10-708 PGM Project Final Report
Learning SNP-Gene Network Using Mixed Graphical Model

Hyun Ah Song HYUNAHS@ANDREW.CMU.EDU
Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA 15213

Ji Oh Yoo JIOHY@ANDREW.CMU.EDU
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213

1. Introduction

Recent progress on large scale genome-wide studies show that many human common diseases, including diabetes, asthma, and cancer, consist of complex reactions of proteins, which are controlled by regulatory networks and are expressed from highly correlated genetic variations (Basso et al., 2005; Chen et al., 2008). Thus, discovering the eQTL mapping between the genetic variations of interest and expression rates of disease-related proteins, rather than relatively simple clinical phenotypes, is necessary to analyze the occurrence mechanism of the diseases and the functional roles of those proteins in the mechanism. This requires the joint study of the genetic variants and the expression of the related proteins rather than combining the results from the independent study on each genome or phenome. Multiple regression studies have been conducted to identify these mappings between single-nucleotide polymorphisms (SNPs) and related gene expression rates, and to develop the more efficient and accurate methods for this eQTL mappings discovery. We summarize these studies in **section 2**. Our contribution is the adoption of conditional undirected mixed graphical model, which can embed both discrete variables for SNPs and continuous variables for gene expression rates, to learn a more accurate generative model.

2. Background & Related Work

2.1. Related Work

There have been various types of solutions to the problem of learning SNPs-gene network. One popular solution is to solve the problem as multi-task regression. Given SNPs information as inputs, the goal is to find the regression parameters that can map the input to the outputs of gene expressions, which can reveal the structured sparsity in input and output as well.

Tree-guided group lasso, or GFlasso was proposed by Kim and coworkers (Kim & Xing, 2010). GFlasso aims to learn the common set of inputs for each cluster of outputs, using group lasso penalty and systematic weighting scheme for inputs, where inputs are grouped together to be mapped to the outputs, and output clusters with strong correlation are guided to share common input groups. GFlasso requires the prior knowledge of output tree structure, which is not always possible. Also, the weighting scheme of guiding the common set of inputs for highly correlated clusters is such a strong assumption that may not be true in reality.

An algorithm called multivariate regression with covariance estimation (MRCE) that aims to learn both multivariate regression parameters and the correlation of outputs was introduced by Rothman and coworkers (Rothman et al., 2010). By regularizing regression parameter and correlation matrix separately, the MRCE solve bi-convex problem, which may not lead to the global optimum. Although MRCE is favorable in a way that it learns the output structure, MRCE does not force structured sparsity when learning regression parameters for inputs, which is an important property when dealing with SNPs-gene data.

An algorithm that bring together the advantages of the two previous works (Kim & Xing, 2010) (Rothman et al., 2010) was introduced by Sohn and coworkers (Sohn & Kim, 2012). The proposed algorithm jointly learns regression parameters and output structure with constraint of structure sparsity in inputs, by learning conditional Gaussian graph model. Although proposed algorithm resolves the main problems in previous studies, it assumes that input and output are treated as continuous data, which is not true for SNPs-gene data. Therefore, it leaves some space for further refinement of the assumptions used in this algorithm.

In order to solve the problem in more natural way, we can adopt 'mixed graphical models' into our problem. Mixed graphical models refer to graphical models that allow for both discrete and continuous variables.

Lauritzen first proposed a mixed graphical model for vari-

ables of both discrete and continuous (Lauritzen & Wermuth, 1989). Although this mixed graphical model is carefully designed for general cases, this makes resulting conditional distribution complex. Also, the mean and covariance matrices exist for every possible configurations of states of discrete variables, which results in exponential increase in number of parameters to learn depending on the number of discrete variables.

Later by Jason and coworkers, the mixed graphical model was further developed into more simplified and intuitive version (Lee & Hastie, 2013). The authors proposed a mixed graphical model that provides intuitive forms of conditional distributions: conditional distribution of a discrete variable reduces to multi-class logistic regression, and that of a continuous variable to Gaussian linear regression. By making assumptions on the general mixed graphical models (Lauritzen & Wermuth, 1989), proposed algorithm scales up more efficiently: it has common covariance matrix, and additive mean.

To our knowledge, there has not been any study that learns the SNP-gene network as multi-task regression problem using mixed graphical models. In this project, we would like to extend the basic concepts proposed by Sohn (Sohn & Kim, 2012) using mixed graph models (Lee & Hastie, 2013), to solve the problem in more natural way.

