Sparse Covariance Modeling in High Dimensions with Gaussian Processes

Rui Li[1], Kishan KC[1], Feng Cui[2], Justin Domke[3], and Anne R. Haake[1]

[1]Golisano College of Computing and Information Sciences, Rochester Institute of Technology
[2]Gosnell College of Life Sciences, Rochester Institute of Technology
[3]College of Information and Computer Sciences, University of Massachusetts Armherst

Abstract

This paper studies statistical relationships among components of high-dimensional observations varying across non-random covariates. We propose to model the observation elements' changing covariances as sparse multivariate stochastic processes. In particular, our novel covariance modeling method reduces dimensionality by relating the observation vectors to a lower dimensional subspace. To characterize the changing correlations, we jointly model the latent factors and the factor loadings as collections of basis functions that vary with the covariates as Gaussian processes. Automatic relevance determination (ARD) encodes basis sparsity through their coefficients to account for the inherent redundancy. Experiments conducted across domains show superior performances to the state-of-the-art methods.

1 Introduction

In many applications, the complex relationships among components of high-dimensional observations change across non-random covariates (e.g., experimental conditions, times). For example, a major challenge for computational gene regulatory network (GRN) inference is that the topological structures of GRNs are context-dependent. Different interactions of gene activities will be active in different experimental conditions (e.g., culture media, temperatures, pH), leading to a different GRN structure [1, 2]. Another scenario is that crime occurrences exhibit correlations across spatially disjoint regions, meanwhile, the spatial correlations evolve over time [3].

The modeling methods typically combine heterogeneous data from different experimental conditions or times in a single data set by assuming homoscedastic models with independent and identically distributed (i.i.d.) errors. For example, let $\mathbf{y} = (y_1, \ldots, y_d)^T \in R^d$ denote a vector of d gene expression levels measured under an experimental condition indexed by a non-random covariate $x \in \mathbf{X} \subset R$. These methods compute the conditional mean $E(\mathbf{y}|x) = \mu(x)$ while assuming the conditional covariance matrix $cov(\mathbf{y}|x) = \Sigma$ to be a constant. This leads to inappropriately biased estimates, and obscures the distinguishing variations of GRN structures subject to specific experimental conditions.

Instead, we propose a novel covariance modeling method that allows $cov(\mathbf{y}|x) = \Sigma(x)$ to change flexibly with $\mathbf{X}$. We make low-rank approximations to covariate-dependent covariance matrices with latent factor models. In particular, we characterize the loadings as a sparse combination of unknown basis functions. The basis functions vary over $\mathbf{X}$ with Gaussian processes as their convenient priors. This leads to more flexible covariance matrices than modeling $\Sigma(x)$ as a quadratic function of x by assuming a linear mapping from covariates to observations. For the coefficient matrix, we employ the automatic relevance determination (ARD) method to place a shrinkage prior on its size with the loadings increasingly shrunk towards zero as their column index increases. The induced covariance matrices are regularized quadratic functions of these basis elements. Our method allows

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

covariate-specific correlations to share statistical strength while retaining their distinctive property. Since commonly only a portion of observation components have latent statistical relationships, ARD encodes sparsity to handle the inherent redundancy. The posterior computation is tractable with conjugate posterior updates.

We evaluate our covariance modeling method across domains: for GRN inference, using benchmark gene expression microarray datasets for a eukaryotic model organism (*S. cerevisiae*), a human pathogen (*S. aureus*), and a prokaryotic model organism (*E. coli*), we achieve robustly better performances than the state-of-the-art competitor methods across the organisms; for crime occurrence prediction, by capturing the spatial correlations of weekly crime rates among the 180 census tracts in Washington D.C evolving over time between 2016-2018, we outperform two state-of-the-art methods.

2 Related works

GRN inference is a long-standing challenge, and a wide variety of computational approaches are proposed [4]. These approaches generally make a homoscedastic assumption without considering GRN structure variations. The regression-based methods with feature selection (e.g., L1-regularization) identify a subset of transcription factors (TFs) that are the most informative to predict the expression level of a target gene [5, 6]. The correlation-based approaches compute averaged variation in gene expressions across different experimental conditions. The Pearson and Spearman correlations are the common measures. Mutual information is introduced to capture nonlinear relationships between gene expressions [7, 8] such as CLR and ARACNE. Tree-based ensemble methods and artificial neural networks (ANNs) are also implemented to estimate TF-target gene relationships [9]. Probabilistic graphical model approaches are applied to infer gene regulatory interactions via Bayesian network and Markov random field [10, 2]. These approaches rely on a locally defined neighborhood structure that does not directly capture potential long-range dependencies. For crime prediction, autoregressive mixture models with Poisson processes are proposed [11]. The most recent extension (PoINAR) incorporates a stochastic process prior to group spatial correlation modes across multiple time series, and achieves the-state-of-the-art performance [3].

A common method for estimating $cov(\mathbf{y}|x) = \Sigma(x)$ applies regression models to the entries of the log or Cholesky decomposition of $\Sigma(x)$ or $\Sigma(x)^{-1}$ [12]. The method is computational expensive to high dimensional applications due to fitting $d(d+1)/2$ separate regression models. Besides, multivariate stochastic volatility models and Wishart processes are proposed to capture $\Sigma(x)$ as it evolve over time to form a multivariate time series [13, 14, 15]. Their Markov assumption causes the failure to capture long-range dependencies. Scaling to high-dimensional data is still a problem. An extension to estimate the covariance function $cov(\mathbf{y}|x)$ formulates the factor loading matrix as a linear combination of the predictor variable x, which limits the model's flexibility [16]. Another relevant work used compactly supported covariance functions to model short-scale variability to encode sparsity, and factorize the covariance functions for spatial and temporal domains to reduce the complexity [17]. To exclude long-scale variability seems not sensible for experiment-dependent gene network inference, and the complexity from GP prior part still depends on data dimension d.

