Detecting change-points in the structure of a network: Exact Bayesian inference

S. Robin

Joint work with A. Cleynen, E. Lebarbier, G. Rigaill,
L. Schwaller, M. Stumpf

Statistical Analysis of Networks, INI, Dec. 2016, Cambridge

Example: Gene regulatory network along time

Data: [AFI+02]

$$Y_{jt} = \text{expression of } j \text{ at time } t$$

Example: Gene regulatory network along time

Data: [AFI+02]

$$Y_{jt} = \text{expression of } j \text{ at time } t$$

'Model':

$$\begin{aligned} G_t &= \text{gene regulatory network at time } t \\ &= \text{graphical model of } (Y_{jt})_j \end{aligned}$$

Example: Gene regulatory network along time

Data: [AFI+02]

$$Y_{jt} = \text{expression of } j \text{ at time } t$$

'Model':

$$\begin{aligned} G_t &= \text{gene regulatory network at time } t \\ &= \text{graphical model of } (Y_{jt})_j \end{aligned}$$

Questions:

- Is G_t constant along time or is there some 'gene rewiring'?
- If not, when does it change?
- And what is the network within each period?

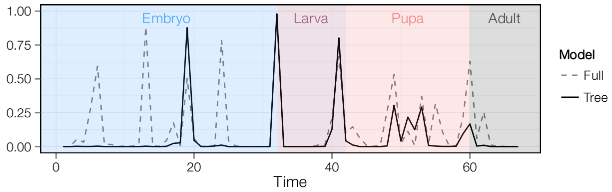
Example of output

Data: $N = 67$ time points, $p = 11$ genes, four expected regions

Example of output

Data: $N = 67$ time points, $p = 11$ genes, four expected regions

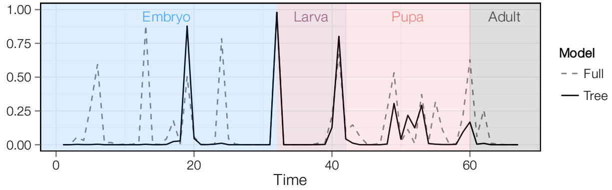
Posterior probability of change-points:



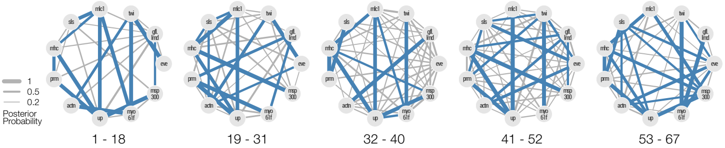
Example of output

Data: $N = 67$ time points, $p = 11$ genes, four expected regions

Posterior probability of change-points:



Inferred networks:



Similar problems

Ecology:

$$Y_{it} = \text{abundance of species } i \text{ at time } t \text{ in a given medium}$$

G_t = interaction structure between species at time t.

- Time-evolving species interaction network?

Similar problems

Ecology:

$$Y_{it} = \text{abundance of species } i \text{ at time } t \text{ in a given medium}$$

G_t = interaction structure between species at time t.

- Time-evolving species interaction network?

Neuroscience:

$$Y_{it} = \text{activity of brain region } i \text{ at time } t$$

G_t = connectivity structure between regions at time t.

- Time-evolving connectivity network?

Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion

Bayesian inference with discrete parameters

Mixed parameter: (θ, T)

$$
\begin{aligned}
\theta &= (\text{means, variances, correlations}) && \in \Theta = \text{continuous set,} \\
T &= (\text{segmentation, graph}) && \in \mathcal{T} = \text{discrete (finite) set,}
\end{aligned}
$$

$$
\Rightarrow \qquad p(Y) = \sum_{T \in \mathcal{T}} \int_\Theta p(Y, \theta, T) \, \mathrm{d}\theta = \sum_{T \in \mathcal{T}} p(Y, T)
$$

Bayesian inference with discrete parameters

Mixed parameter: (θ, T)

$$\begin{array}{lll} \theta = (\text{means, variances, correlations}) & & \in \Theta = \text{continuous set}, \\ T = (\text{segmentation, graph}) & & \in \mathcal{T} = \text{discrete (finite) set}, \end{array}$$

$$\Rightarrow \qquad p(Y) = \sum_{T \in \mathcal{T}} \int_\Theta p(Y, \theta, T) \, \mathrm{d}\theta = \sum_{T \in \mathcal{T}} p(Y, T)$$

Suppose that the integration wrt θ raise no issue, the summation

$$\sum_{T \in \mathcal{T}}$$

can often not be achieved in a naive way because of the combinatorial complexity.

$\rightarrow$ Need to find algorithmic or algebraic shortcuts

Main issue

Size of $\mathcal{T}$.

- No big deal if $\mathcal{T}$ is small (e.g. model selection within a small collection).

- Big issue if $\#\mathcal{T}$ grows (super-)exponentially with the number of observations n or the number of variables p.

Main issue

Size of $\mathcal{T}$.

- No big deal if $\mathcal{T}$ is small (e.g. model selection within a small collection).
- Big issue if $\#\mathcal{T}$ grows (super-)exponentially with the number of observations n or the number of variables p.

Examples.

- Change-point detection:
- 'Network inference' = inference of the structure of a graphical model
- Combination of both

Outline

Bayesian inference with discrete parameters

Change-point detection

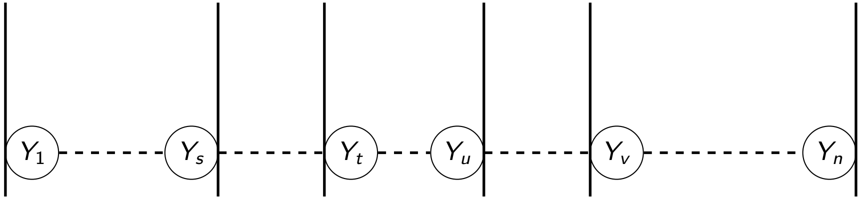
Network inference

Detecting changes in a graphical model

Discussion

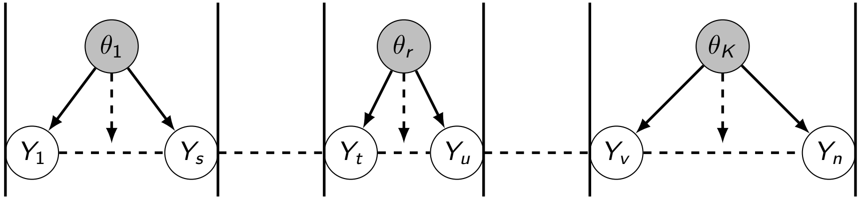
A change-point detection model

Segmentation T = set of adjacent segments. $\mathcal{T}^K = \mathcal{T}_{1:n}^K$ set of all possible segmentations.



