Tuning parameter selection for voxel-wise brain connectivity estimation via low dimensional submatrices

Bin Nan

Department of Biostatistics
University of Michigan

Joint work with Hai Shu

January 31 – February 5, 2016
BIRS Workshop on Neuroimaging, Banff

Notation

- Data matrix: $\mathbf{X}_{p \times n} = (X_{ij})_{p \times n}$
- Columns of $\mathbf{X}_{p \times n}$: $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$, a series of brain images (resting state fMRI) acquired at n consecutive time points, where each image contains p voxels.
- Parameters of interest:
  - **Covariance matrix:** $\boldsymbol{\Sigma} = \mathrm{cov}(\boldsymbol{X}_k) = (\mathrm{cov}(X_{ik}, X_{jk}))_{p \times p} = (\sigma_{ij})_{p \times p}$ (or its normalization, the correlation matrix $\mathbf{R}$), which is the same for all $k = 1, \ldots, n$.
  - **Precision matrix:** $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$.

Temporal dependence

- There is a rich literature about estimating large covariance matrix Σ (with $p > n$) when $\boldsymbol{X}_1, ..., \boldsymbol{X}_n$ are i.i.d.
- Recently we have obtained the convergence results for large covariance/precision matrix estimates for **temporally dependent** data under the following assumption:

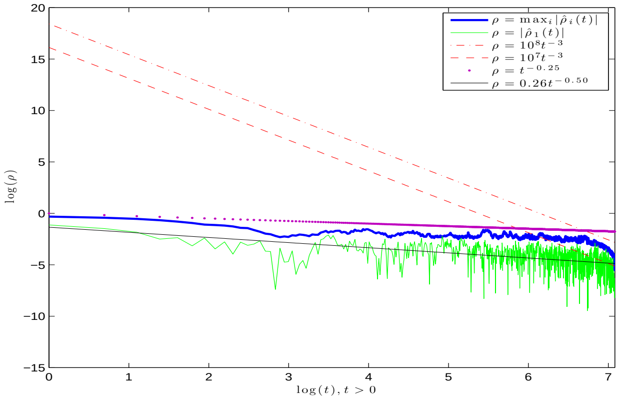
$$\max_{k,l} |\rho_{kl}^{ij}| \leq C_0 |i-j|^{-\alpha}$$

for all $i \neq j$, where $\rho_{kl}^{ij} = \mathrm{cov}(X_{ki}, X_{lj}) / \sqrt{\sigma_{kk}\sigma_{ll}}$ is the cross-correlation, and C_0 and α are fixed positive constants.

- We name it polynomial-decay-dominated (PDD) temporal dependence.

An image data example

rs-fMRI from the Human Connectome Project to assess brain functional connectivity:



Generalized thresholding for covariance matrix

- For a thresholding parameter $\tau \geq 0$, define a generalized thresholding function by $s_\tau : \mathbb{R} \to \mathbb{R}$ satisfying:
  - (i) $|s_\tau(z)| \leq |z|$;
  - (ii) $s_\tau(z) = 0$ for $|z| \leq \tau$;
  - (iii) $|s_\tau(z) - z| \leq \tau$.
- Here we only consider hard thresholding $s_\tau^H(z) = z\mathbf{1}(|z| > \tau)$ and soft thresholding $s_\tau^S(z) = \text{sign}(z)(|z| - \tau)_+$
- Thresholding estimation: $S_\tau(\hat{\Sigma}) = (s_\tau(\hat{\sigma}_{ij}))_{p\times p}$, where $\hat{\Sigma}$ is the sample covariance matrix.

