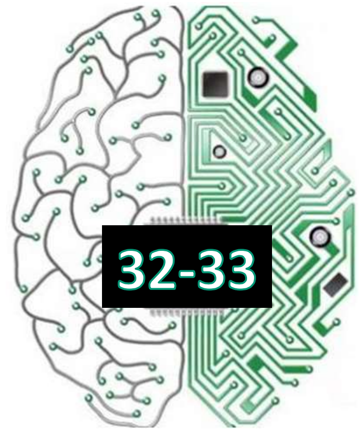
Open Elective Course [OE]
Course Code: CSO507
Winter 2023-24

Lecture#

Deep Learning

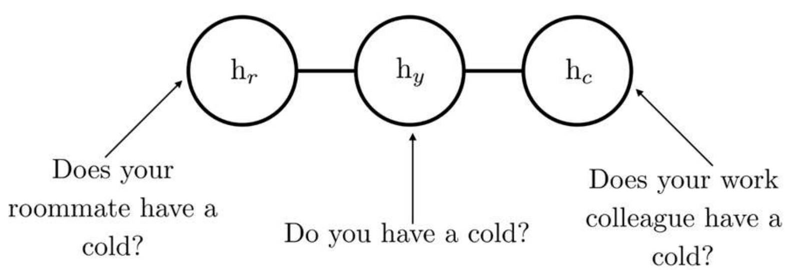
Unit-7: Structured Probabilistic Models (Part-II)
Unit-8: Generative Models (Part-I)



Course Instructor:
Dr. Monidipa Das
Assistant Professor
Department of Computer Science and Engineering
Indian Institute of Technology (Indian School of Mines) Dhanbad, Jharkhand 826004, India

Undirected Models

- **Markov random fields (MRFs) or Markov networks:**

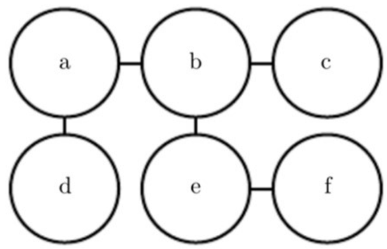


- Formally, an undirected graphical model is a structured probabilistic model defined on an undirected graph G.

Undirected Models

3

- For each clique C in the graph, ϕ(C) is a factor (called clique potential)
  - measures the affinity of the variables in that clique for being in each of their possible joint states.



- The factors are constrained to be non-negative.
- Together they define an ***unnormalized*** probability distribution: $\tilde{p}(\mathbf{x}) = \prod_{C \in G} \phi(C)$

Partition Function

4

- Normalized Probability Distribution

$$p(\mathbf{x}) = \frac{1}{Z}\tilde{p}(\mathbf{x})$$

where Z is the value that results in the probability distribution summing or integrating to 1:

$$Z = \int \tilde{p}(\mathbf{x})d\mathbf{x}$$

Normalizing constant

Also called partition function

It is possible to specify the factors in such a way that Z does not exist.

Choice of factors is important!

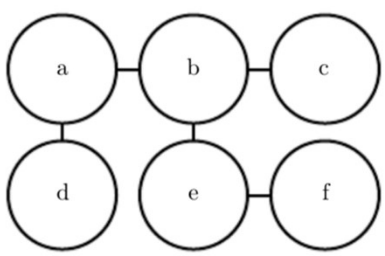
Energy-Based Models (EBMs)

- $\tilde{p}(x) = \exp(-E(x))$ Energy function
- enforces $\forall x,\ \tilde{p}(x) > 0$

Boltzmann distribution

- Unconstrained optimization.
- The probabilities in an energy-based model can approach arbitrarily close to zero but never reach it.
- Many energy-based models are called **Boltzmann machines**

Energy-based Models



$$E(a,b,c,d,e,f) = E_{a,b}(a,b) + E_{b,c}(b,c) + E_{a,d}(a,d) + E_{b,e}(b,e) + E_{e,f}(e,f)$$

$$\boxed{p(a,b,c,d,e,f) = \frac{1}{Z}\phi_{a,b}(a,b)\phi_{b,c}(b,c)\phi_{a,d}(a,d)\phi_{b,e}(b,e)\phi_{e,f}(e,f)}$$

Different cliques in undirected graph correspond to different terms of the energy function

Free Energy instead of Probability

7

- Algorithms don't need $p_{\text{model}}(\boldsymbol{x})$ but only

$$\log \tilde{p}_{\text{model}}(\boldsymbol{x}) \quad \text{where} \quad \tilde{p}(\mathbf{x}) = \exp(-E(\mathbf{x}))$$

