Supplementary Material to: The generalized ridge estimator of the inverse covariance matrix

Wessel N. van Wieringen
Department of Epidemiology and Biostatistics,
Amsterdam School of Public Health, Amsterdam UMC,
P.O. Box 7057, 1007 MB Amsterdam, The Netherlands
&
Department of Mathematics, VU University Amsterdam
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
w.vanwieringen@vumc.nl

February 20, 2019

Abstract

This supplementary material (SM) contains the results of the benchmarking of the proposed algorithms and additional figures corresponding to the application. **AANPASSEN!!!**

Alternative algorithm for the generalized graphical lasso

Different algorithms may be conceived to find the generalized graphical lasso estimator. As suggested by a reviewer we explore the line of thought that the generalized graphical lasso estimator may be found by a modification of the block-wise approach of the graphical lasso algorithm. The generalized graphical lasso estimator maximizes:

$$\log(|\boldsymbol{\Omega}|) - \text{tr}(\mathbf{S}\boldsymbol{\Omega}) - \|\boldsymbol{\Lambda} \circ (\boldsymbol{\Omega} - \mathbf{T})\|_1, \tag{1}$$

where the $\circ$-operator is the Hadamard product. The corresponding estimating equation, in analogy to Friedman *et al.* (2008), is:

$$\boldsymbol{\Omega}^{-1} - \mathbf{S} - \boldsymbol{\Lambda} \circ \text{sign}(\boldsymbol{\Omega} - \mathbf{T}) \;\; = \;\; \mathbf{0}_{pp},$$

where the sign-function is to be applied element-wise. To find its root(s) write the involved matrices as 2×2-block matrices:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\sigma}_{12} \\ \boldsymbol{\sigma}_{12}^\top & \sigma_{22} \end{pmatrix} = \boldsymbol{\Omega}^{-1} = \begin{pmatrix} \boldsymbol{\Omega}_{11} & \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}_{12}^\top & \omega_{22} \end{pmatrix}^{-1} = \begin{pmatrix} * & -\boldsymbol{\Sigma}_{11}\boldsymbol{\omega}_{12}\omega_{22}^{-1} \\ * & * \end{pmatrix}, \tag{2}$$

and similar partitioned apply to $\mathbf{S}$ and $\mathbf{T}$. Then, in line with the graphical lasso algorithm we solve for $\boldsymbol{\Sigma}$ (from which $\boldsymbol{\Omega}$ is eventually obtained). When temporarily assuming the matrix $\boldsymbol{\Sigma}_{11}$ known, the estimating equations for the remaining blocks are:

$$\sigma_{22} - s_{22} - \lambda_{22}\text{sign}(\omega_{22} - t_{22}) \;\; = \;\; 0, \tag{3}$$
$$\boldsymbol{\Sigma}_{12} - \mathbf{S}_{12} - \boldsymbol{\Lambda}_{12} \circ \text{sign}(\boldsymbol{\omega}_{12} - \mathbf{t}_{12}) \;\; = \;\; \mathbf{0}_{p-1}. \tag{4}$$

Having found the solutions of these equations, the current estimate of $\boldsymbol{\Sigma}$ is updated, and the another row/column is selected for updating. This process runs over all rows/columns and and is then repeated until convergence.

Estimating equations (3) and (4) may be solved as follows. First note that, effectively, an estimate of $\boldsymbol{\Sigma}$ is obtained, which is then inverted to yield one of $\boldsymbol{\Omega}$. We first concentrat on estimating equation (4). Still pursuing the analogy with Friedman *et al.* (2008), this equation is rewritten to

$$\boldsymbol{\Sigma}_{11}\boldsymbol{\omega}_{12}\omega_{22}^{-1} + \mathbf{S}_{12} + \boldsymbol{\Lambda}_{12} \circ \text{sign}(\boldsymbol{\omega}_{12} - \mathbf{t}_{12}) \;\; = \;\; \mathbf{0}_{p-1},$$

where we have used that $\boldsymbol{\Sigma}_{12} = -\boldsymbol{\Sigma}_{11}\boldsymbol{\omega}_{12}\omega_{22}^{-1}$ which is obtained from (2). To solve this apply the change-of-variable $\boldsymbol{\beta} = (\boldsymbol{\omega}_{12} - t_{12})\omega_{22}^{-1}$, which yields:

$$\boldsymbol{\Sigma}_{11}\boldsymbol{\beta} + \boldsymbol{\Sigma}_{11}t_{12}\omega_{22}^{-1} + \mathbf{S}_{12} + \boldsymbol{\Lambda}_{12} \circ \text{sign}(\boldsymbol{\beta}) \;\; = \;\; \mathbf{0}_{p-1}, \tag{5}$$

where we have used the fact that the sign of ω_{22}^{-1} is positive. In this one recognizes a lasso regression estimating equation of the following lasso problem:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{p-1}} \frac{1}{2}\|\boldsymbol{\Sigma}_{11}^{1/2}\boldsymbol{\beta} + \boldsymbol{\Sigma}_{11}^{1/2}t_{12}\omega_{22}^{-1} + \boldsymbol{\Sigma}_{11}^{-1/2}\mathbf{S}_{12}\|_2^2 + \|\boldsymbol{\Lambda}_{12} \circ \boldsymbol{\beta}\|_1.$$

This may be solved by, e.g., coordinate-wise ascent (as used in the `glmnet`-package).

To obtain a solution of estimating equation (2), however, requires at each update knowledge of ω_{22}. In the regular graphical lasso only knowledge of its sign is needed, which is provided by the fact that it is a diagonal element of the precision matrix and therefore positive, which simplifies – for the regular graphical lasso – its estimating equation (3) and yields $\sigma_{22} = s_{22} + \lambda_{22}$. In case of the generalized graphical lasso estimator knowledge of the sign of $\omega_{22} - t_{22}$ is required. This is not *a priori* known, and – in principle – all three values, $\{-1, 0, 1\}$, need to be considered. In certain cases, the solution corresponding to the negative sign can be discarded if it is negative. Irrespectively, at least two solutions need to be considered, and for each the lasso regression problem needs to be solved. At this stage a choice between the found solutions needs to be made. This can be done by choosing that that combination of $(\boldsymbol{\sigma}_{12}, \sigma_{22})$ updates that yields the largest penalized likelihood (1) (evaluated through substitution of the possible solutions). However, in this $\boldsymbol{\sigma}_{12}$ still depends on the actual value of ω_{22} (or its inverse), which is still unknown. An obvious way around this is to perform a line search:

- choose a value for ω_{22},
- find $\boldsymbol{\beta}$ through solving estimating equation (5),
- put the above together to obtain $\boldsymbol{\Sigma}$,
- use its inverse of $\boldsymbol{\Sigma}$ to evaluate the penalized loglikelihood (1),
- repeat this for a grid of ω_{22},
- choose that that combination of ω_{22} and $\boldsymbol{\beta}$ that maximizes the penalized loglikelihood (1).

The above line search needs to be done at each row/column update. Then, run over the rows/columns until convergence.

