Robust methods for inferring sparse network structures

Veronica Vinciotti[a,*], Hussein Hashem[a,b]

[a]*School of Information Systems, Computing and Mathematics, Brunel University, UK*
[b]*Department of Mathematics, Faculty of Science, University of Duhok, Iraq.*

Abstract

Networks appear in many fields, from finance to medicine, engineering, biology and social science. They often comprise of a very large number of entities, the nodes, and the interest lies in inferring the interactions between these entities, the edges, from relatively limited data. If the underlying network of interactions is sparse, two main statistical approaches are used to retrieve such a structure: covariance modeling approaches with a penalty constraint that encourages sparsity of the network, and nodewise regression approaches with sparse regression methods applied at each node. In the presence of outliers or departures from normality, robust approaches have been developed which relax the assumption of normality. Robust covariance modeling approaches are reviewed and compared with novel nodewise approaches where robust methods are used at each node. For low-dimensional problems, classical deviance tests are also included and compared with penalised likelihood approaches. Overall, copula approaches are found to perform best: they are comparable to the other methods under an assumption of normality or mild departures from this, but they are superior to the other methods when the assumption of normality is strongly violated.

Keywords: Penalised inference, covariance graphical models, robust regression, regularised regression, copula

1. Background

Interactions between entities of a system, such as a biological system or a social or telecommunication network, are graphically represented by links amongst

*Department of Mathematics, 208 John Crank Building, UB83PH Middlesex, UK; veronica.vinciotti@brunel.ac.uk; tel: +44 (0)1895267469; fax: +44 (0)1895 269732.

a set of nodes. In statistics, this naturally points to the use of graphical models. Graphical models are in fact defined by a set of vertices, or nodes, and a set of edges, or links between the nodes. The nodes correspond to p random variables, which in a multivariate framework can be represented by the vector $Y = (Y^{(1)}, \ldots, Y^{(p)})$. A Gaussian graphical model makes the assumption that the vector Y follows a multivariate Gaussian distribution, so

$$Y \sim N(\mu, \Sigma),$$

with mean $\mu = (\mu_1, \ldots, \mu_p)$ and variance-covariance matrix $\Sigma = (\sigma_{ij})_{ij}$. Of particular importance is the inverse of the variance-covariance matrix, also called precision or concentration matrix, which is usually denoted by

$$\Theta = \Sigma^{-1} = (\theta_{ij})_{ij}.$$

This matrix holds a special role in Gaussian graphical models: in fact, whereas zeros in the covariance matrix Σ are equivalent, under the Gaussian assumption, to marginal independence between the corresponding variables, zeros in the precision matrix correspond to conditional independence between the corresponding variables, i.e. the absence of an edge in the corresponding graph. Thus inferring the network of interactions can be recasted into the problem of estimating the precision matrix Θ and extracting its zero structure.

Given that the conditional distribution of every node given all other nodes reflects the conditional independence structure of the graph, an alternative and equivalent representation of a graphical model is by viewing it as a set of regression functions for each node against all remaining nodes. The idea goes back to Meinshausen and Bühlmann (2006). Let us write the joint multivariate Gaussian vector Y as $Y = (Y^{(-p)}, Y^{(p)})$, where $Y^{(-p)} = (Y^{(1)}, \ldots, Y^{(p-1)})$ is the vector of all nodes except for the last one. Then by partitioning the mean μ and covariance Σ as

$$\mu = \begin{pmatrix} \mu_{-p} \\ \mu_p \end{pmatrix}, \; \Sigma = \begin{pmatrix} \Sigma_{-p,-p} & \sigma_{-p,p} \\ \sigma^t_{-p,p} & \sigma_{p,p,} \end{pmatrix},$$

we can write the conditional distribution of node $Y^{(p)}$ on all other nodes as

$$Y^{(p)} | Y^{(-p)} = y \sim N\left(\mu_p + (y - \mu_{-p})^t \Sigma^{-1}_{-p,-p} \sigma_{-p,p}, \sigma_{p,p} - \sigma^t_{-p,p} \Sigma^{-1}_{-p,-p} \sigma_{-p,p}\right).$$

So the regression coefficients $\beta = \Sigma^{-1}_{-p,-p} \sigma_{-p,p}$ determine the conditional independence structure. If $\beta_j = 0$, then $Y^{(p)}$ and $Y^{(j)}$ are conditionally independent

given the rest. Furthermore, one can show that the coefficients β can be written in terms of the precision matrix as

$$\beta = -\theta_{-p,p}/\theta_{p,p},$$

and therefore knowledge about the regression coefficients β is equivalent to knowledge about the structure of the precision matrix.

The problem of estimating the structure of a graph has proven to be particularly challenging for high-dimensional cases, where the number of nodes p is very large, easily in the range of thousands, and the number of observations n is relatively small, the familiar $n << p$ case. In this case, maximum likelihood estimation does not provide a good estimate of the covariance matrix and the inverse of the sample covariance matrix does not exist, so alternatives have been proposed for the detection of a sparse representation of the interactions between the nodes. Here the assumption is that the underlying true network of interactions is sparse, which is often believed to be the case. Two main approaches have been developed for this purpose. The first approach is to estimate the precision matrix within a penalised likelihood approach where a penalty is chosen to encourage sparsity of the network, i.e. many zeros in the precision matrix. The second approach is to use sparse regression methods, such as lasso, for each node of the graph. Here the penalty is imposed on the regression coefficients for each node. Both approaches and their relative merits are described extensively by Bühlmann and van de Geer (2011).

Most of the methods available in the literature for estimating sparse undirected graphs in high dimensional problems rely on the assumption of normality. However, many real applications show departures from normality, often as a result of data contamination and the presence of outliers. A small number of methods have been developed to overcome this limitation in the context of penalised likelihood estimation for high-dimensional problems (Finegold and Drton, 2011; Liu et al., 2009). Robust estimators of the covariance matrix and adjusted statistical tests have also been developed, but they are limited to low-dimensional problems with $n < p$ (Miyamura and Kano, 2006; Gottard and Pacillo, 2010; Vogel and Fried, 2011). All these methods lead to a robust estimation of a partial correlation graph. In this context, we remark that, in contrast to the Gaussian case, a zero partial correlation cannot in general be interpreted as conditional independence of the corresponding variables. Rather, in this case, it should be treated as linear independence, conditioning on all the other variables (Vogel and Fried, 2011).

In this paper, inspired by the nodewise lasso approach of Meinshausen and Bühlmann (2006), we consider novel nodewise approaches which use robust sparse

regression methods at each node. In particular, we consider the use of regression methods where the coefficients are estimated by minimizing a Huber or Least Absolute Deviation (LAD) loss function together with an L_1 penalty for sparsity, e.g. Lambert-Lacroix and Zwald (2011). In Section 2, we describe these methods in detail. In Section 3, we present an extensive simulation study: firstly, we compare a number of robust regression methods and, secondly, we assess their performance within a nodewise graphical model approach, in comparison with robust covariance estimation methods. For low-dimensional cases, we consider also adjusted deviance tests. Finally, we present a real application from the field of biology and draw some final conclusions.

