Exact Bayesian inference for some models with discrete parameters

S. Robin

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

INRA / AgroParisTech

FMJH, Sept 2016, IHES
A reminder on Bayesian inference

Parametric inference. A statistical model describes how the distribution of the observed data $Y$ depends on a parameter of interest:

$$Y \sim p(\cdot; \vartheta)$$
A reminder on Bayesian inference

Parametric inference. A statistical model describes how the distribution of the observed data $Y$ depends on a parameter of interest:

$$Y \sim p(\cdot; \vartheta)$$

'Classical' or 'frequentist' framework: $\vartheta$ supposed to be fix.

- Try to provide a estimate $\hat{\vartheta}$ not too far from the true $\vartheta^*$. 

A reminder on Bayesian inference

Parametric inference. A statistical model describes how the distribution of the observed data $Y$ depends on a parameter of interest:

$$Y \sim p(\cdot; \vartheta)$$

'Classical' or 'frequentist' framework: $\vartheta$ supposed to be fix.

- Try to provide an estimate $\hat{\vartheta}$ not too far from the true $\vartheta^*$. 

Bayesian framework: $\vartheta$ is random with prior distribution $p(\vartheta)$.

- Try to establish its conditional distribution given the data:

$$p(\vartheta|Y) = \frac{p(\vartheta)p(Y|\vartheta)}{p(Y)} = \text{posterior distribution.}$$
Illustration: Beta binomial (1/2)

Aim: Infer a success probability $\theta$ based on $n$ independent trials.
Illustration: Beta binomial (1/2)

Aim: Infer a success probability $\theta$ based on $n$ independent trials.

- Define a prior distribution $p(\theta)$, e.g.
  
  \[
  \theta \sim \text{B}(a, b)
  \]
Illustration: Beta binomial (1/2)

Aim: Infer a success probability $\theta$ based on $n$ independent trials.

- Define a prior distribution $p(\theta)$, e.g.
  $$\theta \sim B(a, b)$$
- Write the conditional distribution of $Y = \text{number of success}$:
  $$Y|\theta \sim B(n, \theta)$$
Illustration: Beta binomial (1/2)

Aim: Infer a success probability $\theta$ based on $n$ independent trials.

- Define a prior distribution $p(\theta)$, e.g.
  $$\theta \sim B(a, b)$$
- Write the conditional distribution of $Y = \text{number of success}$:
  $$Y|\theta \sim B(n, \theta)$$
- Deduce the conditional distribution of $\theta$ given $Y$ (posterior):
  $$p(\theta|Y = y) = \frac{p(\theta)P(Y = y|\theta)}{P(Y = y)}$$
  where $P(Y = y) = \int P(Y = y|\theta)p(\theta) \, d\theta$
Illustration: Beta binomial (1/2)

Aim: Infer a success probability $\theta$ based on $n$ independent trials.

- Define a prior distribution $p(\theta)$, e.g.
  $$\theta \sim B(a, b)$$

- Write the conditional distribution of $Y = \text{number of success}$:
  $$Y | \theta \sim B(n, \theta)$$

- Deduce the conditional distribution of $\theta$ given $Y$ (posterior):
  $$p(\theta | Y = y) = \frac{p(\theta)P(Y = y | \theta)}{P(Y = y)}$$
  where $P(Y = y) = \int P(Y = y | \theta)p(\theta) \, d\theta$

In this case we get
  $$(\theta | Y = y) \sim B(\tilde{a}, \tilde{b}) : \quad \tilde{a} = a + y, \quad \tilde{b} = b + n - y$$
Illustration: Beta binomial (2/2)
Illustration: Beta binomial (2/2)

1. Define a (flat) prior for $\theta$
Bayesian inference with discrete parameters

Illustration: Beta binomial (2/2)

1. Define a (flat) prior for $\theta$
2. Get the data ($n = 20$, $y = 4$)
Illustration: Beta binomial (2/2)

1. Define a (flat) prior for $\theta$
2. Get the data ($n = 20$, $y = 4$)
3. Compute the posterior distribution

A different prior gives in a different posterior.
Bayesian inference with discrete parameters

Illustration: Beta binomial (2/2)

1. Define a (flat) prior for $\theta$
2. Get the data ($n = 20$, $y = 4$)
3. Compute the posterior distribution
4. Compute, e.g. a credibility interval: $\text{Cl}_{95\%}(\theta | Y)$. 