2.2. Background

One of the most commonly used method in estimation of SNP-gene network is standard regression method with lasso penalty (Tibshirani, 1996), which solves the following optimization problem.

$$\arg\min_{\mathbf{B}} \text{tr}((\mathbf{Y}-\mathbf{XB})(\mathbf{Y}-\mathbf{XB})^T) + \lambda\|\mathbf{B}\|_1, \quad (1)$$

where $\mathbf{X} = (\mathbf{x_1}, ..., \mathbf{x_N})^T$, $\mathbf{Y} = (\mathbf{y_1}, ..., \mathbf{y_N})^T$ are input and output data, and $\mathbf{x_i} = (x_{i1}, ..., x_{iJ})^T \in \mathbb{R}^J$, and $\mathbf{y_i} = (y_{i1}, ..., y_{iK})^T \in \mathbb{R}^K$. The output is estimated by regression: $\mathbf{y_i} = \mathbf{Bx_i} + \varepsilon_\mathbf{i}$, where $\varepsilon \sim \mathcal{N}(0, \mathbf{\Psi})$. By enforcing sparsity in $\mathbf{B}$, it is able to take into account of the problem setting that we have $J >> N$ (SNPs data comprises of high-dimensional vector).

To address problem of learning structured sparsity and ouput structure using regression model, Sohn (Sohn & Kim, 2012) proposed a model that extends the problem of learning standard regression model into learning Gaussian graphical model.

By formulating the joint distribution of input and output as 2, where $\mathbf{\Sigma}^{-1} = \begin{pmatrix} \mathbf{\Theta}_{xx} & \mathbf{\Theta}_{xy} \\ \mathbf{\Theta}_{xy}^T & \mathbf{\Theta}_{yy} \end{pmatrix}$, the authors show that it can be interpreted as different parameterization of standard regression model 1 by setting $\mathbf{B} = \mathbf{\Sigma}_{xy}^T \mathbf{\Sigma}_{xx}^{-1} = -\mathbf{\Theta}_{yy}^{-1}\mathbf{\Theta}_{xy}^T$, and $\mathbf{\Psi} = \mathbf{\Theta}_{yy}^{-1}$.

$$\begin{bmatrix} \mathbf{x_i} \\ \mathbf{y_i} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0}_J \\ \mathbf{0}_K \end{bmatrix}, \Sigma \right) \quad (2)$$

Although the method by Sohn suits the needs for the SNP-gene network estimation problem in a way that it learns both structured sparsity and output structure, and formulates the problem using the standard regression problem, the underlying assumption is that both input and output data constitute of continuous variables.

By raising question of how to estimate SNPs-gene network in more natural way, we turn our attention to other various types of graphical models that can reflect the nature of discrete and continuous variables.

For continuous variables, multivariate Gaussian model is one of the commonly used graphical model, $y \sim \mathcal{N}(0, \Theta^{-1})$, where the inverse covariance Θ is estimated via graphical lasso. For discrete variables, pairwise Markov random field $p(x) \propto exp \sum_{r<j} \phi_{rj}(x_r, x_j)$ is one of the commonly used graphical model. Studies on mixed graphical model, which combines two commonly used graphical models of Gaussian and pair-wise graphical models so that it can handle data consisting of both continuous and discrete variables, have been conducted by authors in (Lauritzen & Wermuth, 1989), where the conditional distribution is modeled as $p(y|x) = \mathcal{N}(\mu(x), \Sigma(x))$. Despite of the novelty and specific applications of the mixed graphical model, it has not been used as widely due to its complexity in learning the parameters, and lack of intuition when derived to the other types of distributions.

The authors in (Lee & Hastie, 2013) have proposed a special case of the mixed graphical model that resolves the problem of the complexity in parameter learning, and vague forms of derived distributions, by making several assumptions. The detailed explanation follows in section 3.

In our work, we plan to investigate the behavior of mixed graphical model, whose assumption on the types of data it can handle does not violate our problem setting for SNP-gene network estimation. We plan to compare the performance of the mixed graphical model when applied to SNP-gene network estimation problem, and that of commonly used method.

3. Methods

Our goal is to improve the method of learning SNP-gene network by adopting the conditional undirected mixed graphical model in which we can embed both discrete variables for SNPs and continuous variables for gene expres-

sion rates.