2. Methods

2.1. Robust methods for penalised covariance estimation

A sparse estimate of the precision matrix Θ can be obtained by imposing the L_1-penalty constraint on the entries of the precision matrix. This results in the optimization

$$\max_{||\Theta||_1 \leq \rho} \left[\log|\Theta| - \text{Trace}(S\Theta)\right],$$

where $||\Theta||_1 = \sum_{i,j} |\theta_{ij}|$, S is the sample covariance matrix, and ρ is a non-negative tuning parameter. As both the negative log-likelihood and the region defined by the constraint are convex in Θ, one can equivalently work with the Lagrangian dual form, resulting in the penalised likelihood optimization

$$\max_{\Theta} \left[\log|\Theta| - \text{Trace}(S\Theta) - \lambda||\Theta||_1\right],$$

with λ the non-negative Lagrange multiplier. When $\lambda = 0$, the optimal solution corresponds to the maximum likelihood; the larger the value of λ the sparser the solution, so the larger the number of zero elements in the precision matrix Θ, but the lower the associated likelihood. Hence, this optimization problem allows to obtain a sparse estimate of the precision matrix. Furthermore, the optimal solution has the property of being symmetric, which is a requirement of variance-covariance and precision matrices, and always invertible when $\lambda > 0$ (Banerjee et al., 2008). Friedman et al. (2008) provide an efficient optimization procedure for this problem, by maximising the penalised log-likelihood iteratively for each node and, at each step, by re-writing the problem into an equivalent lasso regression problem. The latter is estimated efficiently using coordinate descent methods.

The method is implemented in the `glasso` R package and has been successfully used in many applications due to its efficiency (Witten et al., 2011).

Two significant extensions of this approach have been developed in order to model data which show departures from normality. In Finegold and Drton (2011), an approach is developed based on the assumption of a multivariate t-distribution. This approach uses the fact that given a normal multivariate variable $X \sim N(0,\Psi)$ and an independent Gamma random variable $\tau \sim \Gamma(\nu/2,\nu/2)$, the variable $Y = \mu + X/\sqrt{\tau}$ has a multivariate t-distribution with ν degrees of freedom, $t_{p,\nu}(\mu,\Psi)$, and the inverse of Ψ provides the structure of the corresponding graph. Given this result, the authors devise an EM algorithm, which makes use of the efficient `glasso` approach. In Liu et al. (2009), a semiparametric Gaussian copula is presented, which allows the transformation of non-normal data to normal data, on which a `glasso` approach can then be used. In particular, the multivariate variable Y is transformed into a normally distributed variable $Z = f(Y) = (f_1(Y^{(1)}),\ldots,f_p(Y^{(p)}))$ using a Gaussian copula transformation with parameters μ and Σ, with Σ^{-1} giving the partial correlation graph. In order to estimate Σ^{-1}, Liu et al. (2009) show that $f_j(y) = \mu_j + \sigma_j\Phi^{-1}(F_j(y))$, with Φ the cumulative distribution function (cdf) of a $N(0,1)$ distribution, $F_j(y)$ the cdf of $Y^{(j)}$, and μ_j and σ_j the mean and standard deviation of $Y^{(j)}$, respectively. The cdf $F_j(y)$ is estimated using a robust truncation of the empirical distributions. In particular, if $\hat{F}_j(y)$ is the empirical cdf, then the truncated cdf is defined by

$$\tilde{F}_j(y) = \begin{cases} \delta_n & \text{if } \hat{F}_j(y) < \delta_n \\ \hat{F}_j(y) & \text{if } \delta_n < \hat{F}_j(y) < 1-\delta_n \\ 1-\delta_n & \text{if } \hat{F}_j(y) > 1-\delta_n, \end{cases}$$

with δ_n chosen as $\delta_n = \dfrac{1}{4n^{1/4}\sqrt{\pi \log n}}$ (Liu et al., 2009). From this, $\tilde{f}_j(y) = \hat{\mu}_j + \hat{\sigma}_j\Phi^{-1}(\tilde{F}_j(y))$, with $\hat{\mu}_j$ and $\hat{\sigma}_j$ chosen as the sample mean and sample standard deviation of $Y^{(j)}$, respectively. Finally, a standard `glasso` is used on the transformed variables $(\tilde{f}_1(Y^{(1)}),\ldots,\tilde{f}_p(Y^{(p)}))$ to return an estimate of Σ^{-1}.

2.2. Robust nodewise regression methods

An alternative to the global approaches above is to estimate the precision matrix by regressing each node against the remaining nodes. Here sparsity of the precision matrix is obtained by imposing a penalty on the size of the regression coefficients. Meinshausen and Bühlmann (2006) were the first to suggest this approach in the context of Gaussian graphical models: in order to obtain zeros in the

precision matrix, they suggest using an L_1 penalty, that is a lasso regression for each node. So for each node $Y^{(i)}$, the regression coefficients $\beta^{(i)}$ are estimated by minimising the loss function

$$\sum_{j=1}^{n}(y_{ij}-\alpha_i-x_j\beta^{(i)})^2+\lambda\sum_{t=1}^{p}|\beta_t^{(i)}|, \tag{1}$$

where n is the sample size, $\beta^{(i)}$ is the $p-1$ dimensional vector of regression coefficients for node i and x_j is the $n\times(p-1)$ matrix of observations on the nodes $Y^{(j)}$ for $j\neq i$. This method is very efficient and it works well in practice (Bühlmann and van de Geer, 2011), but it has the clear drawback that the resulting precision matrix is not necessarily symmetric: it can happen that β_{ij} is found to be zero when predicting $Y^{(j)}$ from the rest, but β_{ji} is not zero when $Y^{(i)}$ is predicted from the rest. This is expected with regression models but it is not true for conditional independence statements, which instead are symmetric. Meinshausen and Bühlmann (2006) suggest two possible ways of overcoming this problem: one can either use an AND rule, that is a link is included if both associated regression coefficients are non-zero, or an OR rule, where it is enough that one of the two coefficients is non-zero for allowing an edge. Obviously an AND rule results in a sparser network than an OR rule and so it is preferred if one is interested in a very sparse solution. One other possible problem with the lasso approach is that by imposing sparsity on each node independently, one reduces the chances of obtaining hubs in the resulting network. These are expected in some applications such as biological networks.

The nodewise lasso approach has been extended to an adaptive version, which is implemented in the R package `parcor` (Krämer et al., 2009): here adaptive lasso is used at each node, by minimising

$$\sum_{j=1}^{n}(y_{ij}-\alpha_i-x_j\beta^{(i)})^2+\lambda\sum_{t=1}^{p}w_t|\beta_t^{(i)}|, \tag{2}$$

with the weights defined by $w_t=\dfrac{1}{|\beta_t^{(i),\text{lasso}}|}$ and $\beta^{(i),\text{lasso}}$ taken as the lasso solution at node i. This approach leads to a precision matrix which is at least as sparse as the nodewise lasso solution, since it corresponds to a weighted lasso regression on the variables with a non-zero lasso coefficient.

