* structure Learning Recap.

Approach 1. ML for DAG

$$\text{SCORE}(G) = \sum_{i=1}^{n} I_{\hat{p}}(X_i ; X_{\pi_i})$$

tractable for $G \in$ TREE, using max-weight spanning tree [Chow-Liu algorithm]

Approach 2. ML for Ising Model

$$\min_{\theta \in \mathbb{R}^{n \times n}} \quad \log Z_G(\theta) - \langle \hat{\mu}, \theta \rangle + \lambda \|\theta\|_{L_1}$$

intractable for general graph even if it is sparse.

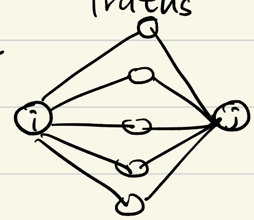
Approach 3. Local Independence Tests.

Enumerate all $|S| = K$ neighborhood for each node i and select one with best "Independence" score.

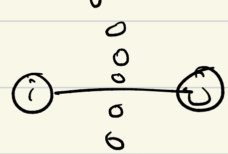
$O(n^{K+1})$

$O(n^2) \longrightarrow$ alternatively, one could check "Correlation" for each pair

Truths



Algorithm output.



even if i, j are not directly connected

but if there are many paths connecting i to j,

* None of the algorithms above work in large scale practical problems.

Q. Can we design an algorithm that is efficient & works well in practice?

Consider Binary Random variables $\mathcal{X} = \{0, 1\}$, and an undirected G.M

$$P(x_1, x_2, \cdots, x_n) = \frac{1}{Z(\theta)} \cdot \exp\left\{ \sum_{i=1}^{n} \theta_{ii} x_i + \sum_{(i,j)\in E} \theta_{ij} x_i x_j \right\}$$

claim: A conditional distribution of x_1 given the rest is

$$\frac{P(x_1 = 1 \mid x_2, \cdots, x_n)}{P(x_1 = 0 \mid x_2, \cdots, x_n)} = \exp\left\{ \theta_{11} + \sum_{j \in \partial_1} \theta_{1j} \cdot x_j \right\}$$

this is a Logistic Regression problem, which can be solved efficiently.

$$P(x_1 = 1 \mid x_2, \cdots, x_n) = \frac{e^{\theta_{11} + \sum_{j\in\partial_1} \theta_{1j} x_j}}{1 + e^{\theta_{11} + \sum_{j\in\partial_1} \theta_{1j} x_j}}$$

$$= \text{sigmoid}\Big(\underbrace{\theta_{11} + \sum_{j\in\partial_1} \theta_{1j} x_j}_{\text{linear in } \theta_{1\cdot}}\Big)$$

sigmoid = Logistic function

$\theta_{1\cdot}$ = first row of $\theta \in \mathbb{R}^{n\times n}$

proof:

$$P(x_1 = 1 \mid x_2^n) = \frac{P(x_1 = 1 \wedge x_2^n)}{P(x_2^n)}$$

$$= \frac{\cancel{\frac{1}{Z(\theta)}} \cdot e^{\left\{\theta_{11} + \sum_{j\in\partial_1} \theta_{1j} \cdot x_j + \sum_{\substack{i,j>1 \\ (i,j)\in E}} \theta_{ij} x_i x_j\right\}}}{\cancel{P(x_2^n)}}$$

$$P(x_1 = 0 \mid x_2^n) = \frac{\cancel{\frac{1}{Z(\theta)}} \cdot e^{\left\{\sum_{\substack{i,j>1 \\ (i,j)\in E}} \theta_{ij} x_i x_j\right\}}}{\cancel{P(x_2^n)}}$$

* Review of logistic Regression.

