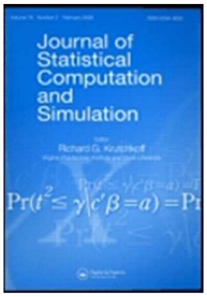


A computationally fast alternative to cross-validation in penalized Gaussian graphical models

Journal:	*Journal of Statistical Computation and Simulation*
Manuscript ID:	GSCS-2014-0129.R1
Manuscript Type:	Original Paper
Date Submitted by the Author:	28-Sep-2014
Complete List of Authors:	Vujacic, Ivan; University of Groningen, Statistics and Probability Abbruzzo, Antonino; University of Palermo, Department of Statistics Wit, Ernst; University of Groningen, Statistics and Probability
Areas of Interest:	LASSO, MODEL SELECTION, SHRINKAGE
<a href="http://www.ams.org/mathscinet/msc/msc2010.html" target="_blank">2010 Mathematics Subject Classification</a>:	62

To appear in the *Journal of Statistical Computation and Simulation*
Vol. 00, No. 00, Month 20XX, 1–14

A computationally fast alternative to cross-validation in penalized Gaussian graphical models

Ivan Vujačić[a*], Antonino Abbruzzo[b] and Ernst Wit[a]

[a] *Department of Statistics and Probability, University of Groningen, The Netherlands*;
[b] *Department of Statistics, University of Palermo, Italy*

(*Received 00 Month 20XX; final version received 00 Month 20XX*)

We study the problem of selection of regularization parameter in penalized Gaussian graphical models. When the goal is to obtain a model with good predicting power, cross validation is the gold standard. We present a new estimator of Kullback-Leibler loss in Gaussian Graphical model which provides a computationally fast alternative to cross-validation. The estimator is obtained by approximating leave-one-out-cross validation. Our approach is demonstrated on simulated data sets for various types of graphs. The proposed formula exhibits superior performance, especially in the typical small sample size scenario, compared to other available alternatives to cross validation, such as Akaike's information criterion and Generalized approximate cross validation. We also show that the estimator can be used to improve the performance of the BIC when the sample size is small.

Keywords: Gaussian graphical model; Penalized estimation; Kullback-Leibler loss; Cross-validation; Generalized approximate cross-validation; Information criteria .

AMS Subject Classification: F1.1; F4.3 **(... for example; authors are encouraged to provide two to six 2010 Mathematics Subject Classification codes)**

1. Introduction

Let $\boldsymbol{Y} = (Y_1, \ldots, Y_p)$ be a p-dimensional Gaussian random vector with zero mean and positive definite covariance matrix $\boldsymbol{\Sigma}$, i.e. $\boldsymbol{Y} \sim \mathcal{N}_p(\boldsymbol{0}, \boldsymbol{\Sigma})$. In many applications, like gene network reconstruction, estimating the precision matrix, denoted by $\boldsymbol{\Omega} = (\omega_{ij}) = \boldsymbol{\Sigma}^{-1}$ is of main interest. The element ω_{ij} in $\boldsymbol{\Omega}$ is proportional to the partial correlation between the ith and jth components of $\boldsymbol{Y}$ conditional on all others. Consequently $\omega_{ij} = 0$ if and only if Y_i and Y_j are conditionally independent given the rest of the variables in $\boldsymbol{Y}$. This gives the appealing graphical interpretation of vector $\boldsymbol{Y}$ as a Gaussian graphical model [1–4]. Vector $\boldsymbol{Y}$ can be represented by an undirected graph $\mathcal{G} = (V, E)$, where V is the set of vertices corresponding to the p coordinates of the vector $\boldsymbol{Y}$ and the edges $E = (e_{ij})_{1 \le i < j \le p}$ represent conditional dependency relationships between variables Y_i and Y_j. The edge e_{ij} between Y_i and Y_j exists if and only if $\omega_{ij} \neq 0$. Hence, for estimating the graphical structure it is not only important to estimate the parameters but also to identify the null entries in the precision matrix.

A popular method for precision matrix estimation is the penalized likelihood method [5–8]. This method is based on the optimization of an objective function which is the sum of the scaled likelihood and some penalty function of the precision matrix. Popular penalties are LASSO, SCAD and adaptive LASSO [8, 9]. The selection of the tuning parameter in this method is equivalent with the model selection of a particular graphical

[*] Corresponding author. Email: i.vujacic@rug.nl

model. The methods that have been used in the literature for selecting the regularization parameter include the Bayesian Information Criterion (BIC) [5, 10–13], the Extended Bayesian Information Criterion (EBIC) [13, 14], Stability Approach to Regularization Selection (StARS) [15], Cross-validation (CV) [8, 10, 16, 17], Generalized Approximate Cross Validation (GACV) [12] and the Aikaike's Information Criterion (AIC) [11, 12, 15].

If the aim is graph identification then the criteria BIC, EBIC and StARS are appropriate. BIC is shown to be consistent for penalized graphical models with adaptive LASSO and SCAD penalties for fixed p [12, 13]. Numerical results suggest that BIC is not consistent with the LASSO penalty [13, 14]. When also p tends to infinity EBIC is shown to be consistent for the graphical LASSO, though only for decomposable graphical models [14]. The disadvantage of EBIC is that it includes an additional parameter that needs to be tuned. [13] fix this parameter to one and show that in this case EBIC is consistent with the SCAD penalty. StARS has the property of partial sparsistency which means that when the sample size goes to infinity all the true edges will be included in the selected model [15].

On the other hand, using cross-validation (CV), generalized approximate cross-validation (GACV) and AIC will result with a model with a good predicting power. Cross-validation and AIC are both estimators of the Kullback-Leibler (KL) information [18], which under some assumptions are asymptotically equivalent [19]. GACV is also an estimator of KL since it is derived as an approximation to leave-one-out cross-validation (LOOCV) [12]. Advantage of AIC and GACV is that they are not as computationally expensive as CV.

In this paper, we propose an estimator of KL of the model defined by the estimated precision matrix. The Kullback-Leibler information or divergence [20] is also known as the entropy loss. The formula that we propose exhibits superior performance compared to its competitors AIC and GACV. As it is the case with CV, using the proposed estimator will result with the model that has good predictive power. For the graph identification problem, we show how our estimator can be used to improve the performance of the BIC when the sample size is small.

The rest of the paper is organized as follows. In section 2 we present an example which clarifies the purpose of different selection methods. In Section 3 a closed-form approximation of leave-one-out-cross validation is proposed and its derivation is given in Section 4. Section 5 covers the details of the implementation of the method, while Section 6 includes a simulation study that shows the performance of the proposed estimator. Finally, we discuss the usage of the obtained estimator to graph identification problem in Section 7. We conclude with Section 8. Appendix contains proofs and auxiliary material.

2. Prediction power VS graph structure

Let $\mathbf{\Omega}_0$ be a precision matrix that corresponds to the true non-complete graph $\mathcal{G}$ and let $\mathbf{\Omega}_\epsilon$ be the matrix obtained by adding $\epsilon > 0$ to every entry of matrix $\mathbf{\Omega}$. The matrix $\mathbf{\Omega}_\epsilon$ is positive definite since it is a sum of one positive definite matrix and one positive semi-definite matrix. Indeed, $\mathbf{\Omega}_\epsilon = \mathbf{\Omega}_0 + \boldsymbol{x}_\epsilon \boldsymbol{x}_\epsilon{}^\top$, where $\boldsymbol{x}_\epsilon = (\sqrt{\epsilon}, \ldots, \sqrt{\epsilon})^\top$ is a vector of dimension p. Hence, $\mathbf{\Omega}_\epsilon$ belongs to the class of precision matrices and it corresponds to some graph $\mathcal{G}_\epsilon$. The Kullback-Leibler divergence of $\mathcal{N}_p(\mathbf{0}, \mathbf{\Omega}_\epsilon^{-1})$ from $\mathcal{N}_p(\mathbf{0}, \mathbf{\Omega}_0^{-1})$, denoted by $\mathrm{KL}(\mathbf{\Omega}_0; \mathbf{\Omega}_\epsilon)$, is equal to

$$\mathrm{KL}(\mathbf{\Omega}_0; \mathbf{\Omega}_\epsilon) = \frac{1}{2}\{\mathrm{trace}(\mathbf{\Omega}_0^{-1}\mathbf{\Omega}_\epsilon) - \log|\mathbf{\Omega}_0^{-1}\mathbf{\Omega}_\epsilon| - p\}. \tag{1}$$

(see [21]). Since $\epsilon \to 0$ implies $\mathbf{\Omega}_\epsilon \to \mathbf{\Omega}_0$, by continuity of log determinant and trace it follows that

$$\lim_{\epsilon \downarrow 0} \mathrm{KL}(\mathbf{\Omega}_0; \mathbf{\Omega}_\epsilon) = 0.$$

However, for every $0 < \epsilon < \min_{i,j} |\omega_{ij}|$ the matrix $\mathbf{\Omega}_\epsilon$ is a matrix without zero entries and consequently the graph $\mathcal{G}_\epsilon$ is the full graph. Thus, the conclusion is that even though a matrix can be close to the precision matrix of the true distribution with respect to KL loss, the corresponding graph can be completely different from the true one.