Benchmarking: precision matrices and results

The following precision matrices have been employed in benchmarking:

Banded: The precision matrix is fully parametrized by $(\mathbf{\Omega})_{j,j} = 1$ for $j = 1, \ldots, p$, $(\mathbf{\Omega})_{j,j+1} = 0.5 = (\mathbf{\Omega})_{j+1,j}$ for $j = 1, \ldots, p-1$, $(\mathbf{\Omega})_{j,j+2} = 0.2 = (\mathbf{\Omega})_{j+2,j}$ for $j = 1, \ldots, p-2$, $(\mathbf{\Omega})_{j,j+3} = 0.1 = (\mathbf{\Omega})_{j+3,j}$ for $j = 1, \ldots, p-3$, and zero otherwise.

Full: The precision matrix is fully parametrized by $(\mathbf{\Omega})_{j,j} = 1$ for $j = 1, \ldots, p$ and 0.5 otherwise.

Blocked: The precision matrix contains five equally sized square block matrices along the diagonal. Each block is fully parametrized by as the full precision matrix above. Elements outside the blocks are all zero.

Hub: The precision matrix is fully parametrized by $(\mathbf{\Omega})_{j,j} = 1$ for $j = 1, \ldots, p$, $(\mathbf{\Omega})_{1,j} = 1/j = (\mathbf{\Omega})_{j,1}$ for $j = 2, \ldots, p$, and zero otherwise.

Table 1: Accuracy ($\times 10^{-9}$) of the proposed algorithm for a banded precision matrix with various choices of n (1^{st} column) and p (2^{nd} column). The 3^{rd} and 4^{th} columns contain the Frobenius and supremum loss, respectively, of the two ridge precision estimator algorithms. The 5^{th} and 6^{th} columns contain the Frobenius and supremum loss, respectively, of the two algorithms for the ridge precision estimator with known chordal support. The 7^{th} and 8^{th} columns contain the Frobenius and supremum loss, respectively, of the two lasso precision estimator algorithms. All accuracies are to be multiplied by 10^{-9}.

		ridge vs. gen. ridge		lasso vs. gen. ridge		chordal ridge vs. gen. ridge	
		Frobenius	supremum	Frobenius	supremum	Frobenius	supremum
n	p	$\|\widehat{\boldsymbol{\Omega}}_{\text{r}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_F$	$\|\widehat{\boldsymbol{\Omega}}_{\text{r}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_\infty$	$\|\widehat{\boldsymbol{\Omega}}_{\text{rc}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_F$	$\|\widehat{\boldsymbol{\Omega}}_{\text{r}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_\infty$	$\|\widehat{\boldsymbol{\Omega}}_{\text{gl}} - \widehat{\boldsymbol{\Omega}}_{\text{ggl}}\|_F$	$\|\widehat{\boldsymbol{\Omega}}_{\text{gl}} - \widehat{\boldsymbol{\Omega}}_{\text{ggl}}\|_\infty$
10	10	0.71432	0.37363	3960.40	1360.79	232722.4	159294.6
10	50	1.56462	0.43841	32618.23	7080.29	3668748.0	1214063.3
10	100	2.09675	0.45846	77320.61	14585.24	7123890.3	1522635.9
50	10	0.68086	0.37770	2533.83	926.91	34679.0	9106.1
50	50	1.57712	0.43232	19718.57	4171.40	968972.8	572887.1
50	100	2.26400	0.45245	39612.16	7195.54	1489323.8	803768.7
100	10	0.71450	0.37317	2297.79	811.15	31500.9	8736.7
100	50	1.57185	0.43897	13012.74	3387.29	600332.5	385868.6
100	100	2.26802	0.45617	28385.48	5649.60	1072888.2	673200.7

Table 2: Accuracy ($\times 10^{-9}$) of the proposed algorithm for a uniform precision matrix with various choices of n (1^{st} column) and p (2^{nd} column). The 3^{rd} and 4^{th} columns contain the Frobenius and supremum loss, respectively, of the two ridge precision estimator algorithms. The 5^{th} and 6^{th} columns contain the Frobenius and supremum loss, respectively, of the two algorithms for the ridge precision estimator with known chordal support. The 7^{th} and 8^{th} columns contain the Frobenius and supremum loss, respectively, of the two lasso precision estimator algorithms. All accuracies are to be multiplied by 10^{-9}.

		ridge vs. gen. ridge		lasso vs. gen. ridge		chordal ridge vs. gen. ridge	
		Frobenius	supremum	Frobenius	supremum	Frobenius	supremum
n	p	$\|\widehat{\boldsymbol{\Omega}}_{\text{r}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_F$	$\|\widehat{\boldsymbol{\Omega}}_{\text{r}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_\infty$	$\|\widehat{\boldsymbol{\Omega}}_{\text{rc}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_F$	$\|\widehat{\boldsymbol{\Omega}}_{\text{r}} - \widehat{\boldsymbol{\Omega}}_{\text{gr}}\|_\infty$	$\|\widehat{\boldsymbol{\Omega}}_{\text{gl}} - \widehat{\boldsymbol{\Omega}}_{\text{ggl}}\|_F$	$\|\widehat{\boldsymbol{\Omega}}_{\text{gl}} - \widehat{\boldsymbol{\Omega}}_{\text{ggl}}\|_\infty$
10	10	0.71358	0.36602	not applicable	not applicable	219646.4	134245.1
10	50	1.57030	0.43565	not applicable	not applicable	3813712.7	1183803.2
10	100	2.10661	0.45351	not applicable	not applicable	7577082.3	1460170.7
50	10	0.70095	0.36809	not applicable	not applicable	29584.1	7588.9
50	50	1.58580	0.42757	not applicable	not applicable	147183.9	30556.2
50	100	2.31228	0.45686	not applicable	not applicable	844113.3	469993.0
100	10	0.70340	0.36254	not applicable	not applicable	24075.3	5789.5
100	50	1.54659	0.43197	not applicable	not applicable	98387.7	6905.9
100	100	2.28135	0.44783	not applicable	not applicable	197518.9	7855.1

Table 3: Accuracy ($\times 10^{-9}$) of the proposed algorithm for a block precision matrix with various choices of n (1st column) and p (2nd column). The 3rd and 4th columns contain the Frobenius and supremum loss, respectively, of the two ridge precision estimator algorithms. The 5th and 6th columns contain the Frobenius and supremum loss, respectively, of the two algorithms for the ridge precision estimator with known chordal support. The 7th and 8th columns contain the Frobenius and supremum loss, respectively, of the two lasso precision estimator algorithms. All accuracies are to be multiplied by 10^{-9}.

		ridge vs. gen. ridge		lasso vs. gen. ridge		chordal ridge vs. gen. ridge	
		Frobenius	supremum	Frobenius	supremum	Frobenius	supremum
n	p	$\|\widehat{\mathbf{\Omega}}_{\mathrm{r}} - \widehat{\mathbf{\Omega}}_{\mathrm{gr}}\|_F$	$\|\widehat{\mathbf{\Omega}}_{\mathrm{r}} - \widehat{\mathbf{\Omega}}_{\mathrm{gr}}\|_\infty$	$\|\widehat{\mathbf{\Omega}}_{\mathrm{rc}} - \widehat{\mathbf{\Omega}}_{\mathrm{gr}}\|_F$	$\|\widehat{\mathbf{\Omega}}_{\mathrm{r}} - \widehat{\mathbf{\Omega}}_{\mathrm{gr}}\|_\infty$	$\|\widehat{\mathbf{\Omega}}_{\mathrm{gl}} - \widehat{\mathbf{\Omega}}_{\mathrm{ggl}}\|_F$	$\|\widehat{\mathbf{\Omega}}_{\mathrm{gl}} - \widehat{\mathbf{\Omega}}_{\mathrm{ggl}}\|_\infty$
10	10	0.70059	0.37263	0.76549	0.36282	93429.1	59531.5
10	50	1.54269	0.43545	2.93732	0.44105	3798381.6	1274041.7
10	100	2.11993	0.45915	5.56982	0.44844	7420574.8	1480242.8
50	10	0.67681	0.35508	0.69580	0.36144	26581.1	8574.4
50	50	1.60251	0.43182	1.93964	0.43645	132193.4	9287.4
50	100	2.27878	0.4477	3.26871	0.45340	851590.3	493502.0
100	10	0.68343	0.36419	0.68659	0.35872	24339.2	8303.4
100	50	1.57218	0.43831	1.73714	0.41516	96955.2	7061.9
100	100	2.26573	0.45390	2.79204	0.43924	192824.2	7631.6

