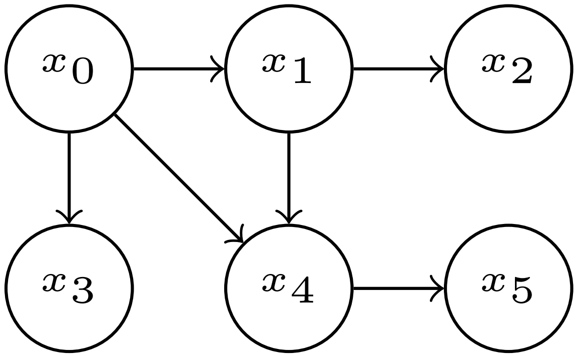
Chapter 16

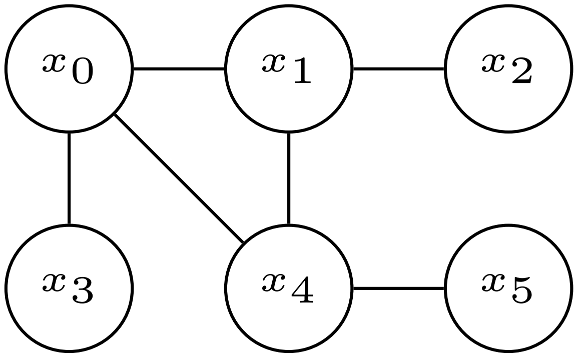
Structured Probabilistic Models for Deep Learning

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

- Thing we will be concerned with around the graphical models
  - Learning the model structure $p(\boldsymbol{x})$ and parameters $\boldsymbol{\theta}$

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

  - Drawing samples from the learned model

$$\boldsymbol{x} \sim p(\boldsymbol{x}; \boldsymbol{\theta}^*) \text{ or } \boldsymbol{x_2} \sim p(\boldsymbol{x_2} | \boldsymbol{x_1}; \boldsymbol{\theta}^*)$$

  - Doing approximate or exact inference

$$\arg\max_{\boldsymbol{x_2}} p(\boldsymbol{x_2} | \boldsymbol{x_1}; \boldsymbol{\theta}^*) \approx \arg\max_{\boldsymbol{x_2}} q(\boldsymbol{x_2} | \boldsymbol{x_1}; \boldsymbol{w})$$

Directed Graphical Models

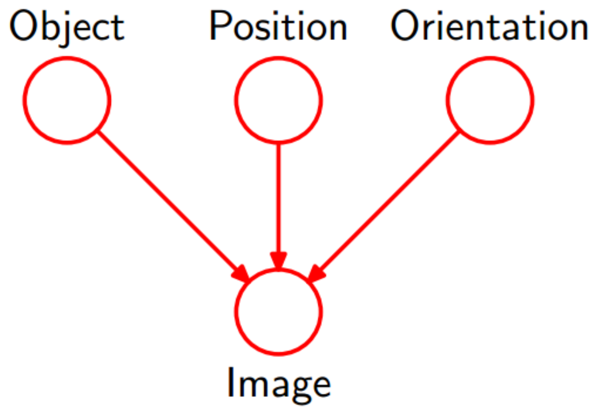
- A directed model defined on $\boldsymbol{x}$ is specified by
  1. A directed acyclic graph $\mathcal{G}$ with nodes denoting elements x_i of $\boldsymbol{x}$
  2. A set of local conditional probability distributions $p(x_i|Pa_{\mathcal{G}}(x_i))$ with $Pa_{\mathcal{G}}(x_i)$ giving the parent nodes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

