Supplementary Information to Structural MRI based parcellation of the human brain using spatial hierarchical clustering

Tobias Tietz[1*], Simon B. Eickhoff[2,3,4] and Holger Schwender[1]

[1*]Mathematical Institute, Heinrich-Heine University, Düsseldorf, Germany.
[2]Institute of Neuroscience and Medicine (INM-1, INM-7), Research Centre Juelich, Germany.
[3]Institute of Systems Neuroscience, Medical Faculty, Heinrich-Heine University, Düsseldorf, Germany.
[4]Institute of Clinical Neuroscience and Medical Psychology, Heinrich-Heine University, Düsseldorf, Germany.

*Corresponding author(s). E-mail(s): tobias.tietz@uni-duesseldorf.de;
Contributing authors: simon.eickhoff@uni-duesseldorf.de; holger.schwender@uni-duesseldorf.de;

Keywords: structural MRI, spatial hierarchical clustering, ensemble clustering, Voxel-based morphometry, brain parcellation, clustering stability

A Simulation study

In this section we provide a more detailed description of the simulation study than that given in the main manuscript.

A.1 Setup of simulation study

Two different settings are considered in the simulation study. In the first setting, 3D images with equally sized clusters and, in the second setting, 3D images with clusters of three different sizes are simulated. All simulated 3D

images in all simulations are on a cubic grid of $18 \times 18 \times 18$ voxels. Hence, each simulated 3D image consists of $V = 18 \cdot 18 \cdot 18 = 5832$ voxels.

In a first step, one true parcellation for the first setting and one true parcellation for the second setting is determined. In both settings the true parcellation consists of 27 larger clusters which again split up into two equally sized smaller clusters. Hereby, all these clusters are spatially contiguous. However, in the first setting, the 27 larger clusters are all of size $6 \times 6 \times 6$ and, following, all 54 smaller clusters are of size $6 \times 6 \times 3$. In the second setting, 9 larger clusters are of size $3 \times 6 \times 6$, 9 larger clusters are of size $6 \times 6 \times 6$ and 9 larger clusters are of size $9 \times 6 \times 6$ which entails that 18 smaller clusters are of size $3 \times 6 \times 3$, 18 smaller clusters are of size $6 \times 6 \times 3$ and 18 smaller clusters are of size $9 \times 6 \times 3$. The true parcellations from the two settings are displayed in Figure A.1. In the following, let

$$\boldsymbol{T}_{54}^{(t)} = \{T_{1,1}^{(t)}, T_{1,2}^{(t)}, T_{2,1}^{(t)}, T_{2,2}^{(t)}, \ldots, T_{27,1}^{(t)}, T_{27,2}^{(t)}\}$$

be the true parcellation with 54 smaller clusters and, thus, let

$$\boldsymbol{T}_{27}^{(t)} = \{T_{1,1}^{(t)} \cup T_{1,2}^{(t)}, T_{2,1}^{(t)} \cup T_{2,2}^{(t)}, \ldots, T_{27,1}^{(t)} \cup T_{27,2}^{(t)}\} := \{T_1^{(t)}, T_2^{(t)}, \ldots, T_{27}^{(t)}\}$$

be the true parcellation with 27 larger clusters, where $t = 1, 2$ indicates the setting.

In a second step, the voxel intensities are simulated from a multivariate normal distribution. Since 3D images should be simulated, where each image consists of $V = 5832$ voxels, let $\boldsymbol{Y}$ be a V-dimensional random vector which follows a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, i.e. $\boldsymbol{Y} \sim N_V(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Remember that the general assumption is that voxels belonging to the same true brain region are (highly) correlated among each other. Therefore, the clusters are simulated solely via the covariance matrix $\boldsymbol{\Sigma}$, whereas $\boldsymbol{\mu} = \boldsymbol{0}_V$, i.e. the mean vector does not include any cluster information. In order to allow a better interpretability of the clustering results, the covariance matrix is chosen to be a correlation matrix with only two different correlation values $1 > \sigma_S \geq \sigma_L \geq 0$. Hereby, σ_S is the correlation between voxels from the same smaller cluster and σ_L is the correlation between voxels that are in the same larger cluster but not in the same smaller cluster. Moreover, the correlation between voxels from different larger clusters is chosen to be zero. Hence, for setting $t, t = 1, 2$,

$$\sigma_{j\ell}^{(t)}(\sigma_S, \sigma_L) = \begin{cases} 1, & \text{if } j = \ell, \\ \sigma_S, & \text{if } j \neq \ell \text{ and } \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_j) = \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_\ell), \\ \sigma_L, & \text{if } j \neq \ell,\ \boldsymbol{T}_{27}^{(t)}(\boldsymbol{x}_j) = \boldsymbol{T}_{27}^{(t)}(\boldsymbol{x}_\ell) \text{ and } \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_j) \neq \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_\ell), \\ 0, & \text{otherwise.} \end{cases}$$

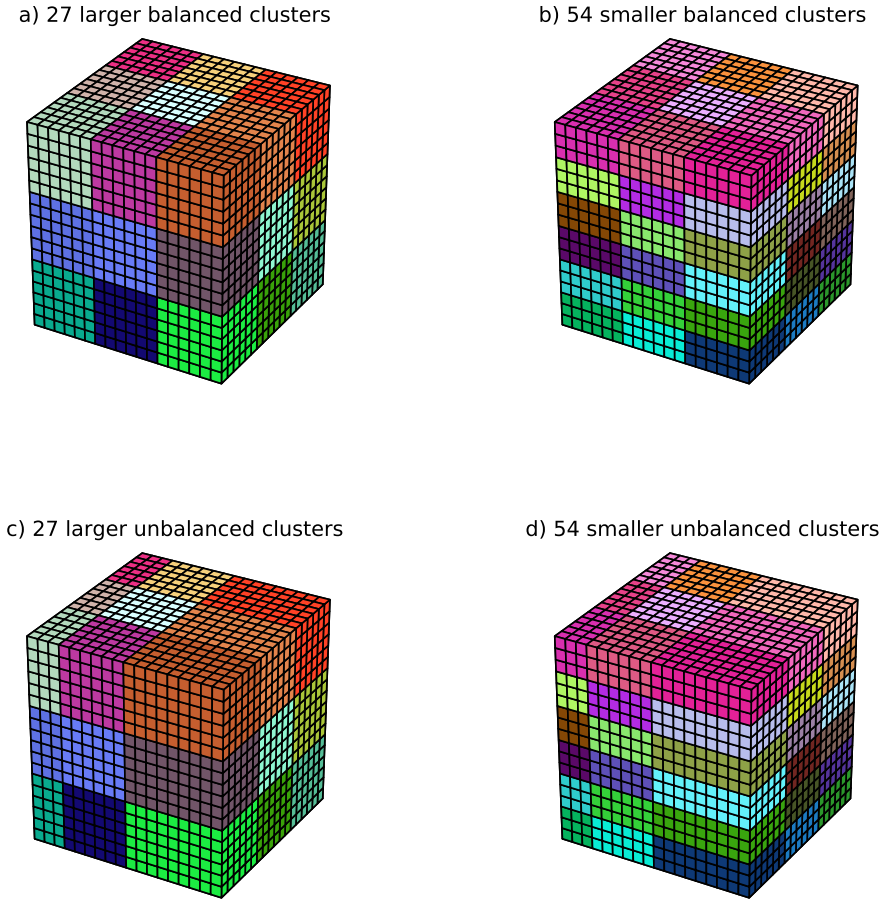

Fig. A.1 The true parcellation of the 3D cubic images: a) and b) display the true parcellation of the first setting with 54 smaller clusters of equal size embedded pairwisely in 27 larger clusters of equal size; c) and d) display the true parcellation of the second setting with 54 smaller clusters of three different sizes embedded pairwisely in 27 larger clusters of three different sizes. Note that this figure is (almost) identical to Figure 1 of the main manuscript

describes the correlation between voxels $\boldsymbol{x}_j$ and $\boldsymbol{x}_\ell$, where

$$\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L) = \left(\sigma_{j\ell}^{(t)}(\sigma_S, \sigma_L)\right)_{j,\ell=1,\ldots,V}.$$

Note that $\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L)$ is positive definite and, therefore, a valid correlation matrix.

To show that $\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L)$ is indeed positive definite, the squared matrix

$$\boldsymbol{A}_k^{(t)} = \begin{pmatrix} \boldsymbol{A}_{k,1}^{(t)} & \boldsymbol{A}_{k,2}^{(t)} \\ \boldsymbol{A}_{k,2}^{(t)} & \boldsymbol{A}_{k,1}^{(t)} \end{pmatrix}$$

is considered, $k = 1, \ldots, 27$, where

$$\boldsymbol{A}_{k,1}^{(t)} = a_{k,1}^{(t)} \cdot \mathbf{1}_{\frac{|\,T_k^{(t)}\,|}{2} \times \frac{|\,T_k^{(t)}\,|}{2}},$$
$$\boldsymbol{A}_{k,2}^{(t)} = a_{k,2}^{(t)} \cdot \mathbf{1}_{\frac{|\,T_k^{(t)}\,|}{2} \times \frac{|\,T_k^{(t)}\,|}{2}}$$

with $a_{k,1}^{(t)}, a_{k,2}^{(t)} \in \mathbb{R}$ and $\mathbf{1}_{\frac{|\,T_k^{(t)}\,|}{2} \times \frac{|\,T_k^{(t)}\,|}{2}}$ being the $\left(\frac{|\,T_k^{(t)}\,|}{2} \times \frac{|\,T_k^{(t)}\,|}{2}\right)$-matrix with all entries equal to 1. Hence, $\boldsymbol{A}_k^{(t)} \in \left\{a_{k,1}^{(t)}, a_{k,2}^{(t)}\right\}^{|T_k^{(t)}| \times |T_k^{(t)}|}$ and

$$\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} = \begin{pmatrix} \boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,1}^{(t)} + \boldsymbol{A}_{k,2}^{(t)}\boldsymbol{A}_{k,2}^{(t)} & 2\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,2}^{(t)} \\ 2\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,2}^{(t)} & \boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,1}^{(t)} + \boldsymbol{A}_{k,2}^{(t)}\boldsymbol{A}_{k,2}^{(t)} \end{pmatrix}$$

is a positive semidefinite matrix. The idea is to choose $a_{k,1}^{(t)}$ and $a_{k,2}^{(t)}$ such that

$$\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,1}^{(t)} + \boldsymbol{A}_{k,2}^{(t)}\boldsymbol{A}_{k,2}^{(t)} = \sigma_S \cdot \mathbf{1}_{\frac{|\,T_k^{(t)}\,|}{2} \times \frac{|\,T_k^{(t)}\,|}{2}} \quad \text{and}$$
$$2\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,2}^{(t)} = \sigma_L \cdot \mathbf{1}_{\frac{|\,T_k^{(t)}\,|}{2} \times \frac{|\,T_k^{(t)}\,|}{2}}$$

which is equivalent to

$$(I) \quad \left(\left(a_{k,1}^{(t)}\right)^2 + \left(a_{k,2}^{(t)}\right)^2\right) \frac{|\,T_k^{(t)}\,|}{2} = \sigma_S,$$
$$(II) \quad 2a_{k,1}^{(t)}a_{k,2}^{(t)} \frac{|\,T_k^{(t)}\,|}{2} = \sigma_L.$$

Solving this system of equations results in four possible solutions, i.e.

$$a_{k,2}^{(t)} = \pm\sqrt{\frac{\sigma_S}{|\,T_k^{(t)}\,|} \pm \sqrt{\frac{\sigma_S^2}{|\,T_k^{(t)}\,|^2} - \frac{\sigma_L^2}{|\,T_k^{(t)}\,|^2}}}$$

and

$$a_{k,1}^{(t)} = \frac{\sigma_L}{\mid T_k^{(t)} \mid a_{k,2}^{(t)}}.$$

Since $\sigma_S \geq \sigma_L \geq 0$, all four solutions are real. Hence, for any of these four solutions,

$$\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} = \begin{pmatrix} \sigma_S \cdot \mathbf{1} & \sigma_L \cdot \mathbf{1} \\ \frac{\mid T_k^{(t)} \mid}{2} \times \frac{\mid T_k^{(t)} \mid}{2} & \frac{\mid T_k^{(t)} \mid}{2} \times \frac{\mid T_k^{(t)} \mid}{2} \\ \sigma_L \cdot \mathbf{1} & \sigma_S \cdot \mathbf{1} \\ \frac{\mid T_k^{(t)} \mid}{2} \times \frac{\mid T_k^{(t)} \mid}{2} & \frac{\mid T_k^{(t)} \mid}{2} \times \frac{\mid T_k^{(t)} \mid}{2} \end{pmatrix}$$

is a positive semidefinite matrix. i.e. all eigenvalues of $\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)}$ are nonnegative. Then,

$$\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L) = \left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} + (1 - \sigma_S)\boldsymbol{I}$$

is positive definite. This is because for any eigenvector $\boldsymbol{e}$ of $\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)}$ with corresponding eigenvalue $\lambda \geq 0$

$$\begin{aligned} \boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)\boldsymbol{e} &= \left(\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} + (1 - \sigma_S)\boldsymbol{I}\right)\boldsymbol{e} = \left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)}\boldsymbol{e} + (1 - \sigma_S)\boldsymbol{e} \\ &= \lambda\boldsymbol{e} + (1 - \sigma_S)\boldsymbol{e} = (\lambda + 1 - \sigma_S)\boldsymbol{e} \end{aligned}$$

and, hence, $\boldsymbol{e}$ is also an eigenvector of $\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)$ with eigenvalue $\lambda + 1 - \sigma_S > 0$. Since all eigenvalues of $\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)$ are positive, $\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)$ is positive definite. Finally, without loss of generality, assume the voxels are ordered such that firstly the voxels from $T_{1,1}^{(t)}$ then the voxels from $T_{1,2}^{(t)}$ and so on until finally the voxels from $T_{27,2}^{(t)}$ occur in the ordering. Then, it follows that

$$\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L) = \text{diag}\left\{\boldsymbol{\Sigma}_1^{(t)}(\sigma_S, \sigma_L), \ldots, \boldsymbol{\Sigma}_{27}^{(t)}(\sigma_S, \sigma_L)\right\}$$

and $\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L)$ is as block diagonal matrix with positive definite diagonal blocks also positive definite.