Table 4: Accuracy ($\times 10^{-9}$) of the proposed algorithm for a star/hub precision matrix with various choices of n (1^{st} column) and p (2^{nd} column). The 3^{rd} and 4^{th} columns contain the Frobenius and supremum loss, respectively, of the two ridge precision estimator algorithms. The 5^{th} and 6^{th} columns contain the Frobenius and supremum loss, respectively, of the two algorithms for the ridge precision estimator with known chordal support. The 7^{th} and 8^{th} columns contain the Frobenius and supremum loss, respectively, of the two lasso precision estimator algorithms. All accuracies are to be multiplied by 10^{-9}.

		ridge vs. gen. ridge		lasso vs. gen. ridge		chordal ridge vs. gen. ridge	
		Frobenius	supremum	Frobenius	supremum	Frobenius	supremum
n	p	$\|\widehat{\mathbf{\Omega}}_{\text{r}} - \widehat{\mathbf{\Omega}}_{\text{gr}}\|_F$	$\|\widehat{\mathbf{\Omega}}_{\text{r}} - \widehat{\mathbf{\Omega}}_{\text{gr}}\|_\infty$	$\|\widehat{\mathbf{\Omega}}_{\text{rc}} - \widehat{\mathbf{\Omega}}_{\text{gr}}\|_F$	$\|\widehat{\mathbf{\Omega}}_{\text{r}} - \widehat{\mathbf{\Omega}}_{\text{gr}}\|_\infty$	$\|\widehat{\mathbf{\Omega}}_{\text{gl}} - \widehat{\mathbf{\Omega}}_{\text{ggl}}\|_F$	$\|\widehat{\mathbf{\Omega}}_{\text{gl}} - \widehat{\mathbf{\Omega}}_{\text{ggl}}\|_\infty$
10	10	0.69747	0.37136	2392.67	1097.68	52026.5	29765.9
10	50	1.55732	0.43747	NA	NA	1643625.1	999534.2
10	100	2.13252	0.45642	NA	NA	3407280	1446042.5
50	10	0.68110	0.37198	2227.25	961.41	27541.2	8819.8
50	50	1.53463	0.44135	11783.35	4597.35	80344.1	9115.3
50	100	2.25040	0.45333	19994.67	7153.77	148065.8	8997.8
100	10	0.67821	0.37341	2531.88	1067.16	25324.4	8675.8
100	50	1.55951	0.43940	11294.70	4387.91	59610.5	8637.4
100	100	2.22942	0.45090	22083.76	8933.61	106685.3	9048.2

$\boldsymbol{\Lambda}$ as $n \to \infty$

Proposition 3 in the main text hinges upon the assumption that the minimum over the elements of the penalty matrix tends to zero as the sample size increases. While the validity of this assumption is assessed in the Supplementary Material of **?**) for the regular ridge precision estimator with only a single penalty parameter, it is here assessed for the generalized ridge precision estimator with a penalty matrix parameterized by either two or three parameters. This is done in simulation. Data are drawn from a zero-mean multivariate normal with one the following precision matrices:

2×2 *block*: A 2×2 block matrix $\boldsymbol{\Omega}$ with each block a 25×25 dimensional matrix. The left-upper block, denoted by $\boldsymbol{\Omega}_{11}$, is banded and fully parametrized by $(\boldsymbol{\Omega}_{11})_{j,j} = 1$ for $j = 1, \ldots, 25$, $(\boldsymbol{\Omega})_{j,j+1} = 0.5 = (\boldsymbol{\Omega})_{j+1,j}$ for $j = 1, \ldots, 24$, $(\boldsymbol{\Omega}_{11})_{j,j+2} = 0.2 = (\boldsymbol{\Omega}_{11})_{j+2,j}$ for $j = 1, \ldots, 23$, $(\boldsymbol{\Omega})_{j,j+3} = 0.1 = (\boldsymbol{\Omega})_{j+3,j}$ for $j = 1, \ldots, 22$, and zero otherwise. The elements of the off-diagonal blocks, denoted $\boldsymbol{\Omega}_{12}$ and $\boldsymbol{\Omega}_{21}$, all equal 0.1. The right-lower block, denoted $\boldsymbol{\Omega}_{22}$, is parametrized by $(\boldsymbol{\Omega}_{11})_{j,j} = 1$ for $j = 1, \ldots, 25$ and $(\boldsymbol{\Omega}_{11})_{j,j} = 0.1$ otherwise.

3×3 *block*: A 3×3 block matrix $\boldsymbol{\Omega}$ with diagonal blocks of dimensions 20×20, 20×20, and 10×10, (in that order). The first diagonal block, denoted by $\boldsymbol{\Omega}_{11}$, is banded and fully parametrized by $(\boldsymbol{\Omega}_{11})_{j,j} = 1$ for $j = 1, \ldots, 20$, $(\boldsymbol{\Omega})_{j,j+1} = 0.5 = (\boldsymbol{\Omega})_{j+1,j}$ for $j = 1, \ldots, 19$, $(\boldsymbol{\Omega}_{11})_{j,j+2} = 0.2 = (\boldsymbol{\Omega}_{11})_{j+2,j}$ for $j = 1, \ldots, 18$, $(\boldsymbol{\Omega})_{j,j+3} = 0.1 = (\boldsymbol{\Omega})_{j+3,j}$ for $j = 1, \ldots, 17$, and zero otherwise. The second diagonal block, denoted $\boldsymbol{\Omega}_{22}$, is parametrized by $(\boldsymbol{\Omega}_{11})_{j,j} = 1$ for $j = 1, \ldots, 25$ and $(\boldsymbol{\Omega}_{11})_{j,j} = 0.1$ otherwise. The third diagonal block, denoted $\boldsymbol{\Omega}_{33}$, is the unit matrix of appropriate dimensions. The elements of the off-diagonal blocks all equal 0.1.

