ST 443: Group Project Instruction

Due by **4pm, 11 December, 2019**

1 Problems

1.1 Real World Data

The first part of the project is to apply statistical machine learning techniques on some real world data sets. The students are expected to find the data sets they are interested in from any resource and evaluate the sample performance of different regression or/and classification approaches we have covered in class. I suggest the report includes but not limited to

- Description of the data and the questions that you are interested in answering.
- Review of some of the approaches you tried.
- Summary of the final approach you used and the reason why you chose that approach.
- Summary of the results and conclusion.

1.2 Estimation of graphical models using lasso-related approaches

The graphical model is used to depict the conditional dependence structure among p random variables, $\mathbf{X} = (X_1, \ldots, X_p)^T$. Such a network consists of p nodes, one for each variable and a number of edges connecting a subset of the nodes. The edges describe the conditional dependence structure of the p variables. Specifically, for each $1 \leq j, l \leq p$, let

$$c_{jl} = \mathrm{Cov}\left(X_j, X_l | X_k, 1 \leq k \leq p, k \neq j, l\right)$$

represent the covariance between X_j and X_k conditional on the remaining variables. **Then nodes j and l are connected by an edge if and only if** $c_{jl} \neq 0$. Let $G = (V, E)$ denotes an undirected graph with vertex set $V = \{1, \ldots, p\}$ and edge set

$$E = \{(j, l) : c_{jl} \neq 0, 1 \leq j, l \leq p, j \neq l\} \tag{1}$$

1.2.1 Node-wise lasso approach

Under the assumption that $\mathbf{X}$ is multivariate Gaussian, to estimate the edge set in (1), one can use the **node-wise lasso approach**. Specifically, for each node $j \in V$, regress X_j on the remaining variables $X_l, l \in V, l \neq j$, in the form of

$$X_j = \sum_{1 \leq l \leq p, l \neq j} \beta_{jl} X_l + \varepsilon_{jl}.$$

To obtain a sparse solution for β_{jl}'s, we can implement the lasso approach to select which component in $\{\beta_{jl}, l \in V, l \neq j\}$ is non-zero. Suppose that, with a certain choice of the tuning parameter, the lasso estimator for β_{jl} is $\widehat{\beta}_{jl}$. If $\widehat{\beta}_{jl} \neq 0$, we say nodes j and l are estimated to be connected. One can consider the following rules, named **node-wise lasso 1** and **node-wise lasso 2**, respectively, to estimate E in (1):

$$\widehat{E}_1 = \{(j,l) : \text{both } \beta_{jl} \text{ and } \beta_{lj} \text{ are nonzero}, 1 \leq j, l \leq p, j \neq l\},$$

$$\widehat{E}_2 = \{(j,l) : \text{either } \beta_{jl} \text{ or } \beta_{lj} \text{ is nonzero}, 1 \leq j, l \leq p, j \neq l\}.$$

1.2.2 Graphical lasso approach

Under the assumption that $\mathbf{X}$ is multivariate Gaussian with covariance matrix $\mathbf{\Sigma} \in \mathbb{R}^{p \times p}$, one can show that $c_{jl} = 0$ if and only if $\Theta_{jl} = 0$, where Θ_{jl} is the (j,l)-th entry of the inverse covariance matrix, $\mathbf{\Theta} = \mathbf{\Sigma}^{-1}$. So the edge set can also be represented by

$$E = \{(j,l) : \Theta_{jl} \neq 0, 1 \leq j, l \leq p, j \neq l\}.$$

In practice Θ_{jl} and hence the network structure, can be estimated based on a set of n observed p-dimensional realizations, $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of $\mathbf{X}$. See the Appendix for the description of the graphical lasso approach.

With a certain choice of the tuning parameter, the estimated edge set is

$$\widehat{E}_3 = \left\{(j,l) : \widehat{\Theta}_{jl} \neq 0, 1 \leq j, l \leq p, j \neq l\right\}.$$

One can use the glasso package in R to obtain $\widehat{\mathbf{\Theta}}$. It is quite easy to use this package to implement the graphical lasso approach.

1.2.3 Target of the project

The goal to do this project involves

- Learn the idea of node-wide lasso approach described in Section 1.2.1. Use the **R package glmnet**[1] to implement the lasso.
- Learn the idea of the graphical lasso approach described in Section 1.2.2. Learn to use the **R** package **glasso** to implement the graphical lasso.

[1] We will use this package in the Week 6's computer workshop

- Conduct simulations to compare the sample performance of different approaches in recovering the edge set, E, in (1).

  Specifically, let $\mathbf{\Theta} = \mathbf{B} + \delta \mathbf{I}_p \in \mathbb{R}^{p \times p}$, where $\mathbf{I}_p$ is the identity matrix, each off-diagonal entry in $\mathbf{B}$ (symmetric matrix) is generated independently and equals 0.5 with probability 0.1 or 0 with probability 0.9. $\delta > 0$ is chosen such that $\mathbf{\Theta}$ is positive definite. Finally, the matrix is standardized to have unit diagonals (transforming from covariance matrix to correlation matrix). The sparsity pattern in $\mathbf{\Theta}$ corresponds to the true edge set in (1).

