



**HAL**  
open science

# Statistical Inference with Ensemble of Clustered Desparsified Lasso

Jérôme-Alexis Chevalier, Joseph Salmon, Bertrand Thirion

► **To cite this version:**

Jérôme-Alexis Chevalier, Joseph Salmon, Bertrand Thirion. Statistical Inference with Ensemble of Clustered Desparsified Lasso. MICCAI, 2018, Grenade, Spain. hal-01815255

**HAL Id: hal-01815255**

**<https://hal.science/hal-01815255>**

Submitted on 13 Jun 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Statistical Inference with Ensemble of Clustered Desparsified Lasso

Jérôme-Alexis CHEVALIER<sup>1,2</sup>, Joseph SALMON<sup>2</sup>, and Bertrand THIRION<sup>1</sup>

<sup>1</sup> Parietal Team, Inria, CEA, Paris-Saclay University, France

<sup>2</sup> Telecom ParisTech, Paris, France

`jerome-alexis.chevalier@inria.fr`

**Abstract.** Medical imaging involves high-dimensional data, yet their acquisition is obtained for limited samples. Multivariate predictive models have become popular in the last decades to fit some external variables from imaging data, and standard algorithms yield point estimates of the model parameters. It is however challenging to attribute confidence to these parameter estimates, which makes solutions hardly trustworthy. In this paper we present a new algorithm that assesses parameters statistical significance and that can scale even when the number of predictors  $p \geq 10^5$  is much higher than the number of samples  $n \leq 10^3$ , by leveraging structure among features. Our algorithm combines three main ingredients: a powerful inference procedure for linear models –the so-called Desparsified Lasso– feature clustering and an ensembling step. We first establish that Desparsified Lasso alone cannot handle  $n \ll p$  regimes; then we demonstrate that the combination of clustering and ensembling provides an accurate solution, whose specificity is controlled. We also demonstrate stability improvements on two neuroimaging datasets.

## 1 Introduction

Prediction problems in medical imaging are typically high-dimensional small-sample problems. Training such models can be seen as an inference procedure. As in all research fields, this inference has to come with probabilistic guarantees in order to assess its reliability and to clarify further interpretation. In such settings, linear models have raised a strong interest. In particular the Lasso, introduced in [9], has been thoroughly investigated in [5]. Specifically, for settings in which the number of features  $p$  is greater than the number of samples  $n$  –though commensurate– numerous inference solutions have been proposed: see among others [2, 3, 8, 11, 12]. However, when  $n \ll p$ , these inference solutions are not scalable. In practice they fail to be informative, as we will show in our first simulation (cf. Sec. 3.1). Indeed, in these regimes, due to the curse of dimensionality, localizing statistical effects becomes much harder and an informative inference seems hopeless without dimensionality reduction. However, in high dimension, datasets often exhibit a particular data structure and inter-predictors correlation. This makes dimension reduction possible by the means of clustering algorithms which should respect data structure as described in [10]. The issue

with clustering-based solutions is that clustering is almost surely suboptimal and unstable; it carries some arbitrariness related to initialization or estimators heuristics. A solution to mitigate this confounding factor is to embed it in a bagging strategy, as done e.g. in [10].

Our contribution is an algorithm for statistical inference in high-dimensional scenarios, combining the Desparsified Lasso procedure, first introduced in [12], with clustering and bagging steps: the Ensemble of Clustered Desparsified Lasso. We describe it in detail and provide experiments on simulated and real data to assess its potential on multivariate linear models for medical imaging.

## 2 Theoretical and Algorithmic Framework

**Notations.** For clarity, scalars are denoted with normal font, vectors with bold lowercase, and matrices with bold uppercase letters. For  $p \in \mathbb{N}$ ,  $[p] = \{1, \dots, p\}$ .

