*Overview

- Graphical Models & Markov Properties
- Inference Problems: given $P_G(x)$ find
  - $P(x_i)$
  - $\arg\max_x P_G(x)$
  - Compute Z
  - Belief Propagation
  - Variational methods
  - Gibbs Sampling
- Learning Graphical model

  given samples: $x^{(1)}, x^{(2)}, \ldots, x^{(N)} \in \mathcal{X}^n$

  ↳ Learn the structure of the graph G

  ↳ Learn the parameters of factors

*Structural Learning

- $X \in \mathcal{X}^n$ Random Vector
- Directed Graphical Model : $|E| = \binom{n}{2} = \frac{n(n-1)}{2}$

$$\#\text{ of possible graphs} \leq 3^{|E|} \approx 3^{\frac{n(n-1)}{2}}$$

- Given $x^{(1)}, \ldots, x^{(N)}$ indep. samples from unknown $P(x)$
  - How do we "score" each graph?
  - How do we find the graph with highest score?
- There are 2 ways to approach such statistical problems.

Frequentist

- Assume graph G and $P = \{P(X_i | X_{\pi_i})\}_{i=1}^n$ are deterministic but unknown
- Maximum likelihood (ML) estimate find (G, P) that maximizes:

$$\max_{G,P} \underbrace{\sum_{j=1}^{N} \log P_{G,P}(x^{(j)})}_{\text{SCORE}(G,P)}$$

Bayesian

- Assume (G, P) is randomly drawn from known dist. $Q_{G,P}$
- Maximum A Posteriori (MAP) estimate, maximizes:

$$\max_{G,P} P(G, P | x^{(1)}, \ldots, x^{(N)})$$

* Frequentist : structural learning

$$\hat{G} = \arg\max_{G} \left\{ \underbrace{\max_{P} \frac{1}{N} \sum_{j=1}^{N} \log P_{G,P}(x^{(j)})}_{\text{SCORE}(G)} \right\}$$

* Simple case with $n=2$, $X=(X_1, X_2) \in \{0,1\}^2$

samples $(0,0)$ $(0,1)$ $(1,0)$ $(1,1)$ (x_1 ↓, x_2 →)

$\hat{P}_{x_1,x_2}$

$x_1 \backslash x_2$	0	1
0	.1	.3
1	.2	.4

Case 1: $G_1 = (① \ ②)$ independent $\rightarrow P_{12}(x_1,x_2) = P_1(x_1) \cdot P_2(x_2)$

ML estimate:

$$\max_{P_1} \frac{1}{N} \sum_{j=1}^{N} \log P_1(x_1^{(j)}) + \max_{P_2} \frac{1}{N} \sum_{j=1}^{N} \log P_2(x_2^{(j)})$$

def. of empirical distribution

$$= \max_{P_1=(P_1(0),P_1(1))} \hat{P}_1(0) \cdot \log P_1(0) + \hat{P}_1(1) \cdot \log P_1(1) + \cdots$$

$$= \max_{P_1} \sum_{x_1} \hat{P}_1(x_1) \cdot \log P_1(x_1) + \cdots$$

$$= \max_{P_1} \sum_{x_1} \hat{P}_1(x_1) \log \hat{P}_1(x_1) + \sum_{x_1} \hat{P}_1(x_1) \log \frac{P_1(x_1)}{\hat{P}_1(x_1)} + \cdots$$

$\sum_{x_1} \hat{P}_1(x_1) \log \hat{P}_1(x_1) = -H(\hat{P}_1)$ ↳ Does not depend P_1

$\sum_{x_1} \hat{P}_1(x_1) \log \frac{P_1(x_1)}{\hat{P}_1(x_1)} = -D_{KL}(\hat{P}_1 \| P_1) \leq 0$

max achieved when

$$P_1 = \hat{P}_1, \quad P_2 = \hat{P}_2$$

for G_1 $(① \ ②)$ $\max_{P_1 \times P_2} \frac{1}{N} \sum_{j=1}^{N} \log P_1(x_1^{(j)}) \cdot P_2(x_2^{(j)})$

