COMPSCI 688: Probabilistic Graphical Models

Lecture 7: Undirected Graphical Models: Examples and Inference

Dan Sheldon

Manning College of Information and Computer Sciences
University of Massachusetts Amherst

Partially based on materials by Benjamin M. Marlin (marlin@cs.umass.edu) and Justin Domke (domke@cs.umass.edu)

Review

Markov Random Fields

A Markov random is a distribution that factors over a set of "cliques" $\mathcal{C}$:

$$p(\mathbf{x}) = \frac{1}{Z}\prod_{c\in\mathcal{C}} \phi_c(\mathbf{x}_c), \quad Z = \sum_{\mathbf{x}}\prod_{c\in\mathcal{C}} \phi_c(\mathbf{x}_c)$$

The *dependence graph* $\mathcal{G} = (V, E)$ is the graph where nodes i and j are connected by an edge if they appear together in some factor.

We say that $p(\mathbf{x})$ *factors* over $\mathcal{G}$, and denote this property as (F).

Markov Properties

The *global Markov property* (G) connects conditional indpendence to graph separation.

Distribution $p(\mathbf{x})$ satisfies the global Markov property with respect to $\mathcal{G}$ if

$$\text{sep}_{\mathcal{G}}(A, B|S) \implies \mathbf{X}_A \perp \mathbf{X}_B \mid \mathbf{X}_S \tag{G}$$

There are two other Markov properties (*local* and *pairwise*) implied by the global Markov property.

Factorization and Markov Properties

It's easy to show that factorization implies Markov: (F) ⇒ (G).

There is a famous partial converse. For a *positive* distribution: (G) ⇒ (F)

Theorem (Hammersley-Clifford). If $p(\mathbf{x}) > 0$ for all $\mathbf{x}$, then (F) $\iff$ (G)

Examples

Example: Ising Model

- $\mathcal{G}$ is a lattice and $X_i \in \{-1, 1\}$
- Have *unary potential* β_i for each node i and *pairwise potential* β_{ij} for each edge (i, j)

$$p(\mathbf{x}) = \frac{1}{Z} \prod_i \beta_i(x_i) \prod_{(i,j)\in E} \beta_{ij}(x_i, x_j)$$

$$\beta_i(x_i) = \exp(\overset{10}{b_i} x_i)$$

$$\begin{cases} e^{10} & x_i = 1 \\ e^{-10} & x_i = -1 \end{cases}$$

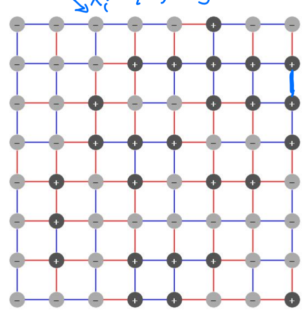
$$\beta_{ij}(x_i, x_j) = \exp(\underset{10}{b_{ij}} x_i x_j)$$

- $b_i > 0 \implies X_i$ likes to be positive
- $b_{ij} > 0 \implies X_i$ and X_j like to be the same

$$\begin{cases} e^{10} & x_i = x_j \\ e^{-10} & x_i \neq x_j \end{cases}$$

$\beta_i(x_i)$

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



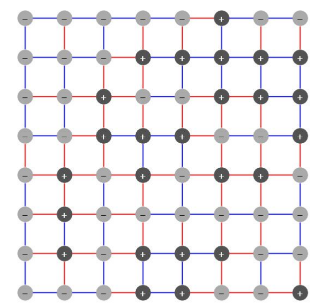
x_i

x_j

$\beta_{ij}(x_i, x_j)$

Example: Ising Model

- In general, Markov networks can be seen as expressing preferences for certain local configurations of the variables.
- Joint configurations with high probability balance the preferences of all factors.



Example: Simulating an Ising Model

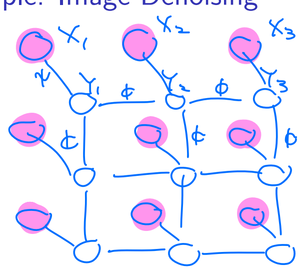
Demo: Ising Model

$$p(\mathbf{x}) = \frac{\exp\left(\frac{1}{T}\sum_{(i,j)\in E} x_i x_j\right)}{Z} = \frac{1}{Z} \prod_{(i,j)\in E} \exp\left(\frac{1}{T} x_i x_j\right)$$

Example: Statistical Image Models

The Ising model with $b_{ij} > 0$ prefers smoothness, and can be used as a model for images in denoising procedures:

Example: Image Denoising



$y_i \in \{0, \dots, 255\}$ unknown true pixel values

$x_i \in \{0, \dots, 255\}$ observed noisy pixel value

$\Rightarrow$ dist. $p(\vec{x}, \vec{y}) = \frac{1}{Z} \prod_{(i,j)\in E} \phi(y_i, y_j) \cdot \prod_{i\in V} \psi(x_i, y_i)$

denoise: observe $\vec{x}$, sample from or maximize $p(\vec{y} \mid \vec{x})$

clean, noisy

Design process: potentials $\phi(x_i, x_j)$

$\psi(x_i, y_i)$

Example: Data Privacy

p(M, S, E, R, H, WC, O, I)

