arXiv:cs.CE/0506023 v1 8 Jun 2005

Sparse Covariance Selection via Robust Maximum Likelihood Estimation

Onureena Banerjee*, Alexandre d'Aspremont†, Laurent El Ghaoui‡

June 9, 2005

Abstract

We address a problem of covariance selection, where we seek a trade-off between a high likelihood against the number of non-zero elements in the inverse covariance matrix. We solve a maximum likelihood problem with a penalty term given by the sum of absolute values of the elements of the inverse covariance matrix, and allow for imposing bounds on the condition number of the solution. The problem is directly amenable to now standard interior-point algorithms for convex optimization, but remains challenging due to its size. We first give some results on the theoretical computational complexity of the problem, by showing that a recent methodology for non-smooth convex optimization due to Nesterov can be applied to this problem, to greatly improve on the complexity estimate given by interior-point algorithms. We then examine two practical algorithms aimed at solving large-scale, noisy (hence dense) instances: one is based on a block-coordinate descent approach, where columns and rows are updated sequentially, another applies a dual version of Nesterov's method.

1 Introduction

Consider a data set with n variables, drawn from a multivariate Gaussian distribution $\mathcal{N}(0, \Sigma)$, where the covariance matrix Σ is unknown. When the number of variables n is large, estimating the entries of Σ becomes a significant problem.

In 1972, Dempster [Dem72] suggested reducing the number of parameters to be estimated by setting to zero some elements of the inverse covariance matrix Σ^{-1}. This idea, known as covariance selection, can lead to a more robust estimate of Σ if enough entries in its inverse are set to zero. Furthermore, conditional independence properties of the distribution are determined by the locations of zeros in Σ^{-1}. Hence the approach can be used to simultaneously determine a robust estimate of the covariance matrix and, perhaps more importantly, discover structure, namely conditional independence properties, in the underlying graphical model, which is a useful information in its own right. Specific applications of covariance selection include speech recognition [Bil99, CG99, Bil00] and gene expression data analysis [DW04, DHJ$^+$04].

In [Bil00], Bilmes proposed a method for covariance selection based on choosing statistical dependencies according to conditional mutual information computed using training data. Other recent work involves identifying those Gaussian graphical models that are best supported by the data and any available prior information on the covariance matrix. This approach is used by [JCD$^+$04, DW04] and is applied to gene expression data. Recently, [HLP05, DRV05] considered penalized maximum likelihood estimation and [DRV05] in particular propose a set of large scale methods to solve *sparse* problems.

In this paper we focus on the problem of computing a sparse estimate of the covariance matrix using only a large-scale, a priori *dense* and noisy sample covariance matrix Σ. Our approach is based on l_1-penalized maximum likelihood, and can be interpreted as a "robust maximum likelihood" method, where we assume that the true covariance matrix is within a component-wise bounded perturbation of the sample one, and the estimate is chosen to maximize

*EECS Department, UC Berkeley, Berkeley, CA 94720. onureena@eecs.berkeley.edu
†ORFE Department, Princeton University, Princeton, NJ 08544. aspremon@princeton.edu
‡EECS Department, UC Berkeley, Berkeley, CA 94720. elghaoui@eecs.berkeley.edu

the worst-case (minimal) likelihood. One of our goals is to provide an efficient algorithm to discover structure, rather than solve problems where the inverse covariance matrix has an already known sparse structure, as in [DRV05].

Our contributions are as follows: we specify the problem and outline some of its basic properties (section 2); we describe how one can apply a recent methodology for convex optimization due to Nesterov [Nes03], and obtain as a result a computational complexity estimate that has a much better dependence on problem size than interior-point algorithms (section 3); we present two algorithms for solving large dense problems (section 4): a version of Nesterov's method applied to the dual problem, and a block coordinate descent method. In section 5 we present the results of some numerical experiments comparing these two algorithms.

