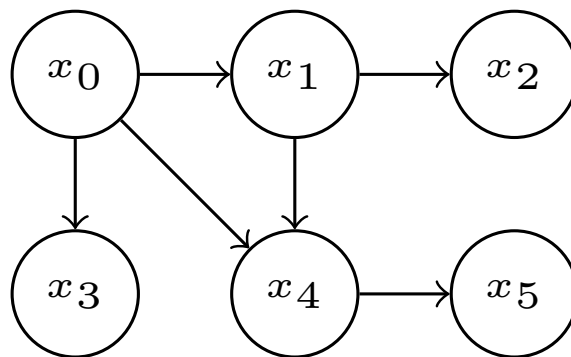


Chapter 16

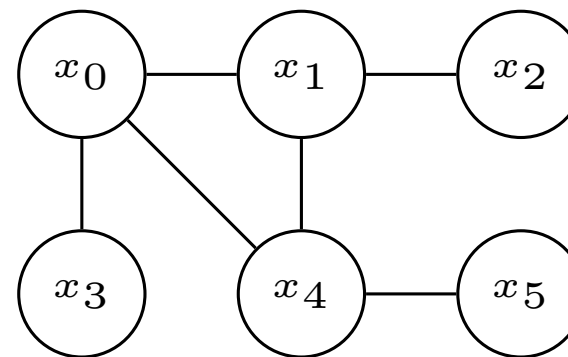
Structured Probabilistic Models for Deep Learning

Structured Probabilistic Models

- A way of using graphs to describe a probability distribution with an emphasis on visualizing which random variables interact with each other directly
 - Each node represents a random variable
 - Each edge represents a direct interaction



Directed models (Bayesian Nets)



Undirected models (Markov Nets)

- Also known as **probabilistic graphical models**, or **graphical models**

Learning, Sampling, and Inference

- Things we will be concerned with around the graphical models
 - Learning the model structure $p(\mathbf{x})$ and parameters θ

$$\theta^* = \arg \max_{\theta} p(\mathbf{x}; \theta)$$

- Drawing samples from the learned model

$$\mathbf{x} \sim p(\mathbf{x}; \theta^*) \text{ or } \mathbf{x}_2 \sim p(\mathbf{x}_2 | \mathbf{x}_1; \theta^*)$$

- Doing approximate or exact inference

$$\arg \max_{\mathbf{x}_2} p(\mathbf{x}_2 | \mathbf{x}_1; \theta^*) \approx \arg \max_{\mathbf{x}_2} q(\mathbf{x}_2 | \mathbf{x}_1; \mathbf{w})$$

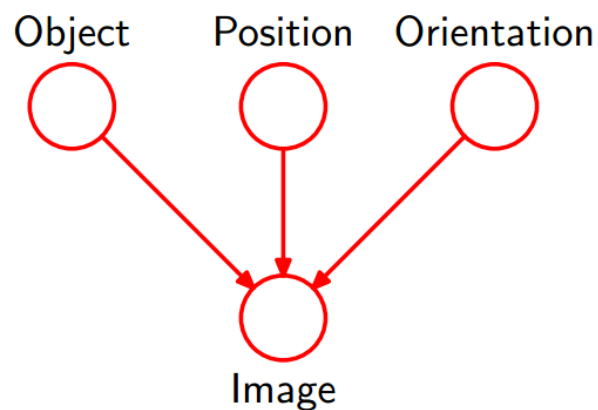
Directed Graphical Models

- A directed model defined on \mathbf{x} is specified by
 1. A directed acyclic graph \mathcal{G} with nodes denoting elements x_i of \mathbf{x}
 2. A set of local conditional probability distributions $p(x_i | Pa_{\mathcal{G}}(x_i))$ with $Pa_{\mathcal{G}}(x_i)$ giving the parent nodes of x_i in \mathcal{G}and factorizes the joint distribution of the node variables as

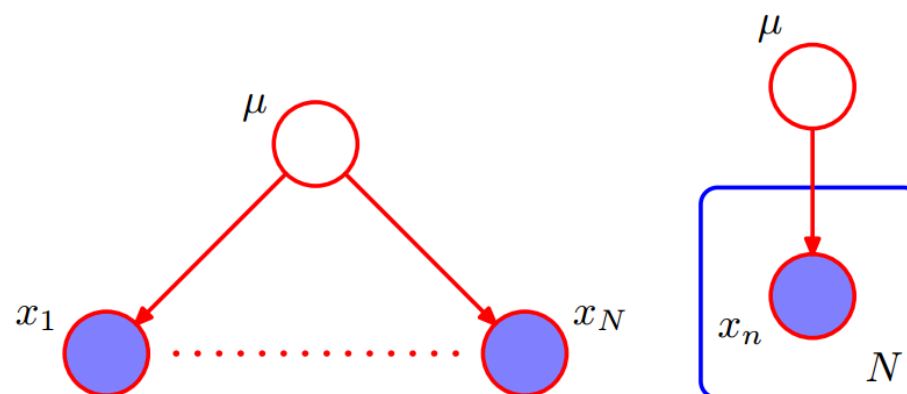
$$p(\mathbf{x}) = \prod_i p(x_i | Pa_{\mathcal{G}}(x_i))$$

