Sparse Gaussian graphical models for dynamic gene regulatory networks

Veronica Vinciotti
Brunel University London

Dynamic Networks
Isaac Newton Institute

16 December 2016

Joint work with Ernst Wit
Features of a Dynamic Genomic Process

- **Structure:**
  - Highly complex and structured phenomenon.
  - Possibly with additional topographical structure (small world).

- **Sparsity:** Only a small number of links between nodes.
What is a gene regulatory network?

“A collection of DNA segments in a cell which “interact” with each other via their RNA or proteins and with other substances in the cell, thereby governing the rates at which genes in the network are transcribed.”
Data

- Observations are on nodes/variables, not on edges/relationships.
  - microarray, RNA-seq data, mass spectrometry data, ...

- Typically many variables, few units ("p >> n")

- Data can be measured over time (with/without destructive sampling)

This talk: network inference from longitudinal data.
Sparse Gaussian Graphical Models

A popular tool for inference of networks from biological data.

Definition

A GGM is made of:

- A graph $G = (\Gamma, E)$, where $\Gamma$ is a set of $p$ genes.
- Genomic interactions $E \subset \Gamma \times \Gamma$, which determine non-zeros in $\Theta = \Sigma^{-1}$,

\[
\mathbf{Y}_i \sim N(\mu, \Sigma), \quad i = 1, \ldots, n.
\]

For $p > n$, the precision matrix $\Theta$ can be estimated under an $L_1$ penalty (Friedman et al 2008).
For very large \( p \) and given some known partitioning of the nodes, one can assume a very constrained covariance \( \Sigma \):

\[
\Sigma = \begin{pmatrix}
\sigma_1 & \sigma_{21} \mathbf{1}_M & \ldots & \sigma_{1G} \mathbf{1}_M \\
\sigma_{12} \mathbf{1}_M & \sigma_2 & \ldots & \sigma_{2G} \mathbf{1}_M \\
\vdots & \vdots & \ddots & \vdots \\
\sigma_{G1} \mathbf{1}_M & \sigma_{G2} \mathbf{1}_M & \ldots & \sigma_G
\end{pmatrix},
\]

\[
\sigma_g = \begin{pmatrix}
\delta_g & \sigma_{gg} & \ldots & \sigma_{gg} \\
\sigma_{gg} & \delta_g & \ldots & \sigma_{gg} \\
\vdots & \vdots & \ddots & \vdots \\
\sigma_{gg} & \sigma_{gg} & \ldots & \delta_g
\end{pmatrix}_{M \times M}
\]

This results in a block-wise structure also for the inverse covariance \( \Theta \) and a network structure at the level of the groups (e.g. Luo (2014) for an application on brain networks).
Dynamic Graphical Models

Definition (Dynamic genomic Gaussian Graphical Model)

... is a Gaussian graphical model defined on:

- a dynamic graph $G = (\Gamma \times T, E)$, where $\Gamma$ is a set of genes; $\Gamma \times T$ the nodes of the graph
- Genomic interactions $E \subset \Gamma T \times \Gamma T$, which determine non-zeros in $\Theta$

$$Y \sim \mathcal{N}(\mu, \Theta^{-1}).$$
Structural Constraints: An Example

- Example: four genes, two time points

- $N_0$: edges within the two groups, excluding self-links, represent the instantaneous or lag zero network.
- $N_1$: edges between the two groups, excluding links between the same unit, represent the lag 1 network.
- $S_0$, $S_1$: self-self interaction with lag 0 and lag 1, respectively.
- We impose constraints on the structure, e.g. in this example

\[(S_0 \prec 1), N_0 \prec T, S_1 \prec 1, N_1 \prec 0.\]
In analogy with ANOVA, we define the following structured graphs (within each partition):

- **Zero Model 0** ⇒ Empty graph.
- **Constant Model 1**
- **Main Time Effect Model** $T$
- **Main Unit Effect Model** $\Gamma$
- **Interaction Effect Model** $\Gamma T$