A different prior gives a different posterior.
Illustration: Beta binomial (2/2)

1. Define a (flat) prior for $\theta$
2. Get the data ($n = 20$, $y = 4$)
3. Compute the posterior distribution
4. Compute, e.g. a credibility interval: $\text{Cl}_{95\%}(\theta|Y)$.

A different prior
Illustration: Beta binomial (2/2)

1. Define a (flat) prior for $\theta$
2. Get the data ($n = 20$, $y = 4$)
3. Compute the posterior distribution
4. Compute, e.g. a credibility interval: $\text{CI}_{95\%}(\theta | Y)$.

A different prior gives in a different posterior.
Bayesian inference

Generic Bayesian framework:

prior: \( p(\vartheta) \)

likelihood: \( p(Y|\vartheta) \)  (given by the model)

\[ \rightarrow \text{posterior: } p(\vartheta|Y) \]
Bayesian inference

Generic Bayesian framework:

\[
\begin{align*}
\text{prior:} & \quad p(\vartheta) \\
\text{likelihood:} & \quad p(Y|\vartheta) \quad \text{(given by the model)} \\
\rightarrow \text{posterior:} & \quad p(\vartheta|Y)
\end{align*}
\]

A typical issue: Deriving

\[p(\vartheta|Y) = p(\vartheta)p(Y|\vartheta) / p(Y)\]

is often not trivial, typically because

\[p(Y) = \int_{\Theta} p(Y|\vartheta)p(\vartheta) \, d\vartheta\]

is intractable when the parameter space \(\Theta\) is huge.
Posterior distribution

3 main approaches
Posterior distribution

3 main approaches

1. Sampling (Monte Carlo, Monte Carlo - Markov chain, sequential MC, Importance sampling, ...):

   \[ \text{sample } (\theta^b) \sim p(\theta | Y). \]
Posterior distribution

3 main approaches

1. Sampling (Monte Carlo, Monte Carlo - Markov chain, sequential MC, Importance sampling, ...):

\[
\text{sample } (\vartheta^b) \sim p(\vartheta | Y).
\]

2. Approximation (e.g. Variational Bayes, Expectation propagation, ...):

\[
\text{find } q_Y(\vartheta) \simeq p(\vartheta | Y).
\]
Posterior distribution

3 main approaches

1. Sampling (Monte Carlo, Monte Carlo - Markov chain, sequential MC, Importance sampling, ...):

\[
\text{sample } (\vartheta^b) \sim p(\vartheta | Y).
\]

2. Approximation (e.g. Variational Bayes, Expectation propagation, ...):

\[
\text{find } q_Y(\vartheta) \simeq p(\vartheta | Y).
\]

3. Exact:

\[
\text{actually compute } p(\vartheta | Y)
\]

or some marginal of interest.
Models with discrete parameters

Mixed parameter: \( \vartheta \rightarrow (\theta, T) \)

\[ \theta \in \Theta = \text{continuous set}, \quad T \in \mathcal{T} = \text{discrete (countable) set}, \]

\[ \Rightarrow \quad p(Y) = \sum_{T \in \mathcal{T}} \int_{\Theta} p(Y, \theta, T) \, d\theta \]
Models with discrete parameters

Mixed parameter: $\vartheta \rightarrow (\theta, T)$

$\theta \in \Theta = \text{continuous set}, \quad T \in \mathcal{T} = \text{discrete (countable) set},$

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

Size of $\mathcal{T}$.

- No big deal of $\mathcal{T}$ 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

Suppose that the calculation wrt $\theta$ raise no issue\(^1\), the calculation of

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

can often not be achieved in a naive way because of the combinatorial complexity\(^2\).

→ Need to find algorithmic or algebraic shortcuts

\(^1\)Using e.g. conjugate priors.
\(^2\)The frequentist counterpart often raises similar issues.
Main issue

Suppose that the calculation wrt $\theta$ raise no issue\(^1\), the calculation of

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

can often not be achieved in a naive way because of the combinatorial complexity\(^2\).

$\rightarrow$ Need to find algorithmic or algebraic shortcuts

Examples.

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

\(^1\)Using e.g. conjugate priors.
\(^2\)The frequentist counterpart often raises similar issues.
Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion
A change-point detection model

Model.
A change-point detection model

Model.