In this paper, we take this approach further and exploit the use of alternative regression methods at each node. With a view to achieving robustness against

departure from normality, we use robust regression methods at each node, while preserving the L_1 penalty for sparseness. A number of approaches have been developed in this context. Among these, the two most popular approaches replace the quadratic loss in equations (1) and (2) with the Least Absolute Deviation (LAD) and Huber losses, respectively. Recent methods have appeared which suggest alternative losses, such as L_1 and L_2 combined losses (Bradic and Fan, 2011) or a weighted LAD loss (Arslan, 2012).

More in detail, Li and Zhu (2008) introduce L_1-norm regularized quantile regression. In a graphical modeling context, we propose to estimate a sparse median regression for each node, by minimizing the loss function

$$\sum_{j=1}^{n} |y_{ij} - \alpha_i - x_j\beta^{(i)}| + \lambda \sum_{t=1}^{p} |\beta_t^{(i)}|, \tag{3}$$

that is replacing the quadratic loss in equation (1) by the absolute values. We will also consider the adaptive version of the LAD-lasso regression method above, which has been developed by Wu and Liu (2009) and Xu and Ying (2010) using as initial weights the LAD solution. A second approach is developed by Rosset and Zhu (2007), where the Huber loss is considered in place of the quadratic or LAD loss. In the graphical modeling context, this corresponds to estimating the regression coefficients for each node by minimizing the loss function

$$\sum_{j=1}^{n} L_H(y_{ij}, x_j, \alpha_i, \beta^{(i)}) + \lambda \sum_{t=1}^{p} |\beta_t^{(i)}|, \tag{4}$$

with the loss L_H defined in terms of the Huber proposal function (Huber, 1981) by

$$L_H(y_{ij}, x_j, \alpha_i, \beta^{(i)}) = \begin{cases} 2M|y_{ij} - \alpha_i - x_j\beta^{(i)}| - M^2 & |y_{ij} - \alpha_i - x_j\beta^{(i)}| > M \\ (y_{ij} - \alpha_i - x_j\beta^{(i)})^2 & |y_{ij} - \alpha_i - x_j\beta^{(i)}| \leq M. \end{cases}$$

The L_H definition shows how the loss is quadratic for small residuals but it becomes linear for large residuals, thus penalising outliers. This method has been used for regression problems in a number of applications and has shown robustness against outliers. The constant M depends on the level of noise and outliers in the data and is often set to the value $M = 1.345$, which has been shown to perform well in real applications. An adaptive version of the Huber lasso regression has been recently proposed by Lambert-Lacroix and Zwald (2011), with the initial weights defined by the Huber lasso solution.

In the next section, we first show a comparison of the regularized and robust regression methods on simulated high-dimensional data. We will then use these methods within a nodewise approach for the inference of network structure and compare them with global penalised covariance approaches. Finally, we will compare penalised likelihood approaches with more traditional model selection approaches for graphical models.

3. Results and Discussion

3.1. Comparison of robust regression methods on simulated data

In this section, we compare regularized regression methods in a high dimensional setting where the number of variables $p = 100$ and the number of observations $n = 50$. We use a classical simulation setting, e.g. (Bradic and Fan, 2011), where $y = \alpha + x\beta + \varepsilon$, with $\alpha = 0$ and $\beta = (3, 1.5, 0, 0, 2, 0, \ldots, 0)$. We draw the independent variables x from a multivariate normal distribution, $N(0, \Sigma_x)$, with $(\Sigma_x)_{i,j} = r^{|i-j|}$. For the error ε, we choose a range of distributions in order to test the robustness of the methods to departures from normality. In particular, we consider the following cases: $\varepsilon \sim N(0,1)$, DoubleExponential (DE), t_3, Gamma(3,1) and Mixture. We design a mixture distribution with large outliers, similar to Lambert-Lacroix and Zwald (2011), by drawing 90% of the data from a $N(0,1)$ distribution and 10% from a $N(0,1000)$ distribution. Under all these cases, we compare the regularized regression methods described in the previous section, namely lasso (Tibshirani, 1996), LAD (Li and Zhu, 2008) and Huber lasso (Rosset and Zhu, 2007), with their adaptive versions (Xu and Ying, 2010; Lambert-Lacroix and Zwald, 2011). For lasso we use the R package `lars`, for LAD and Huber lasso we use the R implementations provided by Li and Zhu (2008) and Rosset and Zhu (2007), respectively, for adaptive lasso we adapt some of the functions in the `parcor` R package and we code in a similar way the adaptive LAD and adaptive Huber lasso methods. For the adaptive versions of the methods, we define the weights using the corresponding non-adaptive lasso versions with a penalty parameter chosen to optimize a BIC criterion. As for the main penalty parameter, we fix this to the parameter that selects exactly three non-zero coefficients, for each of the six methods. In this way, all methods can be compared at the same level of sparseness and the true positives can be directly compared.

Figure 1 reports the results of the simulation. We consider both the case of low correlation ($r = 0.5$) and that of high correlation ($r = 0.95$) of the predictors. The top panels report the median model error over 500 iterations (similar results for the mean error), with the model error computed by $(\hat{\beta} - \beta)^t S_x (\hat{\beta} - \beta)^t$, where $\hat{\beta}$ are

the estimated parameters and S_x the sample covariance. The bottom panels report the true positives, that is the number of correctly classified non-zero coefficients. Here three corresponds to the case of all non-zero coefficients being correctly detected. The results support existing knowledge about the performance of the methods: lasso does not perform well when the predictors are highly correlated, the adaptive methods tend to outperform their non-adaptive versions, particularly for the adaptive LAD lasso method, and the robust methods generally outperform the non-robust ones as departures from normality increases. This is particularly evident for the case of the mixture model simulation, which has a severe departure from normality.

For the results in Figure 1, we fixed the value of the penalty parameter λ such that exactly three non-zero coefficients are selected. The choice of the penalty parameter is in general the crucial question when applying regularized methods, particularly in a high-dimensional setting. This is not the main focus of this paper, as long as a consistent approach is chosen for all the models compared. However, in the context of non-normal data, there is also a question about the possible sensitivity of the penalty parameter to outliers and departures from normality. Figure 2 shows the degree of sparsity achieved at different penalization levels, for lasso, LAD and Huber regression methods. For each method, the x-axis reports the L_1 norm of the β coefficients along the path of solutions, divided by the L_1 norm of the final solution on the path. The y-axis reports the average number of non-zero coefficients, over 500 iterations, for a grid of fraction values between 0 and 1. On the left plot, we generate data from a normal distribution with low correlation of the predictors, whereas on the right plot we generate data from a mixture distribution with high correlation of the predictors. The plots show how the methods achieve a similar trade-off between sparsity and penalization on normal data, whereas lasso generally returns sparser solutions than the robust methods on non-normal data.