Since CV, AIC and GACV are estimators of KL they should be used for obtaining the model with a good predictive power. For graph identification, BIC, EBIC and StARS are more appropriate, because of their graph selection consistency properties. Consequently, we treat these two problems separately. Next section we devote to a new estimator of KL and in Section 7 we show how it can be used to improve the performance of E(BIC).

3. KLCV: An approximation of leave-one-out-cross validation

In this section we introduce a closed-form approximation of leave-one-out-cross validation (LOOCV) that we call *Kullback-Leibler cross-validation* (KLCV). The reason for this terminology comes from the fact that cross-validating the log-likelihood loss provides an estimate to Kullback-Leibler divergence [20].

Suppose we have n multivariate observations of dimension p from distribution $\mathcal{N}_p(\mathbf{0}, \mathbf{\Omega}_0^{-1})$. Using the notation $\mathbf{S}_k = \boldsymbol{y}_k \boldsymbol{y}_k{}^\top$ for the empirical covariance matrix of a single observation, we have that the empirical covariance matrix is given as $\mathbf{S} = \sum_{k=1}^n \mathbf{S}_k / n$. The log-likelihood of the data is, up to an additive constant, $l(\mathbf{\Omega}) = n\{\log|\mathbf{\Omega}| - \mathrm{trace}(\mathbf{\Omega}\mathbf{S})\}/2$. When $n > p$ the precision matrix $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ can be estimated by maximizing the scaled log-likelihood function

$$\frac{2}{n} l(\mathbf{\Omega}) = \log|\mathbf{\Omega}| - \mathrm{trace}(\mathbf{\Omega}\mathbf{S}),$$

over positive definite matrices $\mathbf{\Omega}$. The global maximizer is the maximum likelihood estimator (MLE) given by $\hat{\mathbf{\Omega}} = \mathbf{S}^{-1}$. When $n \leq p$ MLE does not exist. If $n > p$ and the true precision matrix is known to be sparse, the MLE has a non-desirable property: with probability one all elements of the precision matrix are nonzero. An alternative approach which yields a sparse estimator can be obtained by maximizing

$$\hat{\mathbf{\Omega}}_\lambda = \mathrm{argmax}_{\mathbf{\Omega}} \log|\mathbf{\Omega}| - \mathrm{trace}(\mathbf{\Omega}\mathbf{S}) - \sum_{i=1}^p \sum_{j=1}^p p_{\lambda_{ij}}(|\omega_{ij}|), \tag{2}$$

over positive definitive matrices $\mathbf{\Omega}$. Here, $p_{\lambda_{ij}}$ is a penalty function and ω_{ij} is the (i,j) element of matrix $\mathbf{\Omega}$ and $\lambda_{ij} > 0$ is the corresponding regularization parameter.

Let the maximum penalized likelihood estimator (MPLE) $\hat{\mathbf{\Omega}}_\lambda$ be defined by (2) and let $\mathrm{KL}(\mathbf{\Omega}_0; \hat{\mathbf{\Omega}}_\lambda)$ be the Kullback-Leibler divergence of the model $\mathcal{N}_p(\mathbf{0}, \hat{\mathbf{\Omega}}_\lambda^{-1})$ from the true distribution $\mathcal{N}_p(\mathbf{0}, \mathbf{\Omega}_0^{-1})$. According to (1) we have that

$$\mathrm{KL}(\mathbf{\Omega}_0; \hat{\mathbf{\Omega}}_\lambda) = -\frac{1}{n} l(\hat{\mathbf{\Omega}}_\lambda) + \mathrm{bias},$$

where $l(\mathbf{\Omega}) = n\{\log|\mathbf{\Omega}| - \mathrm{trace}(\mathbf{\Omega}\mathbf{S})\}/2$ and $\mathrm{bias} = \mathrm{trace}(\hat{\mathbf{\Omega}}_\lambda(\mathbf{\Omega}_0^{-1} - \mathbf{S}))/2$. We propose an estimator of the Kullback-Leibler divergence of the model $\mathcal{N}_p(\mathbf{0}, \hat{\mathbf{\Omega}}_\lambda^{-1})$ to the true distribution

$$\mathrm{KLCV}(\lambda) = -\frac{1}{n} l(\hat{\mathbf{\Omega}}_\lambda) + \widehat{\mathrm{bias}}_{\mathrm{KLCV}}, \tag{3}$$

where

$$\widehat{\mathrm{bias}}_{\mathrm{KLCV}} = 1/n(n-1) \sum_{k=1}^{n} \mathrm{vec}\{(\hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\}^\top \mathrm{vec}[\hat{\mathbf{\Omega}}_\lambda \{(\mathbf{S} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\} \hat{\mathbf{\Omega}}_\lambda] \tag{4}$$

and $\mathbf{I}_\lambda$ is the indicator matrix, whose entry is 1 if the corresponding entry in the precision matrix $\hat{\mathbf{\Omega}}_\lambda$ is nonzero and zero if the corresponding entry in the precision matrix is zero. Here, $\circ$ is the Schur or Hadamard product of matrices and vec is the vectorization operator which transforms a matrix into a column vector obtained by stacking the columns of the matrix on top of one another.
In this paper we propose to select $\hat{\mathbf{\Omega}}_{\lambda^*}$ for that λ^* that minimizes KLCV(λ) over $\lambda > 0$. The resulting estimator will give a model with good predictive power. While for the MLE we do not need any assumptions to derive the KLCV, for the MPLE the derivation uses the assumption of the sparsistency of the estimator. An estimator is *sparsistent* if all parameters in the true precision matrix that are zero are estimated as zero with probability tending to one when sample size tends to infinity [9].

4. Derivation of the KLCV

4.1. *Derivation for MLE*

We follow the idea of [22], i.e. we introduce an approximation for LOOCV via several first order Taylor expansions. [12] uses the idea to derive GACV for MPLE in GGM, where in deriving the formula, the partial derivatives corresponding to the zero elements of the precision matrix are ignored. Here, unlike in [12], we apply the idea only for MLE estimator and therefore we avoid all technical difficulties that ignoring the derivatives entails. In the next section we extend the derived formula for MLE to MPLE. Denote the log-likelihood of observation $\boldsymbol{y}_k$ with

$$l_k(\mathbf{\Omega}) = \frac{1}{2}\{\log|\mathbf{\Omega}| - \mathrm{trace}(\mathbf{\Omega}\mathbf{S}_k)\}$$

and consider the following function of two variables

$$f(\mathbf{S}, \mathbf{\Omega}) = \frac{2}{n} l(\mathbf{\Omega}) = \log|\mathbf{\Omega}| - \mathrm{trace}(\mathbf{\Omega}\mathbf{S}).$$

With this notation we have the identity

$$\sum_{k=1}^{n} f(\mathbf{S}_k, \mathbf{\Omega}) = n f(\mathbf{S}, \mathbf{\Omega}). \tag{5}$$

Let $\hat{\mathbf{\Omega}}^{(-k)}$ be the estimator of the precision matrix defined in (2) with $\lambda_{ij} = \lambda = 0$ based on the data excluding the kth data point. The leave-one-out cross validation score (see

[18]) is defined by

$$\begin{aligned}\mathrm{LOOCV} &= -\frac{1}{n}\sum_{k=1}^{n} l_k(\hat{\mathbf{\Omega}}^{(-k)}) = -\frac{1}{2n}\sum_{k=1}^{n} f(\mathbf{S}_k, \hat{\mathbf{\Omega}}^{(-k)}) \\ &= -\frac{1}{2n}\sum_{k=1}^{n}\{f(\mathbf{S}_k, \hat{\mathbf{\Omega}}^{(-k)}) - f(\mathbf{S}_k, \hat{\mathbf{\Omega}}) + f(\mathbf{S}_k, \hat{\mathbf{\Omega}})\} \\ &\overset{(5)}{=} -\frac{1}{2} f(\mathbf{S}, \hat{\mathbf{\Omega}}) - \frac{1}{2n}\sum_{k=1}^{n}\left\{f(\mathbf{S}_k, \hat{\mathbf{\Omega}}^{(-k)}) - f(\mathbf{S}_k, \hat{\mathbf{\Omega}})\right\} \\ &\approx -\frac{1}{n} l(\hat{\mathbf{\Omega}}) - \frac{1}{2n}\sum_{k=1}^{n}\frac{\mathrm{d}f(\mathbf{S}_k, \hat{\mathbf{\Omega}})}{\mathrm{d}\mathbf{\Omega}}\mathrm{vec}(\hat{\mathbf{\Omega}}^{(-k)} - \hat{\mathbf{\Omega}}).\end{aligned}$$