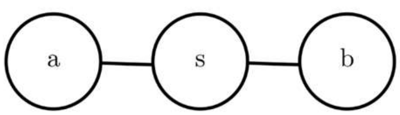
- EBMs with hidden units $\boldsymbol{h}$ use the negative of this quantity, called the *free energy*

$$F(\boldsymbol{x}) = -\log \sum_{h} \exp\left(-E(\boldsymbol{x}, \boldsymbol{h})\right)$$

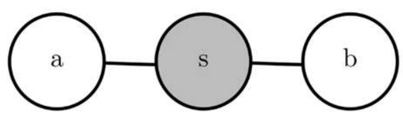
Separation

8

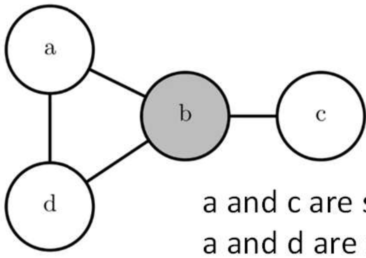
Conditional independence in undirected models



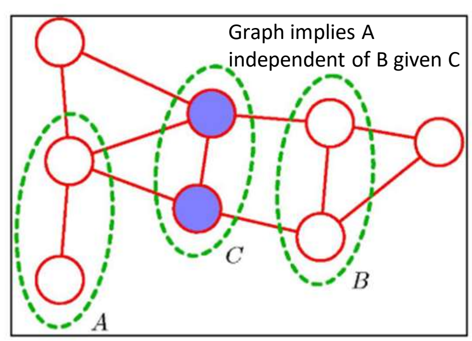
When s is not observed, influence can flow from a to b and vice versa through s.



When s is observed, it blocks the flow of influence between a and b: they are *separated*



a and c are separated given b
a and d are not separated given b

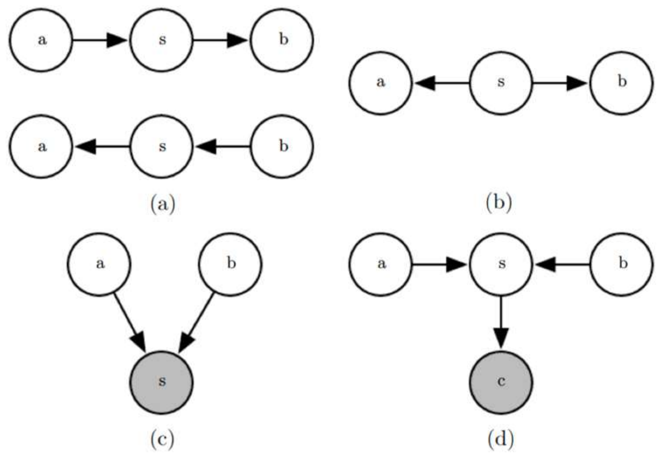


D-Separation

9

Separation concept in case of directed models

The flow of influence is more complicated for directed models

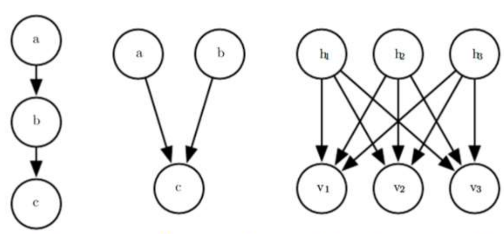


Converting directed to undirected

10

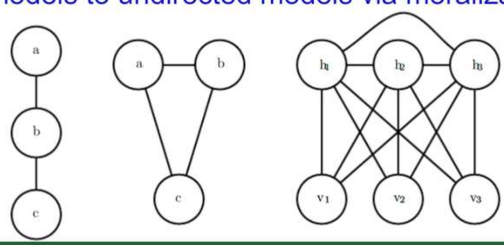
Resulting undirected model implies exactly the same set of independences and conditional independences

Directed Models

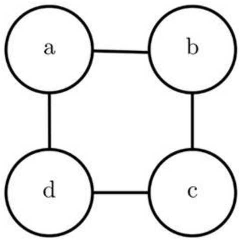


Converting directed models to undirected models via moralization

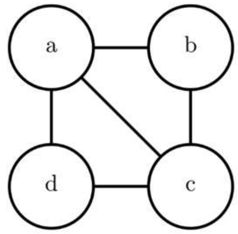
Undirected Models



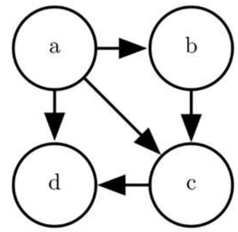
Converting undirected to directed



No loops of length greater than three allowed!