3.2. Robust covariance and nodewise regression methods on simulated networks

In this section, we investigate the use of robust lasso methods in a nodewise approach for high-dimensional graphical modeling, and compare their performance with traditional nodewise methods and robust covariance modeling approaches. We use a simulation setup similar to Finegold and Drton (2011), where we simulate p covariates from either a multivariate normal $N(0, \Theta^{-1})$ or a multivariate t-distribution, $t_{p,3}(0, \Theta^{-1})$. In defining the concentration matrix Θ, we allow only a certain percentage of non-zero edges, which for our simulations we set to 10%, and randomly assign edges. As in Finegold and Drton (2011), we also consider

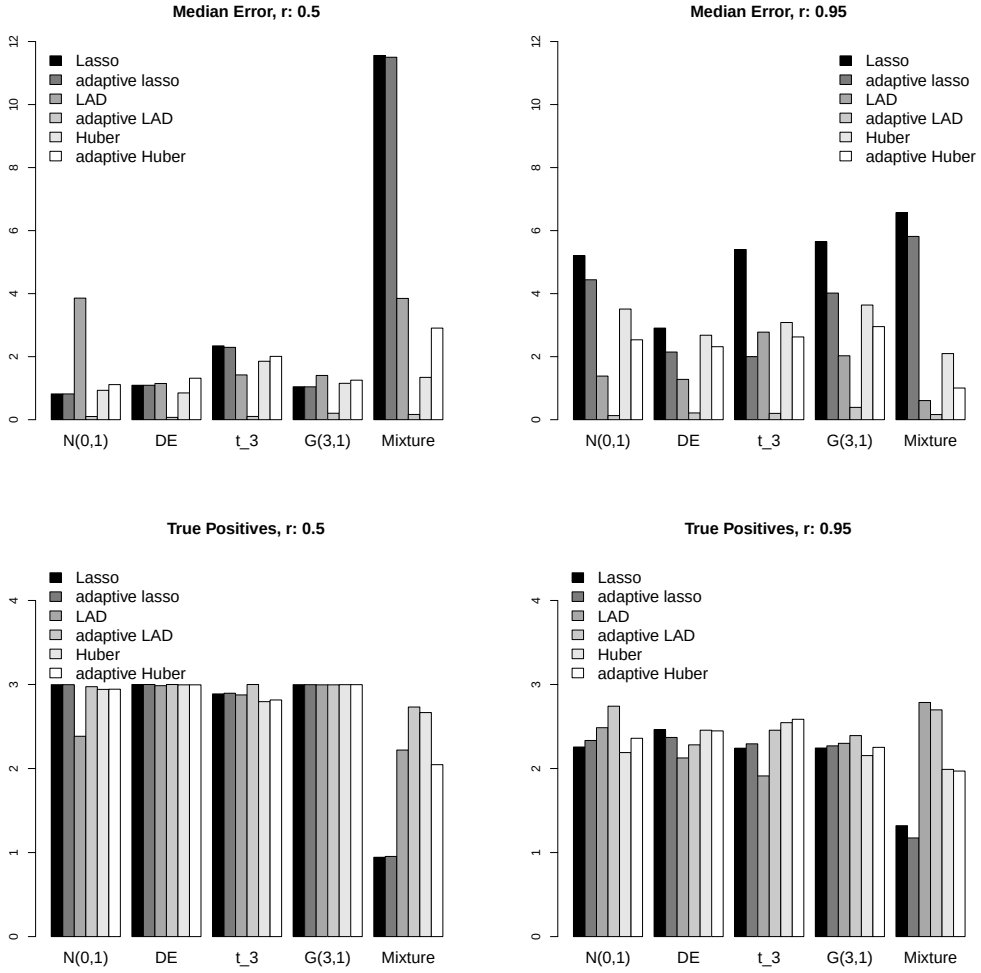

Figure 1: Comparison of regression lasso methods under different error distributions, for low (left) and high (right) correlated predictors. The top panels plot the median model error over 500 iterations and the bottom panels the average true positives.

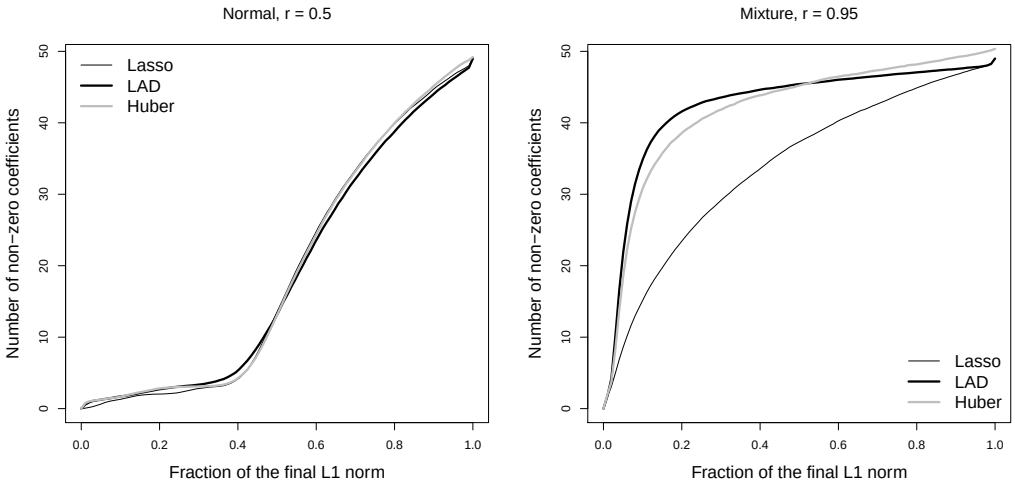

Figure 2: Sparsity versus penalization for lasso, LAD and Huber regression methods, for normal data with low correlation (left) and mixture data with high correlation (right).

a third case where we contaminate multivariate normal data. In particular, we simulate data from a multivariate normal distribution and contaminate a certain percentage of the data for each predictor using a $N(\mu, 0.2)$ distribution, with μ chosen to be 2.5 times the largest diagonal element of Θ^{-1}. This is a case of mild contamination, which would generally not be picked up manually.