- Such graphical models are also known as **Bayesian/belief networks**

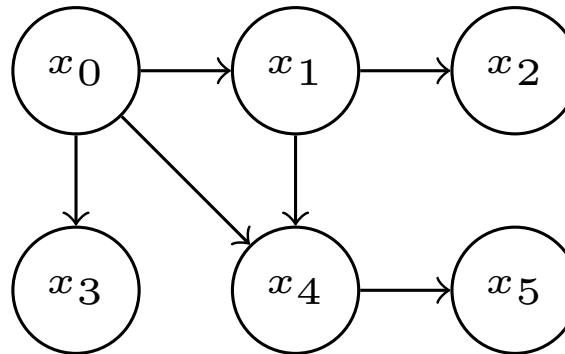
- They are most naturally applicable in situations where there is **clear causality between variables**



- For convenience, we sometimes introduce **plate** notation



- As an example, we have for the following graph



$$p(x_0, x_1, x_2, x_3, x_4, x_5) = p(x_0)p(x_1|x_0)p(x_2|x_1)p(x_3|x_0) \\ p(x_4|x_1, x_0)p(x_5|x_4)$$

- When compared to the chain rule of probability,

$$p(\mathbf{x}) = \prod_{i=0} p(x_i|x_{i-1}, x_{i-2}, \dots, x_0),$$

the graph factorization implies certain conditional independence, e.g.

$$p(x_2|x_1, x_0) = p(x_2|x_1)$$

$$p(x_3|x_2, x_1, x_0) = p(x_3|x_0)$$

- Note however it only specifies which variables are allowed to appear in the arguments; there is **no constraint on how we define each conditional probability distribution**
- In the present example, we may as well specify

$$p(x_1|x_0) = f_1(x_1, x_0) = p(x_1)$$

$$p(x_2|x_1) = f_2(x_2, x_1) = p(x_2)$$

$$p(x_3|x_0) = f_3(x_3, x_0) = p(x_3)$$

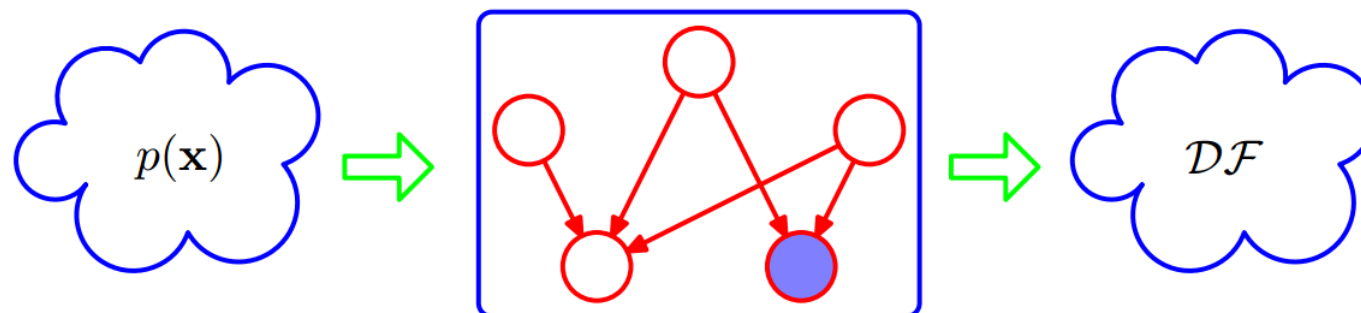
$$p(x_4|x_1, x_0) = f_4(x_4, x_1, x_0) = p(x_4)$$

$$p(x_5|x_4) = f_5(x_5, x_4) = p(x_5)$$

to arrive at a fully factorized distribution

$$p(x_0, x_1, x_2, x_3, x_4, x_5) = p(x_0)p(x_1)p(x_2)p(x_3)p(x_4)p(x_5)$$

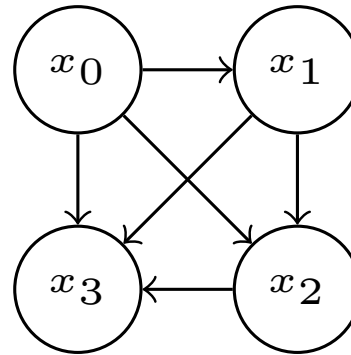
- As such, there could be several distributions that satisfy the graph factorization; it is helpful to think of a directed graph as a filter



where \mathcal{DF} denotes the set of distributions that satisfy the factorization described by the graph