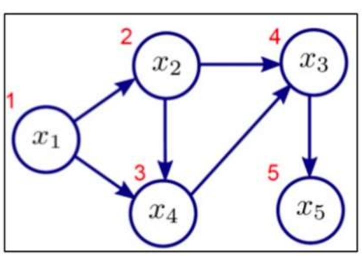
Add edges to triangulate long loops



Assign directions to edges. No directed cycles allowed.

Sampling from graphical models

- Sampling from directed models (BNs)
  - Ancestral Sampling



To generate one sample:

1. Sample x_1^* from $Pr(x_1)$
2. Sample x_2^* from $Pr(x_2 | x_1^*)$
3. Sample x_4^* from $Pr(x_4 | x_1^*, x_2^*)$
4. Sample x_3^* from $Pr(x_3 | x_2^*, x_4^*)$
5. Sample x_5^* from $Pr(x_5 | x_3^*)$

  - Without topological sorting, we might attempt to sample a variable before its parents are available

Sampling from graphical models

13

- Sampling from undirected models (MNs)
  - Gibbs Sampling
    - Simplest approach for sampling from an MN
  - Gibbs Sampling with M variables
    - Initialize first sample: $\{z_i, i = 1,\ldots,M\}$
    - For $t=1,\ldots,T$, T = no of samples
      - Sample $z_1^{(\tau+1)} \sim p(z_1|z_2^{(\tau)}, z_3^{(\tau)}, \ldots, z_M^{(\tau)})$
      - Sample $z_2^{(\tau+1)} \sim p(z_2|z_1^{(\tau+1)}, z_3^{(\tau)}, \ldots, z_M^{(\tau)})$
      - .....
      - Sample $z_j^{(\tau+1)} \sim p(z_j|z_1^{(\tau+1)}, \ldots z_{j-1}^{(\tau+1)}, z_{j+1}^{(\tau)} \ldots, z_M^{(\tau)})$
      - .....
      - Sample $z_M^{(\tau+1)} \sim p(z_M|z_1^{(\tau+1)}, z_2^{(\tau+1)}, \ldots, z_{M-1}^{(\tau+1)})$
    - $p(z_j|z_{-j})$ is called a *full conditional* for variable j

Advantages of Structured Modeling

14

- Reduce cost of representing distributions
- Operations use less runtime and memory
- Convey information by leaving edges out
- Sampling accelerated for directed models

Deep learning approach to structured models

15

- PGM: Probabilistic Graphical Model

- Traditional PGMs vs. PGMs in deep learning
  - 1.Depth
  - 2.Proportion of observed to latent variables
  - 3.Latent semantics (meaning of a latent variable)
  - 4.Connectivity and inference algorithm
  - 5.Intractability and approximation

Deep learning approach to structured models

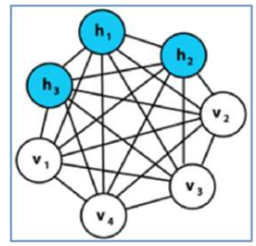
16

- PGMs in deep learning are not deep PGMs
- Deep Learning has more latent variables than observed variables
- Deep Learning does not take any specific semantics ahead of time
- Deep learning PGMs have large groups of units connected other large groups of units

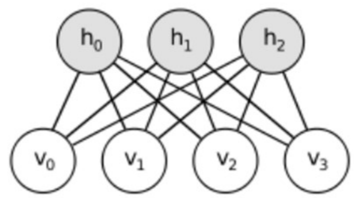
Example: RBMs

17

- **Restricted Boltzmann machine** (RBM)
  – quintessential example of how graphical models are used for deep learning.
- RBM is a bipartite graph
- RBM is a special case of Boltzmann machines and Markov networks
- RBM itself is not a deep model