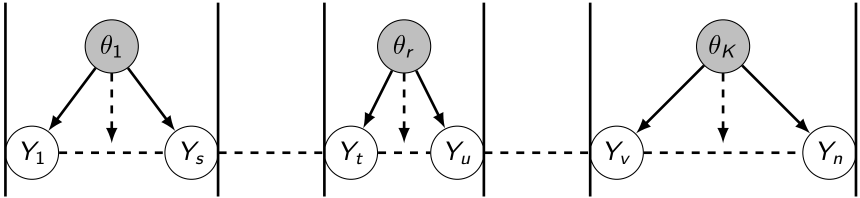
A change-point detection model

Segmentation T = set of adjacent segments. $\mathcal{T}^K = \mathcal{T}^K_{1:n}$ set of all possible segmentations.



A change-point detection model

Segmentation T = set of adjacent segments. $\mathcal{T}^K = \mathcal{T}^K_{1:n}$ set of all possible segmentations.



$$\begin{aligned} p(Y \mid T) &= \prod_{r \in T} \underbrace{\int p(\theta_r) \prod_{t \in r} p(Y_t \mid \theta_r) \, \mathrm{d}\theta_r} \\ &= \prod_{r \in T} \qquad p(Y^r), \qquad\qquad Y^r = (Y_t)_{t \in r} \end{aligned}$$

Bayesian inference

Factorability assumptions

- Independent parameters in each segment:

$$p(\theta \mid T) = \prod_{r \in T} p(\theta_r)$$

- Data are independent from one segment to another

$$p(Y \mid T, \theta) = \prod_{r \in T} p(Y^r \mid \theta_r)$$

- Prior distribution for the segmentation:

$$p(T \mid K) = \prod_{r \in T} a_r, \qquad \text{e.g. } a_r = n_r^\alpha$$

Some quantities of interest

Marginal likelihood.

$$p(Y \mid K) = \sum_{T \in \mathcal{T}^K} p(T \mid K) p(Y \mid T) \propto \sum_{T \in \mathcal{T}^K} \prod_{r \in T} a_r p(Y^r)$$

where the normalizing constant is

$$\sum_{T \in \mathcal{T}^K} \prod_{r \in T} a_r, \qquad \text{where} \quad \#\mathcal{T}_{1:n}^K = \binom{n-1}{K-1}.$$

Some quantities of interest

Marginal likelihood.

$$p(Y \mid K) = \sum_{T \in \mathcal{T}^K} p(T \mid K) p(Y \mid T) \propto \sum_{T \in \mathcal{T}^K} \prod_{r \in T} a_r p(Y^r)$$

where the normalizing constant is

$$\sum_{T \in \mathcal{T}^K} \prod_{r \in T} a_r, \qquad \text{where} \quad \#\mathcal{T}^K_{1:n} = \binom{n-1}{K-1}.$$

Posterior distribution of a change-point.

$$\Pr\{\tau_k = t \mid Y, K\} \propto \left(\sum_{T \in \mathcal{T}^k_{1:t}} \prod_{r \in T} a_r p(Y^r) \right) \left(\sum_{T \in \mathcal{T}^{K-k}_{t+1:n}} \prod_{r \in T} a_r p(Y^r) \right)$$

Some simple algebra

$A = (n+1) \times (n+1)$ upper-triangular matrix (with zero diagonal):

$$[A]_{i,j+1} = f_{i,j}, \qquad 1 \leq i \leq j \leq n$$

Some simple algebra

$A = (n+1) \times (n+1)$ upper-triangular matrix (with zero diagonal):

$$[A]_{i,j+1} = f_{i,j}, \qquad 1 \leq i \leq j \leq n$$

$$\begin{aligned} [A^2]_{1,n+1} &= f_{1,1} f_{2,n} + \dots + f_{1,i} f_{i+1,n} + \dots + f_{1,n-1} f_{n,n} \\ &= \sum_{1 \leq i < n} f_{1,i} \; f_{i+1,n} \; = \; \sum_{T \in \mathcal{T}^2} \prod_{r \in T} f_r \end{aligned}$$

Some simple algebra

$A = (n+1) \times (n+1)$ upper-triangular matrix (with zero diagonal):

$$[A]_{i,j+1} = f_{i,j}, \qquad 1 \leq i \leq j \leq n$$

$$\begin{aligned}
[A^2]_{1,n+1} &= f_{1,1} f_{2,n} + \dots + f_{1,i} f_{i+1,n} + \dots + f_{1,n-1} f_{n,n} \\
&= \sum_{1 \leq i < n} f_{1,i} \; f_{i+1,n} \; = \sum_{T \in \mathcal{T}^2} \prod_{r \in T} f_r
\end{aligned}$$

$$\begin{aligned}
[A^3]_{1,n+1} &= f_{1,1} f_{2,2} f_{3,n} + \dots + f_{1,i} f_{i+1,j} f_{j+1,n} + \dots + f_{1,n-2} f_{n-1,n-1} f_{n,n} \\
&= \sum_{1 \leq i \leq j < n} f_{1,i} \; f_{i+1,j} \; f_{j+1,n} \; = \sum_{T \in \mathcal{T}^3} \prod_{r \in T} f_r
\end{aligned}$$

Summing over segmentations [RLR11]

Property: *Define the upper triangular* $(n+1) \times (n+1)$ *matrix* A*:*

$$A_{i,j+1} = f_r \qquad \textit{for } r = [\![i, j]\!]$$

Summing over segmentations [RLR11]

Property: *Define the upper triangular* $(n+1) \times (n+1)$ *matrix* A*:*

$$A_{i,j+1} = f_r \qquad \textit{for } r = [\![i,j]\!]$$

Then

$$\left[A^K\right]_{1,n+1} = \sum_{T \in \mathcal{T}_{1:n}^K} \prod_{r \in T} f_r$$

$\rightarrow$ *all terms are computed in* $O(Kn^2)$*.*

Summing over segmentations [RLR11]

Property: *Define the upper triangular* $(n+1) \times (n+1)$ *matrix* A*:*

$$A_{i,j+1} = f_r \qquad \text{for } r = [\![i, j]\!]$$

Then

$$\left[A^K\right]_{1,n+1} = \sum_{T \in \mathcal{T}^K_{1:n}} \prod_{r \in T} f_r$$

$\rightarrow$ *all terms are computed in* $O(Kn^2)$*.*

- To compute $p(Y \mid K)$, take $f_r = a_r p(Y^r)$.

