Shrinkage Tuning Parameter Selection in Precision Matrices Estimation

Heng Lian
Division of Mathematical Sciences
School of Physical and Mathematical Sciences
Nanyang Technological University
Singapore 637371

ABSTRACT

Recent literature provides many computational and modeling approaches for covariance matrices estimation in a penalized Gaussian graphical models but relatively little study has been carried out on the choice of the tuning parameter. This paper tries to fill this gap by focusing on the problem of shrinkage parameter selection when estimating sparse precision matrices using the penalized likelihood approach. Previous approaches typically used K-fold cross-validation in this regard. In this paper, we first derived the generalized approximate cross-validation for tuning parameter selection which is not only a more computationally efficient alternative, but also achieves smaller error
rate for model fitting compared to leave-one-out cross-validation. For consistency in the selection of nonzero entries in the precision matrix, we employ a Bayesian information criterion which provably can identify the nonzero conditional correlations in the Gaussian model. Our simulations demonstrate the general superiority of the two proposed selectors in comparison with leave-one-out cross-validation, ten-fold cross-validation and Akaike information criterion.

**Keywords**: Adaptive Lasso; BIC; Generalized approximate cross-validation; Precision matrix; SCAD penalty.

1 INTRODUCTION

Undirected graphical models provide an easily understood way of describing and visualizing the relationships between multiple random variables. Usually the goal is to seek the simplest model that adequately explains the observations. In the Gaussian case, the zero entries in the inverse of the covariance matrix, or the precision matrix, correspond to independence of those two random variables conditional on all others.

Efficient estimation of a covariance matrix or its inverse is an important statistical problem. In network modeling, for example, correct identification of the nonzero entries provides an understanding of the relationships between the expression levels of different genes. There also exists an abundance of methods in multivariate analysis that requires the estimation of the covari-
ance matrix or its inverse, such as principal component analysis (PCA) or linear discriminant analysis (LDA).

Due to the recent surge in interests in the estimation of the covariance matrix with the appearance of a large amount of data generated by modern experimental advances such as microarrays, a large number of different estimators have been proposed by now. Mostly motivated by the fact that the sample covariance matrix is typically a noisy estimator when the sample size is not significant and the resulting dense matrix is difficult to interpret, almost all modern estimators regularize the matrix by making it sparse. This notion of sparsity in the context of estimating covariance matrices has been noticed by some author as early as in Dempster (1972) who simplified the matrix structure by setting some entries to zero.

Traditionally, for the identification of zero elements, forward-selection or backward-selection is performed with each element tested at each step. However, it is computationally infeasible even for data with a moderate number of variables. Li and Gui (2006) proposed using threshold gradient descent for estimating the sparse precision matrix in the Gaussian graphical models. Bickel and Levina (2008) developed asymptotic theories on banding methods for both covariance and precision matrix estimation based on direct thresholding of the elements. Another way to estimate the graphical model is to perform a regression for each variable on all the remaining ones. For example, Dobra and West (2004) used a Bayesian approach that employs a stochastic algorithm which can deal with tens of thousands of variables.

Recently, there has been much interest on the estimation of sparse covari-

3
ance or precision matrices using penalized likelihood method. Meinshausen and Buhlmann (2006) estimates the conditional independence separately for each random variable using the Lasso penalty. Note that setting up separate regressions for each node does not result in a valid likelihood for the covariance matrix and thus in order to obtain a positive-definite estimator, some post-processing is typically performed as the last step. Yuan and Lin (2007) used semi-definite programming algorithms to achieve sparsity by penalizing the normal likelihood with Lasso penalty on the elements of the precision matrix.

The Lasso penalty will force some of the coefficients to be exactly zero. Compared to traditional model selection methods using information criteria, Lasso is continuous and thus more stable. However, several authors (Meinshausen and Buhlmann, 2006; Zhao and Yu 2006) have noted that Lasso is in general not consistent for model selection unless some nontrivial conditions on the covariates are satisfied. Even when those conditions are indeed satisfied, the efficiency of the estimator is compromised when one insists on variable selection consistency since the coefficients are over-shrunk. To address these shortcomings of Lasso, Fan and Li (2001) proposed the smoothly clipped absolute deviation (SCAD) penalty which takes into account several desired properties of the estimator such as sparsity, continuity, asymptotic unbiasedness. They also show that the resulting estimator possesses the oracle property, i.e. it is consistent for variable selection and behaves the same as when the zero coefficients are known in advance. These results are extended to the case with a diverging number of covariates in Fan and Peng (2004). Zou (2006) proposed adaptive Lasso using a weighted $L_1$ penalty
with weights determined by an initial estimator and similar oracle property followed. The idea behind the adaptive lasso is to assign higher penalty for zero coefficients and lower penalty for larger coefficients.

In the context of Gaussian graphical models, Lam and Fan (2007) studied rates of convergence of sparse covariance/precision matrices estimation via a general penalty function including SCAD and adaptive Lasso penalties as special cases and showed that these penalty functions attenuated the bias problem associated with Lasso penalty. In Fan et al. (2008), through local linear approximation to the SCAD penalty function, efficient computation of the penalized likelihood is achieved using the graphical Lasso algorithm of Friedman et al. (2008). Oracle properties as defined in Fan and Li (2001) were shown for the precision matrix estimator in Fan et al. (2008).