  and factorizes the joint distribution of the node variables as

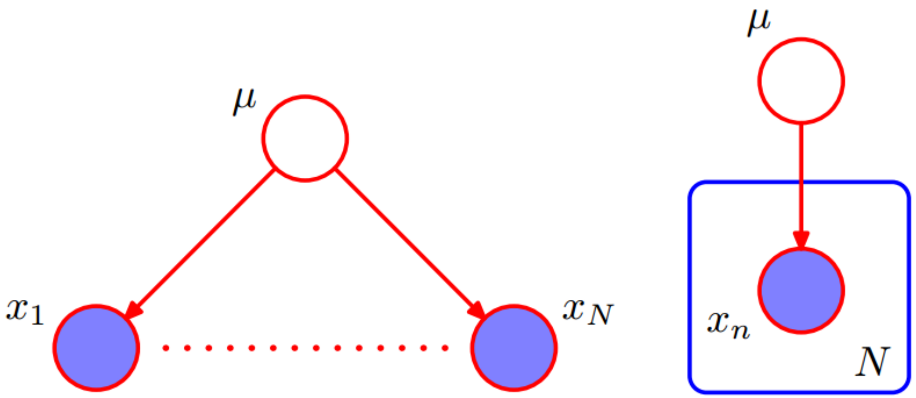
$$p(\boldsymbol{x}) = \prod_i p(x_i|Pa_{\mathcal{G}}(x_i))$$

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

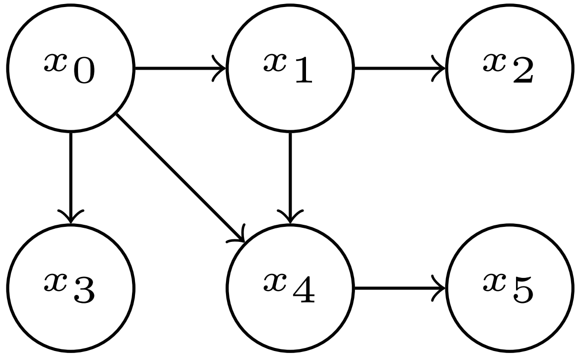
- They are most naturally applicable in situations where there is clear causality between variables



- For convenience, we sometimes introduce plate notation



- As an example, we have for the following graph



$$
\begin{aligned}
p(x_0, x_1, x_2, x_3, x_4, x_5) = &p(x_0)p(x_1|x_0)p(x_2|x_1)p(x_3|x_0) \\
&p(x_4|x_1, x_0)p(x_5|x_4)
\end{aligned}
$$

- When compared to the chain rule of probability,

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

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

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

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

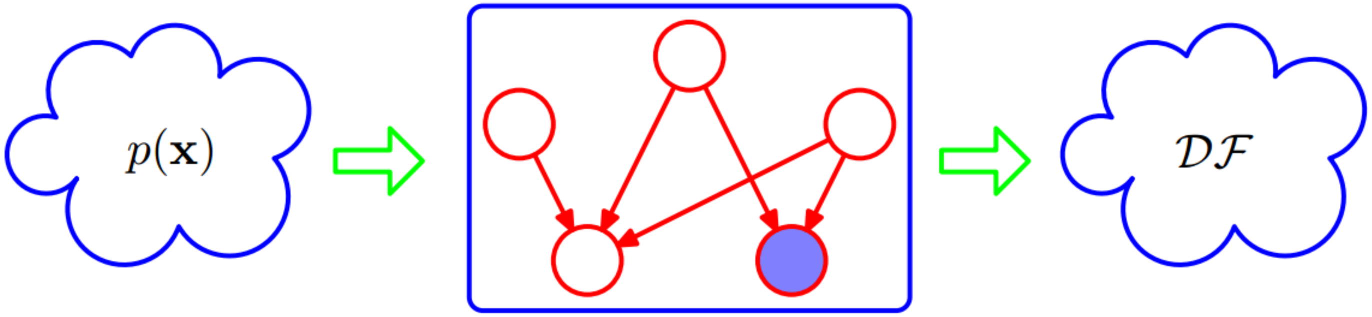
- Note however it only specifies which variables are allowed to appear in the arguments; there is no constraint on how we define each conditional probability distribution
- In the present example, we may as well specify

$$p(x_1|x_0) = f_1(x_1, x_0) = p(x_1)$$
$$p(x_2|x_1) = f_2(x_2, x_1) = p(x_2)$$
$$p(x_3|x_0) = f_3(x_3, x_0) = p(x_3)$$
$$p(x_4|x_1, x_0) = f_4(x_4, x_1, x_0) = p(x_4)$$
$$p(x_5|x_4) = f_5(x_5, x_4) = p(x_5)$$

to arrive at a fully factorized distribution

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

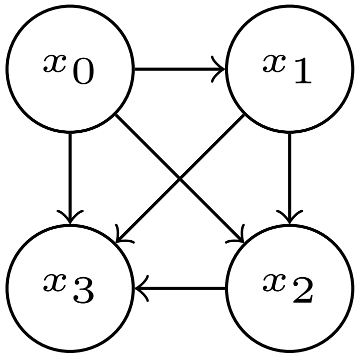
- As such, there could be several distributions that satisfy the graph factorization; it is helpful to think of a directed graph as a filter



