Networks and Sparse Graphical Models

Ernst Wit
e.c.wit@rug.nl

Johann Bernoulli Institute
University of Groningen

21 January, 2014
Motivation: Features of a Dynamic Genomic Process

- “p >> n”:
  - Number of observations smaller than number of variables.
  - Thousands of variables and hundreds of observations.
- **Structure**:
  - Highly complex and structured phenomenon.
  - Possibly with additional topographical structure (small world).
- **Sparsity**: only small number of links between nodes.
“Network” as Graphical Model

In Genomics: typically have measurements of nodes
Examples: RNA-seq, GWAS, proteomics

Proposal: interpret network as conditional independence relations.
Motivation: Dynamic Genomic Networks

- Transcription: snap-shot of gene activity in time and space.
- Microarray and RNA-seq data measure gene activity.

Running example: T-cell time-series dataset.
- Temporal expression of 58 genes for 10 spaced time points.
- At each time point there are 44 separate measurements.

Definition (Aim)

Determine dynamic genomic graph $G$ on basis of $\{Y_{gt}^{(i)}\}_{gti}$. 
Dynamic Genomic Graphs

- $\Gamma$ be a set of “genes”.
- $T$ be a set of ordered “time points”.

**Definition (Dynamic genomic graph)**

A dynamic genomic graph is a pair $G = (V, E)$.

- Vertices: $V = \{v_{ij}\}_{i \in \Gamma, j \in T}$, where $\Gamma$ and $T$ are finite sets.
- Links: ordered pair of elements $E \subseteq V \times V$. 
Definition (Coloured graph)

A coloured graph is a triplet $G_F = (V, E, F)$, where $G = (V, E)$ is a graph and $F$ is a mapping on the links, i.e.:

$$F : E \rightarrow C,$$

where $C$ is a finite set of colours.
Special kind of coloured graphs: Factorial Graphs

Denote mapping \( F : E \rightarrow C \) by \( E \prec F \).

In analogy with ANOVA, we define the following colouring:

- \( E \prec 0 \Rightarrow \) an empty graph.
- \( E \prec F_1 \).
- \( E \prec F_T \).
- \( E \prec F_{\Gamma} \).
- \( E \prec F_{\Gamma T} \).

\[
\begin{array}{c}
\text{Time 1} \\
v_{11} & \bullet \\
v_{21} & \bullet \\
v_{31} & \bullet \\
\text{Time 2} \\
v_{12} & \bullet \\
v_{22} & \bullet \\
\end{array}
\]
Denote mapping $F : E \rightarrow C$ by $E \prec F$.

In analogy with ANOVA, we define the following colouring:

- $E \prec 0 \Rightarrow$ an empty graph.
- $E \prec F_1$: same colour for all links
- $E \prec F_T$.
- $E \prec F_T$.
- $E \prec F_{TT}$.
Special kind of coloured graphs: Factorial Graphs

Denote mapping $F : E \rightarrow C$ by $E \prec F$.

In analogy with ANOVA, we define the following colouring:

- $E \prec 0 \Rightarrow$ an empty graph.
- $E \prec F_1$.
- $E \prec F_T$: same colour across all genes
- $E \prec F_\Gamma$.
- $E \prec F_{\Gamma T}$.

\begin{align*}
\begin{array}{c}
\text{Time 1} \\
\text{Time 2}
\end{array}
\end{align*}
Special kind of coloured graphs: Factorial Graphs

Denote mapping $F : E \rightarrow C$ by $E \prec F$.

In analogy with ANOVA, we define the following colouring:

- $E \prec 0 \Rightarrow$ an empty graph.
- $E \prec F_1$.
- $E \prec F_T$.
- $E \prec F_{\Gamma}$: same colour across all times
- $E \prec F_{\Gamma T}$. 

\[ \begin{array}{ccc}
  v_{11} & v_{21} & v_{31} \\
  c_1 & & \\
  v_{12} & v_{22} & v_{32} \\
  c_2 & & \\
  & c_1 & c_2 \\
\end{array} \]
Denote mapping \( F : E \to C \) by \( E \prec F \).

