CSC 412 (Lecture 4): Undirected Graphical Models

Raquel Urtasun

University of Toronto

Feb 2, 2016

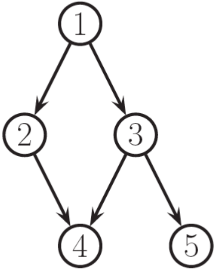
Today

Undirected Graphical Models:

- Semantics of the graph: conditional independence
- Parameterization
  - Clique
  - Potentials
  - Gibbs Distribution
  - Partition function
  - Hammersley-Clifford Theorem
- Factor Graphs
- Learning

Directed Graphical Models

- Represent large joint distribution using "local" relationships specified by the graph
- Each random variable is a node
- The edges specify the statistical dependencies
- We have seen directed acyclic graphs



Directed Acyclic Graphs

- Represent distribution of the form

$$p(y_1, \cdots, y_N) = \prod_i p(y_i | y_{\pi_i})$$

with π_i the parents of the node i

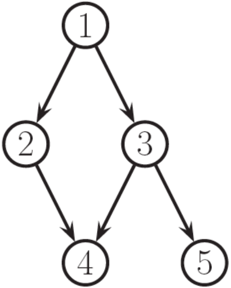
- Factorizes in terms of local conditional probabilities
- Each node has to maintain $p(y_i | y_{\pi_i})$
- Each variable is CI of its non-descendants given its parents

$$\{y_i \perp y_{\tilde{\pi}_i} | y_{\pi_i}\} \qquad \forall i$$

with $y_{\tilde{\pi}_i}$ the nodes before y_i that are not its parents

- Such an ordering is a "topological" ordering (i.e., parents have lower numbers than their children)
- Missing edges imply conditional independence

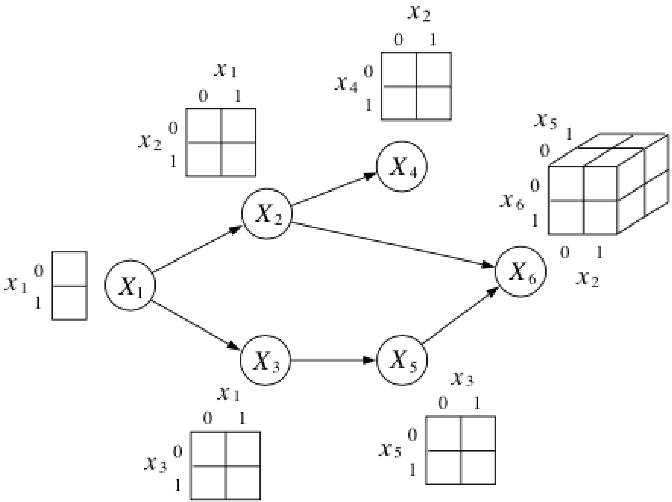
Example



What's the joint probability distribution?

Internal Representation

- For discrete variables, each node stores a conditional probability table (CPT)



Are DGM Always Useful?

- Not always clear how to choose the direction for the edges
- Example: Modeling dependencies in an image

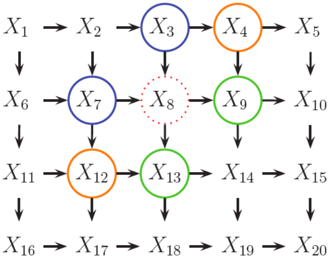
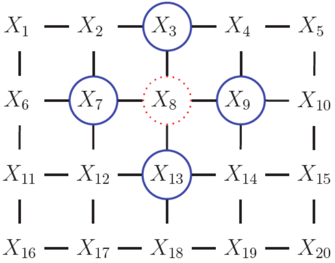


Figure : Causal MRF or a Markov mesh

- Unnatural conditional independence, e.g., see Markov Blanket $mb(8) = \{3, 7\} \cup \{9, 13\} \cup \{12, 4\}$, parents, children and co-parents
- Alternative: Undirected Graphical models (UGMs)

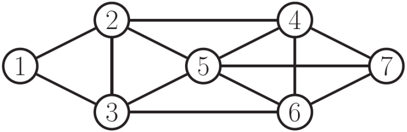
Undirected Graphical Models

- Also called Markov random field (MRF) or Markov network
- As in DGM, the nodes in the graph represent the variables
- Edges represent probabilistic interaction between neighboring variables
- How to parametrize the graph?
  - In DGM we used CPD (conditional probabilities) to represent distribution of a node given others
  - For undirected graphs, we use a more symmetric parameterization that captures the affinities between related variables.