Using matrix differential calculus (see the Appendix) we have $\mathrm{d}f(\mathbf{S}_k, \hat{\mathbf{\Omega}})/\mathrm{d}\mathbf{\Omega} = \{\mathrm{vec}(\hat{\mathbf{\Omega}}^{-1} - \mathbf{S}_k)\}^\top$. The term $\mathrm{vec}(\hat{\mathbf{\Omega}}^{(-k)} - \hat{\mathbf{\Omega}})$ is obtained by applying the Taylor expansion of the function $\left\{\frac{\mathrm{d}f(\mathbf{S},\mathbf{\Omega})}{\mathrm{d}\mathbf{\Omega}}\right\}^\top$ around $(\mathbf{S}, \hat{\mathbf{\Omega}})$ in the point $(\mathbf{S}^{(-k)}, \hat{\mathbf{\Omega}}^{(-k)})$. We expand the transposed term because we consider vectors as columns.

$$\mathbf{0}_{p^2} = \left\{\frac{\mathrm{d}f(\mathbf{S}^{(-k)}, \hat{\mathbf{\Omega}}^{(-k)})}{\mathrm{d}\mathbf{\Omega}}\right\}^\top \approx \left\{\frac{\mathrm{d}f(\mathbf{S}, \hat{\mathbf{\Omega}})}{\mathrm{d}\mathbf{\Omega}}\right\}^\top + \frac{\mathrm{d}^2 f(\mathbf{S}, \hat{\mathbf{\Omega}})}{\mathrm{d}\mathbf{\Omega}^2}\mathrm{vec}(\hat{\mathbf{\Omega}}^{(-k)} - \hat{\mathbf{\Omega}}) + \frac{\mathrm{d}^2 f(\mathbf{S}, \hat{\mathbf{\Omega}})}{\mathrm{d}\mathbf{\Omega}\mathrm{d}\mathbf{S}}\mathrm{vec}(\mathbf{S}^{(-k)} - \mathbf{S}),$$

where $\mathbf{0}_{p^2}$ is the column vector of zeros of dimension p^2. From here it follows that

$$\mathrm{vec}(\hat{\mathbf{\Omega}}^{(-k)} - \hat{\mathbf{\Omega}}) \approx -\left\{\frac{\mathrm{d}^2 f(\mathbf{S}, \hat{\mathbf{\Omega}})}{\mathrm{d}\mathbf{\Omega}^2}\right\}^{-1}\frac{\mathrm{d}^2 f(\mathbf{S}, \hat{\mathbf{\Omega}})}{\mathrm{d}\mathbf{\Omega}\mathrm{d}\mathbf{S}}\mathrm{vec}(\mathbf{S}^{(-k)} - \mathbf{S}).$$

We have $\mathrm{d}f(\mathbf{S}, \hat{\mathbf{\Omega}})/\mathrm{d}\mathbf{\Omega} = \{\mathrm{vec}(\hat{\mathbf{\Omega}}^{-1} - \mathbf{S})\}^\top$, so $\mathrm{d}^2 f(\mathbf{S}, \hat{\mathbf{\Omega}})/\mathrm{d}\mathbf{\Omega}\mathrm{d}\mathbf{S} = -\mathbf{I}_{p^2}$, $\mathrm{d}^2 f(\mathbf{S}, \hat{\mathbf{\Omega}})/\mathrm{d}\mathbf{\Omega}^2 = -\hat{\mathbf{\Omega}}^{-1} \otimes \hat{\mathbf{\Omega}}^{-1}$ and consequently

$$\mathrm{vec}(\hat{\mathbf{\Omega}}^{(-k)} - \hat{\mathbf{\Omega}}) \approx -(\hat{\mathbf{\Omega}} \otimes \hat{\mathbf{\Omega}})\mathrm{vec}(\mathbf{S}^{(-k)} - \mathbf{S}).$$

It follows that the approximation of LOOCV, denoted by KLCV, has the form

$$\mathrm{KLCV} = -\frac{1}{n} l(\hat{\mathbf{\Omega}}) + \frac{1}{2n}\sum_{k=1}^{n}\{\mathrm{vec}(\hat{\mathbf{\Omega}}^{-1} - \mathbf{S}_k)\}^\top(\hat{\mathbf{\Omega}} \otimes \hat{\mathbf{\Omega}})\mathrm{vec}(\mathbf{S}^{(-k)} - \mathbf{S}).$$

After simplifying the term in the sum we finally obtain

$$\mathrm{KLCV} = -\frac{1}{n} l(\hat{\mathbf{\Omega}}) + 1/2n(n-1)\sum_{k=1}^{n} T_k \qquad (6)$$

where

$$T_k = \{\mathrm{vec}(\hat{\mathbf{\Omega}}^{-1} - \mathbf{S}_k)\}^\top(\hat{\mathbf{\Omega}} \otimes \hat{\mathbf{\Omega}})\mathrm{vec}(\mathbf{S} - \mathbf{S}_k).$$

This formula is equivalent to that from (4) and we will show this in the end of the next section. Also, this formula is equivalent to the one obtained in [12] who proposed it for both, MLE and MPLE. We do not advocate using this formula for MPLE since it ignores the sparsity assumption. For this reason, we treat the case of MPLE separately in the next section. We also show that the obtained formula for the MPLE is an extension of the formula for the MLE.

4.2. *Extension to MPLE*

Before we propose the formula for the MPLE we formulate two auxiliary results.

Lemma 4.1 *Let $\mathbf{A}$ and $\mathbf{\Omega}$ be a symmetric matrices of order p. The following identity holds*

$$(\mathbf{\Omega} \otimes \mathbf{\Omega})\mathrm{vec}(\mathbf{A}) = \mathbf{M}_p(\mathbf{\Omega} \otimes \mathbf{\Omega})\mathrm{vec}(\mathbf{A}), \tag{7}$$

where $\mathbf{M}_p = 1/2(\mathbf{I}_{p^2} + \mathbf{K}_p)$, and $\mathbf{I}_{p^2}$ and $\mathbf{K}_p$ are identity matrix and commutation matrix of order p^2 respectively.

Commutation matrix $\mathbf{K}_p$ is a square matrix of dimension p^2 that has the property $\mathbf{K}_p\mathrm{vec}\mathbf{A} = (\mathrm{vec}\mathbf{A})^\top$ for any matrix $\mathbf{A}$ of dimension p.

Lemma 4.2 *Let $\mathbf{A}$ be a symmetric matrix of order p and $\boldsymbol{x}, \boldsymbol{y}$ any vectors of dimension p. Then the value of the bilinear form*

$$B(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x}^\top \mathbf{A}\boldsymbol{y},$$

when ith row (column) of the matrix $\mathbf{A}$ is set to zero is the same as the value of $B(\boldsymbol{x}, \boldsymbol{y})$ when ith entry of the vector $\boldsymbol{x}$ ($\boldsymbol{y}$) is set to zero.

The proof of Lemma 4.1 is given in the Appendix, while Lemma 2 is obtained by straightforward calculation. Aaccording to the Lemma 4.1

$$T_k = \{\mathrm{vec}(\hat{\mathbf{\Omega}}^{-1} - \mathbf{S}_k)\}^\top \mathbf{M}_p(\hat{\mathbf{\Omega}} \otimes \hat{\mathbf{\Omega}})\mathrm{vec}(\mathbf{S} - \mathbf{S}_k), \tag{8}$$

and that $2\mathbf{M}_p(\hat{\mathbf{\Omega}} \otimes \hat{\mathbf{\Omega}})$ is an estimator of the asymptotic covariance matrix of $\hat{\mathbf{\Omega}}$ [23]. To obtain the formula for the MPLE we assume standard conditions like in [9] that guarantee sparsistent estimator. These conditions imply that $\lambda \to 0$ when $n \to \infty$, so we use formula (6), derived for the MLE, as an approximation in the penalized case. By sparsistency, with probability one the zero coefficients will be estimated as zero when n tends to infinity. This means that asymptotically the covariances between zero elements and nonzero elements are equal to zero. Thus, to obtain the term T_k for the MPLE we do not only plug in the expression $\hat{\mathbf{\Omega}}_\lambda$ in formula (8), but we also set the elements of the matrix $\mathbf{M}_p(\hat{\mathbf{\Omega}}_\lambda \otimes \hat{\mathbf{\Omega}}_\lambda)$ that correspond to covariances between zero and nonzero elements to zero. According to Lemma 4.2 this is equivalent to setting the corresponding entries of vectors $\mathrm{vec}(\hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k)$ and $\mathrm{vec}(\mathbf{S}^{(-k)} - \mathbf{S})$ to zero, i.e. we define

$$T_k(\lambda) = [\mathrm{vec}\{(\hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\}]^\top \mathbf{M}_p(\hat{\mathbf{\Omega}}_\lambda \otimes \hat{\mathbf{\Omega}}_\lambda)\mathrm{vec}\{(\mathbf{S} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\},$$

where $\mathbf{I}_\lambda$ is the indicator matrix, whose entry is 1 if the corresponding entry in the precision matrix $\hat{\mathbf{\Omega}}_\lambda$ is nonzero and zero if the corresponding entry in the precision matrix is zero. The obtained formula involves matrices of order p^2, which entails high cost in terms of both, memory usage and floating-point operations. For this reason, we rewrite the formula in a way that it is computationally feasible. Following [12] we apply the Lemma 4.1 and the identity $\mathrm{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\mathrm{vec}\mathbf{B}$ and obtain

$$T_k(\lambda) = \mathrm{vec}\{(\hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\}^\top \mathrm{vec}[\hat{\mathbf{\Omega}}_\lambda\{(\mathbf{S} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\}\hat{\mathbf{\Omega}}_\lambda]. \tag{9}$$