In analogy with ANOVA, we define the following colouring:

- \( E \prec 0 \Rightarrow \) an empty graph.
- \( E \prec F_1 \).
- \( E \prec F_T \).
- \( E \prec F_\Gamma \).
- \( E \prec F_{\Gamma T} \): all different colours.
Definition (Natural partition)

Let $E = \{S_i, N_i\}_{i=0}^{n_T-1}$ be subsets of links where $S_i, N_i$ are defined as follows:

$$S_i = \{((v_{jt}, v_{j,t+i}), (v_{j,t+i}, v_{jt})) | j \in \Gamma, t = 1, \ldots, n_T - i\},$$

and

$$N_i = \{((v_{jt}, v_{k,t+i}), (v_{k,t+i}, v_{jt})) | \forall j \neq k \in \Gamma, t = 1, \ldots, n_T - i\}.$$ 

The natural partitions imply subgraphs of $G$ and imply partitions of $\Theta$ for GGMs:

$$\Theta = \begin{bmatrix}
S_0 & N_0 & S_1 & N_1 & S_2 & N_2 & \ldots & \ldots \\
S_0 & N_1 & S_1 & N_2 & S_2 & \ldots & \ldots \\
S_0 & N_0 & S_1 & N_1 & S_2 & N_2 & \ldots & \ldots \\
S_0 & N_0 & S_1 & N_1 & S_2 & N_2 & \ldots & \ldots \\
S_0 & N_0 & S_1 & N_1 & S_2 & N_2 & \ldots & \ldots \\
S_0 & N_0 & S_1 & N_1 & S_2 & N_2 & \ldots & \ldots \\
\end{bmatrix}$$
A factorial Gaussian graphical model is a graphical model defined on:
- a dynamic factorial graph $G = (V, E, F)$, where
- a factorial colouring $F$ is applied separately to natural partitions

$$S_i \prec F_{S_i}, \quad N_i \prec F_{N_i}, \quad i = 0, \ldots, n_T - 1$$

- which determines $\Theta$ in

$$Y \sim N(\mu, \Theta^{-1}).$$
Example: Factorial Gaussian Graphical Model

Model:

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

Factorial coloured graph:

![Factorial coloured graph](image)

Precision Matrix:

\[
\Theta = \begin{pmatrix}
\theta_1 & \theta_2 & \theta_2 & \theta_2 & \theta_4 & 0 & 0 & 0 & 0 \\
\theta_1 & \theta_2 & \theta_2 & 0 & \theta_4 & 0 & 0 \\
\theta_1 & \theta_2 & 0 & 0 & \theta_4 & 0 \\
\theta_1 & 0 & 0 & 0 & \theta_4 \\
\theta_1 & \theta_3 & \theta_3 & \theta_3 \\
\theta_1 & \theta_3 & \theta_3 \\
\theta_1 & \theta_3 \\
\theta_1 &
\end{pmatrix}
\]
Consider an experiment: $|\Gamma|$ genes measured across $|T|$ time points.

Assume $n$ iid samples $y^{(1)}, \ldots, y^{(n)}$, where $y^{(i)} = (y_1^{(i)}, \ldots, y_{\Gamma T}^{(i)})$.

Assume $Y^{(i)} \sim N(0, \Theta^{-1})$, then

Likelihood:

$$l(\Theta|y) \propto \log(|\Theta|) - \text{tr}(S\Theta).$$

**AIM:** Optimization of penalized likelihood:

$$\hat{\Theta} := \arg\max_\Theta \{l(\Theta|y)\}$$

subject to

- $\Theta \succeq 0$;
- $\|\Theta\|_1 \leq 1/\lambda$;
- some factorial colouring $F$. 

William Occam (1288-1348) proposed a meta-theory of knowledge: “For nothing ought to be posited without necessity.”