Semantics of the Graph: Conditional Independence

- Global Markov Property: $x_A \perp x_B | x_C$ iff C separates A from B (no path in the graph), e.g., $\{1, 2\} \perp \{6, 7\} | \{3, 4, 5\}$



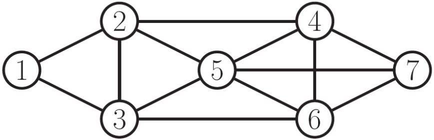
- Markov Blanket (local property) is the set of nodes that renders a node t conditionally independent of all the other nodes in the graph

$$t \perp \mathcal{V} \setminus cl(t) | mb(t)$$

where $cl(t) = mb(t) \cup t$ is the closure of node t. It is the set of neighbors, e.g., $mb(5) = \{2, 3, 4, 6, 7\}$.

- Pairwise Markov Property

$$s \perp t | \mathcal{V} \setminus \{s, t\} \iff G_{st} = 0$$

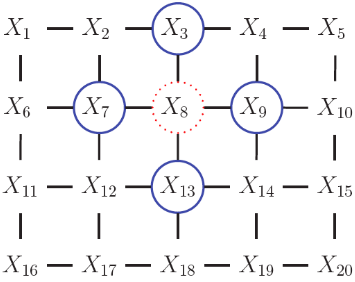


- Pairwise: $1 \perp 7 | \text{rest}$
- Local: $1 \perp \text{rest} | 2, 3$
- Global: $1, 2 \perp 6, 7 | 3, 4, 5$

$$\begin{tikzcd} G \arrow[r, Rightarrow] & L \arrow[r, Rightarrow] & P \arrow[ll, Rightarrow, bend left, "p(x) > 0"] \end{tikzcd}$$

→ See page 119 of Koller and Friedman for a proof

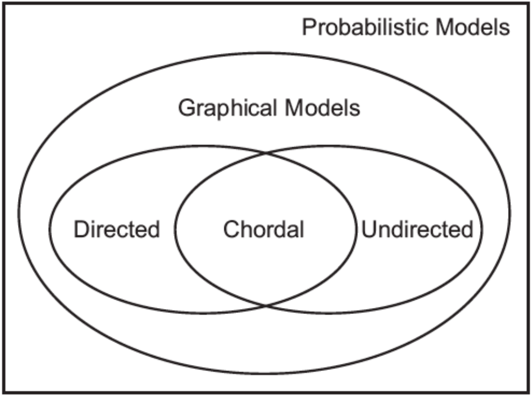
Image Example



Complete the following statements:

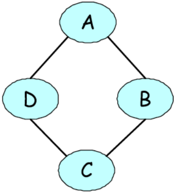
- Pairwise: $1 \perp 7 |$rest?, $1 \perp 20|$rest?,$1 \perp 2|$rest?
- Local: $1 \perp$ rest$|$?, $8 \perp$ rest$|$?
- Global: $1, 2 \perp 15, 20|$?

DGM and UGM

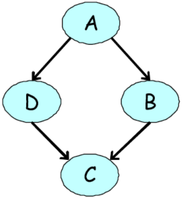


- From Directed to Undirected via moralization
- From Undirected to Directed via triangulation
- See (Kohler and Friedman) book if interested

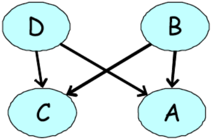
Not all UGM can be represented as DGM



(a)



(b)

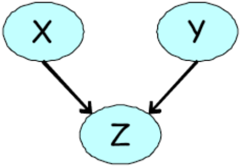


(c)

- Fig. (a) Two independencies: $(A \perp C | D, B)$ and $(B \perp D | A, C)$
- Can we encode this with a DGM?
- Fig. (b) First attempt: encodes $(A \perp C | D, B)$ but it also implies that $(B \perp D | A)$ but dependent given both A, C
- Fig. (c) Second attempt: encodes $(A \perp C | D, B)$, but also implies that B and D are marginally independent.

