

# Learning Markov Networks : Maximum Bounded Tree-Width Graphs

David Karger and Nathan Srebro

Graphical Models Reading Group

January 15/22, 2004

## The Learning Problem

- Suppose that the random variables  $(X_1, X_2, \dots, X_n)$  are distributed according to the distribution  $P_{\text{true}}[\cdot]$ .
- The problem is that this distribution is unknown to us, and we wish to estimate or *learn* it by sampling from it.
- Let  $x^{(1)}, x^{(2)}, x^{(3)}, \dots, x^{(k)}$  be  $k$  independent samples drawn from this distribution.
- We may estimate the distribution to be the empirical distribution given by

$$P_{\text{emp}}[X = x] = \frac{|\{l \mid x^{(l)} = x\}|}{T}$$

- Many problems with this approach (overfitting, curse of dimensionality etc. )

## Dealing with overfitting

- Solution : Become Bayesians - restrict the acceptable distributions.
- Let  $\mathcal{D}$  be a subset of the set of all distributions on  $(X_1, \dots, X_n)$ . Given some measure of goodness  $Q(P)$  of a distribution  $P$ , we seek

$$\mathcal{P} = \arg \min_{P \in \mathcal{D}} Q(P)$$

- $\mathcal{D}$  represents our prior assumptions about the true distribution.  $Q(\cdot)$  represents our belief about the quality of the chosen distribution.

## Choosing $\mathcal{D}$ and $Q$

**Definition 1 (likelihood estimate).** *For a given distribution  $P$ , the likelihood estimate of  $P$  is the probability of observing the data if  $P$  is the true distribution. Therefore,  $Q(P) = P(X^{(1)} = x^{(1)}, \dots, X^{(k)} = x^{(k)})$ . The maximum-likelihood distribution is the distribution from the distribution class  $\mathcal{D}$  that maximizes this quantity.*

$$\begin{aligned} P_{\text{ML}} &= \arg \max_{P \in \mathcal{D}} Q(P) \\ &= \arg \max_{P \in \mathcal{D}} P(X^{(1)} = x^{(1)}, \dots, X^{(k)} = x^{(k)}) \\ &= \arg \max_{P \in \mathcal{D}} \prod_{l=1}^k P(X^{(l)} = x^{(l)}) \\ &= \arg \max_{P \in \mathcal{D}} \sum_{l=1}^k \log P(X^{(l)} = x^{(l)}) \end{aligned}$$

## Information Theoretic Connections

$$\begin{aligned}
 P_{\text{ML}} &= \arg \max_{P \in \mathcal{D}} \sum_{l=1}^k \log P(X^{(l)} = x^{(l)}) \\
 &= \arg \max_{P \in \mathcal{D}} \sum_{x \in \text{Range}(X)} \log P(x) \cdot \left| \{l \mid x^{(l)} = x\} \right| \\
 &= \arg \max_{P \in \mathcal{D}} \sum_{x \in \text{Range}(X)} \log P(x) \cdot \frac{|\{l \mid x^{(l)} = x\}|}{k} \\
 &= \arg \max_{P \in \mathcal{D}} \sum_{x \in \text{Range}(X)} P_{\text{emp}}[x] \log P(x) \\
 &= \arg \max_{P \in \mathcal{D}} H(P_{\text{emp}}) - D(P_{\text{emp}} \| P) \\
 &= \arg \min_{P \in \mathcal{D}} D(P_{\text{emp}} \| P)
 \end{aligned}$$

## Triangulations

**Definition 2 (Triangulated Graph).** *A graph is triangulated if it contains no chordless cycle of length greater than three.*

**Lemma 3. TFAE**

1.  *$G$  is triangulated.*
2.  *$G$  contains no induced subgraph isomorphic to  $C_n$  for  $n > 3$ .*
3. *Every minimal vertex separator of  $G$  induces a clique of  $G$ .*
4.  *$G$  is the intersection graph of a family of subtrees of a tree.*

## Hammersley-Clifford

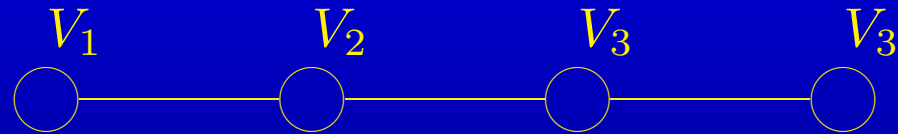
**Theorem 4 (Hammersley-Clifford).** *A Markov network/MRF (with positive distribution) can be factored over its cliques :*

$$P_G [X = x] = \prod_{H \in \mathcal{C}(G)} \phi_H(x_H)$$

- $\mathcal{C}(G)$  must contain all maximal cliques.
- An explicit formula for  $\phi_H(x_H)$  when  $G$  is triangulated is

$$\phi_H(x_H) = \frac{P_H [X_H = x_H]}{\prod_{H' \subseteq H} \phi_{H'}(x_{H'})}$$

