COIN

Introduction to pseudolikelihood and marginal pseudolikelihood inference

Jukka Corander
Department of Mathematics and statistics
University of Helsinki, Finland

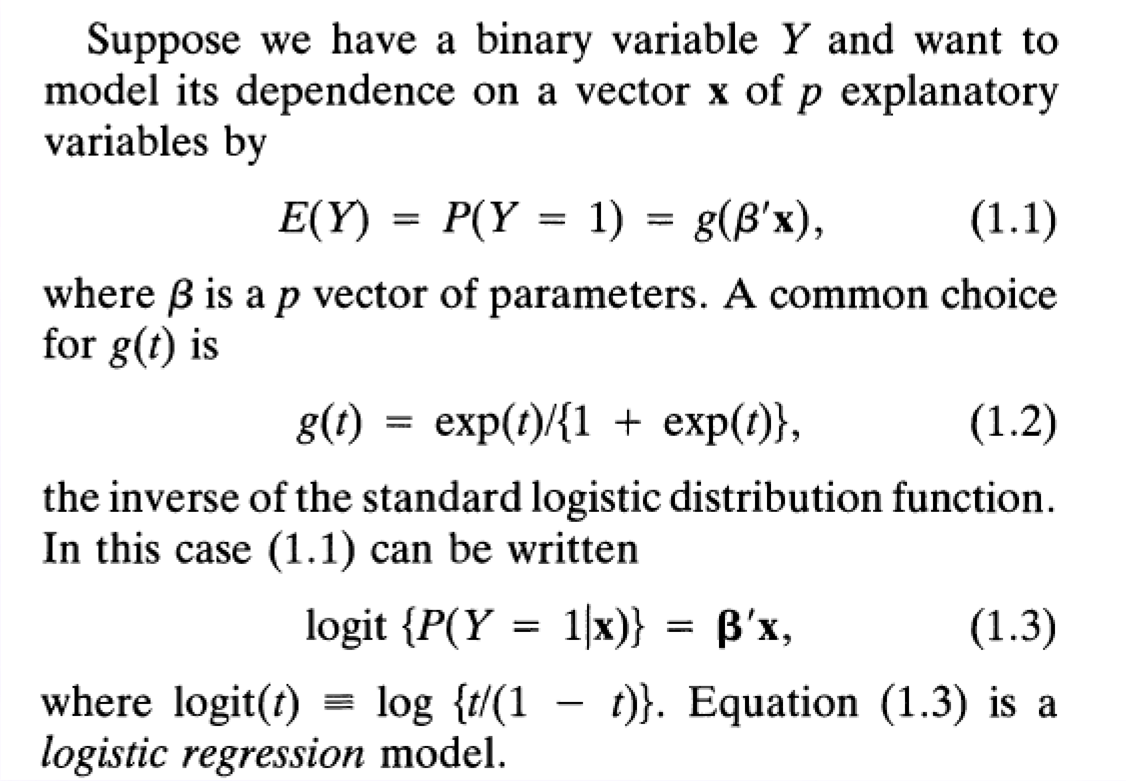
Aalto University

UNIVERSITY OF HELSINKI

What pseudolikelihood?

- Pseudolikelihood is an approximate inference technique originally introduced by Julian Besag in 1972
- Replaces tricky likelihood function by a product over suitably chosen model components
- Pseudolikelihood allows often use of logistic regression for parameter estimation
- Pseudolikelihood has recently experienced a strong revival due to large-scale modeling needs in computational physics and computational biology

Primer on logistic regression by David Strauss



Suppose we have a binary variable Y and want to model its dependence on a vector $\mathbf{x}$ of p explanatory variables by

$$E(Y) = P(Y = 1) = g(\beta'\mathbf{x}), \tag{1.1}$$

where β is a p vector of parameters. A common choice for $g(t)$ is

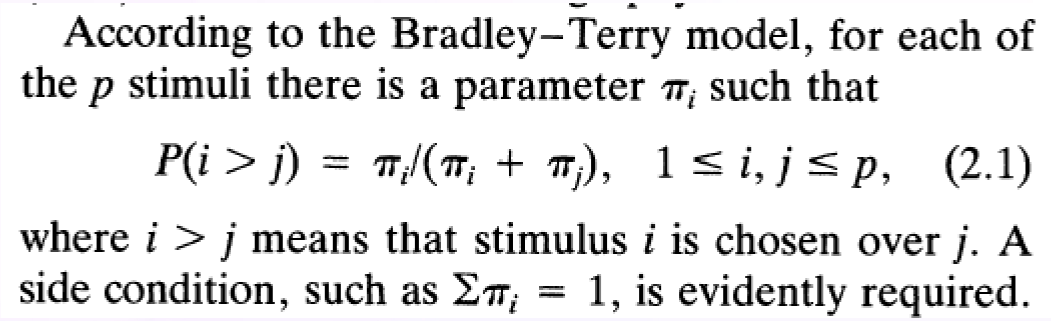
$$g(t) = \exp(t)/\{1 + \exp(t)\}, \tag{1.2}$$

the inverse of the standard logistic distribution function. In this case (1.1) can be written

$$\operatorname{logit}\{P(Y = 1|\mathbf{x})\} = \boldsymbol{\beta}'\mathbf{x}, \tag{1.3}$$

where $\operatorname{logit}(t) \equiv \log\{t/(1 - t)\}$. Equation (1.3) is a *logistic regression* model.