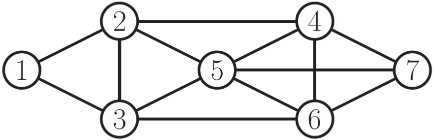
Not all DGM can be represented as UGM

- Example is the V-structure



- Undirected model fails to capture the marginal independence $(X \perp Y)$ that holds in the directed model at the same time as $\neg(X \perp Y|Z)$

Cliques



- A clique in an undirected graph is a subset of its vertices such that every two vertices in the subset are connected by an edge
  $\rightarrow$ i.e., the subgraph induced by the clique is complete
- The maximal clique is a clique that cannot be extended by including one more adjacent vertex
- The maximum clique is a clique of the largest possible size in a given graph
- What are the maximal cliques? And the maximum clique in the figure?

Parameterization of an UGM

- $\mathbf{y} = (y_1, \cdots, y_m)$ the set of all random variables
- Unlike DGM, since there is no topological ordering associated with an undirected graph, we can't use the chain rule to represent $p(\mathbf{y})$
- Instead of associating conditional probabilities to each node, we associate potential functions or factors with each maximal clique in the graph
- For a clique c, we define the potential function or factor

$$\psi_c(\mathbf{y}_c|\theta_c)$$

  to be any non-negative function, with $\mathbf{y}_c$ the restriction to a subset of variables in $\mathbf{y}$
- The joint distribution is then proportional to the product of clique potentials
- Any positive distribution whose CI are represented with an UGM can be represented this way (let's see this more formally)

Factor Parameterization

Theorem (Hammersley-Clifford)

A positive distribution $p(\mathbf{y}) > 0$ satisfies the CI properties of an undirected graph G iff p can be represented as a product of factors, one per maximal clique, i.e.,

$$p(\mathbf{y}|\theta) = \frac{1}{Z(\theta)} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\theta_c)$$

with $\mathcal{C}$ the set of all (maximal) cliques of G, and $Z(\theta)$ the ***partition function*** *defined as*

$$Z(\theta) = \sum_{\mathbf{y}} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\theta_c)$$

Proof.

Can be found in (Koller and Friedman book) □

We need the partition function as the potentials are not conditional distributions.

In DGMs we don't need it

The partition function

The joint distribution is

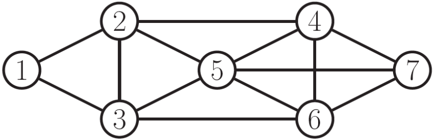
$$p(\mathbf{y}|\theta) = \frac{1}{Z(\theta)} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\theta_c)$$

with the partition function

$$Z(\theta) = \sum_{\mathbf{y}} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\theta_c)$$

- This is the hardest part of learning and inference. Why?
- Factored structure of the distribution makes it possible to more efficiently do the sums/integrals needed to compute it.

Example



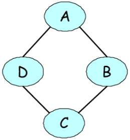
$$p(\mathbf{y}) \propto \psi_{1,2,3}(y_1, y_2, y_3)\psi_{2,3,5}(y_2, y_3, y_5)\psi_{2,4,5}(y_2, y_4, y_5) \psi_{3,5,6}(y_3, y_5, y_6)\psi_{4,5,6,7}(y_4, y_5, y_6, y_7)$$

- Is this representation unique?
- What if I want a pairwise MRF?

Representing Potentials

- If the variables are discrete, we can represent the potential or energy functions as tables of (non-negative) numbers

$$p(A, B, C, D) = \frac{1}{Z}\psi_{a,b}(A, B)\psi_{b,c}(B, C)\psi_{c,d}(C, D)\psi_{a,d}(A, D)$$



$\phi_1[A,B]$			$\phi_2[B,C]$			$\phi_3[C,D]$			$\phi_4[D,A]$		
a^0	b^0	30	b^0	c^0	100	c^0	d^0	1	d^0	a^0	100
a^0	b^1	5	b^0	c^1	1	c^0	d^1	100	d^0	a^1	1
a^1	b^0	1	b^1	c^0	1	c^1	d^0	100	d^1	a^0	1
a^1	b^1	10	b^1	c^1	100	c^1	d^1	1	d^1	a^1	100

- The potentials are NOT probabilities
- They represent compatibility between the different assignments

Factor product

- Given 3 disjoint set of variables $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$, and factors $\psi_1(\mathbf{X}, \mathbf{Y})$, $\psi_2(\mathbf{Y}, \mathbf{Z})$, the factor product is defined as

