Introduction to Graphical Models

Prof. Dr. H. H. Takada

Quantitative Research – Itaú Asset Management
Institute of Mathematics and Statistics – University of São Paulo

Graphical Models (GM)

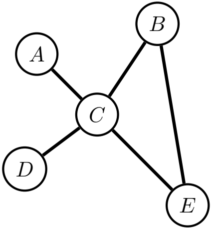
- Graphical modeling is the discipline of representing probability models graphically.
- Belief networks intuitively describe which variables ‘causally’ influence others and are represented using directed graphs.
- A Markov network is represented by an undirected graph. Markov networks are historically important in physics and may be used to understand how global collaborative phenomena can emerge from only direct local dependencies.
- Factor graphs describe the factorization of functions and are not necessarily related to probability distributions.
- Graphical models are generally limited in their ability to represent all the possible logical consequences of a probabilistic model.

Markov Network

Clique: Fully connected subset of nodes.

Maximal Clique: Clique which is not a subset of a larger clique.

A Markov Network is an undirected graph in which there is a potential (non-negative function) ψ defined on each maximal clique.



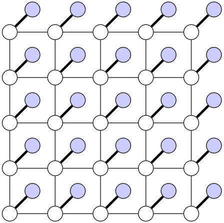
The joint distribution is proportional to the product of all clique potentials.

$$p(A,B,C,D,E) = \frac{1}{Z}\psi(A,C)\psi(C,D)\psi(B,C,E)$$

$$Z = \sum_{A,B,C,D,E} \psi(A,C)\psi(C,D)\psi(B,C,E)$$

Example Application of Markov Network – Part I

Problem: Markov Random Field (MRF) We want to recover a binary image from the observation of a corrupted version of it.



$X = \{X_i, i = 1, \ldots, D\}$ $X_i \in \{-1, 1\}$: clean pixel

$Y = \{Y_i, i = 1, \ldots, D\}$ $Y_i \in \{-1, 1\}$: corrupted pixel

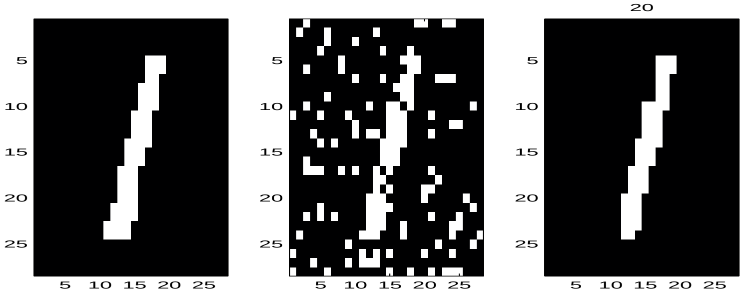
$\phi(Y_i, X_i) = e^{\gamma X_i Y_i}$ encourage Y_i and X_i to be similar

$\psi(X_i, X_j) = e^{\beta X_i X_j}$ encourage the image to be smooth

$$p(X, Y) \propto \left[\prod_{i=1}^{D} \phi(Y_i, X_i)\right] \left[\prod_{i \sim j} \psi(X_i, X_j)\right],$$

where $i \sim j$ indicates the set of variables that are neighbors in the MRF. Finding the most likely X given Y is not easy (since the graph is not singly-connected), but approximate algorithms often work well.

Example Application of Markov Network – Part II



left Original clean image

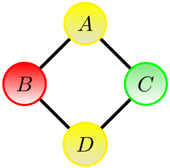
middle Observed (corrupted) image

right Most likely clean image $\underset{X}{\operatorname{argmax}}\ p(X|Y) = \underset{X}{\operatorname{argmax}}\ p(X, Y)$

Matlab

$>>$ `setup;` $>>$ `demoMRFclean;`

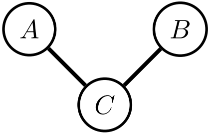
Independence in Markov Networks



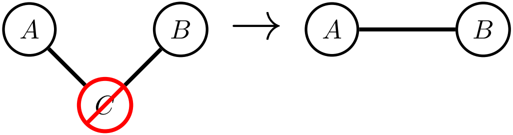
$$B \perp\!\!\!\perp C \mid A, D?$$
$$p(B|A,D,C) = p(B|A,D)?$$

$$\begin{aligned}
p(B|A,D,C) &= \frac{p(A,B,C,D)}{p(A,C,D)} \\
&= \frac{p(A,B,C,D)}{\sum_B p(A,B,C,D)} \\
&= \frac{\psi(A,B)\cancel{\psi(A,C)}\psi(B,D)\cancel{\psi(C,D)}}{\sum_B \psi(A,B)\cancel{\psi(A,C)}\psi(B,D)\cancel{\psi(C,D)}} \\
&= p(B|A,D)
\end{aligned}$$