SPICE estimation for precision matrix

- SPICE is a modification of the graphic lasso method.
- Consider

$$\hat{\Omega}_\lambda = \hat{\boldsymbol{W}}^{-1}\hat{\boldsymbol{K}}_\lambda\hat{\boldsymbol{W}}^{-1}$$

with

$$\hat{\boldsymbol{K}}_\lambda = \arg\min_{\boldsymbol{K}\succ 0,\, \boldsymbol{K}=\boldsymbol{K}^T} \left\{ \mathrm{tr}(\boldsymbol{K}\hat{\boldsymbol{R}}) - \log\det(\boldsymbol{K}) + \lambda|\boldsymbol{K}|_{1,\mathrm{off}} \right\},$$

$\hat{\mathbf{W}} = \mathrm{diag}\{\sqrt{\hat{\sigma}_{11}}, \ldots, \sqrt{\hat{\sigma}_{pp}}\}$, and $\hat{\boldsymbol{R}}$ is the sample correlation matrix.

Tuning parameter

- We write $x_n \asymp y_n$ if there exists some constant $C > 1$ such that $C^{-1} \leq \liminf y_n/x_n \leq \limsup y_n/x_n \leq C$.
- For any fixed α, if $p \geq n^c$ for some constant c, define

$$\tau' \asymp \begin{cases} n^{-\frac{\alpha}{2}}(\log p)^{\frac{1}{2}}, & 0 < \alpha < 1, \\ n^{-\frac{1}{2}}\left[(\log n)(\log p)\right]^{\frac{1}{2}}, & \alpha = 1, \\ n^{-\frac{1}{2}}(\log p)^{\frac{1}{2}}, & \alpha > 1. \end{cases}$$

Covariance matrix estimator

- For sufficiently large constant $M > 0$, if $\tau = M\tau'$, then

$$\|S_\tau(\hat{\boldsymbol{\Sigma}}) - \boldsymbol{\Sigma}\|_2 = O_p\left(c_0(p)\tau'^{1-q}\right),$$

$$\frac{1}{p}\|S_\tau(\hat{\boldsymbol{\Sigma}}) - \boldsymbol{\Sigma}\|_F^2 = O_p\left(c_0(p)\tau'^{2-q}\right)$$

- The results above hold when $\hat{\boldsymbol{\Sigma}}, \boldsymbol{\Sigma}$ is replaced by $\hat{\mathbf{R}}, \mathbf{R}$ respectively, where $\hat{\mathbf{R}}$ is the sample correlation matrix.

Note: $S_\tau(\hat{\mathbf{R}}) = \left(s_\tau(\hat{\rho}_{ij})\mathbf{1}(i \neq j) + \mathbf{1}(i = j)\right)_{p\times p}$.

Precision matrix estimator

- For sufficiently large constant $M > 0$, if $\lambda = M\tau'$ and $\tau' = o(1/\sqrt{1+s_p})$, then (without assuming irrepresentability)

$$\|\hat{\mathbf{\Omega}}_\lambda - \mathbf{\Omega}\|_2 = O_P\left(\tau'\sqrt{1+s_p}\right),$$

$$\frac{1}{\sqrt{p}}\|\hat{\mathbf{\Omega}}_\lambda - \mathbf{\Omega}\|_F = O_P\left(\tau'\sqrt{1+s_p/p}\right).$$

Gap-block cross-validation

1. Split the data $\mathbf{X}_{p\times n}$ into $H_1 \geq 4$ (almost) equal-size nonoverlapping blocks $\mathbf{X}_i^*, i = 1, ..., H_1$ such that $\mathbf{X}_{p\times n} = (\mathbf{X}_1^*, \mathbf{X}_2^*, ..., \mathbf{X}_{H_1}^*)$. For $i = 1, ..., H_1$, block $\mathbf{X}_i^*$ is used as the validation data, and the remaining data excluding $(\mathbf{X}_{i-1}^*, \mathbf{X}_{i+1}^*)$, denoted by $\mathbf{X}_i^{**}$, are the training data.

2. Randomly subsample H_2 blocks $\mathbf{X}_{H_1+1}^*, ..., \mathbf{X}_{H_1+H_2}^*$ from $\mathbf{X}_{p\times n}$, where $\mathbf{X}_{H_1+j}^*$ consists of $\lceil n/H_1 \rceil$ consecutive columns of $\mathbf{X}_{p\times n}$ for each $j = 1, ..., H_2$. Note that these subsampled blocks can be overlapping. For $i = H_1 + 1, ..., H_1 + H_2$, block $\mathbf{X}_i^*$ is used as the validation data, and the remaining data excluding the $\lceil n/H_1 \rceil$ columns on either side of $\mathbf{X}_i^*$, denoted by $\mathbf{X}_i^{**}$, are the training data.