## More on clique factors



Consider the simple Markov chain shown. The set of complete cliques is

$$\{\{V_1\}, \{V_2\}, \{V_3\}, \{V_4\}, \{V_1, V_2\}, \{V_2, V_3\}, \{V_3, V_4\}\}$$

and the set of maximal cliques is

$$\{\{V_1, V_2\}, \{V_2, V_3\}, \{V_3, V_4\}\}$$

## More on clique factors

Let  $\text{Prob}(V_1V_2V_3V_4 = v_1v_2v_3v_4) = P(v_1v_2v_3v_4)$ . Then we have

$$\begin{aligned}
 P(v_1v_2v_3v_4) &= \psi_{V_1}(v_1)\psi_{V_2}(v_2)\psi_{V_1V_2}(v_1v_2)\psi_{V_3}(v_3)\psi_{V_2V_3}(v_2v_3)\psi_{V_4}(v_4)\psi_{V_3V_4}(v_3v_4) \\
 &= [\psi_{V_1}(v_1)\psi_{V_2}(v_2)\psi_{V_1V_2}(v_1v_2)] \cdot [\psi_{V_3}(v_3)\psi_{V_2V_3}(v_2v_3)] \cdot \\
 &\quad [\psi_{V_4}(v_4)\psi_{V_3V_4}(v_3v_4)] \\
 &= \Psi_{V_1V_2}(v_1v_2) \cdot \Psi_{V_2V_3}(v_2v_3) \cdot \Psi_{V_3V_4}(v_3v_4) \\
 &= [\psi_{V_1}(v_1)\psi_{V_1V_2}(v_1v_2)] \cdot [\psi_{V_3}(v_3)\psi_{V_2}(v_2)\psi_{V_2V_3}(v_2v_3)] \cdot \\
 &\quad [\psi_{V_4}(v_4)\psi_{V_3V_4}(v_3v_4)] \\
 &= \Phi_{V_1V_2}(v_1v_2) \cdot \Phi_{V_2V_3}(v_2v_3) \cdot \Phi_{V_3V_4}(v_3v_4)
 \end{aligned}$$

## Cliques : Maximal or All ?

- The set of clique factors  $\{\phi_H\}_{H \in \mathcal{C}(G)}$  can either be the set of maximal cliques or the set of all cliques.
  - ★ When  $\mathcal{C}(G)$  is the set of all cliques, then the clique factors are purely local, and depend only on the local marginals, independent of graph structure.
  - ★ When  $\mathcal{C}(G)$  is the set of maximal cliques, then the clique factors in general depend on the structure of the graph, not just local marginals.
  - ★ The factorization over maximal cliques depends on which maximal clique to collapse smaller cliques into, which in turn depends on graph structure.

# Graphical Models and Maximum-likelihood

$$\begin{aligned}
 \log \prod_{l=1}^k \mathbb{P}_G \left[ X^{(l)} = x^{(l)} \right] &= \sum_{l=1}^k \sum_{H \in \text{Clique}(G)} \log \phi_H(x_H^{(l)}) \\
 &= \sum_{H \in \text{Clique}(G)} \sum_{l=1}^k \log \phi_H(x_H^{(l)}) \\
 &= \sum_{H \in \text{Clique}(G)} \sum_{x_H \in \text{Range}(X_H)} k \cdot \mathbb{P}_{\text{emp}} [X_H = x_H] \log \phi_H(x_H) \\
 &= k \sum_{H \in \text{Clique}(G)} \mathbb{E}_{\mathbb{P}_{\text{emp}}} [\log \phi_H(X_H)] \\
 &= k \sum_{H \in \text{Clique}(G)} w(H)
 \end{aligned}$$

## Clique Weights and Entropy

Suppose that we consider the factorization over all cliques. Then we have

$$\phi_H(x_H) = \frac{\text{Prob}(X_H = x_H)}{\prod_{H' \subseteq H} \phi_{H'}(x_{H'})}$$