We are given labelled samples of features $Z^{(l)} = (Z_1^{(l)}, \cdots, Z_L^{(l)}) \in \mathbb{R}^L$ and labels $Y^{(l)} \in \{0, 1\}$

Data : $\{(Z^{(l)}, Y^{(l)})\}_{l=1}^{N}$

We want to find a model parameter $W \in \mathbb{R}^L$ such that

$$P(Y=1 \mid Z) = \frac{\exp(\sum_{k=1}^{L} W_k Z_k)}{1 + \exp(\sum_{k=1}^{L} W_k Z_k)}$$

$$P(Y=0 \mid Z) = \frac{1}{1 + \exp(\sum_{k=1}^{L} W_k Z_k)}$$

We will find the maximum likelihood estimator.

log-likelihood $\mathcal{L}(\{(Z^{(l)}, Y^{(l)})\}_{l=1}^{N}, W) = \frac{1}{N} \sum_{l=1}^{N} \log P_W(Y^{(l)} \mid Z^{(l)})$

$$= \frac{1}{N} \sum_{l=1}^{N} \left\{ Y^{(l)} \log P(Y=1 \mid Z^{(l)}) + (1 - Y^{(l)}) \log P(Y=0 \mid Z^{(l)}) \right\}$$

$$= \frac{1}{N} \sum_{l=1}^{N} \left\{ Y^{(l)} \left(\sum_{k=1}^{L} W_k Z_k^{(l)} \right) - \log \left(1 + \exp\left(\sum_{k=1}^{L} W_k Z_k^{(l)} \right) \right) \right\}$$

this is a concave function in $W \in \mathbb{R}^L$, which is maximized using Gradient Descent. /Ascent.

Initialize $W^{(0)} = \mathbb{1}$.

Repeat $W^{(t+1)} = W^{(t)} + \frac{1}{t} \cdot \nabla_W \mathcal{L}(\{Y^{(l)}, Z^{(l)}\}_{l=1}^{N}, W^{(t)})$

$$\nabla_W \mathcal{L}(\{Y^{(l)}, Z^{(l)}\}_{l=1}^{N}, W^{(t)}) = \frac{1}{N} \sum_{l=1}^{N} Y^{(l)} \cdot Z^{(l)} - \frac{\exp(\sum_{k=1}^{L} W_k^{(t)} Z_k^{(l)})}{1 + \exp(\sum_{k=1}^{L} W_k^{(t)} Z_k^{(l)})} \cdot Z^{(l)}$$

$$= W^{(t)} + \frac{1}{t} \cdot \frac{1}{N} \sum_{l=1}^{N} Z^{(l)} \cdot \left(Y^{(l)} - \mathbb{P}_{W^{(t)}}(Y=1 \mid Z^{(l)}) \right)$$

* Logistic regression for neighborhood selection.

Structural Learning	Logistic regression
$P(X_1=1 \mid X_2,\cdots,X_n) = \dfrac{e^{\theta_{11}+\sum_{j\in\partial 1}\theta_{1j}X_j}}{1+e^{\theta_{11}+\sum_{j\in\partial 1}\theta_{1j}X_j}}$	$P(Y=1 \mid Z) = \dfrac{\exp(\sum_{k=1}^{L} w_k z_k)}{1+\exp(\sum_{k=1}^{L} w_k z_k)}$
$P(X_1=0 \mid X_2 \cdots X_n) = \dfrac{1}{1+\exp\{\theta_{11}+\sum_{j\in\partial 1}\theta_{1j}X_j\}}$	$P(Y=0 \mid Z) = \dfrac{1}{1+\exp(\sum_{k=1}^{L} w_k z_k)}$

for node 1, If we know G, then we can find θ_{1j}'s using Logistic Regression approaches

with features $\underbrace{(1, X_{\partial 1})}_{= Z}$ and label $\underbrace{X_1}_{= Y}$

As we do not know G, suppose the ground truths $|\theta_{ij}| \le 1$ and degree $\le K$.