Summing over segmentations [RLR11]

Property: *Define the upper triangular $(n+1) \times (n+1)$ matrix A:*

$$A_{i,j+1} = f_r \qquad \textit{for } r = [\![i,j]\!]$$

Then

$$\left[A^K\right]_{1,n+1} = \sum_{T \in \mathcal{T}_{1:n}^K} \prod_{r \in T} f_r$$

$\rightarrow$ *all terms are computed in $O(Kn^2)$.*

- To compute $p(Y \mid K)$, take $f_r = a_r p(Y^r)$.
- Similar ideas in [Fea06].

Summing over segmentations [RLR11]

Property: *Define the upper triangular $(n+1) \times (n+1)$ matrix A:*

$$A_{i,j+1} = f_r \qquad \text{for } r = [\![i, j]\!]$$

Then

$$\left[A^K\right]_{1,n+1} = \sum_{T \in \mathcal{T}_{1:n}^K} \prod_{r \in T} f_r$$

$\rightarrow$ *all terms are computed in $O(Kn^2)$.*

- To compute $p(Y \mid K)$, take $f_r = a_r p(Y^r)$.
- Similar ideas in [Fea06].

$\rightarrow$ R package EBS (exact Bayesian segmentation) [CR14]

Gene regulatory network

Data: $N = 67$ time points, $p = 11$ genes, four expected regions [AFI+02]

Model:

$$t \in r : \qquad Y_t \,|\, \theta_r = (\mu_r, \Sigma_r) \sim \mathcal{N}(\mu_r, \Sigma_r)$$

$\rightarrow$ Saturated graphical model.

Gene regulatory network

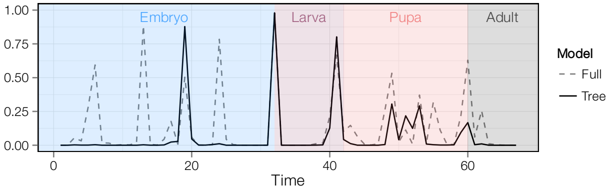
Data: $N = 67$ time points, $p = 11$ genes, four expected regions [AFI+02]

Model:

$$t \in r : \qquad Y_t \mid \theta_r = (\mu_r, \Sigma_r) \sim \mathcal{N}(\mu_r, \Sigma_r)$$

$\rightarrow$ Saturated graphical model.

Posterior probability of change-points: dotted line



Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion

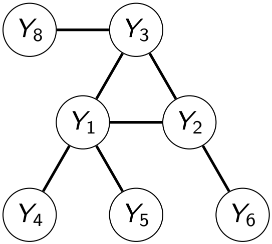
Graphical model framework

Property [Hammersley-Clifford]. $p(Y) = p(Y_1, \dots Y_p)$ is Markov wrt the (decomposable) graph G iff it factorizes wrt the maximal cliques of G:

$$p(Y) \propto \prod_{C \in \mathcal{C}(G)} \psi_c(Y^c), \qquad Y^c = (Y_j)_{j \in C}.$$

$\rightarrow$ G reveals the structure of conditional independences between the variables $Y_1, \dots Y_p$.

Graphical model



Means that

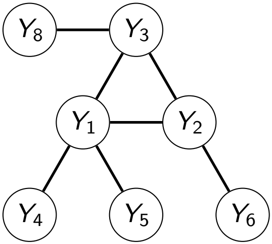
$$\begin{aligned} p(Y_1, \dots, Y_8) \quad \propto \quad & \psi_1(Y_1, Y_2, Y_3) \\ & \psi_2(Y_1, Y_4)\ \psi_3(Y_1, Y_5) \\ & \psi_4(Y_2, Y_6)\ \psi_5(Y_3, Y_8) \end{aligned}$$

which implies[1] that

$$\begin{aligned} Y_4 \perp Y_3 \quad & | \quad Y_1 \\ (Y_6, Y_7) \perp Y_3 \quad & | \quad Y_2 \\ \dots \end{aligned}$$

[1]Under fairly general assumptions on p

Graphical model



Means that

$$\begin{aligned} p(Y_1, \dots, Y_8) \quad \propto \quad & \psi_1(Y_1, Y_2, Y_3) \\ & \psi_2(Y_1, Y_4)\, \psi_3(Y_1, Y_5) \\ & \psi_4(Y_2, Y_6)\, \psi_5(Y_3, Y_8) \end{aligned}$$

which implies[1] that

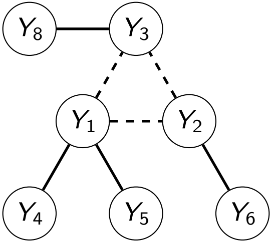
$$\begin{aligned} Y_4 \perp Y_3 \quad & | \quad Y_1 \\ (Y_6, Y_7) \perp Y_3 \quad & | \quad Y_2 \\ \dots & \end{aligned}$$

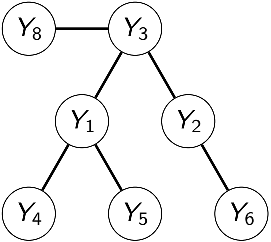
'Network inference' problem: Based on $\{(Y_{i1}, \dots Y_{ip})\}_i$ iid $\sim p$, infer G.

[1] Under fairly general assumptions on p

Tree-structures network

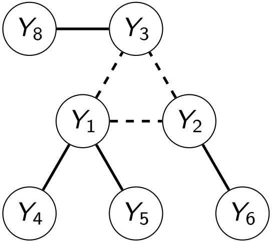
Tree assumption: the graph G is a spanning tree T.

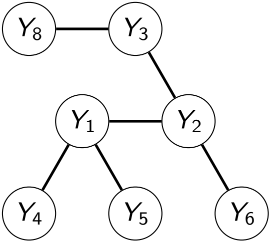




Tree-structures network

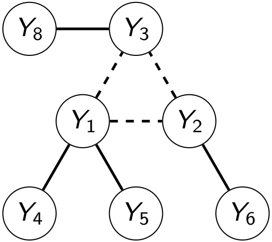
Tree assumption: the graph G is a spanning tree T.