![Graphical illustration of models](https://via.placeholder.com/150)
In analogy with ANOVA, we define the following structured graphs (within each partition):

- **Zero Model 0**
- **Constant Model 1** $\Rightarrow \theta_{it,js} = c.$
- **Main Time Effect Model $T$**
- **Main Unit Effect Model $\Gamma$**
- **Interaction Effect Model $\Gamma T$**
Structural Constraints: In General

In analogy with ANOVA, we define the following structured graphs (within each partition):

- Zero Model 0
- Constant Model 1
- Main Time Effect Model $T \Rightarrow \theta_{it,js} = c_{ts}$
- Main Unit Effect Model $\Gamma$
- Interaction Effect Model $\Gamma T$

\[\begin{aligned}
\text{Time 1} & \\
v_{11} & \quad v_{21} & \quad v_{31} \\
& \quad c_1 \\
\text{Time 2} & \\
v_{12} & \quad v_{22} & \quad v_{32}
\end{aligned}\]
In analogy with ANOVA, we define the following structured graphs (within each partition):

- Zero Model 0
- Constant Model 1
- Main Time Effect Model $\Gamma T$
- **Main Unit Effect Model** $\Gamma \Rightarrow \theta_{it,js} = c_{ij}$
- Interaction Effect Model $\Gamma T$

![Diagram](image)
In analogy with ANOVA, we define the following structured graphs (within each partition):

- **Zero Model 0**
- **Constant Model 1**
- **Main Time Effect Model T**
- **Main Unit Effect Model Γ**
- **Interaction Effect Model ΓT \( \Rightarrow \theta_{it,js} = c_{itjs} \)**

![Diagram showing structural constraints]

Veronica Vinciotti  
Graphical Models for Regulatory Networks
Model:

\[(S_0 < 1), \quad N_0 \prec T, \quad S_1 \prec 1, \quad N_1 \prec 0.\]

Precision Matrix:

\[
\Theta = \begin{bmatrix}
\theta_1 & \theta_2 & \theta_2 & \theta_2 & \theta_4 & 0 & 0 & 0 \\
\theta_1 & \theta_2 & \theta_2 & \theta_2 & 0 & \theta_4 & 0 & 0 \\
\theta_1 & \theta_2 & 0 & 0 & \theta_4 & 0 \\
\theta_1 & 0 & 0 & \theta_4 & 0 \\
\theta_1 & \theta_3 & \theta_3 & \theta_3 & \theta_1 \\
\theta_1 & \theta_3 & \theta_3 & \theta_3 & \theta_1 \\
\theta_1 & \theta_3 & \theta_3 & \theta_3 & \theta_1 \\
\theta_1 & \theta_3 & \theta_3 & \theta_3 & \theta_1 \\
\end{bmatrix}
\]
Likelihood under Structural Constraints and Sparsity

**Likelihood:**

\[
\ell(\Theta) \propto \frac{n}{2} \left\{ \log |\Theta| - \text{tr}(S\Theta) \right\},
\]

with \( S \) the sample covariance matrix.

**Optimization of penalized likelihood:**

\[
\hat{\Theta}_\rho := \arg\max_\Theta \{ \ell(\Theta) \}
\]

subject to

- \( \Theta \geq 0; \)
- \( \|\Theta\|_1 \leq \rho; \)
- structural constraints.

Network \( \hat{\Theta}_\rho \) selected via a model selection criterion.
Algorithm for Parameter Estimation

- LogdetPPA algorithm combines a proximal point algorithm (PPA) inside a preconditioned conjugate gradient solver needed for Newton’s method (Wit and Abbruzzo, 2015).
  - Fast for a single tuning parameter, but not efficient overall across a path of solutions.

- Cyclic coordinate descent method across a path of solutions
  - Implemented in the R package `sglasso`.
  - Function `fglasso` specifically for dynamic networks.
  - Algorithm described in Vinciotti, Augugliaro, Abbruzzo, Wit (2016).
Gene expression data on Neisseria meningitidis wildtype.

We consider 60 proteins, previously known to be associated with a master regulator FarR.