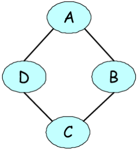
$$\psi_{x,y,z}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = \psi_{x,y}(\mathbf{X}, \mathbf{Y})\phi_{y,z}(\mathbf{Y}, \mathbf{Z})$$

a^1	b^1	0.5
a^1	b^2	0.8
a^2	b^1	0.1
a^2	b^2	0
a^3	b^1	0.3
a^3	b^2	0.9

b^1	c^1	0.5
b^1	c^2	0.7
b^2	c^1	0.1
b^2	c^2	0.2

a^1	b^1	c^1	0.5·0.5
a^1	b^1	c^2	0.5·0.7
a^1	b^2	c^1	0.8·0.1
a^1	b^2	c^2	0.8·0.2
a^2	b^1	c^1	0.1·0.5
a^2	b^1	c^2	0.1·0.7
a^2	b^2	c^1	0·0.1
a^2	b^2	c^2	0·0.2
a^3	b^1	c^1	0.3·0.5
a^3	b^1	c^2	0.3·0.7
a^3	b^2	c^1	0.9·0.1
a^3	b^2	c^2	0.9·0.2

Query about probabilities: marginalization



$\phi_1[A,B]$			$\phi_2[B,C]$			$\phi_3[C,D]$			$\phi_4[D,A]$		
a^0	b^0	30	b^0	c^0	100	c^0	d^0	1	d^0	a^0	100
a^0	b^1	5	b^0	c^1	1	c^0	d^1	100	d^0	a^1	1
a^1	b^0	1	b^1	c^0	1	c^1	d^0	100	d^1	a^0	1
a^1	b^1	10	b^1	c^1	100	c^1	d^1	1	d^1	a^1	100

Assignment				*Unnormalized*	*Normalized*
a^0	b^0	c^0	d^0	300000	0.04
a^0	b^0	c^0	d^1	300000	0.04
a^0	b^0	c^1	d^0	300000	0.04
a^0	b^0	c^1	d^1	30	$4.1 \cdot 10^{-6}$
a^0	b^1	c^0	d^0	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^0	d^1	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^1	d^0	5000000	0.69
a^0	b^1	c^1	d^1	500	$6.9 \cdot 10^{-5}$
a^1	b^0	c^0	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^0	d^1	1000000	0.14
a^1	b^0	c^1	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^1	d^1	100	$1.4 \cdot 10^{-5}$
a^1	b^1	c^0	d^0	10	$1.4 \cdot 10^{-6}$
a^1	b^1	c^0	d^1	100000	0.014
a^1	b^1	c^1	d^0	100000	0.014
a^1	b^1	c^1	d^1	100000	0.014

- What's the $p(b^0)$? Marginalize the other variables!

Query about probabilities: conditioning

a^1	b^1	0.5
a^1	b^2	0.8
a^2	b^1	0.1
a^2	b^2	0
a^3	b^1	0.3
a^3	b^2	0.9

b^1	c^1	0.5
b^1	c^2	0.7
b^2	c^1	0.1
b^2	c^2	0.2

a^1	b^1	c^1	0.5·0.5
a^1	b^1	c^2	0.5·0.7
a^1	b^2	c^1	0.8·0.1
a^1	b^2	c^2	0.8·0.2
a^2	b^1	c^1	0.1·0.5
a^2	b^1	c^2	0.1·0.7
a^2	b^2	c^1	0·0.1
a^2	b^2	c^2	0·0.2
a^3	b^1	c^1	0.3·0.5
a^3	b^1	c^2	0.3·0.7
a^3	b^2	c^1	0.9·0.1
a^3	b^2	c^2	0.9·0.2

(Original)

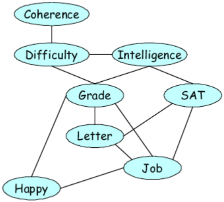
a^1	b^1	c^1	0.25
a^1	b^2	c^1	0.08
a^2	b^1	c^1	0.05
a^2	b^2	c^1	0
a^3	b^1	c^1	0.15
a^3	b^2	c^1	0.09

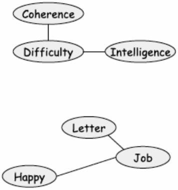
(Cond. on c^1)

- Conditioning on an assignment $\mathbf{u}$ to a subset of variables $\mathbf{U}$ can be done by
  1. Eliminating all entries that are inconsistent with the assignment
  2. Re-normalizing the remaining entries so that they sum to 1

Reduced Markov Networks

- Let $\mathcal{H}$ be a Markov network over $\mathbf{X}$ and let $\mathbf{U} = u$ be the context. The reduced network $\mathcal{H}[u]$ is a Markov network over the nodes $\mathbf{W} = \mathbf{X} - \mathbf{U}$ where we have an edge between X and Y if there is an edge between then in $\mathcal{H}$





- If $\mathbf{U} = \mathit{Grade}$?
- If $\mathbf{U} = \{\mathit{Grade}, \mathit{SAT}\}$?