The attractive oracle property of the penalized estimator depends critically on the appropriate choice of the tuning parameter. For penalized likelihood method, most of the studies mentioned above employed cross-validation (CV) for the selection of the tuning parameter. Cross-validation requires fitting the model based on different subsets of the observations multiple times, which increased the computational complexity of this approach. As an approximation to leave-one-out cross-validation (LOOCV), Craven and Wahba (1979) proposed generalized cross-validation (GCV) for smoothing spline, and further derived generalized approximate cross-validation (GACV) for non-Gaussian data (Dong and Wahba, 1996). We will follow similar strategies and derive GACV for the Gaussian graphical model that can be computed efficiently without multiple fitting of the model. However, for linear
regression, Wang et al. (2007) showed that generalized cross validation cannot select the tuning parameter satisfactorily. In particular, the tuning parameter chosen by GCV tend to overestimate the model size. Because of the asymptotic equivalence of generalized cross-validation, leave-one-out cross-validation and Akaike information criterion (AIC) in this simple model, the authors proposed to use Bayesian information criterion (BIC) for consistent model selection. We will demonstrate that similar conclusions can be reached for our problem of precision matrices estimation.

In this paper, we estimate the precision matrices using the same computational approach as in Fan et al. (2008). However, we focus on the problem of tuning parameter selection in penalized Gaussian graphical model. In the next section, two selectors, GACV and BIC, are proposed in this context. Simulation studies are carried out in Section 3 which demonstrate the superior performance of the proposed selectors. Finally, some remarks conclude the article in Section 4.

2 PENALIZED ESTIMATION AND TUNING PARAMETER SELECTION

Suppose \( x_1, \ldots, x_n \) are i.i.d. observations generated from a \( p \)-dimensional multivariate Gaussian distribution with mean 0 and unknown covariance matrix \( \Sigma_0 \), where \( x_i = (x_{i1}, \ldots, x_{ip})^T \). Denote the sample covariance matrix by \( S = \sum_{i=1}^{n} x_i x_i^T / n \). The inverse of \( S \) is a natural estimator of the precision
matrix \( \Omega_0 = \Sigma_0^{-1} \) which is the estimator that maximizes the Gaussian log-likelihood

\[
\max_\Omega \log |\Omega| - \text{Tr}(S\Omega),
\]

where \( |\Omega| \) is the determinant of the matrix \( \Omega \).

However, if the true precision matrix is known to be sparse or the dimension of the random vector is large, the performance of the estimator can typically be improved by maximizing the penalized log-likelihood instead:

\[
\max_\Omega \log |\Omega| - \text{Tr}(S\Omega) - \sum_{i=1}^p \sum_{j=1}^p p_{\lambda_{ij}}(|\omega_{ij}|),
\]

where \( \omega_{ij}, i = 1, \ldots, p, j = 1, \ldots, p, \) are the entries of \( \Omega \) and \( \lambda_{ij} \) are the tuning parameters, which for now are left unspecified to allow for very general penalty functions. Note that we also penalize the diagonal elements as in Fan et al. (2008).

In the above, using the penalty \( \lambda_{ij} = \lambda, p_\lambda(|x|) = \lambda|x| \) produces the Lasso estimator, while using \( \lambda_{ij} = \lambda/|\tilde{\omega}_{ij}|^\gamma \), where \( \gamma > 0 \) and \( \tilde{\omega}_{ij} \) is any consistent estimator of \( \omega_{ij} \), paired with the same \( p_\lambda(|x|) = \lambda|x| \) produces the adaptive Lasso estimator. Another commonly used penalty function proposed by Fan and Li (2001) is the SCAD penalty defined by its derivative

\[
p_\lambda(|x|) = \lambda I(|x| \leq \lambda) + \frac{(a\lambda - |x|)_+}{(a - 1)} I(|x| > \lambda),
\]

with \( a = 3.7 \) according to the suggestion made in Fan and Li (2001). Unlike the Lasso estimator, which produces substantial biases for large elements,
the adaptive Lasso as well as the SCAD estimator achieved the oracle property as defined in Fan and Li (2001), which estimates the precision matrix as well as in the ideal situation where the zero elements are known. Efficient maximization with either Lasso or adaptive Lasso penalty can be directly performed using the graphical Lasso algorithm (Friedman et al., 2008). To take advantage of graphical Lasso algorithm, Fan et al. (2008) suggested using local linear approximation to recast the SCAD penalized likelihood as weighted Lasso in each step. It was pointed out that a one-step algorithm can perform as well as the fully iterative local linear approximation algorithm. The reader are referred to Fan et al. (2008) for further details.

Here we focus on tuning parameter selection which consists of a single parameter $\lambda$ for all three penalty functions mentioned above. In Fan et al. (2008), it was shown that for both adaptive Lasso penalty and the SCAD penalty, when $\lambda \to 0$ and $\sqrt{n} \lambda \to \infty$, the resulting estimators attain the oracle property. Hence, the choice of $\lambda$ is critical. In Fan et al. (2008), they proposed to use K-fold cross-validation (KCV), with $K = 10$ probably the most popular choice in the literature. In K-fold cross-validation, one partitions the data into $K$ disjoint subsets and chooses $\lambda$ that maximizes the score

$$KCV(\lambda) = \sum_{k=1}^{K} n_k \left( \log |\hat{\Omega}^{(-k)}(\lambda)| - \text{Tr}(S^{(-k)} \hat{\Omega}^{(-k)}(\lambda)) \right),$$

where $n_k$ is the sample size of the $k-$th partition, $S^{(-k)}$ is the sample covariance matrix based on the data with $k-$th partition excluded, and $\hat{\Omega}^{(-k)}(\lambda)$ is the estimate of the precision matrix based on the data excluding the $k-$th
partition with $\lambda$ as the tuning parameter.