- $K$ segments
A change-point detection model

Model.

- \( K \) segments
- \( T = (\tau_k)_k \) change points

\[ r_k = \left[ \tau_{k-1} + 1; \tau_k \right] \]
A change-point detection model

Model.

- \( K \) segments
- \( T = (\tau_k)_k \) change points
  \[ r_k = [\tau_{k-1} + 1; \tau_k] \]
- \( \theta = (\theta_k)_k \) parameters
A change-point detection model

Model.

- $K$ segments
- $T = (\tau_k)_k$ change points
- $r_k = [\tau_{k-1} + 1; \tau_k]$ for $k = 1, \ldots, K$
- $\theta = (\theta_k)_k$ parameters
- $Y = (Y_t)_{1 \leq t \leq n}$ observed data
- $Y^r = (Y_t)_{t \in r}$ for $r = 1, \ldots, K$

\[
\{Y^r\}_r \text{ indep, } \quad Y^r \sim p(\cdot | \theta_r)
\]
A change-point detection model

Model.

- **K** segments
- **T** = (τ_k)_k change points
  \[ r_k = [τ_{k-1} + 1; τ_k] \]
- **θ** = (θ_k)_k parameters
- **Y** = (Y_t)_{1 ≤ t ≤ n} observed data
  \[ Y^r = (Y_t)_{t ∈ r} \]

Bayesian version: on the top of this, add \( p(K), p(T|K), p(θ|K) \).
Maximum likelihood inference (1/2)

Log-likelihood:

$$\log p(Y; \theta, T) = \sum_{r \in T} \log p(Y^r; \theta^r)$$
Maximum likelihood inference (1/2)

Log-likelihood:

$$\log p(Y; \theta, T) = \sum_{r \in T} \log p(Y^r; \theta^r)$$

Inference

- continuous part ($\theta$):

$$\hat{\theta}_r = \arg\max_{\theta_r} \log p(Y^r; \theta^r) \quad \rightarrow \text{standard MLE}$$
Maximum likelihood inference (1/2)

Log-likelihood:
\[
\log p(Y; \theta, T) = \sum_{r \in T} \log p(Y^r; \theta^r)
\]

Inference

- continuous part ($\theta$):
  \[
  \hat{\theta}_r = \arg \max_{\theta_r} \log p(Y^r; \theta^r) \quad \rightarrow \text{standard MLE}
  \]

- discrete part ($T$):
  \[
  \hat{T} = \arg \max_T \sum_{r \in T} \log p(Y^r; \hat{\theta}^r) = \arg \max_T \sum_{r \in T} \log \hat{p}(Y^r)
  \]
  \[
  \rightarrow \text{discrete optimization problem}
  \]
Maximum likelihood inference (2/2)

Segmentation space $\mathcal{T} = \mathcal{T}_{1:n}^K$ = set of all possible segmentations of $[1; n]$ with $K$ segments:

$$|\mathcal{T}| = \binom{n - 1}{K - 1} \approx \left(\frac{n}{K}\right)^K$$

$\rightarrow$ exhaustive search is prohibited.
Maximum likelihood inference (2/2)

Segmentation space $\mathcal{T} = \mathcal{T}_{1:n}^K = \text{set of all possible segmentations of } [1; n] \text{ with } K \text{ segments}$:

$$|\mathcal{T}| = \binom{n-1}{K-1} \approx \left( \frac{n}{K} \right)^K$$

→ exhaustive search is prohibited.

Dynamic programming allows to retrieve $\hat{T} [1]$ using

$$\max_{T \in \mathcal{T}_{1:j}^K} \sum_{r \in T} \log \hat{p}(Y^r) = \max_{K-1 \leq i < j} \left( \max_{T \in \mathcal{T}_{1:i-1}^{K-1}} \sum_{r \in T} \log \hat{p}(Y^r) \right) + \log \hat{p}(Y^{[i+1:j]})$$
Maximum likelihood inference (2/2)

Segmentation space $\mathcal{T} = \mathcal{T}_{1:n}^K =$ set of all possible segmentations of $[1; n]$ with $K$ segments:

$$|\mathcal{T}| = \binom{n-1}{K-1} \approx \left(\frac{n}{K}\right)^K$$

→ exhaustive search is prohibited.