3 Sparse covariance modeling

We develop a parsimonious likelihood to model correlations among components of covariate-dependent high-dimensional observation vectors. The factor loading matrix is further factorized into a collection of basis functions and their coefficient matrix. By relating the covariate-dependent basis elements with a sparse Gaussian process prior, we are able to capture variations in observation elements' correlations across the covariates.

3.1 Multivariate covariance likelihood specification

We characterize observation vectors $\mathbf{y}_c$ by formulating a multivariate Gaussian covariance model as

$$p(\mathbf{y}_c) = N(\mu_{\mathbf{y}}(x_c), \Sigma_{\mathbf{y}}(x_c)), \qquad c = 1, \ldots, n \tag{1}$$

with $\mathbf{X} = \{x_1, \ldots, x_n\}$ a set of covariates (e.g., indexes of experimental conditions), and $\mathbf{y}_c$, an observation vector indexed by x_c, the dimension of which is $dim(\mathbf{y}) = d$.

We explain the correlations in elements of $\mathbf{y}_c$ by assuming some latent variables z_c, and take the above model to be induced through the latent factor model

$$p(\mathbf{y}_c|\mathbf{z}_c) = N(A(x_c)\mathbf{z}_c, \Sigma_0) \qquad p(\mathbf{z}_c) = N(\mu_{\mathbf{z}}(x_c), I_k) \tag{2}$$

where $A(x_c)$ is a $d \times k$ factor loading matrix specific to x_c, $\mathbf{z}_c = (z_{c1}, \ldots, z_{ck})^T$ are latent factors associated with $\mathbf{y}_c$, and I_k denotes a $k \times k$ identity matrix. We let $\Sigma_0 = diag(\sigma_1^2, \ldots, \sigma_d^2)$, the diagonal elements of which are independently generated from inverse chi-square priors $\sigma_i^2 \sim inv\chi^2(\nu_0, \sigma_0^2)$, where $i = 1, \ldots, d$.

The latent factor model provides a lower dimensional description of the observation vectors. In particular, the marginal distribution of $\mathbf{y}_c$ is

$$p(\mathbf{y}_c) = \int p(\mathbf{y}_c|\mathbf{z}_c)p(\mathbf{z}_c)d\mathbf{z}_c = N(A(x_c)\mu_{\mathbf{z}}(x_c), A(x_c)A^T(x_c) + \Sigma_0) \tag{3}$$

$\mathbf{z}_c$ represents a latent subspace indexed by x_c that captures the observations' statistical variability, $\mu_{\mathbf{z}}(x_c)$ enables these latent factors to change over $\{x_c\}$, and $A(x_c)$ is a low-rank description of the observation element correlations subject to x_c.

3.2 Combining Gaussian processes with a sparse prior

In order to characterize $\mu_{\mathbf{z}}(x)$ and $A(x)$ changing across $\{x_c\}$ and enable information sharing, we use Gaussian process priors to generate the sets of basis functions for them, respectively.

For $\mu_{\mathbf{z}}(\mathbf{X}) = \{\mu_{z_1}(\mathbf{X}), \ldots, \mu_{z_k}(\mathbf{X})\}$, we have

$$\mu_{z_j}(x) \sim GP(\mathbf{0}, \kappa_\mu(x, x')) \tag{4}$$

with $\kappa_\mu(x, x')$ a positive definite kernel function, which is defined as $\sigma_{zv}^2 \exp(-\frac{1}{2\sigma_{zh}^2}||x - x'||_2^2)$. The hyper-parameters σ_{zv}^2 and σ_{zh}^2 are vertical and horizontal scales of the Gaussian process. k indicates the latent factor dimension. For any finite set of xs, this process defines a joint Gaussian:

$$p(\mu_{z_j}|\mathbf{X}) = N(\mathbf{0}, K_\mu) \tag{5}$$

where $K_{cc'}^{\{\mu\}} = \kappa_\mu(x_c, x_{c'})$.

The method of using Gaussian process priors for $d \times k$ elements of $A(\mathbf{X})$ can be computationally expensive given large d [17]. We instead factorize the factor loading matrix into a $d \times t$ coefficient matrix B and a $t \times k$ matrix of basis function elements $U(\mathbf{X})$,

$$A(\mathbf{X}) = BU(\mathbf{X}) \tag{6}$$

where $B \in R^{d \times t}$ and $U(\mathbf{X}) = \{u_{lj}(\mathbf{X})\}_{l=1,\ldots,t;j=1,\ldots,k}$. Thus, t specifies the size of the basis functions given k. We then let $k \ll d$ and $t \ll d$. Instead of directly defining a GP prior for each element of the $d \times k$ matrix $A(\mathbf{X})$, this factorization enables our method to scale to high dimensions d. These basis functions are generated from independent Gaussian process priors,

$$u_{lj}(x) \sim GP(\mathbf{0}, \kappa_u(x, x')) \tag{7}$$

where $\kappa_u(x, x') = \sigma_{uv}^2 \exp(-\frac{1}{2\sigma_{uh}^2}||x - x'||_2^2)$ is the kernel function. Analogously to that of μ_{z_j}, this process defines a joint Gaussian for any finite set of points:

$$p(u_{lj}|\mathbf{X}) = N(\mathbf{0}, K_u) \tag{8}$$

where $K_{cc'}^{\{u\}} = \kappa_u(x_c, x_{c'})$.