- To be precise, for any given graph, the \mathcal{DF} will include any distributions that have additional independence properties beyond those described by the graph

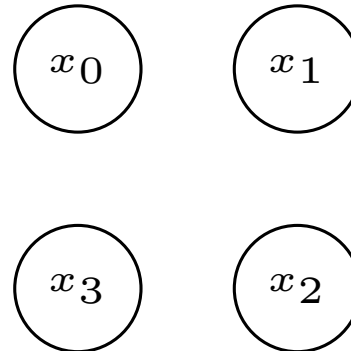
- **Extreme case I:** A fully connected graph will accept any possible distribution over the given variables



$$p(x_0, x_1, x_2, x_3) = p(x_0)p(x_1|x_0)p(x_2|x_1, x_0)p(x_3|x_2, x_1, x_0)$$

(simply the chain rule of probability)

- **Extreme case II:** A fully disconnected graph will only accept a fully factorized distribution



$$p(x_0, x_1, x_2, x_3) = p(x_0)p(x_1)p(x_2)p(x_3)$$

- It is also straightforward to see that a fully factorized distribution will pass through any graph

- In general, to model n discrete variables each having k values, we need a table of size $\mathcal{O}(k^n)$; the conditional independence implied by the graph can reduce the table size to $\mathcal{O}(k^m)$, given m is the maximum number of conditioning variables for all x_i
- This suggests that as long as each variable has few parents in the graph, the distribution can be represented with very few parameters

Undirected Graphical Models

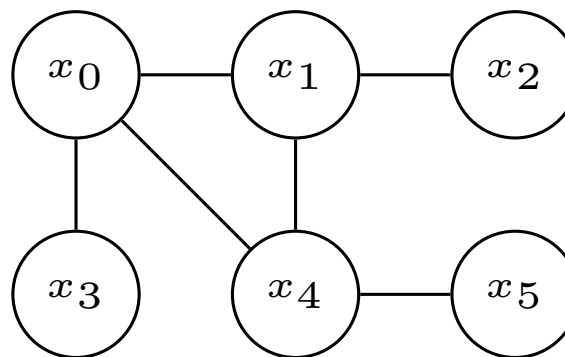
- An undirected graphical model is defined on an undirected graph \mathcal{G} and factorizes the joint distribution of its node variables as a product of potential functions $\phi(\mathcal{C})$ over the maximum cliques \mathcal{C} of the graph

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{\mathcal{C} \in \mathcal{G}} \phi(\mathcal{C}) = \frac{1}{Z} \tilde{p}(\mathbf{x})$$

where

- $\tilde{p}(\mathbf{x})$ is an unnormalized distribution
 - Z is a normalization constant (called the partition function)
 - $\phi(\mathcal{C})$ is a clique potential and is non-negative
- They are also known as **Markov random fields** or **Markov networks**

- A **clique** is a subset of the nodes in a graph \mathcal{G} in which there exists a link between every pair of nodes in the subset
- A **maximum clique** \mathcal{C} is a clique such that it is not possible to include any other nodes in the graph without ceasing to be a clique
- As an example, we have for the following graph



$$p(\mathbf{x}) = \frac{1}{Z} \phi_a(x_0, x_3) \phi_b(x_0, x_1, x_4) \phi_c(x_1, x_2) \phi_d(x_4, x_5)$$

- The **clique potential ϕ** measures the affinity of its member variables in each of their possible joint states
- One choice for ϕ is the energy-based model (**Boltzmann distribution**)

$$\phi(\mathcal{C}) = \exp(-E(\mathbf{x}_{\mathcal{C}}))$$

where $\mathbf{x}_{\mathcal{C}}$ denote the variables in that clique

- The choice of ϕ needs some attention; **not every choice would result in a legitimate probability distribution**, e.g.,