The sample size ranges from $n = 10$ to $n = 100000$. For each data set the optimal penalty matrix $\boldsymbol{\Lambda}$ for the generalized ridge estimation of the precision matrix is determined by means of K-fold cross-validation (with $K = 5$). In this, for both choices of $\boldsymbol{\Omega}$, the target matrix $\mathbf{T}$ is banded and parametrized by $(\mathbf{T})_{j,j} = 1$ for $j = 1, \ldots, 50$, $(\boldsymbol{\Omega})_{j,j+1} = 0.1 = (\mathbf{T})_{j+1,j}$ for $j = 1, \ldots, 49$, and zero otherwise. The penalty matrix $\boldsymbol{\Lambda}$ shares the block structure with the employed precision matrix. All elements of a diagonal block are equal to the same penalty parameter, e.g. in the case of the 2×2 block matrix this gives $\boldsymbol{\Lambda}_{11} = \lambda_1 \mathbf{1}_{25,25}$ and $\boldsymbol{\Lambda}_{22} = \lambda_2 \mathbf{1}_{25,25}$. The elements of the off-diagonal blocks are set equal to the product of the corresponding penalty parameters, e.g. in the case of the 2×2 block matrix $\boldsymbol{\Lambda}_{12} = \lambda_1 \lambda_2 \mathbf{1}_{25,25} = \boldsymbol{\Lambda}_{21}$. The optimal penalty parameters then maximize the cross-validation loglikelihood. This maximum is sought in the interval $[10^{10}, 1000]$ for each penalty parameter and found by means of gradient ascent. The results, the optimal penalty parameters, are plotted against the sample size (Figure 1). The panels of Figure 1 do not invalidate the assumption of Proposition 3 of the main text.

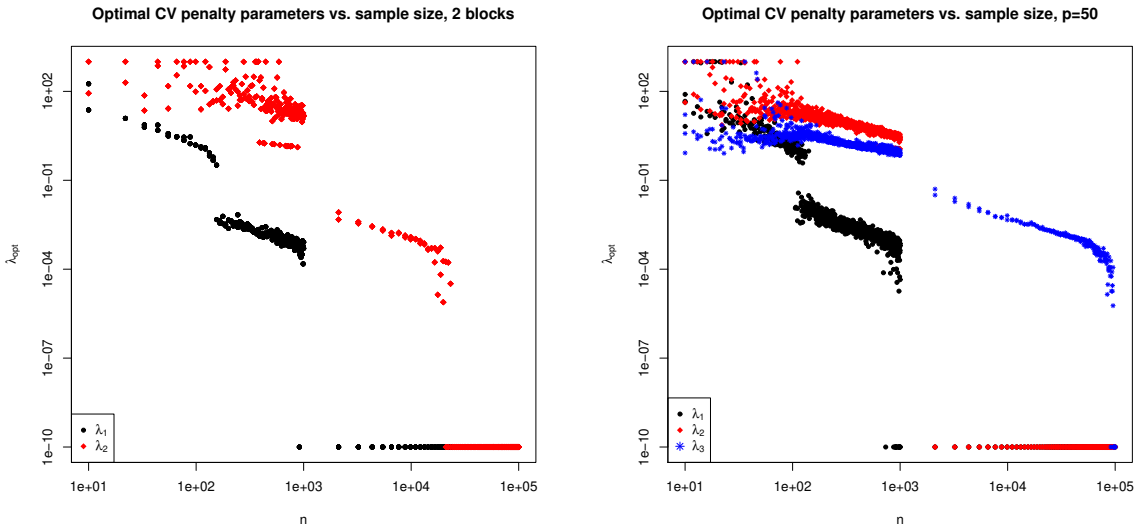

Figure 1: The 5-fold cross-validated penalty parameters (y-axis) plotted against the sample size n (x-axis). The left panel: data sampled from a 2×2 block structured precision matrix $\mathbf{\Omega}$ and similarly structured penalty matrix $\mathbf{\Lambda}$ parametrized by λ_1 and λ_2 in the estimation (see text for details). The right panel: data sampled from a 3×3 block structured precision matrix $\mathbf{\Omega}$ and a similarly structured penalty matrix $\mathbf{\Lambda}$ parametrized by λ_1, λ_2 and λ_3 is used in the estimation (see text for details).

Comparison: regular vs. generalized ridge precision estimation

Matrices employed in Simulation I (banded precision):

- $\boldsymbol{\Omega}$: The precision matrix is banded and fully parametrized by $(\boldsymbol{\Omega})_{j,j} = 1$ for $j = 1, \ldots, p$, $(\boldsymbol{\Omega})_{j,j+1} = 0.5 = (\boldsymbol{\Omega})_{j+1,j}$ for $j = 1, \ldots, p-1$, $(\boldsymbol{\Omega})_{j,j+2} = 0.2 = (\boldsymbol{\Omega})_{j+2,j}$ for $j = 1, \ldots, p-2$, $(\boldsymbol{\Omega})_{j,j+3} = 0.1 = (\boldsymbol{\Omega})_{j+3,j}$ for $j = 1, \ldots, p-3$, and zero otherwise.
- $\mathbf{T}$: The target matrix is banded and fully parametrized by $(\mathbf{T})_{j,j} = 0.5$ for $j = 1, \ldots, p$, $(\mathbf{T})_{j,j+1} = 0.1 = (\mathbf{T})_{j+1,j}$ for $j = 1, \ldots, p-1$, and zero otherwise.
- $\boldsymbol{\Lambda}$: The penalty matrix for the regular ridge precision estimator is of course $\boldsymbol{\Lambda} = \lambda \mathbf{1}_{pp}$, while for the generalized ridge precision estimator is parametrized: $(\boldsymbol{\Lambda})_{j,j} = \lambda$ for $j = 1, \ldots, p$, $(\boldsymbol{\Lambda})_{j,j+1} = 2\lambda = (\boldsymbol{\Lambda})_{j+1,j}$ for $j = 1, \ldots, p-1$, $(\boldsymbol{\Lambda})_{j,j+2} = 3\lambda = (\boldsymbol{\Lambda})_{j+2,j}$ for $j = 1, \ldots, p-2$, $(\boldsymbol{\Lambda})_{j,j+3} = 4\lambda = (\boldsymbol{\Lambda})_{j+3,j}$ for $j = 1, \ldots, p-3$, and so on until $(\boldsymbol{\Lambda})_{1,p} = p\lambda = (\boldsymbol{\Lambda})_{p,1}$.

Matrices employed in Simulation II (partially known precision):

- $\boldsymbol{\Omega}$: The precision matrix is banded and fully parametrized by $(\boldsymbol{\Omega})_{j,j} = 1$ for $j = 1, \ldots, p$, $(\boldsymbol{\Omega})_{j,j+1} = 0.5 = (\boldsymbol{\Omega})_{j+1,j}$ for $j = 1, \ldots, p-1$, $(\boldsymbol{\Omega})_{j,j+2} = 0.2 = (\boldsymbol{\Omega})_{j+2,j}$ for $j = 1, \ldots, p-2$, and all remaining elements equal 0.1.
- $\mathbf{T}$: The target matrix comprises of only zeros expect for the first row and column that are set equal to those of the precision matrix: $\mathbf{T}_{1,*} = \boldsymbol{\Omega}_{1,*}$ and $\mathbf{T}_{*,1} = \boldsymbol{\Omega}_{*,1}$
- $\boldsymbol{\Lambda}$: The penalty matrix for the regular ridge precision estimator is of course $\boldsymbol{\Lambda} = \lambda \mathbf{1}_{pp}$, while for the generalized ridge precision estimator is parametrized: $(\boldsymbol{\Lambda})_{1,j} = 10^{10} = (\boldsymbol{\Lambda})_{j,1}$ for $j = 1, \ldots, p$, and $(\boldsymbol{\Lambda})_{j,j'} = \lambda$ for all $j, j' = 2, \ldots, p$.