**Inference on Linear Models.** Our aim is to give confidence bounds on the coefficients of the parameter vector denoted  $\mathbf{w}^*$  in the following linear model:

$$\mathbf{y} = \mathbf{X}\mathbf{w}^* + \sigma_*\boldsymbol{\varepsilon} \ , \quad (1)$$

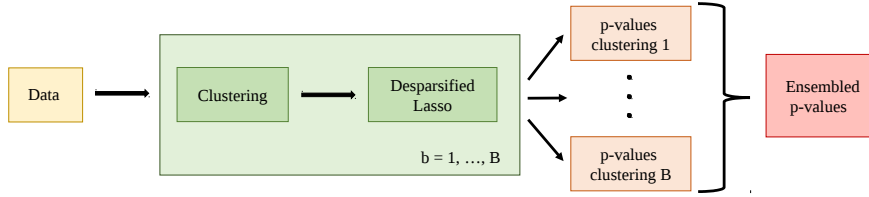
where  $\mathbf{y} \in \mathbb{R}^n$ ,  $\mathbf{X} \in \mathbb{R}^{n \times p}$ ,  $\mathbf{w}^* \in \mathbb{R}^p$ ,  $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$  and  $\sigma_* > 0$ . The matrix  $\mathbf{X}$  is the design matrix, its columns are called the predictors,  $\mathbf{y}$  is called the response vector,  $\boldsymbol{\varepsilon}$  is the noise (or the error vector) of the model and  $\sigma_*$  is the (unknown) noise standard deviation. The signal to noise ratio (SNR), defined by  $\text{SNR}_y = \|\mathbf{X}\mathbf{w}^*\|_2 / (\sigma_* \|\boldsymbol{\varepsilon}\|_2)$ , is a measure that describes the noise regime in any given experiment. The true support is defined as  $S_* = \{j \in [p]; w_j^* \neq 0\}$  and its size is  $s_* = |S_*|$ . It is noteworthy that our problem is not a prediction problem, we are not aiming at finding  $\hat{\mathbf{w}}$  minimizing  $\|\mathbf{X}\hat{\mathbf{w}} - \mathbf{X}\mathbf{w}^*\|_2$ , but an estimation problem, in which we want to control  $\|\hat{\mathbf{w}} - \mathbf{w}^*\|_\infty$  statistically.

**Desparsified Lasso for High-Dimensional Inference.** The Desparsified Lasso (DL) estimator denoted  $\hat{\mathbf{w}}^{\text{DL}}$ , introduced in [12], can be seen as a generalization of the Ordinary Least Squares (OLS) estimator for inference in  $n < p$  settings. Under some assumptions (notably the sparsity of  $\mathbf{w}^*$ ) that are made explicit in [3],  $\hat{\mathbf{w}}^{\text{DL}}$  has the following property:

$$\forall j \in [p], \quad \sigma_*^{-1}(\Omega_{jj})^{-1/2}(\hat{w}_j^{\text{DL}} - w_j^*) \sim \mathcal{N}(0, 1) \ , \quad (2)$$

where the diagonal of  $\boldsymbol{\Omega}$  (estimated precision matrix; inverse of  $\mathbf{X}^\top \mathbf{X}/n$ ) is computed concurrently with  $\hat{\mathbf{w}}^{\text{DL}}$ , as described in [12]. From (2) we can compute confidence intervals and p-values of the coefficients of the weight vector.

In [3], several high-dimensional inference solutions are discussed and compared. DL displays an interesting trade-off between good control of the family-wise error rate (FWER) and strong power. The FWER is defined as  $\text{FWER} = \text{Prob}(\text{FP} \geq 1)$  where FP is the number of false positives.



**Fig. 1.** To leverage Desparsified Lasso inference procedure on medical images, our algorithm relies on feature clustering and an ensembling step on randomized solutions.

**Clustering to Handle Structured High-Dimensional Data.** In high-dimensional inference, variables are often highly correlated. Specifically, a medical image has a 3D representation and a given voxel is highly correlated with its neighbors;  $\mathbf{w}$  obviously carries the same structure. In addition  $n \ll p$  and  $n < s_*$  make the statistical inference challenging without data structure assumptions as shown in [3]. To leverage data structure, we introduce a clustering step that reduces data dimensionality before applying our inference procedure. Here, we consider a spatially-constrained hierarchical clustering algorithm described in [10] that uses Ward criterion while respecting the image geometrical structure. The combination of this clustering algorithm and the DL inference procedure will be referred to as the Clustered Desparsified Lasso (CDL) algorithm.