For each gene, there are 4 replicates across 10 time points. This results in a $600 \times 600$ covariance matrix: about 180,000 parameters to estimate from 2,400 observations in total!

We impose structural constraints:
- Up to first order time dependencies.
- Gene-specific interaction strengths, constant within time lags.

This reduces number of parameters to about 5500 in total.

Shrinkage induced by $L_1$ penalty further stabilizes estimates.

Data generated by Prof. Nigel Saunders, Brunel University London
Neisseria: Final Network

- Lag 0: (static) interactions within a time point $t$
- Lag 1: (dynamic) interactions from $t$ to $t + 1$.
- Red: -ve, Blue: +ve partial correlation.

Lag 0 Regulatory Network

Lag 1 Regulatory Network

Veronica Vinciotti
Graphical Models for Regulatory Networks
Dynamic models are more naturally thought in terms of Markovian dynamics.

If one assumes first-order Markov properties and a Gaussian conditional distribution of present given past, then

\[ Y_t | Y_{t-1} \sim N_p(\Gamma Y_{t-1}, \Theta^{-1}) \] (Abegaz and Wit, 2013).

- \( \Theta_{p \times p} \): instantaneous dependencies, \( \Gamma_{p \times p} \): dynamic dependencies.
- R package \textit{SparseTSCGM}: VAR(1) and VAR(2), SCAD and L1 penalised inference.

Extension to time-varying \( \Gamma \) (Dondelinger et al, 2013).
- R package \textit{EDISON}: \( \Gamma \) varies across time-segments but \( \Theta \) is assumed diagonal, Bayesian inference.
Neisseria: Comparison of Methods on a Small Network

sglasso(lag 0) – BIC

TSCGM (Precision) – BIC

EDISON – NMB1994 change points

sglasso(lag 1) – BIC

TSCGM (Autoregressive) – BIC

EDISON (prob>0.4)
Neisseria: Comparison of Methods

sglasseo(lag 0) – BIC

TSCGM (Precision) – BIC

sglasseo(lag 1) – BIC

TSCGM (Autoregressive) – BIC

EDISON – NMB1994 change points

EDISON (prob>0.4)
Neisseria: Comparison of Methods

- **sglasso(lag 0) – BIC**
- **TSCGM (Precision) – BIC**
- **TSCGM (Autoregressive) – BIC**
- **EDISON (prob>0.4)**

---

**EDISON – NMB1994 change points**

- **EDISON − NMB1994 change points**

---

**Posterior Probability**

- **Posterior Probability**

---

**Veronica Vinciotti**

Graphical Models for Regulatory Networks
Does it make sense biologically? Can we find a simpler representation that could aid biological validation and generation of biological hypotheses?
Network Enrichment Test

- The inferred networks are very large, making biological validation tricky.
- **Network enrichment tests** can be helpful in extracting useful information and generating further hypotheses.
The inferred networks are very large, making biological validation tricky.

**Network enrichment tests** can be helpful in extracting useful information and generating further hypotheses.

Given two functional sets $A$ and $B$ (e.g. GO sets or KEGG pathways):

1. Test based on $n_{AB} = \text{number of links between genes } \in A \text{ and genes } \in B \text{ in a network.}$

2. $n_{AB}$ realization from the r.v. $N_{AB}$ with mean $\mu_{AB}$.

3. If there is a relationship between functions described by sets $A$ and $B$, then $\mu_{AB}$ should be higher/lower than $\mu_0$ (expectation in absence of enrichment).
Alexeyenko et al (2012) and McCormack et al (2013) propose the following strategy to test enrichment in a network:

1. Compute the number of links between nodes in $A$ and $B$, $n_{AB}$;
2. $n_{AB}$ realization from $N_{AB}$;
3. In absence of enrichment ($H_0$), $N_{AB} \sim N(\mu_0, \sigma_0^2)$;
4. Estimate $(\mu_0, \sigma_0^2)$ computing the mean and variance of $n_{AB}$ in a sequence of random permutations of the network.