Figure 3 shows ROC curves for four different cases, averaged over 500 iterations. Here, we simulate data with $p = 20$ and $n = 50$. We use the R package `huge` for the nodewise lasso, glasso and copula methods, whereas we use the code provided by Finegold and Drton (2011) for the tLasso method. The ROC curves show the true positive rate (percentage of correctly detected edges) versus the false positive rate (percentage of missing edges incorrectly classified as edges) as the penalty λ varies over a carefully selected grid of values. For the adaptive methods, we use the same penalty λ at both steps of the procedure, i.e. for defining the weights and for estimating the final regression coefficients. This is so that a wide range of sparseness of the solution can be obtained. In particular, it also means that the adaptive lasso nodewise approach uses the lasso nodewise solution to set the initial weights for each node. Finally, we use the OR rule for nodewise approaches, so an edge is included if at least one of the two corresponding β coefficients is non-zero. The R code for the nodewise adaptive methods

is available at http://people.brunel.ac.uk/~mastvvv/Software. The plots in Figure 3 show how the methods perform all similarly well for the case of data generated from a Normal distribution (top left), whereas the tLasso method outperforms the other methods when the data is generate by a multivariate t-distribution (top right), but even in this case it is very closely followed by the copula approach. The two plots at the bottom show the case of mild contamination (2% of the data, bottom left), versus a more severe case (10% of the data, bottom right). It is clear how the robust methods perform better than the non-robust ones as the contamination levels increase, supporting the results in the previous section. Looking at all four cases, the copula approach seems to be the clear favorite, as it performs well in all cases and it clearly outperforms the other methods for high departures from normality. This strengthens the results of Liu et al. (2009) where the approach was compared only against glasso. The adaptive robust methods outperform the non-robust methods, such as lasso and glasso, only for high levels of contamination, which seems to support the results found in the previous section. However, the improvement is quite small. Generally, one drawback on the use of LAD and Huber-regression methods for graphical modelling is that these methods are not robust against outliers in the explanatory variables and protect only against outliers in the response variable.

The power of all these methods is on the high dimensional setting. In this second simulation, we evaluate the performance of the methods as dimensionality increases. For the error, we consider the contaminated normal error, with 2% contamination, as before (bottom left case of Figure 3). To overcome biases in the selection of the sparsity penalty parameter λ, we fix this to the value that gives approximately 20% of unconnected nodes. Figure 4 (top) shows the results of this simulation: the left plot shows the true positive rates as the number of variables increase from $p = 20$ (as in the ROC curves) to $p = 100$, and the number of observations is fixed at $n = 50$. On the right panel, the true discovery rate is plotted, that is the ratio of the number of true edges detected versus the number of all discovered edges. In the bottom plots, we investigate the performance of the methods as n increases, while keeping $p = 20$. This is the more traditional case of n larger than p, where sparse regression methods are often used as variable selection methods. This simulation is closer to our real data application. The plots show a general outperformance of the copula method, particularly for cases with large number of variables and relatively small number of observations, both in terms of true positive and true discovery rates. In general, the differences between the methods become less pronounced as the difference between n and p gets smaller. However, for extreme cases with very large sample sizes and a rel-

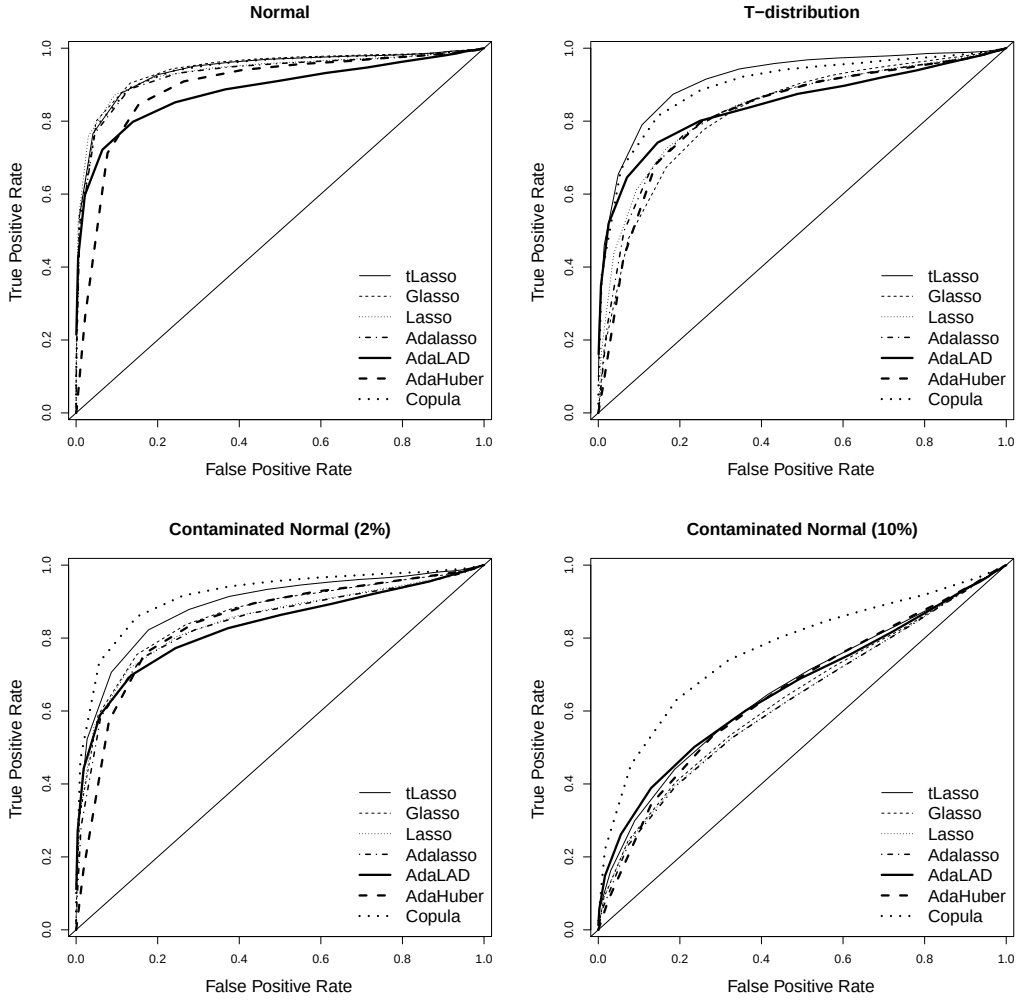

Figure 3: ROC curves, averaged over 500 iterations and networks of 10% density: $n = 50, p = 20$, true positive rate and false positive rate across 30 values of the penalty parameter λ.