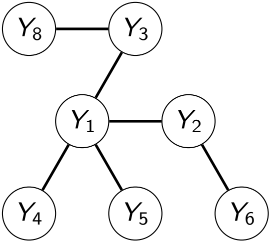




Tree-structures network

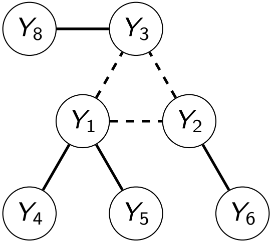
Tree assumption: the graph G is a spanning tree T.

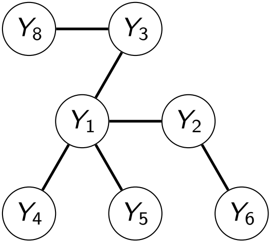




Tree-structures network

Tree assumption: the graph G is a spanning tree T.

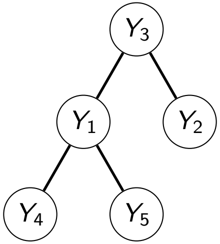




- Consistent with the usual assumption that the graph is sparse (although much stronger).
- Not true in general, but may be sufficient for the inference on local structures, such as the existence of a given edge.

Hyper-Markov prior

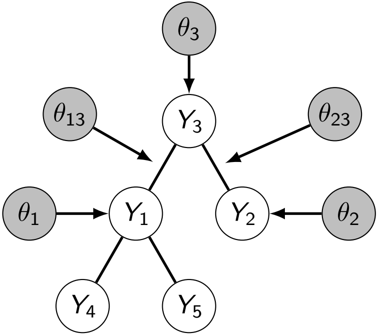
Graphical model:
$p(Y \mid \theta, T)$ factorizes wrt edges of T
(Markov wrt T)



Hyper-Markov prior

Graphical model:
$p(Y \mid \theta, T)$ factorizes wrt edges of T
(Markov wrt T)

Desirable prior:
$p(\theta \mid T)$ factorizes wrt edges of T as well
(Hyper-Markov wrt T)

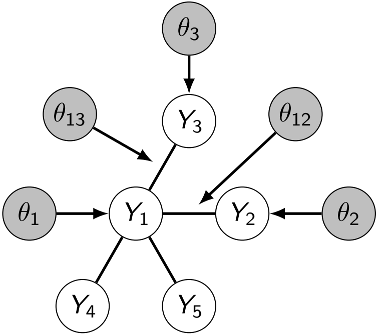


Hyper-Markov prior

Graphical model:
$p(Y \mid \theta, T)$ factorizes wrt edges of T
(Markov wrt T)

Desirable prior:
$p(\theta \mid T)$ factorizes wrt edges of T as well
(Hyper-Markov wrt T)

Averaging over T:
This should hold fot any tree T



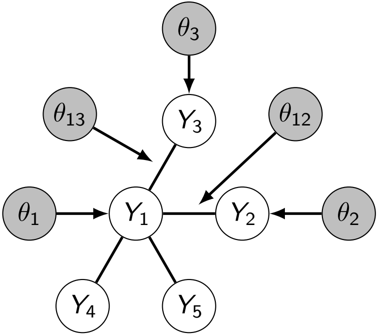
(θ_4, θ_5, θ_{14}, θ_{15} not drawn.)

Hyper-Markov prior

Graphical model:
$p(Y \mid \theta, T)$ factorizes wrt edges of T
(Markov wrt T)

Desirable prior:
$p(\theta \mid T)$ factorizes wrt edges of T as well
(Hyper-Markov wrt T)

Averaging over T:
This should hold fot any tree T



(θ_4, θ_5, θ_{14}, θ_{15} not drawn.)

Compatible family of strong Markov hyper-dist. [DL93]:
$\longrightarrow$ multinomial-Dirichlet (conjugacy),
$\longrightarrow$ normal-Wishart (conjugacy),
$\longrightarrow$ Gaussian copulas (numerical integration), ...?

Bayesian inference for tree-structured network [SRS15]

$p(Y \mid T)$ Markov wrt T

$$
\begin{aligned}
p(Y \mid T) &= \prod_j p(Y_j) \prod_{(j,k) \in T} \frac{p(Y_j, Y_k)}{p(Y_j)p(Y_k)} \\
&= \prod_{(j,k) \in T} p(Y_j, Y_k) \Bigg/ \prod_j p^{d_j - 1}(Y_j)
\end{aligned}
$$

where d_j is the degree (number of neighbors in T) of node j.

Bayesian inference for tree-structured network [SRS15]

$p(Y \mid T)$ Markov wrt T

$$p(Y \mid T) = \prod_j p(Y_j) \prod_{(j,k)\in T} \frac{p(Y_j, Y_k)}{p(Y_j)p(Y_k)}$$
$$= \prod_{(j,k)\in T} p(Y_j, Y_k) \bigg/ \prod_j p^{d_j-1}(Y_j)$$

where d_j is the degree (number of neighbors in T) of node j.

Prior on T: factorizes over the edges:

$$p(T) \propto \prod_{(j,k)\in T} a_{jk}$$

Quantities of interest

Marginal distribution.

$$p(Y) = \sum_{T \in \mathcal{T}} p(T) p(Y \mid T) \propto \sum_{T \in \mathcal{T}} \prod_{j,k} \frac{a_{jk} p(Y_j, Y_k)}{p(Y_j) p(Y_k)}$$

where $\mathcal{T}$ stands for the set of all spanning trees.

Quantities of interest

Marginal distribution.

$$p(Y) = \sum_{T \in \mathcal{T}} p(T) p(Y \mid T) \propto \sum_{T \in \mathcal{T}} \prod_{j,k} \frac{a_{jk} p(Y_j, Y_k)}{p(Y_j) p(Y_k)}$$

where $\mathcal{T}$ stands for the set of all spanning trees.

Posterior probability for an edge to be absent.

$$\Pr\{(j,k) \notin T \mid Y\} \propto \sum_{T \in \mathcal{T} : (j,k) \notin T} \prod_{j,k} \frac{a_{jk} p(Y_j, Y_k)}{p(Y_j) p(Y_k)}$$

Quantities of interest

Marginal distribution.

$$p(Y) = \sum_{T \in \mathcal{T}} p(T) p(Y \mid T) \propto \sum_{T \in \mathcal{T}} \prod_{j,k} \frac{a_{jk} p(Y_j, Y_k)}{p(Y_j) p(Y_k)}$$

where $\mathcal{T}$ stands for the set of all spanning trees.