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

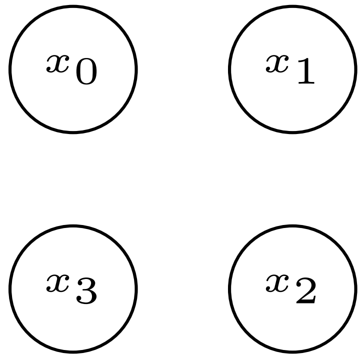
- **Extreme case I:** A fully connected graph will accept any possible distribution over the given variables



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

(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

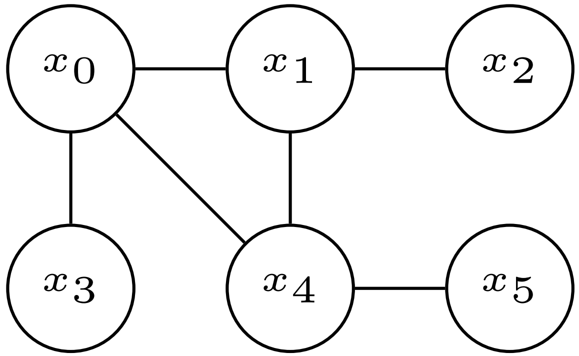
$$p(\boldsymbol{x}) = \frac{1}{Z} \prod_{\mathcal{C} \in \mathcal{G}} \phi(\mathcal{C}) = \frac{1}{Z} \tilde{p}(\boldsymbol{x})$$

where

  - $\tilde{p}(\boldsymbol{x})$ 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



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

- 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**)

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

  where $\boldsymbol{x}_{\mathcal{C}}$ 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.

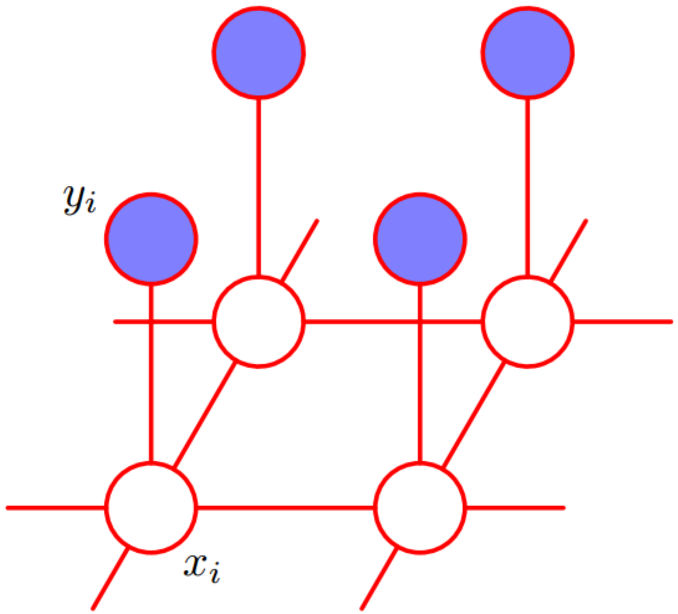
$$\phi(x) = \exp(-\beta x^2)$$

  with $x \in \mathbb{R}$ and 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

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

- 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

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

- The (complete) energy function is assumed to be

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

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

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

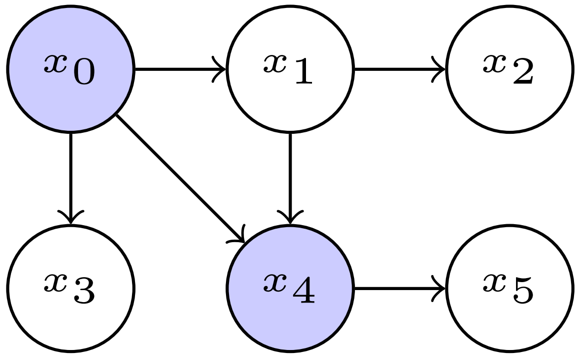
– De-noising can be cast as an inference problem