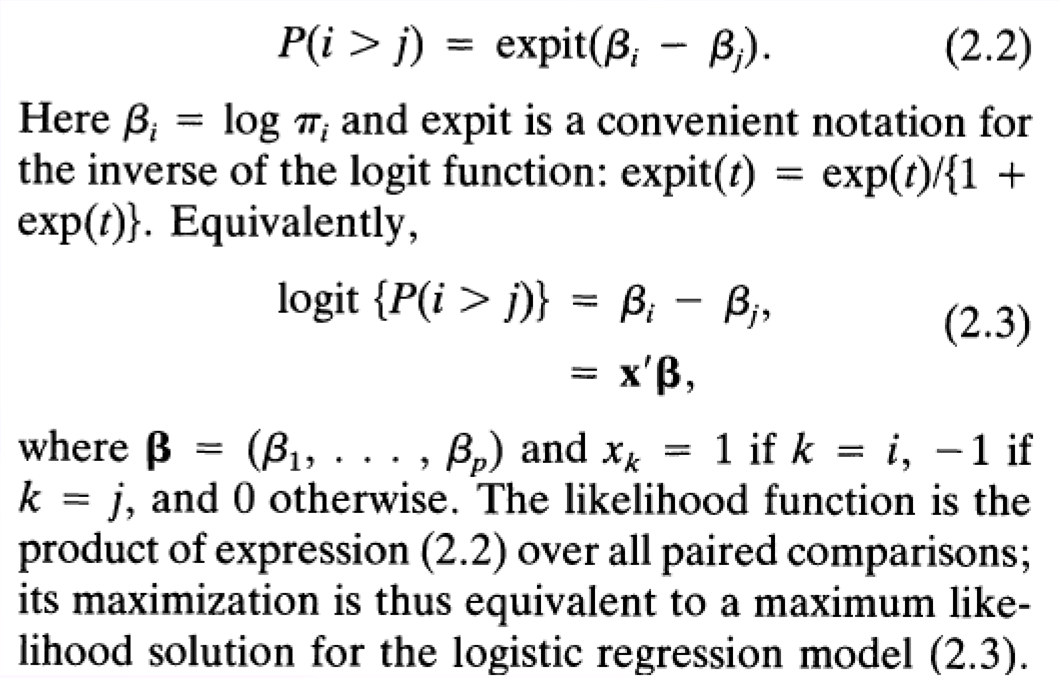
Pseudolikelihood with logistic regression for psychological Bradley-Terry model



According to the Bradley–Terry model, for each of the p stimuli there is a parameter π_i such that

$$P(i > j) = \pi_i/(\pi_i + \pi_j), \quad 1 \leq i, j \leq p, \qquad (2.1)$$

where $i > j$ means that stimulus i is chosen over j. A side condition, such as $\Sigma\pi_i = 1$, is evidently required.



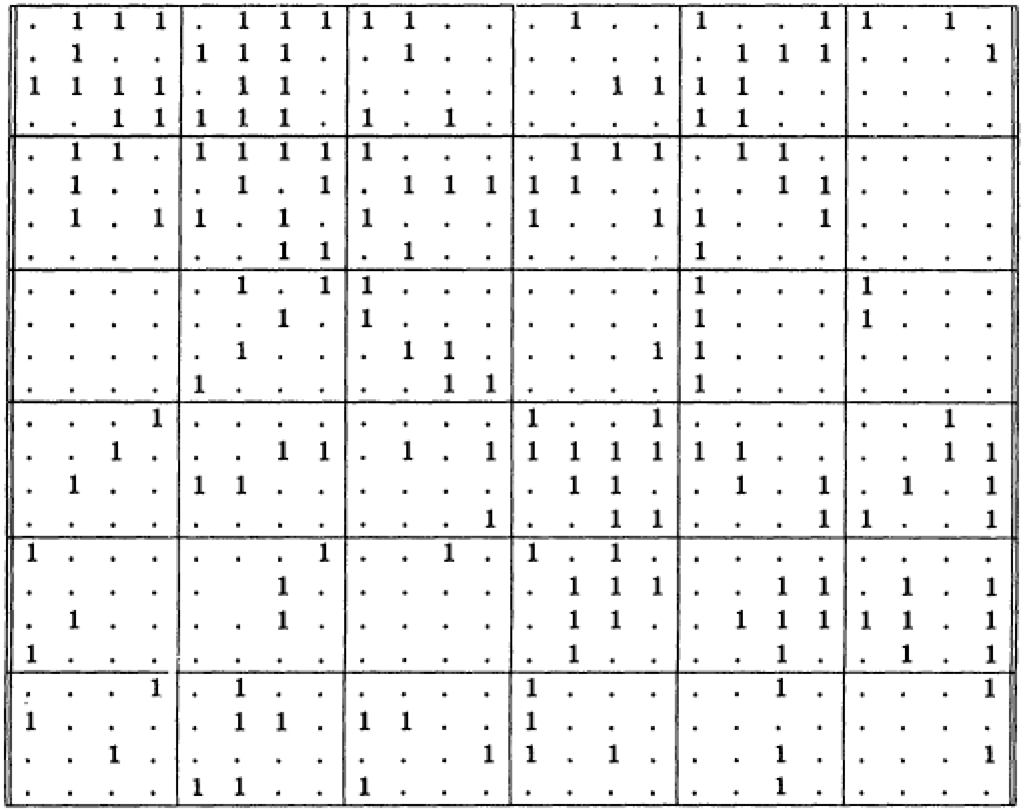
$$P(i > j) = \operatorname{expit}(\beta_i - \beta_j). \qquad (2.2)$$

Here $\beta_i = \log \pi_i$ and expit is a convenient notation for the inverse of the logit function: $\operatorname{expit}(t) = \exp(t)/\{1 + \exp(t)\}$. Equivalently,

$$\begin{aligned} \operatorname{logit}\{P(i > j)\} &= \beta_i - \beta_j, \\ &= \mathbf{x}'\boldsymbol{\beta}, \end{aligned} \qquad (2.3)$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ and $x_k = 1$ if $k = i$, -1 if $k = j$, and 0 otherwise. The likelihood function is the product of expression (2.2) over all paired comparisons; its maximization is thus equivalent to a maximum likelihood solution for the logistic regression model (2.3).