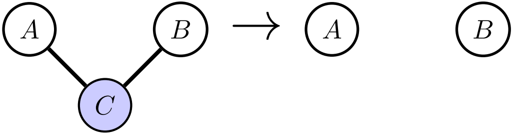
Properties of Markov Networks



$$p(A, B, C) = \phi_{AC}(A, C)\phi_{BC}(B, C)/Z$$

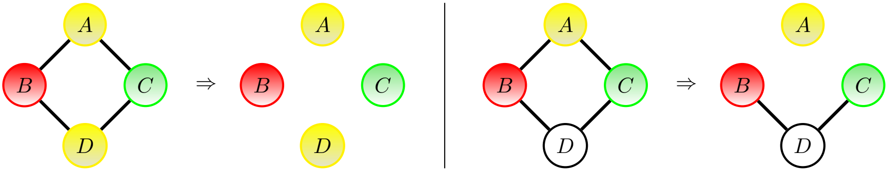


Marginalizing over C makes A and B (graphically) dependent. In general $p(A, B) \neq p(A)p(B)$.



Conditioning on C makes A and B independent: $p(A, B|C) = p(A|C)p(B|C)$.

General Rule for Independence in Markov Networks

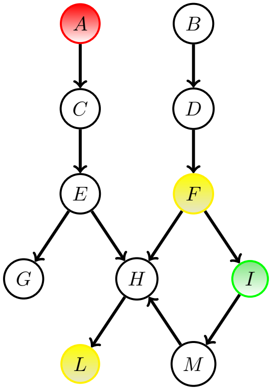


- Remove all links neighbouring the variables in the conditioning set $\mathcal{Z}$.
- If there is no path from any member of $\mathcal{X}$ to any member of $\mathcal{Y}$, then $\mathcal{X}$ and $\mathcal{Y}$ are conditionally independent given $\mathcal{Z}$.

Alternative Rule for Independence in Belief Networks

$\mathcal{X} \perp\!\!\!\perp \mathcal{Y} | \mathcal{Z}$?

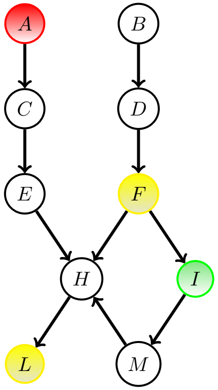
- Ancestral Graph: Remove any node which is neither in $\mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z}$ nor an ancestor of a node in this set, together with any edges in or out of such nodes.
- Moralization: Add a line between any two nodes which have a common child. Remove arrowheads.
- Separation: Remove all links from $\mathcal{Z}$.
- Independence: If there are no paths from any node in $\mathcal{X}$ to one in $\mathcal{Y}$ then $\mathcal{X} \perp\!\!\!\perp \mathcal{Y} \mid \mathcal{Z}$.



Alternative Rule for Independence in Belief Networks

$\mathcal{X} \perp\!\!\!\perp \mathcal{Y} | \mathcal{Z}$?

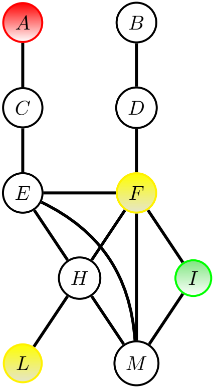
- Ancestral Graph: Remove any node which is neither in $\mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z}$ nor an ancestor of a node in this set, together with any edges in or out of such nodes.
- Moralization: Add a line between any two nodes which have a common child. Remove arrowheads.
- Separation: Remove all links from $\mathcal{Z}$.
- Independence: If there are no paths from any node in $\mathcal{X}$ to one in $\mathcal{Y}$ then $\mathcal{X} \perp\!\!\!\perp \mathcal{Y} \,|\, \mathcal{Z}$.



Alternative Rule for Independence in Belief Networks

$\mathcal{X} \perp\!\!\!\perp \mathcal{Y} | \mathcal{Z}$?

