A sum-product algorithm with polynomials for computing exact derivatives of the likelihood in parametric Bayesian networks

Alexandra Lefebvre and Grégory Nuel
(PhD supervisors: A. Duval and G. Nuel)

LPSM (Probability and Statistics), CNRS, *Sorbonne Université, Paris, France*

9th International Conference on
Probabilistic Graphical Models
PGM, Prague, September 11 - 14, 2018

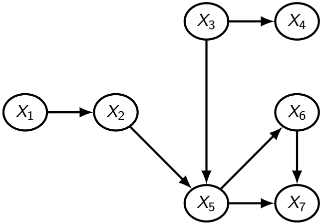
Outline

1. Context
   - Recall, notations
   - Likelihood of a Bayesian network
   - How to compute derivatives of the likelihood ?
2. Methods
   - Derivative generating function
   - Leibniz's product
   - Sum-product algorithm with polynomials
3. Applications
   - Binary toy-example
   - Two-point linkage in genetics

Outline

Notations, brief recalls and objective

- **Parametric Bayesian network (BN)**



$X_{\mathcal{U}} = \{X_u\}_{u \in \mathcal{U}}$

pa_u : set of labels for parents of X_u

$\mathcal{X}_u$: set of values for X_u

$\mathrm{ev} = \{X_u \in \mathcal{X}_u^* \subset \mathcal{X}_u\}, u \in \mathcal{U}$

$$\varphi_u(X_u | X_{\mathrm{pa}_u}; \theta) = \begin{cases} \mathbb{P}(X_u, X_u \in \mathcal{X}_u^* | X_{\mathrm{pa}_u}; \theta) \\ 0 \text{ if } X_u \notin \mathcal{X}_u^* \end{cases}$$

with $\theta \in \mathbb{R}^p, p \geq 1$

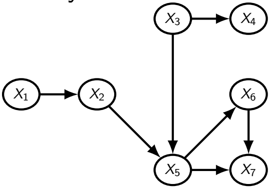
- **Likelihood of a BN**

$$\mathbb{P}(\mathrm{ev}|\theta) = \sum_{X_{\mathcal{U}}} \prod_{u \in \mathcal{U}} \varphi_u(X_u | X_{\mathrm{pa}_u}; \theta) = L(\theta)$$

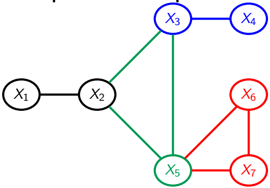
Notations, brief recalls and objective

- **Sum-product algorithm**

Bayesian network

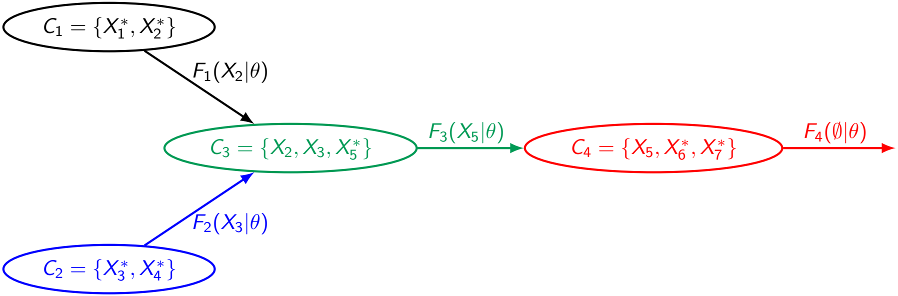


Clique decomposition



$$L(\theta) = \sum_{X_5} \sum_{X_6} \sum_{X_7} \Bigg\{ \sum_{X_2} \sum_{X_3} \Bigg(\overbrace{\sum_{X_1} \varphi_1(X_1)\varphi_2(X_2|X_1)}^{F_1(X_2)} \overbrace{\sum_{X_4} \varphi_3(X_3)\varphi_4(X_4|X_3)}^{F_2(X_3)} \varphi_5(X_5|X_2,X_3) \Bigg) \varphi_6(X_6|X_5)\varphi_7(X_7|X_5,X_6) \Bigg\}$$