We defined the clique weights by

$$\begin{aligned} w(H) &= \mathbb{E} [\log \phi_H(X_H)] \\ &= \mathbb{E} \left[ \log \frac{\text{Prob}(X_H = x_H)}{\prod_{H' \subseteq H} \phi_{H'}(X_{H'})} \right] \\ &= \mathbb{E} \left[ \log \text{Prob}(X_H) - \sum_{H' \subseteq H} \log \phi_{H'}(x_{H'}) \right] \\ &= -H(X_H) - \sum_{H' \subseteq H} w(H') \end{aligned}$$

Therefore, we get

$$w(H) = - \sum_{H' \subseteq H} (-1)^{(|H| - |H'|)} H(X_{H'})$$

In particular

1. The weights of all 1 cliques are negative.
2. The weight of a 2-clique is

$$w(\{u, v\}) = -H(X_u X_v) + H(X_u) + H(X_v) = I(X_u; X_v) \geq 0$$

3. The weight of a 3-clique is

$$\begin{aligned} w(\{u, v, w\}) &= -H(X_u X_v X_w) \\ &\quad + H(X_u X_v) + H(X_v X_w) + H(X_w X_u) \\ &\quad - H(X_u) - H(X_v) - H(X_w) \\ &\geq 0 \end{aligned}$$

## Maximum-likelihood Markov Models

- This being a graphical-models reading group,  $\mathcal{D}$  is a subset of the class of distributions arising from graphical models.
- If we set  $\mathcal{D}$ , the allowable class of distributions to be the class of all Markov models, then the optimal graphical model is the complete graph.
- The number of parameters to learn is exponential in the size of the largest clique.
- The class of models we will consider will be the class of (triangulated) graphs in which the size of the largest clique is  $k + 1$ .
- For  $k = 1$ , this problem was solved in [2].

## $k$ -Trees

- Important subclass of the family of triangulated graphs.
- 1-trees are trees.
- 2-trees are two-terminal series-parallel networks.
- All planar graphs are (partial) 3-trees.

**Definition 5.**  $k$ -trees are defined recursively as follows.

1. A clique with  $k + 1$  vertices is a  $k$ -tree.
2. Given a  $k$ -tree  $T_n$  on  $n$  vertices, a  $k$ -tree on  $n + 1$  vertices is obtained by adding a (distinct) vertex to  $T_n$  and connecting it (only) to a  $k$ -clique of  $T_n$ .

## More on $k$ -trees

**Lemma 6.** *A graph  $G$  is a  $k$ -tree if and only if*

1.  *$G$  is connected.*
2.  *$\omega(G) = k + 1$ .*
3. *Every minimal separator of  $G$  is a  $k$ -clique.*

**Definition 7.** *A partial  $k$ -tree is a subgraph of a  $k$ -tree.*

**Lemma 8.** *Every partial  $k$ -tree with at least  $k + 1$  vertices can be triangulated into a  $k$ -tree.*

## Still more on $k$ -trees

**Definition 9.** *The treewidth of a graph  $G$  is the minimum value  $k$  for which  $G$  is a partial  $k$ -tree.*

- $G_1 \subseteq G_2 \Rightarrow \text{TW}(G_1) \leq \text{TW}(G_2)$
- $\text{GW}(G) \leq \text{size of largest clique in } G$ .
- A graph of treewidth  $k$  can be colored using  $k$ -colors, using a polynomial time algorithm for fixed  $k$ .
- There is a polynomial time algorithm to determine the treewidth of a graph (for fixed  $k$ ).

## Hypergraphs and coverings

### Definition 10.

*A hypergraph  $H$  is a pair  $(V, \mathcal{E})$ , where  $\mathcal{E} \subseteq 2^V$ .*

*We will associate  $H = (V, \mathcal{E}) \rightarrow G = (V, E)$ , where*

$$E = \{(x, y) \mid x \neq y, \{x, y\} \subseteq e \in \mathcal{E}\}$$

*A hypergraph  $H$  is called a hyperforest if there are no cycles in  $H$ .*

*The width of a hypergraph is the size of the largest edge in it.*

*A tree decomposition of  $G = (V, E)$  is a clique-hypergraph  $H = (V, \mathcal{E})$  associated with it.*

**Lemma 11.**  *$G$  is covered by a  $k$ -hyperforest iff  $G$  has tree-width at most  $k$ .*

## The maximum weight hypertree problem

**Problem 12 (Maximum weight hypertree problem).** *Let  $k$  be a fixed integer. Suppose that the input is a vertex set  $V$  and a weight function  $\binom{V}{k+1} \rightarrow \mathbb{R}$ . Find a hyperforest  $H(V)$  of width at most  $k$  that maximizes  $\sum_{h \in H} w(h)$ .*

Note that if  $w(h) \geq 0$  for all  $h \in H$ , then the hyperforest can always be expanded into a hypertree. However, this condition is only sufficient, not necessary.