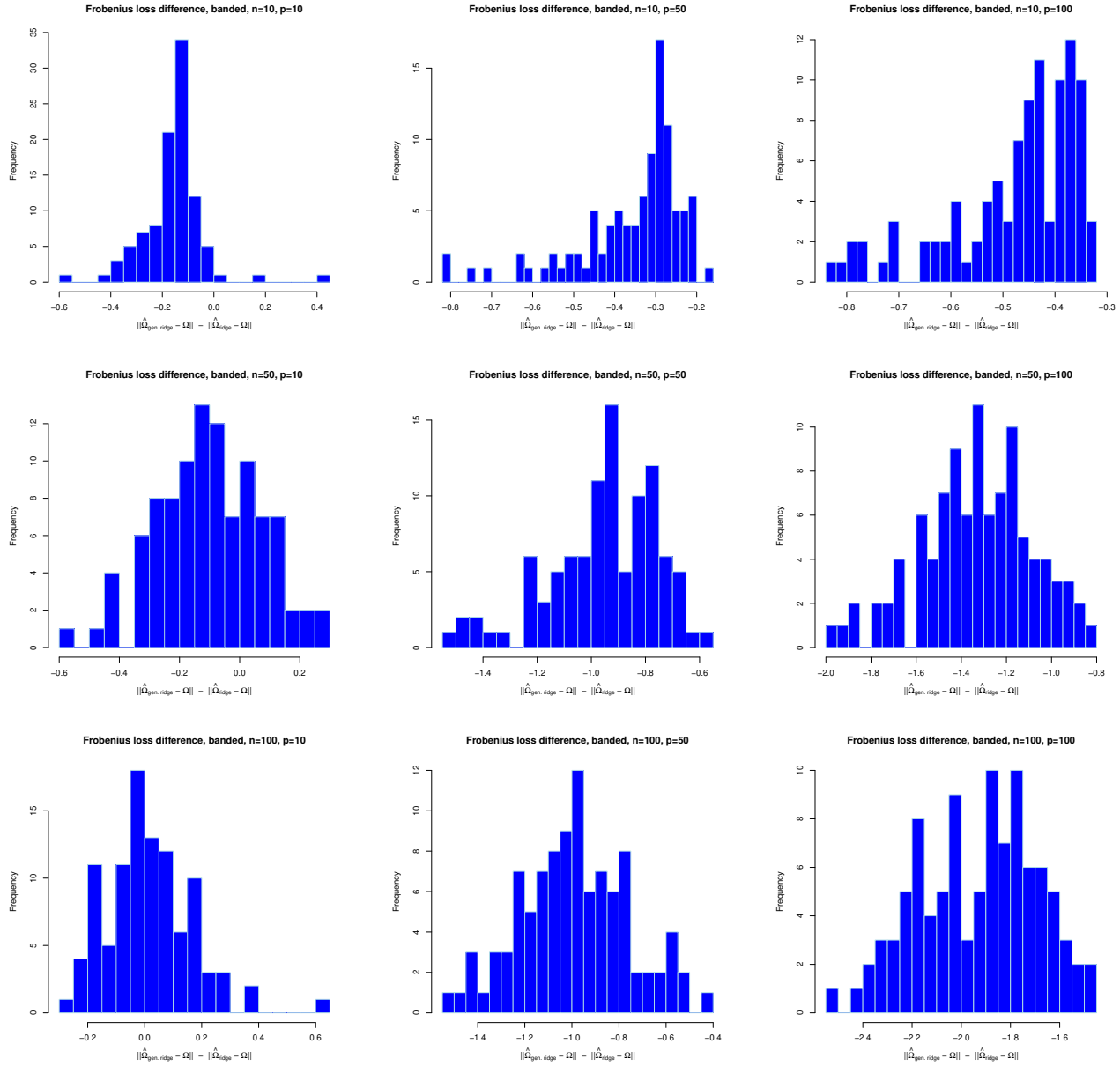

Figure 2: Histograms of the Frobenius loss difference between the generalized and regular precision estimators, $\|\widehat{\mathbf{\Omega}}_{\text{gen}}(\mathbf{\Lambda}, \mathbf{T}) - \mathbf{\Omega}\|_F$ - $\|\widehat{\mathbf{\Omega}}_{\text{reg}}(\lambda, \mathbf{T}) - \mathbf{\Omega}\|_F$, with banded $\mathbf{\Omega}$ and $\mathbf{T}$ for various sample sizes and dimensions. Rows and columns correspond sample size and dimension, $n, p \in \{10, 50, 100\}$, in ascending order.