$$SCORE(G_1) = -H(\hat{P}_1) - H(\hat{P}_2)$$

Case 2: $G_2 = (① \rightarrow ②)$

$$\max_{P_{12}(x_1,x_2)} \frac{1}{N} \sum_{j=1}^{N} P_{12}(x_{12}^{(j)}) = -H(\hat{P}_{12}) - D_{KL}(\hat{P}_{12} \| P_{12})$$

max achieved at $P_{12} = \hat{P}_{12}$

$$SCORE(G_2) = -H(\hat{P}_{12})$$

$$\mathcal{L}(① ②) = -H(\hat{P}_1) - H(\hat{P}_2)$$

$$\mathcal{L}(① \to ②) = -H(\hat{P}_{12})$$

follows from Independence makes entropy ↑

$H(P_{12}) \leq H(P_1) + H(P_2)$

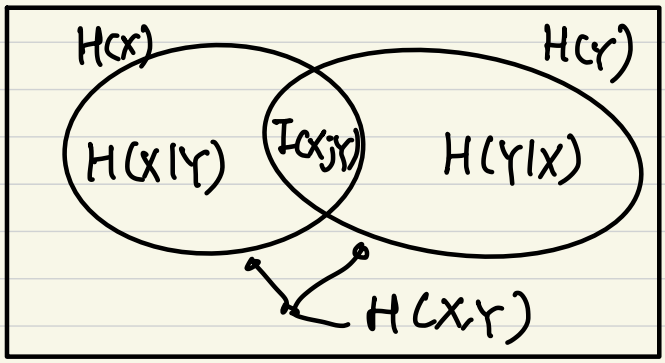
· Remark 1: $-H(\hat{P}_{12}) \geq -H(\hat{P}_1) - H(\hat{P}_2)$

Refresher on H, I. X, Y random vectors from joint $P_{X,Y}(x,y)$

$$I(X;Y) \triangleq \sum_{x,y} P(x,y) \cdot \log \frac{P(x,y)}{P(x) \cdot P(y)}$$

$$H(X) \triangleq \sum_{x} -P(x) \log P(x)$$

$$H(Y|X) \triangleq \sum_{x,y} -P(x,y) \log P(y|x)$$



$$H(Y) = H(Y|X) + I(X;Y)$$

$$H(X) + H(Y) = H(X,Y) + I(X;Y)$$

$$\mathcal{L}(① \to ②) \geq \mathcal{L}(① ②)$$

$$\| \qquad\qquad \|$$

$$-H(\hat{P}_{12}) \qquad -H(\hat{P}_1) - H(\hat{P}_2)$$

always get the densest graph = Overfitting

↳ even if the truth is ① ②, you choose ① → ②

· Remark 2. depending on N and target false positive rate β decision is made by:

$$\mathcal{L}(① \to ②) - \mathcal{L}(① ②) = -H(\hat{P}_{12}) + H(\hat{P}_1) + H(\hat{P}_2)$$

$$= +I_{\hat{P}_{12}}(X_1; X_2)$$

output { ① ② if $I_{\hat{P}_{12}}(X_1; X_2) > \frac{t_\beta}{N}$; ① → ② else

* Maximum Likelihood Approach for a DAG

$$G^* \in \operatorname*{argmax}_G \left\{ \max_{\{P(x_i|x_{\pi_i})\}_{i=1}^n} \frac{1}{N} \sum_{j=1}^{N} \log \prod_{i=1}^{n} P_i(x_i^{(j)}|x_{\pi_i}^{(j)}) \right\}$$

$$P_i(x_i|x_{\pi_i}) = \hat{P}_i(x_i|x_{\pi_i})$$

SCORE (G)

$$\Rightarrow \frac{1}{N}\sum_{j=1}^{N} \log\left(\prod_{i=1}^{n} \hat{P}_i(x_i^{(j)}|x_{\pi_i}^{(j)})\right)$$

$$= \sum_{i=1}^{n} \frac{1}{N}\sum_{j=1}^{N} \log \hat{P}_i(x_i^{(j)}|x_{\pi_i}^{(j)})$$

def empirical distribution →

$$= \sum_{i=1}^{n} \sum_{x_i, x_{\pi_i}} \hat{P}_i(x_i, x_{\pi_i}) \cdot \log \hat{P}_i(x_i|x_{\pi_i})$$

$$= \sum_{i=1}^{n} \{ - H_{\hat{P}}(X_i|X_{\pi_i}) \} \quad \leftarrow H(X|Y) = H(X) - I(X;Y)$$

$$= \sum_{i=1}^{n} I_{\hat{P}}(X_i; X_{\pi_i}) - H_{\hat{P}}(X_i)$$

$I_{\hat{P}}(X_i; X_{\pi_i})$: depends on Graph

$H_{\hat{P}}(X_i)$: does not depend on G.