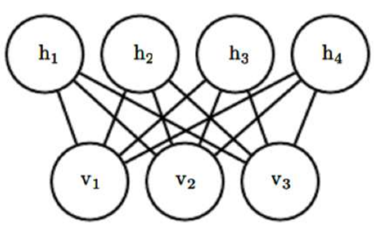


General BM

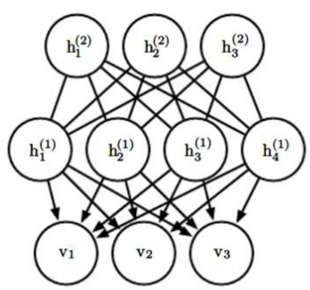


Models constructed using RBMs

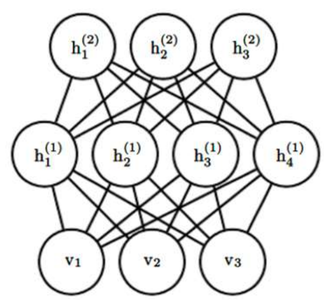
18



RBM



Deep belief network



Deep Boltzmann Machine

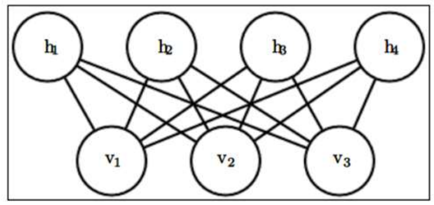
Properties of RBMs

19

- Restrictions of RBM structure yields nice properties:

$p(\boldsymbol{h}|\boldsymbol{v})=\Pi_i\, p(h_i|\boldsymbol{v})$ and
$p(\boldsymbol{v}|\boldsymbol{h})=\Pi_i\, p(v_i|\boldsymbol{h})$

Since nodes at same level are independent



- Individual conditionals are simple to compute

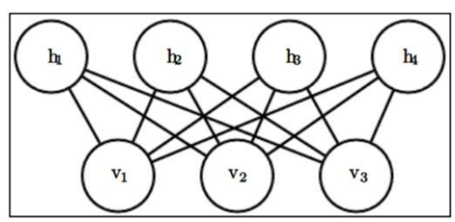
RBM: an energy-based model

20

- Joint-probability distribution is specified by the energy function:

$P(\mathbf{v}=\boldsymbol{v},\mathbf{h}=\boldsymbol{h})=(1/Z)\exp(-\boldsymbol{E}(\boldsymbol{v},\boldsymbol{h}))$

  – The energy function for an RBM is
  – $E(\boldsymbol{v},\boldsymbol{h})= -\boldsymbol{b}^{\mathrm{T}}\boldsymbol{v} - \boldsymbol{c}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{v}^{\mathrm{T}} W\boldsymbol{h}$
  – Z is the partition function
  $Z = \Sigma_v\, \Sigma_h\, E(\boldsymbol{v},\boldsymbol{h})$
  – Since Z is intractable $P(\boldsymbol{v})$ is also intractable



RBM conditionals are tractable

21

- Although $P(v)$ is intractable,
  - Conditionals $P(\boldsymbol{h}|\boldsymbol{v})$, $P(\boldsymbol{v}|\boldsymbol{h})$ are factorial & easily computed:

$$P(\boldsymbol{h} \mid \boldsymbol{v}) = \frac{P(\boldsymbol{h},\boldsymbol{v})}{P(\boldsymbol{v})} = \frac{1}{P(\boldsymbol{v})}\frac{1}{Z}\exp\left\{\boldsymbol{b}^T\boldsymbol{v} + \boldsymbol{c}^T\boldsymbol{h} + \boldsymbol{v}^T W\boldsymbol{h}\right\} = \frac{1}{Z'}\exp\left\{\boldsymbol{c}^T\boldsymbol{h} + \boldsymbol{v}^T W\boldsymbol{h}\right\}$$
$$= \frac{1}{Z'}\exp\left\{\sum_{j=1}^{n_h} c_j h_j + \sum_{j=1}^{n_h}\boldsymbol{v}^T W_{:,j}\boldsymbol{h}_j\right\} = \frac{1}{Z'}\prod_{j=1}^{n_h}\exp\left\{\boldsymbol{c}_j\boldsymbol{h}_j + \boldsymbol{v}^T W_{:,j}\boldsymbol{h}_j\right\}$$

  - Normalizing the distributions over individual binary $\boldsymbol{h}$

$$P(h_j = 1 \mid \boldsymbol{v}) = \frac{\tilde{P}(h_j = 1 \mid \boldsymbol{v})}{\tilde{P}(h_j = 0 \mid \boldsymbol{v}) + \tilde{P}(h_j = 1 \mid \boldsymbol{v})} = \frac{\exp\left\{c_j + \boldsymbol{v}^T W_{:,j}\right\}}{\exp\{0\} + \exp\left\{c_j + \boldsymbol{v}^T W_{:,j}\right\}} = \sigma\left(c_j + \boldsymbol{v}^T W_{:,j}\right)$$