Notations, brief recalls and objective



$$\forall i \in \{1, \dots 4\}; \quad \phi_i(C_i|\theta) = \prod_{X_u \in C_i{}^*} \varphi_u(X_u|X_{\mathrm{pa}_u}; \theta)$$

$$F_i(S_i|\theta) = \sum_{C_i \setminus S_i} \left(\prod_{j \in \mathrm{from}_i} F_j(S_j|\theta) \right) \times \phi_i(C_i|\theta); \qquad F_4(\emptyset|\theta) = L(\theta)$$

Our objective: Compute $L(\theta)$ and its derivatives up to a chosen order d.

State of art

- **Sensitivity analysis**: express $L(\theta)$ as a polynomial in θ
  $\Rightarrow$ not always possible (*e.g.* $\mathbb{P}(X_u = 1) = e^\theta/(1 + e^\theta)$)
  $\Rightarrow$ prohibitive if degree in θ is large (*e.g.* θ in many potentials)
- **Smoothing recursions** (Cappé and Moulines, 2005): express $\nabla^d \log L(\theta)$ as function of $\mathbb{E}[\nabla^d \log \varphi_u | \mathrm{ev}]$ (Fisher-Louis' formulas)
  $\Rightarrow$ developed for Hidden Markov Models
  $\Rightarrow$ many terms and complex recursions when d increases
- **Polynomial computations**: perform sum-product with polynomials
  $\Rightarrow$ moments of additive functional in BN (Cowel, 1992; Nilsson, 2001)
  $\Rightarrow$ moment generating function / probability generating function for regular expr. in Markov models (Nuel, 2008, 2010)

Our idea: Use polynomial computations in the framework of derivatives

Outline

Method: Derivative generating function

$$L(\theta) = \sum_{X_{\mathcal{U}}} \prod_{u \in \mathcal{U}} \varphi_u(X_u | X_{\mathrm{pa}_u}; \theta) \Rightarrow \sum_{X_{\mathcal{U}}} \bigstar_{u \in \mathcal{U}} D^d \varphi_u(X_u | X_{\mathrm{pa}_u}; \theta)$$

- **Derivative generating function (Dgf)**
  Let f be a function of class $\mathcal{C}^d$ of $\theta \in \mathbb{R}$, $d \in \mathbb{N}$

$$D^d f(\theta) = \sum_{k=0}^{d} f^{(k)}(\theta) z^k$$

  where z is a dummy variable.
  Example: $D^2 f(\theta) = f(\theta) + f'(\theta) z + f''(\theta) z^2$

  $\Rightarrow$ **Polynomial potentials of degree d**

$$\forall u \in \mathcal{U}; \quad D^d \varphi_u(X_u | X_{\mathrm{pa}_u}; \theta) = \sum_{k=0}^{d} {\varphi_u}^{(k)}(X_u | X_{\mathrm{pa}_u}; \theta) z^k$$

Définition "Leibniz's product"

$$P = \sum_{k=0}^{d} a_k z^k \quad \text{with} \quad a_k = f^{(k)}(\theta)$$

$$Q = \sum_{k=0}^{d} b_k z^k \quad \text{with} \quad b_k = g^{(k)}(\theta)$$

$$P \star Q = \sum_{k=0}^{d} c_k z^k \quad \text{with} \quad c_k = (fg)^{(k)}(\theta)$$

$$(fg)^{(k)}(\theta) = \sum_{i=0}^{k} \binom{k}{i} f^{(i)}(\theta)\, g^{(k-i)}(\theta) \Rightarrow c_k = \sum_{i=0}^{k} \binom{k}{i} a_i b_{k-i}$$

$$P \star Q = \sum_{k=0}^{d} \sum_{i=0}^{k} \binom{k}{i} a_i b_{k-i} z^k$$

Multidimensional parameter $\Theta = \theta_1, \ldots, \theta_p$

- Dgf of f function $\mathcal{C}^d$ of $\Theta \in \mathbb{R}^p$

$$D^d f(\Theta) = \sum_{k_1+\ldots+k_p \leq d} \frac{\partial^{(k_1+\ldots+k_p)} f(\theta)}{\partial {\theta_1}^{k_1} \ldots \partial {\theta_p}^{k_p}} {z_1}^{k_1} \ldots {z_p}^{k_p}$$