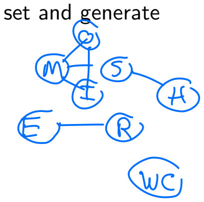
In differential privacy, graphical models are used to model a data set and generate synthetic data from privacy-preserving measurements.[1]

```
marginals = [('marital-status', 'sex'),
             ('education-num', 'race'),
             ('sex', 'hours-per-week'),
             ('workclass',),
             ('marital-status', 'occupation', 'income>50K')]

# MEASURE the marginals and log the noisy answers
measurements <- noise-perturbed marginals (for privacy)

# GENERATE synthetic data using PGM
engine = FactoredInference(data.domain, iters=2500)
model = engine.estimate(measurements)
synth = model.synthetic_data()
```

p(M,S)
p(E,R) ⇒
p(S,H)



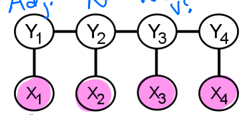
[1] Example from https://differentialprivacy.org/synth-data-1.

Conditional Random Fields

The image denoising model was an example of a **conditional random fields** (CRFs), a very important model class in machine learning. A CRF is essentially a Markov network where one set of nodes is always conditioned on.

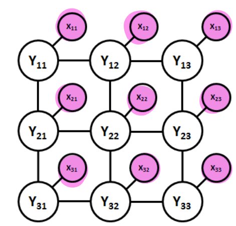
Adj. N N? V?

POS



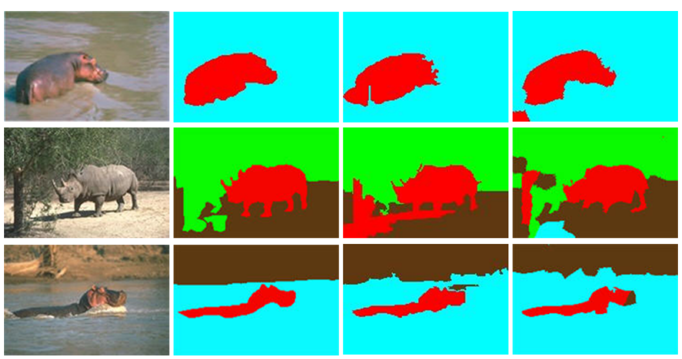
British airways rose —

word tokens

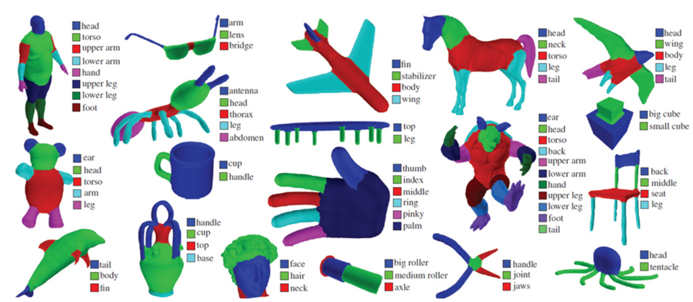


The $\mathbf{y}$ nodes are *labels*, and the $\mathbf{x}$ nodes are *features*.

Example: Image Segmentation



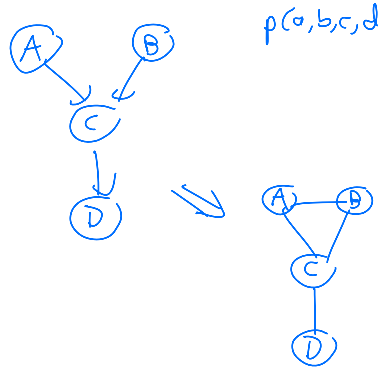
Example: 3D Mesh Segmentation

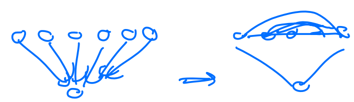


Example: Bayes Nets as MRFs

$$p(a,b,c,d) = p(a)\,p(b)\,p(c|a,b)\,p(d|c)$$

$$= \phi_1(a)\,\phi_2(b)\,\phi_3(a,b,c)\,\phi_4(c,d)$$





$p(x|a,b,c,d\ldots)$ $\quad$ $\phi(a,b,c,d\ldots,x)$

Example: Bayes Nets as MRFs

Some structure is lost in this transformation. When we replace $p(a|b,c)$ by $\phi(a,b,c)$, we "forget" that a Bayes net is **locally normalized**

$$\sum_a \phi(a,b,c) = 1 \quad \forall b, c.$$

This is a special property of Bayes nets and is central to V-structures, explaining away, and D-separation. It occurs "internally" to the factor $\phi(a,b,c)$ and is not represented in the MRF graph structure.

Similarly, when we replace $\prod_i p(x_i|\mathbf{x}_{\mathsf{pa}(i)})$ by $\frac{1}{Z}\prod_{c\in\mathcal{C}}\phi_c(\mathbf{x}_c)$, we "forget" that a Bayes net is **globally normalized**:

$$\sum_x \prod_{c\in\mathcal{C}} \phi_c(\mathbf{x}_c) = 1 \implies Z = 1.$$

This is another special property of Bayes nets that makes learning easy.