduction techniques is to find a meaningful underlying low-dimensional representation, $\mathcal{G} \in \mathbb{R}^{n\times m}$ with $m \ll p$, hidden in the high-dimensional ambient space. These methods can be grouped into linear and non-linear techniques. The former use a linear transformation to unravel the latent representation, while techniques in the second category use non-linear transformations. BrainSpace provides three of the most widely used dimensionality reduction techniques for macroscale gradient mapping: principal component analysis (PCA) for linear embedding, and laplacian eigenmaps (LE) and diffusion mapping (DM) for non-linear dimensionality reduction.

1. PCA is a linear approach that transforms the data to a low-dimensional space represented by a set of orthogonal components that explain maximal data variance. Given a column-wise demeaned version of the input matrix $X_d$, the low-dimensional representation is computed as follows:

$$\mathcal{G}_{PCA} = US^T, \tag{1}$$

where $U$ are the left singular vectors and $S$ a diagonal matrix of singular values obtained after factorizing the input matrix using singular value decomposition (SVD), $X_d = USV^T$. Note that, although here we present the SVD version, PCA can also be performed based on the eigendecomposition of the covariance matrix of $X$, for example.

2. LE is a non-linear dimensionality reduction technique that uses the graph Laplacian of the affinity matrix $A$ to perform the embedding:

$$L = D - A, \tag{2}$$

where the degree matrix $D$ is a diagonal matrix defined as $D(i, i) = \sum_j A(i, j)$ and $L$ is the graph Laplacian matrix. Note that we can also work with its normalized version instead $L_s = D^{-1/2}LD^{-1/2}$ (Ng et al., 2001). LE then proceeds to solve the generalized eigenvalue problem:

$$L\mathbf{g} = \lambda D\mathbf{g}, \tag{3}$$

where the eigenvectors $\mathbf{g}_k$ corresponding to the $m$ smallest eigenvalues $\lambda_k$ (excluding the first eigenvalue) are used to build the new low-dimensional representation:

$$\mathcal{G}_{LE} = [\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_m]. \tag{4}$$

3. DM also seeks a non-linear mapping of the data based on the diffusion operator $P_\alpha$, which is defined as follows:

$$P_\alpha = D_\alpha^{-1} W_\alpha, \tag{5}$$

where $\alpha \in \left[0, 1\right]$ is the anisotropic diffusion parameter used by the diffusion operator, $W_\alpha = D^{-1/\alpha}AD^{-1/\alpha}$ is built by normalizing the affinity matrix according to the diffusion parameter and $D_\alpha$ is the degree matrix derived from $W_\alpha$. When $\alpha = 0$, the diffusion amounts to normalized graph Laplacian on isotropic weights, for $\alpha = 1$, it approximates the Laplace-Beltrami operator and for the case where $\alpha = 0.5$ it approximates the Fokker-Planck diffusion (Coifman and Lafon, 2006). It controls the influence of the density of sampling points on the manifold ($\alpha = 0$, maximal influence; $\alpha = 1$, no influence). In the gradient literature, the anisotropic diffusion hyper-parameter is commonly set to $\alpha = 0.5$ (Margulies et al., 2016; Vos de Wael et al., 2018; Larivière et al., 2019), a choice that retains the global relations between data points in the embedded space.
Similarly to LE, DM computes the eigenvalues and eigenvectors of the diffusion operator. However, in this case, the new representation is constructed with the scaled eigenvectors corresponding to the largest eigenvalues, after omitting the eigenvector with the largest eigenvalue:

$$\mathcal{G}_{DM} = [\lambda_1^t \mathbf{g}_1, \lambda_2^t \mathbf{g}_2, \ldots, \lambda_m^t \mathbf{g}_m], \tag{6}$$

where $t$ is the time parameter that represent the scale.

### *2.5. Alignment of gradients*

Gradients computed separately for two or more datasets (e.g., patients vs controls, left vs right hippocampi) may not be directly comparable due to different eigenvector orderings in case of eigenvalue multiplicity (i.e., eigenvalues with the same value) and sign ambiguity of the eigenvectors (Lombaert et al., 2013). Aligning gradients improves comparability and correspondence. However, we recommend always visually inspecting the output of gradient alignments; if the manifold spaces are substantially different then alignments may not provide sensible output. In the first version of the BrainSpace toolbox, gradients can be aligned using Procrustes analysis (Langs et al., 2015) or implicitly by joint embedding.

*Procrustes analysis*

Briefly, given a source $\mathcal{G}_s$ and a target $\mathcal{G}_t$ representations, Procrustes analysis seeks an orthogonal linear transformation $\psi$ to align the source representation to the target, such that $\psi(\mathcal{G}_s)$ and $\mathcal{G}_t$ are superimposed. Translation and scaling can also be performed by initially centering and normalizing the data prior to finding the transformation. For multiple datasets, a generalized Procrustes analysis is employed. Let $\mathcal{G}_k, k = 1, 2, \ldots, N$ be the low-dimensional representations of $N$ different datasets (i.e., input matrices $X_k$). The procedure proceeds iteratively by aligning all representations $\mathcal{G}_k$ to a reference and updating the reference $\mathcal{G}_R = \frac{1}{N}\sum_k \psi(\mathcal{G}_k)$ by averaging the aligned representations. In the first iteration, the reference can be chosen from the available representations, or an out-of-sample template can be provided (e.g., from a hold-out group).

*Joint embedding*

Joint embedding is a dimensionality reduction technique that finds a common underlying representation of multiple datasets by using a simultaneous embedding (Xu et al., 2019). The main challenge of this technique is to find a meaningful approach to establish correspondences between the original datasets (i.e., $X$). In the current version of BrainSpace, joint alignment is implemented based on spectral embedding and it is available for LE and DM. The only difference with these methods is that the embedding, rather than using the affinity matrices individually, is based on the joint affinity matrix $\mathcal{J}$, which is built as:

$$\mathcal{J} = \begin{pmatrix} A_1 & A_{12} & \cdots & A_{1N} \\ A_{12}^T & A_2 & \cdots & A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1N}^T & A_{2N}^T & \cdots & A_N \end{pmatrix}, \tag{7}$$

where $A_k$ is the intra-dataset affinity of input matrix $X_k$ and $A_{ij}$ is the inter-dataset affinity between $X_i$ and $X_j$. As of the current version, both sets of affinities are built using the same kernel. It is important to note, therefore, that joint embedding can only be used if the input matrices have the same features (e.g., identical target regions). After the embedding, the resulting shared representation $\mathcal{G}_J = [\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_N]^T$ will be composed of $N$ individual low-dimensional representations, such that for the $k$-th input matrix $X_k \in \mathbb{R}^{n_k \times p}$, the corresponding representation is $\mathcal{G}_k \in \mathbb{R}^{n_k \times m}$, where $n_k$ is the number of seeds (i.e., rows) of $X_k$.