**Drawbacks:**

1. Normal approximation for $N_{AB}$ discrete;
2. Time-consuming (especially for big networks);
3. Implemented for undirected networks only.
Alexeyenko et al (2012) and McCormack et al (2013) propose the following strategy to test enrichment in a network:

1. Compute the number of links between nodes in $A$ and $B$, $n_{AB}$;
2. $n_{AB}$ realization from $N_{AB}$;
3. In absence of enrichment ($H_0$), $N_{AB} \sim N(\mu_0, \sigma_0^2)$;
4. Estimate $(\mu_0, \sigma_0^2)$ computing the mean and variance of $n_{AB}$ in a sequence of random permutations of the network.

Drawbacks:

1. normal approximation for $N_{AB}$ discrete;
2. time-consuming (especially for big networks);
3. implemented for undirected networks only.
Our Proposal: Hypergeometric Distribution on $N_{AB}$

- Hypergeometric: # of successes in a random sample without replacement
  1. sample size: $n$
  2. population: $N$
  3. $K$ successes out of $N$ cases.

- In our case, let us consider
  1. success: an arrow reaches $B$;
  2. unsuccess: an arrow reaches $B^C$.

- In absence of enrichment from $A$ to $B$, $N_{AB}$ is the number of successes in a random sample where
  1. $n =$ arrows going out from $A$;
  2. $N =$ arrows in the graph;
  3. $K =$ arrows in the graph reaching $B$. 
Null Distribution (Directed Networks)

\[ H_0 : \mu_{AB} = \mu_0 \text{ (no enrichment) } \]
\[ H_1 : \mu_{AB} \neq \mu_0 \text{ (enrichment) } \]

**Parametric assumption:** in absence of enrichment \((H_0 \text{ true})\)

\[ N_{AB} \sim \text{hypergeom}(n = o_A, K = i_B, N = i_V), \]

where

- \(o_A\) is the total outdegree (number of outgoing arrows) of \(A\);
- \(i_B\) is the total indegree (number of incoming arrows) of \(B\);
- \(i_V\) is the total indegree of \(V = B \cup B^C\).

Implemented in the R package `neat`.

*Signorelli, Vinciotti and Wit (2016), BMC Bioinformatics, 17:352.*
\[ A = \{1, 4\}, \quad B = \{3, 5, 7\}. \quad o_A = 5 \rightarrow^B n_{AB} = 2 \ (40\%), \text{ whereas} \]
\[ i_V = 15 \rightarrow^B i_B = 4 \ (27\%). \text{ Also: } n_{AB} = 2 > \mu_0 = 1.33. \]

\textbf{neat test: } \[ p = 0.48 \rightarrow \textbf{Conclusion: NO enrichment!} \]
Real Application: *Glucose network*

- 94 RNA blood samples from individuals at extreme glucose fasting concentrations
- 1500 genes selected and Gaussian graphical network inferred
- Enrichment test for 62 functional groups defined by KEGG pathways
Conclusions

- We have presented Gaussian graphical models for dynamic network inference.
- Penalised likelihood approaches for sparsity and computational efficiency.
- Biological validation of the inferred networks via network enrichment tests.

References:
- Vinciotti, Augugliaro, Abbruzzo, Wit (2016) SAGMB, 15, 3, 193-212
- R package sglasso
- Signorelli, Vinciotti and Wit (2016), BMC Bioinformatics, 17:352
  R package neat

One more thing ...
COSTNET: a European Project on Network Science

A network of (currently) 200 scientists working in the field of networks

Chaired by Ernst Wit. Three working groups:
- WG1: Exploring (massive) network data sets (Leader: Clelia di Serio)
- WG2: Network Modelling (Leader: Steffen Lauritzen)
- WG3: Network Inference and Prediction (Leader: Arnoldo Frigessi)

Funding available until 30 April 2020 for networking events (conferences, training schools, STSMs, ...).
COSTNET: You can join too!

- Create an e-cost account

- Contact one of the WG leaders (Clelia di Serio, Steffen Lauritzen, Arnoldo Frigessi) to join a WG.