$$\phi(x) = \exp(-\beta x^2)$$

with $x \in \mathbb{R}$ and $\beta < 0$

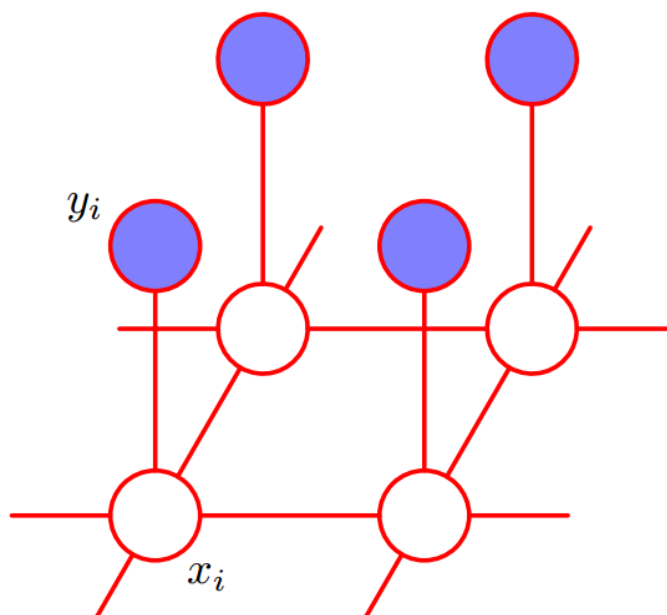
- In the present case, the unnormalized joint distribution is also a Boltzmann distribution with a total energy given by the sum of the

energies of all the maximum cliques

$$\tilde{p}(\mathbf{x}) = \exp(-E(\mathbf{x})), \text{ with } E(\mathbf{x}) = \sum_{\mathcal{C} \in \mathcal{G}} E(\mathbf{x}_{\mathcal{C}})$$

- Each energy term imposes a particular soft constraint on the variables

Example: Image de-noising



- $y_i \in \{-1, +1\}$: Observed image pixels
- $x_i \in \{-1, +1\}$: Hidden noise-free image pixels

- The maximum cliques of the graph are seen to be

$$\{x_i, y_i\}, \{x_i, x_j\}$$

- The joint distribution is given by

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp(-E(\mathbf{x}, \mathbf{y}))$$

- The (complete) energy function is assumed to be

$$\begin{aligned} E(\mathbf{x}, \mathbf{y}) &= \sum_i E(x_i, y_i) + \sum_{i,j} E(x_i, x_j) \\ &= -\eta \sum_i x_i y_i - \beta \sum_{i,j} x_i x_j + h \sum_i x_i \end{aligned}$$

- Z is an (intractable) function of model parameters η , β and h

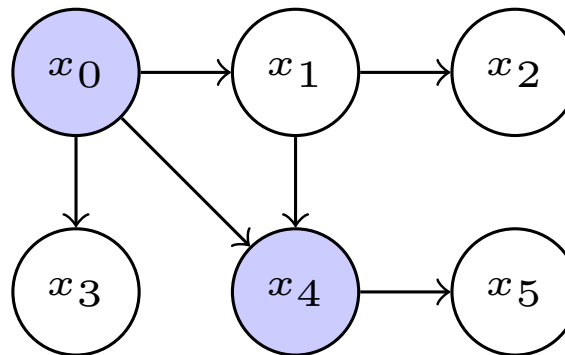
$$Z = \sum_{\mathbf{x}, \mathbf{y}} \exp(-E(\mathbf{x}, \mathbf{y}))$$

- De-noising can be cast as an inference problem

$$\arg \max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y})$$

D-Separation

- We often want to know which subsets of variables are conditionally independent given the values of the other sets of variables



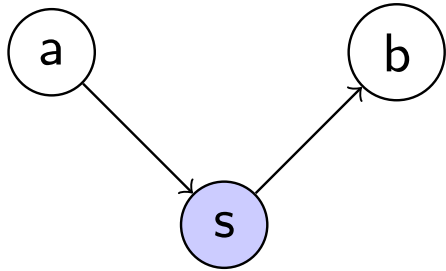
- Is the set of variables $\{x_1, x_2\}$ conditionally independent of the variable x_5 , given the values of $\{x_0, x_4\}$?

$$p(x_1, x_2, x_5 | x_0, x_4) \stackrel{?}{=} p(x_1, x_2 | x_0, x_4) p(x_5 | x_0, x_4),$$

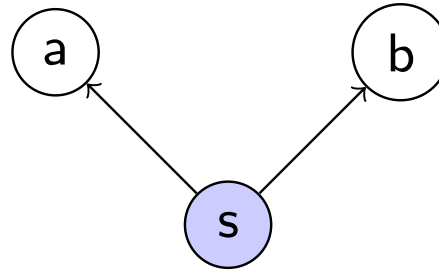
or equivalently,

$$p(x_1, x_2 | x_0, x_4, x_5) \stackrel{?}{=} p(x_1, x_2 | x_0, x_4)$$

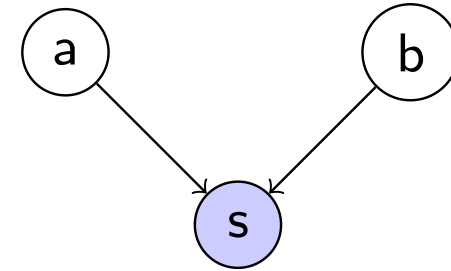
- The key rules can be deduced from observing three simple examples