2 Preliminaries

Problem setup. For a given covariance matrix $\Sigma \in \mathbf{S}^n_+$, and reals numbers $0 \leq \alpha < \beta$, our problem is formulated as

$$p^* := \max_X \{\log\det X - \langle \Sigma, X\rangle - \rho\|X\|_1 : \alpha \mathbf{I}_n \preceq X \preceq \beta \mathbf{I}_n\} \tag{1}$$

with variable $X \in \mathbf{S}^n$, and $\|X\|_1 := \sum_{i,j=1}^n |X_{ij}|$. The parameter $\rho > 0$ controls the size of the penalty, hence the sparsity of the solution. Here $\langle \Sigma, X\rangle = \mathbf{Tr}\, \Sigma X$ denotes the scalar product between the two symmetric matrices Σ, X. The penalty term involving the sum of absolute values of the entries of X is a proxy for the number of its non-zero elements, and is often used in regression techniques, such as LASSO in [Tib96], when sparsity of the solution is a concern. In our model, the bounds (α, β) on the eigenvalues of X are fixed, and user-chosen; although we allow $\alpha = 0$, $\beta = +\infty$ in our model, such bounds are useful in practice to control the condition number of the solution.

For $\rho = 0$, and provided $\Sigma \succ 0$, problem (1) has a unique solution $X^* = \Sigma^{-1}$, and the corresponding maximum-likelihood estimate $\hat{\Sigma} = \Sigma$. Due to noise in the data, in practice, the sample estimate Σ may not have a sparse inverse, even if the underlying graphical model exhibits conditional independence properties; by striking a trade-off between maximality of the likelihood and number of non-zero elements in the inverse covariance matrix, our approach is potentially useful at *discovering structure*, precisely conditional independence properties in the data. At the same time, it serves as a regularization technique: when Σ is rank-deficient, there is no well-defined maximum-likelihood estimate, whereas it can be shown that the solution to problem (1) is always unique and well-defined for $\rho > 0$, even if $\alpha = 0$ and/or $\beta = \infty$.

Robustness, duality, and bounds. In the case when $\alpha = 0$, $\beta = \infty$, we write (1) as

$$\max_{X \succ 0} \min_{\|U\|_\infty \leq \rho} \log\det X - \langle X, \Sigma + U\rangle,$$

where $\|U\|_\infty$ denotes the maximal absolute value of the entries of U. This corresponds to seeking an estimate with maximal worst-case likelihood, over all componentwise bounded additive perturbations $\Sigma + U$ of the sample covariance matrix Σ. Such a "robust optimization" interpretation can be given to a number of estimation problems, most notably support vector machines for classification.

The above leads to the following equivalent, dual problem (with $\alpha = 0$, $\beta = \infty$):

$$d^* := \min\ -\log\det(\Sigma + U) - n : \|U\|_\infty \leq \rho, \quad \Sigma + U \succ 0. \tag{2}$$

Note that the diagonal elements of an optimal U are simply $U_{ii} = \rho$. The corresponding covariance matrix estimate is $\hat{\Sigma} := \Sigma + U$. Since the above dual problem has a compact feasible set, both primal and dual problems are equivalent.

In the case when $\alpha = 0$, $\beta = \infty$, we can derive finite, a priori bounds on the condition number of the solution. Indeed, we can show that we can always assume $\alpha(n)\mathbf{I}_n \preceq X \preceq \beta(n)\mathbf{I}_n$, where $\alpha(n) := 1/(\|\Sigma\| + n\rho)$ and $\beta(n) = n/\rho$. These bounds guarantee strict convexity of the objective, as well as existence, boundedness and uniqueness of the solution when $\rho > 0$, even if no bounds are set a priori.