Posterior probability for an edge to be absent.

$$\Pr\{(j,k) \notin T \mid Y\} \propto \sum_{T \in \mathcal{T} : (j,k) \notin T} \prod_{j,k} \frac{a_{jk} p(Y_j, Y_k)}{p(Y_j) p(Y_k)}$$

Typical form:

$$\sum_{T \in \mathcal{T}} \prod_{(j,k) \in T} f_{jk}, \qquad \text{with} \quad \#\mathcal{T} = p^{p-2}.$$

Summing over spanning trees

Matrix-tree theorem. [Cha82]

- $F = [f_{jk}]$: *a symmetric matrix with* $f(j,j) = 0, f_{jk} > 0$;
- $\Delta = [\Delta_{jk}]$ *its Laplacian:* $\Delta_{jj} = \sum_k f_{jk}, \Delta_{jk} = -f_{jk}$.

Summing over spanning trees

Matrix-tree theorem. [Cha82]

- $F = [f_{jk}]$: *a symmetric matrix with* $f(j,j) = 0, f_{jk} > 0$;
- $\Delta = [\Delta_{jk}]$ *its Laplacian:* $\Delta_{jj} = \sum_k f_{jk}, \Delta_{jk} = -f_{jk}$.

Then the minors $|\Delta^{uv}|$ *of* Δ *are equal and*

$$|\Delta^{uv}| = \sum_{T \in \mathcal{T}} \prod_{(j,k) \in T} f_{jk}.$$

Summing over spanning trees

Matrix-tree theorem. [Cha82]

- $F = [f_{jk}]$: *a symmetric matrix with* $f(j,j) = 0, f_{jk} > 0$;
- $\Delta = [\Delta_{jk}]$ *its Laplacian:* $\Delta_{jj} = \sum_k f_{jk}, \Delta_{jk} = -f_{jk}$.

Then the minors $|\Delta^{uv}|$ *of* Δ *are equal and*

$$|\Delta^{uv}| = \sum_{T \in \mathcal{T}} \prod_{(j,k) \in T} f_{jk}.$$

- $p(Y)$ and the normalizing constant of $p(T)$, ... can be computed at the cost of computing a $p \times p$ determinant, ie $O(p^3)$.
- Already used in [MJ06,Kir07] for tree learning.

Posterior probability of an edge

The existence of an edge between variables Y_j and Y_k can be assessed by

$$\Pr\{(j,k) \in T \mid Y\} \propto \sum_{T \ni (j,k)} p(T) p(Y \mid T)$$

which depends on the prior $p(T)$.

The prior probability $\Pr\{(j,k) \in T\}$ can be tuned

- with the prior coefficient a_{jk}
- or set to an arbitrary value using an edge-specific probability change.

Posterior probability of an edge

The existence of an edge between variables Y_j and Y_k can be assessed by

$$\Pr\{(j,k) \in T \mid Y\} \propto \sum_{T \ni (j,k)} p(T) p(Y \mid T)$$

which depends on the prior $p(T)$.

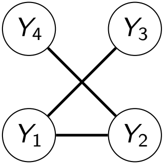
The prior probability $\Pr\{(j,k) \in T\}$ can be tuned

- with the prior coefficient a_{jk}
- or set to an arbitrary value using an edge-specific probability change.

All posterior probabilities can still be computed in $O(p^3)$ [Kir07].

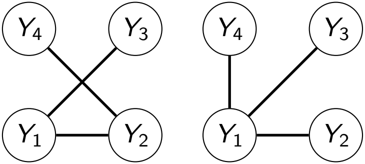
$\rightarrow$ R package Saturnin (spanning trees used for network inference) [SRS15]

Tree averaging



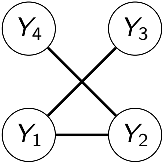
$P\{T = T_1|Y\}$

Tree averaging

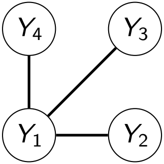


$P\{T = T_1 | Y\}$ $\quad$ $P\{T = T_2 | Y\}$

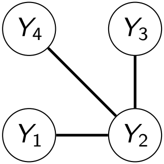
Tree averaging



$P\{T = T_1 | Y\}$

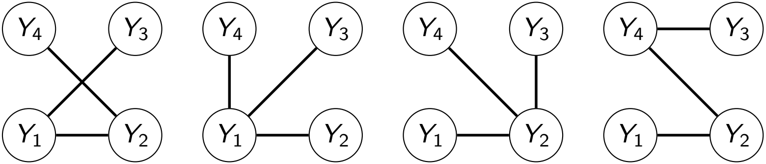


$P\{T = T_2 | Y\}$



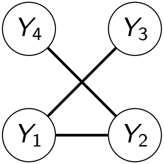
$P\{T = T_3 | Y\}$

Tree averaging

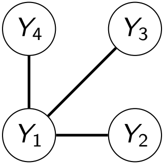


$P\{T = T_1 | Y\}$ $\quad$ $P\{T = T_2 | Y\}$ $\quad$ $P\{T = T_3 | Y\}$ $\quad$ $P\{T = T_4 | Y\}$

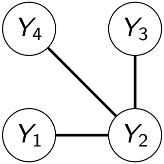
Tree averaging



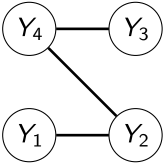
$P\{T = T_1|Y\}$



$P\{T = T_2|Y\}$

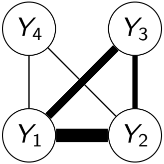


$P\{T = T_3|Y\}$



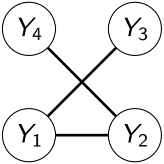
$P\{T = T_4|Y\}$

Edge posterior probabilities:

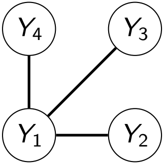


$P\{(j,k) \in T|Y\}$

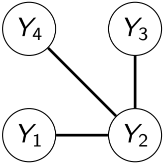
Tree averaging



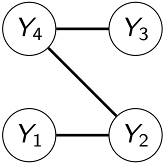
$P\{T = T_1 | Y\}$



$P\{T = T_2 | Y\}$

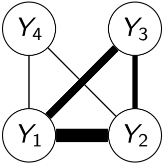


$P\{T = T_3 | Y\}$



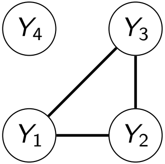
$P\{T = T_4 | Y\}$

Edge posterior probabilities:



$P\{(j,k) \in T | Y\}$

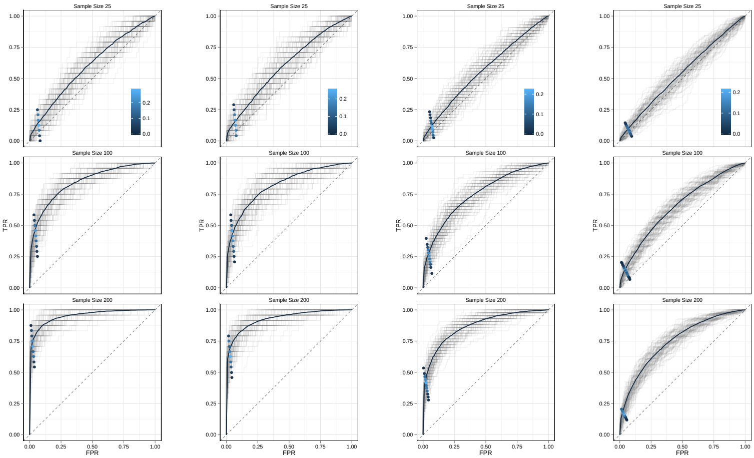
Thresholding probabilities:



$P\{(j,k) \in T | Y\}$

Simulations: ROC curves for edge detection

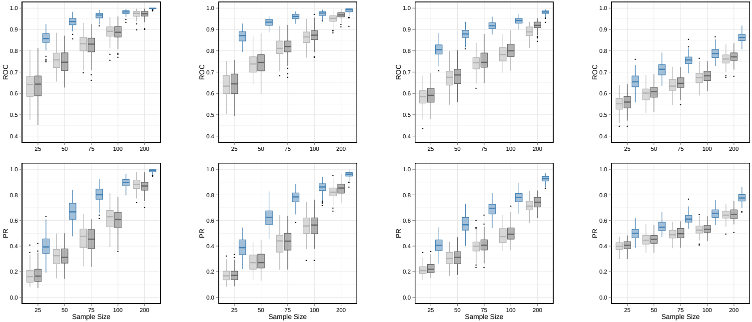
For various graph topologies ($p = 25$, $n = 25, 50, 200$, $B = 100$ simulations)



Tree | Erdös-Rényi $p_c = 2/p$ | Erdös-Rényi $p_c = 4/p$ | Erdös-Rényi $p_c = 8/p$

Simulations: Comparison with sampling among DAGs

[NPK11]: MCMC sampling over the directed acyclic graphs (multinomial case)

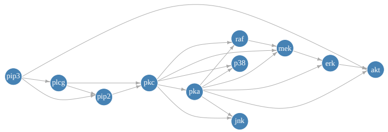


Tree	Erdös-Rényi $p_c = 2/p$	Erdös-Rényi $p_c = 4/p$	Erdös-Rényi $p_c = 8/p$

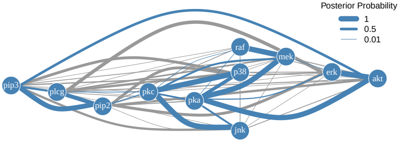
Area under the curves: top=ROC, bottom=PR
light grey = multinomial trees (2.2"), dark grey: multinomial DAGs (1393")

Illustration: Raf pathway

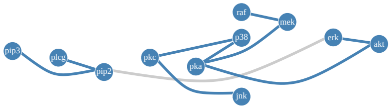
Flow cytometry data for $p = 11$ proteins from the Raf signaling pathway [SPP+05]



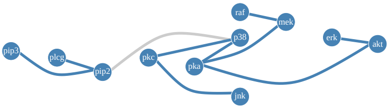
'ground truth'



posterior probabilities



most likely tree



second most likely tree

Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

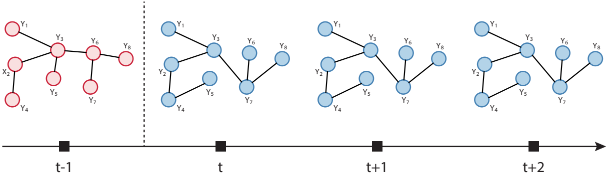
Discussion

Change-point in a graphical model

Change-point in a graphical model

Problem: [SR16]

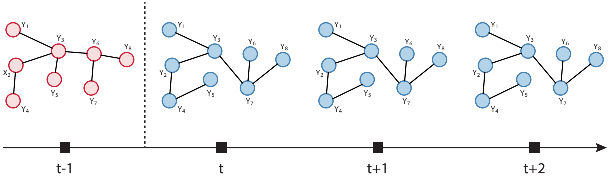
- Consider p variables observed along time;
- Consider the graph G_t supporting the graphical model at time t;
- Does the graph G_t remain the same along time?



Change-point in a graphical model

Problem: [SR16]

- Consider p variables observed along time;
- Consider the graph G_t supporting the graphical model at time t;
- Does the graph G_t remain the same along time?



Examples:

1. Gene regulatory network along the *Drosophila* life cycle?

2. Connections between brain regions along different tasks?

Model

$$K = \text{number of segments} \qquad p(K);$$

Model

$$
\begin{array}{rcll}
K & = & \text{number of segments} & p(K); \\
R = (r_k)_k & = & \text{segmentation} & p(R|K) \propto \prod_{r \in R} a_r;
\end{array}
$$

Model

$$
\begin{array}{rcll}
K & = & \text{number of segments} & p(K); \\[1em]
R = (r_k)_k & = & \text{segmentation} & p(R|K) \propto \prod_{r \in R} a_r; \\[1em]
T = (T_r)_r & = & \text{set of trees} & p(T|R) = \prod_{r \in R} p(T_r|R), \\
T_r & = & \text{graphical model in segment } r &
\end{array}
$$

Model

$$
\begin{array}{rcll}
K & = & \text{number of segments} & p(K); \\[2ex]
R = (r_k)_k & = & \text{segmentation} & p(R|K) \propto \prod_{r \in R} a_r; \\[2ex]
T = (T_r)_r & = & \text{set of trees} & p(T|R) = \prod_{r \in R} p(T_r|R), \\
T_r & = & \text{graphical model in segment } r & \\[2ex]
\theta = (\theta_r)_r & = & \text{parameter in each segment} & p(\theta|T) = \prod_{r \in R} p(\theta_r|T_r) \\
 & & \text{typically: support}(\Sigma_r^{-1}) = T_r &
\end{array}
$$

Model

$$
\begin{array}{rcll}
K & = & \text{number of segments} & p(K); \\
R = (r_k)_k & = & \text{segmentation} & p(R|K) \propto \prod_{r \in R} a_r; \\
T = (T_r)_r & = & \text{set of trees} & p(T|R) = \prod_{r \in R} p(T_r|R), \\
T_r & = & \text{graphical model in segment } r & \\
\theta = (\theta_r)_r & = & \text{parameter in each segment} & p(\theta|T) = \prod_{r \in R} p(\theta_r|T_r) \\
 & & \text{typically: support}(\Sigma_r^{-1}) = T_r & \\
Y_t = (Y_{jt}) & = & \text{data collected at time } t & p(Y|R, \theta) = \prod_{r \in R} \prod_{t \in r} p(Y_t|\theta_r)
\end{array}
$$