**Bagging to Alleviate Dependency on Clustering.** It is preferable not to rely on a particular clustering as small perturbations on it have a dramatic impact on the final solution. We followed the approach presented in [10] that argues in favor of the randomization over a spatially-constrained clustering method: to build  $B$  clusterings of the predictors, they use the same clustering method but with  $B$  different random subsamples of size  $\lfloor 0.7n \rfloor$  from the full sample.

**Inference with Ensemble of Clustered Desparsified Lasso.** We now have all the elements to present our Ensemble of Clustered Desparsified Lasso (ECDL) algorithm which is summarized in Fig. 1. ECDL consists in  $B$  repetitions of the CDL algorithm (using random subsamples of size  $\lfloor 0.7n \rfloor$  for the clustering and the full sample for the DL procedure) and an ensembling step analogous to the bagging method introduced by [1]. Once the CDL algorithm has been run  $B$  times, we have  $B$  partitions into  $C$  clusters, each cluster being associated with a p-value. We denote by  $P^{(b,c)}$  the p-value for the  $c^{th}$  cluster in the  $b^{th}$  fold. The p-value  $P_j^{(b)}$  of the coefficient  $j \in [p]$  in the  $b^{th}$  repetition is  $P^{(b,c)}$  whenever  $j$  belongs to cluster  $c$ , i.e. we attribute the same p-value to all the predictors in a given cluster. This yields  $B$  p-values for each coefficient. Finally, to ensemble the p-values one has to use specific techniques which ensure that the resulting p-value is meaningful as a frequentist hypothesis test. Thus, to derive the p-value  $P_j$  of the  $j^{th}$  coefficient, we have considered the ensembling solution presented in [8] that has the required properties and consists in taking the median of  $\{P_j^{(b)} \text{ for } b \in [B]\}$  multiplied by 2.

### 3 Simulation and Experimental Results

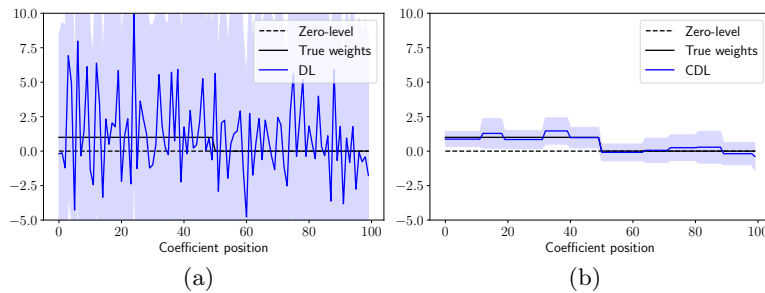
#### 3.1 First Simulation: the Importance of Dimension Reduction

**Simulation.** This simulation has a 1D structure and we set  $n = 100$  and  $p = 2000$ . We construct the design matrix  $\mathbf{X}$  such that predictors are normally distributed and two consecutive predictors have a fixed correlation  $\rho = 0.95$ . The weight  $\mathbf{w}^*$  is such that  $w_j^* = 1$  for  $1 \leq j \leq 50$  and  $w_j^* = 0$  otherwise, then  $s_* = 50$ . We also set  $\sigma_* = 10$  such that  $\text{SNR}_y = 3$  (cf. Sec. 3.3).

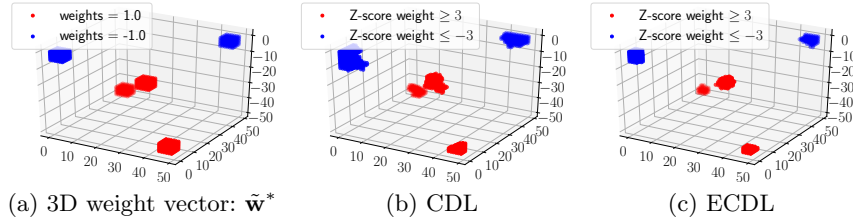
**Results.** We compare the DL procedure applied to the uncompressed data, displayed in Fig. 2-(a), and the CDL algorithm in Fig. 2-(b). The number of clusters, whose impact will be discussed in Sec. 4, has been set to  $C = 200$ , allowing to reduce the dimension from  $p = 2000$  to  $C = 200$  before performing the inference. This reduction tames the estimator variance and yields useful confidence intervals that could not be reached by DL only.