This motivates the following formulation

$$\min_{\substack{\theta_{1\cdot} \\ = (\theta_{11},\theta_{12},\cdots,\theta_{1n})\in\mathbb{R}^n}} \; -\mathcal{L}(\{X_1^{(\ell)},(1,X_2^{(\ell)},\cdots,X_n^{(\ell)})\}_{\ell=1}^{N}, \theta_{1\cdot})$$

$$\text{s.t.} \quad \|\theta_{1\cdot}\|_{L1} \le K.$$

OR equivalently, we can solve

$$(*) \quad \min_{\theta_{1\cdot}} \; -\mathcal{L}(\{X_1^{(\ell)},(1,X_2^{(\ell)},\cdots,X_n^{(\ell)})\}_{\ell=1}^{N}, \theta_{1\cdot}) + \lambda\cdot\|\theta_{1\cdot}\|_{L1}.$$

Theorem [Klivans, Meka, 2017]

If max degree $\leq K$

$P(X_i = 1 \mid X_{-i}) \in [\delta, 1-\delta]$ for some $\delta > 0$

number of samples $N \geq c \cdot \log n \cdot \frac{1}{\varepsilon^2}$

Then (*) achives $\|\hat{\theta} - \theta^*\|_\infty \leq \varepsilon$

in run-time $O(n^2 \, \mathrm{polylog}(\frac{n}{\varepsilon}))$

if all nonzero $\theta^*_{ij} > 2\varepsilon$, then we can threshold at $\hat{\theta}_{ij} \gtrless \varepsilon$ to recover the structure exactly.

* This principle can be used for more general graphical models. For example, Gaussian Graphical Models.

$$P(X) = \frac{1}{Z} \exp\{-\frac{1}{2} x^T \theta x\}$$

w.l.o.g suppose $h=0$.

$\mathcal{X} = \mathbb{R}$

we first compute the conditional.

$$P(X_1 \mid X_2^n) = \frac{1}{Z_1} \cdot \exp\{-\frac{1}{2}\left(\theta_{11} x_1^2 + 2\overbrace{\left(\sum_{j \in \partial 1} \theta_{1j} \cdot X_j\right)}^{\beta} X_1\right)\}$$

$$= \frac{1}{\sqrt{2\pi/\theta_{11}}} \cdot \exp\left\{-\frac{1}{2} \frac{(X_1 + \beta/\theta_{11})^2}{1/\theta_{11}}\right\}$$

Applying maximum likelihood to estimate X_1 from X_2^n,

$$\min -\sum_{l=1}^N \log P(X_1^{(l)} \mid X_2^{(l)} \cdots X_n^{(l)}) = \frac{\theta_{11}}{2} \sum_{l=1}^N \left(X_1^{(l)} - \sum_{j \in \partial 1} \frac{\theta_{1j}}{-\theta_{11}} \cdot X_j^{(l)}\right)^2$$

$$\underbrace{-\frac{1}{2} \log \theta_{11}}_{\text{ensures we don't make } \theta_{11} \text{ to small.}}$$

If we only care about the graph structure, for now, we can re-parametrize and solve for each node.

$$\min \sum_{l=1}^{N} \left(X_i^{(l)} - \sum_{j=2}^{n} W_j \cdot X_j^{(l)} \right)^2 + \lambda \, \|w\|_{L1} \qquad [HW4]$$

and threshold the resulting w to recover the neighborhood.

* An alternative (and also very popular) way to learn the structure of a Gaussian Graphical Model is "Graph Lasso".

Step 1. Compute Covariance Matrix $S_{ij} = \frac{1}{N} \sum_{l=1}^{N} X_i^{(l)} X_j^{(l)}$

step 2 maximize likelihood for information matrix J.

$$J^* = \arg\max_{J} \Big\{ \underbrace{\underbrace{\log|J|}_{\text{Determinant}} - \underbrace{\mathrm{Tr}(S \cdot J)}_{\text{Trace}(A) = \sum_{i=1}^{n} A_{ii}}}_{\text{log-likelihood}} - \lambda \underbrace{\underbrace{\|J\|_{L1}}_{\text{sparsity encouraging regularizer}}}_{\text{regularizer}} \Big\}$$

$$\|J\|_{L1} = \sum_{i,j} |J_{ij}|$$

* This is a concave maximization, and solved with gradient ascent.

* This is more accurate than node-wise neighborhood learning as all edges are learnt jointly.