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

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

Markov Random Fields

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

$$p(x_1, x_2, x_3, x_4) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \phi_c(x_c)$$

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

Markov Properties

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

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

$$\text{sep}_G(A, B | S) \implies X_A \perp X_B | X_S \quad (G)$$

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: \( (F) \Rightarrow (G) \).

There is a famous partial converse. For a positive distribution: \( (G) \Rightarrow (F) \)

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

**Example: Ising Model**

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

\[
p(x) = \frac{1}{Z} \prod_{i \in V} \beta_i(x_i) \prod_{(i,j) \in E} \beta_{ij}(x_i, x_j)
\]

\[
\beta_i(x_i) = \exp(b_i x_i)
\]

\[
\beta_{ij}(x_i, x_j) = \exp(b_{ij} x_i x_j)
\]

- \( b_i > 0 \iff X_i \) likes to be positive
- \( b_{ij} > 0 \iff X_i \) and \( X_j \) like to be the same

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

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

Demo: Ising Model

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.
Conditional Random Fields

The previous two examples were examples of 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.

The y nodes are labels, and the x nodes are features.

Example: Image Segmentation
Example: 3D Mesh Segmentation

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 \( \prod_i p(x_i|x_{pa(i)}) \) by \( \frac{1}{Z} \prod_{c \in C} \phi_c(x_c) \), we “forget” that a Bayes net is **globally normalized**:

\[
\sum_x \prod_{c \in C} \phi_c(x_c) = 1 \implies Z = 1.
\]

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

Inference: Conditioning
Inference in Markov Networks

Given a Markov network, the main task is probabilistic inference, which means answering probability queries of the form:

\[ p(y|x) = \sum_u p(u,y|x) \]

- condition on \( x \) ("evidence")
- marginalize latent variables \( u \)
- compute the joint distribution of query variables \( y \)

Most steps of inference can be viewed as transforming a Markov network into a different one with fewer or simpler factors:
- Conditioning is easy
- Marginalization is hard! (exponential time in worst case, but linear in best case)

For fixed \( x_2 \), the conditional \( p(x_1|x_2) \) is proportional to the joint \( p(x_1,x_2) \). We can use the same factor, but hard-code \( x_2 \) so that only \( x_1 \) is a free variable:

\[ p(x_1,x_2) = p(x_1,x_2)' = \frac{1}{Z'} \phi'(x_1) \]

### Conditioning: Single Factor

Suppose we have a single-factor MRF \( p(x_1, x_2) = \frac{1}{Z} \phi(x_1, x_2) \) for two binary variables. We are given a fixed value for \( x_2 \), and want an MRF for \( p(x_1|x_2) \), i.e.:

\[ p(x_1|x_2) = \frac{1}{Z'} \phi'(x_1) \]

Observe

\[ p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)} = \frac{1}{p(x_2)} \cdot \frac{1}{Z} \phi(x_1, x_2) = Z' \cdot \phi'(x_1) \]

### Conditioning: General Case

For a general MRF, we can apply the same reasoning to reduce every factor by hard-coding the evidence variables.
Factor Reduction: Example

\[
\Phi_1(Y_1, Y_2) \\
\Phi_2(Y_2, X_2) \\
\Phi_3(Y_3, X_1)
\]

\[
\begin{array}{c|c|c}
Y_1=0 & Y_2=1 \\
Y_1=1 & 1 & 2 \\
\end{array}
\]

\[
\begin{array}{c|c|c}
Y_2=0 & X_1=0 & X_1=1 \\
Y_2=1 & 3 & 9 \\
Y_2=1 & 1 & 4 \\
\end{array}
\]

\[
\begin{array}{c|c|c}
Y_3=0 & X_1=0 & X_1=1 \\
Y_3=1 & 6 & 2 \\
Y_3=1 & 2 & 7 \\
\end{array}
\]

Query: \(P(Y_1, Y_2 | X_1=0, X_2=1)\)

Factor Reduction: Step 1

\[
\Phi_1(Y_1, Y_2) \\
\Phi'_2(Y_1) \\
\Phi'_3(Y_2)
\]

\[
\begin{array}{c|c|c}
Y_1=0 & Y_2=1 \\
Y_1=1 & 1 & 2 \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
Y_2=0 & X_1=0 & X_1=1 & Y_3=0 \\
Y_2=1 & 1 & 4 & 1 \\
\end{array}
\]

Query: \(P(Y_1, Y_2 | X_1=0, X_2=1)\)

Factor Reduction: Step 2

\[
\Phi_1(Y_1, Y_2) \\
\Phi'_2(Y_1) \\
\Phi'_3(Y_2)
\]

\[
\begin{array}{c|c|c}
Y_1=0 & Y_2=1 \\
Y_1=1 & 1 & 2 \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
Y_2=0 & X_1=0 & X_1=1 & Y_3=0 \\
Y_2=1 & 1 & 4 & 1 \\
\end{array}
\]

Query: \(P(Y_1, Y_2 | X_1=0, X_2=1) \propto \Phi_2(Y_1, Y_2) \Phi'_2(Y_1) \Phi'_3(Y_2)\)
Factor Reduction: General Algorithm

Suppose $p(x) = \frac{1}{Z} \prod_{c \in C} \phi_c(x_c)$ and we observe $X_i = x_i$ for a single node $i$

We obtain a new MRF for $p(x_{-i}|x_i)$ by the following procedure:

For each factor $\phi_c$ such that $i \in c$ is non-empty

- Replace $\phi_c(x_c)$ by $\phi_c'(x_c \setminus i) := \phi_c(x_c \setminus i, x_i)$
- The $x_c \setminus i$ variables remain “free”, and $x_i$ is hard-coded

To condition on many variables, we can repeat this procedure. Since order doesn’t matter, we can hard-code all evidence variables in each factor at the same time.

Next time, we’ll study *marginalization*, which is the process of summing over some variables to get the marginal distribution over others or the partition function.