### *2.6. Null models*

Many researchers may use gradients to compare axes of the brain to other continuous brain markers such as cortical thickness measures or estimates of myelination. Given the spatial autocorrelation present in many modalities, a linear regression or similar methods may provide biased test statistics. To circumvent this issue, we recommend comparing the observed test statistic to those of a set of distributions with similar spatial autocorrelation. To this end we provide two methods: spin permutations (Alexander-Bloch et al., 2018) and Moran spectral randomization (MSR) (Wagner and Dray, 2015; Dray, 2011). In cases where the input data lies on a surface and most of the sphere is used, we recommend using spin permutation otherwise we recommend MSR. When performing a statistical test with multiple gradients as either predictor or response variable, we recommend randomizing the non-gradient variable as these randomizations need not maintain statistical independence across different eigenvectors.

*Spin permutations.*

Spin permutation analysis leverages the spherical representations of the cerebral cortex, such as those derived from FreeSurfer (Fischl, 2012) or CIVET (Kim et al., 2005), to address the problem of spatial autocorrelation in statistical inference. In short, spin permutations estimate the null distribution by randomly rotating the spherical projections of the cortical surface while preserving the spatial relationships within the data (Alexander-Bloch et al., 2018). Let $V \in \mathbb{R}^{l\times 3}$ be the matrix of vertex coordinates in the sphere, where $l$ is the number of vertices, and $R \in \mathbb{R}^{3\times 3}$ a matrix representing a rotation along the three axes uniformly sampled from all possible rotations (Blaser and Fryzlewicz, 2016). The rotated sphere $V_r$ is computed as follows:

$$V_r = VR. \tag{8}$$

Samples of the null distribution are then created by assigning each vertex on $V_r$ the data of its nearest neighbor on $V$.

*Moran spectral randomization.*

Borrowed from the ecology literature, MSR can also be used to generate random variables with identical or similar spatial autocorrelation (in terms of Moran's $I$ correlation coefficient (Cliff and Ord, 1973)). This approach requires building a spatial weight matrix defining the relationships between the different locations. For our particular case, given a surface mesh with $l$ vertices (i.e., locations), its topological information is used to build the spatial weight matrix $L \in \mathbb{R}^{l\times l}$, such that $L(i, j) > 0$ if vertices $i$ and $j$ are neighbors, and $L(i, j) = 0$ otherwise. In BrainSpace, $L$ is built using the inverse distance between each vertex and the vertices in its immediate neighborhood, although other neighborhoods and weighting schemes (e.g., binary or Gaussian weights) could be used. Then, $L$ is doubly centered and eigendecomposed into its whole spectrum, with the resulting eigenvectors $\mathcal{M} \in \mathbb{R}^{l\times(l-1)}$ being the so-called Moran eigenvector maps. Note that the eigenvector with 0 eigenvalue is dropped. The advantage of MSR is that we can work with the original cortical surfaces, and thus skip the potential distortions introduced by the spherical mesh parameterization.

In order to generate the null distributions, let $\mathbf{u} \in \mathbb{R}^l$ be an input feature vector defined on each vertex of our surface (e.g., cortical thickness), and $\mathbf{r} \in \mathbb{R}^{l-1}$ the correlation coefficients of $\mathbf{u}$ with each spatial eigenvector in $\mathcal{M}$. MSR aims to find a randomized feature vector $\mathbf{z}$ that respects the autocorrelation observed in $\mathbf{u}$ as follows:

$$\mathbf{z} = \bar{\mathbf{u}} + \sigma_u \sqrt{l-1} \mathcal{M} \mathbf{a}^T, \tag{9}$$

where $\bar{\mathbf{u}}$ and $\sigma_u$ stand for mean and standard deviation of $\mathbf{u}$ respectively, and $\mathbf{a} \in \mathbb{R}^{l-1}$ is a vector of random coefficients. Three different methods exist for generating vector $\mathbf{a}$: singleton, pair, and triplet procedures. Currently, only the singleton and pair procedures are supported in BrainSpace. Let $\mathbf{v} = \mathcal{M}\mathbf{r}^T$, in the singleton procedure, $\mathbf{a}$ is computed by randomizing the sign of each element in $\mathbf{v}$ (i.e., $\mathbf{a}_i = \pm \mathbf{v}_i$). In the pair procedure, the elements of $\mathbf{v}$ are randomly changed in pairs. Let $(\mathbf{v}_i, \mathbf{v}_j)$ be a pair of elements randomly chosen, then $\mathbf{a}$ is updated such that $\mathbf{a}_i = q_{ij}\cos(\phi)$ and $\mathbf{a}_j = q_{ij}\sin(\phi)$, with $q_{ij} = \sqrt{\mathbf{v}_i^2 + \mathbf{v}_j^2}$ and $\phi \sim \mathcal{U}(0, 2\pi)$ randomly drawn from a uniform distribution. If the number of elements of $\mathbf{v}$ is odd, the singleton procedure is used for the remaining element.

As opposed to the singleton procedure, the null data generated by the pair procedure does not fully preserve the observed spatial autocorrelation. We therefore recommend the singleton procedure, unless the number of required randomizations exceeds $2^{l-1}$, which is the maximum number of unique randomizations that can be produced using the singleton procedure.

## 3. Examples and evaluations

In this section, we demonstrate the usage of our toolbox through several examples on gradient mapping and null model generation. Evaluations are based on a subset of the Human Connectome Project (HCP) dataset (Van Essen et al., 2013), as in prior work (Vos de Wael et al., 2018). Matlab code is presented in the main version of the paper. The python version is available in Appendix A.

### *3.1. Generating gradients*

To first illustrate the basic functionality of the toolbox, we computed gradients derived from resting-state functional MRI functional connectivity (FC). In short, the input matrix was made sparse (to 10% sparsity) and a cosine similarity matrix was computed. Next, we applied the three different manifold algorithms (i.e., PCA, LE, DM) and plot their first and second gradients on the cortical surface (Sample Code 1). Resulting gradients (Figure 2) derived from all dimensionality reduction techniques resemble those published previously (Margulies et al., 2016), although for PCA the somatomotor to visual gradient explains more variance than the default mode to sensory gradient.