The usual leave-one-out cross-validation (LOOCV) is just a special case of the above with $K = n$ so that each partition consists of one single observation. Computation of KCV entails $K$ maximization problems fitted with each partition deleted in turn for each fixed $\lambda$ as well as a final maximization using the optimal tuning parameter. Thus the computation of KCV is infeasible for large datasets especially when $K$ is large. In the smoothing spline literature, the most popular approach for tuning parameter selection is the generalized cross-validation (GCV) for Gaussian data. For non-Gaussian data, Dong and Wahba (1996) proposed the generalized approximate cross-validation (GACV), which is obtained by constructing an approximation to the LOOCV based on the log-likelihood. The formula presented there does not directly apply to our problem since their derivation is based on regression problems. We also derive a GACV score based on an approximation to the LOOCV for our problem. The derivation of GACV is complicated by the multivariate nature of each left-out observation. The detail is deferred to Appendix A. Maybe surprisingly, even though GACV is motivated by an efficient approximation to LOOCV, it almost always performs better than LOOCV and sometimes better than ten-fold CV as demonstrated by the simulation studies in the next section.

In classical model selection literature, it is well understood that CV, GCV and AIC share similar asymptotic properties. All these criteria are less efficient but inconsistent for model selection (Shao, 1997; Yang, 2005). For penalized linear regression with the SCAD penalty, it has been verified by
Wang et al. (2007) that GCV is not able to identify the true model consistently when it is used for tuning parameter selection. To address this problem, Wang et al. (2007) employed a variable selection criterion known to be consistent in the classical literature, BIC, as the tuning parameter selector and proved that the resulting tuning parameter can identify the true model consistently. Similar conclusion has been drawn for adaptive Lasso (Wang and Leng, 2007). In Wang et al. (2008), the theory is further extended to linear regression problems with a diverging number of parameters. In our context, with BIC, we select the optimal $\lambda$ by minimizing

$$BIC(\lambda) = -\log |\hat{\Omega}(\lambda)| + \text{Tr}(\hat{S}\hat{\Omega}(\lambda)) + k \log \frac{n}{n},$$

where $k$ is the number of nonzero entries in the upper diagonal portion of the estimated precision matrix $\hat{\Omega}(\lambda)$.

We conjecture that our GACV proposed above is also not appropriate for model selection, although a formal proof seems illusive due to the complicated form of the GACV score. Nevertheless, we can extend the consistency proof for BIC using empirical process theory to show that the tuning parameter selected by BIC will result in consistent model identification. The result is stated below while the proof is relegated to Appendix B.

**Theorem 1** Denoting the optimal tuning parameter selected using BIC by $\hat{\lambda}_{BIC}$, if Conditions 1-3 in Appendix B hold, then the penalized likelihood estimator correctly identifies all the zero elements in the true precision matrix. That is, $\text{pr}(S_{\hat{\lambda}_{BIC}} = S_T) \to 1$, where $S_{\lambda} = \{(i,j): i \leq j, \hat{\omega}_{ij}(\lambda) \neq 0\}$ is
the set of nonzero entries above and including the diagonal in the estimated precision matrix and $\mathcal{S}_T$ is similarly defined to be the set of nonzero entries in the true precision matrix $\Omega_0$.

The conditions imposed on the penalty function can be verified for both SCAD penalty and adaptive Lasso penalty. Thus for these two penalty functions, BIC identifies the correct model for the precision matrix.

3 SIMULATIONS

In this section we compare the performance of different tuning parameter selectors for Lasso, adaptive Lasso and SCAD penalty estimators in Gaussian graphical models. For adaptive Lasso penalty with $\lambda_{ij} = \lambda/|\tilde{\omega}_{ij}|^\gamma$, we use the Lasso estimator as the initial consistent estimator and set $\gamma = 0.5$ following Fan et al. (2008). Besides CV, KCV (ten-fold), GACV and BIC, we also use AIC as the tuning parameter selector, which is defined by

$$AIC(\lambda) = -\log |\hat{\Omega}(\lambda)| + \text{Tr}(S\hat{\Omega}(\lambda)) + 2k/n,$$

where $k$ is the number of nonzero entries in the upper diagonal part of $\hat{\Omega}(\lambda)$.

We use three examples with different covariance structures in our simulation. The first one has a tridiagonal precision matrix:

$$\Omega_0 : \omega_{ii}^0 = 1, \omega_{i,i-1}^0 = \omega_{i,i+1}^0 = 0.5.$$
The second one has a dense precision matrix with exponential decay:

$$\Omega_0 : \omega^0_{ij} = 0.5^{|i-j|},$$

where no entries are exactly zero but many are so small that penalization is expected to reduce the variability of the estimators. The third one has a more general sparse precision matrix. For each non-diagonal element $$\omega^0_{ij}, i < j$$ of $$\Omega_0$$, with probability 0.8 we set it to be zero, otherwise we sample a value for $$\omega^0_{ij}$$ from a uniform distribution on $$[-1, -0.5] \cup [0.5, 1]$$. Each diagonal entry is set as twice of the sum of the absolute values of the corresponding row elements excluding the diagonal entry itself.

For each example, we use $$n$$ i.i.d. random vectors generated from a multivariate Gaussian distribution $$N(0, \Omega_0^{-1})$$ with dimension $$p = 20$$. We consider both $$n = 50$$ and $$n = 100$$. The errors are calculated based on 500 simulated datasets in each example. To compare the performance of different selectors, we use the Frobenius norm $$||\Omega_0 - \hat{\Omega}||_F$$ as well as the entropy loss (Yuan and Lin, 2007) defined by

$$\text{Entropy}(\Omega_0, \hat{\Omega}) = -\log|\Omega_0^{-1}\hat{\Omega}| + \text{Tr}(\Omega_0^{-1}\hat{\Omega}) - p.$$
The results for the Gaussian models are summarized in Tables 1-3 for Lasso penalty, adaptive Lasso penalty and SCAD penalty respectively, which gives the average losses and the average number of false positives and negatives for each case together with the corresponding standard errors. For sparse precision matrices, the BIC approach outperforms LOOCV, KCV and AIC in terms of the two loss measures as well as the sum of false positives and false negatives. For dense matrices, although the number of false negatives is generally larger compared to other selectors, which is certainly as expected, the performance of BIC in terms of loss is still superior. Based on our simulations, tuning parameter selected by AIC generally performs the worst. Finally, maybe surprisingly, the performance of GACV is almost always better than LOOCV, and in many cases also better than KCV. The reader should note that GACV can be computed much faster than either LOOCV or ten-fold CV.