For each setting it should be analyzed whether the clustering accuracy increases for stronger pronounced clusters, i.e. for clusters of higher correlated voxels. Hence, for each setting three different simulation scenarios are considered. In the first, second or third scenario of each setting strongly, moderately or weakly pronounced clusters are simulated by sampling 3D images from a multivariate normal distribution with covariance matrix $\boldsymbol{\Sigma}^{(t)}(0.2, 0.1)$, $\boldsymbol{\Sigma}^{(t)}(0.1, 0.05)$ or $\boldsymbol{\Sigma}^{(t)}(0.05, 0.025)$, respectively. In each of the six simulation scenarios $H = 25$ data sets are sampled, where each data set consists of $N = 100$ 3D images. Let in the following SimBS, SimBM and SimBW denote

the first, second and third simulation scenario from the first setting, and let SimUS, SimUM and SimUW denote the first, second and third simulation scenario from the second setting, respectively. Also let $\boldsymbol{X}_h^{(g)} \in \mathbb{R}^{N \times V}, h = 1, \ldots, 25, g = 1, \ldots, 6$, be the h-th data set from the g-th simulation scenario. Since structural MR images have only positive intensity values, each data matrix is normalized to be in $[0, 1]^{N \times V}$, i.e.

$$\boldsymbol{X}_h^{(g)} = \frac{\boldsymbol{X}_h^{(g)} - \min\left\{\boldsymbol{X}_h^{(g)}\right\}}{\max\left\{\boldsymbol{X}_h^{(g)}\right\} - \min\left\{\boldsymbol{X}_h^{(g)}\right\}}$$

is calculated, where $\min\left\{\boldsymbol{X}_h^{(g)}\right\}$ is the minimum entry of $\boldsymbol{X}_h^{(g)}$.

A typical preprocessing step for structural MRI data is smoothing (**?**). Therefore, smoothing of the (normalized) spatial images, i.e. of the rows of $\boldsymbol{X}_h^{(g)}$, is also performed by using a multidimensional Gaussian filter (Jones et al, 2005) with full width at half maximum (FWHM) being equal to two, i.e. FWHM $= 2$. Hereby, the 3D images are extended at the borders by reflecting about the edge of the border voxels. Finally, another optional preprocessing step is to standardize the columns of $\boldsymbol{X}_h^{(g)}$ to have zero mean and unit variance, i.e. to consider standardized data sets.

A.2 Analysis of simulation study

Five different spatial clustering algorithms are considered for comparison, namely, SPARTACUS, $\text{SHAC}_{\text{Ward}}$, $\text{SHAC}_{\text{AL, corr}}$, $\text{SHAC}_{\text{AL, Eucl}}$ and SSPEC. Moreover, in order to analyze whether standardization influences clustering quality, all these five algorithms are applied to standardized data sets, whereas only the non-correlation based algorithms $\text{SHAC}_{\text{Ward}}$, $\text{SHAC}_{\text{AL, Eucl}}$ and SSPEC are also applied to non-standardized data sets. Hence, in total eight different clustering methods are compared in the performance analysis. Clustering methods that are applied to standardized data sets are indicated by $method^S$, e.g., SPARTACUS^S.

A.2.1 Performance comparison of spatial clustering methods

All eight clustering methods are employed to generate two estimated parcellations, one with $K = 27$ clusters and one with $K = 54$ clusters, from each simulated data set $\boldsymbol{X}_h^{(g)}, h = 1, \ldots, 25, g = 1, \ldots, 6$. The quality of the estimated parcellations is evaluated using two different approaches. In the first approach, each estimated parcellation is compared with its corresponding true parcellation ($\boldsymbol{T}_{27}^{(t)}, \boldsymbol{T}_{54}^{(t)}, t = 1, 2$) using the adjusted Rand index (ARI). The mean is taken over the 25 ARI scores corresponding to the same simulation scenario, the same clustering method and the same number of clusters. In the second approach, each estimated parcellation is evaluated on the same simulated data set it has been generated on usind the absolute correlation based simplified silhouette coefficient (SSC). Again, the mean is taken over the corresponding 25 SSC scores. These mean scores based on both approaches are displayed in Table A.1.

Three results can be concluded from these tables. Firstly, the SHAC algorithms outperform the SSPEC algorithms, especially for $K = 54$. E.g., for Sim 1 the mean ARI (SSC) values of all SHAC algorithms are in $[0.976, 1.000]$ ($[0.669, 0.685]$) for $K = 27$ and in $[0.9267, 0.9976]$ ($[0.678, 0.684]$) for $K = 54$, i.e. they are clearly larger than the mean ARI (SSC) values of the SSPEC algorithms, which are in $[0.633, 0.935]$ ($[0.475, 0.639]$) for $K = 27$ and in $[0.277, 0.539]$ ($[0.028, 0.340]$) for $K = 54$. Secondly, except for $\text{SHAC}_{\text{Ward}}$, the standardized algorithms ouperform their corresponding non-standardized algorithms, especially if the simulated effect is small. E.g., for Sim 3 the mean ARI values are 0.9286 ($K = 27$) and 0.7105 ($K = 54$) for $\text{SHAC}^S_{\text{AL, Eucl}}$ as opposed to 0.0038 ($K = 27$) and 0.1429 ($K = 54$) for $\text{SHAC}_{\text{AL, Eucl}}$. Moreover, $\text{SSPEC}_{\text{RBF}}$ shows the worst performance among the SSPEC algorithms in all scenarios. Thirdly, for $K = 54$ SPARTACUS^S, $\text{SHAC}^S_{\text{Ward}}$ and $\text{SHAC}_{\text{Ward}}$ perform better than $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$ in all scenarios , but especially if the simulated effect is small. However, in the second setting with unbalanced parcellations, SPARTACUS^S, $\text{SHAC}^S_{\text{Ward}}$ and $\text{SHAC}_{\text{Ward}}$ are slightly outperformed by $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$ for $K = 27$.

Moreover, in order to further investigate the poor performance of $\text{SSPEC}_{\text{RBF}}$ and $\text{SHAC}_{\text{AL, Eucl}}$, the partitions with $K = 27$ and $K = 54$ clusters

Table A.1 For each simulation scenario and for each of eight clustering methods the mean over 25 ARI and SSC scores is displayed, where each ARI value compares a predicted parcellation with $K = 27$ or $K = 54$ clusters with the respective true parcellation and each SSC score evaluates the quality of a predicted parcellation on the training data. This table supplements Table 2 from the main manuscript

	SimBS		SimBM		SimBW	
	ARI	SSC	ARI	SSC	ARI	SSC
	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54
SPARTACUS^S	**1.000**, **0.998**	**0.685**, **0.684**	0.991, **0.970**	0.583, 0.551	0.930, **0.861**	0.450, 0.407
SHAC$_{\text{Ward}}$	0.999, **0.998**	0.683, **0.684**	0.984, 0.968	0.579, 0.550	0.907, 0.846	0.441, 0.401
SHAC$^S_{\text{Ward}}$	**1.000**, **0.998**	**0.685**, **0.684**	0.989, 0.968	0.582, 0.550	0.923, 0.852	0.448, 0.404
SHAC$^S_{\text{AL, corr}}$	**1.000**, 0.955	**0.685**, **0.684**	0.990, 0.850	0.583, 0.561	**0.936**, 0.721	**0.452**, **0.430**
SHAC$_{\text{AL, Eucl}}$	0.976, 0.927	0.669, 0.678	0.582, 0.679	0.419, 0.494	0.004, 0.143	-0.139, 0.142
SHAC$^S_{\text{AL, Eucl}}$	**1.000**, 0.960	**0.685**, **0.684**	**0.992**, 0.850	**0.584**, **0.562**	0.929, 0.710	0.450, **0.430**
SSPEC	0.633, 0.277	0.475, 0.028	0.633, 0.278	0.417, 0.067	0.630, 0.278	0.342, 0.111
SSPEC^S	0.878, 0.539	0.639, 0.340	0.849, 0.442	0.541, 0.242	0.831, 0.345	0.430, 0.183

	SimUS		SimUM		SimUW	
	ARI	SSC	ARI	SSC	ARI	SSC
	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54
SPARTACUS^S	0.956, **0.998**	0.677, 0.684	0.938, **0.972**	0.570, 0.550	0.875, **0.865**	0.435, 0.405
SHAC$_{\text{Ward}}$	0.939, **0.998**	0.668, 0.684	0.917, 0.967	0.559, 0.548	0.854, 0.853	0.424, 0.400
SHAC$^S_{\text{Ward}}$	0.955, **0.998**	0.676, 0.684	0.938, 0.969	0.569, 0.549	0.869, 0.856	0.431, 0.402
SHAC$^S_{\text{AL, corr}}$	**1.000**, 0.941	**0.683**, **0.686**	**0.991**, 0.845	**0.581**, **0.562**	**0.941**, 0.733	**0.448**, **0.428**
SHAC$_{\text{AL, Eucl}}$	0.990, 0.935	0.674, 0.675	0.604, 0.680	0.403, 0.493	0.005, 0.130	-0.136, 0.113
SHAC$^S_{\text{AL, Eucl}}$	**1.000**, 0.944	**0.683**, **0.686**	0.988, 0.838	0.580, 0.561	0.936, 0.709	0.447, 0.424
SSPEC	0.407, 0.299	0.153, 0.052	0.401, 0.296	0.150, 0.095	0.403, 0.295	0.144, 0.132
SSPEC^S	0.692, 0.561	0.432, 0.331	0.612, 0.483	0.318, 0.272	0.547, 0.420	0.235, 0.221

generated by $\text{SSPEC}_{\text{RBF}}$ or $\text{SHAC}_{\text{AL, Eucl}}$ in application to the first simulated data set of SimBM or SimUW, respectively, are displayed in Figure A.2. This figure reveals that $\text{SSPEC}_{\text{RBF}}$ generates cubical shaped clusters of equal size, regardless of the underlying clustering structure. $\text{SHAC}_{\text{AL, Eucl}}$ tends to form a few large clusters and assigns outlier voxels or small groups of voxels to singleton or small clusters, respectively. These clustering behaviours explain the poor performance of these methods.