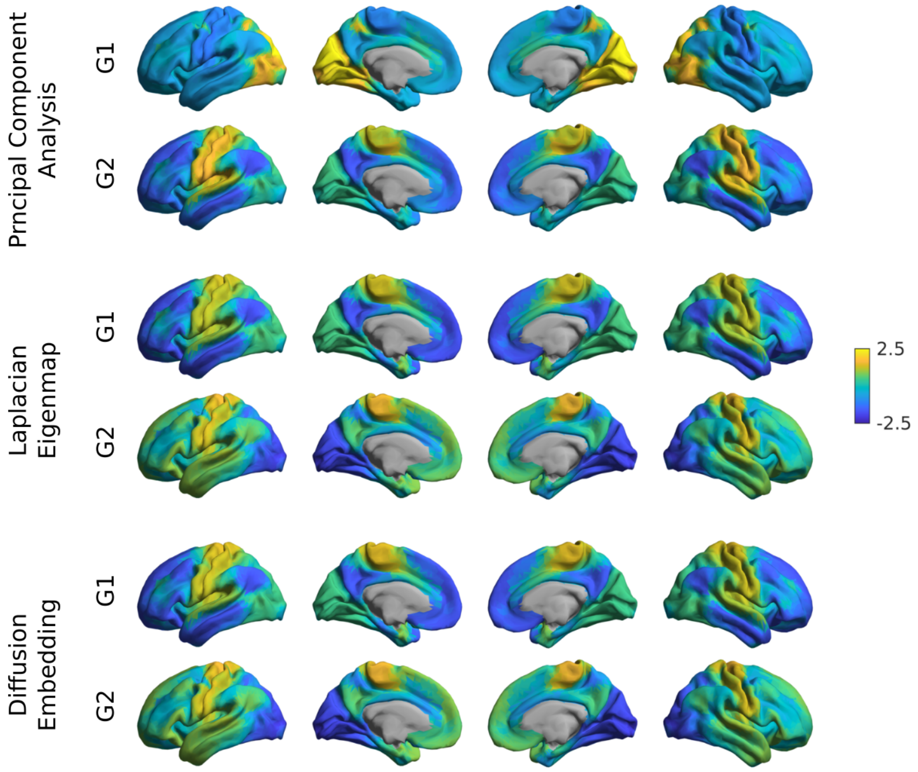

Figure 2: Gradient construction with different dimensionality reduction techniques. Gradient 1 (G1) and 2 (G2) of FC were computed using a cosine similarity affinity computation, followed by either PCA, LE, or DM. Gradients were z-scored before plotting.

Sample Code 1: A minimal Matlab example for plotting the first gradient of an input data matrix on the cortical surface. Equivalent Python code is provided in sample code 5 in Appendix A.

```
% Create a GradientMaps object
G = GradientMaps('kernel','cosine', 'approach','dm');

% Apply GradientMaps to the data
G = G.fit(data_matrix);

% Load surfaces
left_surface = read_surface('left_surface_file.obj');
right_surface = read_surface('right_surface_file.obj');

% Plot first gradient on the cortical surface
plot_hemispheres(G.gradients{1}(:,1), {left_surface,right_surface})
```

### *3.2. Aligning gradients*

#### *a) Gradient alignment across modalities*

Based on subjects present in both the FC dataset as well as those used in the validation group of (Paquola et al., 2019), we assessed the correspondence between gradients computed from different modalities and evaluated increases in correspondence when gradient alignment was used. The two modalities that were evaluated are FC and microstructural profile covariance (MPC). Here we compare gradients of the two modalities in an unaligned form, as well as after Procrustes alignment and joint embedding (Sample Code 2, Figure 3). Gradient correspondence increases following Procrustes alignment and a marked increase following joint embedding. Beyond maximizing correspondence, the use of Procrustes vs joint embedding can also depend on the specific applications. Procrustes alignment preserves the overall shape of the different gradients and can thus be a preferable approach to compare different gradients. Joint embedding, on the other hand, identifies a joint solution that maximizes their similarity, so the resulting gradient may be more of a ’hybrid’ of the input manifolds. Joint embedding is thus a technique to identify correspondence and map from one space to another, and is conceptually related to widely used multivariate associative techniques such as canonical correlation analysis or partial least squares which seek to maximize the linear associations between two multidimensional datasets (Mcintosh and Misic, 2013). Note that the computational cost of joint embedding is substantially higher, so Procrustes analysis may be the preferred option when computational resources are a limiting factor.

Sample Code 2: A minimal Matlab example for creating and plotting gradients from different modalities, with different alignments. Equivalent Python code is provided in sample code 6 in Appendix A.

```matlab
% Create two GradientMaps objects with different alignments
Gp = GradientMaps('kernel','cosine', 'approach','dm', ...
                  'alignment','procrustes');
Gj = GradientMaps('kernel','cosine', 'approach','dm', ...
                  'alignment','joint');

% Apply GradientMaps to the data
Gp = Gp.fit({mpc,fc});
Gj = Gj.fit({mpc,fc}});

% Load surfaces
left_surface = read_surface('left_surface_file.obj');
right_surface = read_surface('right_surface_file.obj');

% Plot first MPC gradient of Procrustes alignment
plot_hemispheres(Gp.aligned{1}(:,1), {left_surface,right_surface});

% Plot first FC gradient of joint embedding
plot_hemispheres(Gj.aligned{2}(:,1), {left_surface,right_surface});
```

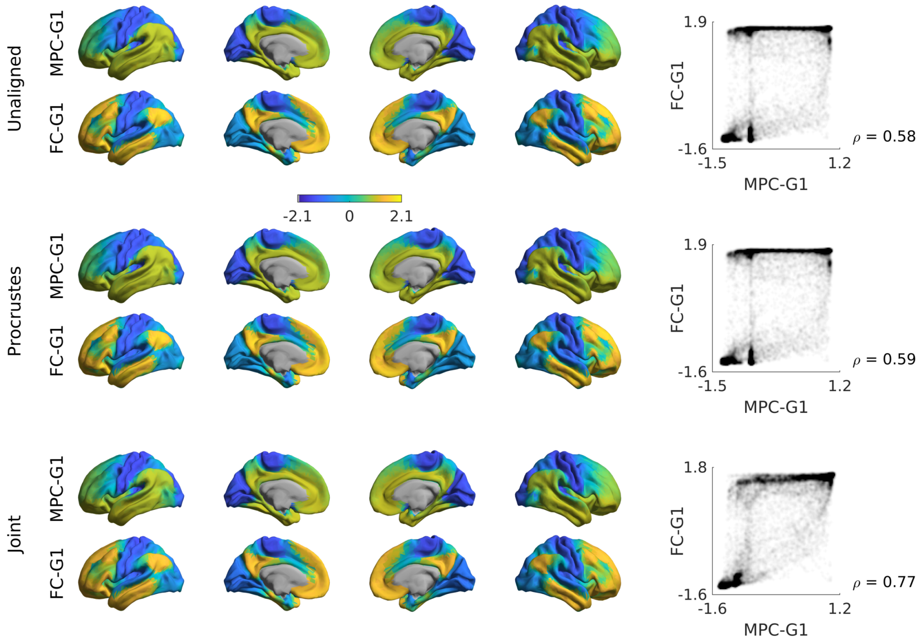

Figure 3: Comparison of alignment methods across modalities. Unaligned gradients 1 (top) of MPC and FC were derived using cosine similarity and diffusion embedding. Alignments using Procrustes analyses (middle) and joint embedding (bottom) are also shown. Smoothed scatter plots show a moderate increase in Spearman correlation after joint embedding. Gradients were z-scored before plotting.