Can be interpreted statistically as a

- **Aesthetic principle**: enhances model interpretability through parsimonious representation
- **Pragmatic principle**: computability.
- **Ontological principle**: represents expectation about nature of solution.
- **Prediction principle**: bias-variance trade-off
LogdetPPA. Newton-CG primal proximal point algorithm (Wang et al., 2010, including Kim Toh and Defeng Sun) is used to solve optimization:

$$\hat{\Theta} := \arg\min_{\Theta} \{ \log|\Theta| - \text{tr}(\Theta S) + \lambda'\theta^+ + \lambda'\theta^- : A(\Theta) = 0, $$

$$B(\Theta) - \theta^+ + \theta^- = 0, \Theta \succeq 0, \theta^+, \theta^- \geq 0 \}$$

- $A(\Theta)$: linear constraints which depend on coloured graph.
- $B(\Theta)$: $\ell_1$-norm penalty on elements of precision matrix.
- $\theta^+$ and $\theta^-$ are additional variables (slack variables).
- $\Theta \succeq 0$: semi-positive definite constraint.

Solves $\hat{\Theta}$ up to $2000 \times 2000$. 
Simulations: Lag 0 network identification

Table: \( n = 50 \) and \( t = 10 \) for various number of genes (\( p \))

<table>
<thead>
<tr>
<th>Method</th>
<th>Case 1 ( p = 50 )</th>
<th>Case 2 ( p = 100 )</th>
<th>Case 3 ( p = 50 )</th>
<th>Case 4 ( p = 100 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCAD</td>
<td>SEN</td>
<td>SPE</td>
<td>SEN</td>
<td>SPE</td>
</tr>
<tr>
<td></td>
<td>0.987</td>
<td>0.957</td>
<td>0.992</td>
<td>0.971</td>
</tr>
<tr>
<td></td>
<td>0.990</td>
<td>0.989</td>
<td>0.945</td>
<td>0.946</td>
</tr>
<tr>
<td>Distance</td>
<td>6.289</td>
<td>7.484</td>
<td>9.576</td>
<td>30.192</td>
</tr>
<tr>
<td>GLASSO</td>
<td>SEN</td>
<td>SPE</td>
<td>SEN</td>
<td>SPE</td>
</tr>
<tr>
<td></td>
<td>0.930</td>
<td>0.946</td>
<td>0.975</td>
<td>0.923</td>
</tr>
<tr>
<td></td>
<td>0.989</td>
<td>0.967</td>
<td>0.944</td>
<td>0.942</td>
</tr>
<tr>
<td>Distance</td>
<td>6.821</td>
<td>13.67</td>
<td>9.641</td>
<td>30.517</td>
</tr>
</tbody>
</table>
## Simulations: Lag 1 network identification

**Table:** \( n = 50 \) and \( t = 10 \) for various number of genes (\( p \))

<table>
<thead>
<tr>
<th></th>
<th>( p = 50 )</th>
<th>( p = 100 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCAD</td>
<td>SEN 0.998</td>
<td>0.948</td>
</tr>
<tr>
<td></td>
<td>SPE 0.985</td>
<td>0.899</td>
</tr>
<tr>
<td></td>
<td>Distance 0.602</td>
<td>1.665</td>
</tr>
<tr>
<td>EBDBN</td>
<td>SEN 0.343</td>
<td>0.195</td>
</tr>
<tr>
<td></td>
<td>SPE 0.615</td>
<td>0.793</td>
</tr>
<tr>
<td></td>
<td>Distance 37.910</td>
<td>52.048</td>
</tr>
<tr>
<td>GeneNet</td>
<td>SEN 0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>SPE 0.969</td>
<td>0.971</td>
</tr>
<tr>
<td></td>
<td>Distance 10.607</td>
<td>15.092</td>
</tr>
</tbody>
</table>
**Aim.** Use large time-course experiment to characterize response of human T-cell line (Jurkat) to PMA and ionomycin treatment.

**T-cell time-series dataset**
- Temporal expression of 58 genes for 10 unequally spaced time points.
- At each time point there are 44 separate measurements.
- See Rangel et al. (2004) for more details.
Application to T-cell data

\[ S_0 \prec F_1, N_0 \prec F_\Gamma, S_1 \prec F_T, N_1 \prec F_\Gamma \]