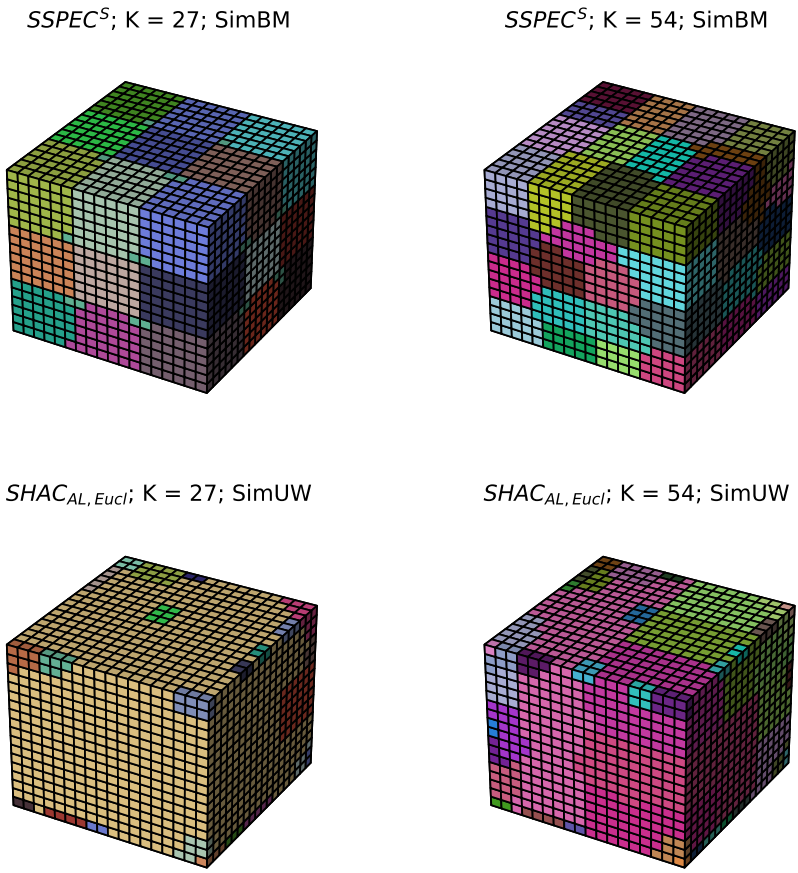

Fig. A.2 Visualization of the estimated partitions for $K = 27$ and $K = 54$ generated by SSPEC^S or $\text{SHAC}_{\text{AL,Eucl}}$ applied to the first simulated data set from SimBM or SimUW, respectively.

A.2.2 Performance of spatial ensemble clustering

Spatial ensemble clustering (SEC) methods are applied to all simulation scenarios in order to investigate whether these methods generate more accurate and more robust parcellations. Since in the first step the standardized clustering methods outperform the non-standardized clustering methods, the SEC methods are only based on the standardized clustering methods. Homogeneous cluster ensembles are generated by drawing $B = 50$ subsamples and considering the five standardized clustering methods as base clustering methods. In the consensus function step, the single linkage distance and the average linkage distance are considered as agglomeration measures in a SHAC algorithm, in order to obtain the final ensemble parcellations with $K = 27$ and $K = 54$ clusters from the cluster ensembles. Hence, in total, $5 \cdot 2 = 10$ SEC methods are considered, where, e.g., $\text{SEC}_{\text{AL}}\left(\text{SPARTACUS}^S\right)$ refers to the SEC method using SPARTACUS^S as only base clustering method and average linkage as agglomeration measure.

Analogously to the first step, on the one hand, the ARI is employed to compare the ensemble parcellations with $K = 27$ and $K = 54$ clusters with the respective true parcellations ($\boldsymbol{T}_{27}^{(t)}$ and $\boldsymbol{T}_{54}^{(t)}, t = 1, 2$). On the other hand, the SSC is used to evaluate the performance of the parcellations on the respective data sets the parcellations are generated on. The results of this analysis are presented in Table A.2.

From this table it can be observed that SEC_{SL} fails as SEC method, if the performance of the respective base clustering method is not already close to perfect. E.g., in scenario SimBW the mean ARI (SSC) scores for $K = 27$ and $K = 54$ based on $\text{SEC}_{\text{SL}}\left(\text{SPARTACUS}^S\right)$ are with 0.765 and 0.674 (0.361 and 0.376) clearly smaller compared to 0.930 and 0.861 (0.450 and 0.407) based on SPARTACUS^S, respectively. Also, SEC_{SL} decreases the performance of the SSPEC^S method in all scenarios. E.g., in scenario SimUM the mean ARI (SSC) scores for $K = 27$ and $K = 54$ based on $\text{SEC}_{\text{SL}}\left(\text{SSPEC}^S\right)$ are with 0.101 and 0.119 (-0.004 and -0.033) clearly smaller compared to 0.612 and 0.483 (0.318 and 0.272) based on SSPEC^S, respectively.

In contrast, SEC_{AL} improves the accuracy and robustness of the corresponding base clustering method, if the base clustering method does not already achieve perfect clustering. When considering the simulation scenarios SimBS, SimBM, SimUS and SimUM, i.e. the simulation scenarios with a large or moderate simulated effect, the mean ARI or SSC values of $\text{SEC}_{\text{AL}}\left(\text{SHAC}^S\right)$, i.e. of SEC_{AL} in combination with any SHAC^S method, are almost identical to the mean ARI or SSC values of SHAC^S. However, for the simulation scenarios SimBW and SimUW, i.e. when the simulated effect is small and, thus, the SHAC^S methods do not achieve perfect clustering, $\text{SEC}_{\text{AL}}\left(\text{SHAC}^S\right)$ improves clustering quality in comparison with SHAC^S. E.g., in SimUW the mean ARI (SSC) scores for $K = 27$ and $K = 54$ based on $\text{SEC}_{\text{AL}}\left(\text{SPARTACUS}^S\right)$ are 0.948 and 0.968 (0.450 and 0.440), respectively. In contrast, the mean ARI

Table A.2 For each simulation scenario and for each of ten SEC methods the mean over 25 ARI and SSC scores is displayed, where each ARI value compares a predicted parcellation with $K = 27$ or $K = 54$ clusters with the respective true parcellation and each SSC score evaluates the quality of a predicted parcellation on the training data.

	SimBS		SimBM		SimBW	
	ARI	SSC	ARI	SSC	ARI	SSC
	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54
$\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$	**1.000**, **1.000**	**0.685**, **0.685**	**1.000**, **0.998**	**0.586**, 0.561	**0.993**, **0.972**	**0.467**, **0.443**
$\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{Ward}})$	**1.000**, **1.000**	**0.685**, **0.685**	**1.000**, **0.998**	**0.586**, 0.561	**0.993**, 0.969	**0.467**, **0.443**
$\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{AL, corr}})$	**1.000**, 0.963	**0.685**, **0.685**	**1.000**, 0.852	**0.586**, **0.565**	0.980, 0.721	0.463, 0.433
$\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{AL, Eucl}})$	**1.000**, 0.963	**0.685**, **0.685**	0.998, 0.846	**0.586**, 0.564	0.972, 0.715	0.460, 0.431
$\text{SEC}_{\text{AL}}(\text{SSPEC}^S)$	0.815, 0.576	0.604, 0.369	0.737, 0.454	0.471, 0.250	0.832, 0.337	0.425, 0.177
$\text{SEC}_{\text{SL}}(\text{SPARTACUS}^S)$	**1.000**, **1.000**	**0.685**, **0.685**	0.994, 0.958	0.582, 0.558	0.765, 0.674	0.361, 0.376
$\text{SEC}_{\text{SL}}(\text{SHAC}^S_{\text{Ward}})$	**1.000**, **1.000**	**0.685**, **0.685**	0.992, 0.954	0.581, 0.556	0.739, 0.669	0.353, 0.368
$\text{SEC}_{\text{SL}}(\text{SHAC}^S_{\text{AL, corr}})$	**1.000**, 0.966	**0.685**, **0.685**	0.996, 0.831	0.583, 0.563	0.761, 0.638	0.367, 0.393
$\text{SEC}_{\text{SL}}(\text{SHAC}^S_{\text{AL, Eucl}})$	**1.000**, 0.969	**0.685**, **0.685**	0.995, 0.831	0.584, 0.562	0.760, 0.633	0.364, 0.391
$\text{SEC}_{\text{SL}}(\text{SSPEC}^S)$	0.274, 0.054	0.362, -0.147	0.245, 0.008	0.296, -0.258	0.191, 0.007	0.165, -0.205
	SimUS		SimUM		SimUW	
	ARI	SSC	ARI	SSC	ARI	SSC
	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54	K = 27, 54
$\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$	0.958, **1.000**	0.677, 0.685	0.953, **0.998**	0.574, 0.561	0.948, **0.968**	0.450, **0.440**
$\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{Ward}})$	0.961, **1.000**	0.677, 0.685	0.956, **0.998**	0.575, 0.561	0.956, 0.962	0.450, 0.439
$\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{AL, corr}})$	**1.000**, 0.959	**0.683**, **0.686**	**1.000**, 0.837	**0.584**, **0.565**	**0.988**, 0.721	**0.461**, 0.432
$\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{AL, Eucl}})$	**1.000**, 0.958	**0.683**, **0.686**	0.999, 0.831	0.583, 0.564	0.983, 0.714	0.459, 0.432
$\text{SEC}_{\text{AL}}(\text{SSPEC}^S)$	0.709, 0.608	0.437, 0.354	0.625, 0.536	0.331, 0.289	0.546, 0.415	0.233, 0.220
$\text{SEC}_{\text{SL}}(\text{SPARTACUS}^S)$	0.959, **1.000**	0.677, 0.685	0.945, 0.949	0.569, 0.559	0.761, 0.674	0.350, 0.368
$\text{SEC}_{\text{SL}}(\text{SHAC}^S_{\text{Ward}})$	0.962, **1.000**	**0.683**, 0.685	0.950, 0.950	0.569, 0.559	0.767, 0.666	0.351, 0.360
$\text{SEC}_{\text{SL}}(\text{SHAC}^S_{\text{AL, corr}})$	**1.000**, 0.959	**0.683**, **0.686**	0.998, 0.820	0.583, 0.564	0.791, 0.636	0.367, 0.391
$\text{SEC}_{\text{SL}}(\text{SHAC}^S_{\text{AL, Eucl}})$	**1.000**, 0.959	**0.683**, **0.686**	0.997, 0.814	0.582, 0.563	0.792, 0.632	0.367, 0.391
$\text{SEC}_{\text{SL}}(\text{SSPEC}^S)$	0.333, 0.412	0.172, 0.250	0.101, 0.119	-0.004, -0.033	0.049, 0.021	-0.041, -0.169

(SSC) scores based on SPARTACUS^S are smaller with 0.875 and 0.865 (0.435 and 0.405), respectively.

Furthermore, $\mathrm{SEC}_{\mathrm{AL}}\big(\mathrm{SPARTACUS}^S\big)$ and $\mathrm{SEC}_{\mathrm{AL}}\big(\mathrm{SHAC}^S_{\mathrm{Ward}}\big)$ show the overall best performance among the SEC methods. They almost always correctly identify the true parcellations, which is expressed by the fact that their smallest mean ARI value over all scenarios is 0.948. Hence, these methods are even able to stably identify weakly pronounced clusters. Moreover, they clearly outperform $\mathrm{SEC}_{\mathrm{AL}}\big(\mathrm{SSPEC}^S\big)$ which achieves the worst performance among the SEC methods. Also, they perform much better in identifying $K = 54$ and only slightly worse in identifying $K = 27$ compared to $\mathrm{SEC}_{\mathrm{AL}}\big(\mathrm{SHAC}^S_{\mathrm{AL,\ corr}}\big)$ and $\mathrm{SEC}_{\mathrm{AL}}\big(\mathrm{SHAC}^S_{\mathrm{AL,\ Eucl}}\big)$.

A.2.3 Performance of methods to find interesting numbers of brain regions

In a third step, we apply the subsampling based clustering stability approach and the subsampling based clustering quality approach to all data sets from the simulation study, in order to investigate whether these algorithms are able to identify the true numbers of clusters. In both approaches all eight SHAC and SSPEC methods are employed as spatial clustering methods, $B = 50$ subsamples are drawn from the respective simulated data set and parcellations are generated with $K = 2, \ldots, 100$ numbers of clusters. Moreover, in the subsampling based clustering quality approach the four internal validation measures SC, SSC, $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ are employed. The mean over the $H = 25$ scores due to the subsampling based clustering stability or quality approach that correspond to the same simulation scenario are displayed for $K = 2, \ldots, 100$ in Figure A.3 or Figures A.4-A.7, respectively.

Figure A.3 shows that the performance of the subsampling based clustering stability approach is best in combination with SPARTACUS^S, $\text{SHAC}^S_{\text{Ward}}$ and $\text{SHAC}_{\text{Ward}}$. For all scenarios of the first setting, SPARTACUS^S, $\text{SHAC}^S_{\text{Ward}}$ and $\text{SHAC}_{\text{Ward}}$ always produce two local maxima at $K = 27$ and at $K = 54$, i.e. they identify correctly the true numbers of clusters. Also in the scenarios of the second setting they always correctly produce a maximum at $K = 54$. However, the first maximum occurs always at around $K = 23$, i.e. the true number of larger clusters is slightly miscalculated. In contrast, $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$ always produce the first maximum at $K = 27$ in all six scenarios, whereas $K = 54$ is never identified as interesting number of clusters. $\text{SHAC}_{\text{AL, Eucl}}$ produces useless results in the scenarios SimBM, SimBW, SimUM and SimUW and both SSPEC methods only identify $K = 27$ in the scenarios of the first setting, but, apart from that, fail to identify any other true number of clusters.