#### 3.2 Second Simulation: Improvement by Bootstrap and Aggregation

**Simulation.** Here, we consider a simulation with a 3D structure, that aims at approximating the statistics of the Oasis experiment (cf. Sec. 3.3). The volume considered is a 3D-cube with edge length  $H = 50$ , with  $n = 400$  samples and  $p = H^3 = 125\,000$  predictors (voxels). To construct  $\mathbf{w}^*$ , we define a 3D weight vector  $\tilde{\mathbf{w}}^*$  with five regions of interest (ROIs) represented in Fig. 3-(a) and then make a bijective transformation of  $\tilde{\mathbf{w}}^*$  in a vector of size  $p$ . Each ROI is a cube of width  $h = 6$ , leading to a size of support  $s_* = 5h^3 = 1\,080$ . Four ROIs are situated in corners of the cubic map and the last ROI is situated in the center of the cube. To construct  $\mathbf{X}$ , we first define the 3D design matrix  $\tilde{\mathbf{X}}$  from  $p$  random normal vectors of size  $n$  smoothed with 3D Gaussian filter with bandwidth  $\sigma_{\text{smth}}$  (smoothing is performed across all predictors for each sample), then we use the same transformation as before and derive the  $n \times p$  design matrix. The choice  $\sigma_{\text{smth}} = 2$  is made to achieve similar correlations as for the Oasis experiment. We also set  $\sigma_* = 8$ , i.e.  $\text{SNR}_y = 3$  (cf. Sec. 3.3).



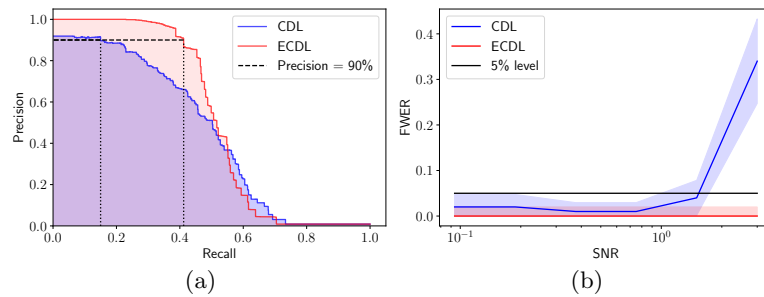
**Fig. 2.** (a) 95% coefficient intervals given by the raw Desparsified Lasso (DL) fail to retrieve the true support. (b) 95% coefficient intervals given by the Clustered Desparsified Lasso (CDL) are much narrower, and yield a good support accurately.



**Fig. 3.** In this simulation, comparing the original 3D weight vector with CDL and ECDL solutions, we observe that the ECDL solution is much more accurate.

**Results.** To derive the ECDL solutions we aggregated  $B = 25$  different CDL solutions during the ensembling step. To obtain the results presented in Fig. 4, we ran 100 simulations. In Fig. 4-(a), we display the precision-recall curve (cf. Scikit-learn `precision_recall_curve` function) of the solutions obtained by each algorithm with  $C = 500$  clusters. ECDL very strongly outperforms CDL: for precision of at least 90%, the ECDL recall is 42% while the CDL recall is only 16%.

In order to check the FWER control, we define a neutral region that separates ROIs from the non-active region. Indeed, since the predictors are highly correlated, the detection of a null predictor in the vicinity of an active one is not a mistake. Thus, neutral regions enfold ROIs with a margin of 5 voxels. We compare different values of  $\sigma_*$  from  $2^3$  to  $2^8$  giving  $\text{SNR}_y$  lying between .1 and 3. In Fig. 4-(b), one can observe that the FWER is always well controlled using ECDL; the later is even conservative since the empirical FWER stays at 0% for a 5% nominal level. On the opposite, the FWER is not well controlled by CDL: its empirical value goes far above the 5% rate for high SNR. This is due to the shape of the discovered regions that do not always correspond to the exact shape and location of ROIs. This effect is also observable watching thresholded Z-score maps yielded by CDL and ECDL in Fig. 3. By increasing the number of clusters, we would obtain discovered regions more similar to the true ROIs, yet their statistical significance would drop and the power would collapse.



**Fig. 4.** (a) The precision-recall curve for the recovery of  $\mathbf{w}$  is much better adding an ensembling step over CDL. (b) FWER (nominal rate 5%) is well controlled by the ECDL algorithm while for high level of SNR it is not controlled by the CDL algorithm.