To encode sparsity for the set of basis functions $u_{lj}(\mathbf{X})$ where $t \to \infty$ in theory, we employ ARD to explore how the basis contribute to the factor loading. Without a shrinkage prior, it leads to full $A(x)$ matrices. This becomes problematic as the number of bases grows in the presence of limited data, and we cannot identify irrelevant basis elements. ARD addresses these issues by encouraging the number of the basis to zero, if their presence is not supported by the data. Specifically, we define independent, zero-mean, spherically symmetric Gaussian priors on the columns of the coefficient matrix B:

$$p(B|\theta) = \prod_{l=1}^{t} N(\mathbf{b}_{\cdot l}; \mathbf{0}, \theta_l^{-1} I_d) \tag{9}$$

Each precision parameter θ_l is given a $\Gamma(\alpha_\theta, \beta_\theta)$ prior. ARD method penalizes non-zero columns of the coefficient matrix by an amount determined by the precision parameters. Iterative estimation of these hyperparameters θ_l and the coefficient matrix B leads to θ_l becoming large for columns whose evidence in the data is insufficient for overcoming the penalty induced by the prior. Having $\theta_l \to \infty$ drives $\mathbf{b}_{\cdot l} \to 0$, which implies that the corresponding basis in the l^{th} row of $U(\mathbf{x})$ does not contribute to the factor loadings.

Theorem 1 *Assume $\mathbf{X}$ is compact, the induced prior $\Pi_U \bigotimes \Pi_B$ on $\{\tilde{\Sigma}(x), x \in \mathbf{X}\}$, where the priors Π_U for $U(\mathbf{X})$ and Π_B for B are specified in 8 and 9, can generate a covariance function $\tilde{\Sigma} : \mathbf{X} \to M^+$ arbitrarily close to any continuous function $\Sigma : \mathbf{X} \to M^+$, with M^+ the space of $n \times n$ positive semidefinite matrices. That is, for $\forall \epsilon > 0$ and $\tilde{k} \geq k$, $\Pi_{\tilde{\Sigma}}(\sup_{x \in \mathbf{X}} ||\tilde{\Sigma}(x) - \Sigma(x)||_2 < \epsilon) > 0$.*

k is the factor dimension of any continuous function mapping from compact domain to positive semidefinite matrices, and $\tilde{k}$ is the factor dimension of the ones generated by our model. The theorem indicates our covariance modeling method's expressive capability of covering the ground-truth matrices.

3.3 Inference via Gibbs sampling

We develop a Gibbs sampling solution to iteratively sample the marginalized basis functions of the covariances and the mean functions given their priors and the observations, and then update the hyper-parameters given the basis functions and the observations.

First, our model's joint probability can be factorized as

$$\begin{aligned} &p(\{\mathbf{y}_c\}, \{\mathbf{z}_c\}, \mu_\mathbf{z}, B, U, \Sigma_0, \theta) \\ &\propto \prod_{c=1}^{n} [p(\mathbf{y}_c|\mathbf{z}_c, B, U, \Sigma_0) \prod_{j=1}^{k} p(z_{cj}|\mu_{z_j})] \prod_{i=1}^{d} p(\sigma_i^2) \prod_{j=1}^{k} [p(\mu_{z_j}) \prod_{l=1}^{t} p(u_{lj})] \prod_{l=1}^{t} [p(\mathbf{b}_{\cdot l}|\theta_l) p(\theta_l)] \end{aligned} \tag{10}$$

We then sample the latent variables from their respective posteriors.

To sample a mean function μ_{z_j} for each latent factor, we readily marginalize over $\{\mathbf{z}_c\}$ as in (3), and let $A_c = BU(x_c)$ and $\mathbf{a}_{\cdot j}^{(c)}$ denote the j^{th} column of A_c. Thus, the likelihood of $\mu_\mathbf{z}$ corresponding to x_c is

$$p(\mathbf{y}_c|\mu_\mathbf{z}(x_c), B, U(x_c), \Sigma_0) = N(\mathbf{a}_{\cdot j}^{(c)} \mu_{z_j}(x_c) + \mathbf{m}_j(x_c), A_c A_c^T + \Sigma_0) \tag{11}$$

where $\mathbf{m}_j(x_c) = (\sum_{r \neq j} a_{1r}^{(c)} \mu_{z_r}(x_c), \ldots, \sum_{r \neq j} a_{dr}^{(c)} \mu_{z_r}(x_c))^T$.

Let $G_j = diag(\mathbf{a}_{\cdot j}^{(1)}, \ldots, \mathbf{a}_{\cdot j}^{(n)})$ and $\Sigma_c = A_c A_c^T + \Sigma_0$, the posterior of $\mu_{z_j}(\mathbf{X})$ by assuming the Gaussian process prior as in (5) results in

$$\begin{aligned} &p(\mu_{z_j}(\mathbf{X})|Y, B, U(x_c), \Sigma_0) = N(\mu_{\mu_\mathbf{z}|Y}, \Sigma_{\mu_\mathbf{z}|Y}) \\ &\Sigma_{\mu_\mathbf{z}|Y}^{-1} = K_\mu^{-1} + G_j^T diag(\Sigma_1^{-1}, \ldots, \Sigma_n^{-1}) G_j \\ &\mu_{\mu_\mathbf{z}|Y} = \Sigma_{\mu_\mathbf{z}|Y} (G_j^T diag(\Sigma_1^{-1}, \ldots, \Sigma_n^{-1}) Y^{-\mathbf{m}_j}) \end{aligned} \tag{12}$$

where $Y^{-\mathbf{m}_j} = (\mathbf{y}_1 - \mathbf{m}_j(x_1), \ldots, \mathbf{y}_n - \mathbf{m}_j(x_n))^T$.