Model

$$
\begin{array}{rcll}
K & = & \text{number of segments} & p(K); \\
R = (r_k)_k & = & \text{segmentation} & p(R|K) \propto \prod_{r \in R} a_r; \\
T = (T_r)_r & = & \text{set of trees} & p(T|R) = \prod_{r \in R} p(T_r|R), \\
T_r & = & \text{graphical model in segment } r & \\
\theta = (\theta_r)_r & = & \text{parameter in each segment} & p(\theta|T) = \prod_{r \in R} p(\theta_r|T_r) \\
 & & \text{typically: support}(\Sigma_r^{-1}) = T_r & \\
Y_t = (Y_{jt}) & = & \text{data collected at time } t & p(Y|R,\theta) = \prod_{r \in R} \prod_{t \in r} p(Y_t|\theta_r)
\end{array}
$$

Typical issue: Evaluate

$$p(Y|K) = \sum_R \sum_{T=(T_r)_{r \in R}} p(R|K) \prod_{r \in R} p(T_r|R) p(Y^r|T_r)$$

Handling two sums

Double discrete structure:

- $\approx (N/K)^K$ possible segmentations into K segments;
- $p^{K(p-2)}$ possible combination of K trees

$\rightarrow$ sum over $\approx (N/K)^K p^{K(p-2)}$ terms.

Handling two sums

Double discrete structure:

- $\approx (N/K)^K$ possible segmentations into K segments;
- $p^{K(p-2)}$ possible combination of K trees

$\rightarrow$ sum over $\approx (N/K)^K p^{K(p-2)}$ terms.

Combining the two preceding tools:

- Summing of all segmentations in $O(KN^2)$,
- Summing over all trees in $O(p^3)$ (one tree per possible segment)

$\rightarrow$ Global complexity $= O(\max\{K, p^3\}N^2)$

Inference

Quantities of interest can be computed in $O(p^3 N^2)$:

- $P(\text{change-point at time } t \mid K, Y)$
- $P(\text{edge } (j,k) \text{ present at time } t \mid K, Y)$
- $P(\text{edge } (j,k) \text{ present at all } t \mid Y)$
- $P(K \text{ segments} \mid Y)$.

Inference

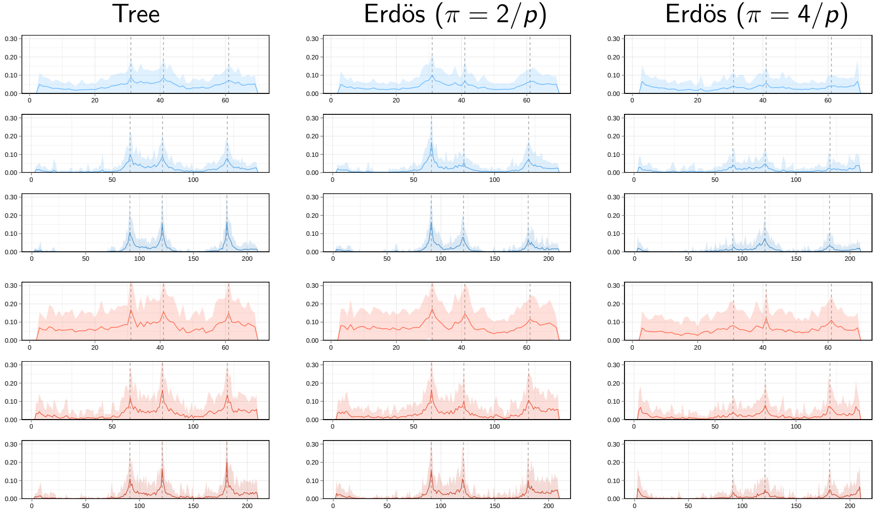
Quantities of interest can be computed in $O(p^3 N^2)$:

- $P(\text{change-point at time } t \mid K, Y)$
- $P(\text{edge } (j, k) \text{ present at time } t \mid K, Y)$
- $P(\text{edge } (j, k) \text{ present at all } t \mid Y)$
- $P(K \text{ segments} \mid Y)$.

$+$ Network comparison

- $P(T_1 = T_2 \mid Y_1, Y_2)$
- $P(\text{edge } (j, k) \text{ present in both } T_1 \text{ and } T_2 \mid Y_1, Y_2)$.

Some simulations



Top to bottom: N = 70, 140, 210. Tree-structured network. Complete network.

Gene regulatory network

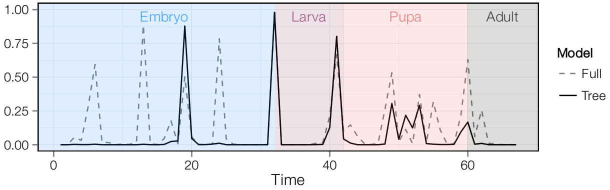
Gene regulatory network

Data: $N = 67$ time points, $p = 11$ genes, four expected regions

Gene regulatory network

Data: $N = 67$ time points, $p = 11$ genes, four expected regions

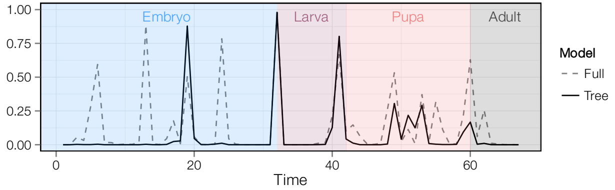
Posterior probability of change-points:



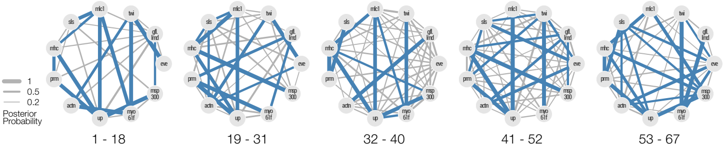
Gene regulatory network

Data: $N = 67$ time points, $p = 11$ genes, four expected regions

Posterior probability of change-points:



Inferred networks:



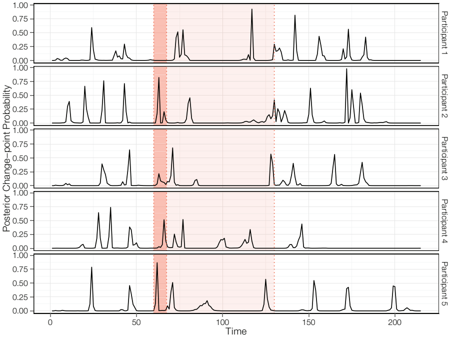
FMRI data [CHA+12]

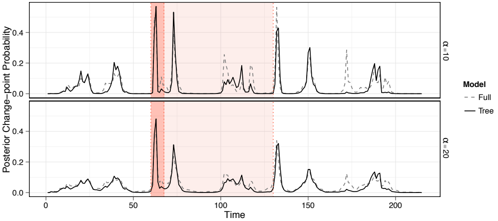
FMRI data collected on 20 patients:
$p = 5$ brain regions,
$n = 215$ time-points.