To conclude this section, we show that the derived formula for MPLE is an extension of the corresponding formula for MLE, meaning that applying the MPLE formula on the MLE yields the same result like the corresponding MLE formula. To this aim, let $\hat{\mathbf{\Omega}}$ be maximum likelihood estimator of the precision matrix, which is the MPLE for $\lambda = 0$, i.e. $\hat{\mathbf{\Omega}} = \hat{\mathbf{\Omega}}_\lambda$, for $\lambda = 0$. Since with probability one all the elements of $\hat{\mathbf{\Omega}}$ are nonzero it follows that $\mathbf{I}_\lambda$ is the matrix with all entries equal to one. This implies that in the formula (9) we have $(\hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k) \circ \mathbf{I}_\lambda = \hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k$ and $(\mathbf{S} - \mathbf{S}_k) \circ \mathbf{I}_\lambda = \mathbf{S} - \mathbf{S}_k$, which in turn implies $T_k(\lambda) = T_k$.

5. Implementation

In this section we show how to implement formula (9) efficiently. Although the formula (9) involves vectorization and transpose operators, they can be avoided in the implementation. Indeed, for any matrices $\mathbf{X} = (x_{ij})$ and $\mathbf{Y} = (y_{ij})$ it holds $(\mathrm{vec}\mathbf{X})^\top \mathrm{vec}\mathbf{Y} = \sum_{i,j} x_{ij}y_{ij}$ so it follows that $(\mathrm{vec}\mathbf{X})^\top \mathrm{vec}\mathbf{Y}$ is just the sum of elements of the matrix $\mathbf{X} \circ \mathbf{Y}$, i.e. $(\mathrm{vec}\mathbf{X})^\top \mathrm{vec}\mathbf{Y} = \sum_{i,j}(\mathbf{X} \circ \mathbf{Y})_{ij}$. Applying this to (9) we obtain

$$T_k(\lambda) = \sum_{i,j} \left((\hat{\mathbf{\Omega}}_\lambda^{-1} - \mathbf{S}_k) \circ \mathbf{I}_\lambda \circ [\hat{\mathbf{\Omega}}_\lambda\{(\mathbf{S} - \mathbf{S}_k) \circ \mathbf{I}_\lambda\}\hat{\mathbf{\Omega}}_\lambda] \right)_{ij}.$$

In statistical programming language R, expression $\sum_{i,j}(\mathbf{X} \circ \mathbf{Y})_{ij}$ can be efficiently implemented with `sum(X*Y)`.

6. Simulation study

In this section we test the performance of the proposed formula in terms of Kullback-Leibler loss. We do this in case of the most popular LASSO penalty for two sparse hub graphs. The graphs have $p = 40$ nodes and 38 edges and $p = 100$ nodes and 95 edges. Sparsity values of these graphs are 0.049 and 0.019 respectively. The graphs are shown in Figure 1. We omit the results for other type of graphs and for the adaptive LASSO and SCAD penalties for the same combinations of n and p. The method was tested for a band graph, a random graph, a cluster graph and a scale-free graph. Our estimator exhibits superior performance in all these cases.

We compare the following estimators: the KL oracle estimator, LOOCV, the proposed KLCV estimator, and the AIC and GACV estimators. The KL oracle estimator is that $\mathbf{\Omega}_\lambda$ in the LASSO solution path that minimizes the KL loss if we knew the true matrix $\mathbf{\Omega}$. Under each model, we generated 100 simulated data sets with different combinations of p and n. We focus on scenario in which $n \le p$ which is more common in applications. For the simulations we use the `huge` package in R [24]. The results are given in Tables 1 and 2. The KLCV method is close to the KL oracle score, even for very small n. Overall, our method exhibits comparable performance to AIC and GACV in large sample size scenarios, but it clearly outperforms both when the sample size is small.

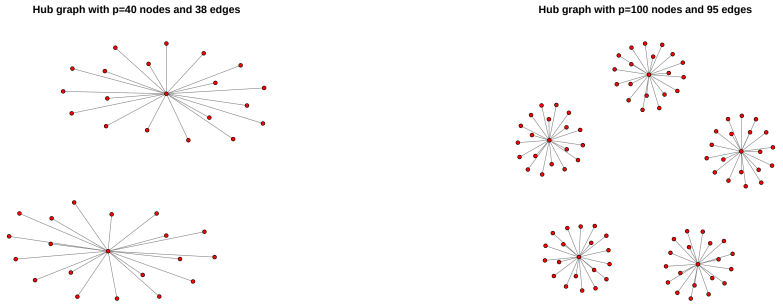

Hub graph with p=40 nodes and 38 edges
Hub graph with p=100 nodes and 95 edges

Figure 1.: Hub graphs with p=40 and $p = 100$ nodes used in the simulation study.

7. Using KLCV for graph estimation

Information criteria, such as AIC, (E)BIC, for model selection in Gaussian graphical model are based on penalizing the likelihood with a term that involves an estimator of the degrees of freedom, which is defined as

$$\mathrm{df}(\lambda) = \sum_{1 \le i < j \le p} I(\hat{\omega}_{ij,\lambda} \neq 0), \tag{10}$$

where $(\hat{\omega}_{ij,\lambda})_{1 \le i < j \le p}$ are the estimated parameters [5]. As we pointed out in section 2, unlike the AIC, the (E)BIC has a graph selection consistency property. However, in sparse data settings both the BIC and the EBIC can perform poorly. The reason is the instability of the degrees of freedom defined in (10). As [25] points out, in the high-dimensional case there is often considerable uncertainty in the number of non-zero elements in the precision matrix. To overcome this uncertainty, the authors propose to use the bootstrap method to determine the statistical accuracy and the importance of each non-zero elements identified. One can then choose only the elements with high probability of being non-zero in the precision matrix across the bootstrap samples. Here we propose an alternative, faster approach.

Recall that AIC has the form

$$\mathrm{AIC}(\lambda) = -2l(\hat{\mathbf{\Omega}}_\lambda) + 2\mathrm{df}(\lambda),$$

where $\mathrm{df}(\lambda)$ is given in (10). AIC is an estimator of KL loss scaled by $2n$. It follows that the degrees of freedom in AIC is the estimator of the bias from the KL loss scaled by $n/2$. Since in the proposed KL loss estimator we provide the estimator of the bias, we can use this estimator scaled by $n/2$ as the degrees of freedom in the BIC. In other words, we define the

$$\mathrm{BIC}_{\mathrm{KLCV}}(\lambda) = -2l(\hat{\mathbf{\Omega}}_\lambda) + \log n \mathrm{df}_{\mathrm{KLCV}}(\lambda),$$

where $\mathrm{df}(\lambda) = \frac{n}{2}\widehat{\mathrm{bias}}_{\mathrm{KLCV}}$. We compare the $\mathrm{BIC}_{\mathrm{KLCV}}$ to BIC and StARS in terms of F1 score defined as

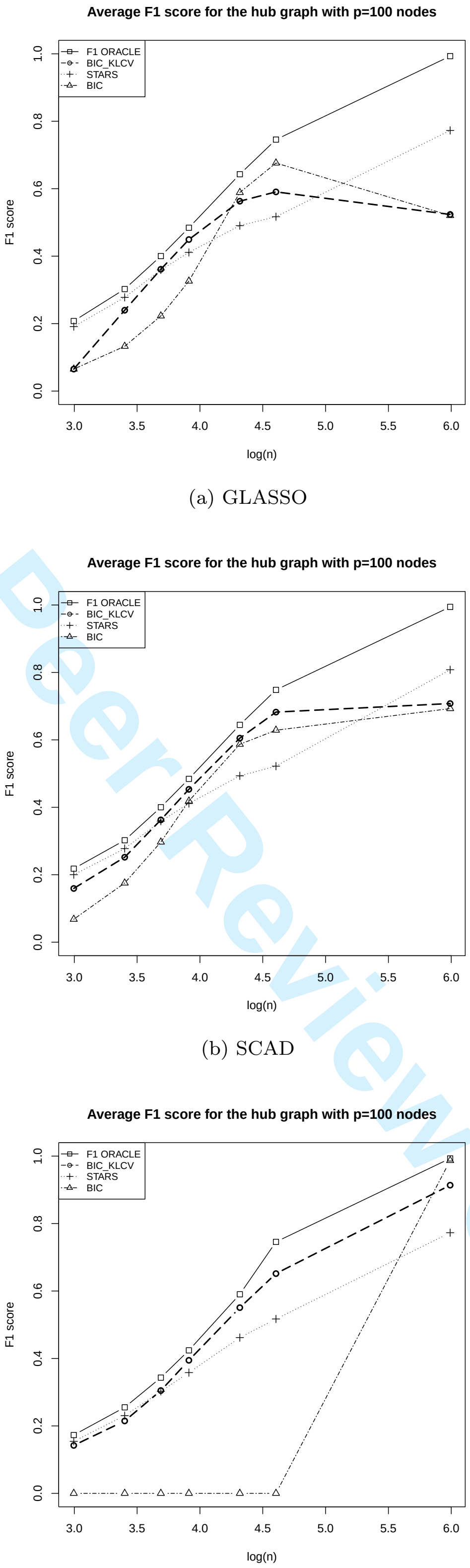
$$\mathrm{F}_1 = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FN} + \mathrm{FP}},$$