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

- As shown, the partition function Z often does not have tractable forms; some approximate algorithms are needed in estimating the model parameters, e.g., with the maximum likelihood principle

D-Separation

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



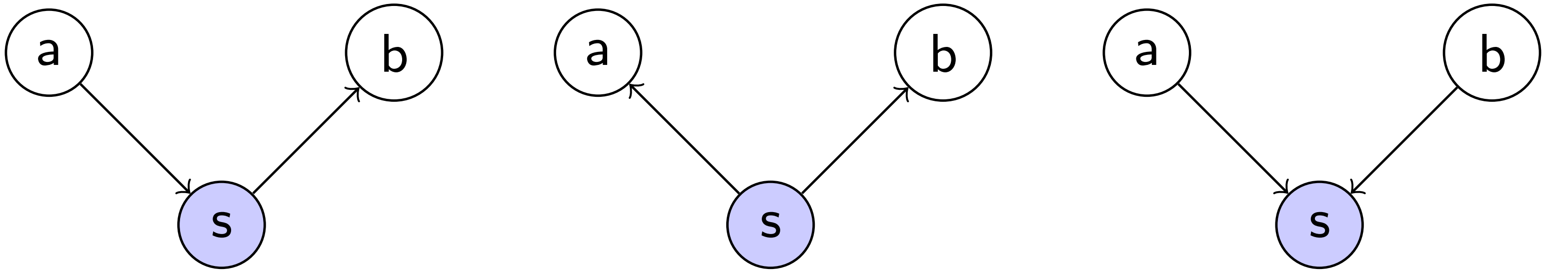
- Is the set of variables $\{x_1, x_2\}$ conditionally independent of the variable x_5, given the values of $\{x_0, x_4\}$?

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

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

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

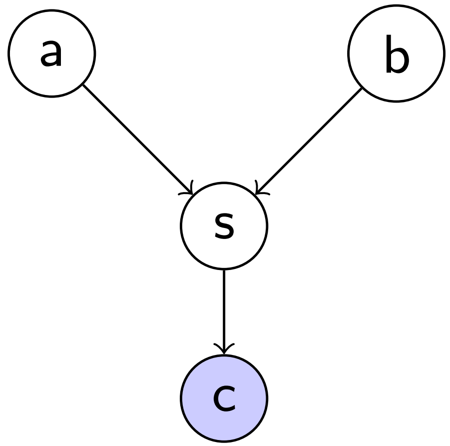
- **Tail-to-Tail:** a and b are **independent** (d-separated) given s

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

- **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

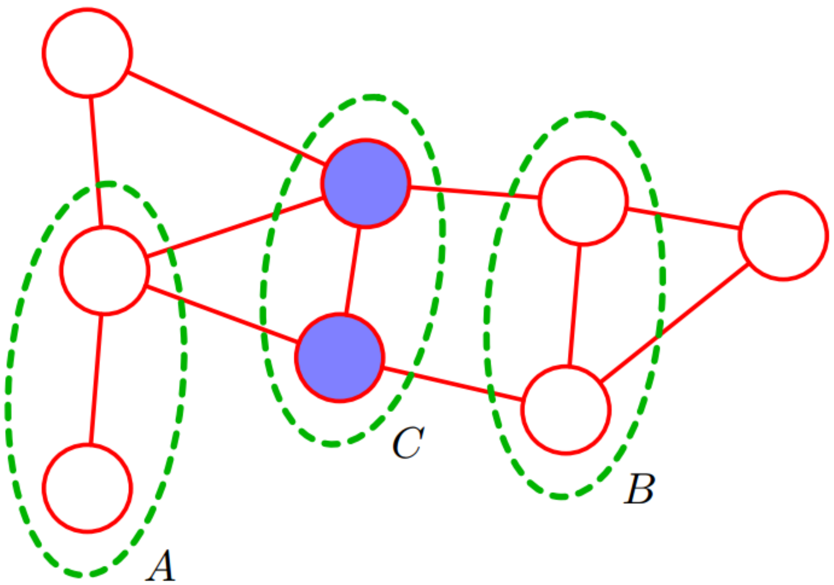


$p(a, b|c) \neq p(a|c)p(b|c)$ in general

- 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 those independencies implied by the graph; recall however that not all independencies of a distribution is captured by the graph (c.f. the filter interpretation)