3 Complexity

First- vs. second-order methods. Of course, problem (1) is convex and can readily be solved using interior point methods (see [BV04] for example). However, such second-order methods become quickly impractical for solving (1), since the corresponding complexity to compute an ϵ-suboptimal solution is $O(n^6 \log(1/\epsilon))$. The authors of [DRV05] developed interior-point algorithms to solve a problem related to (1), where a (sparse) structure of the solution is known a priori. Here our focus is on relatively large, dense problems, for which a solution of moderate accuracy is enough. Note that we cannot expect to do better than $O(n^3)$, which is the cost of solving the non-penalized problem for dense covariance matrices Σ. The recently-developed first-order algorithms due to [Nes03] trade-off a better dependence on problem size against a worse dependence on accuracy, usually $1/\epsilon$ instead of its logarithm. The method we describe next has a complexity of $O(n^{4.5}/\epsilon)$. This is a substantial improvement over interior-point methods when ϵ is not too small, which is often the case in practice. In addition, the memory space requirement of the first-order method is much lower than that of interior-point methods, which involve storing a dense Hessian, and hence become quickly prohibitive with a problem having $O(n^2)$ variables.

Nesterov's format. We can write (2) in the format given in [Nes03]:

$$\min_{X \in \mathcal{Q}_1} \max_{U \in \mathcal{Q}_2} \hat{f}(X) + \langle A(X), U \rangle = \min_{X \in \mathcal{Q}_1} f(X),$$

where we define $\hat{f}(X) = -\log\det X + \langle \Sigma, X \rangle$, $A = \rho I_{n^2}$, and $\mathcal{Q}_1 := \{X \in \mathcal{S}^n : \alpha \mathbf{I}_n \preceq X \preceq \beta \mathbf{I}_n\}$, $\mathcal{Q}_2 := \{U \in \mathcal{S}^n : \|U\|_\infty \leq 1\}$.

Prox-functions and related parameters. To $\mathcal{Q}_1$ and $\mathcal{Q}_2$ we now associate norms and so-called prox-functions. For $\mathcal{Q}_1$, we use the Frobenius norm, and a prox-function

$$d_1(X) = -\log\det X + \log\beta.$$

The function d_1 is strongly convex on $\mathcal{Q}_1$, with a convexity parameter of $\sigma_1 = 1/\beta^2$, in the sense that $\nabla^2 d_1(X)[H, H] = \mathbf{Tr}(X^{-1}HX^{-1}H) \geq \beta^{-2}\|H\|_F^2$ for every H. Furthermore, the center of the set, $X_0 := \arg\min_{X \in \mathcal{Q}_1} d_1(X)$ is $X_0 = \beta \mathbf{I}_n$, and satisfies $d_1(X_0) = 0$. With our choice, we have $D_1 := \max_{X \in \mathcal{Q}_1} d_1(X) = n \log(\beta/\alpha)$.

To $\mathcal{Q}_2$, we also associate the Frobenius norm, and the prox-function $d_2(U) = \|U\|_F^2/2$. With this choice, the center U_0 of $\mathcal{Q}_2$ is $U_0 = 0$. Furthermore, the function d_2 is strictly convex on its domain, with convexity parameter with respect to the 1-norm $\sigma_1 = 1$, and we have $D_2 := \max_{U \in \mathcal{Q}_2} d_2(U) = n^2/2$.

The function $\hat{f}$ has a gradient that is Lipschitz-continuous with respect to the Frobenius norm on the set $\mathcal{Q}_1$, with Lipschitz constant $M = 1/\alpha^2$. Finally, the norm (induced by the Frobenius norm) of the operator A is $\|A\| = \rho$.

Idea of the method. The method is based on replacing the objective of the original problem, $f(X)$, with $f_\epsilon(X)$, where $\epsilon > 0$ is the given desired accuracy, and f_ϵ is the penalized function involving the prox-function d_2:

$$f_\epsilon(X) := \hat{f}(X) + \max_{U \in \mathcal{Q}_2} \langle X, U \rangle - (\epsilon/2D_2) d_2(U). \tag{3}$$