### *b) Gradient alignment across individuals*

Researchers may also be interested in comparing gradient values between individuals (Hong et al., 2019), for example to assess perturbations in FC gradients as a measure of brain network hierarchy. One possible approach could be to first build a group-level gradient template, to which both diagnostic groups are aligned using Procrustes rotation. After that, the two groups can be compared statistically and at each vertex, for example using the SurfStat toolbox (Worsley et al., 2009) or alternative tools. In the example below, we present the steps to derive, align, and compare the principal functional gradient between a cohort of individuals. We computed a template gradient from an out-of-sample dataset of 134 subjects from the HCP dataset (the validation cohort used by (Vos de Wael et al., 2018)). Next, we used Procrustes analysis to align individual's gradients of each subject to the group level template (Sample Code 3, Figure 4).

Sample Code 3: A minimal Matlab example for aligning the gradients of two individuals to a template gradient. Equivalent Python code is provided in sample code 7 in Appendix A.

```
% Create a GradientMaps object for the template
Gt = GradientMaps('kernel','cosine', 'approach','dm');
```

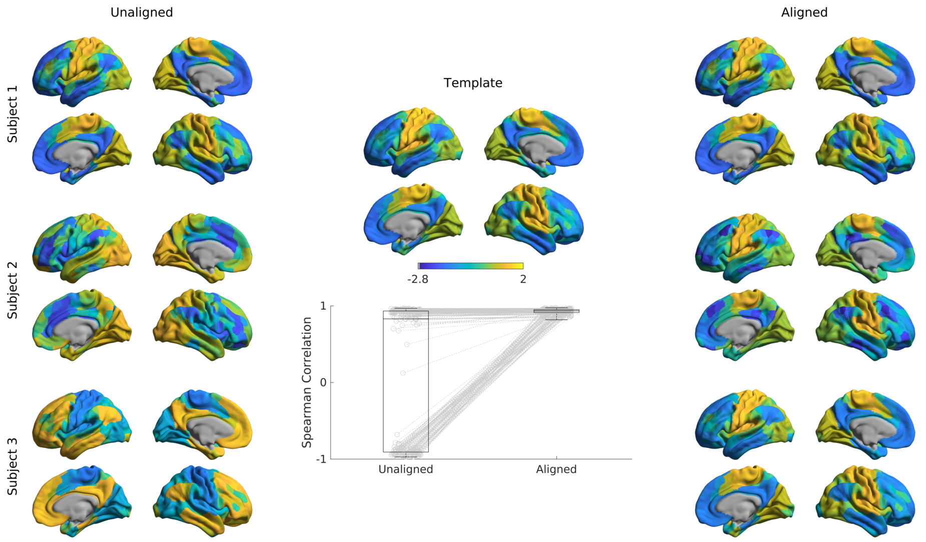

Figure 4: Alignment of subjects to a template. Gradients 1 of single subjects computed with the cosine similarity kernel and diffusion embedding manifold (left) were aligned to an out-of-sample template (middle) using Procrustes Analysis, creating the aligned gradients (right). Box-plot shows the Spearman correlations of each subjects' gradient 1 to the template gradient 1 both before, and after Procrustes alignment

```
% Apply GradientMaps to template data
Gt = Gt.fit(template_data);

% Create a GradientMaps object for the individuals
Gs = GradientMaps('kernel','cosine', 'approach','dm', ...
                 'alignment','procrustes');

% Compute gradients of all subjects and align to template
Gs = Gs.fit({subject1_data,subject2_data}, ...
            'reference',Gt.gradients{1});

% Load surfaces
left_surface = read_surface('left_surface_file.obj');
right_surface = read_surface('right_surface_file.obj');

% Plot the first aligned gradient for subject 2
plot_hemispheres(Gs.aligned{2}(:,1), {left_surface,right_surface});
```

### *3.3. Gradients across different spatial scales*

The gradients presented so far were all derived at a vertex-wise level, which requires considerable computational resources. To minimize time and space requirements and to make results comparable to other parcellation-based studies, some users may be interested in deriving gradients using parcellated data. To illustrate the effect of using different parcellations, we repeated the gradient identification analysis across different spatial scales for both a structural and a functional parcellation. Specifically, we subdivided the conte69 surface into 400, 300, 200, and 100 parcels based on both a clustering of a well-established anatomical atlas (Desikan et al., 2006), as well as a recently published local-global functional clustering (Schaefer et al., 2017) and built FC gradients from these representations (Figure 5). These parcellations and subsampling schemes are provided in the *shared* folder of the BrainSpace toolbox.

Overall, with increasing spatial resolution the FC gradients become more pronounced and gradients derived from functional and structural parcellations were more similar. At a scale of 200 nodes or lower, putative functional boundaries may not be as reliably captured when using anatomically-informed parcellations, resulting in more marked alterations in the overall shape of the gradients.

For further evaluation, we related the above gradients across multiple scales relative to Mesulam's classic scheme of laminar differentiation and hierarchy (Paquola et al., 2019). It shows clear correspondence between the first gradient and the Mesulam hierarchy for high resolution data of 300 nodes and more, regardless of the parcellation scheme. While high correspondence was still seen for functional parcellations at lower granularity, it was markedly reduced for structural. For researchers interested in using Mesulam's parcellation, it has been provided on Conte69 surfaces in the *shared* folder of the BrainSpace toolbox. We also provide the evaluated FC and MPC gradients across these different spatial scales. Such gradients can be used to, for example stratify other imaging measures, including functional activation and connectivity patters (Hong et al., 2019; Murphy et al., 2018), meta-analytical syntheses (Murphy et al., 2019; Margulies et al., 2016), cortical thickness measures or Amyloid-beta PET uptake data (Lowe et al., 2019).

### *3.4. Null models*

Here we present an example to assess the significance of correlations between FC gradients and data from other modalities (cortical thickness and T1w/T2w image intensity in this example). We present code to generate previously proposed spin tests (Alexander-Bloch et al., 2018), which preserve the auto-correlation of the permuted feature(s) by rotating the feature data on the sphere. In our example (Figure 6), one can clearly see that correlations between FC gradients and T1w/T2w stay highly significant even when comparing the correlation to 1000 null models whereas correlations between FC gradients and cortical thickness appear non-significant.