- "Leibniz product"

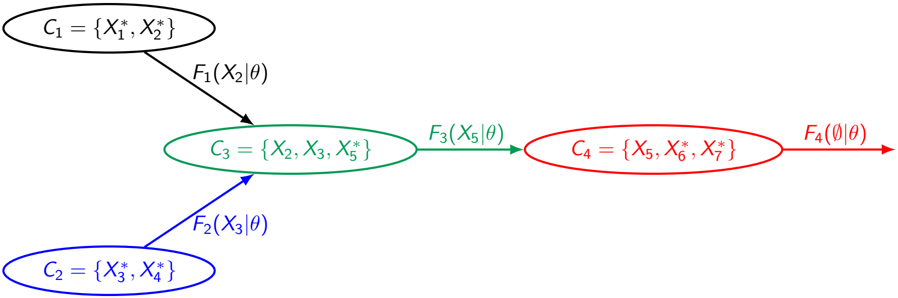
$$P = \sum_{k_1+\ldots+k_p \leq d} a_{k_1,\ldots,k_p} {z_1}^{k_1} \ldots {z_p}^{k_p} \quad \text{with} \quad a_{k_1,\ldots,k_p} = \frac{\partial^{(k_1+\ldots+k_p)} f(\Theta)}{\partial {\theta_1}^{(k_1)} \ldots \partial {\theta_p}^{(k_p)}}$$

Q idem with coef. b instead of a and function g instead of f

$$P \star Q = \sum_{k_1+\ldots+k_p \leq d} c_{k_1,\ldots,k_p} {z_1}^{k_1} \ldots {z_p}^{k_p}$$

with $$c_{k_1,\ldots,k_p} = \sum_{i_1=0}^{k_1} \ldots \sum_{i_p=0}^{k_p} \binom{k_1}{i_1} \ldots \binom{k_p}{i_p} a_{k_1-i_1,\ldots,k_p-i_p} b_{i_1,\ldots,i_p}$$

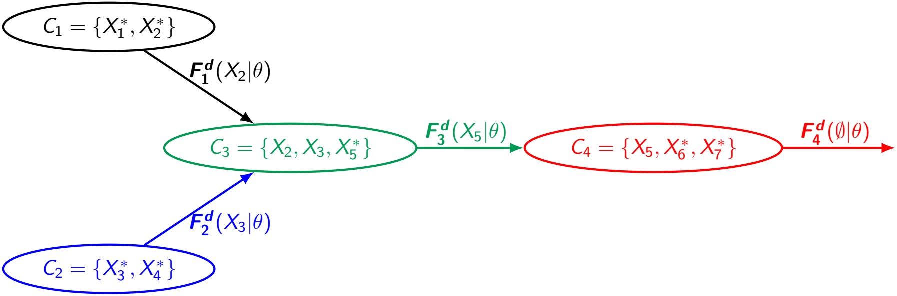
Computation : $\sum \prod \phi_u \rightarrow \sum \star D^d \phi_u$



$$\forall i \in \{1, \ldots 4\}; \quad \phi_i(C_i|\theta) = \prod_{X_u \in C_i{}^*} \varphi_u(X_u|X_{\mathrm{pa}_u};\theta)$$

$$F_i(S_i|\theta) = \sum_{C_i \setminus S_i} \left(\prod_{j \in \mathrm{from}_i} F_j(S_j|\theta) \right) \times \phi_i(C_i|\theta)$$