The above function turns out to be a smooth uniform approximation to f everywhere, with maximal error $\epsilon/2$. Furthermore, the function f_ϵ is Lipschitz-continuous, with Lipschitz constant given by $L(\epsilon) := M + D_2\|A\|^2/(2\sigma_2\epsilon)$. A specific gradient algorithm for smooth, constrained convex minimization is then applied to the smooth convex function f_ϵ, with convergence rate in $O(\sqrt{L(\epsilon)/\epsilon})$. Specifically, the algorithm is guaranteed to produce an ϵ-suboptimal solution after a number of steps not exceeding

$$N(\epsilon) := 4\|A\|\sqrt{\frac{D_1 D_2}{\sigma_1\sigma_2}} \cdot \frac{1}{\epsilon} + \sqrt{\frac{M D_1}{\sigma_1 \epsilon}} = \frac{\kappa\sqrt{n(\log\kappa)}}{\epsilon}(4n\alpha\rho + \sqrt{\epsilon}). \tag{4}$$

Nesterov's algorithm. Choose $\epsilon > 0$ and set $X_0 = \beta \mathbf{I}_n$, and proceed as follows.
For $k = 0, \ldots, N(\epsilon)$ **do**

1. Compute $\nabla f_\epsilon(X_k) = -X^{-1} + \Sigma + U^*(X_k)$, where $U^*(X)$ solves (3).
2. Find $Y_k = \arg\min_Y \{\langle \nabla f_\epsilon(X_k), Y - X_k\rangle + \frac{1}{2}L(\epsilon)\|Y - X_k\|_F^2 \ : \ Y \in \mathcal{Q}_1\}$.
3. Find $Z_k = \arg\min_X \left\{\frac{L(\epsilon)}{\sigma_1} d_1(X) + \sum_{i=0}^k \frac{i+1}{2}\langle \nabla f_\epsilon(X_i), X - X_i\rangle \ : \ X \in \mathcal{Q}_1\right\}$.
4. Update $X_k = \frac{2}{k+3} Z_k + \frac{k+1}{k+3} Y_k$.

Complexity estimate. For step 1, the gradient of f_ϵ is readily computed in closed form, via the computation of the inverse of X. Step 2 essentially amounts to projecting on $\mathcal{Q}_1$, and requires an eigenvalue problem to be solved; likewise for step 3. In fact, each iteration costs $O(n^3)$. The number of iterations necessary to achieve an objective with absolute accuracy less than ϵ is given in (4) by $N(\epsilon) = O(n/\epsilon)$ (if $\rho > 0$). Thus, the overall complexity when $\rho > 0$ is $O(n^{4.5}/\epsilon)$, as claimed.

4 Algorithms

The algorithms presented next address problem (1) in the case when no bounds are given a priori, that is, $\alpha = 0$ and $\beta = \infty$. While they do not share all the nice theoretical computational complexity properties of the algorithm presented earlier, they seem to work well in practice. We are currently implementing the algorithm in section 3 and will include it in our experiments in a future version of this paper.

A dual version of Nesterov's method. In our experiments, we have used a preliminary version of Nesterov's method with $\alpha = 0$ and $\beta = \infty$ that, instead of working on the primal (1), addresses the dual (2). The algorithm is essentially the same as that presented in section 3, with the following setup. Define $\mathcal{Q}_1 := \{x \ : \ U \in \mathcal{S}^n \ : \ \|U\|_\infty \leq 1\}$, $\mathcal{Q}_2 = \{U \in \mathcal{S}_+^n \ : \ \mathbf{Tr}\, U \leq n/\rho\}$, and

$$f(x) = \max_{U \in \mathcal{Q}_2} \langle A(X), U\rangle - \hat{\phi}(U), \quad A = -\mathbf{I}_{n^2}, \quad \hat{\phi}(u) = -\log\det U + \langle \Sigma, U\rangle.$$

To simplify the projections required by the algorithm, we make the assumption that ρ is small enough to ensure that $\mathcal{Q}_1$ is entirely included in the cone of positive semidefinite matrices. In practice, we have found that the algorithm does well, even when this condition does not hold. As in section 3, each step of this algorithm costs $O(n^3)$ operations.