Sample Code 4: A minimal Matlab example for building null models based on spin tests. Equivalent Python code is provided in sample code 8 in Appendix A.

```
% Load spheres
left_sphere = read_surface('left_sphere_file.obj');
```

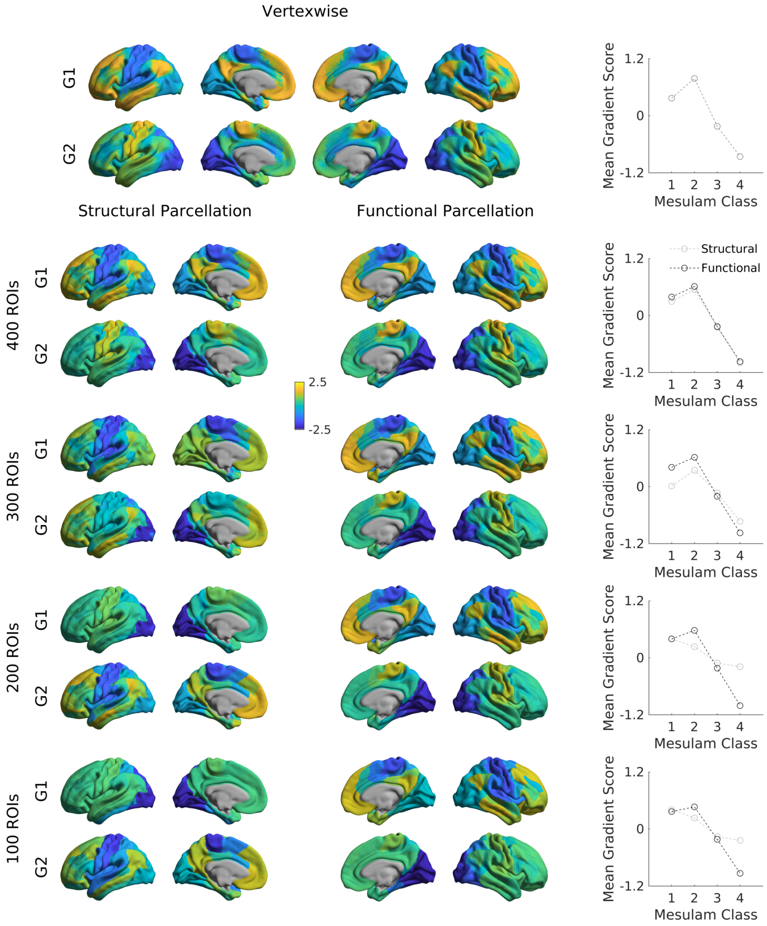

Figure 5: Functional gradients across spatial scales. The cortex was subdivided into 100 (first row), 200 (second row), 300 (third row), and 400 (fourth row) regions of interest based on an anatomical (left) and functional (right) parcellation. Displayed are gradients 1 (G1), and 2 (G2), each for one hemisphere only. Line plots show the average gradient score within each Mesulam class for the functional (dark gray) and structural (light gray).

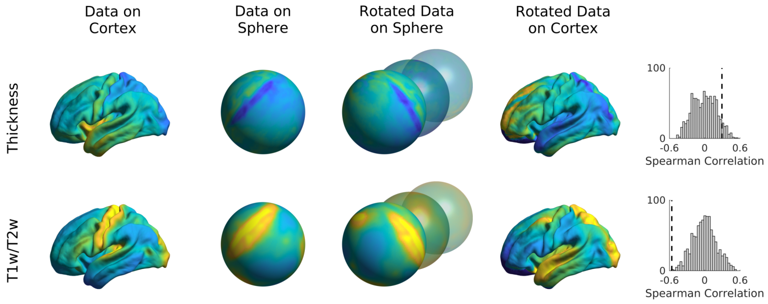

Figure 6: Spin tests of cortical thickness and t1w/t2w intensity. Data were rotated on the sphere 1000 times and the Spearman correlation between FC gradient 1 and the rotated data were computed. Distribution of correlations are shown in the histograms with the dashed lines denoting the true correlation.

```
right_sphere = read_surface('right_sphere_file.obj');

% Generate 1000 spin permutations
n_perm = 1000;
features_spin = spin_permutations({left_features,right_features}, ...
        {left_sphere,right_sphere},n_perm);
```

## 4. Discussion

While tools for unsupervised manifold identification and their alignment are widely available and extensively used in data science across multiple research domains[2], and while some prior studies made their workflow openly accessible (see (Margulies et al., 2016; Haak et al., 2018; Guell et al., 2019)[3][4][5]) we currently lack a unified software package that incorporates the major steps of gradient construction and evaluation for neuroimaging and connectome analysis datasets. We aimed to fill this gap with BrainSpace, a compact toolbox for the identification and analysis of low-dimensional gradients for any given regional or connectome-level feature. As such, the toolbox provides a simple entry point for researchers interested in studying gradients as windows into brain organization and function. Tools are available in Python and Matlab, two widely used programming languages in the neuroimaging and connectomics communities, and openly shared via Github at `http://github.com/MICA-MNI/BrainSpace`. In addition to the theoretical and practical guidelines provided here, an expandable documentation has been published at `http://brainspace.readthedocs.io`, providing further guidance and use-case examples. BrainSpace is a simple and

[2] https://github.com/scikit-learn/scikit-learn
[3] https://github.com/satra/mapalign
[4] https://xaviergp.github.io/littlebrain/
[5] https://github.com/koenhaak/congrads

modular package accessible to beginners, yet expandable for advanced programmers. Core to the toolbox is a simple object-oriented model allowing for flexible computation of different i) affinity matrices, ii) dimensionality reduction techniques, iii) alignment functions, and iv) null models. As a side, we supplied precomputed gradients, a novel subparcellation of the Desikan Killani atlas, and a literature-based atlas of cortical laminar differentiation that we used in a recent study (Paquola et al., 2019; Mesulam, 1998).

As the main purpose of this report was to provide an accessible introduction of the toolboxs basic functionality, we focused on tutorial examples and several selected assessments that demonstrate more general aspects of gradient analyses First, we observed relatively consistent FC gradients were produced by multiple different dimensionality reduction techniques supplied (i.e., PCA, LE, DME), we observed relatively consistent gradients, at least when cosine similarity was chosen for affinity matrix computations. Potential interaction between input modality, affinity matrix formulation, and dimensionality reduction techniques may nevertheless occur and be worthwhile to further explore, as this might also help to understand potential differences in results. Second, while we could show a relative consistency of FC gradients across spatial scales in the case of vertex-wise analyses and when parcellations with 300 nodes or more were used, we observed an interaction between the type of input data and parcellation-substrate at lower spatial scales. In fact, lower resolution structural parcellations might not capture fine-grained functional boundaries, specifically in heteromodal and paralimbic association cortices which may be less constrained by underlying structural features (Paquola et al., 2019). It will be informative for future work to clarify how input modality (e.g. FC, MPC,or diffusion MRI), the choice of parcellation, and the spatial scale impact on the spatial features identified by gradient analyses.