3.1. Baseline Method

Lee and coworkers (Lee & Hastie, 2013) proposed a mixed graphical model of discrete and continuous variables, where the joint distribution is expressed as:

$$p(x, y; \Theta) \propto \exp\Big(\sum_{k=1}^{q}\sum_{l=1}^{q}\phi_{kl}(x_k, x_l) + \sum_{k=1}^{p}\sum_{s=1}^{q}\rho_{ks}(x_k)y_s + \sum_{s=1}^{p}\alpha_s y_s + \sum_{s=1}^{p}\sum_{t=1}^{q} -\frac{1}{2}\beta_{st}y_s y_t\Big) \tag{3}$$

where $x_1, ...x_p$ are discrete variables representing the occurrences for each SNP and $y_1, ..., y_q$ are continuous variables representing the gene expression rate of each gene. The parameters are $\Theta = [\{\phi_{kl}\}, \{\rho_k\}, \{\alpha_s\}, \{\beta_{st}\}]$, where ϕ_{kl}, ρ_k, α_s, β_{st} are the discrete-discrete edge potential, continuous-discrete edge potential, continuous node potential, and continuous-continuous edge potential, respectively.

By making further assumptions on mixed graphical models originally introduced in (Lauritzen & Wermuth, 1989), the mixed graphical model 3 proposed by Lee (Lee & Hastie, 2013) yields intuitive results when reformulated into various forms of distributions. The conditional distribution of continuous variables and discrete variables given the rest reduces to Gaussian linear regression, and multi-class logistic regression, respectively. Furthermore, when the data consists of continuous and discrete variables only, the model reduces to multivariate Gaussian model, and pairwise discrete Markov random field, respectively.

3.2. Revised Method

We are interested in learning the SNP-gene network structure, and the conditional distribution of the gene expression, given observed SNPs. Therefore, we only need to focus on learning the conditional distribution of the network rather than learning the full joint distribution.

The conditional distribution of continuous variables given discrete variables of the mixed graphical model in (Lee & Hastie, 2013) reduces to the form of multivariate Gaussian distribution.

$$p(y|x) = \mathcal{N}(B^{-1}\gamma(x), B^{-1}) \tag{4}$$

$$\{\gamma(x)\}_s = \alpha_s + \sum_k \rho_{ks}(x_k) \tag{5}$$

$$p(x) \propto \exp\Big(\sum_k\sum_l \phi_{kl}(x_k, x_l) + \frac{1}{2}\gamma(x))^\intercal B^{-1}\gamma(x)\Big) \tag{6}$$

where B is a symmetric, positive definite inverse covariance matrix $B = \{\beta_{st}\}$ that is shared across the Gaussian distributions.

From the property of multivariate Gaussian distribution, the log likelihood of the parameters Θ given discrete variables y can be expressed as:

$$\log p(y|x;\Theta) = -\frac{1}{2}\log|B^{-1}| - \frac{k}{2}(2\pi) - \frac{1}{2}(y - B^{-1}\gamma(x)^\intercal B(y - B^{-1}\gamma(x)) \tag{7}$$

$$l_p(\Theta) = -\frac{1}{2}tr\Big((Y - B^{-1}\Gamma(X))^\intercal B(Y - B^{-1}\Gamma(X))\Big) - \frac{N}{2}\log|B^{-1}| + \lambda_1\|\{\beta_{st}\}\|_1 + \lambda_2\|\{\rho_{ks}\}\|_1 \tag{8}$$

The parameter Θ^* that minimizes this L-1 penalized log-likelihood will give us a sparse and consistent estimator, and by comparing the parameters with results in multi-task regression settings, we can analyze the existence of direct and indirect relationship between SNPs and gene expression rates.

For the parameter estimation, we minimize the negative log-likelihood expressed in product of each variable given the rest as follows:

$$\tilde{l}(\Theta|y) = -\sum_{s=1}^{p}\log p(y_s|y_{\setminus s}, x; \Theta), \tag{9}$$

where from 3, $p(y_s|y_{\setminus s}, x; \Theta)$ can be reformulated as follows:

$$p(y_s|y_{\setminus s}, x; \Theta) = \frac{\sqrt{\beta_{ss}}}{\sqrt{2\pi}}\exp\left(\frac{-\beta_{ss}}{2}\left(y_s - \frac{\alpha_s + \sum_j \rho_{sj}(y_j) - \sum_{t=s}\beta_{st}y_t}{\beta_{ss}}\right)\right)^2 \tag{10}$$

3.3. Optimization

For the optimization, proximal Newton method (Schmidt, 2010; Schmidt et al., 2011) is used. Proximal Netwon optimization method breaks down the optimization problem

into $f(x) + g(x)$, where $f(x)$ is smooth and convex function, and $g(x)$ is a convex function, but not necessarily smooth. In our problem, functions $f(x)$ and $g(x)$ correspond to the negative log-likelihood and sparsity constraint, respectively.