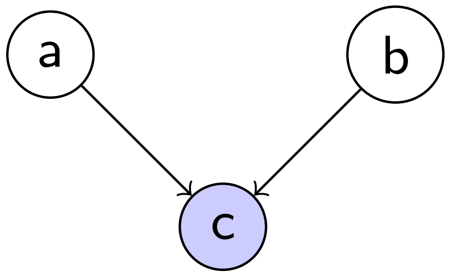
Separation

- 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

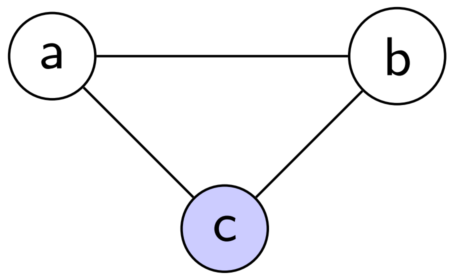


Conversion between Directed and Undirected Models

- 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

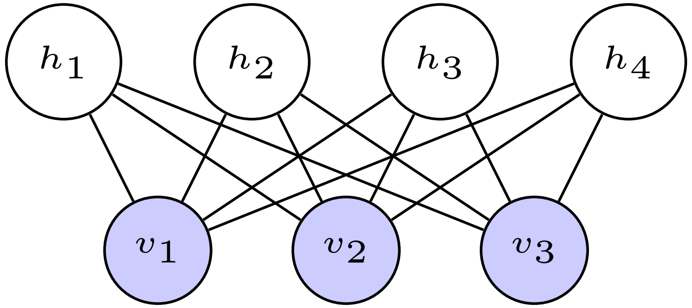
- In the present case, the potential function ϕ is given by

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

- 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



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

- There is no direct interaction between visible units or between hidden units (essentially, a bipartite graph)

- From the separation rules, we have

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

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

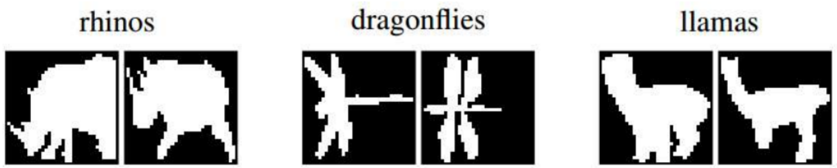
which are both factorial

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

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

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

- It is also possible to sample part of $\boldsymbol{v}$ 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 $\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{c}$ is achieved with the maximum likelihood principle

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

where the marginal distribution of visible units is given by

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

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

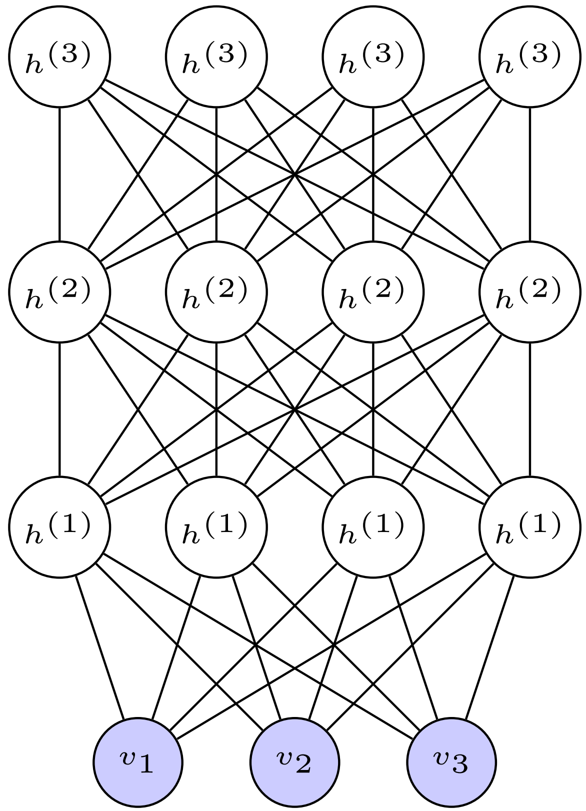
$$Z = \sum_{\boldsymbol{v}, \boldsymbol{h}} \exp(-E(\boldsymbol{v}, \boldsymbol{h}))$$