Figures A.4-A.7 reveal the following results. The internal validation curves due to all four silhouette coefficient variants closely resemble each other. The spatial adaptations $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ produce nearly the same results as SC and SSC, respectively. Moreover, the SC and SSC curves progress very similar to each other, even though SSC generates larger scores than SC. From this finding we conclude that all considered simplifications/modifications of the SC are valid alternatives of less computational complexity for both finding interesting numbers of clusters and evaluating the quality of parcellations in the context of structural MRI data. Moreover, the (mean) internal validation scores generated by the subsampling based clustering quality approach are the larger, the larger the simulated effect is. The subsampling based clustering quality approach performs best in combination with SPARTACUS^S, $\text{SHAC}^S_{\text{Ward}}$ or $\text{SHAC}_{\text{Ward}}$. In all scenarios of the first setting, both $K = 27$ and $K = 54$ are identified as interesting numbers of clusters and also in the scenarios of the second setting $K = 54$ is always identified. Only $K = 27$ is not clearly identified in the scenarios of the second setting in combination with SPARTACUS^S, $\text{SHAC}^S_{\text{Ward}}$ or $\text{SHAC}_{\text{Ward}}$. In contrast, in combination with

$\text{SHAC}^{S}_{\text{AL, corr}}$ and $\text{SHAC}^{S}_{\text{AL, Eucl}}$, only $K = 27$ is identified in all scenarios, while $K = 54$ is not clearly identified in any scenario. Both SSPEC algorithms are neither suitable to produce high quality parcellations nor to identify the true number of clusters. More precisely, all internal validation scores due to the SSPEC algorithms are clearly worse than the internal validation scores due to the SHAC algorithms. Moreover, in the first setting with balanced parcellations only the true number of larger clusters ($K = 27$), but not the true number of smaller clusters ($K = 54$), is correctly identified based on the SSPEC algorithms and in the second setting with unbalanced parcellations the SSPEC algorithms fail completely in identifying any true number of clusters. Finally, except for $\text{SHAC}_{\text{Ward}}$, the performance of the non-standardized clustering methods is worse compared to the standardized clustering methods, especially if the simulated effect is small. E.g., in SimBM, SimBW, SimUM and SimUW $\text{SHAC}_{\text{AL, Eucl}}$ generates the worst SSC^{corr} scores among all SHAC algorithms and is also not able to identify any true number of clusters.

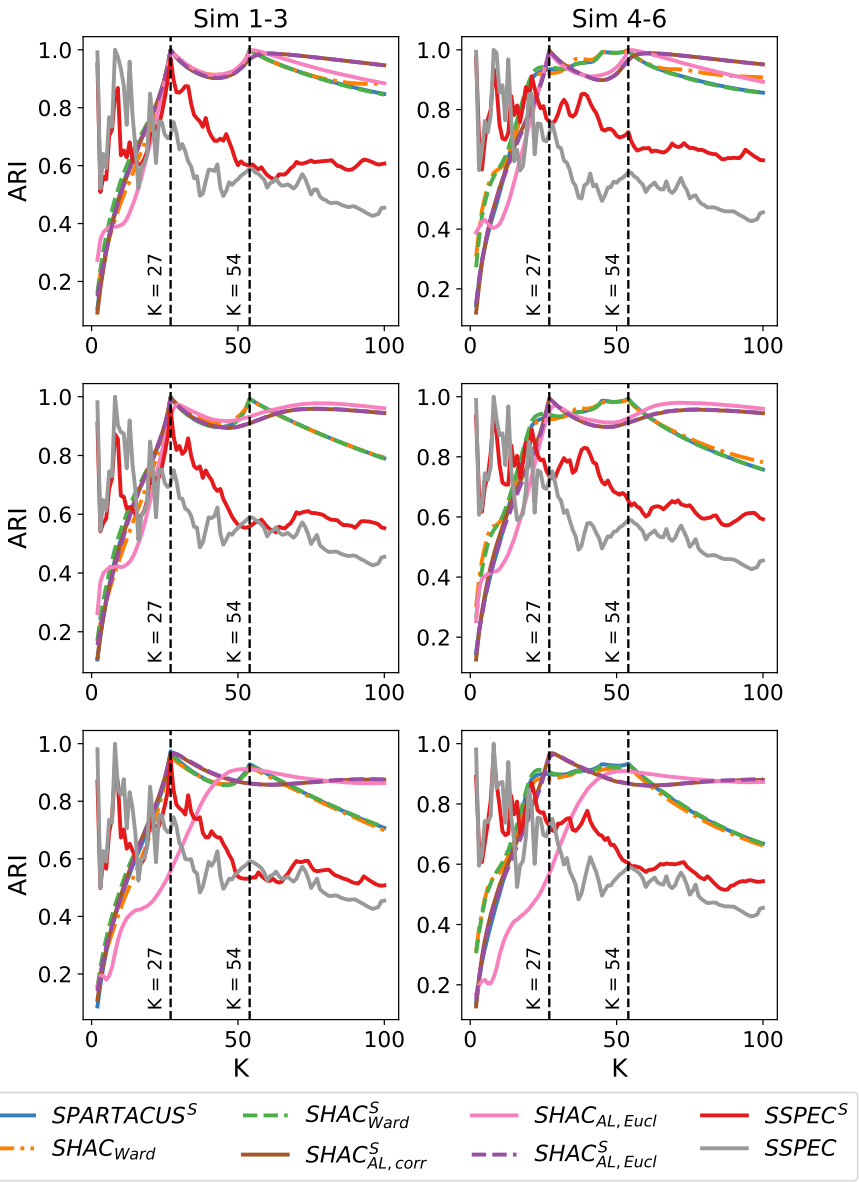

Fig. A.3 For all simulation scenarios the mean over $H = 25$ $\overline{\text{ARI}}$ scores generated by Algorithm 1 from the main manuscript is plotted for each $K = 2, \ldots, 100$.

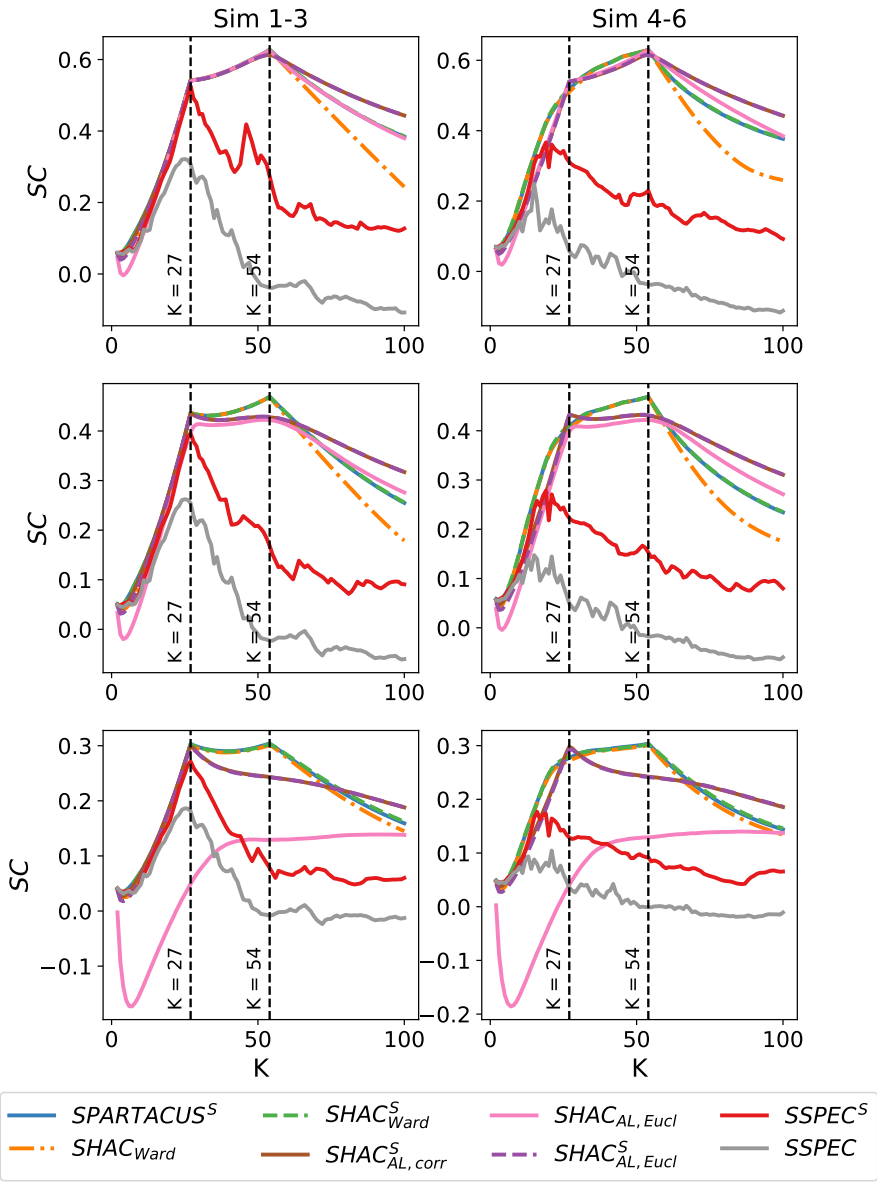

Fig. A.4 For all simulation scenarios the mean over $H = 25$ $\overline{\text{SC}}$ scores generated by Algorithm 2 from the main manuscript is plotted for each $K = 2, \ldots, 100$.

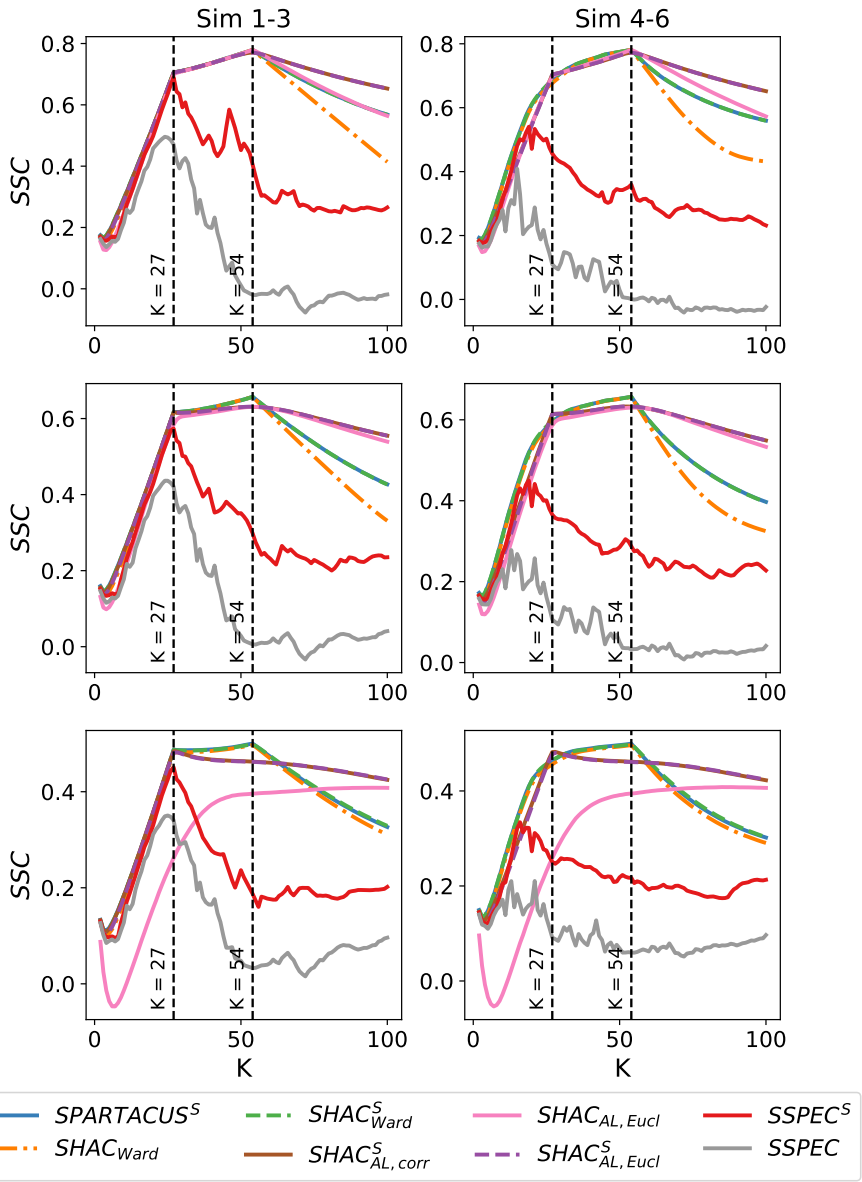

Fig. A.5 For all simulation scenarios the mean over $H = 25$ $\overline{\text{SSC}}$ scores generated by Algorithm 2 from the main manuscript is plotted for each $K = 2, \ldots, 100$.