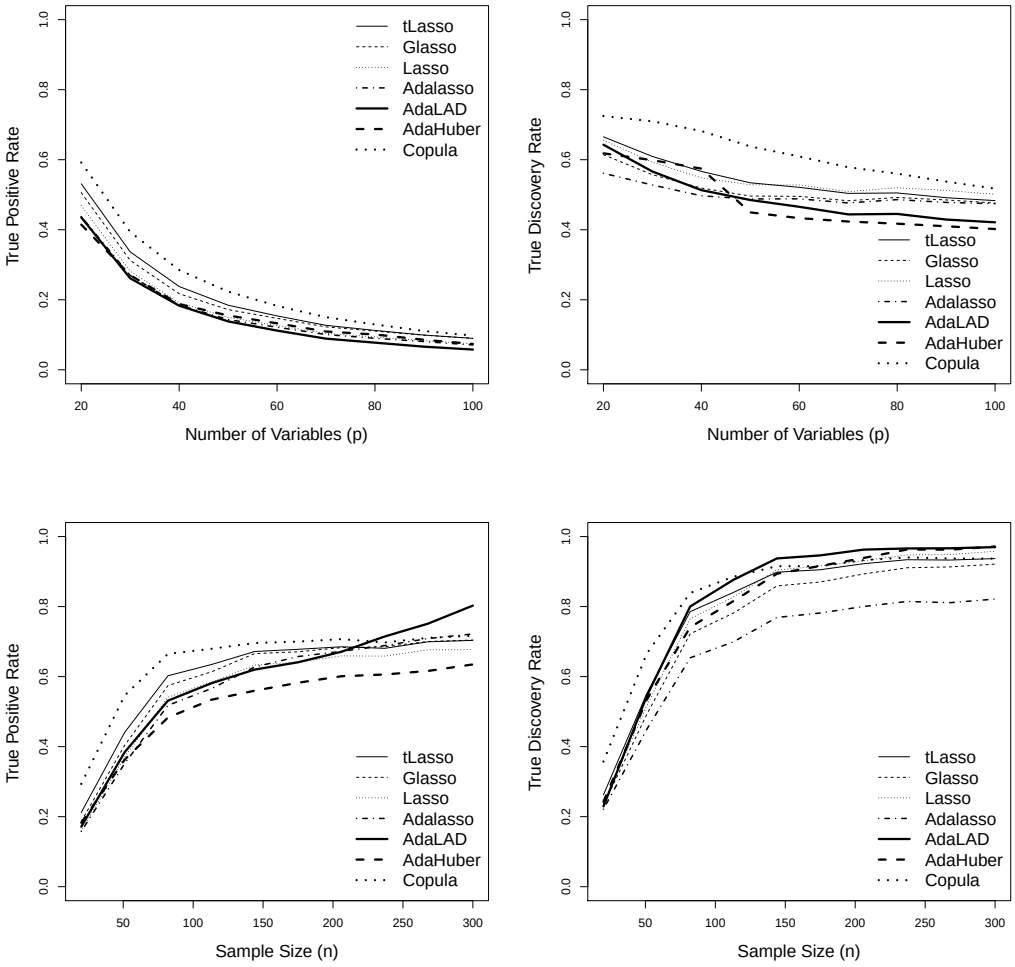

Figure 4: True positive and true discovery rates as the number of variables increases (top panel, $n = 50$) and the number of observations increases (bottom panel, $p = 20$), averaged over 500 iterations. The data are generated by a $N(0, \Theta^{-1})$ distribution with 2% of the data contaminated using a $N(\mu, 0.2)$ distribution for each predictor, with μ equal to 2.5 times the largest diagonal element of Θ^{-1}. The sparsity penalty is chosen so that approximately 20% of networks are unconnected.

atively small number of variables (e.g. $n = 300$ and $p = 20$) the copula approach has a significantly smaller true positive rate than other approaches, such as the nodewise adaptive LAD approach.

In a final simulation, we extend the results of Figure 3 by studying the performance of the methods as the percentage of data contaminated and the degree of contamination increase, respectively. Figure 5 shows the results of this simulation. As before, the data are generated by a $N(0, \Theta^{-1})$ distribution with a percentage of the data contaminated using a $N(\mu, 0.2)$ distribution for each predictor. In the top panel, we increase the percentage of data contaminated from 0% (no contamination) to 20%, while fixing the degree of contamination μ to 2.5 times the largest diagonal element of Θ^{-1}. In the bottom panel, the percentage of data contaminated is set to 2% and μ is varied between 1 and 5 times the largest diagonal element of Θ^{-1}. The results show once again a clear outperformance of the copula method. The bottom panel in particular shows how this method is insensitive to the degree of contamination, in contrast to all other methods which deteriorate as the degree of contamination increases.

3.3. Penalised likelihood versus classical deviance tests on simulated networks

Penalised likelihood methods, such as the ones considered above, perform parameter estimation and model selection at the same time. This is the result of using an L_1 penalty on the parameters of interest, which forces components of the solutions to be exactly zero under penalisation. For completeness, we consider in this section a comparison of these methods with more traditional model selection procedures for graphical models and their robust extensions. We consider the low-dimensional case ($n < p$) where these traditional methods are applicable.

Figure 6 shows the results of a simulation. In the left plot, we simulate data from a normal distribution and compare the tLasso, glasso and copula approaches with a simple one-step model selection procedure. For the latter, the inclusion/exclusion of a potential edge is based on a classical deviance test which compares the full graph without the edge against the full graph with the edge (implemented in the R function `fitConGraph`). The results show how the penalised likelihood methods outperform the classical method in this case. In the right plot, we simulate data from a multivariate $t_{p,3}$ distribution, and perform a similar comparison. For the deviance test, we use the more robust maximum likelihood estimate of the covariance matrix (implemented in the R function `cov.trob`). Furthermore, since the t-distribution is an elliptical distribution, we use the adjusted deviance test of Vogel and Fried (2011) for improved performance. In this case,