To $\mathcal{Q}_1$, we associate the Frobenius norm in $\mathbf{R}^{n\times n}$, and a prox-function defined for $x \in \mathcal{Q}_1$ by $d_1(X) = (1/2)\|X\|_F^2$; the center of $\mathcal{Q}_1$ with respect to d_1 is $X_0 = 0$, and $D_1 = \max_{X\in\mathcal{Q}_1} d_1(X) = \rho^2 n^2/2$. Furthermore, the function d_1 is strictly convex on its domain, with convexity parameter with respect to the Frobenius norm $\sigma_1 = 1$. For $\mathcal{Q}_2$ we use the dual of the standard matrix norm (denoted $\|\cdot\|_2^*$), and a prox-function $d_2(U) = \mathbf{Tr}(U \log U) + \log n$, where $\log$ refers to the *matrix* (and not componentwise) logarithm. The center of the set $\mathcal{Q}_2$ is $U_0 = n^{-1}\mathbf{I}_n$, and $d_2(U_0) = 0$. We have $\max_{u\in\mathcal{Q}_2} d_2(u) \leq \log n := D_2$. The convexity parameter of d_2 on its domain with respect to $\|\cdot\|_2^*$, is bounded below by $\sigma_2 = 1/2$ [BTN04]. Finally, the norm of the operator A, with respect to the Frobenius norm and the dual of the standard norm, is 1.

Dual block-coordinate descent. We now describe an algorithm based on block-coordinate descent, which solves the dual problem (2) by optimizing over one column/row pair at a time. For simplicity, we consider the case when $\alpha = 0$ and $\beta = \infty$, which corresponds to providing no a priori bounds on the condition number of the solution.

The algorithm is initialized with the estimate $\hat{\Sigma} = \Sigma + \rho I \succ 0$, since the optimal diagonal values of U in (2) are simply ρ. To update $\hat{\Sigma}$, we solve (2) where all elements in U, except the off-diagonal values of one column (and corresponding row), are fixed. Using a permutation of rows and columns, we can always express the problem as one of optimizing over the last column/row pair:

$$\min_{u,v} \ \log\det \begin{pmatrix} \hat{\Sigma_{11}} & \hat{\Sigma}_{12} + u \\ * & \hat{\sigma}_{22} \end{pmatrix} \ : \ |u| \leq \rho, \tag{5}$$

where inequalities are understood componentwise, and the current estimate $\hat{\Sigma}$ of the covariance matrix is partitioned into four blocks, with $\hat{\Sigma}_{11}$ the upper-left $(n-1) \times (n-1)$ block, and $(\hat{\Sigma}_{12}, \hat{\sigma}_{22})$ the last column. The sub-problem (5) reduces to a simple box-constrained quadratic program. Specifically, the corresponding column/row $\hat{\Sigma}_{12}$ of the current estimate $\hat{\Sigma}$ is updated with $\hat{\Sigma}_{12} + u^*$, where

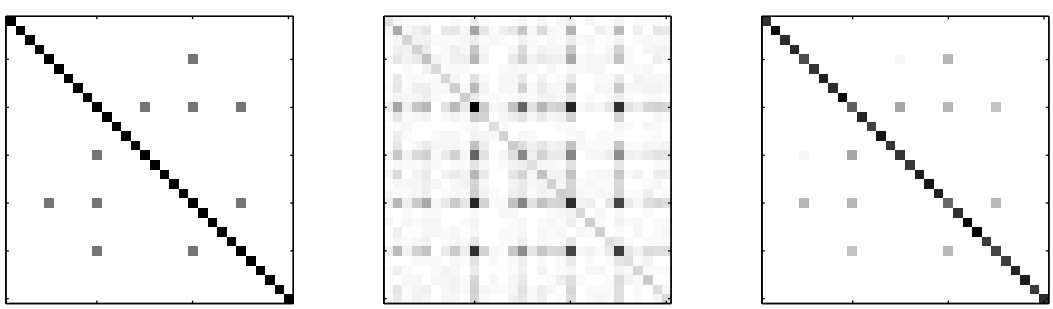
$$u^* = \arg\min_u \; u^T \hat{\Sigma}_{11}^{-1} u \; : \; |u| \leq \rho. \tag{6}$$

The stopping criterion we can use involves checking if the primal-dual gap of the problem (2) is less than a given tolerance ϵ, which translates as $\langle \Sigma, X \rangle + \rho \|X\|_1 \leq n + \epsilon$, where $X = \hat{C}^{-1}$. In practice though, we have found that a fixed number of sweeps (say, $K = 4$) through all column/row pairs is usually enough.