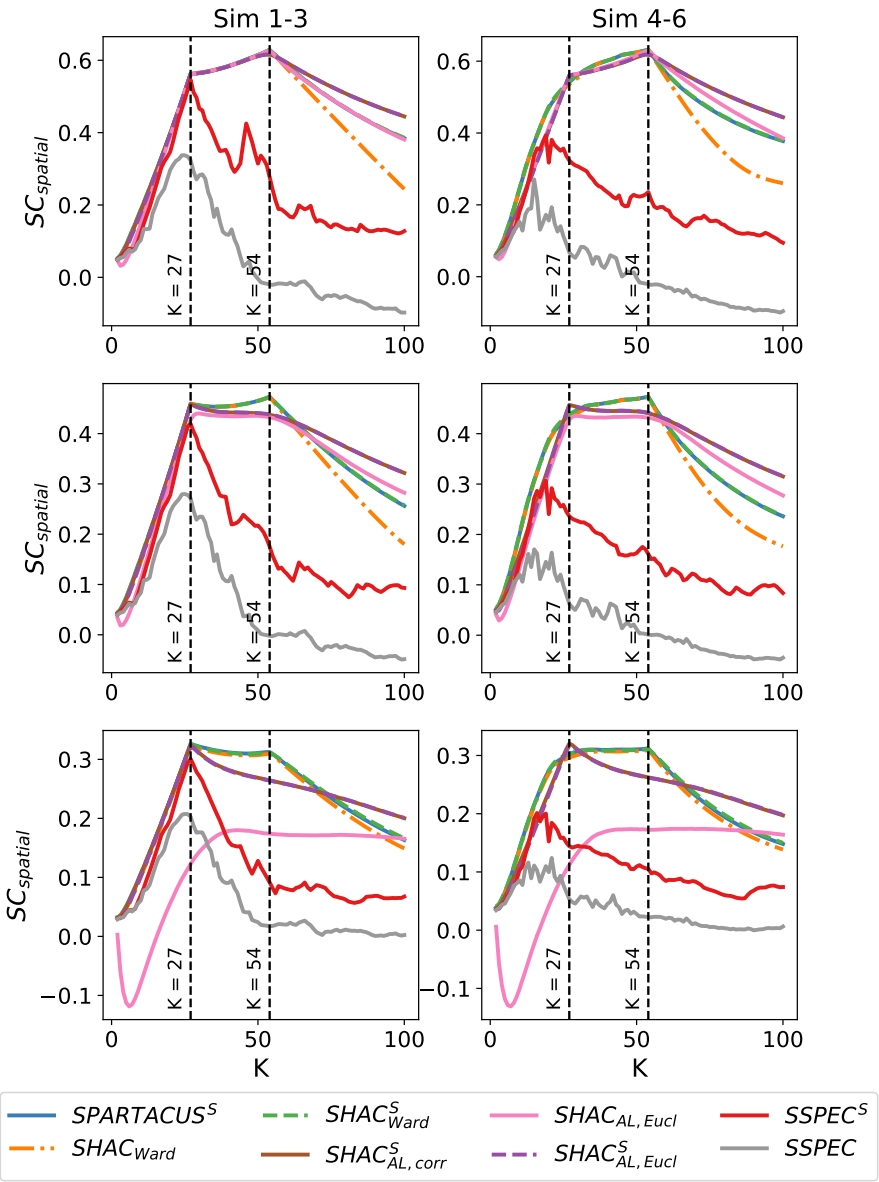

Fig. A.6 For all simulation scenarios the mean over $H = 25$ $\overline{\mathrm{SC}_{\mathrm{spatial}}}$ scores generated by Algorithm 2 from the main manuscript is plotted for each $K = 2, \ldots, 100$.

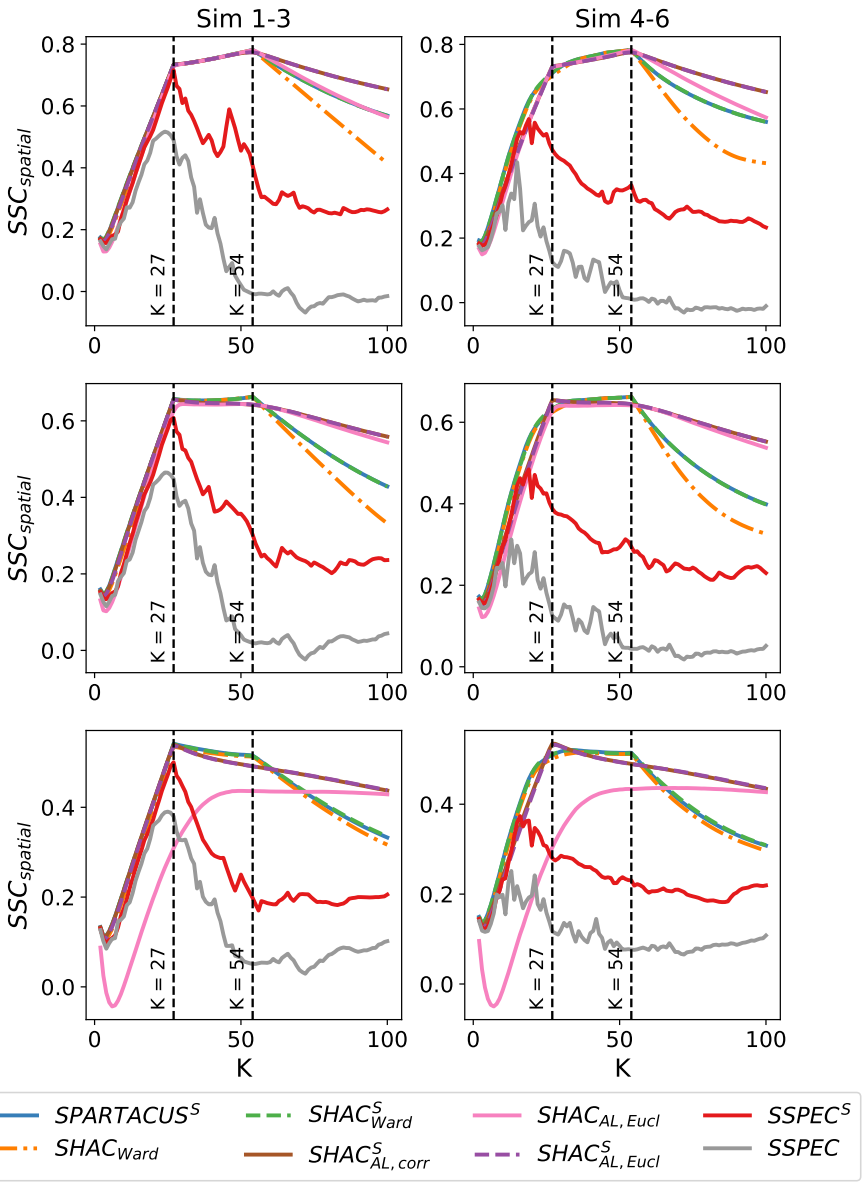

Fig. A.7 For all simulation scenarios the mean over $H = 25$ $\overline{\mathrm{SSC}_{\mathrm{spatial}}}$ scores generated by Algorithm 2 from the main manuscript is plotted for each $K = 2, \ldots, 100$.

B 1000BRAINS analysis

In this section we provide additional results obtained from our analysis of the 1000BRAINS data set that supplement the results presented in the main manuscript.

B.1 Visual analysis of non-standardized SHAC parcellations

In order to get a better understanding why $\text{SHAC}_{\text{Ward}}$ and $\text{SHAC}_{\text{AL, Eucl}}$ produce low quality parcellations (see Figure 3 from the main manuscript), the parcellations with $K = 160$ due to $\text{SHAC}_{\text{Ward}}$ and $\text{SHAC}_{\text{AL,Eucl}}$ are visualized in Figure B.8 and Figure B.9, respectively. Each figure shows a 4×5 slice plot, where each subplot displays a 2D slice over the x and y dimension of the 3D image, while fixating the z dimension. The coordinates are in the 1000BRAINS specific voxel space.

From Figure B.8 it can be observed that $\text{SHAC}_{\text{Ward}}$ produces cluster patterns which can, in a sense, be descibed as central clusters which are surrounded by multiple thin cluster rings. This patterning is not established in the field of brain parcellation at all and, therefore, it is not very surprising that the quality of these parcellations is low. Figure B.9 reveals that $\text{SHAC}_{\text{AL,Eucl}}$ produces a few very large clusters, more precisely, three large clusters with 107079, 85078 and 94755 voxels, and many small clusters (106 clusters include less than 10 voxels). This tendency to form a few large clusters and to assign outlier voxels to small or even singleton clusters could already be observed in the simulation study and is a phenomenon of average linkage hierarchical clustering that is experienced also by other researchers (Senbabaoğlu et al, 2014).

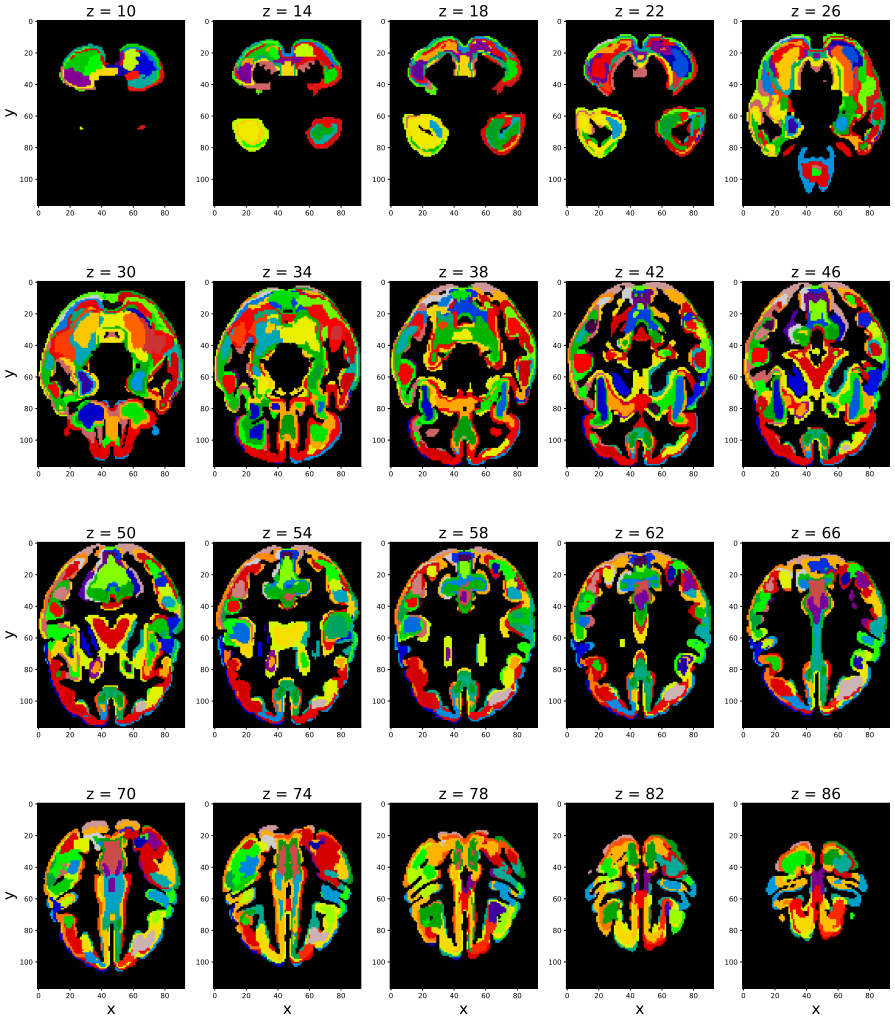

Fig. B.8 Visualization of the partition with $K = 160$ brain regions generated by $\text{SHAC}_{\text{Ward}}$ applied to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