Connections to Statistical Physics

- The Gibbs Distribution is defined as

$$p(\mathbf{y}|\theta) = \frac{1}{Z(\theta)} \exp\left(-\sum_c E(\mathbf{y}_c|\theta_c)\right)$$

where $E(\mathbf{y}_c) > 0$ is the energy associated with the variables in clique c

- We can convert this distribution to a UGM by

$$\psi(\mathbf{y}_c|\theta_c) = \exp(-E(\mathbf{y}_c|\theta_c))$$

- High probability states correspond to low energy configurations.
- These models are named ~~energy based models~~

Log Linear Models

- Represent the log potentials as a linear function of the parameters

$$\log \psi_c(\mathbf{y}_c) = \phi_c(\mathbf{y}_c)^T \theta_c$$

- The log probability is then

$$\log p(\mathbf{y}|\theta) = \sum_c \phi_c(\mathbf{y}_c)^T \theta_c - \log Z(\theta)$$

- This is called log linear model
- Example: we can represent tabular potentials

$$\psi(y_s = j, y_t = k) = \exp([\theta_{st}^T \phi_{st}]_{jk}) = \exp(\theta_{st}(j,k))$$

with $\phi_{st}(y_s, y_t) = [\cdots, I(y_s = j, y_t = k), \cdots)$ and I the indicator function

Example: Ising model

- Captures the energy of a set of interacting atoms.
- $y_i \in \{-1, +1\}$ represents direction of the atom spin.
- The graph is a 2D or 3D lattice, and the energy of the edges is symmetric

$$\psi_{st}(y_s, y_t) = \begin{pmatrix} e^{w_{st}} & e^{-w_{st}} \\ e^{-w_{st}} & e^{w_{st}} \end{pmatrix}$$

  with w_{st} the coupling strength between two nodes. If not connected $w_{st} = 0$
- Often we assume all edges have the same strength, i.e., $w_{st} = J \neq 0$
- If all weights positive, then neighboring spins likely same spin (ferromagnets, associative Markov network)
- If weights are very strong, then two models, all $+1$ and all -1
- If weights negative, then anti-ferromagnets. Not all the constraints can be satisfied, and the prob. distribution has multiple modes
- Also individual node potentials that encode the bias of the individual atoms (i.e., external field)

More on Ising Models

- Captures the energy of a set of interacting atoms.
- $y_i \in \{-1, +1\}$ represents direction of the atom spin.
- The energy associated is

$$P(\mathbf{y}) = \frac{1}{Z} \exp\left(\sum_{i,j} \frac{1}{2} w_{i,j} y_i y_j + \sum_i b_i y_i \right) = \frac{1}{Z} \exp\left(\frac{1}{2} \mathbf{y}^T \mathbf{W} \mathbf{y} + \mathbf{b}^T \mathbf{y} \right)$$

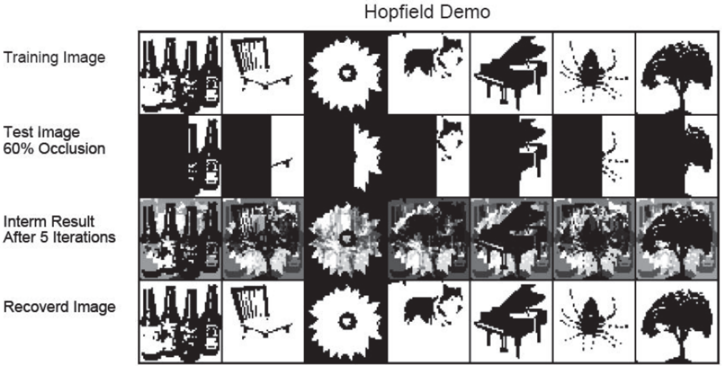
- The energy can be written as

$$E(\mathbf{y}) = -\frac{1}{2} (\mathbf{y} - \boldsymbol{\mu})^T \mathbf{W} (\mathbf{y} - \boldsymbol{\mu}) + c$$