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

- Some specialized training techniques involving sampling are needed

Deep Boltzmann Machines (DBM)

- Introducing layers of hidden units to RBM



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

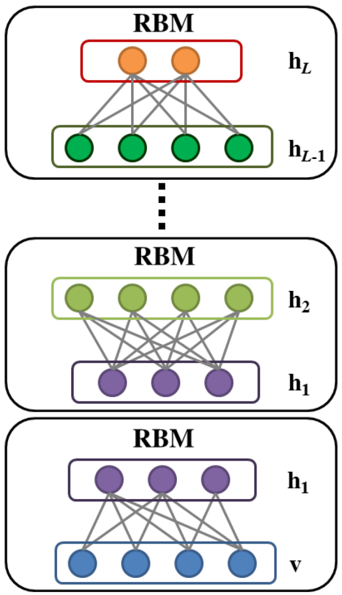
- From the graph, the posterior distribution is no longer factorial

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

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

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

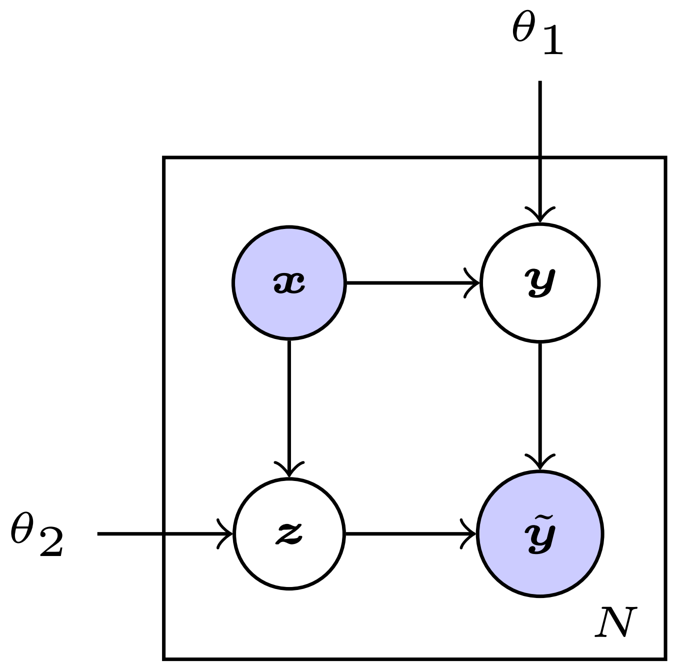
- Layer-wise unsupervised pre-training is also common



More Examples: Label Noise Model

- Another deep learning approach to graphical models is to approximate their conditional distributions with deep neural networks
- **Objective:** To infer ground truth labels for images
- Visible variables (noisy data)
  - x : Image
  - $\tilde{y}$: Noisy label (one-hot vector)
- Latent variables
  - y : True label (one-hot vector)
  - z : Label noise type

- Graphical model



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

- Label noise type and the conditional distribution $p(\tilde{\boldsymbol{y}}|\boldsymbol{y},\boldsymbol{z})$
  - Noise free ($z = 1$): $\tilde{\boldsymbol{y}} = \boldsymbol{y}$

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

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

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

    where
    * $\boldsymbol{U}$ is a matrix of 1's
    * L is the number of possible labels
  - Confusing noise ($z = 3$): $\tilde{\boldsymbol{y}}$ is any value close to the true $\boldsymbol{y}$

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

- Training of θ_1, θ_2 is based on the EM algorithm (study the paper)
- Testing is achieved by the neural network $p(\boldsymbol{y}|\boldsymbol{x};\theta_1)$
- 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