There are two broad ways through which the gradient method, as well as the Brainspace toolbox, may improve our understanding of neural organization and its associated functions. One avenue is the identification of similarities and differences in gradients derived from different brain measures. To address associations between cortical microstructure and macroscale function, a previous study (Paquola et al., 2019) demonstrated that gradients derived from 3D histology and myelin-sensitive MRI measures show both similarities and differences from those derived from resting state-state functional connectivity analysis (Margulies et al., 2016). This raises the possibility that the gradient method may help quantify common and distinct influences on functional and structural macroscale organization and shed light on the neural basis of more flexible (i.e., less structurally constrained) aspects of human cognition (Paquola et al., 2019).Another way that gradients can inform our understanding of how functions emerge from the cortex is through the analysis of how macroscale patterns of organization change in disease. One recent study (Hong et al., 2019), for example, demonstrated differences in the principal functional gradient, identified by Margulies et al. (2016), between individuals with autism spectrum disorder and typically developing controls. In this way, manifold-derived gradient analyses hold the possibility to characterize how macroscale functional organization may become dysfunctional in atypical neurodevelopment. To achieve both of these goals, it is important to consider gradient alignment of different gradients, and this can be achieved by a number of methods we supplied in the toolbox, such as Procrustes rotation. This allows researchers to both compare gradients from different modalities and to homogenize measures across subjects, while minimizing the changes to individual manifolds. Our across-subjects evaluations highlighted an increase in correspondence between individual subjects and the template manifold when Procrustes alignment was used compared to unaligned approaches, mainly driven by reasonably trivial changes in gradient, such as a change in the sign of specific gradients in a subgroup of subjects. As an alternative to the Procrustes alignment, it is also possible to align gradients by applying joint manifold alignments, often referred to as joint-embeddings (Xu et al., 2019).The rotations provided by embedding alignments can augment both cross-subject (Nenning et al., 2017) as well as cross-species analyses  (Xu et al., 2019). Of note, this joint embedding technique generates a new manifold from the mapping between different gradients, which may result in new solutions that do not fully correspond to the initial gradients in important, and in a potentially important way.

When assessing the significance of correlations between gradients and other features of brain organization, there is an increasing awareness to ideally also evaluate correlations against null models with a similar spatial autocorrelations as the the original features. In our toolbox, we present two different approaches for building such null models, including a spin permutation test that is an adaptation of a previously released approach (Alexander-Bloch et al., 2018) and Moran's spectral randomization used in prior ecological studies (Cliff and Ord, 1973). Gradients can also be used as a coordinate system, and stratify cortical features that are not per se gradient-based. Examples include surface-based geodesic distance measures from sensory-motor regions to other regions of cortex (Margulies et al., 2016), task-based functional activation patterns and meta-analytical data (Murphy et al., 2018, 2019; Margulies et al., 2016), as well as MRI-based cortical thickness and PET-derived amyloid beta uptake measures (Lowe et al., 2019). As such, using manifolds as a new coordinate system (Huntenburg et al., 2018) may complement widely used parcellation approaches (Schaefer et al., 2017; Yeo et al., 2011; Glasser et al., 2016) and be can be useful for dimensionality reduction of findings and for the interpretation and communication of results.

## Acknowledgements

Mr. Vos de Wael was funded by the Savoy Foundation for Epilepsy Research. Dr. Oualid Benkarim was funded by a Healthy Brains for Healthy Lives (HBHL) postdoctoral fellowship. Dr. Jessica Royer was supported by a Canadian Open Neuroscience Platform (CONP) fellowship. Dr Smallwood was supported by the European Research Council (WANDERINGMINDS – ERC646927). Dr. Paquola was funded through a postdoctoral fellowship of the Transforming Autism Care Consortium (TACC) and the Fonds de la recherche du Québec

- Santé (FRQ-S). Dr. Boris Bernhardt acknowledges research support from the National Science and Engineering Research Council of Canada (NSERC, Discovery-1304413), the Canadian Institutes of Health Research (CIHR, FDN-154298), the Azrieli Center for Autism Research of the Montreal Neurological Institute (ACAR), SickKids Foundation (NI17-039), and received salary support from FRQS (Chercheur Boursier Junior 1).

## References

Alexander-Bloch, A.F., Shou, H., Liu, S., Satterthwaite, T.D., Glahn, D.C., Shinohara, R.T., Vandekar, S.N., Raznahan, A., 2018. On testing for spatial correspondence between maps of human brain structure and function. Neuroimage 178, 540–551.

Bailey, P., Von Bonin, G., 1951. The isocortex of Man. University of Illinois Press.

Bajada, C., Jackson, R., Haroon, H., Azadbakht, H., GJM, P., Lambon Ralph, M., LL, C., 2017. A graded tractographic parcellation of the temporal lobe. Neuroimage 155, 503–512.

Blaser, R., Fryzlewicz, P., 2016. Random rotation ensembles. The Journal of Machine Learning Research 17, 126–151.

Brodmann, K., 1909. Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues. Barth.

Buckner, R., Margulies, D., 2019. Macroscale cortical organization and a default-like apex transmodal network in the marmoset monkey. Nat Commun 10, 1976.

Buckner, R.L., Krienen, F.M., 2013. The evolution of distributed association networks in the human brain. Trends in cognitive sciences 17, 648–665.

Cerliani, L., Thomas, R.M., Jbabdi, S., Siero, J.C., Nanetti, L., Crippa, A., Gazzola, V., D'Arceuil, H., Keysers, C., 2012. Probabilistic tractography recovers a rostrocaudal trajectory of connectivity variability in the human insular cortex. Human brain mapping 33, 2005–2034.

Cliff, A.D., Ord, J.K., 1973. Spatial autocorrelation. London: Pion.

Coifman, R.R., Lafon, S., 2006. Diffusion maps. Applied and computational harmonic analysis 21, 5–30.

Desikan, R.S., Ségonne, F., Fischl, B., Quinn, B.T., Dickerson, B.C., Blacker, D., Buckner, R.L., Dale, A.M., Maguire, R.P., Hyman, B.T., et al., 2006. An automated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest. Neuroimage 31, 968–980.

Dray, S., 2011. A new perspective about moran's coefficient: Spatial autocorrelation as a linear regression problem. Geographical Analysis 43, 127–141.

von Economo, C.F., Koskinas, G.N., 1925. Die Cytoarchitektonik der Hirnrinde des erwachsenen Menschen. J. Springer.

Eickhoff, S.B., Yeo, B.T., Genon, S., 2018. Imaging-based parcellations of the human brain. Nature Reviews Neuroscience , 1.

Fischl, B., 2012. Freesurfer. Neuroimage 62, 774–81.