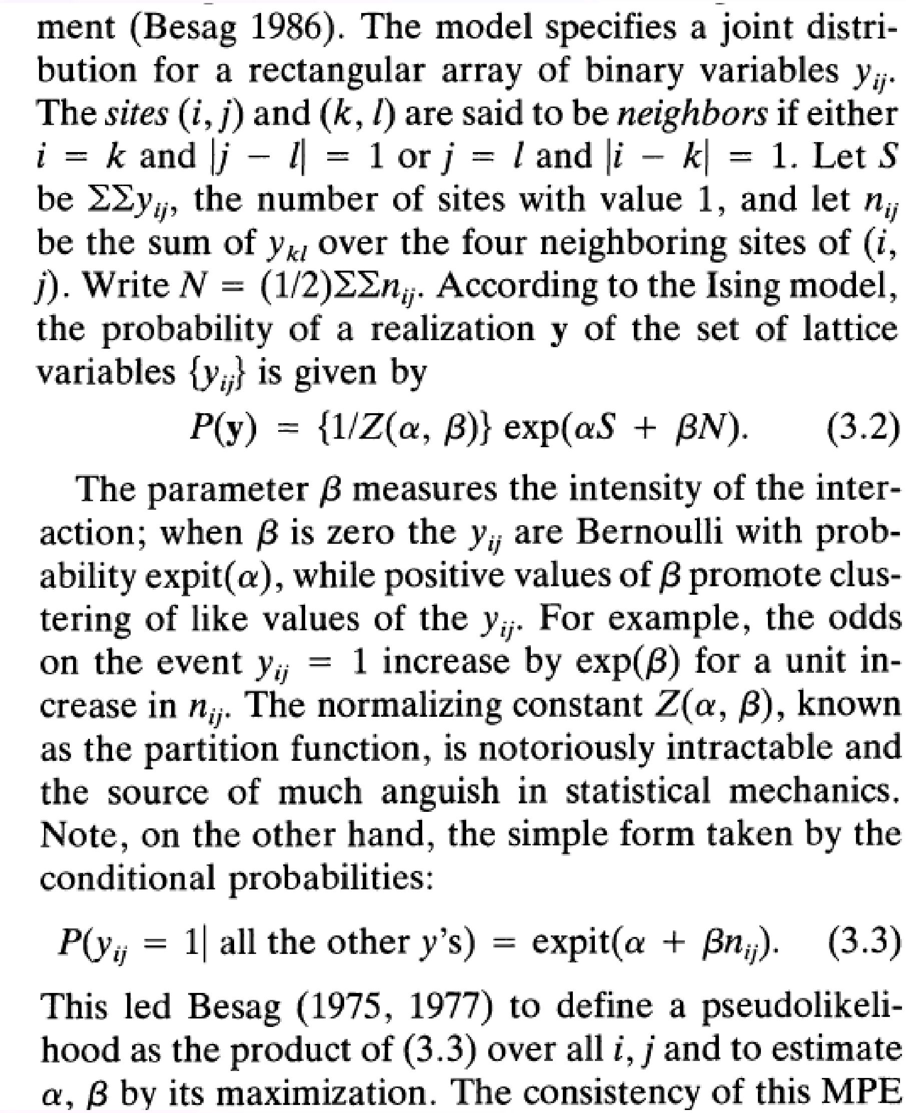
Pseudolikelihood for spatial dependence models



.	1	1	1	.	1	1	1	1	1	.	.	.	1	.	.	1	.	.	1	1	.	1	.
.	1	.	.	1	1	1	.	.	1	.	.	.	.	.	.	.	1	1	1	.	.	.	1
1	1	1	1	.	1	1	.	.	.	.	.	.	.	1	1	1	1	.	.	.	.	.	.
.	.	1	1	1	1	1	.	1	.	1	.	.	.	.	.	1	1	.	.	.	.	.	.
.	1	1	.	1	1	1	1	1	.	.	.	.	1	1	1	.	1	1	.	.	.	.	.
.	1	.	.	.	1	.	1	.	1	1	1	1	1	.	.	.	.	1	1	.	.	.	.
.	1	.	1	1	.	1	.	1	.	.	.	1	.	.	1	1	.	.	1	.	.	.	.
.	.	.	.	.	.	1	1	.	1	.	.	.	.	.	.	1	.	.	.	.	.	.	.
.	.	.	.	.	1	.	1	1	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.
.	.	.	.	.	.	1	.	1	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.
.	.	.	.	.	1	.	.	.	1	1	.	.	.	.	1	1	.	.	.	.	.	.	.
.	.	.	.	1	.	.	.	.	.	1	1	.	.	.	.	1	.	.	.	.	.	.	.
.	.	.	1	.	.	.	.	.	.	.	.	1	.	.	1	.	.	.	.	.	.	1	.
.	.	1	.	.	.	1	1	.	1	.	1	1	1	1	1	1	1	.	.	.	.	1	1
.	1	.	.	1	1	.	.	.	.	.	.	.	1	1	.	.	1	.	1	.	1	.	1
.	.	.	.	.	.	.	.	.	.	.	1	.	.	1	1	.	.	.	1	1	.	.	1
1	.	.	.	.	.	.	1	.	.	1	.	1	.	1	.	.	.	.	.	.	.	.	.
.	.	.	.	.		1	.	.	.	.	.	.	1	1	1	.	.	1	1	.	1	.	1
.	1	.	.	.	.	1	.	.	.	.	.	.	1	1	.	.	1	1	1	1	1	.	1
1	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	1	.	.	1	.	1
.	.	.	1	.	1	.	.	.	.	.	.	1	.	.	.	.	.	1	.	.	.	.	1
1	.	.	.	.	1	1	.	1	1	.	.	1	.	.	.	.	.	.	.	.	.	.	.
.	.	1	.	.	.	.	.	.	.	.	1	1	.	1	.	.	.	1	.	.	.	.	1
.	.	.	.	1	1	.	.	1	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.

Figure 1. 24 × 24 Grid of Presence/Absence of the Plant Carex Arenaria. (From Bartlett 1971.)