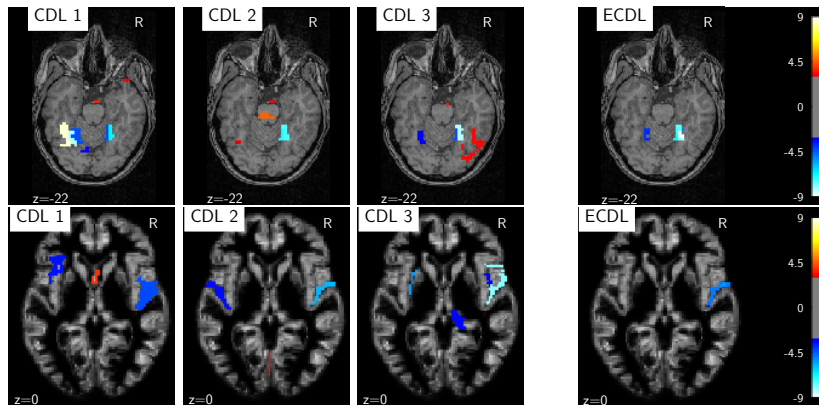
### 3.3 Experiments on MRI Datasets

**Haxby Dataset.** Haxby is a functional MRI dataset that maps the brain responses of subjects watching images of different objects (see [6]). In our study we only consider the responses related to images of faces and houses for the first subject, to identify brain regions that discriminate between these two stimuli, assuming that this problem can be modeled as a regression problem. Here  $n = 200$ ,  $p = 24k$ , (estimated)  $\text{SNR}_y = 1.0$  and we used  $C = 500$  and  $B = 25$ .

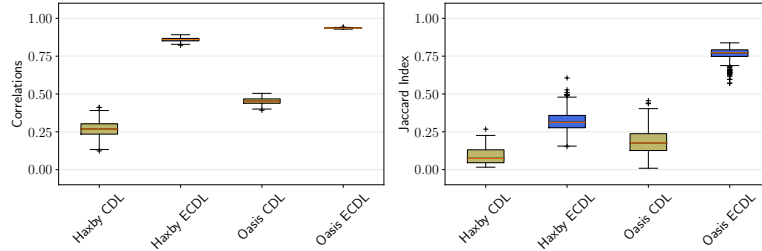
**Oasis Dataset.** The Oasis MRI dataset (see [7]) provides anatomical brain images of several subjects together with their age. The SPM voxel-based morphometry pipeline was used to obtain individual gray matter density maps. We aim at identifying which regions are informative to predict the age of a given subject. Here  $n = 400$ ,  $p = 125k$  and (estimated)  $\text{SNR}_y = 3.0$ ; we also took  $C = 500$  and  $B = 25$  as in Sec. 3.2.

**Results.** The results of these experiments are displayed in Fig. 5 with Z-transform of the p-values. For clarity, we thresholded the Z-score maps at 3 (and  $-3$ ) keeping only the regions that have a high probability of being discriminative. The solutions given by the CDL algorithm with three different choices of clustering look noisy and unstable while the ECDL solution defines a synthesis of the CDL results and exhibits a nice symmetry in the case of Haxby. Thus, these results clearly illustrate that the ensembling step removes the arbitrariness due to the clustering.

**Stability of Bagging Estimator.** This last experiment quantifies the gain in stability when adding the ensembling step. From the two previous experiments,



**Fig. 5.** Results of the CDL and ECDL algorithms on Haxby (top) and Oasis (bottom) experiments. CDL algorithm outcomes are highly dependent on the clustering, which creates a jitter in the solution. Drawing consensus among many CDL results, ECDL removes the arbitrariness related to the clustering scheme.



**Fig. 6.** Correlation (left) and Jaccard Index (right) are much higher with the ECDL algorithm than with CDL across 25 replications of the analysis of the imaging datasets.

we derive 25 ECDL solution maps (with  $B = 25$ ) and 25 CDL solution maps and measure the variability of the results. Correlation between the full maps and Jaccard index of the detected areas (here, voxels with an absolute  $Z$ -score greater than 3) show that ECDL is substantially more stable than CDL.