Gap-block cross-validation (cont'd)

3. Set $H = H_1 + H_2$.
   - For covariance matrix estimation, select optimal τ from $\{\tau_j : j = 1, \ldots, J\}$ by
     $$\tau_S^{\Sigma} = \arg\min_{1 \le j \le J} \frac{1}{H} \sum_{i=1}^{H} \|S_{\tau_j}(\hat{\boldsymbol{\Sigma}}_i^{**}) - \hat{\boldsymbol{\Sigma}}_i^{*}\|_F^2 .$$
   - For the estimation of precision matrix, we choose the optimal tuning parameter using the loss function
     $$\mathrm{tr}(\hat{\boldsymbol{\Omega}}_{\lambda}^{**}\hat{\boldsymbol{\Sigma}}^{*}) - \log\det(\hat{\boldsymbol{\Omega}}_{\lambda}^{**}).$$

A serious issue

The cross-validation is infeasible for large p!

Choosing tuning parameter via submatrices

- Based on the forms of tuning parameter sizes, we propose a method based on estimations of submatrices
- Denote the candidate values of tuning parameter η ($\eta = \tau$ for covariance matrix estimation and $\eta = \lambda$ for precision matrix estimation) to be $\eta_1 < ... < \eta_m$, and the submatrix dimension to be $p_s \times p_s$. In practice, we could choose appropriate p_s such that p/p_s is (almost) an integer. Assume p/p_s is an integer for simplicity. The submatrix approach is implemented as follows:

Step 1: Randomly partition the p random variables into p/p_s groups. For the i-th group, choose the optimal η from $\{\eta_j\}_{j=1}^m$ using the gap-block CV for estimating the i-th sub-covariance-matrix or sub-precision-matrix, and denote it as $\eta_i^{(1)}$.

Step 2: Randomly select p_s random variables p/p_s times without replacement from the total of p random variables. For the i-th sample, choose the optimal η from $\{\eta_j\}_{j=1}^m$ using the gap-block CV as in step 1, and denote it as $\eta_i^{(2)}$.

Step 3: Let $\bar{\eta}$ be the average of $\{\eta_i^{(1)}, \eta_i^{(2)} : i = 1, \ldots, p/p_s\}$. Then scale $\bar{\eta}$ by

$$\bar{\eta}\sqrt{\frac{\log p}{\log p_s}}$$

and use it as the tuning parameter for the original $p \times p$ matrix estimation.

Simulations

- We generate Gaussian data with zero mean and covariance matrix $\mathbf{\Sigma}$ or precision matrix $\mathbf{\Omega}$ from one of the following four models:
  - *Model 1*: $\sigma_{ij} = 0.6^{|i-j|}$;
  - *Model 2*: $\sigma_{ii} = 1, \sigma_{i,i+1} = \sigma_{i+1,i} = 0.6, \sigma_{i,i+2} = \sigma_{i+2,i} = 0.3$, and $\sigma_{ij} = 0$ for $|i-j| \geq 3$;
  - *Model 3*: $\omega_{ij} = 0.6^{|i-j|}$;
  - *Model 4*: $\omega_{ii} = 1, \omega_{i,i+1} = \omega_{i+1,i} = 0.6, \omega_{i,i+2} = \omega_{i+2,i} = 0.3$, and $\omega_{ij} = 0$ for $|i-j| \geq 3$.

Simulations

- For the temporal dependence, we set $\rho_{kl}^{ij} = \theta_{kl}^{ij}\rho_{kl}$ with

$$\theta_{kl}^{ij} = |i - j + 1|^{-\alpha}, \quad 1 \le i, j \le n,$$

so that $|\rho_{kl}^{ij}| \le |\theta_{kl}^{ij}| \sim |i - j|^{-\alpha}$.