\[ \Theta = \begin{bmatrix}
S_0^1 & N_0^1 & S_1^1 & N_1^1 & 0 & 0 & \cdots & \cdots \\
S_0^1 & N_0^1 & S_1^1 & N_1^1 & 0 & 0 & \cdots & \cdots \\
S_0^2 & N_0^2 & S_1^2 & N_1^2 & 0 & 0 & \cdots & \cdots \\
S_0^2 & N_0^2 & S_1^2 & N_1^2 & 0 & 0 & \cdots & \cdots \\
S_0^3 & N_0^3 & S_1^3 & N_1^3 & 0 & 0 & \cdots & \cdots \\
S_0^3 & N_0^3 & S_1^3 & N_1^3 & 0 & 0 & \cdots & \cdots \\
\end{bmatrix} \]

\[ N_0^1 = N_0^2 = \ldots = N_0^{10} \]

\[ N_1^1 = N_1^2 = \ldots = N_1^{10} \]
What have we achieved so far? And problems!

Summary:
- Penalized Gaussian graphical models
- Coloured graphs
- Natural partitions
- Factorially coloured Gaussian graphical models

Problems:
1. Factorial colouring not particularly flexible in modeling time dynamics.
2. Gaussian assumption may be too restrictive for realistic genomic data.
1. Extension: Slowly changing graphical models

- **Problem:** Estimate changes in the dynamic of the network.

**Main Idea:** Penalize changes between graphs at different time points

\[
\|\Delta \Theta\|_1 = \sum_{s=0}^{t-1} \sum_{k=0}^{t-1} \|N_s^k - N_s^{k+1}\|_1.
\]

- **Solution:** Penalized maximum likelihood subject to constraints.
Application to a time-course dataset

\[ S_0 \prec F_{\Gamma_T}, N_0 \prec 1, N_1 \prec 1, N_2 \prec 0 \]

\[ \Theta = \begin{bmatrix} S^1_0 & N^1_0 & S^1_0 & N^1_0 & 0 & 0 & \ldots & \ldots \\ S^1_1 & N^1_1 & S^1_1 & N^1_1 & 0 & 0 & \ldots & \ldots \\ S^2_0 & N^2_0 & S^2_0 & N^2_0 & 0 & 0 \\ S^2_1 & N^2_1 & S^2_1 & N^2_1 & 0 & 0 \\ S^3_0 & N^3_0 & S^3_0 & N^3_0 & S^3_0 & N^3_1 \\ \end{bmatrix} \]

\[
\begin{align*}
|N^1_0| - |N^2_0| & \quad 1 \\
|N^2_0| - |N^3_0| & \quad 7
\end{align*}
\]

\[ N_0 \text{ at time } 1 \]
2. Extension: Non-Normality

- Problem: Non-Normality of the data (e.g. T-cell).

- Solution: Copula Gaussian graphical models
Copula Gaussian graphical models

IDEA:

- Graph exists on a hidden Gaussian variable $Z \sim N(0, \Theta)$,
- $Z$ gives rise to observed non-Gaussian data $Y$.

We consider the $F_i$ as nuisance parameters.

For continuous variables: 1-to-1 relationship between $Z$ and $Y$.
For discrete variables, relationship is more complicated!

Latent Variable: $Z \sim N(\mu, \Theta^{-1})$  Marginal Distribution: $F$  Observed: $Y$
Application of Gaussian copula graphical models to T-cell

\[ S_0 \prec F_1, \mathbf{N}_0 \prec F_T, S_1 \prec F_T, \mathbf{N}_1 \prec F_T \]

\[ N^1_0 = N^2_0 = \ldots = N^{10}_0 \]

\[ N^1_1 = N^2_1 = \ldots \]
Mammary gland gene expression data:
- Microarray experiment
- using mammary tissue from female mice
- across 4 different developmental stages
- for 8,600 genes.
- 3 replicates on each of 18 time points.

30 genes have been identified as activators for developmental stages (Wit and McClure, 2004).

**Objective:**
Study interactions between these crucial mice genes.
Figure: Undirected $N_0$ (left) and directed $N_1$ (right) time series chain graphical model networks inferred from mammary gland time course gene expression data.

Graphical models are a convenient formulation of many genomic networks.

Static boolean networks:
- Sparse GLM inference via dglars.
- Software: R package dglars.

Dynamic continuous networks:
- Chain graphical models infer sparse time dynamics.
- Software: R package SparseTSCGM.