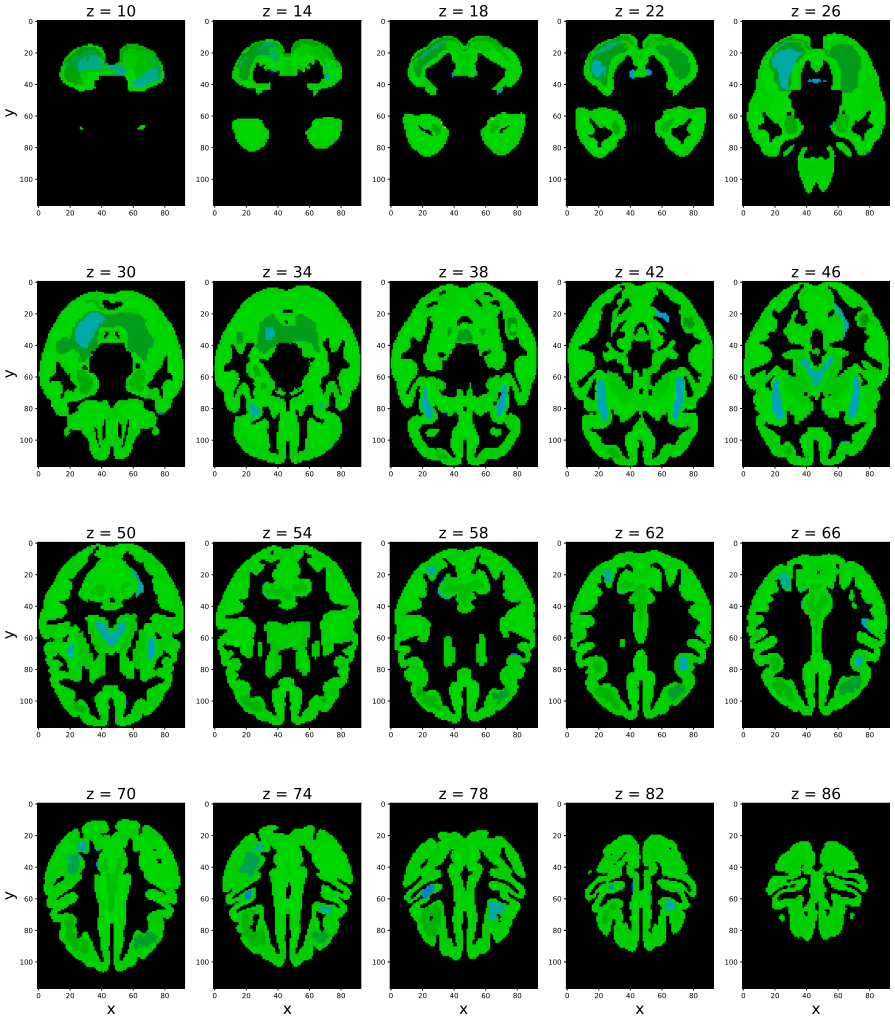

Fig. B.9 Visualization of the partition with $K = 160$ brain regions generated by $\text{SHAC}_{\text{AL,Eucl}}$ applied to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

B.2 Clustering stability and clustering quality to find interesting numbers of brain regions

We employ the subsampling based clustering quality approach as one approach to find interesting numbers of brain regions. The four standardized SHAC methods are used as spatial clustering methods and $\mathrm{SSC}_{\mathrm{spatial}}$ is employed as internal validation measure. $B = 100$ subsamples are drawn in all runs of the algorithm. We display the first derivative of the $\overline{\mathrm{SSC}_{\mathrm{spatial}}}$ curves due to $\mathrm{SHAC}^{S}_{\mathrm{Ward}}$ and $\mathrm{SHAC}^{S}_{\mathrm{AL,\ Eucl}}$ together with the corresponding $\overline{\mathrm{ARI}}$ curves generated by the subsampling based clustering stability approach in Figure B.10. This figure supplements Figure 5 from the main manuscript.

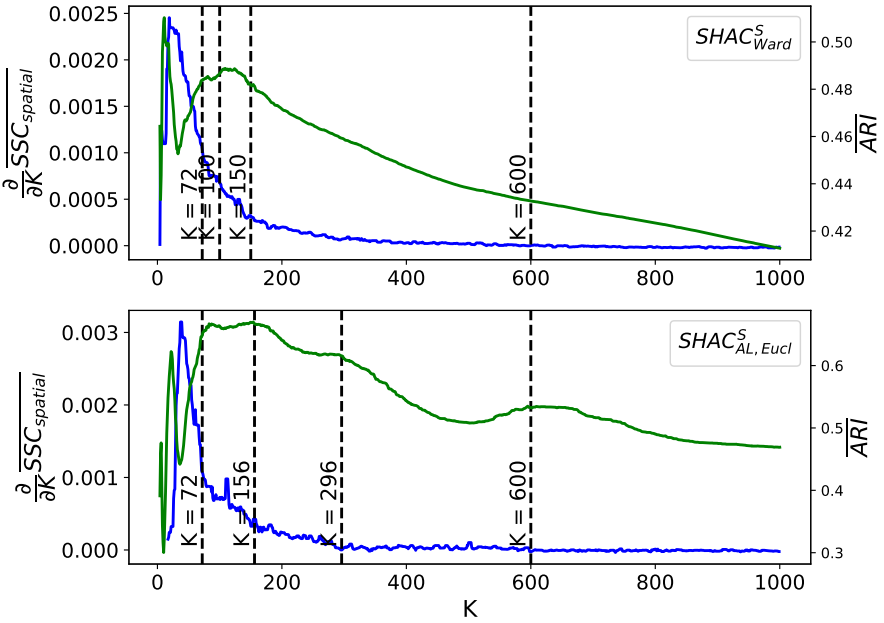

Fig. B.10 The first derivative of the $\overline{\mathrm{SSC}_{\mathrm{spatial}}}$ curve (blue) generated by the subsampling based clustering quality approach and the $\overline{\mathrm{ARI}}$ curve (green) generated by the subsampling based clustering stability approach, both with $K = 2, \ldots, 1000$ and using $\mathrm{SHAC}^{S}_{\mathrm{Ward}}$ or $\mathrm{SHAC}^{S}_{\mathrm{AL,\ Eucl}}$ as spatial clustering algorithm.

B.3 Final ensemble parcellations

The final ensemble parcellations with $K = 70$ and $K = 150$ brain regions generated by $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ and $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^S_{\mathrm{AL,\ corr}})$ based on the 1000BRAINS data are shown in Figures B.11-B.14.

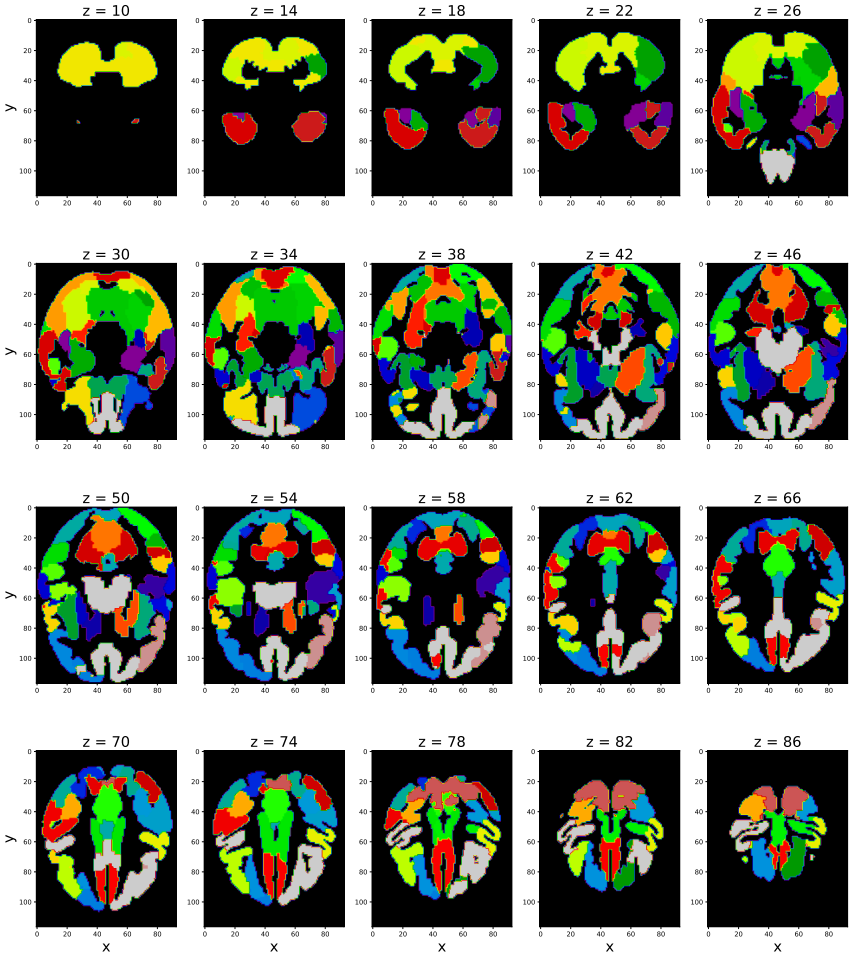

Fig. B.11 Visualization of the parcellation with $K = 70$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ to the 1000BRAINS data set.

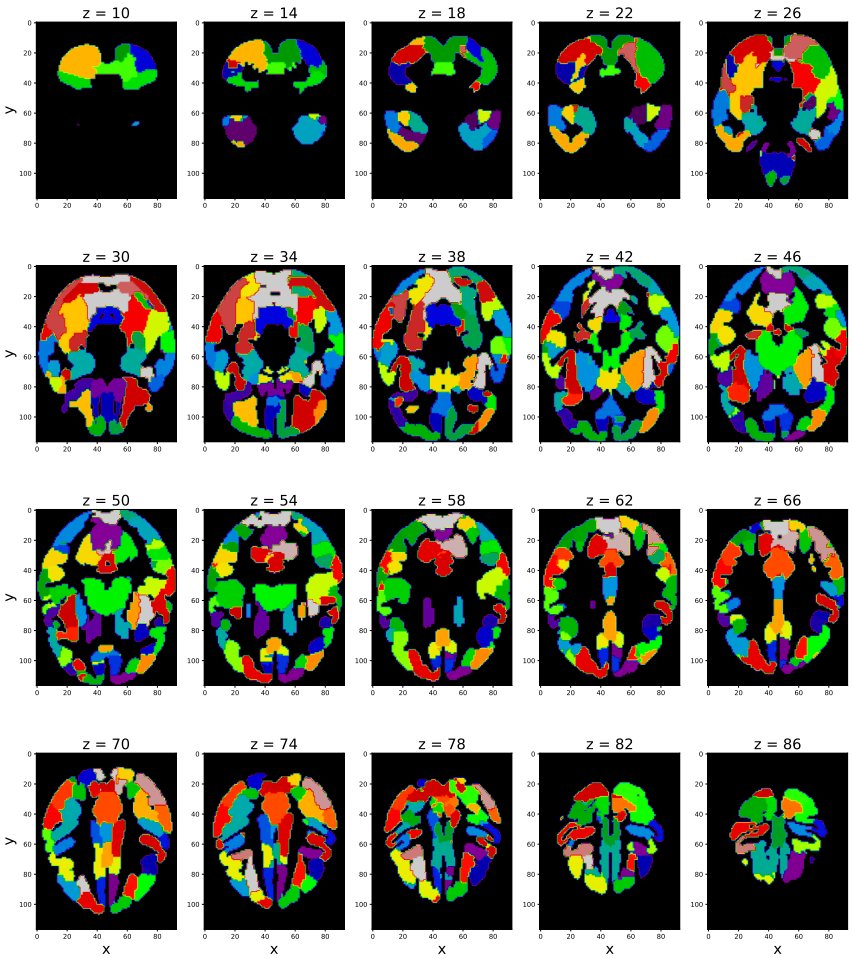

Fig. B.12 Visualization of the parcellation with $K = 150$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ to the 1000BRAINS data set.

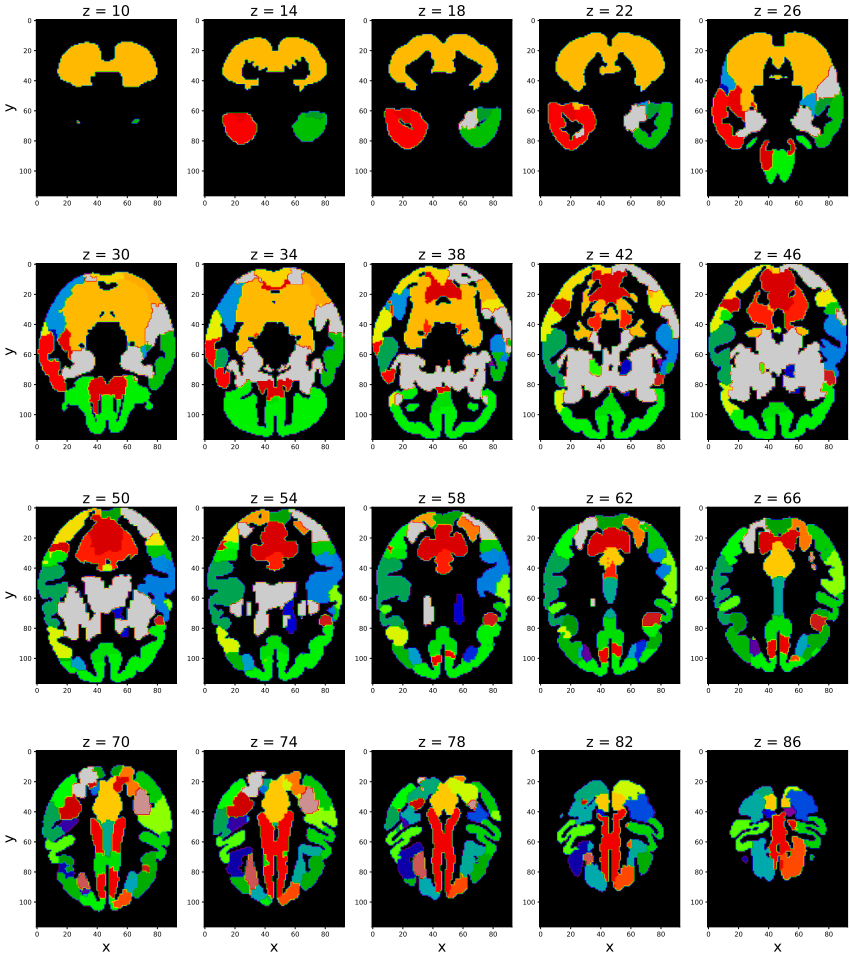

Fig. B.13 Visualization of the parcellation with $K = 70$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^{S}_{\mathrm{AL,\ corr}})$ to the 1000BRAINS data set.