- $p = 600$, $n \in \{300, 900\}$, $\alpha \in \{0.25, 0.5, 1, \infty\}$.
- $p_s = \in \{20, 100\}$, $H_1 = H_2 = 10$ for temporal dependent data and 10-fold cross validation for i.i.d. data, 50 replications.

Table: Correlation matrix estimation, $n = 300$

		Model 1						
α	p_s	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	Time
		Hard Thresholding			Soft Thresholding			
i.i.d.	20	0.30(0.01)	1.33(0.06)	10.22(0.20)	0.18(0.01)	1.82(0.04)	12.39(0.32)	2.60(0.07)
	100	0.26(0.01)	1.17(0.05)	9.19(0.18)	0.14(0.01)	1.58(0.03)	10.28(0.17)	7.81(1.52)
	600	0.21(0.01)	1.04(0.05)	8.28(0.16)	0.12(0.01)	1.48(0.03)	9.66(0.13)	37.01(7.06)
	600	0	6.17(0.15)	34.61(0.09)				
1	20	0.49(0.01)	1.98(0.08)	15.96(0.20)	0.31(0.01)	2.38(0.04)	17.93(0.47)	5.25(0.61)
	100	0.40(0.01)	1.74(0.03)	14.28(0.24)	0.23(0.01)	2.07(0.04)	14.67(0.22)	11.81(2.22)
	600	0.34(0.01)	1.57(0.06)	12.38(0.43)	0.20(0.01)	1.94(0.04)	13.49(0.20)	50.02(14.5)
	600	0	15.81(0.99)	49.92(0.54)				
0.5	20	0.68(0.01)	2.95(0.03)	24.86(0.32)	0.47(0.02)	2.77(0.03)	22.54(0.39)	5.25(0.56)
	100	0.57(0.01)	2.51(0.08)	19.66(0.52)	0.35(0.01)	2.49(0.03)	18.98(0.22)	12.34(1.73)
	600	0.49(0.01)	2.11(0.09)	16.62(0.35)	0.29(0.01)	2.35(0.04)	17.54(0.33)	49.52(10.8)
	600	0	36.72(2.44)	73.54(1.06)				
0.25	20	0.84(0.02)	3.00(0.01)	25.94(0.01)	0.62(0.02)	2.95(0.02)	25.10(0.27)	5.29(0.47)
	100	0.70(0.02)	2.93(0.03)	24.70(0.38)	0.46(0.01)	2.75(0.03)	22.14(0.30)	12.24(1.41)
	600	0.59(0.02)	2.61(0.11)	21.04(0.84)	0.39(0.01)	2.64(0.07)	20.57(0.30)	49.17(11.3)
	600	0	55.38(4.41)	95.52(2.29)				