Head-to-Tail



Tail-to-Tail



Head-to-Head

- **Head-to-Tail:** a and b are **independent** (d-separated) given s

$$p(a, b|s) = \frac{p(a)p(s|a)p(b|s)}{p(s)} = p(a|s)p(b|s)$$

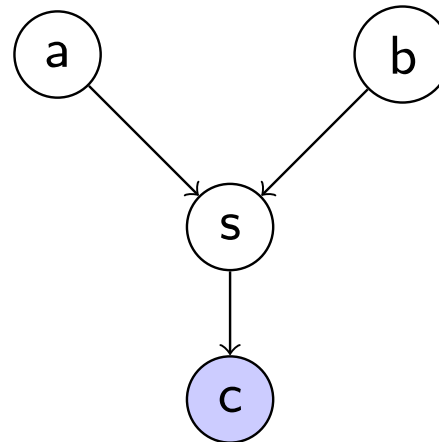
- **Tail-to-Tail:** a and b are **independent** (d-separated) given s

$$p(a, b|s) = \frac{p(s)p(a|s)p(b|s)}{p(s)} = p(a|s)p(b|s)$$

- **Head-to-Head:** a and b are in general **dependent** given s

$$p(a, b|s) = \frac{p(a)p(b)p(s|a, b)}{p(s)} \neq p(a|s)p(b|s)$$

- The head-to-head rule can generalize to the case where a descendant of s is observed

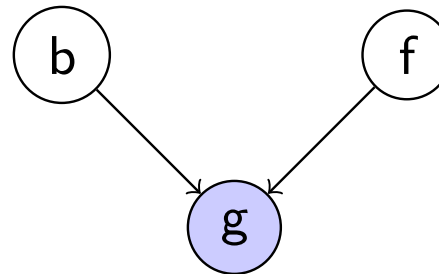


$$p(a, b|c) \neq p(a|c)p(b|c) \text{ in general}$$

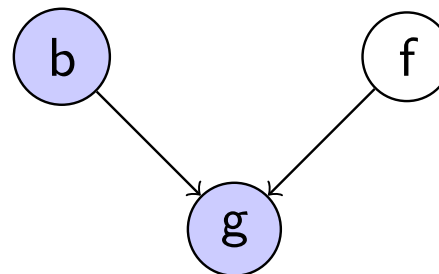
- To summarize, given A, B, C are three non-intersecting sets of nodes, A and B are conditionally independent given C if all paths from any node in A to any node in B satisfy
 - Meeting either head-to-tail or tail-to-tail at a node in C , or
 - Meeting head-to-head at a node, and neither the node, nor any of its descendant, is in C
- In other words, these paths are blocked or inactive
- These rules tell us only the independencies implied by the graph; recall however that not all independencies of a distribution is captured by the graph (c.f. the filter interpretation)

Explaining Away Effects

- A phenomenon associated with the following Bayesian network, where there are two causes b, f which can explain the observation g



- If one of the causes, say b , happens and is observed, the probability that the other cause f also happens will become lower (i.e., the observed cause b **explains away** the possibility of f)



- Example

- $g = 0$: Electric fuel gage reads empty
- $b = 0$: Battery is flat
- $f = 0$: Fuel tank is empty

$$p(b = 1) = 0.9$$

$$p(f = 1) = 0.9$$

$$p(g = 1|b = 1, f = 1) = 0.8$$

$$p(g = 1|b = 1, f = 0) = 0.2$$

$$p(g = 1|b = 0, f = 1) = 0.2$$

$$p(g = 1|b = 0, f = 0) = 0.1$$

- It can be shown that

$$p(f = 0) = 0.1$$

$$p(f = 0|g = 0) \simeq 0.257$$

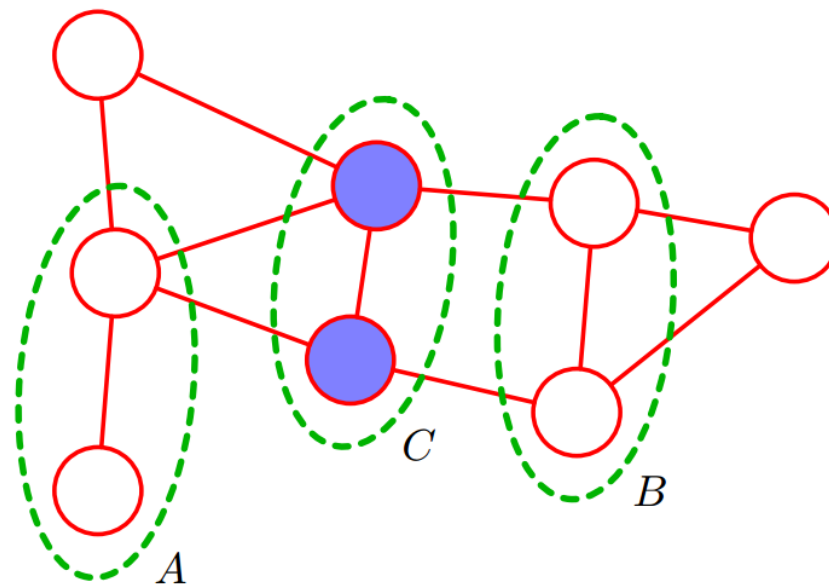
$$p(f = 0|g = 0, \underbrace{b = 0}) \simeq 0.111$$

