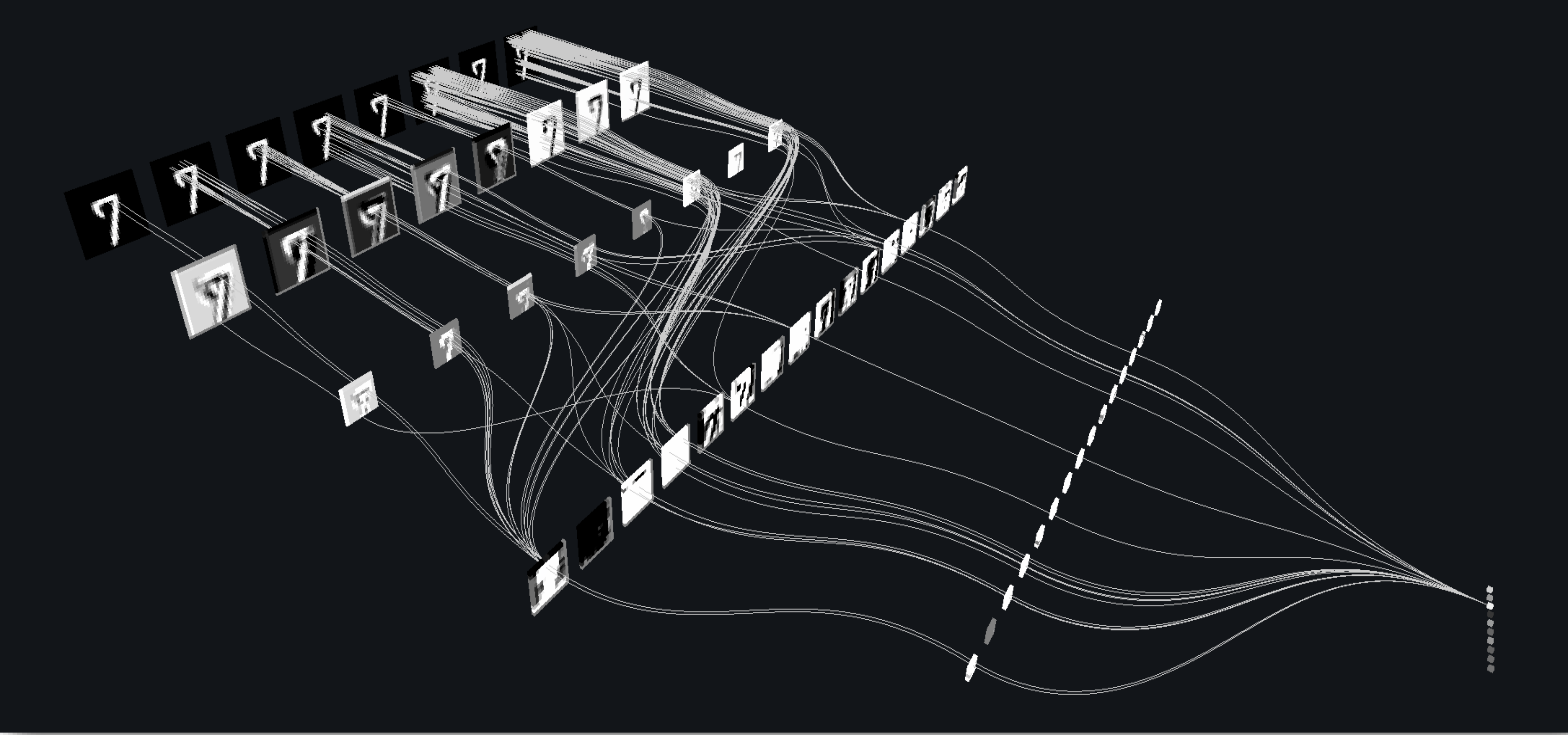
Deep Learning

Md. Jalil Piran, PhD
Asst. Professor
Computer Science and Engineering
Sejong University
Spring, 2021

Outline

- Introduction to Deep Learning (DL)
- The History of DL
- Programming Tools
- Artificial Neural Networks (ANNs)
- Optimization in DL
- Convolutional Neural networks (CNNs)
- **Unsupervised Pre-trained Networks (UPNs)**



ARCHITECTURE OF DEEP LEARNING

DL Architectures

- **Higher-level Architecture**
  - **Convolutional Neural Networks (CNNs)**
  - **Unsupervised Pre-trained Networks (UPNs)**
    - Deep belief networks (DBNs)
    - Autoencoders
    - Generative adversarial networks (GANs)
  - **Recurrent Neural Networks (RNNs)**
    - Bidirectional recurrent neural networks (BRNN)
    - LSTM
  - **Recursive Neural Networks**

Unsupervised Learning

Motivation and Strengths:

- Unsupervised learning is **not expensive** and **time consuming** like supervised learning.
- Unsupervised learning requires **no human intervention**.
- Unlabeled data is **easy** to **find** with large quantities, unlike labeled data which is scarce.