Dynamic programming allows to retrieve $\widehat{T}[1]$ using

$$\max_{T \in \mathcal{T}_{1:j}^K} \sum_{r \in T} \log \hat{p}(Y^r) = \max_{K-1 \leq i < j} \left( \max_{T \in \mathcal{T}_{1:i}^{K-1}} \sum_{r \in T} \log \hat{p}(Y^r) \right) + \log \hat{p}(Y^{[i+1:j]})$$

Further inference is hard to carry out

→ Standard likelihood theory does not apply to discrete parameters
  (no simple confidence intervals for the $\tau_k$).
→ Bayesian inference can circumvent some difficulties.
Bayesian inference

Factorability assumptions

- Prior distribution for the segmentation:
  \[ p(T|K) = \prod_{r \in T} a_r, \quad \text{e.g.} \quad a_r = n_r^\alpha \]

- Independent parameters in each segment (hyper-Markov assumption):
  \[ p(\theta|T) = \prod_{r \in T} p(\theta_r) \]

- Data are independent from one segment to another
  \[ p(Y|T, \theta) = \prod_{r \in T} p(Y^r|\theta_r) \]
Some quantities of interest

Marginal likelihood.

\[
p(Y|K) = \sum_{T \in \mathcal{T}^K} \int p(Y, \theta, T|K) \, d\theta \propto \sum_{T \in \mathcal{T}^K} \prod_{r \in T} a_r p(Y^r)
\]

where \( p(Y^r) = \int p(Y^r|\theta_r)p(\theta^r) \, d\theta_r \) and the normalizing constant is

\[
\sum_{T \in \mathcal{T}^K} \prod_{r \in T} a_r.
\]
Some quantities of interest

Marginal likelihood.

\[ p(Y|K) = \sum_{T \in T^K} \int p(Y, \theta, T|K) \, d\theta \propto \sum_{T \in T^K} \prod_{r \in T} a_r p(Y^r) \]

where \( p(Y^r) = \int p(Y^r|\theta_r) p(\theta^r) \, d\theta_r \) and the normalizing constant is

\[ \sum_{T \in T^K} \prod_{r \in T} a_r. \]

Posterior distribution of a change-point.

\[ \Pr\{\tau_k = t|Y, K\} \propto \left( \sum_{T \in T^k_{1:t}} \prod_{r \in T} a_r p(Y^r) \right) \left( \sum_{T \in T^{K-k}_{t+1:n}} \prod_{r \in T} a_r p(Y^r) \right) \]
Summing over segmentations [10]

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

\[
A_{i,j+1} = f_r \quad \text{for } r = [i, j]
\]
Summing over segmentations [10]

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

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

Then

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

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

S. Robin (INRA / AgroParisTech) Exact inference with discrete parameters FMJH '16 15 / 44
Summing over segmentations [10]

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

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

Then

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

→ all terms are computed in \(O(Kn^2)\).

- To compute \(p(Y)\), take \(f_r = a_r p(Y^r)\).
- Similar ideas in [6].
Summing over segmentations [10]

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

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

Then

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

→ all terms are computed in \(O(Kn^2)\).

- To compute \(p(Y)\), take \(f_r = a_r p(Y^r)\).
- Similar ideas in [6].
- ’sum-product’ = counterpart of ’max-sum’ in the dynamic programming algorithm.
Summing over segmentations [10]

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

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

Then

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

→ all terms are computed in \(O(Kn^2)\).

▶ To compute \(p(Y)\), take \(f_r = a_r p(Y^r)\).

▶ Similar ideas in [6].

▶ 'sum-product' = counterpart of 'max-sum' in the dynamic programming algorithm.

→ R package EBS (exact Bayesian segmentation) [4]
Illustration: Of exons, introns and UTR’s

Regions for a same gene are not adjacent along the genome

[Wikipedia]
Illustration: Of exons, introns and UTR’s

Regions for a same gene are not adjacent along the genome

- The transcribed regions are made of both exons and untranslated regions (UTR)
- Alternative splicing: some exons can be skipped or the boundaries may vary.
Posterior distribution of transcript boundaries in yeast

RNA-seq data:

One gene

✓

Three growth conditions
A, B, C

Change-point detection  Bayesian inference
Comparing change-point locations \([4]\)