4 CONCLUDING REMARKS

In this paper we compare several approaches for tuning parameter selection in penalized Gaussian graphical models. As an approximation to leave-one-out cross-validation, we derived generalized approximate cross-validation in the current context which is much faster to compute. Simulations show that GACV even outperforms the leave-one-out version. For model identification, we employ BIC for tuning parameter selection, and proved its consistency property. In our simulations with sparse matrices or dense matrices with
| Tridiagonal | BIC | AIC | CV  | KCV | GACV |
|------------|-----|-----|-----|-----|------|
| n=100      |     |     |     |     |      |
| Frobenius  | 7.02| 17.40| 7.11| 7.21| 6.37 |
| Entropy    | 0.80| 1.85 | 0.88| 0.89| 0.90 |
| FP         | 98.36| 205.97| 164.93| 153.17| 149.61 |
| FN         | 2.63| 0.16 | 2.90| 3.06| 0.98 |
|            | (1.44) | (7.60) | (1.24) | (1.38) | (1.12) |
|            | (0.17) | (0.51) | (0.14) | (0.15) | (0.15) |
|            | (35.46) | (32.49) | (16.80) | (19.09) | (32.08) |
|            | (2.60) | (0.55) | (2.60) | (2.67) | (1.34) |
| n=50       |     |     |     |     |      |
| Frobenius  | 11.39| 70.03| 12.36| 12.35| 12.03|
| Entropy    | 1.38| 7.29 | 1.41| 1.47| 1.45 |
| FP         | 109.51| 212.61| 190.37| 170.42| 171.36|
| FN         | 7.84| 0.77 | 6.15| 6.21| 0.98 |
|            | (2.93) | (35.30) | (2.02) | (1.99) | (5.68) |
|            | (0.35) | (2.72) | (0.19) | (0.17) | (0.49) |
|            | (51.95) | (25.04) | (33.86) | (35.81) | (35.87) |
| Dense      |     |     |     |     |      |
| n=100      |     |     |     |     |      |
| Frobenius  | 2.08| 5.31 | 2.94| 2.98| 1.84 |
| Entropy    | 0.93| 2.05 | 0.97| 0.94| 0.94 |
| FP         | 0 | 0 | 0 | 0 | 0 |
| FN         | 258.24| 54.38| 243.64| 257.50| 190.36|
|            | (51.63) | (30.45) | (37.94) | (42.15) | (39.54) |
| n=50       |     |     |     |     |      |
| Frobenius  | 3.32| 27.97| 3.57| 4.21| 3.68 |
| Entropy    | 1.48| 6.58 | 1.56| 1.75| 1.55 |
| FP         | 0 | 0 | 0 | 0 | 0 |
| FN         | 256.57| 34.42| 224.57| 256.85| 170.43|
|            | (76.06) | (32.89) | (42.33) | (9.63) | (44.12) |
| Random Sparse |     |     |     |     |      |
| n=100      |     |     |     |     |      |
| Frobenius  | 25.52| 52.63| 27.39| 26.19| 22.69|
| Entropy    | 0.87| 1.89 | 0.91| 0.89| 0.87 |
| FP         | 60.56| 156.34| 91.53| 83.16| 131.10|
| FN         | 47.35| 8.40 | 44.59| 45.34| 30.81 |
|            | (31.16) | (27.77) | (27.11) | (23.16) | (37.72) |
|            | (13.74) | (6.23) | (11.55) | (12.61) | (12.21) |
| n=50       |     |     |     |     |      |
| Frobenius  | 33.42| 93.59| 34.41| 34.69| 34.74|
| Entropy    | 1.36| 5.38 | 1.52| 1.51| 1.55 |
| FP         | 73.19| 277.11| 95.40| 94.89| 139.02|
| FN         | 58.17| 6.68 | 54.89| 55.36| 35.83 |
|            | (48.83) | (27.08) | (18.37) | (19.13) | (50.53) |
|            | (14.66) | (5.85) | (11.00) | (11.19) | (13.43) |

Table 1: Simulation results for Lasso estimators using five different tuning parameter selectors. The reported average errors are based on 500 simulated datasets. The numbers in the brackets are the corresponding standard errors.
### Table 2: Simulation results for adaptive Lasso estimators using five different tuning parameter selectors.