Pseudolikelihood for Ising models



ment (Besag 1986). The model specifies a joint distribution for a rectangular array of binary variables y_{ij}. The *sites* (i, j) and (k, l) are said to be *neighbors* if either $i = k$ and $|j - l| = 1$ or $j = l$ and $|i - k| = 1$. Let S be $\Sigma\Sigma y_{ij}$, the number of sites with value 1, and let n_{ij} be the sum of y_{kl} over the four neighboring sites of (i, j). Write $N = (1/2)\Sigma\Sigma n_{ij}$. According to the Ising model, the probability of a realization $\mathbf{y}$ of the set of lattice variables $\{y_{ij}\}$ is given by

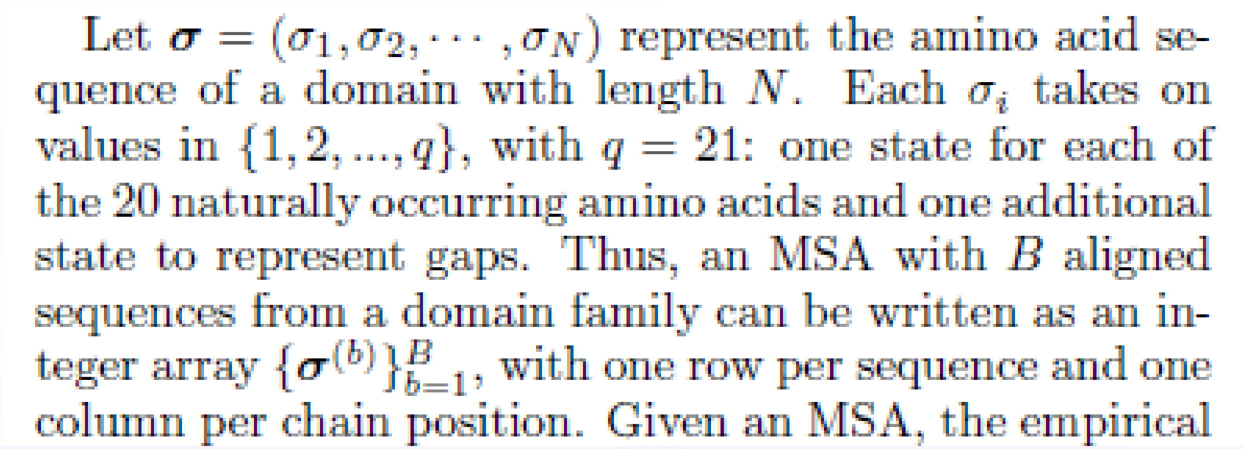
$$P(\mathbf{y}) = \{1/Z(\alpha, \beta)\} \exp(\alpha S + \beta N). \qquad (3.2)$$

The parameter β measures the intensity of the interaction; when β is zero the y_{ij} are Bernoulli with probability expit(α), while positive values of β promote clustering of like values of the y_{ij}. For example, the odds on the event $y_{ij} = 1$ increase by $\exp(\beta)$ for a unit increase in n_{ij}. The normalizing constant $Z(\alpha, \beta)$, known as the partition function, is notoriously intractable and the source of much anguish in statistical mechanics. Note, on the other hand, the simple form taken by the conditional probabilities:

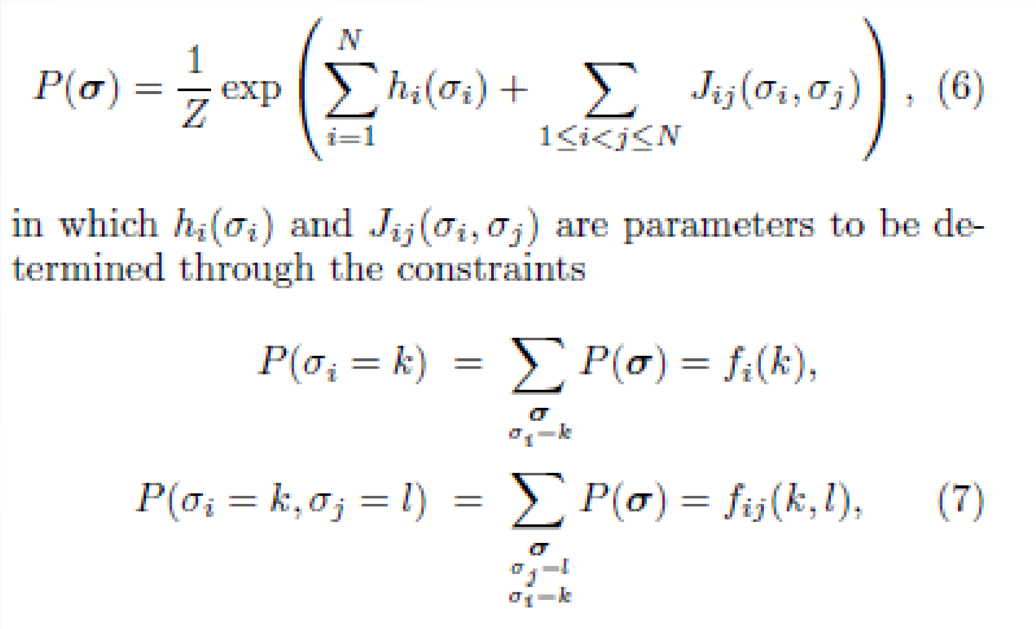
$$P(y_{ij} = 1| \text{ all the other } y\text{'s}) = \operatorname{expit}(\alpha + \beta n_{ij}). \qquad (3.3)$$

This led Besag (1975, 1977) to define a pseudolikelihood as the product of (3.3) over all i, j and to estimate α, β by its maximization. The consistency of this MPE

Potts model for MSA
Ekeberg et al. Phys Rev Lett, 2013



Let $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \cdots, \sigma_N)$ represent the amino acid sequence of a domain with length N. Each σ_i takes on values in $\{1, 2, ..., q\}$, with $q = 21$: one state for each of the 20 naturally occurring amino acids and one additional state to represent gaps. Thus, an MSA with B aligned sequences from a domain family can be written as an integer array $\{\boldsymbol{\sigma}^{(b)}\}_{b=1}^{B}$, with one row per sequence and one column per chain position. Given an MSA, the empirical