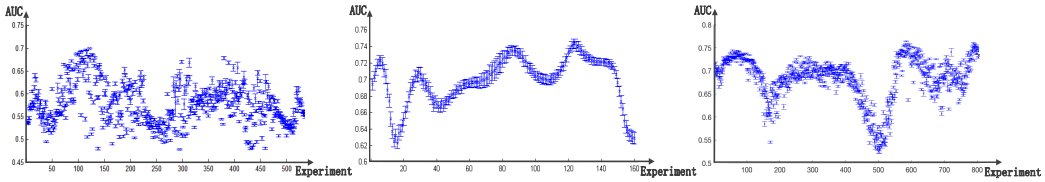
To sample each basis function element u_{lj} from its conditional posterior, its likelihood corresponding to x_c by combining (2) and (6) is

$$p(\mathbf{y}_c|\mathbf{z}_c, B, U(x_c), \Sigma_0) = N(z_{cj} \mathbf{b}_{\cdot l} u_{lj}(x_c) + \mathbf{m}_{lj}(x_c), \Sigma_0) \tag{13}$$

where $\mathbf{m}_{lj}(x_c) = (\sum_{(o,p) \neq (j,l)} z_{co} b_{1p} u_{po}(x_c), \ldots, \sum_{(o,p) \neq (j,l)} z_{co} b_{dp} u_{po}(x_c))^T$.

Let $H_{lj} = diag(z_{1j}\mathbf{b}_l, \ldots, z_{nj}\mathbf{b}_l)$, the following conditional posterior of the basis function $u_{lj}(\mathbf{x})$ can be derived from its Gaussian process prior in (8) and the likelihood in (13)

$$\begin{aligned} &p(u_{lj}(\mathbf{X})|Y, \{z_{cj}\}_c, B, U_{-lj}(\mathbf{X}), \Sigma_0) = N(\mu_{u|Y}, \Sigma_{u|Y}) \\ &\Sigma_{u|Y}^{-1} = K_u^{-1} + H_{lj}^T diag(\Sigma_0^{-1}, \ldots, \Sigma_0^{-1}) H_{lj} \\ &\mu_{u|Y} = \Sigma_{u|Y} (H_{lj}^T diag(\Sigma_0^{-1}, \ldots, \Sigma_0^{-1}) Y^{-\mathbf{m}_{lj}}) \end{aligned} \tag{14}$$



(a) ***S. cerevisiae*** (b) ***S. aureus*** (c) ***E. coli***

Figure 1: AUC scores change across experimental conditions, obtained by computing the experiment-specific gene correlations from the posterior estimate of covariance matrices $\Sigma_{\mathbf{y}}(x_c)$ using independent samples from 5000 to 6000 iterations of the 3 Gibbs sampling chains. The blue crosses indicate the sample means, and the error bars are the respective standard deviations. Each experiment-specific correlation matrix is a classifier aggregating statistical strength from all experiments.

where $Y^{-\mathbf{m}_{lj}} = (\mathbf{y}_1 - \mathbf{m}_{lj}(x_1), \ldots, \mathbf{y}_n - \mathbf{m}_{lj}(x_n))^T$. As in 12 and 14, information from data dominates the posteriors via the second terms. The role of the priors K_μ^{-1} and K_u^{-1} defined based on the index x_c are even weaker, since Gaussian process prior are symmetric and fully-connected.

Using conjugacy, we sample σ_i^2 from its conditional posterior

$$\begin{aligned} &p(\sigma_i^2|\{\mathbf{y}_c\}, \{\mathbf{z}_c\}, B, U(x_c)) = inv\chi^2(\nu_{\sigma_i|y}, \sigma^2_{\sigma_i|y}) \\ &\nu_{\sigma_i|y} = \nu_0 + n \qquad \sigma^2_{\sigma_i|y} = \frac{1}{\nu_{\sigma|y}}(\nu_0\sigma_0^2 + \sum_{c=1}^{n}(y_{ci} - \mathbf{b}_{i\cdot}U(x_c)\mathbf{z}_c)^2) \end{aligned} \tag{15}$$

Let $vec(B) = (\mathbf{b}_{\cdot 1}, \ldots, \mathbf{b}_{\cdot t})^T$ denotes the vectorization of the coefficient matrix B, and the likelihood in (13) can be re-written as

$$p(\mathbf{y}_c|\mathbf{z}_c, B, U(x_c), \Sigma_0) = N(\mathbf{m}_B^T vec(B), \Sigma_0) \tag{16}$$

where $\mathbf{m}_B = (\sum_{j=1}^k z_{cj}u_{1j}(x_c), \ldots, \sum_{j=1}^k z_{cj}u_{tj}(x_c))^T$.