One series. We know how to compute (in \(O(Kn^2)\))

\[
Pr\{\tau_k = t | Y, K\} \quad \text{or} \quad Pr\{\tau_k = t | Y\}.
\]

\(^3\)Requires a probability change, as \(Y^1, \ldots, Y^I\) are not independent conditionally on \(\tau^1_k = \cdots = \tau^I_k\).
Comparing change-point locations [4]

One series. We know how to compute (in $O(Kn^2)$)

$$\Pr\{\tau_k = t | Y, K\} \quad \text{or} \quad \Pr\{\tau_k = t | Y\}.$$ 

Two series ($Y^A$, $Y^B$): Consider the shift of the $k$th change-point

$$\Pr\{\tau^A_k - \tau^B_k = 0 | Y^A, Y^B, K^A, K^B\}$$

\[\text{\footnotesize Requires a probability change, as } Y^1, \ldots, Y^I \text{ are not independent conditionally on } \tau^1_k = \cdots = \tau^I_k.\]
Comparing change-point locations \[4\]

**One series.** We know how to compute (in $O(Kn^2)$)

\[
\Pr\{\tau_k = t|Y, K\} \quad \text{or} \quad \Pr\{\tau_k = t|Y\}.
\]

**Two series ($Y^A, Y^B$):** Consider the shift of the $k$th change-point

\[
\Pr\{\tau^A_k - \tau^B_k = 0|Y^A, Y^B, K^A, K^B\}
\]

**$I$ series ($Y^1, \ldots, Y^I$):** Check if the $k$th change-point is conserved\(^\text{3}\):

\[
\Pr\{\tau^1_k = \cdots = \tau^I_k|Y^1, \ldots, Y^I, K^1, \ldots, K^I\}
\]

\(^3\text{Requires a probability change, as } Y^1, \ldots, Y^I \text{ are not independent conditionally on } \tau^1_k = \cdots = \tau^I_k.\)
Boundary shifts between conditions

3 comparisons \((A/B, A/C, B/C) \times 4\) change points:
Comparing transcript boundaries

Setting $\Pr\{\tau_k^A = \tau_k^B | K\} = 1/2$. 

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
<th>$\tau_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Pr{\tau_k^A = \tau_k^B</td>
<td>Y, K}$</td>
<td>0.32</td>
<td>0.30</td>
<td>0.99</td>
</tr>
<tr>
<td>$\Pr{\tau_k^A = \tau_k^C</td>
<td>Y, K}$</td>
<td>$4 \times 10^{-4}$</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td>$\Pr{\tau_k^B = \tau_k^C</td>
<td>Y, K}$</td>
<td>$5 \times 10^{-2}$</td>
<td>0.60</td>
<td>0.99</td>
</tr>
<tr>
<td>$\Pr{\tau_k^A = \tau_k^B = \tau_k^C</td>
<td>Y, K}$</td>
<td>$10^{-3}$</td>
<td>0.99</td>
<td>0.99</td>
</tr>
</tbody>
</table>

→ Differences at the UTR’s end but not at internal exon boundaries.
Various isoforms in yeast?

$$\Pr\{\tau_k^A = \tau_k^B = \tau_k^C \mid Y, K\}$$ for all yeast genes with 2 expressed exons

$$p_0 = (.5, .5, .5, .5)$$

$$p_0 = (.9, .99, .99, .9)$$
Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion
Graphical model framework

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

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

\( G \) reveals the structure of conditional independences between the variables \( Y_1, \ldots, Y_p \).
Graphical model

Means that

\[ p(Y_1, \ldots, Y_8) \propto \psi_1(Y_1, Y_2, Y_3) \times \psi_2(Y_1, Y_4) \psi_3(Y_1, Y_5) \times \psi_4(Y_2, Y_6) \psi_5(Y_6, Y_7) \times \psi_6(Y_3, Y_8) \]

which implies\(^4\) that

\[ Y_4 \perp Y_3 \mid Y_1 \]

\[ (Y_6, Y_7) \perp Y_3 \mid Y_2 \]

\[ \ldots \]