$$p(f = 0|g = 0, \underbrace{b = 1}) \simeq 0.308$$

- Given that battery is flat (cause 1 happens) and the gage reads empty, the probability of the tank being empty (the other cause happens) decreases from 0.257 to 0.111
- On the other hand, given that battery is not flat (causes 1 does not happen) and the gage reads empty, the probability of the tank being empty (the other cause happens) increases from 0.257 to 0.308

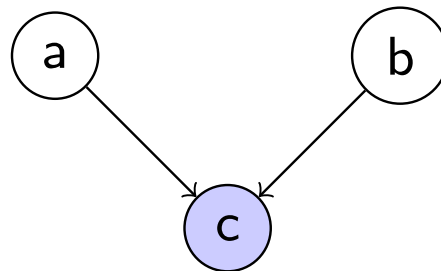
Separation

- Separation refers to the conditional independencies implied by the **undirected graph**
- Given A, B, C are three non-intersecting sets of nodes, A and B are conditionally independent (separated) given C if all paths from any node in A to any node in B pass through one or more nodes in C

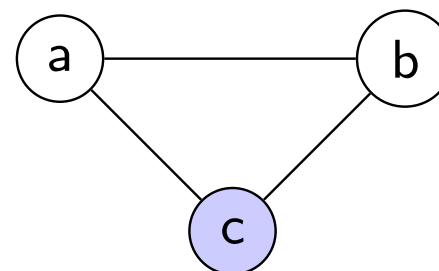


Conversion between Directed and Undirected Models

- Some independencies can be represented by only one of them
- Conversion from a directed model \mathcal{D} to an undirected model \mathcal{U}
 1. Adding an edge to \mathcal{U} for any pair of nodes a, b if there is a directed edge between them in \mathcal{D}
 2. Adding an edge to \mathcal{U} for any pair of nodes a, b if they are both parents of a third node in \mathcal{D}



$a \perp b$ and $a \not\perp b | c$



Moralized graph

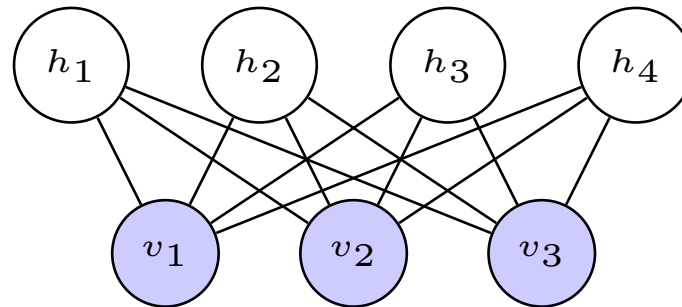
- In the present case, the potential function ϕ is given by

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

- Conversion from an undirected model \mathcal{U} to a directed model \mathcal{U} is much less common, and in general, presents problems due to the normalization constraints (study by yourself)

Restricted Boltzmann Machines (RBM)

- An energy-based model with binary visible and hidden units



$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{b}^T \mathbf{v} - \mathbf{c}^T \mathbf{h} - \mathbf{v}^T \mathbf{W} \mathbf{h}$$

- There is **no direct interaction between visible units or between hidden units** (essentially, a bipartite graph)
- From the separation rules, we have

$$p(\mathbf{h}|\mathbf{v}) = \prod_i p(h_i|\mathbf{v})$$

$$p(\mathbf{v}|\mathbf{h}) = \prod_i p(v_i|\mathbf{h})$$

which are both factorial

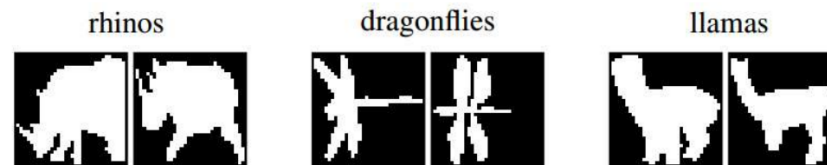
- By the definition of $E(\mathbf{v}, \mathbf{h})$, $p(h_i = 1|\mathbf{v})$ and $p(v_i = 1|\mathbf{h})$ are evaluated to be

$$p(h_i = 1|\mathbf{v}) = \sigma(\mathbf{v}^T \mathbf{W}_{:,i} + c_i)$$

$$p(v_i = 1|\mathbf{h}) = \sigma(\mathbf{W}_{i,:} \mathbf{h} + b_i)$$

- The hidden units \mathbf{h} , although **not interpretable**, denote features that describe visible units \mathbf{v} and can be inferred by $p(h_i = 1|\mathbf{v})$
- Samples of visible units \mathbf{v} can be generated by sampling all of \mathbf{v} given \mathbf{h} and then all of \mathbf{h} given \mathbf{v} via **block Gibbs sampling**