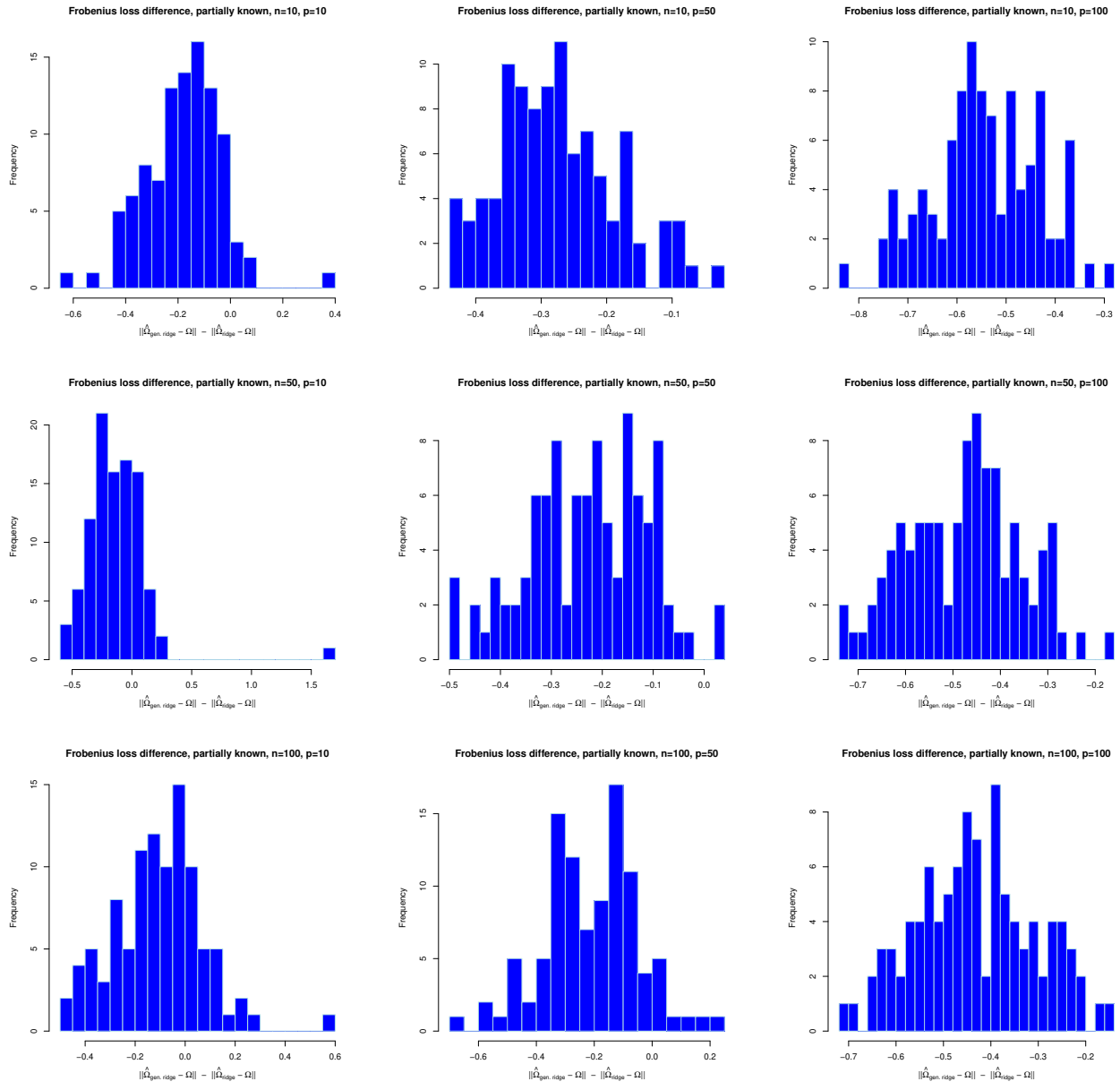

Figure 3: Histograms of the Frobenius loss difference between the generalized and regular precision estimators, $\|\widehat{\mathbf{\Omega}}_{\text{gen}}(\mathbf{\Lambda}, \mathbf{T}) - \mathbf{\Omega}\|_F$ - $\|\widehat{\mathbf{\Omega}}_{\text{reg}}(\lambda, \mathbf{T}) - \mathbf{\Omega}\|_F$, with full $\mathbf{\Omega}$, partially correct $\mathbf{T}$, $\mathbf{\Lambda}$ that totally shrinks $\mathbf{\Omega}$ towards $\mathbf{T}$ on the correct part, and for various sample sizes and dimensions. The loss difference is restricted to that part of the precision matrix that is unknown. In the grid of histograms rows and columns correspond sample size and dimension, $n, p \in \{10, 50, 100\}$, in ascending order.

Application: additional figures

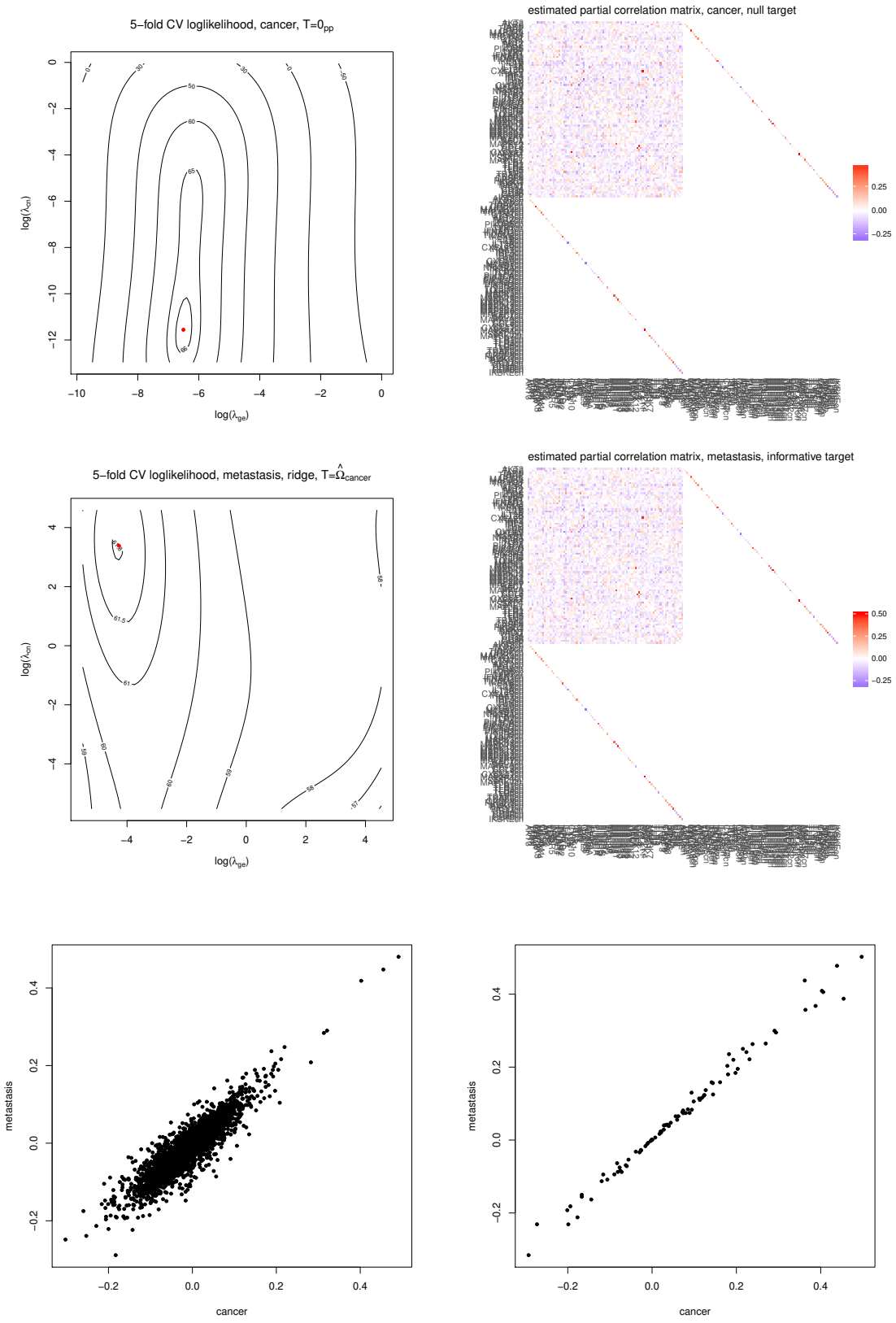

Figure 4: The upper two left-hand side panels contain the contour plots of the 5-fold cross-validated loglikelihood for both cancer (top) and (bottom) metastasis analysis with the generalized ridge estimator. The corresponding right-hand side panels contain heatmaps of the generalized ridge estimates with the optimal cross-validated penalty parameters. The bottom two panels show the scatter plots of the partial correlation in both groups, separated by those among the genes' expression levels and those between a gene's expression level