More generally, our ARD prior on B in (9) is equivalent to a $N(\mathbf{0}, \Sigma_B)$ prior on $vec(B)$, where

$$\Sigma_B = diag(\theta_1 I_d, \ldots, \theta_t I_d)^{-1} \tag{17}$$

The posterior distribution of B is

$$\begin{aligned} &p(vec(B)|\mathbf{y}_c, \mathbf{z}_c, U(x_c), \Sigma_0) = N(\mu_{B|Y}, \Sigma_{B|y}) \\ &\Sigma_{B|Y}^{-1} = \Sigma_B^{-1} + M_B^T\Sigma_0^{-1}M_B \qquad \mu_{B|Y} = \Sigma_{B|Y}^{-1}(M_B^T\Sigma_0^{-1}\mathbf{y}_c) \end{aligned} \tag{18}$$

and $M_B = (\sum_{j=1}^k z_{cj}u_{1j}(x_c)I_d, \ldots, \sum_{j=1}^k z_{cj}u_{tj}(x_c)I_d)$.

Finally, given B and recalling that each precision parameter θ is gamma distributed, the posterior of θ_l is given by

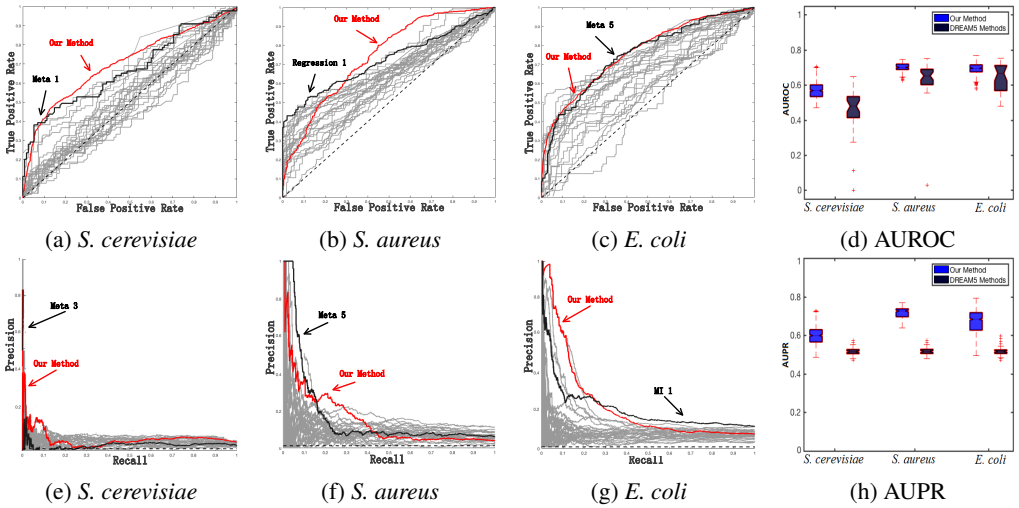
$$p(\theta_l|B) = \Gamma(\alpha_\theta + \frac{|S_l|}{2}, \beta_\theta + \frac{\sum_{i,l\in S_l} b_{il}^2}{2}) \tag{19}$$

The set S_l contains the indices for which b_{il} has prior precision θ_l.

Our proposed factorization reduces the computation of estimating $d(d+1)/2$ parameters to estimating $d \times t$ coefficients and $t \times k$ basis elements plus d variances. As long as $k \ll d$ and $t \ll d$, it is a substantial computation reduction. Since the ARD prior regularizes the dimension of the basis by shrinking the columns of B towards zero, the effective dimensions are small.

4 Experiments

We test our method across two domains. For GRN inference, we leverage variations in gene regulatory interactions across experimental conditions. For crime occurrence prediction, temporal variations in crime rate correlations among regions provide key information.



(a) *S. cerevisiae* (b) *S. aureus* (c) *E. coli* (d) AUROC

(e) *S. cerevisiae* (f) *S. aureus* (g) *E. coli* (h) AUPR

Figure 2: Performance evaluation for the three organisms. The ROC curves are in the top row. The PR curves are in the bottom. The ROC and PR curves of our method (red) are computed from the average experiment-specific correlations $\frac{1}{n}\sum_{c=1}^{n} r_c(I)$. We highlight the best performers of DREAM5 methods (black). Boxplots of the AUROC and AUPR scores of the experiment-specific correlations inferred by our method (blue) and the DREAM5 methods (black) are on the right.

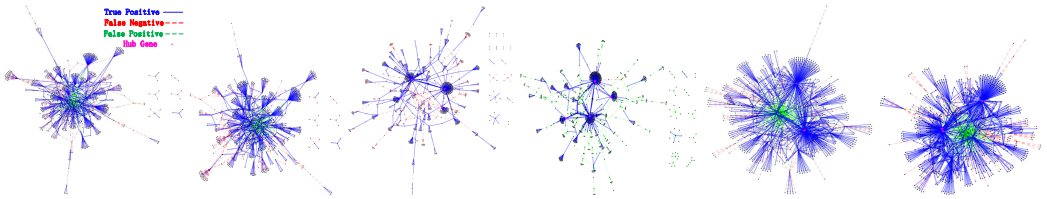
4.1 Gene expression data description

The datasets for gene expression compendia in *S. cerevisiae*, *S. aureus*, and *E. coli* are from the DREAM5 network inference challenge solicited predictions of genome-scale transcriptional regulatory networks [9]. Each compendium is represented as an expression matrix of d genes by n chip measurements. Organism-specific gold standards contain the known TF to target gene interactions which are true positives. All TF-target gene pairs that are not part of the gold standards are negatives. The challenge provides a total of 5,667 genes over 536 microarrays of *S. cerevisiae*, 4,297 genes over 805 microarrays of *E. coli*, and 2,677 genes over 160 microarrays of *S. aureus*. The microarray data sets are collected from a wide variety of experimental conditions with time series or gene deletion experiments, different perturbations, and various stress conditions [9]. Thus, $\mathbf{y}_c$ is a vector of the gene expression levels under experimental condition x_c.