  - We now express full conditional as a factorial distribution

$$P(\boldsymbol{h} \mid \boldsymbol{v}) = \prod_{j=1}^{n_h}\sigma\left((2\boldsymbol{h}-1)\odot(\boldsymbol{c} + W^T\boldsymbol{v})\right)_j$$ and similarly $$P(\boldsymbol{v} \mid \boldsymbol{h}) = \prod_{j=1}^{n_v}\sigma\left((2\boldsymbol{v}-1)\odot(\boldsymbol{b} + W^T\boldsymbol{h})\right)_i$$

Training RBM

22

- RBM properties allow for block Gibbs sampling
  - Alternate between sampling all **h** simultaneously and all **v** simultaneously
- Energy function: $\mathrm{E}(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{b}^{\mathrm{T}}\boldsymbol{v} - \boldsymbol{c}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{v}^{\mathrm{T}}\mathrm{W}\boldsymbol{h}$
  - where $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{W}$ are unconstrained, real-valued learnable parameters
- Since the energy function is a linear function of its parameters, it is easy to take derivatives

$$\frac{\partial}{\partial W_{i,j}} E(\boldsymbol{v},\boldsymbol{h}) = -v_i h_j$$

- These two properties, efficient Gibbs sampling and efficient derivatives make training convenient

Training RBM

23

- Joint configuration (v, h)

$$E(\boldsymbol{v},\boldsymbol{h}) = -\sum_{i\in\text{visible}} a_i v_i - \sum_{j\in\text{hidden}} b_j h_j - \sum_{i,j} v_i h_j w_{ij}$$

$$p(\boldsymbol{v},\boldsymbol{h}) = \frac{1}{Z}e^{-E(\boldsymbol{v},\boldsymbol{h})} \qquad Z = \sum_{\boldsymbol{v},\boldsymbol{h}} e^{-E(\boldsymbol{v},\boldsymbol{h})} \qquad p(\boldsymbol{v}) = \frac{1}{Z}\sum_{\boldsymbol{h}} e^{-E(\boldsymbol{v},\boldsymbol{h})}$$

- Changing probability of v

Likelihood: $P(\{\boldsymbol{v}^{(1)},..\boldsymbol{v}^{(M)}\}) = \prod_m p(\boldsymbol{v}^{(m)})$

Log-likelihood:

$$\ln P(\{\boldsymbol{v}^{(1)},..\boldsymbol{v}^{(M)}\}) = \sum_m \ln p(\boldsymbol{v}^{(m)}) = \sum_m \ln\left(\frac{1}{Z}\sum_{\boldsymbol{h}} e^{-E(\boldsymbol{v},\boldsymbol{h})^{(m)}}\right) = \sum_m \ln\left(\sum_{\boldsymbol{h}} e^{-E(\boldsymbol{v},\boldsymbol{h})^{(m)}}\right) - \sum_m \ln\left(\sum_{\boldsymbol{v},\boldsymbol{h}} e^{-E(\boldsymbol{v},\boldsymbol{h})}\right)$$

Derivative of the log-probability of a training vector wrt a weight:

$$\frac{\partial \ln p(\boldsymbol{v})}{\partial w_{ij}} = \mathbb{E}_{\text{data}}(v_i h_j) - \mathbb{E}_{\text{model}}(v_i h_j)$$

Learning rule for stochastic steepest ascent

$\Delta w_{ij} = \boldsymbol{\varepsilon}\left(\mathbb{E}_{\text{data}}(v_i h_j) - \mathbb{E}_{\text{model}}(v_i h_j)\right)$. where $\boldsymbol{\varepsilon}$ is the learning rate

Samples for Computing Expectations

24

- Getting unbiased samples for $E_{\text{data}}(v_i h_j)$
  - h_j: Given random training image $\boldsymbol{v}$, the binary state h_j for each hidden unit is set to 1 with probability

$$p(h_j = 1 \mid \boldsymbol{v}) = \sigma\left(b_j + \sum_i v_i w_{ij}\right)$$

  - v_i : Given a random training image $\boldsymbol{v}$, the binary state v_i for a visible unit is set to 1 with probability

$$p(v_i = 1 \mid \boldsymbol{v}) = \sigma\left(ai + \sum_j h_j w_{ij}\right)$$

- Getting unbiased samples for $E_{\text{model}}(v_i h_j)$
  - Can be done by starting at a random state of visible units and performing Gibbs sampling for a long time
    - One iteration of alternating Gibbs sampling consists of updating all hidden units in parallel followed by updating all visible units

Questions?