|                | BIC  | AIC  | CV   | KCV  | GACV |
|----------------|------|------|------|------|------|
| **Tridiagonal** |      |      |      |      |      |
| n=100          | 5.74 | 22.49| 7.24 | 7.11 | 6.31 |
|                | (1.64)| (10.01) | (2.38) | (2.21) | (2.40) |
| Entropy        | 0.78 | 1.88 | 0.91 | 0.90 | 0.82 |
|                | (0.18)| (0.57) | (0.23) | (0.21) | (0.24) |
| FP             | 58.08| 197.77| 94.63| 92.63| 73.01|
|                | (22.60)| (55.73) | (15.33) | (14.26) | (23.06) |
| FN             | 2.54 | 0.41 | 0.82 | 0.87 | 1.45 |
|                | (2.79)| (0.88) | (1.19) | (1.25) | (1.77) |
| n=50           | 13.45| 95.56| 20.21| 17.97| 13.94|
|                | (6.81)| (47.25) | (9.08) | (7.11) | (6.06) |
| Entropy        | 1.65 | 5.42 | 2.13 | 1.99 | 1.69 |
|                | (0.49)| (1.64) | (0.58) | (0.47) | (0.44) |
| FP             | 65.21| 277.63| 104.52| 99.21| 71.52|
|                | (26.98)| (57.21) | (20.21) | (19.15) | (26.92) |
| FN             | 7.15 | 2.21 | 4.42 | 4.68 | 6.73 |
|                | (5.33)| (3.07) | (3.39) | (3.58) | (5.18) |
| **Dense**      |      |      |      |      |      |
| n=100          | 1.63 | 6.25 | 2.14 | 1.91 | 1.75 |
|                | (0.64)| (2.76) | (0.62) | (0.52) | (0.44) |
| Entropy        | 0.79 | 1.98 | 0.99 | 0.92 | 0.85 |
|                | (0.17)| (0.57) | (0.20) | (0.18) | (0.17) |
| FP             | 0    | 0    | 0    | 0    | 0    |
|                | (0)  | (0)  | (0)  | (0)  | (0)  |
| FN             | 285.72| 133.80| 233.08| 243.84| 266.06|
|                | (29.37)| (54.91) | (20.58) | (19.12) | (28.14) |
| n=50           | 4.29 | 29.63| 5.34 | 5.27 | 4.04 |
|                | (3.00)| (14.67) | (1.56) | (1.68) | (1.85) |
| Entropy        | 1.84 | 5.94 | 2.26 | 2.23 | 1.82 |
|                | (0.78)| (1.54) | (0.37) | (0.42) | (0.54) |
| FP             | 0    | 0    | 0    | 0    | 0    |
|                | (0)  | (0)  | (0)  | (0)  | (0)  |
| FN             | 267.14| 106.85| 226.00| 227.57| 264.71|
|                | (41.44)| (49.48) | (16.11) | (19.02) | (40.81) |
| **General sparse** |      |      |      |      |      |
| n=100          | 19.35| 55.67| 25.06| 23.87| 20.13|
|                | (5.37)| (26.73) | (9.55) | (8.56) | (6.29) |
| Entropy        | 0.77 | 1.88 | 1.03 | 0.99 | 0.82 |
|                | (0.13)| (0.45) | (0.21) | (0.20) | (0.17) |
| FP             | 29.38| 215.02| 85.22| 79.18| 44.22|
|                | (19.88)| (45.96) | (18.72) | (21.08) | (24.29) |
| FN             | 54.65| 17.20| 44.26| 45.43| 51.36|
|                | (11.53)| (11.42) | (8.62) | (9.28) | (10.96) |
| n=50           | 34.35| 105.55| 50.31| 46.18| 39.06|
|                | (19.80)| (59.75) | (14.94) | (19.53) | (31.00) |
| Entropy        | 1.40 | 4.61 | 1.97 | 1.81 | 1.54 |
|                | (0.48)| (1.33) | (0.36) | (0.46) | (0.69) |
| FP             | 49.11| 212.26| 94.42| 83.44| 58.26|
|                | (26.73)| (47.45) | (15.31) | (25.35) | (38.58) |
| FN             | 54.34| 21.53| 45.11| 47.91| 52.72|
|                | (11.19)| (10.21) | (8.82) | (11.06) | (11.86) |
| Tridiagonal       | BIC     | AIC     | CV      | KCV     | GACV     |
|------------------|---------|---------|---------|---------|---------|
| n=100            | 5.84    | 22.80   | 10.29   | 8.02    | 6.07    |
|                  | (4.06)  | (7.20)  | (3.29)  | (2.83)  | (5.13)  |
| Entropy          | 0.90    | 2.20    | 1.25    | 1.06    | 1.10    |
|                  | (0.26)  | (0.44)  | (0.32)  | (0.31)  | (0.41)  |
| FP               | 64.08   | 158.79  | 99.45   | 61.23   | 84.93   |
|                  | (20.58) | (26.89) | (22.97) | (20.59) | (26.51) |
| FN               | 3.78    | 1.32    | 3.11    | 3.28    | 3.31    |
|                  | (2.93)  | (1.38)  | (2.94)  | (1.58)  | (2.54)  |
| n=50             | 12.39   | 62.83   | 18.95   | 15.38   | 18.39   |
|                  | (8.76)  | (36.70) | (11.29) | (4.24)  | (11.22) |
| Entropy          | 1.71    | 6.55    | 2.18    | 1.94    | 2.17    |
|                  | (1.41)  | (1.71)  | (0.76)  | (0.38)  | (1.74)  |
| FP               | 65.48   | 180.72  | 97.79   | 81.38   | 89.23   |
|                  | (23.53) | (27.83) | (23.24) | (15.75) | (25.80) |
| FN               | 8.15    | 4.76    | 4.85    | 7.69    | 7.84    |
|                  | (4.45)  | (2.85)  | (5.66)  | (7.16)  | (4.21)  |
| Dense            | 1.68    | 6.75    | 1.94    | 1.92    | 1.72    |
|                  | (0.78)  | (2.32)  | (0.23)  | (0.23)  | (1.25)  |
| Entropy          | 0.82    | 2.37    | 0.91    | 0.89    | 0.89    |
|                  | (0.20)  | (0.54)  | (0.12)  | (0.12)  | (0.23)  |
| FP               | 0       | 0       | 0       | 0       | 0       |
|                  | (0)     | (0)     | (0)     | (0)     | (0)     |
| FN               | 268.78  | 112.72  | 203.94  | 202.06  | 212.92  |
|                  | (51.99) | (30.07) | (31.79) | (21.59) | (46.41) |
| n=50             | 2.64    | 29.53   | 3.04    | 3.34    | 3.11    |
|                  | (1.87)  | (10.51) | (0.52)  | (0.57)  | (2.30)  |
| Entropy          | 1.23    | 7.03    | 1.48    | 1.57    | 1.37    |
|                  | (0.83)  | (1.29)  | (0.19)  | (0.19)  | (1.32)  |
| FP               | 0       | 0       | 0       | 0       | 0       |
|                  | (0)     | (0)     | (0)     | (0)     | (0)     |
| FN               | 244.80  | 103.87  | 219.14  | 237.14  | 221.14  |
|                  | (81.30) | (12.99) | (32.97) | (32.97) | (65.51) |
| General sparse   | 19.99   | 72.61   | 30.87   | 21.63   | 20.68   |
|                  | (6.69)  | (26.93) | (17.37) | (6.90)  | (8.05)  |
| Entropy          | 0.78    | 2.41    | 1.26    | 0.92    | 0.95    |
|                  | (0.31)  | (0.39)  | (0.38)  | (0.24)  | (0.27)  |
| FP               | 73.42   | 155.89  | 109.65  | 81.25   | 84.29   |
|                  | (27.41) | (24.76) | (21.94) | (17.53) | (23.37) |
| FN               | 58.10   | 26.51   | 49.84   | 50.07   | 41.29   |
|                  | (12.25) | (9.12)  | (11.46) | (12.54) | (11.64) |
| n=50             | 34.58   | 133.92  | 45.50   | 39.37   | 39.85   |
|                  | (25.36) | (86.84) | (15.84) | (14.19) | (22.57) |
| Entropy          | 1.06    | 5.43    | 2.04    | 1.24    | 1.31    |
|                  | (0.45)  | (1.65)  | (0.68)  | (0.52)  | (10.41) |
| FP               | 67.75   | 167.04  | 102.08  | 83.32   | 79.14   |
|                  | (26.64) | (36.23) | (23.48) | (15.30) | (25.94) |
| FN               | 53.85   | 29.95   | 38.51   | 44.34   | 44.00   |
|                  | (10.29) | (10.65) | (11.05) | (9.37)  | (9.25)  |