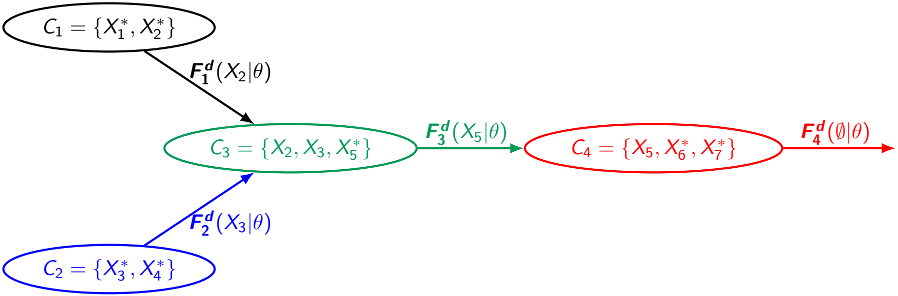
Computation : $\sum \prod \phi_u \rightarrow \sum \bigstar D^d \phi_u$



$$\forall i \in \{1, \dots 4\}; \quad \boldsymbol{\Phi_i^d}(C_i|\theta) = \mathop{\bigstar}_{X_u \in C_i{}^*} D^d \varphi_u(X_u | X_{\mathrm{pa}_u}; \theta)$$

$$\boldsymbol{F_i^d}(S_i|\theta) = \sum_{C_i \setminus S_i} \left(\mathop{\bigstar}_{j \in \mathrm{from}_i} \boldsymbol{F_j^d}(S_j|\theta) \right) \star \boldsymbol{\Phi_i^d}(C_i|\theta)$$

Computation : $\sum \prod \phi_u \to \sum \bigstar D^d \phi_u$



$$\boldsymbol{\Phi_2^d}(X_3, X_4|\theta) = D^d \varphi_3(X_3;\theta) \star D^d \varphi_4(X_4|X_1;\theta)$$

$$\boldsymbol{F_3^d}(X_5) = \sum_{X_2} \sum_{X_3} \boldsymbol{F_1^d}(X_2|\theta) \star \boldsymbol{F_2^d}(X_3|\theta) \star \boldsymbol{\Phi_3^d}(X_2, X_3, X_5|\theta)$$

$$\boldsymbol{F_4^d}(\emptyset|\theta) = D^d L(\theta) = L(\theta) + L'(\theta)z + \ldots + L^{(d)}(\theta)z^d$$

Complexity

- **Empirical computation**

$$L'(\theta) \simeq \frac{L(\theta + h_1) - L(\theta)}{h_1}\,; \quad L''(\theta) \simeq \frac{L(\theta + 2h_2) - 2L(\theta + h_2) + L(\theta)}{{h_2}^2}$$

  for a small enough h_1 and h_2.

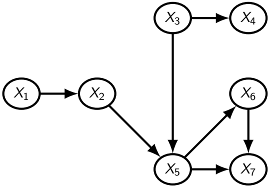
- **Complexities**
  - Let C be the complexity to compute $L(\theta) = \mathbb{P}(\mathrm{ev}|\theta)$
  - Unidimensional parameter & arbitrary $d \rightarrow L(\theta), L'(\theta), \ldots, L^{(d)}(\theta)$
    Complexity $= \mathcal{O}\left(C \times d^2\right)$
  - Multidimensional parameter $(p > 1)$ & $d = 2 \rightarrow L(\theta), \nabla L(\theta), \text{Hess}\, L(\theta)$
    Complexity $= \mathcal{O}\left(C \times p^2\right)$

- **Similar to the empirical method with two main advantages**
  - Exact derivatives
  - No need to choose the step h and converge iteratively to the solution

Outline

1. Context
   - Recall, notations
   - Likelihood of a Bayesian network
   - How to compute derivatives of the likelihood ?
2. Methods
   - Derivative generating function
   - Leibniz's product
   - Sum-product algorithm with polynomials
3. Applications
   - Binary toy-example
   - Two-point linkage in genetics

Application: Simple binary BN



$$\mathbb{P}(X_u = 1 | X_{\mathrm{pa}_u}; \theta) = f_k(\theta), \quad k = \sum_{v \in \mathrm{pa}_u} X_v$$

$$f_k(\theta) = \frac{e^{-0.5+k\theta}}{1 + e^{-0.5+k\theta}}$$

Question: Compute derivatives of $L(\theta)$ up to degree $d = 2$.