Task changes at $t = 60$ and 120.

Top: 5 patients analyzed separately.

Bottom: joint analysis of the same 5 patients





Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion

Summary

To summarize.

- Exact Bayesian inference can be achieved for some fairly complex models with discrete parameter.
- Do not have to care about sampling and convergence.
- No systematic way to check when similar algebraic shortcuts exist
  $\rightarrow$ ad-hoc developments.

Algebraic properties

	Bayesian inference	Maximum likelihood

Algebraic properties

	Bayesian inference	Maximum likelihood
Change-point detection	$\sum_m \prod_{r \in m} p_r$ $\rightarrow$ Matrix power	$\max_m \sum_{r \in m} \log p_r$ $\rightarrow$ Dynamic programing

Algebraic properties

	Bayesian inference	Maximum likelihood
Change-point detection	$\sum_m \prod_{r \in m} p_r$ $\longrightarrow$ Matrix power	$\max_m \sum_{r \in m} \log p_r$ $\longrightarrow$ Dynamic programing
Tree-structured network inference	$\sum_T \prod_{(jk) \in T} p_{jk}$ $\longrightarrow$ Matrix-tree theorem	$\max_T \sum_{(jk) \in T} \log p_{jk}$ $\longrightarrow$ Max. spanning tree

Algebraic properties

	Bayesian inference	Maximum likelihood
Change-point detection	$\sum_m \prod_{r \in m} p_r$ $\longrightarrow$ Matrix power	$\max_m \sum_{r \in m} \log p_r$ $\longrightarrow$ Dynamic programing
Tree-structured network inference	$\sum_T \prod_{(jk) \in T} p_{jk}$ $\longrightarrow$ Matrix-tree theorem	$\max_T \sum_{(jk) \in T} \log p_{jk}$ $\longrightarrow$ Max. spanning tree
Algebra	sum-product	max-sum

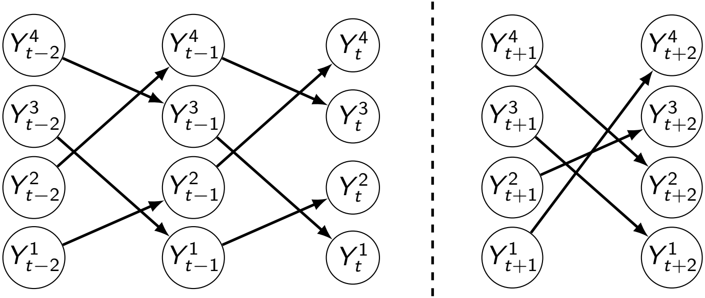
Algebraic properties

	Bayesian inference	Maximum likelihood
Change-point detection	$\sum_m \prod_{r \in m} p_r$ $\longrightarrow$ Matrix power	$\max_m \sum_{r \in m} \log p_r$ $\longrightarrow$ Dynamic programing
Tree-structured network inference	$\sum_T \prod_{(jk) \in T} p_{jk}$ $\longrightarrow$ Matrix-tree theorem	$\max_T \sum_{(jk) \in T} \log p_{jk}$ $\longrightarrow$ Max. spanning tree
Algebra	sum-product	max-sum

Any other example?

Future works

- Dealing with dependency along time.



- Influence of the prior: $p(T)$ depends on n and/or p.

- Solve numerical issues raised by the exact evaluation of all probabilities.

References I

M. N Arbeitman, E. EM Furlong, F. Imam, E. Johnson, B. H Null, B. S Baker, M. A Krasnow, M. P Scott, R. W Davis, and K. P White. Gene expression during the life cycle of drosophila melanogaster. *Science*, 297(5590):2270–2275, 2002.

S. Chaiken. A combinatorial proof of the all minors matrix tree theorem. *SIAM Journal on Algebraic Discrete Methods*, 3(3):319–329, 1982.

I. Cribben, R. Haraldsdottir, L. Y Atlas, T. D Wager, and M. A Lindquist. Dynamic connectivity regression: determining state-related changes in brain connectivity. *Neuroimage*, 61(4):907–920, 2012.

A. Cleynen and S. Robin. Comparing change-point location in independent series. *Statistics and Computing*, pages 1–14, 2014.

A. P. Dawid and S. L. Lauritzen. Hyper Markov Laws in the Statistical Analysis of Decomposable Graphical Models. *The Annals of Statistics*, 21(3):1272–1317, 1993.

P. Fearnhead. Exact and efficient bayesian inference for multiple changepoint problems. *Statistics and computing*, 16(2):203–213, 2006.

S. Kirshner. Learning with tree-averaged densities and distributions. In *NIPS*, pages 761–768, 2007.

M. Meilă and T. Jaakkola. *Tractable Bayesian learning of tree belief networks*. March 2006.

References II

T. Niinimaki, P. Parviainen, and M. Koivisto. Partial order MCMC for structure discovery in bayesian networks. In Fabio Gagliardi Cozman and Avi Pfeffer, editors, *UAI*, 2011.

G. Rigaill, E. Lebarbier, and S. Robin. Exact posterior distributions over the segmentation space and model selection for multiple change-point detection problem. *Stat. Comp.*, 22:917–29, 2011. DOI: 10.1007/s11222-011-9258-8.

K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. N. Causal protein-signaling networks derived from multiparameter single-cell data. *Science (New York, N.Y.)*, 308:523–529, 2005.

L. Schwaller and S. Robin. Exact bayesian inference for off-line change-point detection in tree-structured graphical models. *Statistics and Computing*, pages 1–15, 2016.

L. Schwaller, S. Robin, and M. Stumpf. Bayesian Inference of Graphical Model Structures Using Trees. Technical report, ArXiv:1504.02723, April 2015.