It can be proven that the algorithm converges, however we do not know the complexity of the algorithm. A simple modification of it, where we impose a condition number constraint on the estimate $\hat{\Sigma}$ at each column/row update, can be shown to converge in $O(n^6)$. Hence, the dual BCD algorithm is not really competitive from the theoretical complexity point of view, but the version given here remains attractive because it is simple to implement, and does well in practice. Since each column/row update costs $O(n^3)$, when only a few, fixed number K of sweeps through all columns is done, the cost is $O(Kn^4)$.

5 Numerical Results

Recovering structure. To illustrate our method we consider a synthetic example constructed by adding noise to a covariance matrix which has a sparse inverse. Precisely, we take a random $n \times n$ sparse matrix A, with $n = 30$, and then add a uniform noise of magnitude $\sigma = .13$ to its inverse to form our matrix Σ. Our concern here is to check if our methods reach a degree of precision sufficient to recover the problem structure, despite the noise masking it. Figure 1 shows the original sparse matrix A, the noisy inverse Σ^{-1} and the solution X of problem (1) solved using Nesterov's algorithm detailed in section 4 with parameter $\rho = \sigma$.



Original inverse A | Noisy inverse Σ^{-1} | Solution for $\rho = \sigma$

Figure 1: Recovering the sparsity pattern. We plot the original inverse covariance matrix A, the noisy inverse Σ^{-1} and the solution to problem (1) for $\rho = .13$.

Of course, the simple example above assumes that we set ρ equal to the noise level σ, so we look at what happens when the value of ρ is set above or below σ. In Figure 2 on the *left*, we solve problem (1) for various values of ρ. We plot the minimum $\min\{X_{ij} : A_{ij} \neq 0\}$ (solid) and mean (dash-dot) magnitude of those coefficients in the solution X corresponding to non zero coefficients in A (solid line) against the maximum $\max\{X_{ij} : A_{ij} = 0\}$ (dashed line) and mean (dotted line) magnitude of coefficients in X corresponding to zeros in A (we only consider off-diagonal coefficients). For all values of ρ within the interval V shown in Figure 2, the minimum is larger than the maximum, meaning that for all thresholding levels between the minimum and maximum, we exactly recover the original matrix

A. As we can observe, the range V of values of ρ for which this happens is fairly large and so is the gap between the minimum and maximum within the interval V.

In Figure 2 on the *right*, for various values of ρ, we randomly sample 10 noisy matrices Σ with $n = 50$ and $\sigma = .1$ and compute the number of misclassified zeros and nonzero elements in the solution to (1) produced by the block coordinate descent method. We plot the average percentage of errors (number of elements incorrectly set to zero plus number of elements incorrectly kept nonzero, divided by n^2), as well as error bars corresponding to one standard deviation.