\(^4\)Under fairly general assumptions on \(p\)
Graphical model

Means that

\[ p(Y_1, \ldots, Y_8) \propto \psi_1(Y_1, Y_2, Y_3) \times \psi_2(Y_1, Y_4) \psi_3(Y_1, Y_5) \times \psi_4(Y_2, Y_6) \psi_5(Y_6, Y_7) \times \psi_6(Y_3, Y_8) \]

which implies that

\[ Y_4 \perp Y_3 \mid Y_1 \]
\[ (Y_6, Y_7) \perp Y_3 \mid Y_2 \]
\[ \ldots \]

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

\(^4\text{Under fairly general assumptions on } p\)
Tree-structured network

Suppose the graph $G$ is a tree $T$, $p(Y)$ is Markov wrt $T$ iff

$$p(Y|\theta) = \prod_j p(Y_j|\theta_j) \prod_{(j,k) \in T} \frac{p(Y_j, Y_k|\theta_{jk})}{p(Y_j|\theta_j)p(Y_k|\theta_k)}$$

$$= \prod_{(j,k) \in T} p(Y_j, Y_k|\theta_{jk}) \left/ \prod_j p^{d_j-1}(Y_j|\theta_j) \right.$$ 

where $d_j$ is the degree (number of neighbors in $T$) of node $j$. 
Tree-structured network

Suppose the graph $G$ is a tree $T$, $p(Y)$ is Markov wrt $T$ iff

$$p(Y|\theta) = \prod_j p(Y_j|\theta_j) \prod_{(j,k) \in T} \frac{p(Y_j, Y_k|\theta_{jk})}{p(Y_j|\theta_j)p(Y_k|\theta_k)}$$

$$= \prod_{(j,k) \in T} p(Y_j, Y_k|\theta_{jk}) \left/ \prod_j p^{d_j-1}(Y_j|\theta_j) \right.$$  

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

Tree structure assumption.

- 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.
Maximum likelihood inference (1/2)

Log-likelihood.

$$\log p(Y; \theta, T) = \sum_{(j,k) \in T} \log p(Y_j, Y_k | \theta_{jk}) - \sum_j (d_j - 1) \log p(Y_j | \theta_j)$$

$$= \sum_j \log p(Y_j | \theta_j) + \sum_{(j,k) \in T} \log \frac{p(Y_j, Y_k | \theta_{jk})}{p(Y_j | \theta_j)p(Y_j | \theta_j)}$$
Maximum likelihood inference (1/2)

Log-likelihood.

\[
\log p(Y; \theta, T) = \sum_{(j,k) \in T} \log p(Y_j, Y_k | \theta_{jk}) - \sum_j (d_j - 1) \log p(Y_j | \theta_j)
\]

\[
= \sum_j \log p(Y_j | \theta_j) + \sum_{(j,k) \in T} \log \frac{p(Y_j, Y_k | \theta_{jk})}{p(Y_j | \theta_j)p(Y_k | \theta_k)}
\]

Inference:

- continuous part (\(\theta\)): MLE

\[
\hat{\theta}_j = \arg \max_{\hat{\theta}_j} \log p(\{Y_{ij}\}_i; \theta_j), \quad \hat{\theta}_{jk} = \arg \max_{\hat{\theta}_{jk}} \log p(\{(Y_{ij}, Y_{ik})\}_i; \theta_{jk})
\]
Maximum likelihood inference (1/2)

Log-likelihood.

\[
\log p(Y; \theta, T) = \sum_{(j,k) \in T} \log p(Y_j, Y_k | \theta_{jk}) - \sum_j (d_j - 1) \log p(Y_j | \theta_j)
\]

\[
= \sum_j \log p(Y_j | \theta_j) + \sum_{(j,k) \in T} \log \frac{p(Y_j, Y_k | \theta_{jk})}{p(Y_j | \theta_j)p(Y_j | \theta_j)}
\]

Inference:

- **continuous part (\(\theta\)):** MLE

\[
\hat{\theta}_j = \arg \max_{\theta_j} \log p(\{Y_{ij}\}_i; \theta_j), \quad \hat{\theta}_{jk} = \arg \max_{\theta_{jk}} \log p(\{(Y_{ij}, Y_{ik})\}_i; \theta_{jk})
\]

- **discrete part (\(T\))

\[
\hat{T} = \arg \max_T \sum_{(j,k) \in T} \log \frac{p(Y_j, Y_k | \hat{\theta}_{jk})}{p(Y_j | \hat{\theta}_j)p(Y_k | \hat{\theta}_k)}
\]
Maximum likelihood inference (2/2)