  with $\boldsymbol{\mu} = -\mathbf{W}^{-1}\mathbf{u}, \quad c = \frac{1}{2} \boldsymbol{\mu}^T \mathbf{W} \boldsymbol{\mu}$
- Looks like a Gaussian... but is it?
- Often modulated by a temperature $p(\mathbf{y}) = \frac{1}{Z} \exp(-E(\mathbf{y})/T)$
- T small makes distribution picky

Example: Hopfield networks

- A Hopfield network is a fully connected Ising model with a symmetric weight matrix $\mathbf{W} = \mathbf{W}^T$
- The main application of Hopfield networks is as an associative memory



Example: Potts Model

- Multiple discrete states $y_i \in \{1, 2, \cdots, K\}$
- Common to use

$$\psi_{st}(y_s, y_t) = \begin{pmatrix} e^J & 0 & 0 \\ 0 & e^J & 0 \\ 0 & 0 & e^J \end{pmatrix}$$

- If $J > 0$ neighbors encourage to have the same label
- Phase transition: change of behavior, $J = 1.44$ in example

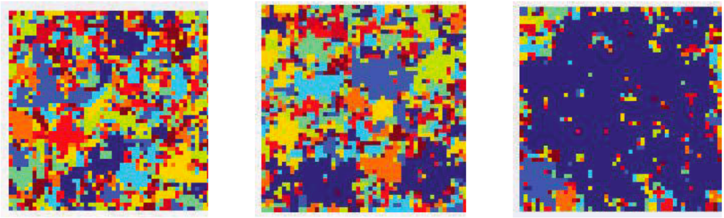
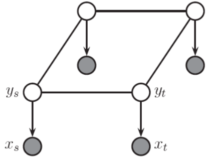


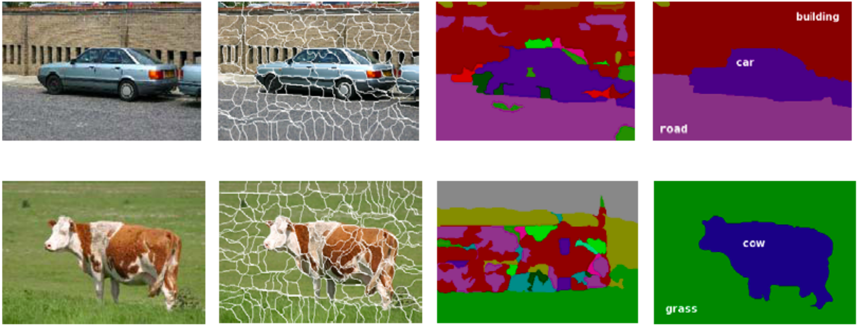
Figure : Sample from a 10-state Potts model of size 128×128 for (a) J = 1.42, (b) J = 1.44, (c) J = 1.46

More on Potts

- Used in image segmentation: neighboring pixels are likely to have the same ... hence belong to the same segment



$$p(\mathbf{y}|\mathbf{x},\theta) = \frac{1}{Z}\prod_i \psi_i(y_i|\mathbf{x}) \prod_{i,j} \psi_{i,j}(y_i, y_j)$$



Example: Gaussian MRF

- Is a pairwise MRF

$$
\begin{aligned}
p(\mathbf{y}|\theta) &\propto \prod_{s\sim t} \psi_{st}(y_s, y_t) \prod_t \psi_t(y_t) \\
\psi_{st}(y_s, y_t) &= \exp\left(-\frac{1}{2} y_s \Lambda_{st} y_t\right) \\
\psi_t(y_t) &= \exp\left(-\frac{1}{2}\Lambda_{tt} y_t^2 + \eta_t y_t\right)
\end{aligned}
$$

- The joint distribution is then

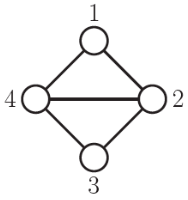
$$
p(\mathbf{y}|\theta) \propto \exp\left[\eta^T \mathbf{y} - \frac{1}{2}\mathbf{y}^T \Lambda \mathbf{y}\right]
$$