Table 3: Simulation results for SCAD estimators using five different tuning parameter selectors.
many small entries, tuning parameter selected based on BIC clearly has better performance than all other approaches.

**APPENDIXES**

**A. Derivation of GACV**

Denote the log-likelihood function by

\[ L(S, \Omega) = \log |\Omega| - \text{Tr}(S\Omega). \]

In this section only, to simplify notation, the shrinkage estimator \( \hat{\Omega}(\lambda) \) is simply denoted by \( \Omega \) and similarly \( \hat{\Omega}^{(-i)}(\lambda) \) is denoted by \( \Omega^{(-i)} \). Thus it is implicitly understood that the estimators depend on a fixed \( \lambda \). Let \( X_i = x_i x_i^T \) be the covariance matrix based on a single observation so that \( S = \sum_{i=1}^n X_i/n \). The LOOCV score is defined by

\[
CV(\lambda) = \sum_{i=1}^n \log |\Omega^{(-i)}| - \text{Tr}(\sum_i X_i \Omega^{(-i)}) = \sum_{i=1}^n L(X_i, \Omega^{(-i)})
\]

\[
= nL(S, \Omega) + \sum_{i=1}^n (L(X_i, \Omega^{(-i)}) - L(X_i, \Omega))
\]

\[
\approx nL(S, \Omega) + \sum_{i=1}^n \left[ \frac{dL(X_i, \Omega)}{d\Omega} \right]^T d\Omega,
\]

where we interpret \( dL(X_i, \Omega)/d\Omega = dL(X_i, \Omega)/d\text{vec}(\Omega) \) to be a \( p^2 \)-dimensional column vector of partial derivatives and similarly \( d\Omega = \text{vec}(\Omega^{(-i)} - \Omega) \). Besides, here as well as in the following, like the definition of generalized degrees
of freedom in Fan and Li (2001) and Wang et al. (2007), the partial derivatives corresponding to the zero elements in $\Omega$ are ignored.

Using matrix calculus such as presented in Bishop (2006), we have

$$\frac{dL(X_i, \Omega)}{d\Omega} = \text{vec}(\Omega^{-1} - X_i)$$

and we only need to deal with the term $d\Omega = \text{vec}(\Omega^{(-i)} - \Omega)$.

Denote the penalized log-likelihood based on the sufficient statistic $S$ by $\bar{L}(S, \Omega) = L(S, \Omega) - \sum_{i,j} p_{\lambda_{ij}} (|\omega_{ij}|)$, Taylor expansion gives us

$$0 = \frac{d\bar{L}(S^{(-i)}, \Omega^{(-i)})}{d\Omega} \approx \frac{d\bar{L}(S, \Omega)}{d\Omega} + \frac{d^2 \bar{L}(S, \Omega)}{d\Omega d\Omega} d\Omega + \frac{d^2 \bar{L}(S, \Omega)}{d\Omega dS} dS,$$

where $d^2 \bar{L}(S, \Omega)/d\Omega^2 = d(\bar{L}(S, \Omega)/d\text{vec}(\Omega))/d\text{vec}(\Omega)$ is the $p^2 \times p^2$ Hessian matrix, and $d^2 \bar{L}(S, \Omega)/d\Omega dS$ is defined similarly. Like before, $d\Omega = \text{vec}(\Omega^{(-i)} - \Omega)$ and $dS = \text{vec}(S^{(-i)} - S)$ actually denote their vectorized version.