Chow & Liu algorithm [3]:

Taking

\[ f_{jk} = \log \frac{p(Y_j, Y_k | \hat{\theta}_{jk})}{p(Y_j | \hat{\theta}_j) p(Y_j | \hat{\theta}_j)} \]

as the weight of edge \((j, k)\),

\[ \hat{T} = \arg \max_T \sum_{(j, k) \in T} f_{jk} \]

is the maximum spanning tree with weights \(\{f_{jk}\}\), which can be retrieved by Kruskal’s algorithm in \(O(p^2)\) [7].
Maximum likelihood inference (2/2)

Chow & Liu algorithm [3]: Taking

$$f_{jk} = \log \frac{p(Y_j, Y_k | \hat{\theta}_{jk})}{p(Y_j | \hat{\theta}_j) \ p(Y_j | \hat{\theta}_j)}$$

as the weight of edge \((j, k)\),

$$\hat{T} = \arg \max_T \sum_{(j,k) \in T} f_{jk}$$

is the maximum spanning tree with weights \(\{f_{jk}\}\), which can be retrieved by Kruskal’s algorithm in \(O(p^2)\) [7].

Retrieves the maximum likelihood tree but with no measure of uncertainty.

→ Exploring the whole tree space allows to evaluate uncertainty.

→ Bayesian inference can again be a solution.
Bayesian setting \cite{13}

Model:

- Prior on $T$: $p(T)$
- Prior on $\theta$: $p(\theta|T)$ \quad $\rightarrow$ \quad Posterior: $p(T|Y)$
- Likelihood: $p(Y|\theta, T)$

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

Prior on $\theta$: displays factorability properties, i.e. needs to satisfy $p(\theta_{jk}|T) = p(\theta_{jk})$ for all $T \ni (j,k)$.

\[
\text{Compatible family of strong Markov hyper-distributions \cite{5}: multinomial-Dirichlet (conjugacy), normal-Wishart (conjugacy), Gaussian copulas (numerical integration), ...?}
\]
Bayesian setting [13]

Model:
- prior on $T$: $p(T)$
- prior on $\theta$: $p(\theta|T)$ → posterior: $p(T|Y)$
- likelihood: $p(Y|\theta, T)$

Prior on $T$: factorizes over the edges:

$$p(T) \propto \prod_{(j,k) \in T} a_{jk}$$
Bayesian setting \[13\]

Model:
- prior on $T$: $p(T)$
- prior on $\theta$: $p(\theta \mid T)$ \rightarrow posterior: $p(T \mid Y)$
- likelihood: $p(Y \mid \theta, T)$

Prior on $T$: factorizes over the edges:

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

Prior on $\theta$: displays factorability properties, i.e. needs to satisfy

$$p(\theta_{jk} \mid T) \equiv p(\theta_{jk}) \text{ for all } T \ni (j, k).$$

\rightarrow Compatible family of strong Markov hyper-distributions \[5\]: multinomial-Dirichlet (conjugacy), normal-Wishart (conjugacy), Gaussian copulas (numerical integration), ...?
Quantities of interest

Marginal distribution.

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

where \( \mathcal{T} \) stands for the set of all spanning trees and \( p(Y_j), p(Y_j, Y_k) \) are integrals with respect to \( \theta_j \) and \( \theta_{j,k} \), respectively.
Quantities of interest

Marginal distribution.

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

where \( \mathcal{T} \) stands for the set of all spanning trees and \( p(Y_j), p(Y_j, Y_k) \) are integrals wrt \( \theta_j \) and \( \theta_{j,k} \), resp.

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.

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

where \( \mathcal{T} \) stands for the set of all spanning trees and \( p(Y_j), p(Y_j, Y_k) \) are integrals wrt \( \theta_j \) and \( \theta_{j,k} \), resp.

Posterior probability for an edge to be absent.