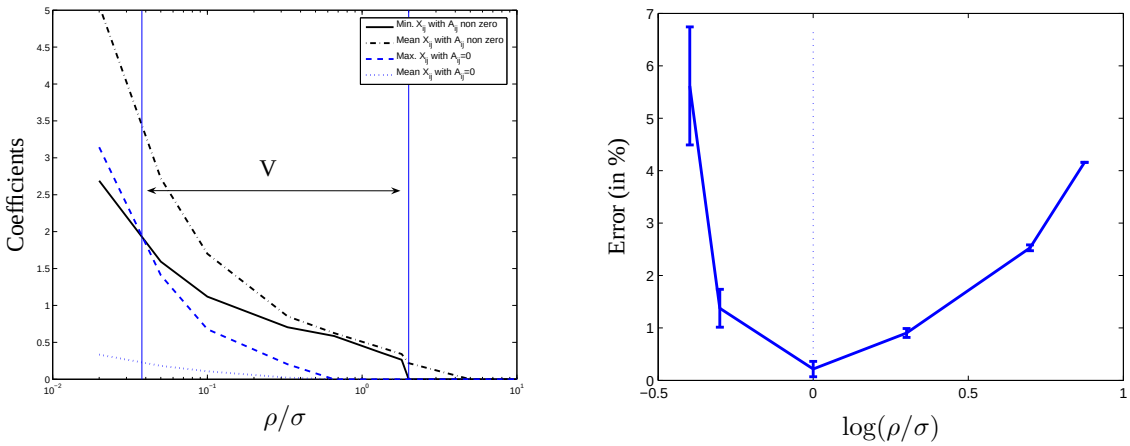


Figure 2: Recovering structure. *Left*: zero and nonzero matrix coefficients versus ρ. *Right*: average and standard deviation of the percentage of errors (false positives + false negatives) versus ρ on random problems.

Large-scale problems. We now compare Nesterov's and coordinate descent algorithms on a set of randomly generated examples. The noisy matrix Σ is generated as above and we plot in Figure 3 CPU time against problem size for various n and duality gap versus CPU time in seconds for a random problem of size 100 with a few nonzero coefficients. In practice, a low-precision solution to (1) is sufficient to identify most of the nonzero coefficients in the original matrix A. In Figure 4 we show the classification error made by the solution on the example with $n = 100$. Typical computing time for a problem with $n = 300$ is about 20 minutes. (All CPU times computed on a 1.5Ghz PowerBook G4 laptop).

Acknowledgments. The authors would like to thank Francis Bach, Peter Bartlett and Martin Wainwright for enlightening discussions on the topic.

References

[Bil99] J. A. Bilmes. Natural statistic models for automatic speech recognition. *Ph.D. thesis, UC Berkeley, Dept. of EECS, CS Division*, 1999.

[Bil00] J. A. Bilmes. Factored sparse inverse covariance matrices. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2000.

[BTN04] A. Ben-Tal and A. Nemirovskii. Non-euclidean restricted memory level method for large-scale convex optimization. *MINERVA Working paper*, 2004.

[BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[CG99] S. S. Chen and R. A. Gopinath. Model selection in acoustic modeling. *EUROSPEECH*, 1999.

[Dem72] A. P. Dempster. Covariance selection. *Biometrics*, 28(1):157–75, 1972.

[DHJ+04] A. Dobra, C. Hans, B. Jones, J.R. J. R. Nevins, G. Yao, and M. West. Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis*, 90(1):196–212, 2004.

[DRV05] J. Dahl, V. Roychowdhury, and L. Vandenberghe. Maximum likelihood estimation of gaussian graphical models: numerical implementation and topology selection. *UCLA preprint*, 2005.

[DW04] A. Dobra and M. West. Bayesian covariance selection. *Working paper, ISDS, Duke University*, 2004.

[HLP05] J. Z. Huang, N. Liu, and M. Pourahmadi. Covariance selection and estimattion via penalized normal likelihood. *Wharton Preprint*, 2005.

[JCD+04] B. Jones, C. Carvalho, C. Dobra, A. Hans, C. Carter, and M. West. Experiments in stochastic computation for high-dimensional graphical models. *ISDS Discussion Paper 04-01*, 2004.

[Nes03] Y. Nesterov. Smooth minimization of nonsmooth functions. *CORE discussion paper 2003/12 (Accepted by Math. Prog.)*, 2003.

[Tib96] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal statistical society, series B*, 58(267-288), 1996.