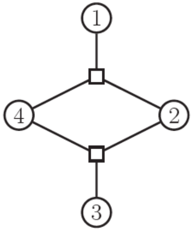
- This is a multivariate Gaussian with $\Lambda = \Sigma^{-1}$ and $\eta = \Lambda\mu$
- If $\Lambda_{st} = 0$ (structural zero), then no pairwise connection and by factorization theorem

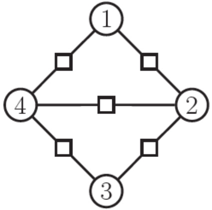
$$
y_s \perp y_t | \mathbf{y}_{-(st)} \iff \Lambda_{st} = 0
$$

- UGM are sparse precision matrices. Used for structured learning

Factor Graphs





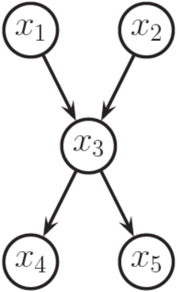


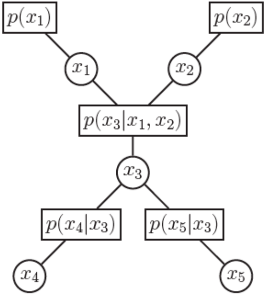
- A factor graph is a graphical model representation that unifies directed and undirected models
- It is an undirected bipartite graph with two kinds of nodes.
    - Round nodes represent variables,
    - Square nodes represent factors

  and there is an edge from each variable to every factor that mentions it.
- Represents the distribution more uniquely than a graphical model

Factor Graphs for Directed Models

- One factor per CPD (conditional distribution) and connect the factor to all the variables that use the CPD





Learning using Gradient methods

- MRF in log-linear form

$$p(\mathbf{y}|\theta) = \frac{1}{Z(\theta)} \exp\left(\sum_c \theta_c^T \phi_c(\mathbf{y}_c)\right)$$

- Given training examples $\mathbf{y}^{(i)}$, the scaled log likelihood is

$$\ell(\theta) = -\frac{1}{N}\sum_i \log p(\mathbf{y}^{(i)}|\theta) = \frac{1}{N}\sum_i \left[-\sum_c \theta_c^T \phi_c(\mathbf{y}_c^{(i)}) + \log Z^{(i)}(\theta)\right]$$

- Since MRFs are in the exponential family, this function is convex in θ
- We can find the global maximum, e.g., via gradient descent

$$\frac{\partial \ell}{\partial \theta_c} = \frac{1}{N}\sum_i \left[-\phi_c(\mathbf{y}_c^{(i)}) + \frac{\partial}{\partial \theta_c} \log Z^{(i)}(\theta)\right]$$

- The first term is constant for each iteration of gradient descent, it is called the ~~empirical means~~

Moment Matching

$$\frac{\partial \ell}{\partial \theta_c} = \frac{1}{N} \sum_i \left[-\phi_c(\mathbf{y}_c^{(i)}) + \frac{\partial}{\partial \theta_c} \log Z^{(i)}(\theta) \right]$$

- The derivative of the log partition function w.r.t. θ_c is the expectation of the c'th feature under the model

$$\frac{\partial \log Z(\theta)}{\partial \theta_c} = \sum_{\mathbf{y}} \phi_c(\mathbf{y}) p(\mathbf{y}|\theta) = E[\phi_c(\mathbf{y})]$$

  Thus the gradient of the log likelihood is

$$\frac{\partial \ell}{\partial \theta_c} = \left[-\frac{1}{N} \sum_i \phi_c(\mathbf{y}_c^{(i)}) \right] + E[\phi_c(\mathbf{y})]$$

- The second term is the contrastive term or unclamped term and requires ~~inference~~ in the model (it has to be done for each step in gradient descent)
- Dif. of the empirical distrib. and model's expectation of the feature vector

$$\frac{\partial \ell}{\partial \theta_c} = -E_{p_{emp}}[\phi_c(\mathbf{y})] + E_{p(\cdot|\theta)}[\phi_c(\mathbf{y})]$$

- At the optimum the moments are matched (i.e., moment matching)

Approximated Methods

- In UGM, no closed form solution to the ML estimate of the parameters, need to do gradient-based optimization
- Computing each gradient step requires inference $\rightarrow$ very expensive (NP-hard in general)
- Many approximations exist: stochastic approaches, pseudo likelihood, etc