Flechsig, P.E., 1920. Anatomie des menschlichen Gehirns und Rückenmarks auf myelogenetischer Grundlage. volume 1. G. Thieme.

Fulcher, B.D., Murray, J.D., Zerbi, V., Wang, X.J., 2019. Multimodal gradients across mouse cortex. Proceedings of the National Academy of Sciences 116, 4689–4695.

Glasser, M.F., Coalson, T.S., Robinson, E.C., Hacker, C.D., Harwell, J., Yacoub, E., Ugurbil, K., Andersson, J., Beckmann, C.F., Jenkinson, M., et al., 2016. A multi-modal parcellation of human cerebral cortex. Nature 536, 171.

Goulas, A., Majka, P., Rosa, M., Higetag, C., 2019. A blueprint of mammalian cortical connectomes. PLoS Biology 17, e2005346.

Guell, X., Goncalves, M., Kaczmarzyk, J., Gabrieli, J., Schmahmann, J., Ghosh, S., 2019. Littlebrain: A gradient-based tool for the topographical interpretation of cerebellar neuroimaging findings. PLoS One 17, e2005346.

Haak, K.V., Marquand, A.F., Beckmann, C.F., 2018. Connectopic mapping with resting-state fmri. Neuroimage 170, 83–94.

Hong, S.J., Vos de Wael, R., Bethlehem, R.A., Lariviere, S., Paquola, C., Valk, S.L., Milham, M.P., Di Martino, A., Margulies, D.S., Smallwood, J., et al., 2019. Atypical functional connectome hierarchy in autism. Nature communications 10, 1022.

Huntenburg, J.M., Bazin, P.L., Goulas, A., Tardif, C.L., Villringer, A., Margulies, D.S., 2017. A systematic relationship between functional connectivity and intracortical myelin in the human cerebral cortex. Cerebral Cortex 27, 981–997.

Huntenburg, J.M., Bazin, P.L., Margulies, D.S., 2018. Large-scale gradients in human cortical organization. Trends in cognitive sciences 22, 21–31.

Kim, J., Singh, V., Lee, J., Lerch, J., Ad-Dab'bagh, Y., MacDonald, D., Lee, J., Kim, S., Evans, A., 2005. Automated 3-d extraction and evaluation of the inner and outer cortical surfaces using a laplacian map and partial volume effect classification. Neuroimage 27, 210–21.

Langs, G., Golland, P., Ghosh, S.S., 2015. Predicting activation across individuals with resting-state functional connectivity based multi-atlas label fusion, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 313–320.

Larivière, S., Vos de Wael, R., Hong, S.J., Paquola, C., Tavakol, S., Lowe, A.J., Schrader, D.V., Bernhardt, B.C., 2019. Multiscale structure–function gradients in the neonatal connectome. Cerebral Cortex .

Lombaert, H., Grady, L., Polimeni, J.R., Cheriet, F., 2013. FOCUSR: Feature oriented correspondence using spectral regularization - A method for precise surface matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 35, 2143–60.

Lowe, A.J., Paquola, C., de Wael, R.V., Girn, M., Lariviere, S., Tavakol, S., Caldairou, B., Royer, J., Schrader, D.V., Bernasconi, A., et al., 2019. Targeting age-related differences in brain and cognition with multimodal imaging and connectome topography profiling. bioRxiv , 601146.

Margulies, D.S., Ghosh, S.S., Goulas, A., Falkiewicz, M., Huntenburg, J.M., Langs, G., Bezgin, G., Eickhoff, S.B., Castellanos, F.X., Petrides, M., et al., 2016. Situating the default-mode network along a principal gradient of macroscale cortical organization. Proceedings of the National Academy of Sciences 113, 12574–12579.

Marquand, A.F., Haak, K.V., Beckmann, C.F., 2017. Functional corticostriatal connection topographies predict goal directed behaviour in humans. Nature Human Behavior 1, 0146.

Mcintosh, R., Misic, B., 2013. Multivariate statistical analyses for neuroimaging data. Annual Review of Psychology 64, 499–525.

Mesulam, M.M., 1998. From sensation to cognition. Brain: a journal of neurology 121, 1013–1052.

Murphy, C., Jefferies, E., Rueschemeyer, S.A., Sormaz, M., Wang, H.T., Margulies, D.S., , Smallwood, J., 2018. Distant from input: Evidence of regions within the default mode network supporting perceptually-decoupled and conceptually-guided cognition. NeuroImage 171, 393–401.

Murphy, C., Wang, H.T., Konu, D., Lowndes, R., Margulies, D.S., Jefferies, E., Smallwood, J., 2019. Modes of operation: A topographic neural gradient supporting stimulus dependent and independent cognition. NeuroImage 186, 487–496.

Nenning, K.H., Liu, H., Ghosh, S.S., Sabuncu, M.R., Schwartz, E., Langs, G., 2017. Diffeomorphic functional brain surface alignment: Functional demons. NeuroImage 156, 456–465.

Ng, A.Y., Jordan, M.I., Weiss, Y., 2001. On spectral clustering: Analysis and an algorithm, in: Advances in neural information processing systems, MIT Press. pp. 849–856.

Palomero-Gallagher, N., Zilles, K., 2017. Cortical layers: Cyto-, myelo-, receptor-and synaptic architecture in human cortical areas. Neuroimage .

Pandya, D., Petrides, M., Cipolloni, P.B., 2015. Cerebral Cortex - architecture, connections and the dual origin concept. Oxford University Press.

Paquola, C., Vos de Wael, R., Wagstyl, K., Bethlehem, R.A., Hong, S.J., Seidlitz, J., Bullmore, E.T., Evans, A.C., Misic, B., Margulies, D.S., et al., 2019. Microstructural and functional gradients are increasingly dissociated in transmodal cortices. PLoS biology 17, e3000284.

Przeździk, I., Faber, M., Fernandez, G., Beckmann, C.F., Haak, K.V., 2019. The functional organisation of the hippocampus along its long axis is gradual and predicts recollection. Cortex 119, 324–335.

Schaefer, A., Kong, R., Gordon, E.M., Laumann, T.O., Zuo, X.N., Holmes, A.J., Eickhoff, S.B., Yeo, B.T., 2017. Local-global parcellation of the human cerebral cortex from intrinsic functional connectivity mri. Cerebral Cortex 28, 3095–3114.

Shine, J., Breakspear, M., Bell, P., Ehgoetz, M.K., Shine, R., Koyejo, O., Sporns, O., Poldrack, R., 2019. Human cognition involves the dynamic integration of neural activity and neuromodulatory systems. Nat Neurosci 22, 289–96.

Sormaz, M., Murphy, C., Wang, H.t., Hymers, M., Karapanagiotidis, T., Poerio, G., Margulies, D.S., Jefferies, E., Smallwood, J., 2018. Default mode network can support the level of detail in experience during active task states. Proceedings of the National Academy of Sciences 115, 9318–9323.