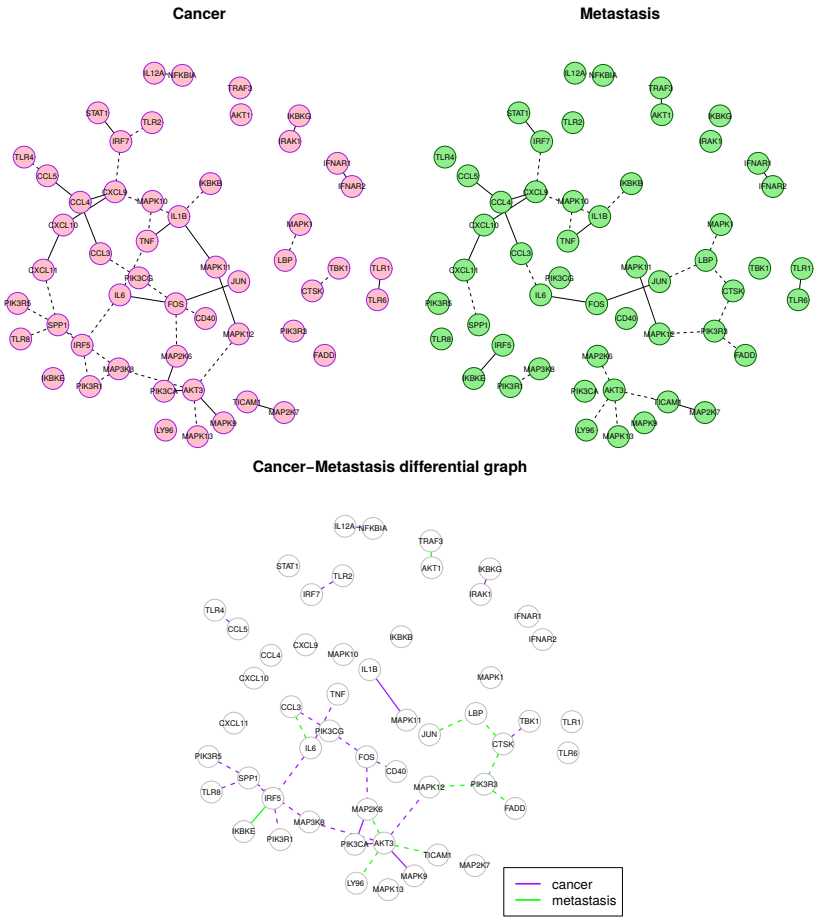

Figure 5: Reconstructed network (from the generalized ridge precision estimates) of the toll-like receptor pathway in cancer (top left panel) and metastasis (middle right panel), now with the same layout. For clarity the networks are limited to inferred interactions among genes' expression levels, ignoring those between a gene' expression levels and its DNA copy number as well as unconnected nodes. Dashed and solid edges indicate negative and positive, respectively, signs of the associated partial correlations. The bottom panel shows the differential graph, i.e. edges that are present in one group and not the other.

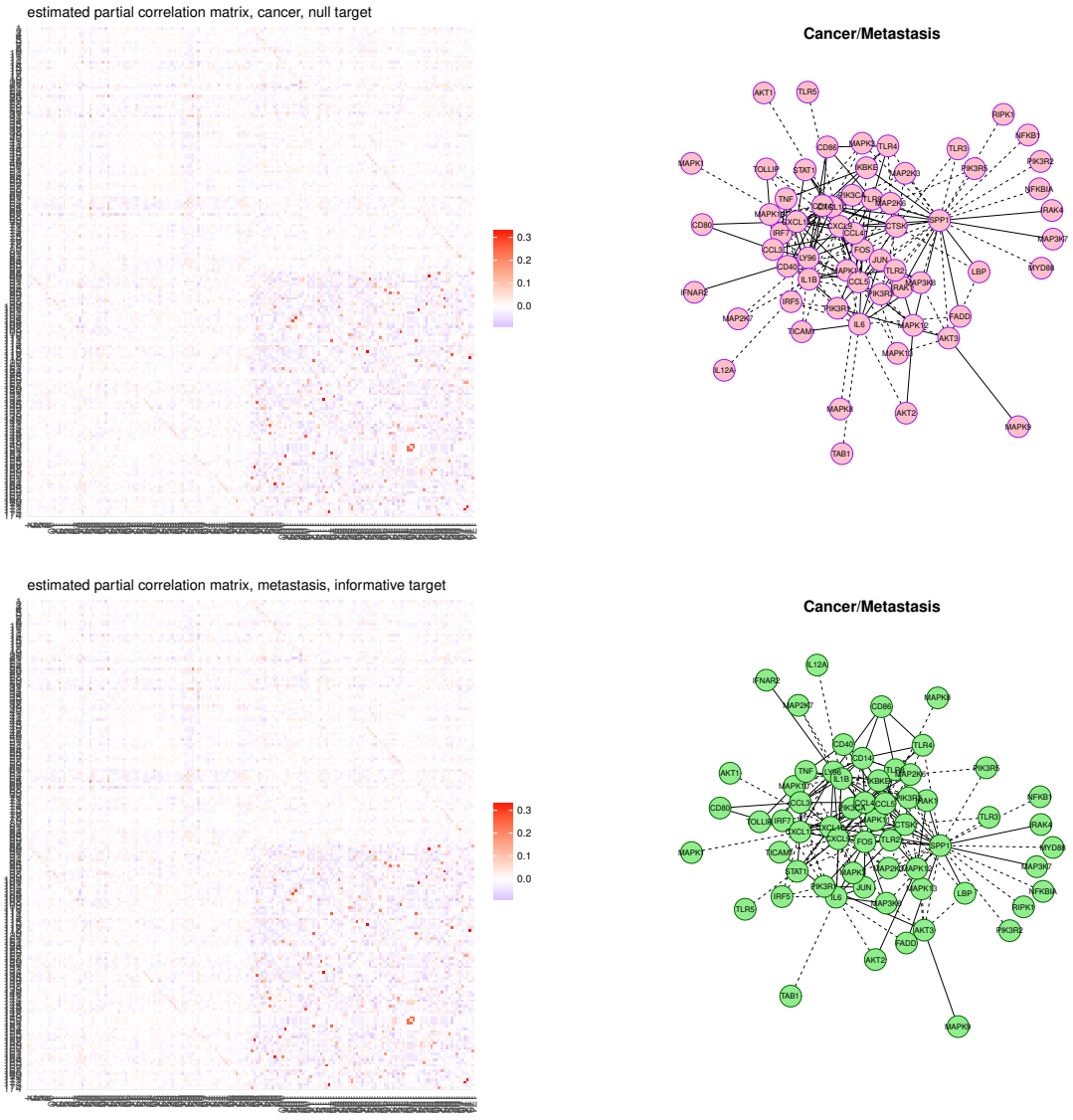

Figure 6: The left-hand side panels contain heatmaps of the regular ridge estimates with the optimal cross-validated penalty parameters of the precision matrix of the toll-like receptor pathway in cancer (top left panel) and metastasis (middle left panel). Reconstructed network (from the regular ridge precision estimates depicted in the heatmaps) of the toll-like receptor pathway in cancer (top left panel) and metastasis (middle right panel), with the same layout. For clarity the networks are limited to inferred interactions among genes' expression levels, ignoring those between a gene' expression levels and its DNA copy number as well as unconnected nodes. Dashed and solid edges indicate negative and positive, respectively, signs of the associated partial correlations.