Table: Correlation matrix estimation, $n = 900$

		Model 1						
α	p_s	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	Time
		Hard Thresholding			Soft Thresholding			
i.i.d.	20	0.17(0.01)	0.79(0.04)	6.03(0.11)	0.10(0.01)	1.25(0.03)	7.73(0.20)	3.49(0.40)
	100	0.15(0.01)	0.71(0.04)	5.46(0.11)	0.07(0.01)	1.08(0.03)	6.47(0.11)	8.15(1.09)
	600	0.12(0.01)	0.64(0.03)	4.93(0.08)	0.06(0.01)	1.01(0.03)	6.09(0.09)	39.48(12.7)
	600	0	3.13(0.09)	19.96(0.07)				
1	20	0.30(0.01)	1.29(0.07)	10.06(0.25)	0.17(0.01)	1.76(0.04)	11.93(0.36)	7.13(10.30)
	100	0.24(0.01)	1.09(0.04)	8.73(0.13)	0.13(0.01)	1.52(0.03)	9.75(0.13)	14.49(2.09)
	600	0.20(0.01)	0.99(0.04)	7.64(0.13)	0.11(0.01)	1.40(0.04)	8.88(0.22)	55.46(11.9)
	600	0	7.33(0.38)	29.62(0.20)				
0.5	20	0.48(0.01)	1.94(0.07)	15.86(0.21)	0.31(0.01)	2.36(0.05)	17.73(0.52)	5.95(0.82)
	100	0.40(0.01)	1.74(0.05)	14.20(0.28)	0.23(0.01)	2.06(0.03)	14.52(0.23)	14.22(2.05)
	600	0.34(0.01)	1.58(0.06)	12.44(0.45)	0.19(0.01)	1.92(0.04)	13.37(0.29)	57.17(10.9)
	600	0	22.47(1.44)	49.71(0.79)				
0.25	20	0.67(0.02)	2.93(0.05)	24.50(0.57)	0.46(0.02)	2.75(0.04)	22.16(0.60)	5.61(0.37)
	100	0.56(0.02)	2.48(0.11)	19.46(0.85)	0.33(0.01)	2.46(0.03)	18.63(0.36)	13.61(1.69)
	600	0.48(0.02)	2.19(0.29)	16.64(0.45)	0.29(0.01)	2.64(0.35)	17.44(0.36)	57.25(8.01)
	600	0	44.06(3.74)	71.93(2.20)				

Table: Correlation matrix estimation, $n = 300$

		Model 2						
α	p_s	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	Time
		Hard Thresholding			Soft Thresholding			
i.i.d.	20	0.30(0.01)	0.60(0.02)	7.27(0.27)	0.20(0.01)	0.93(0.03)	10.00(0.35)	2.85(0.51)
	100	0.26(0.01)	0.55(0.03)	5.50(0.32)	0.15(0.01)	0.77(0.03)	7.80(0.15)	7.14(0.71)
	600	0.23(0.01)	0.51(0.04)	3.83(0.18)	0.13(0.01)	0.74(0.02)	7.18(0.10)	38.46(7.72)
	600	0	5.84(0.13)	34.64(0.09)				
1	20	0.49(0.01)	1.07(0.11)	11.21(0.28)	0.33(0.01)	1.33(0.03)	15.36(0.38)	5.34(0.62)
	100	0.43(0.01)	0.84(0.09)	10.44(0.11)	0.24(0.01)	1.10(0.02)	12.02(0.18)	12.31(1.81)
	600	0.35(0.01)	0.72(0.07)	9.44(0.24)	0.20(0.01)	1.02(0.02)	10.71(0.25)	52.43(13.5)
	600	0	15.56(0.92)	49.90(0.51)				
0.5	20	0.69(0.01)	1.79(0.01)	22.18(0.32)	0.50(0.02)	1.67(0.02)	20.19(0.48)	5.40(0.69)
	100	0.58(0.01)	1.59(0.05)	16.44(0.58)	0.36(0.01)	1.43(0.03)	16.21(0.22)	11.76(1.96)
	600	0.49(0.01)	1.30(0.08)	12.58(0.37)	0.31(0.01)	1.42(0.14)	14.79(0.22)	52.72(13.7)
	600	0	36.51(2.47)	73.47(1.06)				
0.25	20	0.84(0.02)	1.80(0.01)	23.21(0.01)	0.64(0.02)	1.78(0.01)	22.55(0.24)	5.30(0.56)
	100	0.70(0.02)	1.78(0.01)	21.86(0.40)	0.47(0.01)	1.64(0.03)	19.33(0.29)	11.99(1.94)
	600	0.59(0.02)	1.78(0.22)	17.81(1.01)	0.39(0.01)	1.77(0.37)	17.68(0.30)	53.49(14.0)
	600	0	55.37(4.32)	95.53(2.18)				