Tian, Y., Zalesky, A., 2018. Characterizing the functional connectivity diversity of the insula cortex: Subregions, diversity curves and behavior. Neuroimage 183, 716–733.

Tian, Y., Zalesky, A., Bousman, C., Everall, I., Pantelis, C., 2019. Insula functional connectivity in schizophrenia: Subregions, gradients, and symptoms. Biological Psychiatry: Cognitive Neuroscience and Neuroimaging 4, 399–408.

Van Essen, D.C., Smith, S.M., Barch, D.M., Behrens, T.E., Yacoub, E., Ugurbil, K., Consortium, W.M.H., et al., 2013. The wu-minn human connectome project: an overview. Neuroimage 80, 62–79.

Vogel, J.W., La Joie, R., Grothe, M.J., Diaz-Papkovich, A., Doyle, A., Vachon-Presseau, E., Lepage, C., de Wael, R.V., Iturria-Medina, Y., Bernhardt, B., et al., 2019. A molecular gradient along the longitudinal axis of the human hippocampus informs large-scale behavioral systems. bioRxiv , 587071.

Von Bonin, G., Bailey, P., 1947. The neocortex of macaca mulatta. urbana. IL: University of Illinois .

Vos de Wael, R., Larivière, S., Caldairou, B., Hong, S.J., Margulies, D.S., Jefferies, E., Bernasconi, A., Smallwood, J., Bernasconi, N., Bernhardt, B.C., 2018. Anatomical and microstructural determinants of hippocampal subfield functional connectome embedding. Proceedings of the National Academy of Sciences 115, 10154–10159.

Wagner, H.H., Dray, S., 2015. Generating spatially constrained null models for irregularly spaced data using m oran spectral randomization methods. Methods in Ecology and Evolution 6, 1169–1178.

Wagstyl, K., Ronan, L., Goodyer, I.M., Fletcher, P.C., 2015. Cortical thickness gradients in structural hierarchies. Neuroimage 111, 241–250.

Wang, P., Kong, R., Liegeos, R., Orban, C., Deco, G., Van den Heuvel, M.P., Yeo, B.T., 2019. Inversion of a large-scale circuit model reveals a cortical hierarchy in the dynamic resting human brain. Science Avances 5, eaat7854.

Worsley, K.J., Taylor, J.E., Carbonell, F., Chung, M., Duerden, E., Bernhardt, B., Lyttelton, O., Boucher, M., Evans, A., 2009. Surfstat: A matlab toolbox for the statistical analysis of univariate and multivariate surface and volumetric data using linear mixed effects models and random field theory. Neuroimage , S102.

Xu, T., Nenning, K., Schwartz, E., Hong, S., Vogelstein, J., Fair, D., Schroeder, C., Margulies, D., Smallwood, J., MP, M., 2019. Cross-species functional alignment reveals evolutionary hierarchy within the connectome. BioRxiv https://doi.org/10.1101/692616.

Yeo, B.T., Krienen, F.M., Sepulcre, J., Sabuncu, M.R., Lashkari, D., Hollinshead, M., Roffman, J.L., Smoller, J.W., Zöllei, L., Polimeni, J.R., et al., 2011. The organization of the human cerebral cortex estimated by intrinsic functional connectivity. Journal of neurophysiology 106, 1125–1165.

## 680 Appendix A. Python code samples

In this appendix we show the Python code analogous to the Matlab code samples provided in the main body of the paper.

Sample Code 5: Python version for Matlab code sample 1. A minimal example for plotting the first gradient of an input data matrix on the cortical surface.

```
from brainspace.gradient import GradientMaps
from brainspace.plotting import plot_hemispheres
from brainspace.mesh import mesh_io as mio

# Create a GradientMaps object
G = GradientMaps(kernel='cosine', approach='dm')

# Apply GradientMaps to the data
G.fit(data_matrix)

# Load surfaces
left_surface = mio.load_surface('left_surface_file.obj')
right_surface = mio.load_surface('right_surface_file.obj')

# Plot first gradient on the cortical surface
plot_hemispheres(left_surface, right_surface, G.gradients_[:,0])
```

Sample Code 6: Python version for Matlab code sample 2. A minimal example for creating and plotting gradients from different modalities, with different alignments.

```
from brainspace.gradient import GradientMaps
from brainspace.plotting import plot_hemispheres
from brainspace.mesh import mesh_io as mio

# Create two GradientMaps objects with different alignments
Gp = GradientMaps(kernel='cosine', approach='dm',
                  alignment='procrustes')
Gj = GradientMaps(kernel='cosine', approach='dm',
                  alignment='joint')

# Apply GradientMaps to the data
Gp.fit([mpc, fc])
Gj.fit([mpc, fc])

# Load surfaces
left_surface = mio.load_surface('left_surface_file.obj')
right_surface = mio.load_surface('right_surface_file.obj')

# Plot first MPC gradient of Procrustes alignment
```

```
plot_hemispheres(left_surface, right_surface, Gp.aligned_[0][:,0])

# Plot first FC gradient of joint embedding
plot_hemispheres(left_surface, right_surface, Gj.aligned_[1][:,0])
```

Sample Code 7: Python version for Matlab code sample 3. A minimal example for aligning
685 the gradients of two individuals to a template gradient.

```
from brainspace.gradient import GradientMaps
from brainspace.plotting import plot_hemispheres
from brainspace.mesh import mesh_io as mio

# Create a GradientMaps object for the template
Gt = GradientMaps(kernel='cosine', approach='dm')

# Apply to template data
Gt.fit(template_data)

# Create a GradientMaps object for the individuals
Gs = GradientMaps(approach='dm', kernel='cosine',
                  alignment='procrustes')

# Compute gradients for all subjects and align to template
Gs.fit([subject1_data, subject2_data], reference=Gt.gradients_)

# Load surfaces
left_surface = mio.load_surface('left_surface_file.obj')
right_surface = mio.load_surface('right_surface_file.obj')

# Plot the first aligned gradient for subject 2
plot_hemispheres(left_surface, right_surface, Gs.aligned_[1][:, 0])
```

Sample Code 8: Python version for Matlab code sample 4. A minimal Python example for building null models based on spin tests.

```
from brainspace.null_models import SpinPermutations
from brainspace.mesh import mesh_io as mio

# Load spheres
left_sphere = mio.load_surface('left_sphere_file.obj')
right_sphere = mio.load_surface('right_sphere_file.obj')

# Generate 1000 spin permutations
n_perm = 1000
sp = SpinPermutations(n_rep=n_perm)
sp.fit(left_sphere, points_rh=right_sphere)
features_spin = sp.randomize(left_features, x_rh=right_features)
```