- It is also possible to sample part of v given the values of the others for applications such as image completion (essentially, RBM is a fully probabilistic model)



Training input



Results of image completion

- Estimating the model parameters W, b, c is achieved with the maximum likelihood principle

$$\arg \max_{W, b, c} p(v; W, b, c)$$

where the marginal distribution of visible units is given by

$$p(\mathbf{v}; \mathbf{W}, \mathbf{b}, \mathbf{c}) = \frac{1}{Z} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}))$$

- It is however noticed that the partition function Z is intractable

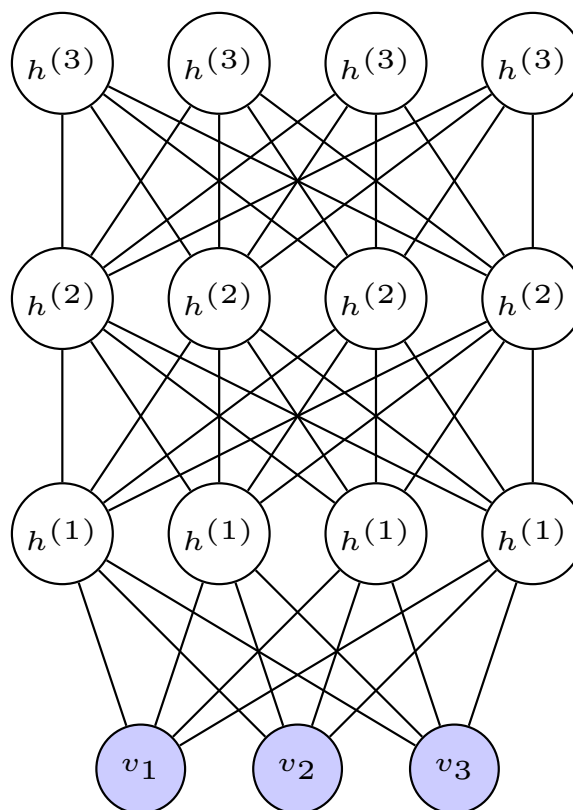
$$Z = \sum_{\mathbf{v}, \mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}))$$

which is a function of the model parameters $\mathbf{W}, \mathbf{b}, \mathbf{c}$

- Some specialized training techniques involving sampling are needed

Deep Boltzmann Machines (DBM)

- Introducing layers of hidden units to RBM



$$E(\mathbf{v}, \mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \mathbf{h}^{(3)}) = -\mathbf{v}^T \mathbf{W}^{(1)} \mathbf{h}^{(1)} - \mathbf{h}^{(1)T} \mathbf{W}^{(2)} \mathbf{h}^{(2)} - \mathbf{h}^{(2)T} \mathbf{W}^{(3)} \mathbf{h}^{(3)}$$

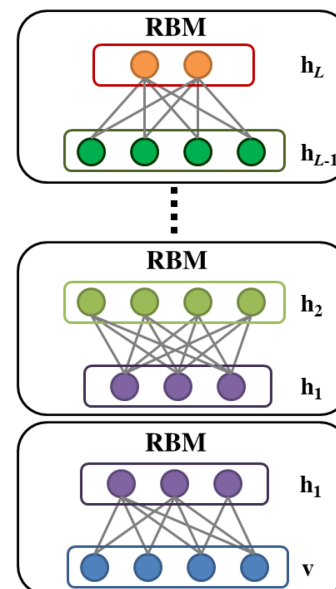
- From the graph, the posterior distribution is no longer factorial

$$p(\mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \mathbf{h}^{(3)} | \mathbf{v}) \neq p(\mathbf{h}^{(1)} | \mathbf{v}) p(\mathbf{h}^{(2)} | \mathbf{v}) p(\mathbf{h}^{(3)} | \mathbf{v})$$

- Approximate inference (based on **variational inference**) is needed

$$p(\mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \mathbf{h}^{(3)} | \mathbf{v}) \approx q(\mathbf{h}^{(1)} | \mathbf{v}) q(\mathbf{h}^{(2)} | \mathbf{v}) q(\mathbf{h}^{(3)} | \mathbf{v})$$