**Definition 13.** *A weight function  $w : \binom{V}{k+1} \rightarrow \mathbb{R}$  is monotone if it is monotone on hyperforests, i.e., if  $H' \subseteq H$ , then  $w(H) \geq w(H')$ .*

**Theorem 14 (Hardness of maximum hypertree).** *The maximum hypertree problem is NP-hard even for treewidth  $k = 2$  and monotone weights.*

## Windmills

**Definition 15. 1.** *Let  $T(V)$  be a rooted tree with depth at most  $k$ . A  $k$ -windmill based on the rooted tree  $T(V)$  is a hypergraph  $H$  whose edges are paths radiating from  $r$  in  $T$ .*

- If all paths radiating from  $r$  in  $T(V)$  are of length  $k$ , then  $H$  is a regular  $k$ -windmill.*
- A  $k$ -windmill-farm is a hypergraph that is a disjoint collection of  $k$ -windmills.*

**Theorem 16. Windmill cover theorem** *For any hyperforest  $H(V)$  with width  $k$  and non-negative weight function  $w(\cdot)$ , there exists a  $k$ -windmill farm  $F(V)$  such that  $w(H) \leq (k + 1)!w(F)$ .*

## Proof of Theorem 16

Color the vertices  $V$  using  $k + 1$  colors. Randomly choose a permutation  $\pi$  of the colors. Let

$$F_\pi(V) = \{h \in H(V) \mid \text{colors of } h = \{\pi_0, \pi_1, \dots, \pi_{|h|-1}\}\}$$

For each hyperedge  $h \in H(V)$ , there is some permutation  $\sigma$  of the colors such that  $h \in F_\sigma(V)$ . Therefore, the expected weight of  $F_\pi(V) \geq \frac{w(H)}{(k+1)!}$ . Further,  $F_\pi(V)$  is a  $k$ -windmill farm, with the colors representing the levels of the vertices in the windmill farm.

## ILP formulation for optimal-2-windmills

Suppose for each  $v \in V$ , we have an assignment of  $\ell(v) \in \{0, 1, 2\}$ . Let  $V_i = \{v \in V \mid \ell(v) = i\}$ . Define variables  $x_{v_1, v_2, v_3}$  for each  $(v_1, v_2, v_3) \in V_0 \times V_1 \times V_2$ . The ILP is then

$$\begin{aligned}
 & \max && \sum_{(v_0, v_1, v_2) \in V_0 \times V_1 \times V_2} x_{v_1, v_2, v_3} w_{v_1, v_2, v_3} \\
 & \text{subject to} && \sum_{v_1 \in V_1} x_{v_1, v_2} \leq 1 && \forall v_2 \in V_2 \\
 & && \sum_{v_1, v_2 \in V_1 \times V_2} x_{v_1, v_2, v_3} \leq 1 && \forall v_3 \in V_3 \\
 & && x_{v_1, v_2, v_3} \leq x_{v_1, v_2}
 \end{aligned}$$

## Bounds on optimal 2-windmills

$$\begin{aligned} w(\text{rounded IP Farm}) &\geq w(\text{LP Fractional Farm})/2 \\ &\geq w(\text{optimal Farm subject to constraints})/2 \\ &\geq w(\text{optimal Farm})/2 \cdot 27 \\ &\geq w(\text{optimal Hypertree})/2 \cdot 27 \cdot 6 \end{aligned}$$

## General Case

1. Variable  $x_p$  for each path  $p$  of length at most  $k$ .
2. Weight of a path  $p.v$  is  $\sum_{h \subseteq p} w(h) - \sum_{h \subseteq q} w(h)$ .
3. Constraints : Path present only if subpaths present.
4. Rounding guarantee :  $\frac{1}{8^k \cdot k!}$ .

## Open Questions

1. Hardness guarantee of problems.
2. Approximation guarantee independent of  $k$ .
3. ILP (or SDP etc.) formulation for optimal  $k$ -trees.
4. Use of metrics other than KL/ML.

## References

- [1] David Karger and Nathan Srebro, *Learning Markov Networks : Maximum Bounded Tree-Width Graphs*. SODA, 2001
- [2] C. K. Chow and C. N. Liu, *Approximating discrete probability distributions with dependence trees*. IEEE ToIT, 1968.
- [3] Nathan Srebro, *Maximum Likelihood Bounded Tree-Width Markov Networks*. JAI, 2003?