Table: Correlation matrix estimation, $n = 900$

		Model 2						
α	p_s	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	τ	$\|\cdot\|_2$	$\|\cdot\|_F$	Time
		Hard Thresholding			Soft Thresholding			
i.i.d.	20	0.17(0.01)	0.15(0.02)	1.28(0.03)	0.11(0.01)	0.54(0.02)	5.79(0.17)	3.32(0.38)
	100	0.18(0.01)	0.16(0.03)	1.28(0.04)	0.09(0.01)	0.44(0.02)	4.51(0.07)	8.69(1.63)
	600	0.18(0.01)	0.15(0.02)	1.28(0.03)	0.07(0.01)	0.43(0.01)	4.15(0.05)	37.36(6.41)
	600	0	2.93(0.06)	19.97(0.07)				
1	20	0.30(0.01)	0.59(0.01)	7.59(0.35)	0.19(0.01)	0.89(0.03)	9.77(0.30)	6.05(0.76)
	100	0.26(0.01)	0.51(0.03)	4.63(0.26)	0.14(0.02)	0.70(0.02)	7.36(0.10)	19.70(4.89)
	600	0.22(0.01)	0.41(0.04)	2.82(0.25)	0.12(0.01)	0.63(0.02)	6.48(0.15)	87.41(38.1)
	600	0	7.24(0.38)	29.65(0.19)				
0.5	20	0.49(0.01)	1.04(0.12)	11.11(0.29)	0.33(0.01)	1.32(0.04)	15.18(0.50)	6.05(0.68)
	100	0.43(0.02)	0.83(0.11)	10.45(0.14)	0.24(0.01)	1.09(0.03)	11.96(0.23)	22.90(6.20)
	600	0.36(0.01)	0.78(0.11)	9.53(0.30)	0.20(0.01)	1.06(0.10)	10.66(0.23)	85.58(37.3)
	600	0	22.44(1.48)	49.74(0.81)				
0.25	20	0.68(0.02)	1.78(0.01)	21.78(0.62)	0.48(0.02)	1.64(0.04)	19.70(0.62)	6.28(0.13)
	100	0.57(0.01)	1.59(0.07)	16.18(0.88)	0.35(0.01)	1.42(0.07)	15.86(0.32)	22.13(4.88)
	600	0.49(0.01)	1.49(0.28)	12.53(0.47)	0.29(0.01)	2.23(0.46)	14.59(0.37)	84.02(30.5)
	600	0	44.11(3.38)	71.95(2.00)				

Table: Precision matrix estimation

			Model 3				Model 4			
n	α	p_s	λ	$\Vert\cdot\Vert_2$	$\Vert\cdot\Vert_F$	Time	λ	$\Vert\cdot\Vert_2$	$\Vert\cdot\Vert_F$	Time
			Correlation-based GLasso				Correlation-based GLasso			
300	i.i.d.	20	0.11	3.19	24.96	16.15	0.10	2.16	22.94	18.91
		100	0.11	3.21	25.20	340.90	0.09	2.13	22.62	603.70
		600	0.09	3.12	24.21	4.9e4	0.07	1.91	19.87	5.6e4
	1	20	0.13	3.22	25.27	55.84	0.12	2.24	23.80	48.59
		100	0.15	3.28	25.96	1226	0.12	2.26	24.04	1724
		600	0.11	3.15	24.55	1.6e5	0.08	2.01	20.63	1.9e5
	0.5	20	0.13	3.07	23.82	59.47	0.12	2.17	22.29	61.41
		100	0.14	3.11	24.22	1750	0.12	2.17	22.32	2268
		600	0.11	2.96	22.96	2.4e5	0.08	1.91	18.81	2.1e5
	0.25	20	0.13	2.68	22.87	73.61	0.12	1.96	19.54	71.51
		100	0.14	2.75	22.92	2268	0.12	1.98	19.72	3418
		600	0.11	2.57	2.29	2.2e5	0.08	1.68	17.75	3.6e5