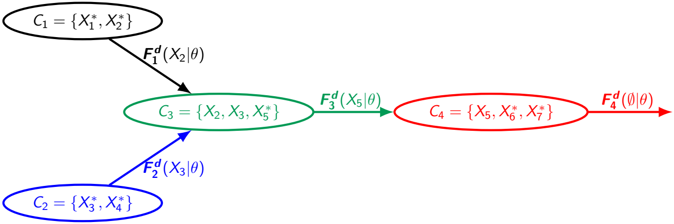
$$D^2\varphi_u \left(X_u \middle| \sum_{v \in \mathrm{pa}_u} X_v = k; \theta \right) = \begin{cases} P_k & \text{if } X_u = 1 \\ 1 - P_k & \text{if } X_u = 0 \end{cases}$$

with $P_k = f_k(\theta) + f'_k(\theta)\ z + f''_k(\theta)\ z^2$

$$f'_k(\theta) = \frac{k \times e^{-0.5+k\theta}}{(1 + e^{-0.5+k\theta})^2} \quad \text{and} \quad f''_k(\theta) = \frac{k^2 \times e^{-0.5+k\theta} \times (1 - e^{-0.5+k\theta})}{(1 + e^{-0.5+k\theta})^3}$$

Results: A sample of chosen polynomials

Simulation with $\theta = 1$ and Observation of $\mathrm{ev} = \{X_1 = 0, X_7 = 1\}$



$\mathbf{\Phi}_2^2(X_3 = 0, X_4 = 0)$	0.38745561900026
$\mathbf{\Phi}_2^2(X_3 = 0, X_4 = 1)$	0.235003712201595
$\mathbf{\Phi}_2^2(X_3 = 1, X_4 = 0)$	$0.142537 - 0.08872346\ z + 0.02173003\ z^2$
$\mathbf{\Phi}_2^2(X_3 = 1, X_4 = 1)$	$0.2350037 + 0.08872346\ z - 0.02173003\ z^2$
$\boldsymbol{F}_3^2(X_5 = 0)$	$0.2767608 - 0.09521838\ z + 0.05045801\ z^2$
$\boldsymbol{F}_3^2(X_5 = 1)$	$0.3456986 + 0.09521838\ z - 0.05045801\ z^2$
$\boldsymbol{F}_4^2(\emptyset) = D^2L(\theta)$	$0.3872484 + 0.1613463\ z - 0.05839165\ z^2$

Results: simulations of n values for $\{X_1, \dots, X_7\}$

- n simulations leading to n_{ab} observed couples $\{X_1 = a, X_7 = b\}$

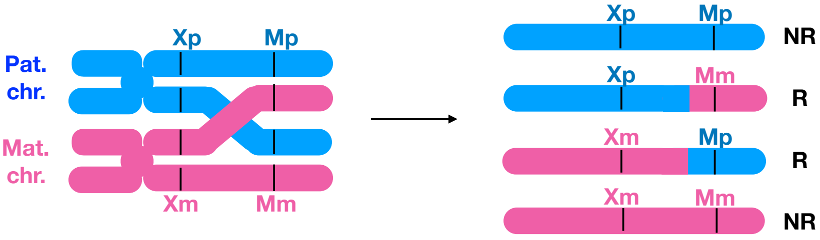
n	n_{00}	n_{01}	n_{10}	n_{11}
50	8	27	3	12
500	123	187	80	110
5000	1154	1873	745	1228
50,000	11878	19277	6671	12174
500,000	117586	193504	68479	120431

- Likelihood of the true parameter $\theta^* = 1$ and its derivatives

n	$\ell_n(1)$	$\ell'_n(1)/n$	$-n/\ell''_n(1)$
50	-6.025×10^1	1.782×10^{-1}	3.173
500	-6.712×10^2	-4.151×10^{-2}	3.059
5000	-6.678×10^3	-9.821×10^{-3}	3.070
50,000	-6.609×10^4	1.910×10^{-3}	3.106
500,000	-6.615×10^5	2.851×10^{-5}	3.096

Application : two-point linkage in genetics

- **Goal** : Locate a gene of interest on the genome.
- **Trick:** uses crossovers (CO), phenomenon happening during meiosis.