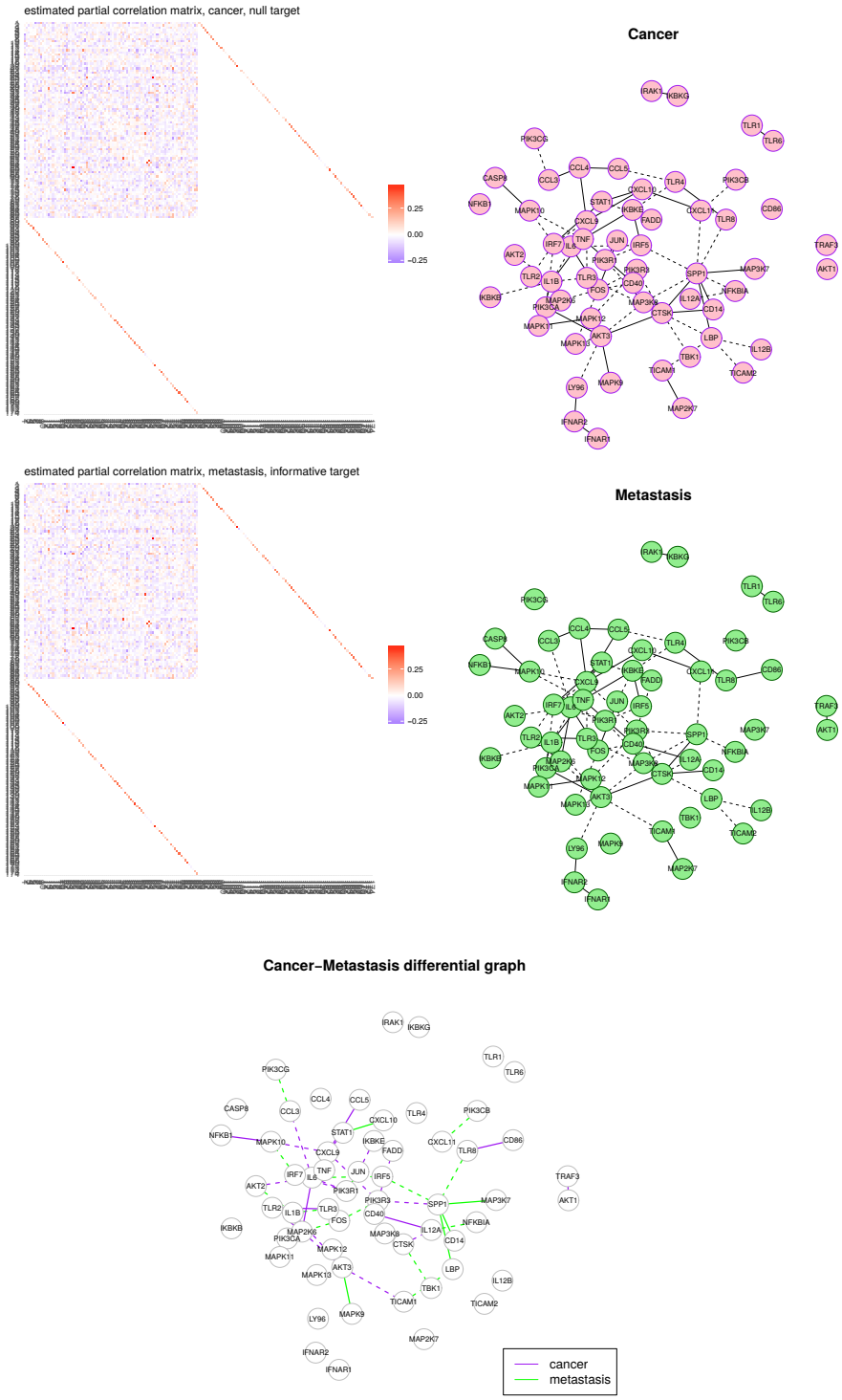

Figure 7: The left-hand side panels contain heatmaps of the regular ridge (with chordal support) estimates with the optimal cross-validated penalty parameters of the precision matrix of the toll-like receptor pathway in cancer (top left panel) and metastasis (middle left panel). Reconstructed network (from the ridge precision estimates with chordal support depicted in the heatmaps) of the toll-like receptor pathway in cancer (top left panel) and metastasis (middle right panel), with the same layout. For clarity the networks are limited to inferred interactions among genes' expression levels, ignoring those between a gene' expression levels and its DNA copy number as well as unconnected nodes. Dashed and solid edges indicate negative and positive, respectively, signs of the associated partial correlations. The bottom panel shows the differential graph, i.e. edges that are present in

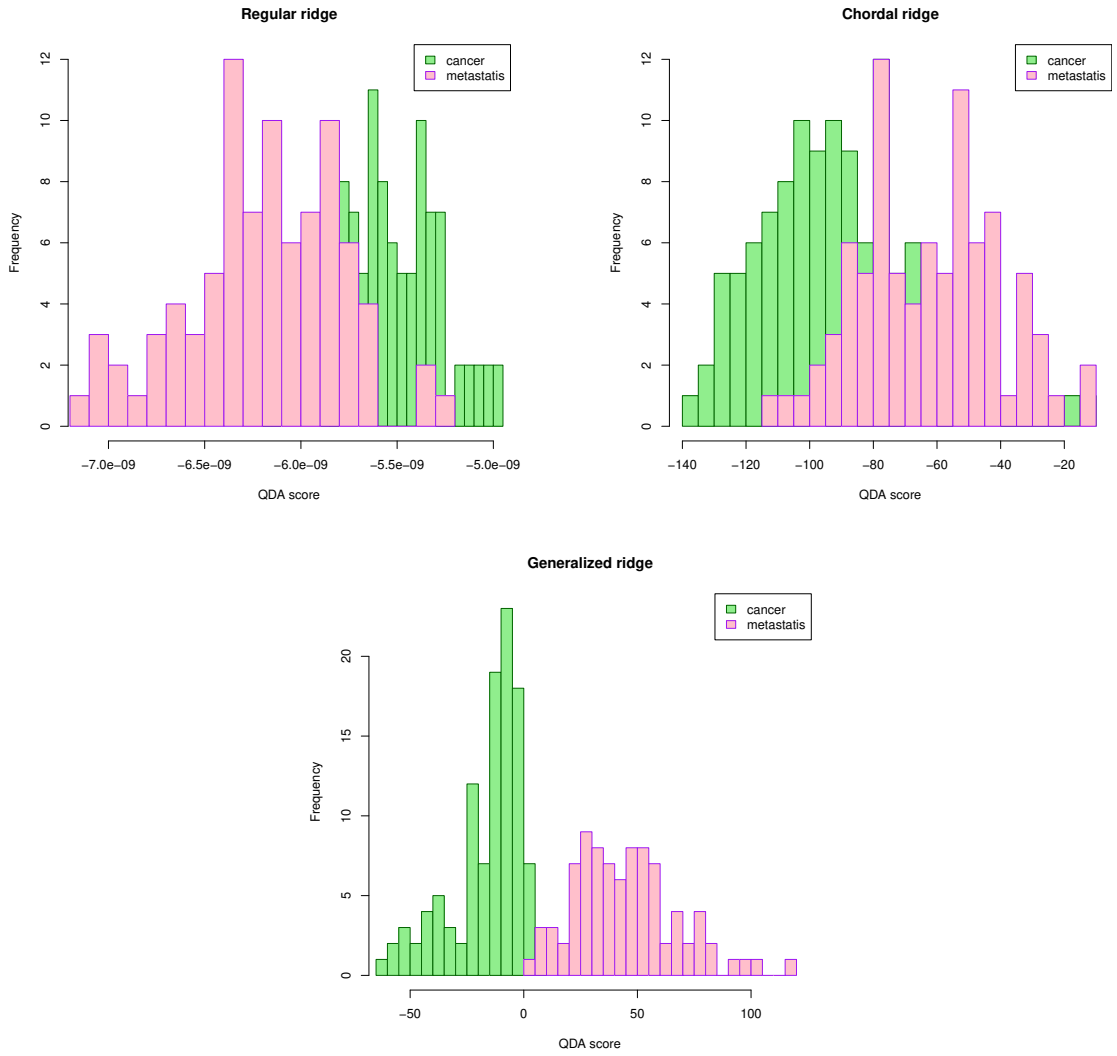

Figure 8: Histograms of QDA (Quadratic Discriminant Analysis) scores for cancer vs. metastasis classification using the regular (top left panel), chordal support (top right panel), and generalized (bottom panel) ridge precision estimates of both groups.