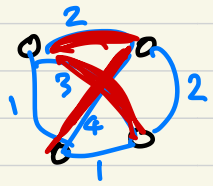
* Remark
- this gives a "SCORE" = likelihood for any given DAG G.
- Search over all DAGs.
- $I_{\hat{P}}(X_i; X_{\pi_i}) \geq I_{\hat{P}}(X_i; X_{\pi_i'})$ if $\pi_i \supset \pi_i'$

⇒ we need to restrict the family of graphs we are searching over. = set of trees

* Chow-Liu algorithm: searches over all Trees, efficiently, exactly.

Step 1: Create a complete graph (undirected) over $V = \{1, 2, \cdots, n\}$
give edge weight $w_{ij} = I_{\hat{p}}(X_i; X_j)$

$$W = \begin{bmatrix} & w_{ij} & \\ & & \end{bmatrix}$$

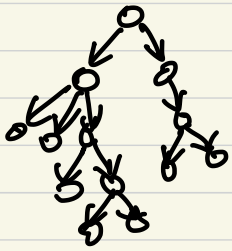


Step 2: find the <u>max-weight spanning Tree</u>, for example, using Kruskal's algorithm.

Claim: Chow Liu finds the optimal tree that maximizes

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

π_i: single node.



* Impractical approach for undirected graphical model learning.

Consider learning an Ising Model $(G, \{\theta_i\}_{i=1}^n, \{\theta_{ij}\}_{i,j=1}^n)$

$x_G \in \{-1, +1\}$

$$\theta = \begin{bmatrix} \theta_1 & & \theta_{ij} \\ & \theta_2 & \\ & & \ddots \end{bmatrix}$$

$$\boxed{p_{G,\theta}(x) = \frac{1}{Z_\theta} \cdot \prod_{i=1}^{N} e^{\theta_i x_i} \cdot \prod_{(ij)\in E} e^{\theta_{ij} x_i x_j}}$$

samples $\{x^{(1)}, x^{(2)}, \dots, x^{(N)}\} = D$

The likelihood

$$p_{G,\theta}(D) = \prod_{l=1}^{N} p_{G,\theta}(x^{(l)})$$

$$= \prod_{l=1}^{N} \frac{1}{Z_\theta} \prod_{i\in V} e^{\theta_i x_i^{(l)}} \cdot \prod_{(i,j)\in E} e^{\theta_{ij} \cdot x_i^{(l)} x_j^{(l)}}$$

$$= \exp\left\{ -N \log Z_\theta + \sum_{i\in V} N \cdot \hat{\mu}_{ii} \cdot \theta_i + \sum_{(i,j)\in E} N \cdot \hat{\mu}_{ij} \cdot \theta_{ij} \right\}$$

$\hat{\mu}_{ii} \triangleq \frac{1}{N} \sum_{l=1}^{N} x_i^{(l)}$ $\quad$ $\hat{\mu}_{ij} = \frac{1}{N} \sum_{l=1}^{N} x_i^{(l)} x_j^{(l)}$

the log-likelihood is

$$\min \quad L(G, \theta, D) = -\frac{1}{N} \log p_{G,\theta}(D)$$

$$= \Phi(\theta) - \log Z_\theta - \langle \hat{\mu}, \theta \rangle \cdot N$$

$\Phi(\theta)$ convex

$\log Z_\theta$ involves exp-time

$\langle \hat{\mu}, \theta \rangle$ linear in θ

$\hat{\mu} = \begin{bmatrix} \hat{\mu}_{11} & & \hat{\mu}_{ij} \\ & \hat{\mu}_{22} & \\ & & \ddots \end{bmatrix}$ $\quad$ $\theta = \begin{bmatrix} \theta_1 & & \theta_{ij} \\ & \theta_2 & \\ & & \ddots \end{bmatrix}$

Remark:

* learning is easier if inference is easier.