4.2 MCMC settings

For each of the three organisms' gene expression datasets, we simulate 3 chains of 6000 Gibbs iterations, and discard the first 3000 as burn-in phase. Each sampling chain is initialized with parameters sampled from their priors. We set $\Gamma(\alpha_\theta, \beta_\theta)$ prior on the ARD precisions as $\alpha_\theta = |S_l|$ and $\beta_\theta = \alpha_\theta/1000$, where S_l is defined in (19). This prior specification is equally informative for various choices of effective coefficient number $|S_l|$ by fixing the prior mean of the prior distribution. We place a $inv\chi^2(1, 10)$ prior on the precision parameter σ_i^2, and set the vertical and horizontal scale hyper-parameters $\sigma_{\cdot v}^2 = 0.01$ and $\sigma_{\cdot h}^2 = 10$ in the Gaussian processes to account for the change rate of gene expression covariances across experiments. Since the ARD prior gives rise to a much smaller number of effective dimensions of the basis by shrinking the columns of B towards zero, we choose a finite k for computation efficiency. We find $k = 20$ and $t = 15$ to be sufficiently large as the last few columns of the posterior samples of B are consistently shrunk close to 0 for the three data sets.

We analytically marginalize the latent GP random functions to consider the posterior of the hyperparameters. The marginal likelihood is $(y_1^T, \cdots, y_n^T)^T | \sigma_h^2, B, z, \Sigma_0 \sim Norm(0, \sum_{l,j}(diag(z_{\cdot j}) \bigotimes b_{\cdot,l})K(diag(z_{\cdot j}) \bigotimes b_{\cdot,l})^T + I_n \bigotimes \Sigma_0)$ where K denotes the GP covariance matrix based on the hyperparameters. As proposed in [18], we then place grid points at the mode, and at a distance $\pm 1sd$ from the mode along each dimension to explore hyper-parameter sensitivity for the three data sets. we find that the results are the same with the length-scale and vertical scale hyper-parameters ranging up to $\sigma_{\cdot v}^2 = 100$ and $\sigma_{\cdot h}^2 = 0.005$ for less correlation across



(a) ***S. cerevisiae*** (b) ***S. aureus*** (c) ***E. coli***

Figure 3: GRN topological structures from our method (left) and the DREAM5 methods with highest AUROC scores (right) for the three organisms. The threshold is the median of the ranking scores of the regulatory interactions. For our model, the ranking scores are the estimated gene correlations. The black-colored nodes denote genes. The edges denote regulatory interactions. We highlight the true positive, false negative, and false positive connection predictions, and the top 4 hub genes.

experimental conditions with large values of $k = t = 50$. This suggests the robustness of our model to the choice of the hyper-parameters. We perform the Gelman-Rubin diagnostic [19] to assess convergence by calculating the within-chain and between-chain variances on the Gibbs samples of the posterior.

4.3 GRN inference

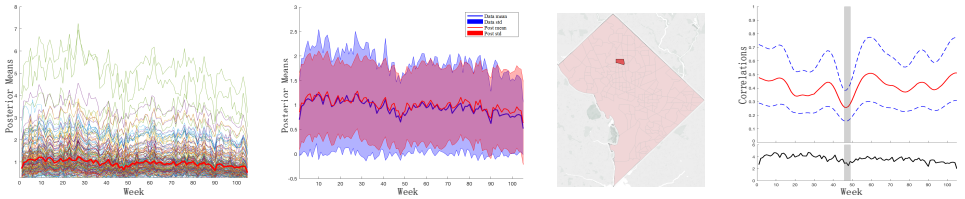
We evaluate network predictions as a binary classification task in terms of edges (regulatory interactions) predicted to be present or absent, and use standard performance metrics, receiver operating characteristic (ROC) curves and precision-recall (PR) curves.

We first evaluate predictions across experimental conditions via area under the ROC curve (AUROC) scores summarizing the performance of each experiment-specific correlation matrix. In Figure 1 (a)-(c), we plot the AUROC scores vary across experimental conditions for the three organisms. The scores are computed by comparing the ranked lists of gene correlations from the experiment-specific covariance matrices $\Sigma_{\mathbf{y}}(x_c)$s against the binary gold standards. The results indicate that the predictive performances of the gene expression correlations are experiment-dependent.

To integrate the predictions of the experiment-specific correlation matrices inferred by our method, we compute an average rank assigned to a possible gene regulatory interaction I as $\frac{1}{n}\sum_{c=1}^{n} r_c(I)$, where $r_c(I)$ is the correlation of I from experiment x_c, as in Figure 2 (a)-(c) and (e)-(g). In particular, we compare our method with 35 individual methods for GRN inference [1]. The methods are classified into six categories (method details are in [9]): regression, mutual information (MI), correlation, Bayesian networks, meta (methods that combine different approaches) and other (methods that do not belong to any of the previous categories, e.g., random forest, ANOVA, and ANNs). In Figure 2, the ROC curves and the PR curves show that our method outperform the best DREAM5 methods for the three organisms. The DREAM5 method with the highest AUROC score for *S. cerevisiae* is Meta 1 which is re-sampling from z-scores for target genes in TF knockouts, time-lagged CLR, and linear ordinary differential equations. For *S. aureus*, the best DREAM5 method is the Regression 1(TIGRESS) [6] combining sparse linear regression with data re-sampling. The best DREAM5 method for *E. coli* is Meta 5. This method combines Pearson's correlation, differential expression, time-series analysis, and naive Bayes. None of these methods model GRN's experiment-dependent variations or share the statistical strength among different conditions with a stochastic process.