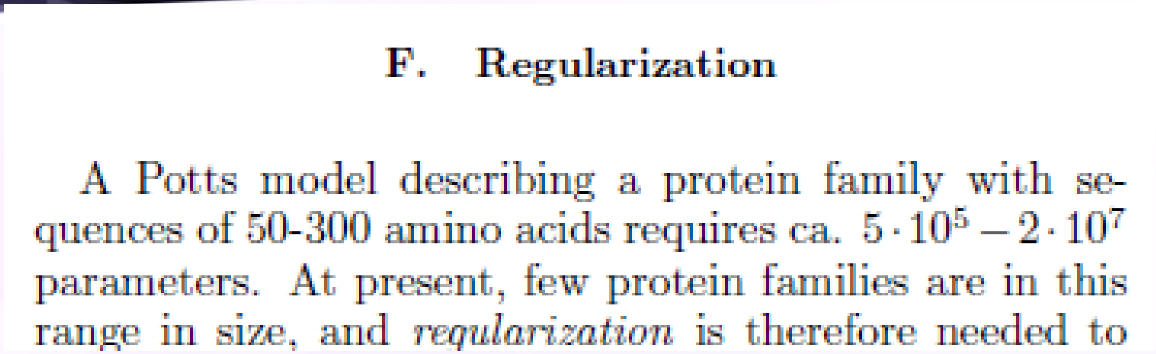
$$P(\boldsymbol{\sigma}) = \frac{1}{Z} \exp\left(\sum_{i=1}^{N} h_i(\sigma_i) + \sum_{1 \le i < j \le N} J_{ij}(\sigma_i, \sigma_j) \right), \quad (6)$$

in which $h_i(\sigma_i)$ and $J_{ij}(\sigma_i, \sigma_j)$ are parameters to be determined through the constraints

$$P(\sigma_i = k) = \sum_{\substack{\boldsymbol{\sigma} \\ \sigma_i = k}} P(\boldsymbol{\sigma}) = f_i(k),$$

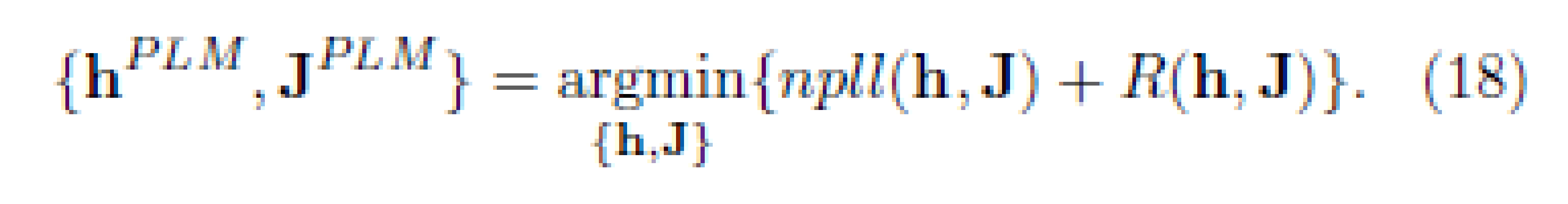
$$P(\sigma_i = k, \sigma_j = l) = \sum_{\substack{\boldsymbol{\sigma} \\ \sigma_j = l \\ \sigma_i = k}} P(\boldsymbol{\sigma}) = f_{ij}(k, l), \quad (7)$$

Pseudolikelihood for Potts model for MSA
Ekeberg et al. Phys Rev Lett, 2013

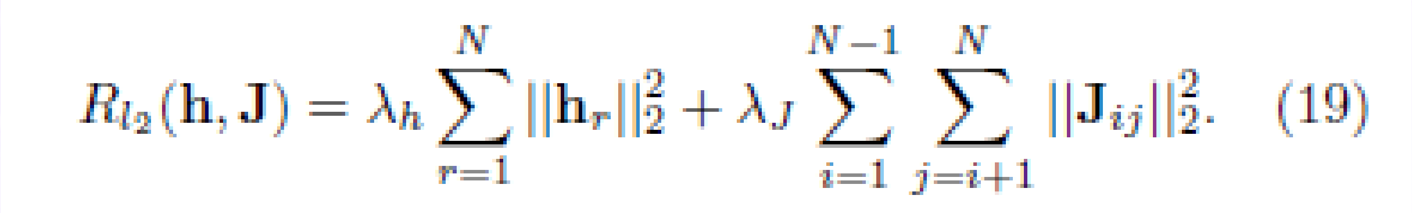


F. Regularization

A Potts model describing a protein family with sequences of 50-300 amino acids requires ca. $5 \cdot 10^5 - 2 \cdot 10^7$ parameters. At present, few protein families are in this range in size, and *regularization* is therefore needed to



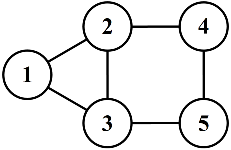
$$\{\mathbf{h}^{PLM}, \mathbf{J}^{PLM}\} = \underset{\{\mathbf{h},\mathbf{J}\}}{\operatorname{argmin}} \{npll(\mathbf{h},\mathbf{J}) + R(\mathbf{h},\mathbf{J})\}. \quad (18)$$



$$R_{l_2}(\mathbf{h},\mathbf{J}) = \lambda_h \sum_{r=1}^{N} ||\mathbf{h}_r||_2^2 + \lambda_J \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} ||\mathbf{J}_{ij}||_2^2. \quad (19)$$

L1 regularization not good for these models, that is why L2 is used here!

Markov network (MN)

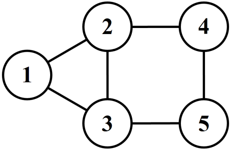


- A MN is a probabilistic graphical model over a set of variables $(X_1,\ldots,X_d)$. (we only consider the discrete case)
- The dependence structure over the variables is represented by an undirected graph $G=(V,E)$.
- The nodes in the graph, $V=\{1,\ldots,d\}$, represent the variables and the edges, $E\subseteq\{V\times V\}$, represent direct dependencies among the variables.
- Absence of edges represents statements of conditional independence, in particular

$$X_i \perp X_{V\setminus\{MB(i)\cup i\}} \mid X_{MB(i)}$$

where $MB(i)=\{j\in V:\{i,j\}\in E\}$ is the Markov blanket of node i.