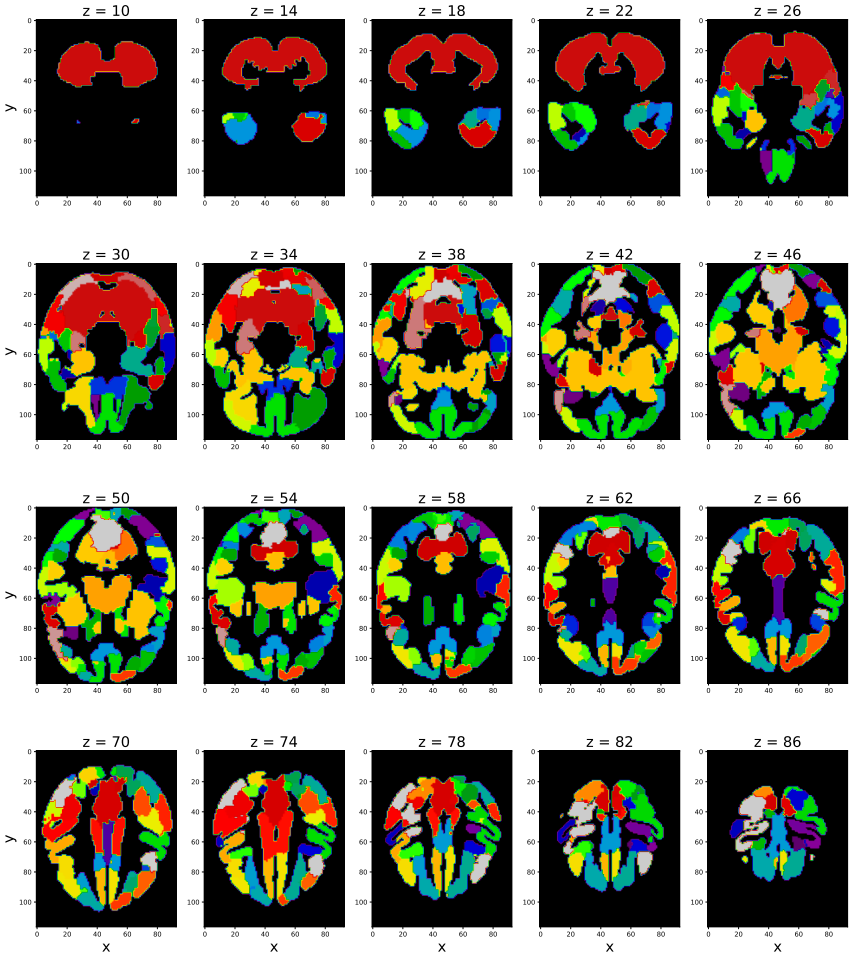

Fig. B.14 Visualization of the parcellation with $K = 150$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^{S}_{\mathrm{AL,\ corr}})$ to the 1000BRAINS data set.

B.4 Spectral and geometric parcellations

The parcellations with $K = 150$ brain regions generated by SSPEC^S and geometric clustering based on the 1000BRAINS data are displayed in Figure B.15 and Figure B.16, respectively.

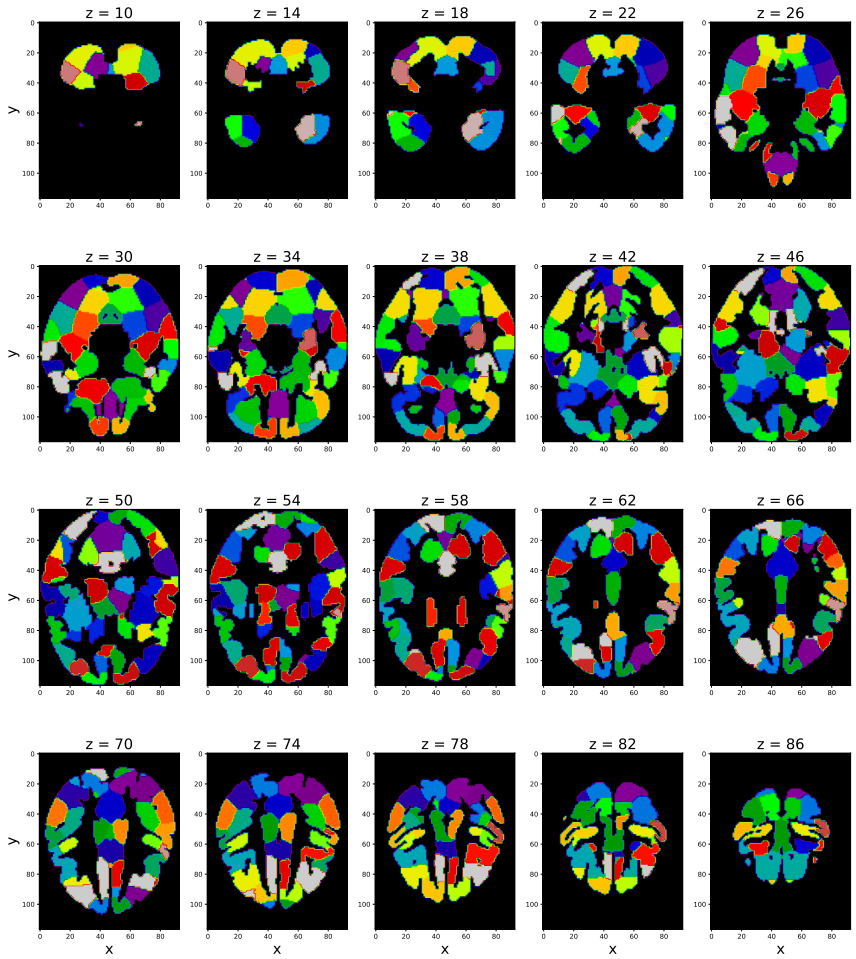

Fig. B.15 Visualization of the parcellation with $K = 150$ brain regions generated by applying SSPEC^S to the 1000BRAINS data set.

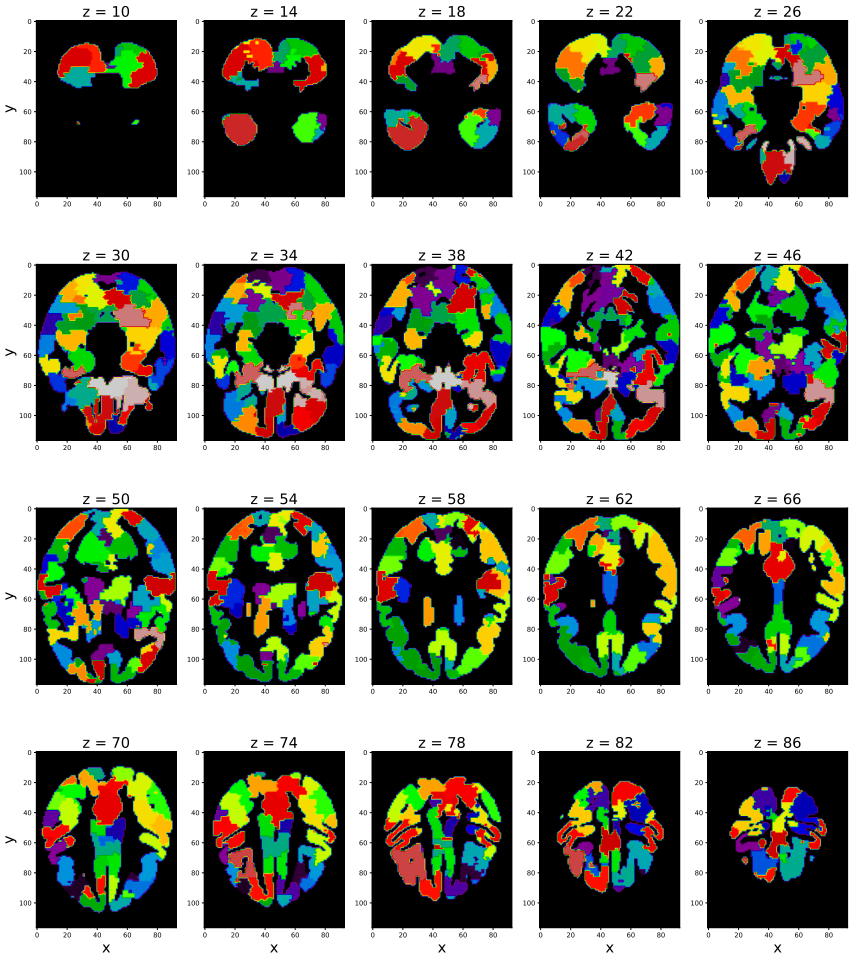

Fig. B.16 Visualization of the parcellation with $K = 150$ brain regions generated by applying geometric clustering to the 1000BRAINS data set.

B.5 Preprocessing steps for convergence analysis

In this section the preprocessing steps which are performed to quantify the convergence between parcellations from different families are described in detail.

Some issues occur when quantifying the convergence between atlases, i.e. the brain atlases neither are registered to the same brain template in MNI space, nor have the same voxel resolution, nor have the same brain coverage nor have the same numbers of brain regions. Hence, some preprocessing steps are employed.

In a first preprocessing step, all atlases are registered to the same brain template. The label information of each atlas is stored in a 3D array and the position of a voxel in this array determines its coordinate in the atlas specific voxel space. However, without further information it is unclear to which position in the scanner this voxel coordinate points to. Let (x, y, z) be a coordinate in the atlas specific voxel space. In order to obtain the corresponding coordinate in the scanner space (measured in mm^3), an affine matrix (Markiewicz, 2021)

$$\boldsymbol{M} = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & a \\ m_{2,1} & m_{2,2} & m_{2,3} & b \\ m_{3,1} & m_{3,2} & m_{3,3} & c \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

is employed. This affine matrix is multiplied with $(x, y, z, 1)^T$, i.e.

$$\boldsymbol{M} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x^* \\ y^* \\ z^* \\ 1 \end{pmatrix},$$

and (x^*, y^*, z^*) is the coordinate in scanner space corresponding to (x, y, z). Hereby, (a, b, c) is a translation which is added to (x, y, z) and

$$\boldsymbol{M}^* = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{pmatrix}$$

is a rotation / zoom matrix. Note that

$$\begin{pmatrix} x^* \\ y^* \\ z^* \end{pmatrix} = \boldsymbol{M}^* \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$

holds and, hence, (x^*, y^*, z^*) results from (x, y, z) via rotating, zooming and shifting. Since each atlas is provided with such an affine matrix, the coordinates from the atlas specific voxel spaces can all be transformed to the same scanner space. However, further calculations are best performed in voxel space. Therefore, the inverse of the $1 \times 1 \times 1\text{mm}^3$ MNI brain mask (Patterson, 2021)

$$\boldsymbol{M}_{\text{MNI}}^{-1} = \begin{pmatrix} -1 & 0 & 0 & 77 \\ 0 & 1 & 0 & -111 \\ 0 & 0 & 1 & -72 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} -1 & 0 & 0 & 77 \\ 0 & 1 & 0 & 111 \\ 0 & 0 & 1 & 72 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

is used to backtransform scanner coordinates to the reference voxel space. By this procedure, each voxel from an atlas specific voxel space is transformed to the same reference voxel space, where voxel labels (or voxel values) are transferred accordingly. Note that this reference voxel space corresponds to a voxel resolution of $1 \times 1 \times 1\text{mm}^3$.