Proximal Newton method is an extension of proximal gradient method that considers the second order information of $f(x)$ instead of the first order. The proximal gradient seeks next point by computing $x_{k+1} = prox_t(x_k - t \bigtriangledown (x_k))$, where $prox_t(x) = \text{argmin}_u(\frac{1}{2t}\|x-u\|^2 + g(u))$, and t is chosen by Armijo line search algorithm. In proximal Newton method, $Hprox$ is used, where the $prox$ operator is extended to $\|\cdot\|_H$, where $H = \bigtriangledown^2 f(x_k)$. For each iteration, proximal Newton method finds the next point that minimizes below function.

$$\bigtriangledown f(x_k)^T(u - x_k) + \frac{1}{2t}(u + x_k)^T H(u - w_k) + g(u)$$
$$= Hprox_t(x_k - tH^{-1} \bigtriangledown f(x_k)). \quad (11)$$

Previous studies on proximal Newton method (Lee et al., 2012) shows that it converges faster especially when n is large, with better accuracy compared to the proximal gradient method.

For the implementation, H is approximated using BFGS. We used PNOPT package (Lee et al., 2012) for implementation of proximal Newton optimization algorithm.

4. Experiments

We compare the performance of our method to lasso, multi-task lasso with L_1/L_2 regularization, and sparse CGGM using synthetic dataset and will compare the performances on real dataset for eQTL mapping later.

4.1. Basic Synthetic Experiments

We analyze the performance of our method on the synthetic dataset. We created a simple model as a ground truth with 10 discrete variables (input) and 10 continuous variables (output). The discrete variables have two possible binary states, and their node potentials is 1 for two consecutive variables and zero otherwise. The continuous variables have 0.25 potentials for two consecutive variables and zero otherwise. The potential between continuous and discrete variables are 1 if they have the same index and zero otherwise. Total 2000 samples are sampled from the true model and used for estimating the edges between the discrete variables and the continuous variables. We analyze the performance based on the two measures: prediction error (mean squared error of predicted output) and accuracy of edge discovery (true positive and true negative out of total number of edges). To find the optimal hyper-parameters, we use 5-fold cross-validation based on the prediction error for lasso, multi-task lasso, and sparse CGGM, and the likelihood for our method.

Table 1. Prediction error on synthetic dataset with methods MG (mixed graphical model), SCGGM (sparse CGGM), Lasso, MT-Lasso (multi-task Lasso).

Method	Prediction Error
MG	3.1606 ± 0.1238
SCGGM	1.1420 ± 0.0398
Lasso	1.1244 ± 0.0558
MTLasso	1.1215 ± 0.0263

Table 4.1 shows the prediction error on test set of all methods. For the baseline methods, multi-task lasso shows the lowest prediction error, but they all lie in the confidence interval of each other, so all the baseline methods do not show the significant difference in their performance. Our method shows the highest and large prediction error. We interpret this low performance of our method as it is because of the large number of samples (2000) relatively to the number of parameters to learn. Even when the objective function is penalized by the L1 penalty, the optimization algorithm tries to maximize the objective by abandoning the sparsity of the model in the abundance of the training data.

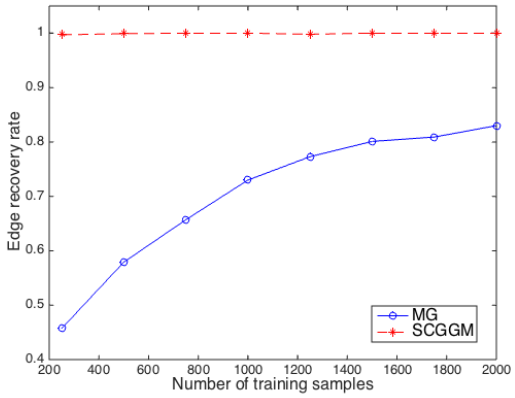

Figure 1. Edge recovery rate of MG and SCGGM on the synthetic dataset. Edge recovery rate of SCGGM is above 95% for all training sample sizes and greater than that of MG.

Figure 4.1 shows the accuracy of edge discovery of sparse CGGM and our method varying the number of samples used from 250 to 2000. The overall accuracy of sparse CGGM is greater than 97% for all number of samples, but our method shows the lower accuracy around 50% on 250 samples and it grows to 80% on increased number of samples. The accuracy is measured by averaging the accuracy of a single run over 10 times for both methods.