Since $d\bar{L}(S, \Omega)/d\Omega = 0$, it immediately follows that

$$d\Omega = -\left(\frac{d^2 \bar{L}(S, \Omega)}{d\Omega^2}\right)^{-1} \frac{d^2 \bar{L}(S, \Omega)}{d\Omega dS} dS.$$

From matrix calculus, we have $d^2 \bar{L}(S, \Omega)/d\Omega dS = -I_{p^2 \times p^2}$ and $d^2 \bar{L}(S, \Omega)/d\Omega^2 = -\left(\Omega^{-1} \otimes \Omega^{-1} + D\right)$ where $D$ is a diagonal matrix with diagonal elements $-p''_{\lambda_{ij}} (|\omega_{ij}|)$. Thus we have the approximation $d\Omega \approx \left[d^2 \bar{L}(S, \Omega)/d\Omega^2\right]^{-1} dS$. 

18
Even for moderate $p$, inversion of this $p^2 \times p^2$ matrix is computationally infeasible. However, note that typically we consider only the situation with $\lambda = o(1)$ and $p_{\lambda ij}''(|\omega_{ij}|) = o(1)$ (for example, the second derivative for SCAD penalty function is exactly zero for nonzero elements). Thus we can approximate $(\Omega^{-1} \otimes \Omega^{-1} + D)^{-1}$ by $(\Omega^{-1} \otimes \Omega^{-1})^{-1} = \Omega \otimes \Omega$ and $d\Omega \approx -(\Omega \otimes \Omega)dS = \text{vec}(\Omega \cdot (S^{(i)} - S \cdot \Omega))$ which involves only $p \times p$ matrices and no inversion of matrices is required.

B. Proof of Theorem 1

We only need to assume the general conditions on the penalty function that guarantees the oracle property of the estimator with appropriately chosen tuning parameter. In particular, we assume that

**Condition 1.** $\max \{|p_{\lambda n}''(|\omega_{ij}^0|) : \omega_{ij}^0 \neq 0\} \to 0$.

**Condition 2.** $\liminf_{n \to \infty} \liminf_{x \to 0^+} p_{\lambda n}'(x)/\lambda_n > 0$.

**Condition 3.** The (theoretically) optimal tuning parameter satisfies $\lambda_n \to 0$ and $\sqrt{n}\lambda_n \to \infty$.

For an arbitrary model $S$ specified by the constraints that only some of the elements in the precision matrix can be nonzero, i.e. $S \subseteq \{(i, j) : i \leq j\}$ is the set of elements not constrained to be zero, denote by $L_S$ the value of the constrained maximized likelihood with infinite data: $L_S = \max_{\Omega} E(\log |\Omega| - \text{Tr}(X\Omega))$, where the maximization is performed over $\Omega$ with zero entries for all $(i, j) \in S$. We partition $\Lambda = [0, \infty)$ into three parts: $\Lambda_0 = \{\lambda \in \Lambda : S_{\lambda} = S_T\}$, $\Lambda_- = \{\lambda \in \Lambda : S_{\lambda} \nsubseteq S_T\}$, $\Lambda_+ = \{\lambda \in \Lambda : S_{\lambda} \supseteq S_T\}$.
where $\mathcal{S}_\lambda$ is the model identified by the estimator when $\lambda$ is used as the tuning parameter, and $\mathcal{S}_T$ is the true model $\mathcal{S}_T = \{(i,j) : i \leq j, \omega_{ij}^0 \neq 0\}$. We will prove separately that under-fitting probability and over-fitting probability are both negligible.

**Bounding the under-fitting probability:** If $\mathcal{S}_\lambda \not\supseteq \mathcal{S}_T$, we have that

$$BIC(\lambda) = -L(S, \hat{\Omega}(\lambda)) + |S_\lambda| \frac{\log n}{n} \geq -L(S, \hat{\Omega}_{S_\lambda}) + |S_\lambda| \frac{\log n}{n} \xrightarrow{p} -L_{S_\lambda},$$

where $\hat{\Omega}_S$ is the unpenalized maximum likelihood estimator based on model $S$. The convergence above is based on the uniform convergence of the empirical distribution since the class of log-likelihood functions is Glivenko-Catelli.

Similarly, with the optimal choice of $\lambda_n$ that satisfies Condition 3,

$$BIC(\lambda_n) = -L(S, \hat{\Omega}(\lambda_n)) + |S_{\lambda_n}| \frac{\log n}{n} \xrightarrow{p} -L_{S_T}.$$

Thus we have $\text{Pr}\{\inf_{\lambda \in \Lambda_-} BIC(\lambda) > BIC(\lambda_n)\} \rightarrow 1$ since $-L_{S_T} < -L_S$ when $S \not\supseteq S_T$.

**Bounding the over-fitting probability:** Now suppose $\lambda \in \Lambda_+$. Since

$$BIC(\lambda_n) = -L(S, \hat{\Omega}(\lambda_n)) + |S_{\lambda_n}| \frac{\log n}{n}$$
and

$$BIC(\lambda) = -L(S, \hat{\Omega}(\lambda)) + |S_\lambda| \frac{\log n}{n},$$

with $|S_\lambda| > |S_{\lambda_n}| = |S_T|$ (with probability 1), we will get $\Pr(\inf_{\lambda \in \Lambda_+} BIC(\lambda) > BIC(\lambda_n)) \to 1$ if it can be shown that $L(S, \hat{\Omega}(\lambda)) - L(S, \hat{\Omega}(\lambda_n)) = O_p(1/n)$.