- **Assumption:** The closest two genes are on the genome, the less chances to be separated during meisosis
- **Parameter:**

$$\theta = \frac{\#\text{R}}{\#\text{R} + \#\text{NR}} = \mathbb{P}(\text{odd } C) \quad \text{with} \quad C = \#\text{CO}$$

Genetic distance h - Haldane, J. B. S. (1919).

- **Genetic distance *h*:**

  Assumption: $C \sim \mathcal{P}(h) \Rightarrow h = \mathbb{E}[C]$

$$\theta = \mathbb{P}(\text{odd } C) = \sum_{k=0}^{\infty} \mathbb{P}(C = 2k+1) = \sum_{k=0}^{\infty} \frac{e^{-h} h^{(2k+1)}}{(2k+1)!} = \frac{1 - e^{-2h}}{2}$$

$$h = -\frac{\log(1 - 2\theta)}{2} \text{ expressed in morgan (M) or centimorgan (cM)}$$

- **LOD score:**

$$\theta \xrightarrow[h \to \infty]{} \frac{1}{2} \qquad \mathrm{LOD}(\theta) = \log_{10}\left(\frac{L(\theta)}{L(\frac{1}{2})}\right)$$

Example with the KUS family (MENDEL package)

Interest RADIN:

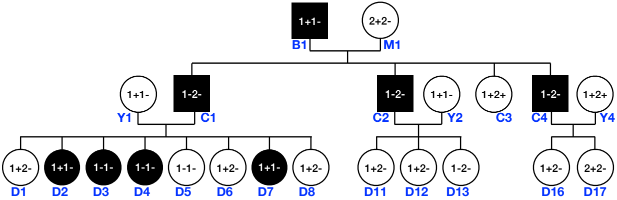
$$\left.\begin{array}{ll} X_p & \in \{1,0\} \\ X_m & \in \{1,0\} \end{array}\right\} \rightarrow \left\{\begin{array}{lll} 11 & \rightarrow & \text{lethal before birth} \\ 10 \text{ or } 01 & \rightarrow & Y=1, \text{ black filling} \\ 00 & \rightarrow & Y=0, \text{ no filling} \end{array}\right.$$

Marker PGM1:

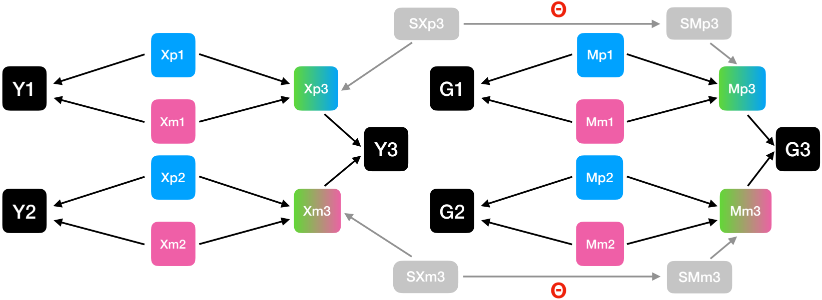
$$\left.\begin{array}{ll} M_p & \in \{1+,1-,2+,2-\} \\ M_m & \in \{1+,1-,2+,2-\} \end{array}\right\} \rightarrow \text{16 configurations}$$

$G \in \{1+1+, 1+1-, \ldots, 2-2-\}$, 10 values

Pedigree (familial structure)



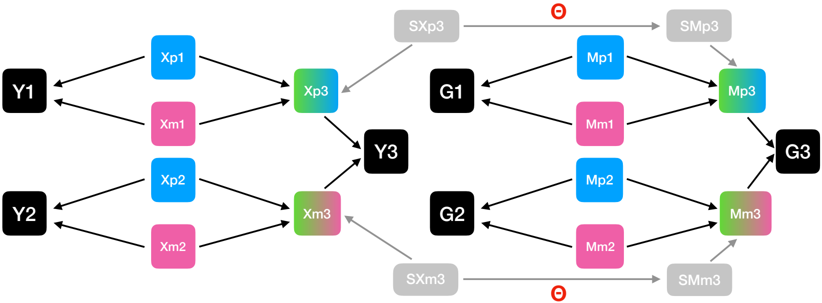
Variables in two-point linkage



Known Structure and **Observed** G and Y.