Figure 2 (d) and (h) shows the boxplots of the AUROC and the AUPR scores of our method's experiment-specific correlation matrices and the 35 DREAM5 methods across the three organisms. Our method performs robustly better with less variations. The AUPR scores of our method are significantly better than the DREAM5 methods. In Figure 3, we plot the GRN topological structures inferred by our method and the best DREAM5 performers, and highlight the correctly identified hub genes with the highest connection degrees. As shown in Figure 3 (b), our method has less false positives (green dashed connections) than Regression 1, and achieves significantly higher AUPR

[1] The DREAM5 methods' predictions are on `https://www.synapse.org/#!Synapse:syn2787209/wiki/70351`.



(a) (b) (c) (d)

Figure 4: (a) Posterior means of the mean function $\mu_{\mathbf{y}}(X)$ for each of the 180 census tracts in Washington, D.C. The thick red line indicates the mean of the crime rates across the tracts. (b) Comparison between the posterior mean function $\mu_{\mathbf{y}}(\mathbf{X})$ and the posterior variance $\Sigma_{\mathbf{y}}(\mathbf{X})$ (red) and the data mean and variance (blue) over the weeks. (c) Census tract 2502, which has highest average correlations with the other tracts, is highlighted in the map of the 180 census tracts in Washington D.C. (d) For census tract 2502, the 25th, 50th, and 75th quantiles of correlation with the 179 other tracts based on the posterior variance $\Sigma_{\mathbf{y}}(\mathbf{X})$. The black line below is weekly averaged crime counts.

scores. This is consistent with the results in Figure 2 (f) and (h). The AUPRs are more informative than the AUROCs, since in high-throughput analysis methods with high precision are preferable.

4.4 Crime event prediction

We apply our covariance modeling method to capture spatio-temporal dependence of crime rates in the 180 census tracts in Washington, D.C. between 2016-2018 for crime occurrence prediction[2]. We analyze the crime rates on a weekly basis, with totally 105 weeks. For our modeling, the observation $\mathbf{y}_c$ denotes a vector of crime rates in the tracts in week $x_c \in \mathbf{X}$ with $\mathbf{X}$ as a set of discrete time indexing the weeks. So we have $dim(\mathbf{y}_c) = 180$ and $n = 105$, respectively.

We follow the model setting strategy as in Section 4.2. In particular, we simulate 3 chains each for $6,000$ MCMC iterations, and discard the first $3,000$ for burn-in. The latent factor dimension $k = 15$ and $t = 10$ with the Gaussian process hyperparameters $\sigma_{.h}^2 = 100$ are sufficiently large to account for the crime rates' temporal variations. For the qualitative analysis, Figure 4 (a) shows the trends of the posterior mean of the 180 components of $\mu_{\mathbf{y}}(x)$ follows the mean of the observed crime rates over time. The results in Figure 4 (b) indicate that we are able to capture both the mean and the variance of the crime rates in the tracts across the weeks $\mathbf{X}$ via the posterior mean estimate $\mu_{\mathbf{y}}(\mathbf{X})$ and the posterior covariance estimate $\Sigma_{\mathbf{y}}(\mathbf{X})$. It demonstrates the flexibility of joint mean-covariance estimation by our method. In Figure 4 (d), for census tract 2502 highlighted in Figure 4 (c), we notice that its correlations with the 179 other tracts vary across the weeks. The shaded gray region indicate the time period of its smallest correlation, and it corresponds to a lowest crime count on average.

We predict the one-week-ahead crime rates in each tract for the first 16 weeks in 2018 based on the estimated weekly covariance matrices in 2016-2017. To estimate the posterior predictive of the weekly crime rate $\mathbf{y}^*$ in 2018, we compute the basis function elements $\{u_{lj}(\mathbf{x}^*)\}$ conditioned on $\{u_{lj}(x_c)\}$, where $\{x_c\}$ indexing the weeks in 2016-2017, with the coefficient matrix B estimated from 2016-2017 by averaging over the Gibbs samples. Table 1 shows the monthly-averaged prediction RMSE, conditioned on the data in 2016-2017. For PoINAR, we use the same setting as in [3]. For Gaussian process regression (GPR), since we model the tracts independently over time, it cannot capture the spatial correlations. Hoff et.al. [16] method overly simplifies the time-varying covariances by making linear assumptions on x_c. The results indicate that our method produces lower RMSE. In Figure 5 (a)-(d), we compare the crime rate maps predicted by our method and the ground truth.