where $\mathrm{TP}, \mathrm{TN}, \mathrm{FP}, \mathrm{FN}$ are the numbers of true positives, true negatives, false positives

p=40	KL ORACLE	LOOCV	KLCV	AIC	GACV
n=8					
KL	3.68	3.86	**3.71**	6.46	26.80
	(0.27)	(0.35)	**(0.28)**	(2.12)	(1.66)
time	1.15	67.54	1.89	0.01	1.84
	(0.07)	(1.66)	(0.57)	(0.03)	(0.38)
n=12					
KL	3.29	3.38	**3.36**	6.58	18.34
	(0.26)	(0.32)	**(0.28)**	(3.54)	(1.61)
time	1.23	84.96	2.76	0.01	2.73
	(0.10)	(10.32)	(0.41)	(0.02)	(0.49)
n=16					
KL	2.93	3.00	**3.01**	6.62	13.07
	(0.26)	(0.29)	**(0.26)**	(3.07)	(1.36)
time	1.14	100.04	3.60	0.01	3.66
	(0.07)	(26.31)	(0.59)	(0.02)	(0.70)
n=20					
KL	2.67	2.70	**2.76**	6.48	10.08
	(0.23)	(0.27)	**(0.25)**	(2.50)	(1.20)
time	1.27	110.97	4.37	0.01	4.46
	(0.09)	(39.32)	(0.34)	(0.02)	(0.93)
n=30					
KL	2.18	2.22	**2.27**	4.59	5.81
	(0.23)	(0.2)	**(0.25)**	(1.11)	(0.66)
time	1.25	172.29	6.36	0.01	6.19
	(0.09)	(14.76)	(1.6)	(0.02)	(1.78)
n=40					
KL	1.91	1.91	**2.00**	3.18	4.13
	(0.19)	(0.98)	**(0.21)**	(0.66)	(0.43)
time	1.25	202.42	8.07	0.01	7.71
	(0.07)	(34.08)	(2.83)	(0.003)	(2.76)
n=100					
KL	1.00	1.04	**1.04**	1.17	1.32
	(0.10)	(0.11)	**(0.11)**	(0.16)	(0.14)
time	1.37	370.6	17.16	0.01	16.15
	(0.09)	(126.13)	(10.41)	(0.004)	(9.31)

Table 1.: Simulation results for hub graph with $p = 40$ nodes. Performance in terms of Kullback-Leibler loss and computational speed given in seconds of different estimators for different sample sizes n is showed. The results are based on 100 simulated data sets. Standard errors are shown in brackets. The best result between KLCV, AIC and GACV is boldfaced.

and false negatives. The F1 score measures the quality of a binary classifier by taking into account both true positives and negatives [26, 27]. The larger the F1 score is, the better the classifier is. The largest possible value of the F1 score is given by the F1 oracle and is evaluated by using the true matrix $\mathbf{\Omega}$. Averaged results over 100 simulations are given in Figure 2. The results suggest that $\mathrm{BIC}_{\mathrm{KLCV}}$ can improve BIC for small sample sizes and can be competitive with STARS.



(a) GLASSO

(b) SCAD

(c) ADAPTIVE GLASSO

Figure 2.: Simulations results for hub graph with $p = 100$ nodes. Average performance in terms of F1 score of different estimators for sample sizes $n = 20, 30, 40, 50, 75, 100, 400$ is showed. The results are based on 100 simulated data sets.

p=100	KL ORACLE	LOOCV	KLCV	AIC	GACV
n=20					
KL	8.06	8.09	**8.60**	12.24	28.59
	(0.37)	(0.34)	**(0.45)**	(0.28)	(19.94)
time	2.07	204.04	18.35	0.04	25.37
	(0.11)	(26.73)	(1.67)	(0.08)	(3.17)
n=30					
KL	6.87	6.81	**7.29**	10.59	32.07
	(0.34)	(0.37)	**(0.39)**	(0.41)	(2.77)
time	2.07	282.99	28.03	0.04	38.34
	(0.10)	(30.56)	(2.65)	(0.08)	(4.06)
n=40					
KL	5.92	5.93	**6.34**	9.15	22.48
	(0.30)	(0.32)	**(0.38)**	(0.59)	(1.88)
time	2.06	364.07	36.97	0.03	50.29
	(0.08)	(6.98)	(5.25)	(0.05)	(6.16)
n=50					
KL	5.24	5.27	**5.63**	7.33	16.93
	(0.27)	(0.37)	**(0.33)**	(0.81)	(1.40)
time	2.06	433.42	45.49	0.03	62.35
	(0.10)	(7.44)	(9.12)	(0.05)	(11.41)
n=75					
KL	4.08	4.11	**4.36**	4.76	9.80
	(0.27)	(0.26)	**(0.31)**	(0.71)	(0.71)
time	2.05	382.69	65.04	0.03	86.17
	(0.10)	(44.75)	(21.14)	(0.03)	(26.2)
n=100					
KL	3.34	3.33	**3.57**	3.63	6.81
	(0.19)	(0.19)	**(0.23)**	(0.48)	(0.52)
time	1.98	481.07	82.71	0.03	98.38
	(0.09)	(63.82)	(34.09)	(0.04)	(49.48)
n=400					
KL	1.13	1.15	1.20	**1.17**	1.24
	(0.07)	(0.06)	(0.08)	**(0.08)**	(0.07)
time	2.09	2092.77	353.26	0.03	479.26
	0.20	(57.03)	(186.64)	(0.04)	(279.72)

Table 2.: Simulation results for hub graph with $p = 100$ nodes. Performance in terms of Kullback-Leibler loss and computational speed given in seconds of different estimators for different sample size n is showed. The results are based on 100 simulated data sets. Standard errors are shown in brackets. The best result between KLCV, AIC and GACV is boldfaced.

8. Conclusion

In this article, we have proposed an alternative to cross-validation in penalized Gaussian graphical models. In simulation study we show that the estimator that we propose is the best available non-computational method for selecting a predictively accurate model in sparse data settings for sparse Gaussian graphical models. We also illustrated that our estimator of KL loss can be useful to for the graph selection problem.

Appendix A. Proof of Lemma 4.1

Commutation matrix $\mathbf{K}_p$ is a square matrix of dimension p^2 that has the property $\mathbf{K}_p\text{vec}\mathbf{A} = \text{vec}(\mathbf{A}^\top)$. By substituting $\mathbf{M}_p = 1/2(\mathbf{I}_{p^2} + \mathbf{K}_p)$ in the equality (7) we obtain that it is equivalent to

$$(\mathbf{\Omega} \otimes \mathbf{\Omega})\text{vec}\mathbf{A} = \mathbf{K}_p(\mathbf{\Omega} \otimes \mathbf{\Omega})\text{vec}\mathbf{A}.$$

To show the above equality, we use identities $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\text{vec}\mathbf{B}$, $\mathbf{K}_p\text{vec}\mathbf{A} = \text{vec}(\mathbf{A}^\top)$ and that $\mathbf{A}$ and $\mathbf{\Omega}$ are symmetric

$$\mathbf{K}_p\mathbf{\Omega} \otimes \mathbf{\Omega}\text{vec}\mathbf{A} = \mathbf{K}_p\text{vec}(\mathbf{\Omega A \Omega}) = \text{vec}\{(\mathbf{\Omega A \Omega})^\top\} = \text{vec}(\mathbf{\Omega A \Omega}) = \mathbf{\Omega} \otimes \mathbf{\Omega}\text{vec}\mathbf{A}.$$