## 4 Discussion

**Recapitulation.** We have introduced ECDL, an algorithm for high-dimensional inference on structured data which scales even when the number of predictors  $p \geq 10^5$  is much higher than the number of samples  $n \leq 10^3$ . It can be summarized as follows: i) perform  $B$  repetitions of the CDL algorithm, that runs Desparsified Lasso (DL) inference on a model compressed by clustering, yielding several p-values for each predictor; ii) use an ensemble method aggregating all p-values to derive a single p-value per predictor. In Sec. 3.1, we have shown that the clustering step, justified by specific data structures and locally high inter-predictor correlation, was necessary to yield an informative inference solution when  $n \ll p$ . Then, we have demonstrated, in Sec. 3.2, that randomizing and bagging the CDL solution improves the control of the FWER and the precision-recall curve. While the ensembling step obviously removes the arbitrariness of the clustering choice, in Sec. 3.3 we showed it also increases stability.

**ECDL Parameter Setting.** The number of clusters  $C$  is the main free parameter, and an optimal value depends on characteristics of the data (inter-predictor correlation, SNR). In our simulations, we observe a bias/variance trade-off: a small number of clusters reduces variance and enhances statistical power, while a greater number yields refined solutions. The ensembling step helps improving shape accuracy without loss in sensitivity, as the combination of multiple CDL solutions recovers finer spatial information.

**Computational Cost of ECDL.** The most expensive step is the DL inference, which includes the resolution of  $\mathcal{O}(C)$  Lasso problems with  $n$  samples and  $C$  features. This is repeated  $B$  times in ECDL, making it embarrassingly parallel; we could run the ECDL algorithm on standard desktop stations with  $n = 400$ ,  $C = 500$  and  $B = 25$  in less than 10 minutes.



**Additional Work Related to ECDL.** In Haxby experiment (cf. Sec. 3.3) we approximated the problem as a regression one. Thus, an interesting extension would be to adapt ECDL to classification settings. Another matter is the comparison with bootstrap and permutation-based approaches e.g. [4], that we leave for future work. Note however that, in [3], a study of bootstrap approaches points out some unwanted properties and they do not outperform DL.

**Usefulness for Medical Imaging.** For structured high-dimensional data, our algorithm is relevant to assess the statistical significance of a set of predictors when fitting a target variable. Our experimental results show that ECDL is very promising for inference problems in medical imaging.

**Acknowledgements.** This research was supported by Labex DigiCosme (project ANR-11-LABEX-0045-DIGICOSME) operated by ANR as part of the program "Investissement d'Avenir" Idex Paris Saclay (ANR-11-IDEX-0003-02). This work is also funded by the FAST-BIG project (ANR-17-CE23-0011).

## References

1. L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
2. P. Bühlmann. Statistical significance in high-dimensional linear models. *Bernoulli*, 19(4):1212–1242, 09 2013.
3. R. Dezeure, P. Bühlmann, L. Meier, and N. Meinshausen. High-dimensional inference: Confidence intervals,  $p$ -values and R-Software hdi. *Statist. Sci.*, 30(4):533–558, 2015.
4. B. Gaonkar and C. Davatzikos. Deriving statistical significance maps for svm based image classification and group comparisons. 2012.
5. T. J. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.
6. J. Haxby, I. Gobbini, M. Furey, A. Ishai, J. Schouten, and P. Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.
7. D. Marcus, T. Wang, J. Parker, J. Csernansky, J. Morris, and R. Buckner. Open access series of imaging studies (oasis): Cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *Journal of Cognitive Neuroscience*, 19(9):1498–1507, 2007.
8. N. Meinshausen, L. Meier, and P. Bühlmann. P-values for high-dimensional regression. *J. Amer. Statist. Assoc.*, 104(488):1671–1681, 2009.
9. Robert Tibshirani. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 58:267–288, 1994.
10. G. Varoquaux, A. Gramfort, and B. Thirion. Small-sample brain mapping: sparse recovery on spatially correlated designs with randomization and clustering. 2012.
11. L. Wasserman and K. Roeder. High-dimensional variable selection. *Ann. Statist.*, 37(5A):2178–2201, 2009.
12. C.-H. Zhang and S. S. Zhang. Confidence intervals for low dimensional parameters in high dimensional linear models. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 76(1):217–242, 2014.