4.2. More Synthetic Experiments

As we believe the low performance of our model is due to the high simplicity and the small number of dimensions and parameters, we are planing to create a model with more dimensions and complexity and to train the models with less samples. We hope we can get results that our model is robust with relatively small size of the data as well as accurate in both prediction and edge discovery.

4.3. Pre-processing SNPs-Gene Expression Data

For exploring the eQTL mapping between SNPs and the expression rates of the related genes, we use the data from Human Liver Cohort(HLC) study (Schadt et al., 2008) [1]. Out of 427 patients with more than 40,000 and more than 700,000 SNPs in the given data, we first extract 137 samples that contain both information on genotype and expression rates. To focus on learning of the structure of the model rather than handling the huge dimension of this dataset, we focus on smaller subset of genes and expression rates as in the work of Sohn and coworker (Sohn & Kim, 2012). We consider only expression rates whose variance is greater than 0.05 and SNPs on chromosome 6 with minor allele frequency greater than 0.01 and pair-wise correlation less than 0.1 to select genes that are not too biased to have major allele and not too related each other according to dbSNP[2]. For a single nucleotide, where major allele is X and minor is Y, SNPs are encoded to categorical variables as XX = 0, XY = 1, YY = 2. SNPs are considered to be discrete variables with three states (0, 1, 2) and expression rates are to be continuous variables in our model.

At this point, the pre-processing of this HLC dataset is finished. After more analyses on the performance and the characteristics of our method, we will run our method on this data and report the results along with other baseline methods.

5. Conclusion

In this project, we adopt mixed graphical model for a better generative model in discovering the eQTL mapping in SNPs-gene networks. In synthetic experiments, our model shows lower performance in both prediction error and edge recovery rate than the baseline models due to the true model's simplicity. We will conduct more synthetic experiments on more complex synthetic model and real dataset from HLC study.

[1] The dataset is downloadable at https://www.synapse.org/#!Synapse:syn4499

[2] http://www.ncbi.nlm.nih.gov/projects/SNP/

References

Basso, Katia, Margolin, Adam A, Stolovitzky, Gustavo, Klein, Ulf, Dalla-Favera, Riccardo, and Califano, Andrea. Reverse engineering of regulatory networks in human b cells. *Nature genetics*, 37(4):382–390, 2005.

Chen, Yanqing, Zhu, Jun, Lum, Pek Yee, Yang, Xia, Pinto, Shirly, MacNeil, Douglas J, Zhang, Chunsheng, Lamb, John, Edwards, Stephen, Sieberts, Solveig K, et al. Variations in dna elucidate molecular networks that cause disease. *Nature*, 452(7186):429–435, 2008.

Kim, Seyoung and Xing, Eric P. Tree-guided group lasso for multi-task regression with structured sparsity. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 543–550, 2010.

Lauritzen, Steffen L and Wermuth, Nanny. Graphical models for associations between variables, some of which are qualitative and some quantitative. *The Annals of Statistics*, pp. 31–57, 1989.

Lee, Jason and Hastie, Trevor. Structure learning of mixed graphical models. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, pp. 388–396, 2013.

Lee, Jason D, Sun, Yuekai, and Saunders, Michael A. Proximal newton-type methods for minimizing convex objective functions in composite form. 2012.

Rothman, Adam J, Levina, Elizaveta, and Zhu, Ji. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4): 947–962, 2010.

Schadt, Eric E, Molony, Cliona, Chudin, Eugene, Hao, Ke, Yang, Xia, Lum, Pek Y, Kasarskis, Andrew, Zhang, Bin, Wang, Susanna, Suver, Christine, et al. Mapping the genetic architecture of gene expression in human liver. *PLoS biology*, 6(5):e107, 2008.

Schmidt, Mark. *Graphical model structure learning with l1-regularization*. PhD thesis, UNIVERSITY OF BRITISH COLUMBIA (Vancouver, 2010.

Schmidt, Mark, Kim, Dongmin, and Sra, Suvrit. Projected newton-type methods in machine learning. 2011.

Sohn, Kyung-Ah and Kim, Seyoung. Joint estimation of structured sparsity and output structure in multiple-output regression via inverse-covariance regularization. In *International Conference on Artificial Intelligence and Statistics*, pp. 1081–1089, 2012.

Tibshirani, Robert. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.