□

Appendix B. Calculations of the derivatives

In the literature there are several definitions of the derivative of a function of a matrix variable. In this paper we use the definition of the derivative given in [28], which is the only natural and viable generalization of the notion of a derivative of a vector function to a derivative of a matrix function. Let $\mathbf{F}$ be a differentiable $m \times p$ real matrix function of an $n \times q$ matrix of real variables $\mathbf{X} = (x_{ij})$. The derivative (or Jacobian matrix) of $\mathbf{F}$ at $\mathbf{X}$ is the $mp \times nq$ matrix

$$\mathsf{D}\mathbf{F}(\mathbf{X}) = \frac{\partial \text{vec}\mathbf{F}(\mathbf{X})}{\partial(\text{vec}\mathbf{X})^\top},$$

where the derivative of vector valued function $\boldsymbol{f} = (f_1, \ldots, f_m)^\top$ of vector $\boldsymbol{x} = (x_1, \ldots, x_n)^\top$ is defined as the matrix $(\partial f_i(\boldsymbol{x})/\partial x_j)$. We also use the following notation for the matrix derivatives of scalar function ϕ of two matrix arguments, which have no common variables

$$\frac{\text{d}\phi(\mathbf{X},\mathbf{Y})}{\text{d}\mathbf{X}} := \mathsf{D}_\mathbf{X}\phi(\mathbf{X},\mathbf{Y}) = \frac{\partial\phi(\mathbf{X},\mathbf{Y})}{\partial(\text{vec}\mathbf{X})^\top}, \tag{B1}$$

$$\frac{\text{d}\phi(\mathbf{X},\mathbf{Y})}{\text{d}\mathbf{X}\text{d}\mathbf{Y}} := \mathsf{D}_\mathbf{X}\left\{\mathsf{D}_\mathbf{Y}\phi(\mathbf{X},\mathbf{Y})\right\}^\top, \tag{B2}$$

where $\mathsf{D}_\mathbf{X}$ and $\mathsf{D}_\mathbf{Y}$ stress that the derivatives are with respect to $\mathbf{X}$ and $\mathbf{Y}$, respectively. The transpose sign of a row vector $\mathsf{D}_\mathbf{Y}\phi(\mathbf{X},\mathbf{Y})$ in (B2) is necessary since, in this framework, the calculus is developed for column vector valued functions.

Regarding the previous comment, in matrix calculus attention should be payed to the dimension of the matrix. Taking the derivative of the matrix is not the same as taking the derivative of the transpose matrix. Indeed, for the matrix $\mathbf{X}$ the derivative of the transpose function $\mathbf{F}(\mathbf{X}) = \mathbf{X}^\top$ is not an identity matrix, but it is given by $\mathsf{D}\mathbf{F}(\mathbf{X}) = \mathbf{K}_\mathsf{p}$, where $\mathbf{K}_p$ is the commutation matrix of order p^2. For more on this subject see [28], on which our exposition is based on and which also contains the following results that we use.

Lemma B.1 *Let $\mathbf{X}$ be a square matrix of order p, $\mathbf{A}$ be a constant matrix od order p and $\mathbf{I}_{p^2}$ and $\mathbf{O}_{p^2}$ the identity and the zero matrix of order p^2, respectively. The following* *identities hold*

$$\mathsf{D}|\mathbf{X}| = |\mathbf{X}|\{\mathrm{vec}(\mathbf{X}^{-1})^\top\}^\top, \tag{B3}$$

$$\mathsf{D}\mathrm{trace}(\mathbf{A}\mathbf{X}) = \{\mathrm{vec}(\mathbf{A}^\top)\}^\top, \tag{B4}$$

$$\mathsf{D}\mathrm{vec}(\mathbf{X}) = \mathbf{I}_{p^2}, \tag{B5}$$

$$\mathsf{D}\mathbf{X}^{-1} = -(\mathbf{X}^\top)^{-1} \otimes \mathbf{X}^{-1}, \tag{B6}$$

$$\mathsf{D}\mathbf{A} = \mathbf{O}_{p^2}. \tag{B7}$$

For the derivation of the KLCV we need to show the following equalities

$$\frac{\mathrm{d}f(\mathbf{S},\boldsymbol{\Omega})}{\mathrm{d}\boldsymbol{\Omega}} = \mathrm{vec}(\boldsymbol{\Omega}^{-1} - \mathbf{S})^\top, \tag{B8}$$

$$\frac{\mathrm{d}^2 f(\mathbf{S},\boldsymbol{\Omega})}{\mathrm{d}\boldsymbol{\Omega}\mathrm{d}\mathbf{S}} = -\mathbf{I}_{p^2}, \tag{B9}$$

$$\frac{\mathrm{d}^2 f(\mathbf{S},\boldsymbol{\Omega})}{\mathrm{d}\boldsymbol{\Omega}^2} = -\boldsymbol{\Omega}^{-1} \otimes \boldsymbol{\Omega}^{-1}. \tag{B10}$$

First, we establish (B8) by using formulas for the derivatives of the determinant and the trace (B3) and (B4), the chain rule and the fact that matrices $\boldsymbol{\Omega}$ and $\mathbf{S}_k$ are symmetric. Equality (B9) follows from (B8), (B5) and (B7). Finally, (B10) follows from (B8), (B6), (B7) and the fact that $\boldsymbol{\Omega}$ is symmetric.

References

[1] Dempster AP. Covariance selection. Biometrics. 1972;:157–175.

[2] Lauritzen SL. Graphical models. Oxford University Press; 1996.

[3] Edwards D. Introduction to graphical modelling. Springer; 2000.

[4] Whittaker J. Graphical models in applied multivariate statistics. Wiley Publishing; 2009.

[5] Yuan M, Lin Y. Model selection and estimation in the gaussian graphical model. Biometrika. 2007;94(1):19–35.

[6] Banerjee O, El Ghaoui L, d'Aspremont A. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. The Journal of Machine Learning Research. 2008;9:485–516.

[7] Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics. 2008;9(3):432–441.

[8] Fan J, Feng Y, Wu Y. Network exploration via the adaptive lasso and scad penalties. The Annals of Applied Statistics. 2009;3(2):521–541.

[9] Lam C, Fan J. Sparsistency and rates of convergence in large covariance matrix estimation. Annals of statistics. 2009;37(6B):4254–4278.

[10] Schmidt M. Graphical model structure learning with l1-regularization [dissertation]. UNIVERSITY OF BRITISH COLUMBIA; 2010.

[11] Menéndez P, Kourmpetis YA, ter Braak CJ, van Eeuwijk FA. Gene regulatory networks from multifactorial perturbations using graphical lasso: application to the dream4 challenge. PloS one. 2010;5(12):e14147.

[12] Lian H. Shrinkage tuning parameter selection in precision matrices estimation. Journal of Statistical Planning and Inference. 2011;141(8):2839–2848.

[13] Gao X, Pu DQ, Wu Y, Xu H. Tuning parameter selection for penalized likelihood estimation of gaussian graphical model. Statistica Sinica. 2012;22(3):1123.

[14] Foygel R, Drton M. Extended bayesian information criteria for gaussian graphical models.

In: Lafferty J, Williams CKI, Shawe-Taylor J, Zemel R, Culotta A, editors. Advances in neural information processing systems 23; 2010. p. 604–612.

[15] Liu H, Roeder K, Wasserman L. Stability approach to regularization selection (stars) for high dimensional graphical models. In: Lafferty J, Williams CKI, Shawe-Taylor J, Zemel R, Culotta A, editors. Advances in neural information processing systems 23; 2010. p. 1432–1440.

[16] Rothman AJ, Bickel PJ, Levina E, Zhu J. Sparse permutation invariant covariance estimation. Electronic Journal of Statistics. 2008;2:494–515.

[17] Fitch AM. Computationally tractable fitting of graphical models: the cost and benefits of decomposable bayesian and penalized likelihood approaches: a thesis presented in partial fulfillment of the requirements for the degree of doctor of philosophy in statistics at massey university, albany, new zealand [dissertation]; 2012.

[18] Yanagihara H, Tonda T, Matsumoto C. Bias correction of cross-validation criterion based on kullback–leibler information under a general condition. Journal of multivariate analysis. 2006;97(9):1965–1975.

[19] Stone M. An asymptotic equivalence of choice of model by cross-validation and akaike's criterion. Journal of the Royal Statistical Society Series B (Methodological). 1977;:44–47.

[20] Kullback S, Leibler RA. On information and sufficiency. The Annals of Mathematical Statistics. 1951;22(1):79–86.

[21] Penny WD. Kullback-liebler divergences of normal, gamma, dirichlet and wishart densities. Wellcome Department of Cognitive Neurology. 2001;.

[22] Xiang D, Wahba G. A generalized approximate cross validation for smoothing splines with non-gaussian data. Statistica Sinica. 1996;6:675–692.

[23] Fried R, Vogel D. On robust gaussian graphical modelling. 2009;.

[24] Zhao T, Liu H, Roeder K, Lafferty J, Wasserman L. huge: High-dimensional undirected graph estimation. 2012; r package version 1.2.4; Available from: http://CRAN.R-project.org/package=huge.