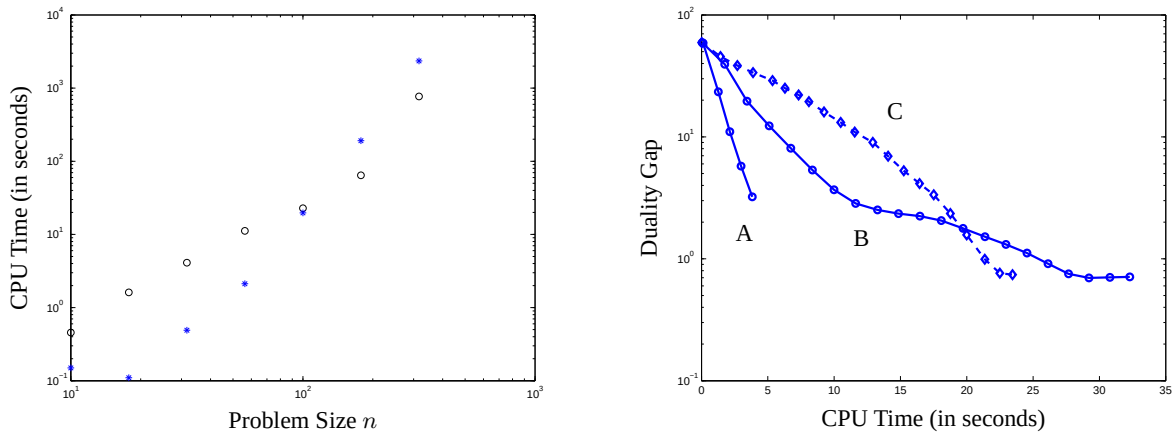


Figure 3: Computing time. *Left*: we plot CPU time (in seconds) to reach gap of $\epsilon = 1$ versus problem size n on random problems, solved using Nesterov's method (stars) and the coordinate descent algorithm (circles). *Right*: convergence plot a random problem with $n = 100$, this time comparing Nesterov's method where $\epsilon = 5$ (solid line A) and $\epsilon = 1$ (solid line B) with one sweep of the coordinate descent method (dashed line C).

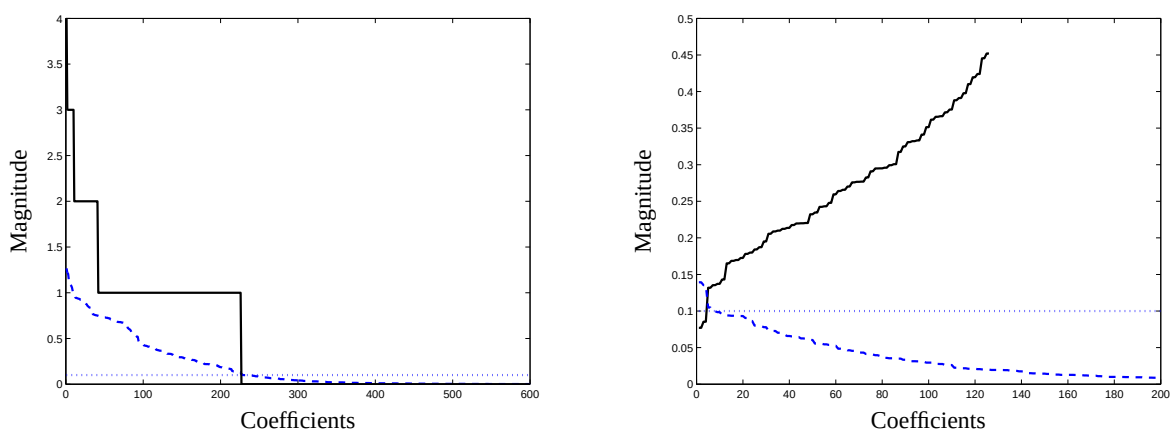


Figure 4: Classification Error. *Left*: we plot the coefficient magnitudes for the original matrix A (solid line) and the solution (dashed line) in decreasing order. The dotted line is at the signal to noise level σ. *Right*: we plot the magnitude of those coefficients in the solution associated with nonzero elements in A (solid line), ranked in increasing magnitude, together with the magnitude of those coefficients in the solution associated with zero elements in A (dashed line), in decreasing order of magnitude. Again, the dotted line is at the signal to noise level σ and we only consider off-diagonal elements.