Markov network (MN)



- A MN is a pair (G, θ_G) where θ_G is a parameterization of a joint distribution P_G over $(X_1, \ldots, X_d)$
- P_G must satisfy the restrictions imposed by G, in particular:

$$X_i \perp X_{V \setminus \{MB(i) \cup i\}} \mid X_{MB(i)} \Leftrightarrow P(X_i \mid X_{V \setminus i}) = P(X_i \mid X_{MB(i)})$$

- We assume that P_G is positive.
- The joint distribution factorizes according to its maximal cliques

$$P_G(X_V) = \frac{1}{Z} \prod_{C \in \mathcal{C}(G)} \phi_C(X_C)$$

where $\phi_C : \mathcal{X}_C \to \mathbb{R}_+$ is a clique factor and $Z = \sum_{x_V \in \mathcal{X}_V} P_G(x_V)$ is the partition function.

Structure learning

- We assume we have a data set $\mathbf{X}$ containing n complete i.i.d. joint observations $\mathbf{x}_k = (x_{k,1}, \ldots, x_{k,d})$ generated from θ_{G^*}.
- The aim is to discover the graph structure G^* from the set of all possible graph structures $\mathcal{G}$.
- Structure learning is basically model class learning.
- Reasons for structure learning:
  - Step in model learning - Learn distribution given the graph.
  - Knowledge discovery - The structure is a goal in itself.
- Structure learning methods can roughly be divided into two categories:
  - Constraint-based - Independence tests.
  - Score-based - Optimization problem.

The Bayesian approach

- We choose the graph with the highest posterior probability given the data:

$$p(G \mid \mathsf{X}) = \frac{p(\mathsf{X} \mid G) \cdot p(G)}{p(\mathsf{X})}$$

- Since $p(\mathsf{X})$ is a normalizing constant, the problem can be formulated as

$$\arg\max_{G \in \mathcal{G}} p(\mathsf{X} \mid G) \cdot p(G).$$

- The key term of the Bayesian score is the marginal likelihood which is evaluated according to

$$p(\mathsf{X} \mid G) = \int_{\theta \in \Theta_G} p(\mathsf{X} \mid \theta, G) \cdot f(\theta \mid G) d\theta.$$

- The marginal likelihood is hard to evaluate for MNs.

The pseudo-likelihood function

- The pseudo-likelihood (Besag, 1975) is given by

$$\hat{p}(\mathrm{X} \mid \theta) = \prod_{j=1}^{d} p(\mathrm{X}_j \mid \mathrm{X}_{V\setminus j}, \theta).$$

- Given a graph, the local Markov property allows us to simplify the pseudo-likelihood as

$$\hat{p}(\mathrm{X} \mid \theta, G) = \prod_{j=1}^{d} p(\mathrm{X}_j \mid \mathrm{X}_{MB(j)}, \theta) = \prod_{j=1}^{d} \prod_{l=1}^{q_j} \prod_{i=1}^{r_j} \theta_{ijl}^{n_{ijl}}.$$

- The marginal pseudo-likelihood (MPL) is evaluated according to

$$\hat{p}(\mathrm{X} \mid G) = \int_{\theta \in \Theta_G} \hat{p}(\mathrm{X} \mid \theta, G) \cdot f(\theta \mid G) d\theta.$$

Marginal pseudo-likelihood

- We assume global and local independence among the parameters (see parameter independence assumption for Bayesian networks, Heckerman et al., 1995).
- This allows us to factorize the parameter prior distribution and solve the MPL analytically:

$$\hat{p}(\mathrm{X} \mid G) = \prod_{j=1}^{d} \prod_{l=1}^{q_j} \frac{\Gamma(\alpha_{jl})}{\Gamma(n_{jl} + \alpha_{jl})} \prod_{i=1}^{r_j} \frac{\Gamma(n_{ijl} + \alpha_{ijl})}{\Gamma(\alpha_{ijl})}$$

- The MPL can in fact be considered the marginal likelihood for a bi-directional dependency network (Heckerman et al., 2001).

Number of possible graphs, $|\mathcal{G}|$

d	$\lvert\mathcal{G}\rvert = 2^{\binom{d}{2}}$
2	2
4	64
8	268435456
16	$1.32\ldots \cdot 10^{36}$
32	$2.04\ldots \cdot 10^{149}$
$\vdots$	$\vdots$

The direct approach

$$\arg\max_{G \in \mathcal{G}} \hat{p}(\mathbf{X} \mid G) \; (\; \cdot p(G) \;)$$

- We assume a uniform prior $p(G) = 1/|\mathcal{G}|$.
- The variable-wise factorization

$$\hat{p}(\mathbf{X} \mid G) = \prod_{j=1}^{d} p(\mathbf{X}_j \mid \mathbf{X}_{MB(j)})$$

makes the MPL a viable candidate for search algorithms based on local changes.

The direct approach

$$\arg\max_{G \in \mathcal{G}} \hat{p}(\mathrm{X} \mid G)$$

- Two graphs G_1 and G_2 are compared by Bayes pseudo-factor

$$K(G_1; G_2) = \frac{\hat{p}(\mathrm{X} \mid G_1)}{\hat{p}(\mathrm{X} \mid G_2)}.$$

- If we assume a single edge difference $\{i, j\}$ between G_1 and G_2, then

$$K(G_1; G_2) = \frac{p(\mathrm{X}_i \mid \mathrm{X}_{MB_1(i)})}{p(\mathrm{X}_i \mid \mathrm{X}_{MB_2(i)})} \cdot \frac{p(\mathrm{X}_j \mid \mathrm{X}_{MB_1(j)})}{p(\mathrm{X}_j \mid \mathrm{X}_{MB_2(j)})}.$$