We will use the usual notion for sample mean: $P_n L(X, \Omega) = \sum_{i=1}^n L(X_i, \Omega)/n = L(S, \Omega)$ and use $PL(X, \Omega)$ to denote the corresponding population mean for a fixed precision matrix $\Omega$. Let $G_n = \sqrt{n}(P_n - P)$ be the empirical process. We write

$$L(S, \hat{\Omega}(\lambda)) - L(S, \hat{\Omega}(\lambda_n))$$

$$\leq P_n L(X, \hat{\Omega}_{S_\lambda}) - P_n L(X, \hat{\Omega}(\lambda_n))$$

$$= P_n L(X, \hat{\Omega}_{S_\lambda}) - PL(X, \hat{\Omega}_{S_\lambda}) - \left( P_n L(X, \hat{\Omega}(\lambda_n)) - PL(X, \hat{\Omega}(\lambda_n)) \right)$$

$$+ PL(X, \hat{\Omega}_{S_\lambda}) - PL(X, \Omega_0) - \left( PL(X, \hat{\Omega}(\lambda_n)) - PL(X, \Omega_0) \right),$$

where $\Omega_0$ is the true precision matrix. Define $M_1(X) = \sqrt{n} \left( L(X, \hat{\Omega}_{S_\lambda}) - L(X, \Omega_0) \right)$, and $M_2(X) = \sqrt{n} \left( L(X, \hat{\Omega}(\lambda_n)) - L(X, \Omega_0) \right)$, then the previous display can be written as

$$L(S, \hat{\Omega}(\lambda)) - L(S, \hat{\Omega}(\lambda_n))$$

$$\leq \frac{1}{n} G_n M_1(X) - \frac{1}{n} G_n M_2(X)$$

$$+ \left( PL(X, \hat{\Omega}_{S_\lambda}) - PL(X, \Omega_0) \right) - \left( PL(X, \hat{\Omega}(\lambda_n)) - PL(X, \Omega_0) \right)$$

$$=: (I)+(II)+(III)+(IV).$$
We first bound (I) from above. Classical maximum likelihood theory tells us that \( H_n := \sqrt{n}(\hat{\Omega}_S - \Omega_0) \) is asymptotically normal (for the non-constrained elements of the matrix). Applying Lemma 19.31 in van der Vaart (1998), we have \( G_n(M_1(X) - H_n^T dL(X, \Omega_0)/d\Omega) \xrightarrow{p} 0 \) and then it is easily seen that \( G_n(M_1(X)) = O_P(1) \). Similarly, given that Fan et al. (2008) have shown that \( \hat{\Omega}(\lambda_n) \) is also asymptotically normal, we have \( G_n M_2(X) = O_P(1) \). For (III) and (IV), a simple Taylor expansion around \( \Omega \) yields both term to be of order \( O_P(1/n) \).

**ACKNOWLEDGEMENT**

This research is supported by Singapore Ministry of Education Tier 1 SUG.

**References**

Bickel, P. J. and Levina, E. (2008), “Covariance regularization by thresholding,” *Annals of Statistics*, in press.

Bishop, C. M. (2006), *Pattern recognition and machine learning*, Information science and statistics., New York, NY: Springer Science+Business Media.

Craven, P. and Wahba, G. (1979), “Smoothing noisy data with spline functions,” *Numerische Mathematik*, 31, 377–403.

Dempster, A. P. (1972), “Covariance selection,” *Biometrics*, 28, 157–175.
Dobra, A. and West, M. (2004), “Bayesian covariance selection,” Working Paper, ISDS, Duke University.

Dong, X. A. and Wahba, G. (1996), “A generalized approximate cross validation for smoothing splines with non-Gaussian data,” Statistica Sinica, 6, 675–692.

Fan, J. Q., Feng, Y., and Wu, Y. (2008), “Network exploration via the adaptive lasso and SCAD penalties,” Annals of Applied Statistics, in press.

Fan, J. Q. and Li, R. Z. (2001), “Variable selection via nonconcave penalized likelihood and its oracle properties,” Journal of the American Statistical Association, 96, 1348–1360.

Fan, J. Q. and Peng, H. (2004), “Nonconcave penalized likelihood with a diverging number of parameters,” Annals of Statistics, 32, 928–961.

Friedman, J., Hastie, T., and Tibshirani, R. (2008), “Sparse inverse covariance estimation with the graphical lasso,” Biostatistics, 9, 432–441.

Lam, C. and Fan, J. Q. (2007), “Sparsistency and rates of convergence in large covariance matrices estimation,” Arxiv online, http://www.citebase.org/abstract?id=oai:arXiv.org:0711.3933.

Li, H. Z. and Gui, J. (2006), “Gradient directed regularization for sparse Gaussian concentration graphs, with applications to inference of genetic networks,” Biostatistics, 7, 302–317.
Meinshausen, N. and Buhlmann, P. (2006), “High-dimensional graphs and variable selection with the Lasso,” *Annals of Statistics*, 34, 1436–1462.

Shao, J. (1997), “An asymptotic theory for linear model selection,” *Statistica Sinica*, 7, 221–242.

van der Vaart, A. W. (1998), *Asymptotic statistics*, Cambridge, UK ; New York: Cambridge University Press.

Wang, H., Li, B., and Leng, C. L. (2008), “Shrinkage tuning parameter selection with a diverging number of parameters,” *Journal of the Royal Statistical Society Series B-Methodological*, in press.

Wang, H., Li, R., and Tsai, C. L. (2007), “Tuning parameter selectors for the smoothly clipped absolute deviation method,” *Biometrika*, 94, 553–568.

Wang, H. S. and Leng, C. L. (2007), “Unified LASSO estimation by least squares approximation,” *Journal of the American Statistical Association*, 102, 1039–1048.

Yang, Y. H. (2005), “Can the strengths of AIC and BIC be shared? A conflict between model indentification and regression estimation,” *Biometrika*, 92, 937–950.

Yuan, M. and Lin, Y. (2007), “Model selection and estimation in the Gaussian graphical model,” *Biometrika*, 94, 19–35.

Zhao, P. and Yu, B. (2006), “On model selection consistency of Lasso,” *Journal of Machine Learning Research*, 7, 2541–2563.
Zou, H. (2006), “The adaptive lasso and its oracle properties,” *Journal of the American Statistical Association*, 101, 1418–1429.