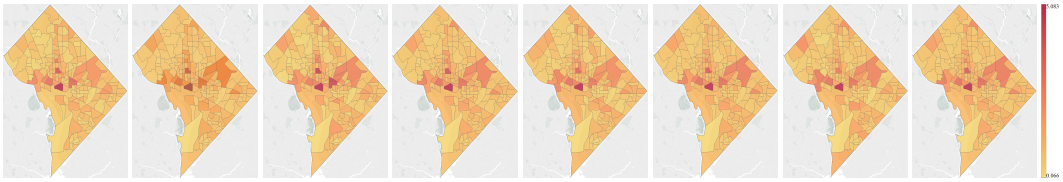
5 Conclusion

We propose a novel sparse covariance modeling method leveraging Gaussian processes to share information among covariate-dependent covariance matrices, and utilizing ARD priors on the coefficients to encode sparsity. We validate our model across domains and demonstrate its superior performance over the state-of-the-art methods. Our method scales well to high-dimensional observations. The in-

[2]The crime data are available on `http://crimemap.dc.gov/CrimeMapSearch.aspx`

Table 1: Monthly average RMSE of one-week-ahead predictions of the crime rates in 2018.

$RMSE \pm error$	Jan. 2018	Feb. 2018	Mar. 2018	April 2018
Our method	0.652 ± 0.018	0.726 ± 0.021	0.863 ± 0.028	0.807 ± 0.035
PoINAR [3]	0.899 ± 0.017	0.851 ± 0.014	0.912 ± 0.086	1.165 ± 0.006
GPR	1.582 ± 0.088	1.826 ± 0.157	1.649 ± 0.175	2.499 ± 0.204
Hoff et.al. [16]	1.751 ± 0.191	1.413 ± 0.168	2.198 ± 0.234	1.906 ± 0.209



(a) Jan. 2018 (b) Feb. 2018 (c) Mar. 2018 (d) April 2018

Figure 5: Monthly averaged crime rates maps of the ground truth (left) and the corresponding maps of predictive posterior mean rates $\mu_{\mathbf{y}}(\mathbf{X})$ using the samples from 5000 to 6000 iterations of the 3 Gibbs sampling chains (right) in 2018. The color scale is on the right.

ference algorithm involves sampling bases from an n-dimensional multivariate Gaussian distribution. For very large n (e.g., number of experiments), the computations are still $O(n^3)$ in general. One of our future work is to develop standard tools for scaling up this Gaussian process computation.

6 Aknowledgement

This work was supported by the National Science Foundation (1062422 to A.H.) and the National Institutes of Health (R15GM116102 to F.C.).

References

[1] David A. Garfield and Gregory A. Wray. The evolution of gene regulatory interactions. *BioScience*, 60(1):15–23, 2010.

[2] Min Zou and Suzanne D. Conzen. A new dynamic bayesian network approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*, 21(1):71–79, August 2004.

[3] Sivan Aldor-Noiman, Lawrence D. Brown, Emily B. Fox, and Robert A. Stine. Spatio-temporal low count processes with application to violent crime events. *Statistica Sinica*, 26(8):1587–1610, December 2016.

[4] Bing He and Kai Tan. Understanding transcriptional regulatory networks using computational models. *Current Opinion in Genetics and Development*, 37:101–108, March 2016.

[5] Timothy S. Gardner, Diego di Bernardo, David Lorenz, and James J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 31(5629):102–109, April 2003.

[6] Anne-Claire Haury, Fantine Mordelet, Paola Vera-Licona, and Jean-Philippe Vert. Tigress: Trustful inference of gene regulation using stability selection. *BMC Systems Biology*, 6(1):145, April 2012.

[7] Atul J. Butte and Isaac S. Kohane. Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements. *Pacific Symposium on Biocomputing*, 5:418–429, February 2000.

[8] Adam A. Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7(Suppl. 1):S7, March 2006.

[9] Daniel Marbach, James C. Costello, Robert Kuffner, Nicole M. Vega, Robert J. Prill, Diogo M. Camacho, Kyle R. Allison, The DREAM5 Consortium, Manolis Kellis, James J. Collins, and Gustavo Stolovitzsky. Wisdom of crowds for robust gene network inference. *Nature Methods*, 9(8):796–804, December 2012.

[10] Adriano V. Werhli and Dirk Husmeier. Reconstructing gene regulatory networks with bayesian networks by combining expression data with multiple sources of prior knowledge. *Statistical Applications in Genetics and Molecular Biology*, 6(1):15–61, 2007.

[11] Matthew A. Taddy. Autoregressive mixture models for dynamic spatial poisson processes: Application to tracking intensity of violent crime. *Journal of the American Statistical Association*, 105(492):1403–1417, January 2010.

[12] Weiping Zhang and Chenlei Leng. A moving average cholesky factor model in covariance modelling for longitudinal data. *Biometrika*, 99(1):141–150, December 2011.

[13] Andrew Harvey, Esther Ruiz, and Neil Shephard. Multivariate stochastic variance models. *The Review of Economic Studies*, 61(2):247–864, April 1994.

[14] Christian S. Gourieroux, Joann Jasiak, and Razvan Sufana. The wishart autoregressive process of multivariate stochastic volatility. *Journal of Econometrics*, 150(2):167–181, June 2009.

[15] Rui Li, Jared Curhan, and M.E. Hoque. Understanding social interpersonal interaction via synchronization templates of facial events. In *AAAI*, pages 1579–1586, February 2018.

[16] Peter D. Hoff and Xiaoyue Niu. A covariance regression model. *Statistica Sinica*, 22:729–953, 2012.

[17] Jaakko Luttinen and Alexander Ilin. Variational gaussian-process factor analysis for modeling spatio-temporal data. In *NIPS*, pages 1177–1185, December 2009.

[18] Havard Rue and Sara Martino. Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of Royal Statistical Society*, 71(2):1–35, January 2009.

[19] Stephen P. Brooks and Andrew E. Gelman. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, November 1998.