The divide-and-conquer approach

- By denoting $MB(G) = \{MB(1), \ldots, MB(d)\}$, we reformulate the original problem:

$$\arg\max_{G \in \mathcal{G}} \hat{p}(\mathsf{X} \mid G)$$

$$\Leftrightarrow$$

$$\arg\max_{MB(G) \in \times_{j \in V} \mathcal{P}(V \setminus j)} \prod_{j=1}^{d} p(\mathsf{X}_j \mid \mathsf{X}_{MB(j)})$$

$$\text{subject to} \quad i \in MB(j) \Rightarrow j \in MB(i) \text{ for all } i, j \in V$$

The divide-and-conquer approach

- Relaxed version of the reformulated problem:

$$\underset{MB(G)\in\times_{j\in V}\mathcal{P}(V\backslash j)}{\arg\max}\prod_{j=1}^{d} p(\mathsf{X}_j \mid \mathsf{X}_{MB(j)})$$

- We now have *d* independent subproblems:

$$\underset{MB(j)\subseteq V\backslash j}{\arg\max}\, p(\mathsf{X}_j \mid \mathsf{X}_{MB(j)}) \quad \text{for } j = 1,\ldots,d.$$

- Independent problems - Parallel solving!
- However, inconsistent solutions...

Forming a MN structure from inconsistent Markov blankets

- Solutions to the relaxed problem are in general inconsistent in the sense that $i \in MB(j)$ but $j \notin MB(i)$.
- Post-process the solution to satisfy the structure of a MN.
- Simple approaches:

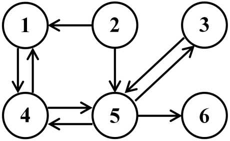
$$E_{AND} = \{\{i,j\} \in \{V \times V\} : i \in MB(j) \textbf{ AND } j \in MB(i)\}$$
$$E_{OR} = \{\{i,j\} \in \{V \times V\} : i \in MB(j) \textbf{ OR } j \in MB(i)\}$$

- A more elaborate approach:

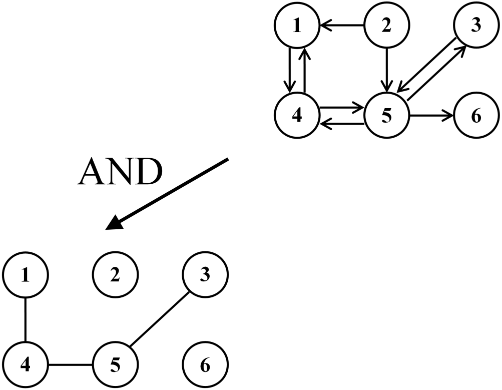
$$E_{HC} = \underset{E \subseteq E_{OR}}{\arg\max}\, \hat{p}(\mathbf{X} \mid G)$$

i.e. we solve the original problem w.r.t the reduced model space $\{G \in \mathcal{G} : E \subseteq E_{OR}\} \subseteq \mathcal{G}$.

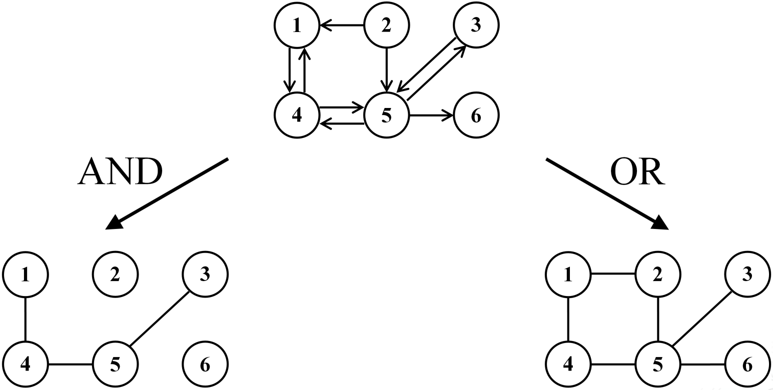
Forming a MN structure from inconsistent Markov blankets



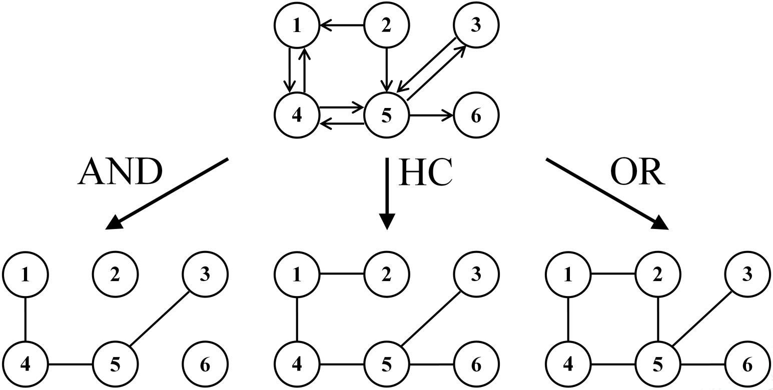
Forming a MN structure from inconsistent Markov blankets



Forming a MN structure from inconsistent Markov blankets



Forming a MN structure from inconsistent Markov blankets

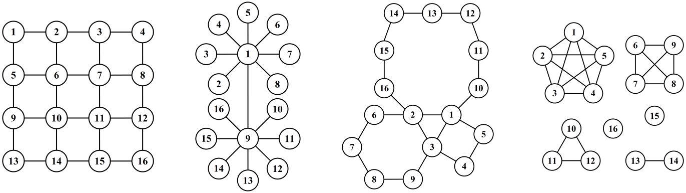


Comparative study of proposed methods

- We compare MPL-AND, -OR and -HC.
- All methods use the same initial Markov blanket discovery phase.
- We generate data from synthetic models and compare the identified structures to the true one.
- The quality of the identified structures are assessed by the Hamming distance (# False positives + # False negatives).
- All results were averaged over 10 distributions and 10 samples per distribution $\Rightarrow$ 100 samples.