Since not all atlases have a voxel resolution of $1 \times 1 \times 1\text{mm}^3$, a second preprocessing step which is performed after the first preprocessing step is to enlarge or shrink atlases from a lower or higher resolution voxel space, respectively, in order to fill the reference voxel space. Enlargement is accomplished by a nearest-neighbor approach, where, depending on the atlas resolution, a labeled voxel in the reference space equips its nearest unlabeled voxels with its label. If, e.g., the atlas resolution is $2\times2\times2\text{mm}^3$, $3\times3\times3\text{mm}^3$ or $4\times4\times4\text{mm}^3$, a labeled voxel in the reference space with coordinate (x^*, y^*, z^*) also equips all (unlabeled) voxels whose coordinates are in $[x^*, x^*+1]\times[y^*, y^*+1]\times[z^*, z^*+1]$, $[x^*-1, x^*+1]\times[y^*-1, y^*+1]\times[z^*-1, z^*+1]$ or $[x^*-1, x^*+2]\times[y^*-1, y^*+2]\times[z^*-1, z^*+2]$ with its label, respectively. A special case is a atlas resolution of $1.5 \times 1.5 \times 1.5\text{mm}^3$ as for the 1000BRAINS parcellations or the atlases by Varikuti et al (2018). Therefore, the following example is considered. The affine matrix of the 1000BRAINS data set is given by

$$\boldsymbol{M} = \begin{pmatrix} -1.5 & 0 & 0 & 91.5 \\ 0 & 1.5 & 0 & -127.5 \\ 0 & 0 & 1.5 & -73.5 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

A coordinate in voxel space is $(53, 50, 6)$. Then the corresponding coordinate in scanner space is calculated as

$$\boldsymbol{M} \cdot \begin{pmatrix} 53 \\ 50 \\ 6 \\ 1 \end{pmatrix} = \begin{pmatrix} 12 \\ -52.5 \\ -64.5 \\ 1 \end{pmatrix}$$

and backtransformation using $\boldsymbol{M}_{\text{MNI}}^{-1}$ yields

$$\boldsymbol{M}_{\text{MNI}}^{-1} \cdot \begin{pmatrix} 12 \\ -52.5 \\ -64.5 \\ 1 \end{pmatrix} = \begin{pmatrix} 65 \\ 58.5 \\ 7.5 \\ 1 \end{pmatrix},$$

i.e. the transformation of coordinate $(53, 50, 6)$ to the scanner space and then to the reference voxel space is $(65, 58.5, 7.5)$. Note that the entries of $(65, 58.5, 7.5)$ are not all integers and, therefore, this coordinate does not specify a position in a 3D array. Therefore, and also to fill the reference

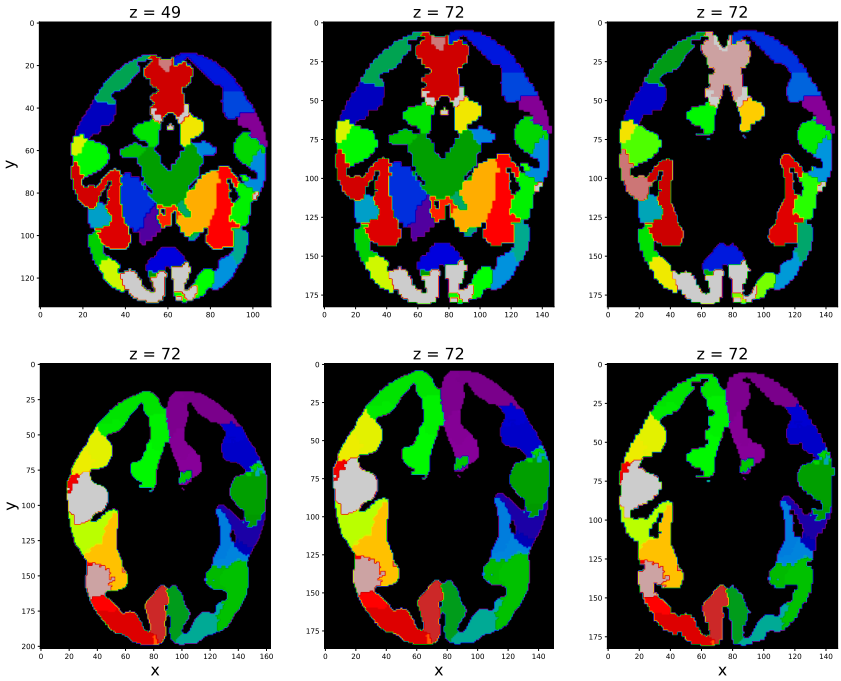

Fig. B.17 The brain slices in the first or second row correspond to the atlas with 100 brain regions from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family or the Schaefer family, respectively. The first, second or third column displays the brain slices in the atlas specific voxel space, the reference voxel space before matching or the reference voxel space after matching, respectively.

voxel space, $(65, 58.5, 7.5)$ equips all (unlabeled) voxels whose coordinates are in $[\lfloor 65 \rfloor, \lceil 65 \rceil] \times [\lfloor 58.5 \rfloor, \lceil 58.5 \rceil] \times [\lfloor 7.5 \rfloor, \lceil 7.5 \rceil]$, i.e. $(65, 58, 7)$, $(65, 59, 7)$, $(65, 58, 8)$ and $(65, 59, 8)$, with its label. More generally, if the atlas resolution is $1.5 \times 1.5 \times 1.5\mathrm{mm}^3$, a labeled voxel in the reference space with coordinate (x^*, y^*, z^*) equips all (unlabeled) voxels whose coordinates are in $[\lfloor x^* \rfloor, \lceil x^* \rceil] \times [\lfloor y^* \rfloor, \lceil y^* \rceil] \times [\lfloor z^* \rfloor, \lceil z^* \rceil]$ with its label. If the voxel resolution in the atlas specific voxel space is higher than the voxel resolution in the reference voxel space, multiple original voxels will correspond to the same reference voxel. Hence, shrinkage is accomplished by equiping each reference voxel with a randomly chosen label from the label-list of its corresponding atlas specific voxels. If, e.g., the atlas resolution is $0.5 \times 0.5 \times 0.5\mathrm{mm}^3$, all labeled voxels in the reference space which have the same coordinate (x^*, y^*, z^*) after all their coordinate entries are rounded down equip the voxel with coordinate (x^*, y^*, z^*) randomly with one of their labels.

As all atlases are registered to the same brain template, the next issue is that the brain coverage is different between the existing atlases and the atlases generated based on the 1000BRAINS data set. Technically this means that, when comparing two atlases with each other, there exist voxels in the reference space which are labeled by exactly one atlas but not by the other.

Therefore, a third preprocessing step is to consider only those voxels which are labeled by both atlases. This pairwise matching might reduce the number of brain regions in both atlases. E.g., when matching the whole brain atlases from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ family with the cerebral cortex atlases from the Schaefer family (Schaefer et al, 2018), the numbers of brain regions of the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ atlases reduce from originally, e.g., $100, 200, 300, 400$ or 500 to $95, 189, 276, 363$ or 448, respectively. In contrast, the numbers of brain regions of the atlases from the Schaefer family are not changed by this matching. This is because the brain coverage of the atlases from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ family is larger than the brain coverage of the atlases from the Schaefer family. Examplary, Figure B.17 contrasts one specific profile of the atlases with 100 brain regions from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ and the Schaefer family in the atlas specific voxel space, in the reference voxel space before matching and in the reference voxel space after matching.

The last issue to be addressed is how to pair parcellations from two different families, where convergence is only determined between pairs of parcellations. While the 1000BRAINS families, i.e. the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$, $\text{SSPEC}^S_{\text{RBF}}$ and geometric family, include parcellations with $2, \ldots, 500$ brain regions, the existing atlas families presented in Table 1 from the main manuscript include far less parcellations with selected numbers of brain regions. An intuitive idea is to assign each parcellation from an existing atlas family to that parcellation from a 1000BRAINS family, which has the same number of brain regions (before matching). However, with this strategy the numbers of brain regions can differ between pairs of parcellations after pairwise matching is performed. E.g., when pairing the parcellations with 500 brain regions from the Schaefer and the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ family, the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ parcellation contains only 448 brain regions after pairwise matching, while the Schaefer parcellation still includes 500 brain regions. Therefore, those parcellations from an existing atlas family and a 1000BRAINS family are paired, whose numbers of brain regions are closest after pairwise matching, but only if the absolute difference between their numbers of brain regions is less than or equal to 20. E.g., the parcellations with 100, 200, 300 and 400 brain regions from the Schaefer family are paired with the parcellations (before matching) with 105, 211, 323 and 444 brain regions from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ family, where, after matching, the numbers of brain regions of the Schaefer parcellations remain the same and the numbers of brain regions of the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ parcellations are 100, 200, 300 and 401, respectively.

B.6 Convergence analysis with existing parcellations

Using the ARI, the convergence of the parcellations belonging to the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$, SSPEC^S or geometric family with the parcellations from the four alternative atlas families Schaefer (Schaefer et al, 2018), Bellec (Bellec et al, 2010), Shen (Shen et al, 2013) or Glasser (Glasser et al, 2016) is quantified. The results of this analysis are shown in Figure B.18. This figure supplements Figure 9 from the main manuscript.

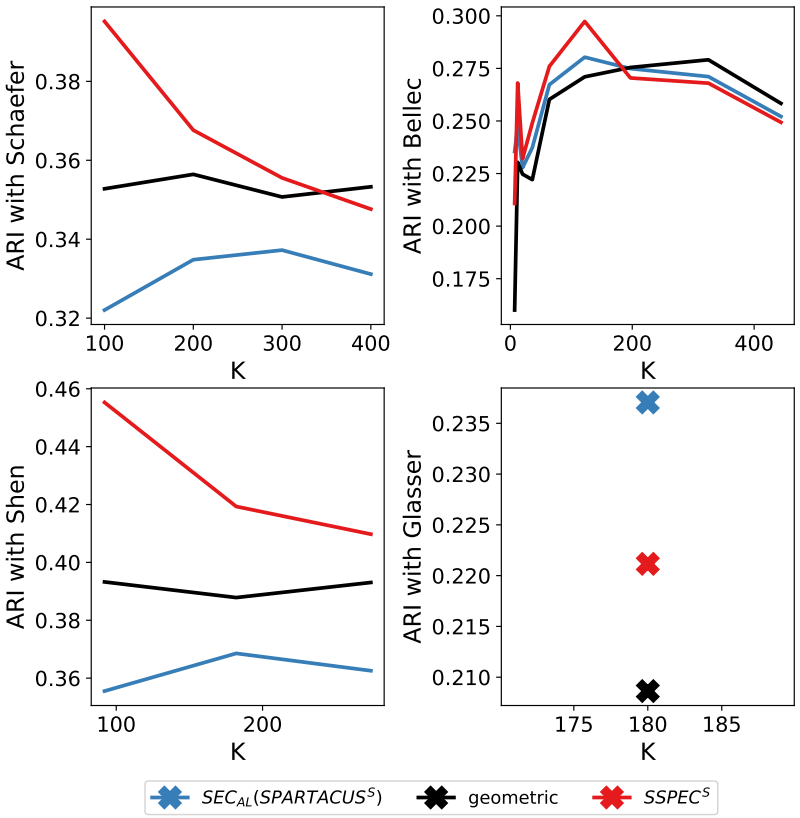

Fig. B.18 Each subplot shows the ARI based convergence of parcellations from the Schaefer, Bellec, Shen or Glasser family with the parcellations from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$, SSPEC^S and geometric family. The values on the x-axis are the numbers of clusters of the parcellations from the alternative atlas family after matching.

References

Bellec P, Rosa-Neto P, Lyttelton OC, et al (2010) Multi-level bootstrap analysis of stable clusters in resting-state fMRI. Neuroimage 51(3):1126–1139. https://doi.org/10.1016/j.neuroimage.2010.02.082

Glasser MF, Coalson TS, Robinson EC, et al (2016) A multi-modal parcellation of human cerebral cortex. Nature 536(7615):171–178. https://doi.org/10.1038/nature18933

Jones DK, Symms MR, Cercignani M, et al (2005) The effect of filter size on VBM analyses of DT-MRI data. Neuroimage 26(2):546–554. https://doi.org/10.1016/j.neuroimage.2005.02.013

Markiewicz C (2021) Coordinate systems and affines. https://nipy.org/nibabel/coordinate_systems.html, last accessed: 2021-08-13

Patterson D (2021) Image Processing Tips and Tricks. https://neuroimaging-core-docs.readthedocs.io/en/latest/pages/image_processing_tips.html, last accessed: 2021-08-12

Schaefer A, Kong R, Gordon EM, et al (2018) Local-global parcellation of the human cerebral cortex from intrinsic functional connectivity MRI. Cerebral Cortex 28(9):3095–3114. https://doi.org/10.1093/cercor/bhx179

Senbabaoğlu Y, Michailidis G, Li JZ (2014) Critical limitations of consensus clustering in class discovery. Scientific Reports 4(1):1–13. https://doi.org/10.1038/srep06207

Shen X, Tokoglu F, Papademetris X, et al (2013) Groupwise whole-brain parcellation from resting-state fMRI data for network node identification. Neuroimage 82:403–415. https://doi.org/10.1016/j.neuroimage.2013.05.081

Varikuti DP, Genon S, Sotiras A, et al (2018) Evaluation of non-negative matrix factorization of grey matter in age prediction. Neuroimage 173:394–410. https://doi.org/10.1016/j.neuroimage.2018.03.007