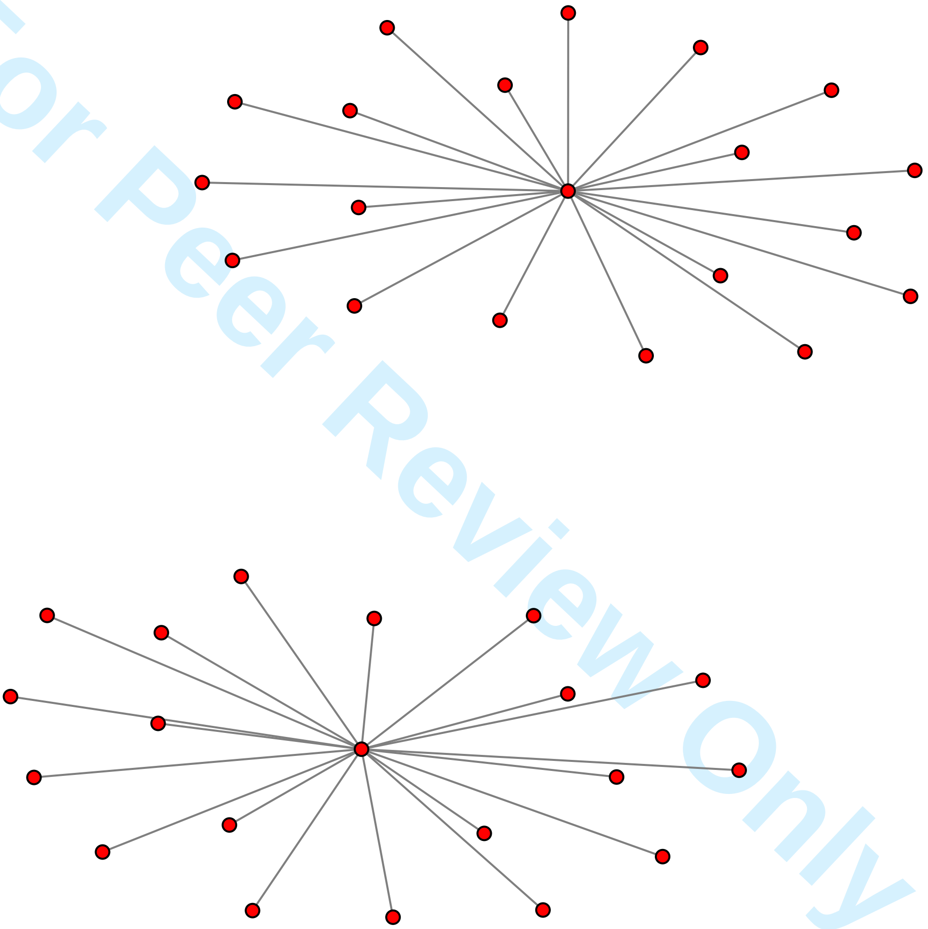
[25] Li H, Gui J. Gradient directed regularization for sparse gaussian concentration graphs, with applications to inference of genetic networks. Biostatistics. 2006;7(2):302–317.

[26] Baldi P, Brunak S, Chauvin Y, Andersen CA, Nielsen H. Assessing the accuracy of prediction algorithms for classification: an overview. Bioinformatics. 2000;16(5):412–424.

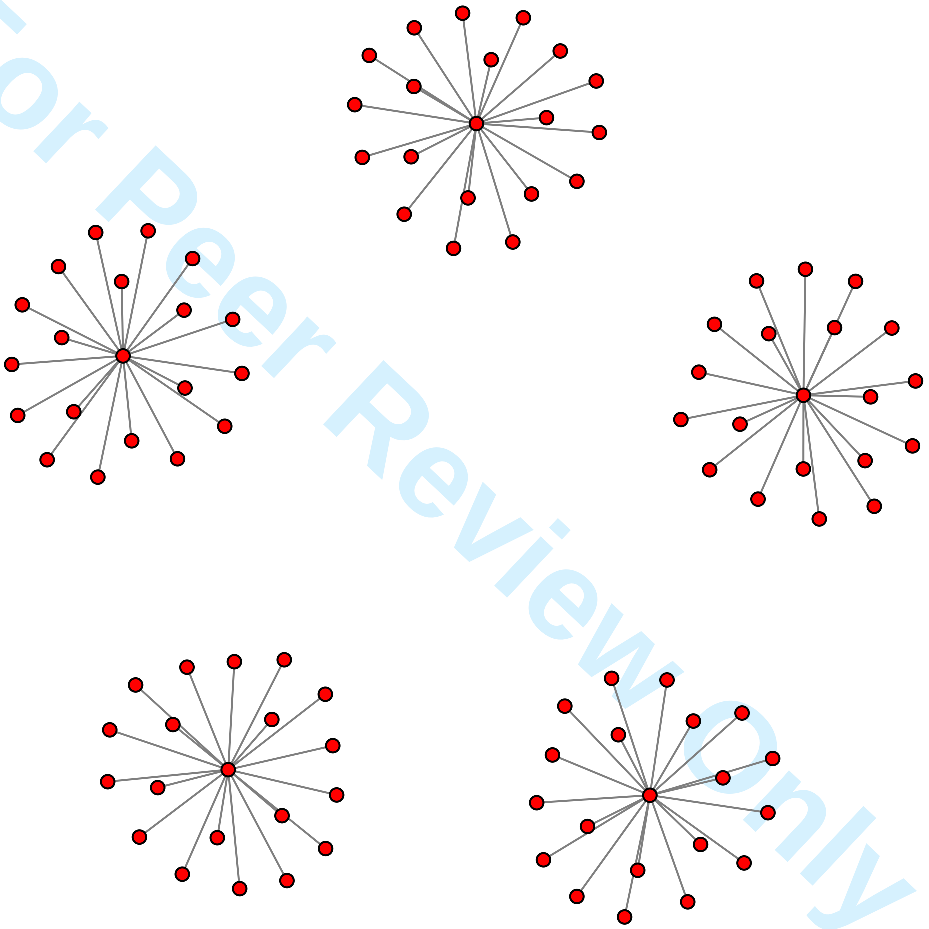
[27] Powers D. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. Journal of Machine Learning Technologies. 2011;2(1):37–63.

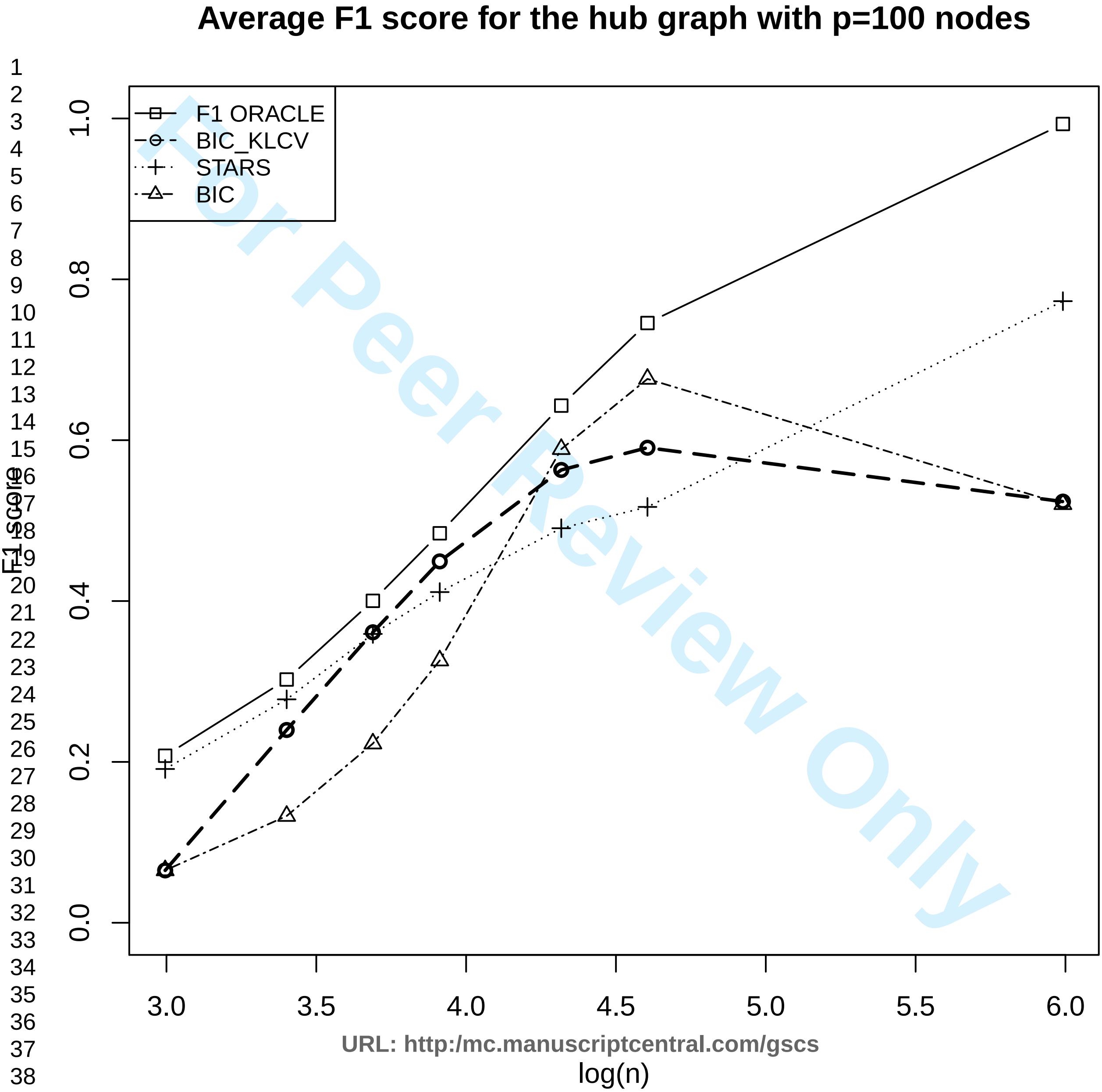
[28] Magnus JR, Neudecker H. Matrix differential calculus with applications in statistics and econometrics. 3rd ed. Wiley; 2007.