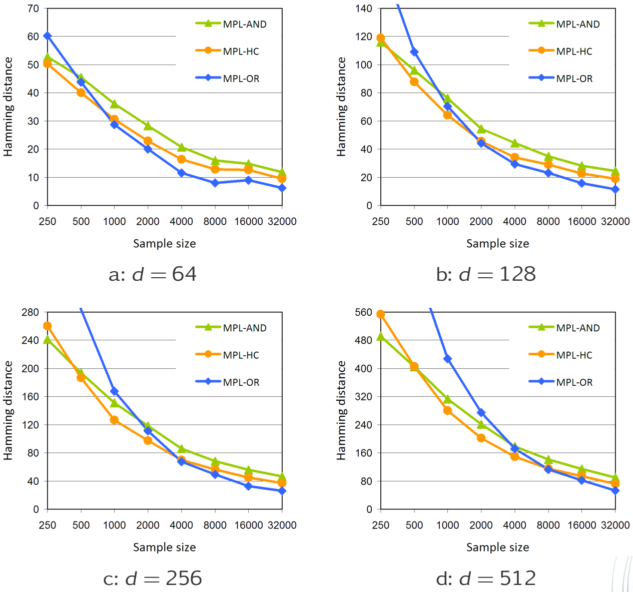
Generating model

- Binary variables.
- Structure - formed by combining disconnected components:



- Distribution - for each $C \in \mathcal{C}$ and $x_C \in \mathcal{X}_C$: $\phi(x_C)$ is drawn from $\mathcal{U}(0,1)$.

Marginal pseudo-likelihood: Example



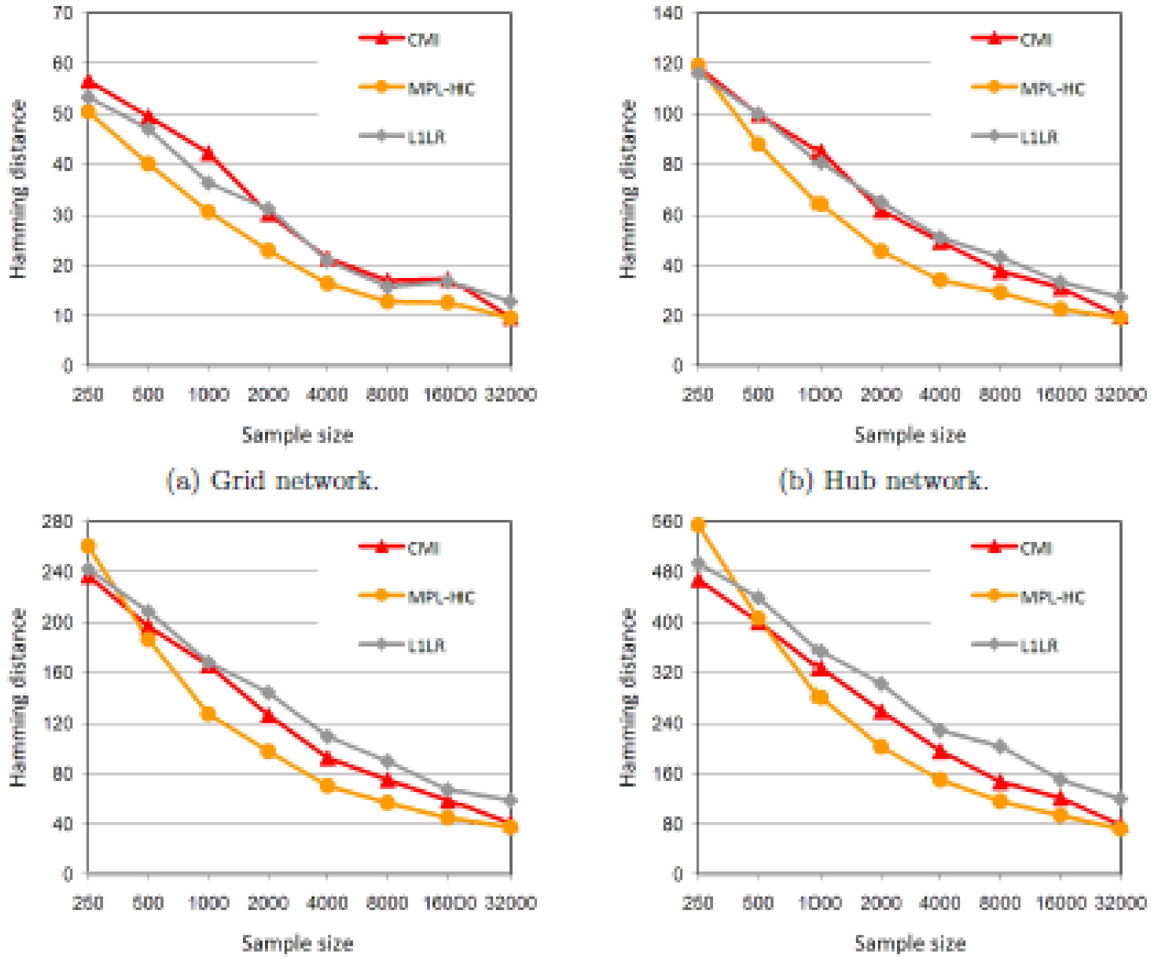
a: $d = 64$

b: $d = 128$

c: $d = 256$

d: $d = 512$

MPL rocks against most popular recent pseudolikelihood methods!



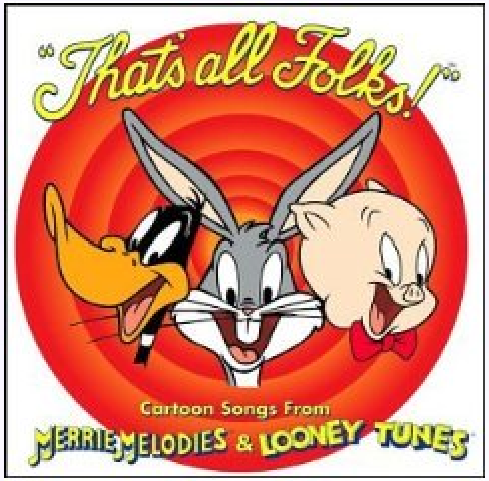
(a) Grid network.

(b) Hub network.

(c) Loop network.

(d) Clique network.

Hope you had some good time!



Aalto University

UNIVERSITY OF HELSINKI