  - Generate n random samples, $\mathbf{x}_1, \ldots, \mathbf{x}_n$, from a multivariate Gaussian distribution with zero mean and the covariance matrix $\mathbf{\Sigma} = \mathbf{\Theta}^{-1}$.
  - Apply the approaches in Sections 1.2.1 and 1.2.2 to estimate E in (1).
  - For each method and a certain choice of the tuning parameter λ, we calculate the true positive rate (TPR_λ) and false positive rate (FPR_λ), in terms of network edges corrected identified. Plotting TPR_λ vs FPR_λ over a fine grid of values of λ produces a ROC curve. Compute the area under the ROC (AUROC) for each method to compare the overall performance on recovering the true edge set.
  - Develop methods to select the optimal tuning parameter λ of each method in Sections 1.2.1 and 1.2.2. Based on the selected optimal tuning parameters, compare the sample performance of each method in terms of the support recovery, e.g. False Positives, False Negatives.
  - Replicate the above procedure 50 times. Compare the mean (standard error) of relevant measure terms for each of the comparison methods. The boxplot can be used.

1.2.4 Content of the report

I suggest the report includes

- A general introduction of the graphical lasso approach and the node-wise lasso approach. The mathematical derivations are not required.
- Discuss the selection of optimal tuning parameters for each method.
- Describe the simulation settings and report the numerical results under different choices of n, p and other structural parameters, e.g. sparsity structure.
- Summarize your findings.

2 Timeline

- Week 5-7: Contact group members, decide who contributes to which part, search for the data sets, and learn the approaches in Sections 1.2.1 and 1.2.2. From the 8th week,

please visit two GTA's or my office hours to let us know the real data you decide to work on.

- Week 8-10: Analyse the data and write codes for applying the lasso related approaches on simulated examples.
- Week 10-11: Write the report and submit the report.

3 Submission and assessment

3.1 Written report

The written report for the first part of the real data problem should be maximum of **3 pages**. The written report for the second part will not have any page limit, but I expect it to be less than **10 pages**.

Note the students are required to submit the reproducible codes for both parts of the project. Also the students should put additional figures, tables, mathematical derivations in the appendix.

Deadline to submit your coursework report: **4:00pm, 11th December 2019, Wednesday of 11th week.** Further details are to be announced.

3.2 Mark scheme

- Real data (50%)
- Graphical models (50%)

The grade for each part will be based on illustration of the dataset, the machine learning approaches you tried, discussion of the results, the readability and efficiency of R code, quality of layout and use of language and etc. The grade for each group member will be a function of the contribution of each group member using the following equation.

$$\text{member grade } = \text{report grade } \times \frac{\text{member contribution}}{\text{maximum contribution}}.$$

For example, for a group with 5 members contributing, 30%, 20%, 20%, 20%, 10% and the report grade is 75 (out of 100), the individual grades are

$$75 \times \frac{30}{30} = 75, \quad 75 \times \frac{20}{30} = 50, \quad 75 \times \frac{20}{30} = 50, \quad 75 \times \frac{20}{30} = 50, \quad 75 \times \frac{10}{30} = 25.$$

A Appendix

A.1 Description of graphical lasso approach

The Gaussian log-likelihood (up to constants) of the graphical models can be written as[2]

$$\log\det\boldsymbol{\Theta} - \text{trace}(\mathbf{S}\boldsymbol{\Theta}) \tag{2}$$

where $\bar{\mathbf{x}} = \sum_{i=1}^{n} \mathbf{x}_i/n$ and $\mathbf{S} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$ is the sample covariance matrix of $\mathbf{x}_1, \ldots, \mathbf{x}_n$. The **graphical lasso** approach, considers a sparse estimate for $\boldsymbol{\Theta}$ by adding an ℓ_1 penalty on the off-diagonal entries of $\boldsymbol{\Theta}$ to (2).

$$\log\det\boldsymbol{\Theta} - \text{trace}(\mathbf{S}\boldsymbol{\Theta}) + \lambda\sum_{j\neq l}\Theta_{jl} \tag{3}$$

The graphical lasso approach considers maximizing (3) over all symmetric positive matrices $\boldsymbol{\Theta} \in \mathbb{R}^{p\times p}$ and provide a sparse estimate, $\widehat{\boldsymbol{\Theta}}$. In a similar fashion to the standard lasso, the ℓ_1 penalty in (3) both regularizes the estimate and ensure $\widehat{\boldsymbol{\Theta}}$ is sparse, i.e. has many zero elements. Note $\lambda \geq 0$ is the tuning parameter to control the sparsity level in $\boldsymbol{\Theta}$ (how? please try this in your numerical experiment).

[2]The formulas of (2) and (3) are not required, we just use it for illustration.