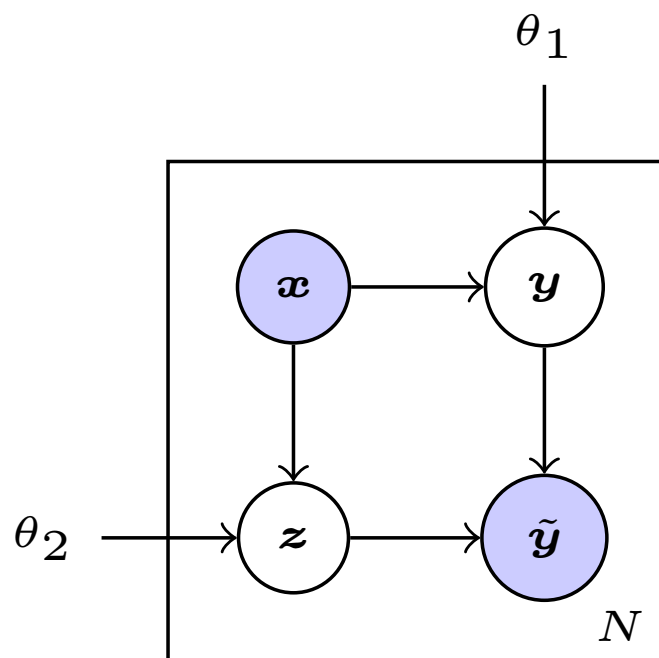
- Layer-wise unsupervised pre-training is also common



More Examples: Label Noise Model

- Modeling conditional distributions with deep neural networks in a graphical model that describes generation of noisy labels
- **Objective:** To infer ground truth labels for images
- Visible variables
 - x : Image
 - \tilde{y} : Noisy label (one-hot vector)
- Latent variables
 - y : True label (one-hot vector)
 - z : Label noise type (discrete variable)

- Graphical model



$$p(\tilde{y}, y, z|x) = \underbrace{p(\tilde{y}|y, z)}_{\text{Hand designed}} \underbrace{p(y|x; \theta_1)}_{\text{N.N.}} \underbrace{p(z|x; \theta_2)}_{\text{N.N.}}$$

- Label noise type and the conditional distribution $p(\tilde{\mathbf{y}}|\mathbf{y}, z)$

- Noise free ($z = 1$): $\tilde{\mathbf{y}} = \mathbf{y}$

$$p(\tilde{\mathbf{y}}|\mathbf{y}, z) = \tilde{\mathbf{y}}^T \mathbf{I} \mathbf{y}$$

- Random noise ($z = 2$): $\tilde{\mathbf{y}}$ is any value other than the true \mathbf{y}

$$p(\tilde{\mathbf{y}}|\mathbf{y}, z) = \frac{1}{L-1} \tilde{\mathbf{y}}^T (\mathbf{U} - \mathbf{I}) \mathbf{y}$$

where

- * \mathbf{U} is a matrix of 1's
- * L is the number of possible labels

- Confusing noise ($z = 3$): $\tilde{\mathbf{y}}$ is any value close to the true \mathbf{y}

$$p(\tilde{\mathbf{y}}|\mathbf{y}, z) = \tilde{\mathbf{y}}^T \mathbf{C} \mathbf{y}$$

- Training θ_1, θ_2 based on the EM algorithm
 - **E-step:** compute the expected value of the complete log-likelihood

$$\begin{aligned}
 J(\theta) &= E_{p(y,z|\tilde{y},x;\theta^{(\text{old})})} \log p(\tilde{y}, y, z|x; \theta) \\
 &= \sum_{y,z} p(y, z|\tilde{y}, x) [\log p(\tilde{y}|y, z; C) + \log p(y|x; \theta_1) + \log p(z|x; \theta_2)]
 \end{aligned}$$

where $\theta = \{\theta_1, \theta_2\}$ and C is assumed to be known

- **M-step:** maximize w.r.t. θ

$$\nabla_{\theta_1} J(\theta_1^{(\text{old})}, \theta_2^{(\text{old})}) = \sum_y p(y|\tilde{y}, x) \nabla_{\theta_1} \log p(y|x; \theta_1^{(\text{old})})$$

$$\nabla_{\theta_2} J(\theta_1^{(\text{old})}, \theta_2^{(\text{old})}) = \sum_z p(z|\tilde{y}, x) \nabla_{\theta_2} \log p(z|x; \theta_2^{(\text{old})})$$

- These are merely (negative) cross-entropy
- Testing is achieved by the neural network $p(\mathbf{y}|\mathbf{x}; \theta_1)$

- Note that unlike RBM/DBM, the hidden variables here are interpretable as is the case with most conventional graphical models

Review

- Directed vs. undirected graphical models
- Probability distributions and their graph representations
- Training, sampling, and inference for graphical models
- Extracting conditional independence: d-separation and separation
- Deep learning with graphical models