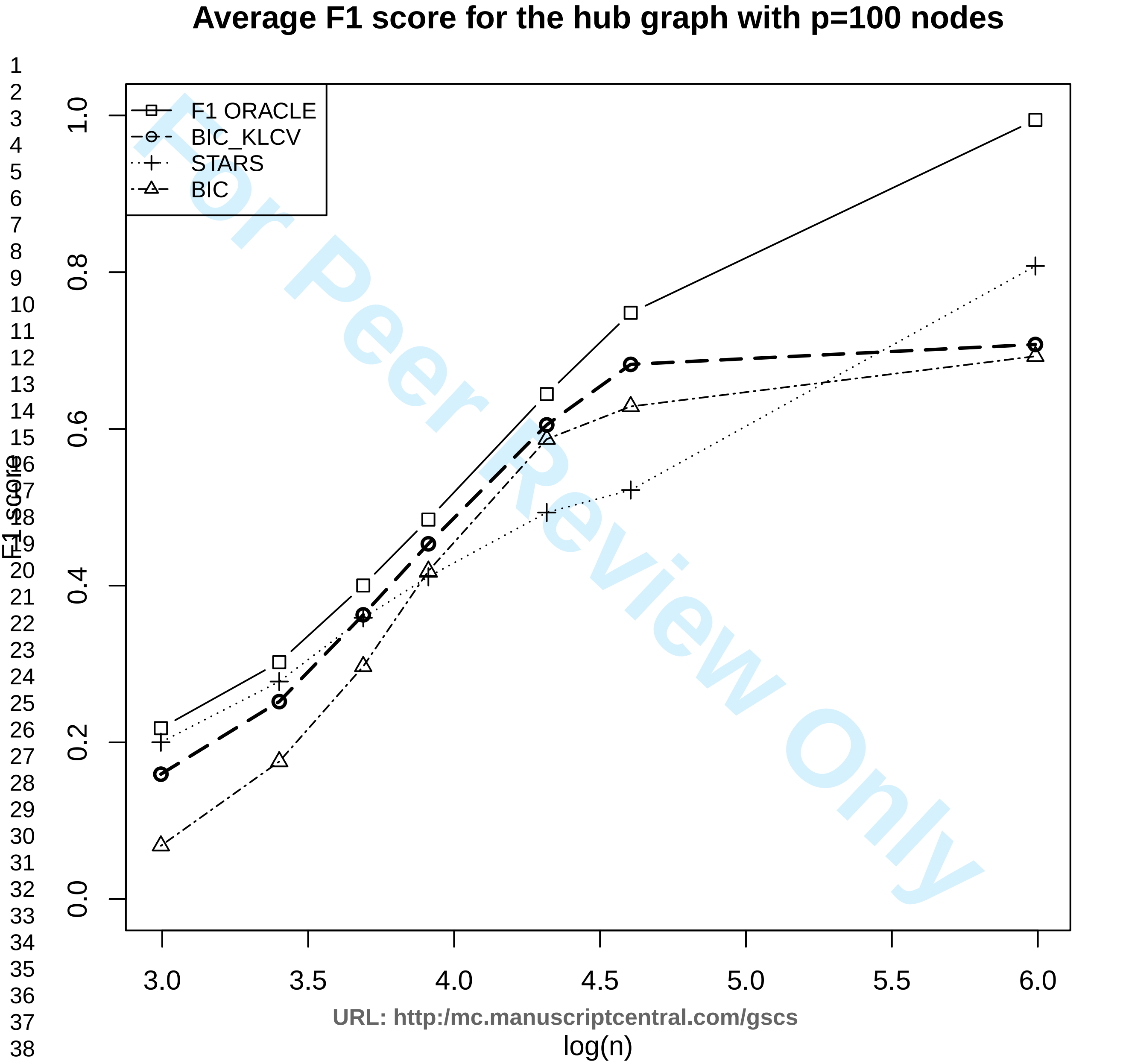
Hub graph with p=40 nodes and 38 edges

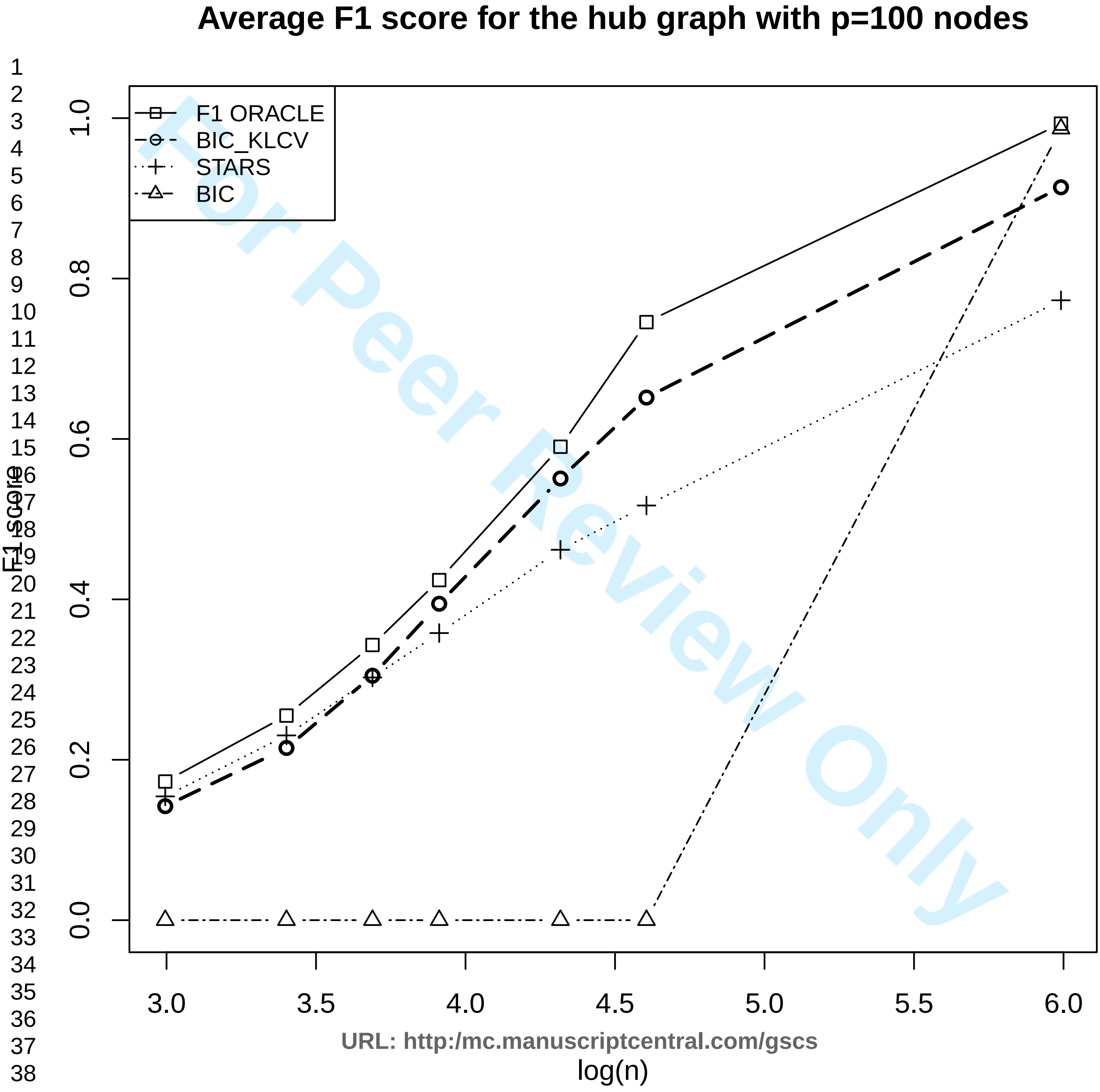


Hub graph with p=100 nodes and 95 edges




Average F1 score for the hub graph with p=100 nodes
F1 ORACLE
BIC_KLCV
STARS
BIC
F1 score
0.0
0.2
0.4
0.6
0.8
1.0
3.0
3.5
4.0
4.5
5.0
5.5
6.0
log(n)


Average F1 score for the hub graph with p=100 nodes
F1 ORACLE
BIC_KLCV
STARS
BIC
F1 score
0.0
0.2
0.4
0.6
0.8
1.0
3.0
3.5
4.0
4.5
5.0
5.5
6.0
log(n)


Average F1 score for the hub graph with p=100 nodes
F1 ORACLE
BIC_KLCV
STARS
BIC
F1 score
1.0
0.8
0.6
0.4
0.2
0.0
3.0
3.5
4.0
4.5
5.0
5.5
6.0
log(n)