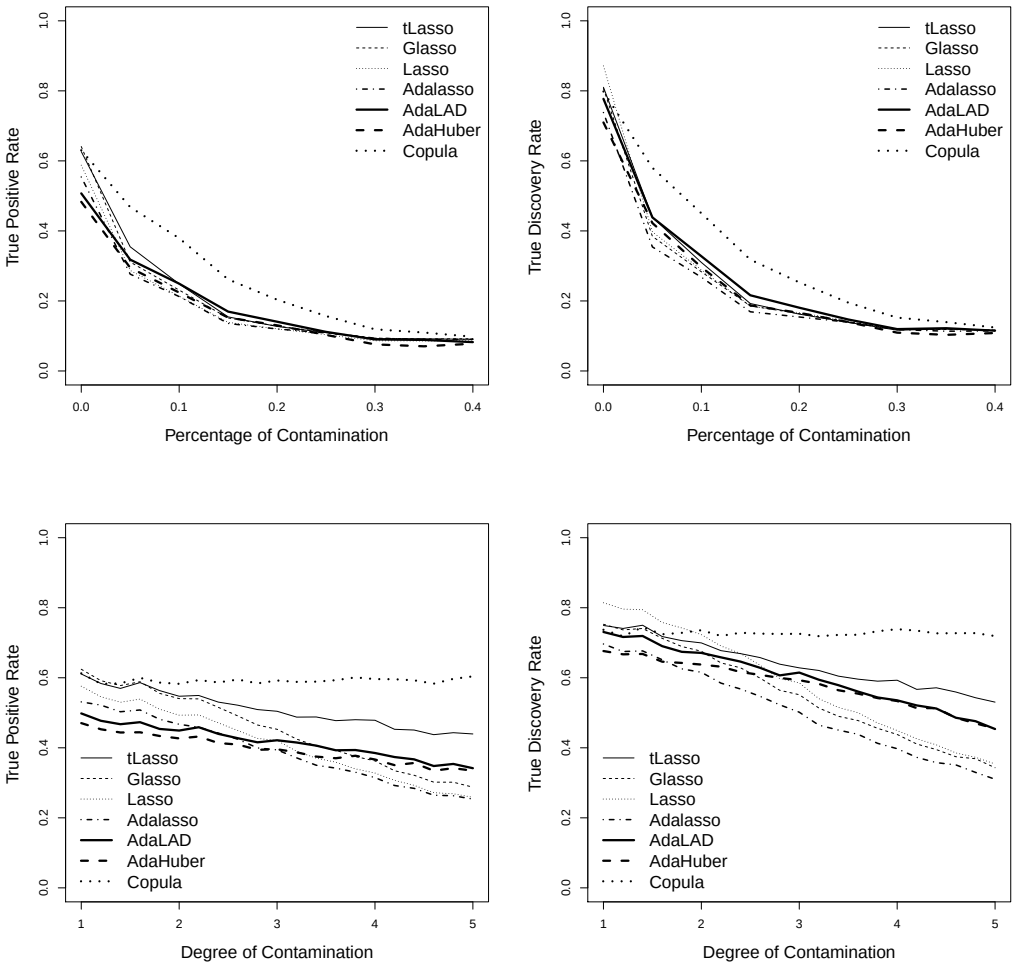

Figure 5: True positive and true discovery rates as the percentage of data contaminated increases (top panel, $n = 50$, $p = 20$) and the degree of contamination increases (bottom panel, $n = 50$, $p = 20$), averaged over 500 iterations. The data are generated by a $N(0, \Theta^{-1})$ distribution with a percentage of the data contaminated using a $N(\mu, 0.2)$ distribution for each predictor. For the top panel, μ is set to 2.5 times the largest diagonal element of Θ^{-1} and the percentage is varied between 0 and 20%. For the bottom panel, the percentage of data contaminated is set to 2% and μ is varied between 1 and 5 times the largest diagonal element of Θ^{-1}. The sparsity penalty is chosen so that approximately 20% of networks are unconnected.

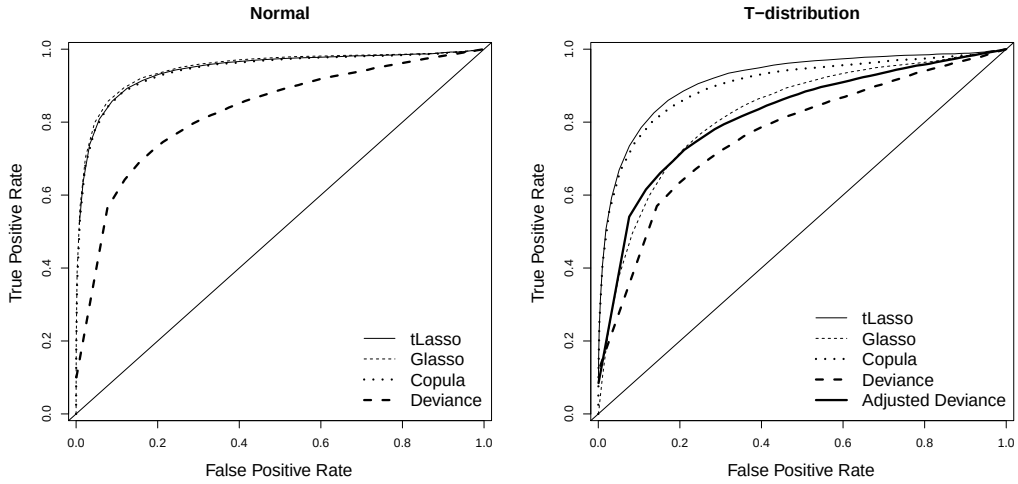

Figure 6: ROC curves, averaged over 500 iterations and networks of 10% density: $n = 50, p = 20$, true positive rate and false positive rate across 30 values of the penalty parameter λ.

the deviance gets adjusted by a factor $\sigma_1 = 1 + \dfrac{2}{p+\nu}$, with ν denoting the degrees of freedom (Tyler, 1983). The results show how the adjusted deviance test is superior to the classical deviance test for non-normal data, but it is comparable to the glasso approach and inferior to both the tLasso and copula approaches.

3.4. Comparison of methods on real data

In this section, we evaluate the performance of the different methods on real data. We consider the gene expression data on yeast generated by the microarray study in Gasch et al. (2000). We restrict our attention to 8 genes involved in a complex network of interactions for the regulation of galactose utilization (Ideker et al., 2001). This dataset is particularly suited for the comparison in this paper as 11 out of the 136 experiments show unusually large negative values for 4 out of these 8 genes: GAL1, GAL2, GAL7, GAL10 (Finegold and Drton, 2011). This corresponds to about 4% of the data being contaminated, so it is a scenario close to our previous simulations (note that a 2% contamination for each predictor in the ROC simulation would correspond to about 8% of the overall data being contaminated). However, the mechanism of contamination is different: here the contamination is concentrated on four genes rather than equally spread out across all predictors. Table 1 reports the results, where the sparsity penalty parameter

Table 1: Comparison of the two networks inferred by the same method on clean and contaminated data. Column 1: overall agreement between the two networks, column 2: number of edges on the contaminated data that are found also on the clean data; columns 3-4: number of links found on clean and contaminated data, respectively, for the subnetworks of the four contaminated genes.

Method	Total Agreement (%)	Common Edges	Density of Subnetwork	
			Clean	Contaminated
Lasso	0.79	6	4	5
Glasso	0.86	7	4	6
Adalasso	0.79	6	4	4
tLasso	0.86	7	3	4
AdaLAD	0.79	6	3	5
AdaHuber	0.79	6	3	4
Copula	0.79	6	3	6

is tuned so that the resulting networks contain only 9 edges, as in Finegold and Drton (2011). For the nodewise adaptive methods, the weights are chosen by a BIC criterion. Furthermore, we scale the data so that each gene has mean 0 and variance 1: we generally find that all of the methods, except for the copula approach, are sensitive to data scaling.

The first column reports the percentage of agreement, in terms of present and missing edges, of the two networks inferred by the same method on the clean and contaminated data, respectively. For the clean data, we remove the expression data for the 11 experiments in question. All methods show generally a good level of agreement, with no clear distinction between the robust and non-robust methods. The tLasso and glasso methods achieve the highest agreement. The second column reports the number of edges (between 1 and 9) in the network inferred from the contaminated data which are found also in the network from the clean data. Given that we select networks with 9 edges, tLasso and glasso show the best results here, with a high consistency of discovered edges in the two networks. Finally in the last two columns, we report the number of edges detected in the subnetwork of the four contaminated genes (6 corresponds to a fully connected subnetwork). These results show a greater robustness of the tLasso method versus the glasso method. In the glasso approach, two edges are detected on the contaminated data which were not detected in the clean data and these are exactly the ones that generate a fully connected subnetwork for GAL1, GAL2, GAL7 and GAL10. This general conclusion supports the one of Finegold and Drton (2011).