Latent: Xp & $Xm \in \{1, 0\}$; $\quad Mp$ & $Mm \in \{1+, 1-, 2+, 2-\}$

Variables in two-point linkage



Known Structure and **Observed** G and Y.

Latent: Xp & $Xm \in \{1, 0\}$; Mp & $Mm \in \{1+, 1-, 2+, 2-\}$;
SXp, SXm, SMp, $SMm \in \{\text{pat}, \text{mat}\}$

$$\varphi_{\text{SXp}_3}(\text{SXp}_3 = \text{pat}) = \varphi_{\text{SXp}_3}(\text{SXp}_3 = \text{mat}) = 0.5$$

$$\varphi_{\text{SMp}_3}(\text{SMp}_3 = \text{SXp}_3 | \text{SXp}_3) = (1 - \theta)$$

$$\varphi_{\text{SMp}_3}(\text{SMp}_3 \neq \text{SXp}_3 | \text{SXp}_3) = \theta$$

Build polynomial potentials

- Polynomials of degree 0 (All except those related to selector of M)
- Polynomials of degree strictly positive (Related to selector of M)

$$D^2\varphi_{\mathrm{SMp}_3}(\mathrm{SMp}_3 = \mathrm{SXp}_3|\mathrm{SXp}_3) = (1-\theta) - z$$

$$D^2\varphi_{\mathrm{SMp}_3}(\mathrm{SMp}_3 \neq \mathrm{SXp}_3|\mathrm{SXp}_3) = \theta + z$$

- Reparametrization: $\theta = e^{\beta}/(1+e^{\beta})$

$$D^2\varphi_{\mathrm{SMp}_3}(\mathrm{SMp}_3 = \mathrm{SXp}_3|\mathrm{SXp}_3) = \frac{1}{1+e^{\beta}} - \frac{e^{\beta}}{(1+e^{\beta})^2}\, z - \frac{e^{\beta}(1-e^{\beta})}{(1+e^{\beta})^3}\, z^2$$

$$D^2\varphi_{\mathrm{SMp}_3}(\mathrm{SMp}_3 \neq \mathrm{SXp}_3|\mathrm{SXp}_3) = \frac{e^{\beta}}{1+e^{\beta}} + \frac{e^{\beta}}{(1+e^{\beta})^2}\, z + \frac{e^{\beta}\,(1-e^{\beta})}{(1+e^{\beta})^3}\, z^2$$

Computational shortcuts

- **Load evidence** and remove sparse potentials & unobserved leaf node.
- **Preprocess variables related to the marker**
  - Build **boolean** potentials related to G, M_p & M_m.
  - Perform Forward Backward & marginalization on truncated DAG.
  - $\Rightarrow$ Shrink state spaces of G, M_p & M_m.
- **Preprocess variables related to the target**
  - Build **boolean** potentials related to Y, X_p & X_m.
  - Perform Forward Backward & marginalization on truncated DAG.
  - Shrink state spaces of Y, X_p & X_m.
- **Break edges that became unnessary**

References:

Kong, A. (1991).
Cottingham,Jr. R. W., (1993).
Lauritzen, S. L. and Sheehan, N. A., (2003).

Computational shortcuts - Comparison

STEPS : **1** - No preprocess. **2** - Load evidence & remove unobs. leaves.
3 - Preprocess. **4** - Break edges that became unnecessary.