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

Typical form:

\[ \sum_{T \in \mathcal{T}} \prod_{(j, k) \in T} f_{jk} \]

with cardinality of \( \mathcal{T} = p^{p-2} \).
Summing over spanning trees

Matrix-tree theorem. [2]

- $F = [f_{jk}]$: a symmetric matrix with $f(j, j) = 0$, $f_{jk} > 0$;
- $\Delta = [\Delta_{jk}]$ its Laplacian
Summing over spanning trees

Matrix-tree theorem. [2]

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

Then the minors \( \Delta^{uv} \) of \( \Delta \) are all equal to

\[
\sum_{T \in \mathcal{T}} \prod_{(j, k) \in T} f_{jk}.
\]
Matrix-tree theorem. [2]

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

Then the minors $\Delta^{uv}$ of $\Delta$ are all equal to

$$\sum_{T \in \mathcal{T}} \prod_{(j,k) \in T} f_{jk}.$$

- Can be used to compute $p(Y)$, the normalizing constant of $p(T)$, ... at the cost of computing a $p \times p$ determinant.
- Already used in [8] for tree learning.
- Again 'sum-product' in place of 'max-sum'.
Posterior probability of an edge

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

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

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 be computed in $O(p^3)$.  
→ R package Saturnin (spanning trees used for network inference) [13]
Simulations: ROC curves for edge detection
For various graph topologies \((p = 25, n = 25, 50, 200, B = 100 \text{ simulations})\)
Simulations: Comparison with sampling among DAGs

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

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 \[11\]

'ground truth'

posterior probabilities

most likely tree

second most likely tree
Detecting changes in a graphical model

Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion
Change-point in a graphical model

Problem:

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?
Detecting changes in a graphical model

Change-point in a graphical model

Problem: [12]

- 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?
**Change-point in a graphical model**

**Problem:** [12]

- 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?
Handling two sums

Double discrete structure:

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

\[ \rightarrow \text{sum over} \approx (N/K)^K p^K(p-2) \text{ terms.} \]
Handling two sums

Double discrete structure:

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

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

Combining the two preceding tricks:
- Summing of all segmentations in \( O(KN^2) \),
- Summing over all trees in \( O(p^3) \) (one tree per possible segment)

\[ \rightarrow \text{Global complexity } = O(\max\{K, p^3\}N^2) \]
Inference

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

- $P(\text{change-point at time } t|K, Y)$
- $P(\text{edge } (i, j) \text{ present at time } t|K, Y)$
- $P(\text{edge } (i, j) \text{ remains present along time}|Y)$
- $P(K \text{ segments}|Y)$. 
Inference

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

- $P(\text{change-point at time } t|K, Y)$
- $P(\text{edge } (i,j) \text{ present at time } t|K, Y)$
- $P(\text{edge } (i,j) \text{ remains present along time } Y)$
- $P(K \text{ segments } Y)$.

+ Network comparison

- $P(T_1 = T_2|Y_1, Y_2)$
- $P(\text{edge } (i,j) \text{ present in both } T_1 \text{ and } T_2|Y_1, Y_2)$. 
Some simulations

Tree-structured network. Complete network.

From top to bottom: \( N = 70, 140, 210 \).
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:

![Graphical Model](image)

Inferred networks:
Outline

Bayesian inference with discrete parameters

Change-point detection

Network inference

Detecting changes in a graphical model

Discussion
To summarize.

- Exact Bayesian inference can still 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 this is possible → ad-hoc developments.
Discussion

To summarize.

- Exact Bayesian inference can still 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 this is possible → ad-hoc developments.

Future works.

- Dealing with dependency along time.
- Influence of the prior: $p(T)$ depends on $n$ and/or $p$.
- The exact evaluation of the key quantity raises numerical issues.
References I

I. E. Auger and C. E. Lawrence.
Algorithms for the optimal identification of segment neighborhoods.

Seth Chaiken.
A Combinatorial Proof of the All Minors Matrix Tree Theorem.

C.K. Chow and C.N. Liu.
Approximating Discrete Probability Distributions with Dependence Trees.

A. Cleynen and S. Robin.
Comparing change-point location in independent series.

A. P. Dawid and S. L. Lauritzen.

P. Fearnhead.
Exact and efficient bayesian inference for multiple changepoint problems.

Joseph B. Kruskal.
On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem.

M. Meilä 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.

Exact posterior distributions over the segmentation space and model selection for multiple change-point detection problem.

K. Sachs, O. Perez, D. Pe’er, D. A. Lauffenburger, and G. P. N.
Causal protein-signaling networks derived from multiparameter single-cell data.

L. Schwaller and S. Robin.
Exact Bayesian inference for off-line change-point detection in tree-structured graphical models.

Bayesian Inference of Graphical Model Structures Using Trees.