Table 2: Comparison of the two networks inferred by the same method on a subset ($n = 55$) of the clean data and on contaminated data. Col 1: overall agreement between the two networks, col 2: # edges on the contaminated data that are found also on the clean data; cols 3-4: # links found on clean and contaminated data, respectively, for the subnetworks of the 4 contaminated genes.

Method	Total Agreement (%)	Common Edges	Density of Subnetwork	
			Clean	Contaminated
Lasso	0.86	7	4	6
Glasso	0.86	7	5	6
Adalasso	0.79	6	3	6
tLasso	0.86	7	4	6
AdaLAD	0.86	7	4	5
AdaHuber	0.86	7	5	6
Copula	1	9	4	4

The copula method did not outperform the other methods on this dataset. The network generated by this method on the contaminated data discovered 3 more edges than the network on the clean data. These three edges generate a fully connected subnetwork for the four genes in question. In fact, a closer inspection reveals that glasso and copula infer exactly the same network on the contaminated data.

The results showed a slight improvement of the tLasso method over the other methods and not much difference overall between robust and non-robust methods. Given our simulation results in the previous section, we have run a comparison for a more extreme case of data contamination. In particular, we consider a subset of the data by randomly selecting 44 of the 125 "clean" experiments. In this way, the resulting dataset has $n = 55$ observations for $p = 8$ genes, with four of the genes having data contaminated for 11 experiments. This corresponds to a 10% level of contamination, as in our ROC comparisons (figure 3, bottom right). Table 2 shows the results in this case.

This comparison reflects closely the results in our previous simulation. Now the copula approach recovers exactly the same network for clean and contaminated data, with the other methods, including tLasso, often detecting a fully connected subnetwork for the four genes in question.

4. Conclusion

Many approaches are developed in statistics that rely on the assumption of normality. These approaches are not suited to data that show clear departures

from normality. This is often the case when data are contaminated, resulting in the presence of outliers. In this paper, we have considered recently developed methods that encourage the use of robust loss functions, such as the Huber or LAD function, in place of the more traditional quadratic loss. In a high dimensional setting, when $p >> n$, an L_1 penalty on the regression coefficients is also considered. In a simulation study, we show how robust methods are superior to the non-robust counterparts, particularly for cases where there is a large departure from normality. Adaptive versions of robust and traditional regression methods have been developed by carefully setting a weight on the β coefficients and these have shown a very good performance, as confirmed also by our simulation study.

Encouraged by these results and inspired by the method proposed by Meinshausen and Bühlmann (2006), the focus of the paper is on the use of these robust lasso regression methods as part of a nodewise approach for graphical modeling. This has not been considered before in the literature. Instead, two main approaches have been developed for robust covariance estimation in high dimensional graphical models, namely tLasso (Finegold and Drton, 2011) and a copula approach (Liu et al., 2009). An extensive simulation study, supported by a real data analysis, show how the nodewise and covariance approaches perform similarly well when the data are generated from a multivariate Gaussian distribution or with a small level of contamination, but the copula approach clearly outperforms all other methods for high level of data contamination. These results extends significantly the comparison of Liu et al. (2009).

The current study has focussed on methods for inferring graphical models from continuous data, such as microarray data. The power of copula methods is that they can be developed also for discrete data. Given the high performance of the copula approach observed in this paper, future work will consider an extension of this method to graphical modeling on discrete data, such as data generated by the latest RNA-sequencing technologies.

Acknowledgment

Thanks to Mike Finegold for providing the tLasso code and advice on the real data analysis, and to Daniel Vogel for advice on the adjusted deviance tests. The authors are grateful to the anonymous referees and the associate editor for their helpful suggestions which greatly improved the original manuscript.

References

Arslan, O., 2012. Weighted LAD-LASSO method for robust parameter estimation and variable selection in regression. Computational Statistics and Data Analysis 56, 1952–1965.

Banerjee, O., Ghaoui, L., D'Aspremont, A., 2008. Model selection through sparse maximum likelihood estimation. Journal of Machine Learning Research 9, 485–516.

Bradic, J., Fan, J., 2011. Penalized composite quasi-likelihood for ultrahigh dimensional variable selection. Journal of the Royal Statistical Society, B 73 (3), 325–349.

Bühlmann, P., van de Geer, S., 2011. Statistics for high-dimensional data. Methods, Theory and Applications. Springer-Verlag.

Finegold, M., Drton, M., 2011. Robust graphical modeling with classical and alternative t-distributions. The Annals of Applied Statistics 5 (2A), 1057–1080.

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

Gasch, A., Spellman, P., Kao, C., Carmel-Harel, O., Eisen, M., Botstein, D., Brown, P., 2000. Genomic expression programs in the response of yeast cells to environmental changes. Molecular Biology of the Cell 11, 4241–4257.

Gottard, A., Pacillo, S., 2010. Robust concentration graph model selection. Computational Statistics and Data Analysis 54, 3070–3079.

Huber, P., 1981. Robust Statistics. Wiley.

Ideker, T., Thorsson, V., Ranish, J., Christmas, R., Buhler, J., Eng, J., Bumgarner, R., Goodlett, D., Aebersold, R., Hood, L., 2001. Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. Science 292 (5518), 929–934.

Krämer, N., Schäfer, J., Boulesteix, A., 2009. Regularized estimation of large-scale gene association networks using graphical Gaussian models. BMC Bioinformatics 10, 384.

Lambert-Lacroix, S., Zwald, L., 2011. Robust regression through the Huber's criterion and adaptive lasso penalty. Electronic Journal of Statistics 5, 1015–1053.

Li, Y., Zhu, J., 2008. L_1-norm quantile regression. Journal of Computational and Graphical Statistics 17 (1), 163–185.

Liu, H., Lafferty, J., Wasserman, L., 2009. The nonparanormal: semiparametric estimation of high dimensional undirected graphs. Journal of Machine Learning Research 10, 2295–2328.

Meinshausen, N., Bühlmann, P., 2006. High-dimensional graphs and variable selection with the lasso. The Annals of Statistics 34, 1436–1462.

Miyamura, M., Kano, Y., 2006. Robust gaussian graphical modeling. Journal of Multivariate Analysis 97, 1525–1550.

Rosset, S., Zhu, J., 2007. Piecewise linear regularized solution paths. The Annals of Statistics 35 (3), 1012–1030.

Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B 58, 267–288.

Tyler, D., 1983. Robustness and efficiency properties of scatter matrices. Biometrika 70 (2), 411–420.

Vogel, D., Fried, R., 2011. Elliptical graphical modelling. Biometrika 98 (4), 935–951.

Witten, D., Friedman, J., Simon, N., 2011. New insights and faster computations for the graphical lasso. Journal of Computational and Graphical Statistics 20 (4), 892–900.

Wu, Y., Liu, Y., 2009. Variable selection in quantile regression. Statistical Sinica 19, 801–817.

Xu, J., Ying, Z., 2010. Simultaneous estimation and variable selection in median regression using Lasso-type penalty. Annals of the Institute of Statistical Mathematics 62, 487–514.