- KUS ($n = 22, \#\mathrm{NA} = 0$)

	var	edges	TW	Complexity
1	200	506	10	518,576
2	200	506	10	6,410
3	200	506	10	1,008
4	156	48	5	476

- BOD+STO ($n = 12, \#\mathrm{NA} = 0$)

	var	edges	TW	Complexity
1	100	226	10	107,760
2	100	226	10	3,431
3	100	226	10	1,946
4	156	48	5	336

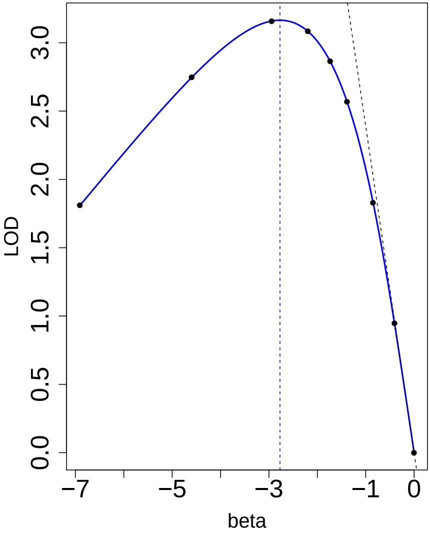
- NEU ($n = 32, \#\mathrm{NA} = 4$)

	var	edges	TW	Complexity
1	288	720	10	739,936
2	284	712	10	12,580
3	284	712	10	1,698
4	225	251	5	818

- KRA ($n = 27, \#\mathrm{NA} = 7$)

	var	edges	TW	Complexity
1	242	602	10	537,176
2	235	585	10	79,095
3	235	585	10	75,239
4	205	360	10	74,292

Results: Derivatives



$$\mathrm{LOD}(\beta) = \frac{\ell\left(\frac{e^\beta}{1+e^\beta}\right) - \ell(0.5)}{\log(10)}$$

$$\hat{\beta} = -2.772409$$

$$\mathrm{LOD}(\hat{\beta}) = 3.164775$$

$$\frac{\partial \mathrm{LOD}}{\partial \beta}(0.0) = -2.38862$$

$$\frac{\partial^2 \mathrm{LOD}}{\partial \beta^2}(\hat{\beta}) = -0.4104178$$

Black dots computed with Mendel 16.0 (Reference software)

Results : Confidence intervales and tests

$$\ell(\beta) = \log L\left(\frac{e^\beta}{1+e^\beta}\right) \quad O(\beta) = -\ell''(\beta) \quad I(\beta) = \mathbb{E}[O(\beta)|\beta]$$

Likelihood ratio test statistic:

$$\mathrm{LR} = 2(\ell(\hat\beta) - \ell(\beta_0)) \underset{H_0}{\sim} \chi^2(\mathrm{df}=1)$$

Wald's test statistic:

$$W = \cancel{I(\beta_0)(\hat\beta - \beta_0)^2} = O(\hat\beta)(\hat\beta - \beta_0)^2 \underset{H_0}{\sim} \chi^2(\mathrm{df}=1)$$

Score test statistic:

$$S = \cancel{\frac{\ell'(\beta_0)^2}{I(\beta_0)}} = \frac{\ell'(\beta_0)^2}{O(\hat\beta)} \underset{H_0}{\sim} \chi^2(\mathrm{df}=1)$$

	$\hat\theta$ (95% IC)	LR	W	S
KUS ($n = 22$)	0.059 (0.008, 0.320)	14.574	7.264	32.010
ALL ($n = 93$)	0.193 (0.106, 0.326)	17.012	15.821	12.900
KUS ($n = 22$)		$1.3 \cdot 10^{-4}$	$7.0 \cdot 10^{-3}$	$1.5 \cdot 10^{-8}$
ALL ($n = 93$)		$3.7 \cdot 10^{-5}$	$7.0 \cdot 10^{-5}$	$3.3 \cdot 10^{-4}$

Summary and perspectives

Summary:

- Sum-product with polynomials and “Leibniz product”
- univariate: derivatives up to order d in $\mathcal{O}(d^2 \times C)$
- multivariate $\theta \in \mathbb{R}^p$: gradient and Hessian in $\mathcal{O}(p^2 \times C)$
- confidence intervals, theoretical distributions under H0

Perspectives:

- joint estimation of allele frequencies and θ
- extensive simulations under H0 and H1
- other genetic models: segregation, TDT, IBD, etc.

Acknowledgement: This work was funded by the *Ligue Nationale Contre le Cancer*.