- Ancestral Graph: Remove any node which is neither in $\mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z}$ nor an ancestor of a node in this set, together with any edges in or out of such nodes.
- Moralization: Add a line between any two nodes which have a common child. Remove arrowheads.
- Separation: Remove all links from $\mathcal{Z}$.
- Independence: If there are no paths from any node in $\mathcal{X}$ to one in $\mathcal{Y}$ then $\mathcal{X} \perp\!\!\!\perp \mathcal{Y} \mid \mathcal{Z}$.



Alternative Rule for Independence in Belief Networks

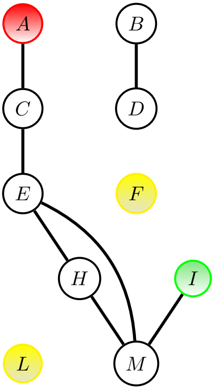
$\mathcal{X} \perp\!\!\!\perp \mathcal{Y} | \mathcal{Z}$?

- Ancestral Graph: Remove any node which is neither in $\mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z}$ nor an ancestor of a node in this set, together with any edges in or out of such nodes.

- Moralization: Add a line between any two nodes which have a common child. Remove arrowheads.

- Separation: Remove all links from $\mathcal{Z}$.

- Independence: If there are no paths from any node in $\mathcal{X}$ to one in $\mathcal{Y}$ then $\mathcal{X} \perp\!\!\!\perp \mathcal{Y} \,|\, \mathcal{Z}$.



$A \top\!\!\!\top I | F, L$

The Boltzmann machine

A MN on binary variables $\mathrm{dom}(x_i) = \{0, 1\}$ of the form

$$p(\mathbf{x}|\mathbf{w}, b) = \frac{1}{Z(\mathbf{w}, b)} e^{\sum_{i<j} w_{ij} x_i x_j + \sum_i b_i x_i}$$

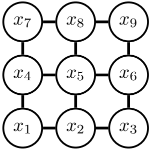
where the interactions w_{ij} are the 'weights' and the b_i the biases.

- This model has been studied in the machine learning community as a basic model of distributed memory and computation. The $x_i = 1$ represents a neuron 'firing', and $x_i = 0$ not firing. The matrix $\mathbf{w}$ describes which neurons are connected to each other. The conditional

$$p(x_i = 1|x_{\setminus i}) = \sigma\left(b_i + \sum_{j \neq i} w_{ij} x_j\right), \qquad \sigma(x) = e^x/(1 + e^x).$$

- The graphical model of the BM is an undirected graph with a link between nodes i and j for $w_{ij} \neq 0$. For all but specially constrained $\mathbf{w}$ inference will be typically intractable.
- Given a set of data $\mathbf{x}^1, \ldots, \mathbf{x}^n$, one can set the parameters $\mathbf{w}, b$ by maximum likelihood (though this is computationally difficult).

The Ising model



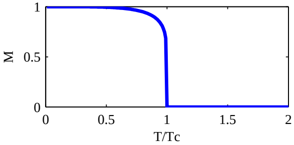
$x_i \in \{+1, -1\}$:

$$p(x_1, \ldots, x_9) = \frac{1}{Z} \prod_{i \sim j} \phi_{ij}(x_i, x_j)$$

$$\phi_{ij}(x_i, x_j) = e^{-\frac{1}{2T}(x_i - x_j)^2}$$

$i \sim j$ denotes the set of indices where i and j are neighbors in the graph. The potential encourages neighbors to be in the same state.

Spontaneous global behavior

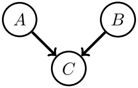


$M = |\sum_{i=1}^{N} x_i|/N$. As the temperature T decreases towards the critical temperature T_c a phase transition occurs in which a large fraction of the variables become aligned in the same state. Even though we only 'softly' encourage neighbours to be in the same state, for a low but finite T, the variables are all in the same state. Paradigm for 'emergent behavior'.

Expressiveness of Belief and Markov Networks

Cannot represent independence information in certain belief networks with a Markov network.

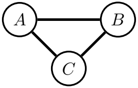
A Belief network



$$A \perp\!\!\!\perp B$$

Markov representation?

Since we have a term $p(C|A, B)$, the MN must have the clique A, B, C:

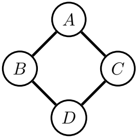


$$A \top\!\!\!\top B$$

Expressiveness of Belief and Markov Networks

Cannot represent independence information in certain Markov networks with a Belief network.

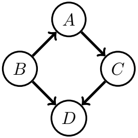
A Markov network



$$B \perp\!\!\!\perp C | A, D$$

Belief Network representation?