Table: Precision matrix estimation

n	α	p_s	Model 3 λ	Model 3 $\|\cdot\|_2$	Model 3 $\|\cdot\|_F$	Model 3 Time	Model 4 λ	Model 4 $\|\cdot\|_2$	Model 4 $\|\cdot\|_F$	Model 4 Time
			Correlation-based GLasso				Correlation-based GLasso			
900	i.i.d.	20	0.06	2.93	22.31	23.87	0.06	1.80	18.96	24.14
		100	0.06	2.93	22.33	273.60	0.05	1.73	18.16	747.00
		600	0.04	2.73	20.28	5.6e4	0.03	1.39	14.16	7.8e4
		600	$\widehat{\Sigma}^{-1}$	33.00	127.50		$\widehat{\Sigma}^{-1}$	31.73	125.30	
	1	20	0.09	3.09	23.93	37.68	0.08	2.03	21.43	53.89
		100	0.10	3.15	24.54	724.10	0.08	2.03	21.44	1093
		600	0.07	2.94	22.33	1.8e5	0.05	1.65	16.83	2.2e5
		600	$\widehat{\Sigma}^{-1}$	56.31	221.10		$\widehat{\Sigma}^{-1}$	53.50	216.80	
	0.5	20	0.11	3.09	23.91	48.05	0.10	2.12	21.96	60.15
		100	0.12	3.15	24.54	1072	0.10	2.11	21.80	1500
		600	0.08	2.90	22.07	2.3e5	0.05	1.68	16.35	2.9e5
		600	$\widehat{\Sigma}^{-1}$	115.00	463.00		$\widehat{\Sigma}^{-1}$	109.10	452.50	
	0.25	20	0.10	2.78	21.42	62.81	0.10	1.93	18.73	73.98
		100	0.11	2.82	21.67	1477	0.09	1.90	18.37	1923
		600	0.08	2.62	20.50	2.6e5	0.06	1.49	14.13	3.5e5
		600	$\widehat{\Sigma}^{-1}$	246.30	989.50		$\widehat{\Sigma}^{-1}$	231.90	965.60	

Largest hub in the correlation matrix of rs-fMRI data, connected with 3465 voxels

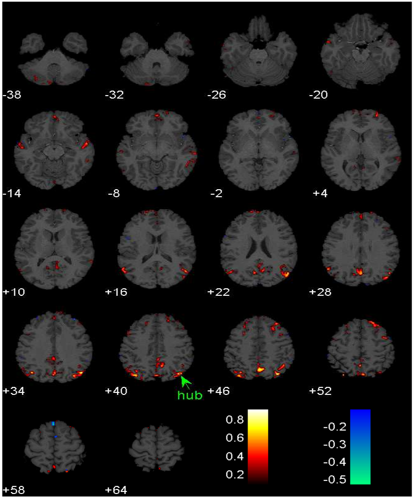
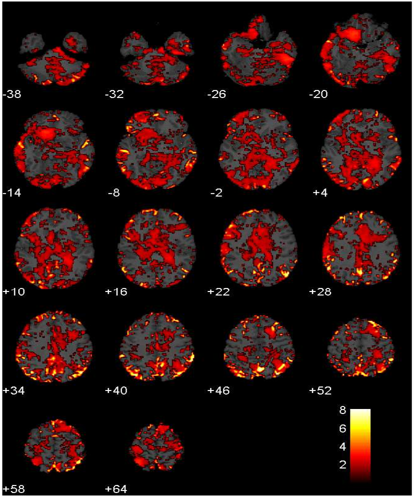


Table: Connectivity of the largest hub in right inferior parietal cortex

Region	# connected voxels	% in the region
Right inferior parietal cortex	444	16.61%
Left inferior parietal cortex	267	12.16%
Right precuneus cortex	189	10.59%
Left precuneus cortex	162	9.70%
Right middle temporal cortex	127	5.32%
Right rostral middle frontal cortex	82	2.59%
Right superior frontal cortex	75	1.89%
Left Cerebellum Cortex	68	0.67%
Left middle temporal cortex	67	3.53%
Left superior frontal cortex	66	1.68%

Numbers of connections: rs-fMRI data



Precision matrix for rs-fMRI data

- It was claimed that the BigQuick algorithm via parallel computing (Hsieh et al 2013, NIPS) can handle 1 million voxels for estimating $\boldsymbol{\Omega}$
- We are having some issues in running BigQuick package...

THANK YOU!