Any DAG on A, B, C, D must have a collider.



$$B \not\perp\!\!\!\perp C | A, D$$

Representations of distributions

- For a distribution P, form list of all the independence statements $\mathcal{L}_P$.
- For a graph G, form list of all the independence statements $\mathcal{L}_G$.

Then we define:

$$
\begin{array}{ll}
\mathcal{L}_P \subseteq \mathcal{L}_G & \text{Dependence Map (D-map)} \\
\mathcal{L}_P \supseteq \mathcal{L}_G & \text{Independence Map (I-map)} \\
\mathcal{L}_P = \mathcal{L}_G & \text{Perfect Map}
\end{array}
$$

In the above, we assume the statement l is contained in $\mathcal{L}$ if it is consistent with (can be derived from) the independence statements in $\mathcal{L}$.

Representations of distributions

Consider the distribution (class) defined on variables t_1, t_2, y_1, y_2

$$p(t_1, t_2, y_1, y_2) = p(t_1)p(t_2) \sum_h p(y_1|t_1, h)p(y_2|t_2, h)p(h)$$

In this case the list of all independence statements (for all distribution instances consistent with p) is

$$\mathcal{L}_P = \{t_1 \perp\!\!\!\perp (t_2, y_2), \;\; t_2 \perp\!\!\!\perp (t_1, y_1)\}$$

Consider the graph of the BN

$$p(y_2|y_1, t_1, t_2)p(y_1|t_1)p(t_1)p(t_2)$$

For this we have $\mathcal{L}_G = \{t_2 \perp\!\!\!\perp t_1, t_2 \perp\!\!\!\perp y_1\}$

- $\mathcal{L}_G \subset \mathcal{L}_P$ (since from the statement $t_2 \perp\!\!\!\perp (t_1, y_1)$ in $\mathcal{L}_P$ we can derive $t_2 \perp\!\!\!\perp t_1$ and $t_2 \perp\!\!\!\perp y_1$) so that the BN is an I-MAP for p since every independence statement in the BN is true for the corresponding graph.
- Since $\mathcal{L}_P \not\subseteq \mathcal{L}_G$ the BN is not a D-MAP for p.
- In this case no perfect map (a BN or a MN) can represent p.

Representing dependence?

GMs are generally most suited to represented independence. The reason is that local dependence doesn't imply global dependencies. For example

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

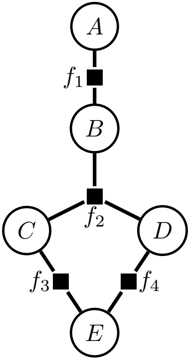
$$p(a) = \begin{pmatrix} 3/5 \\ 2/5 \end{pmatrix}, p(b|a) = \begin{pmatrix} 1/4 & 15/40 \\ 1/12 & 1/8 \\ 2/3 & 1/2 \end{pmatrix}, p(c|b) = \begin{pmatrix} 1/3 & 1/2 & 15/40 \\ 2/3 & 1/2 & 5/8 \end{pmatrix}$$

For these tables, $a \top\!\!\top b$, $b \top\!\!\top c$, but $a \perp\!\!\!\perp c$.

- Local dependence does not guarantee dependence of path-connected variables.
- Graphical independence $\rightarrow$ distribution independence.
- Graphical dependence $\nrightarrow$ distribution dependence.
- The moral of the story is that graphical models cannot generally enforce distributions to obey the dependencies implied by the graph.

Factor Graphs

A square node represents a factor (non negative function) of its neighbouring variables.

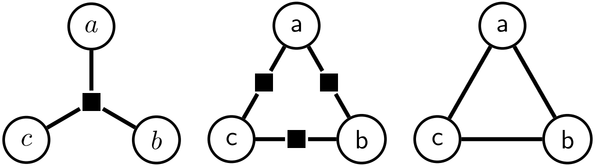


The joint function is the product of all factors:

$$f(A,B,C,D,E) = f_1(A,B)f_2(B,C,D)f_3(C,E)f_4(D,E)$$

Factor graphs are useful for performing efficient computations (not just for probability).

Factor Graphs versus Markov Networks



(a) (b) (c)

a $\phi(a, b, c)$

b $\phi(a, b)\phi(b, c)\phi(c, a)$

c $\phi(a, b, c)$

- Both (a) and (b) have the same Markov network (c).
- Whilst (b) contains the same (lack of) independence statements as (a), it expresses more constraints on the form of the potential.