- **Weaknesses**

- More difficult than supervised learning because there is NO **Single objective** (like test set accuracy)

Unsupervised Feature Learning

- Train representations with unlabeled data.
  - Minimize an *unsupervised* training loss.
    - Often based on generic priors about characteristics of good features
    - Usually train 1 layer of features at a time.

UPNs

- Unsupervised pre-trained networks (UPNs)
  - Motivation: representation **leaning** and **transfer learning**
  - Deep belief networks (DBNs)
  - Autoencoders
  - Generative adversarial networks (GANs)

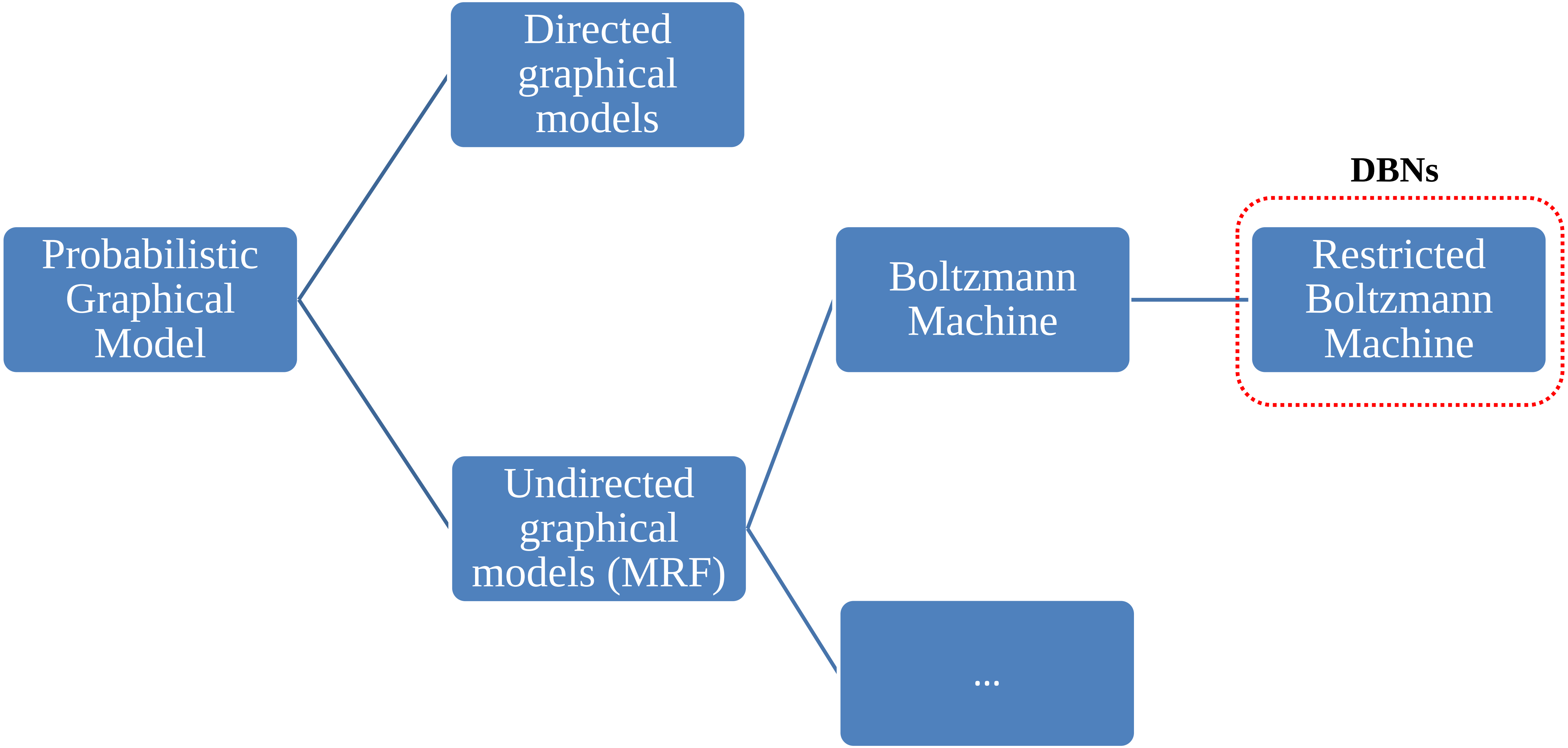
DBN

- DBN’s prerequisite
  - MRF
  - Sampling
  - **RBMs**

Restricted Boltzmann Machine (RBM)

- RBMs are building blocks for the multi-layer leaning architectures, e.g. DBNs.
- RBMs are a special case of general **Boltzmann Machines** (BMs).
- BMs are a particular form of **Markov Random Field (MRF)**, a.k.a. **Markov networks** or **undirected graph models**.

DBNs



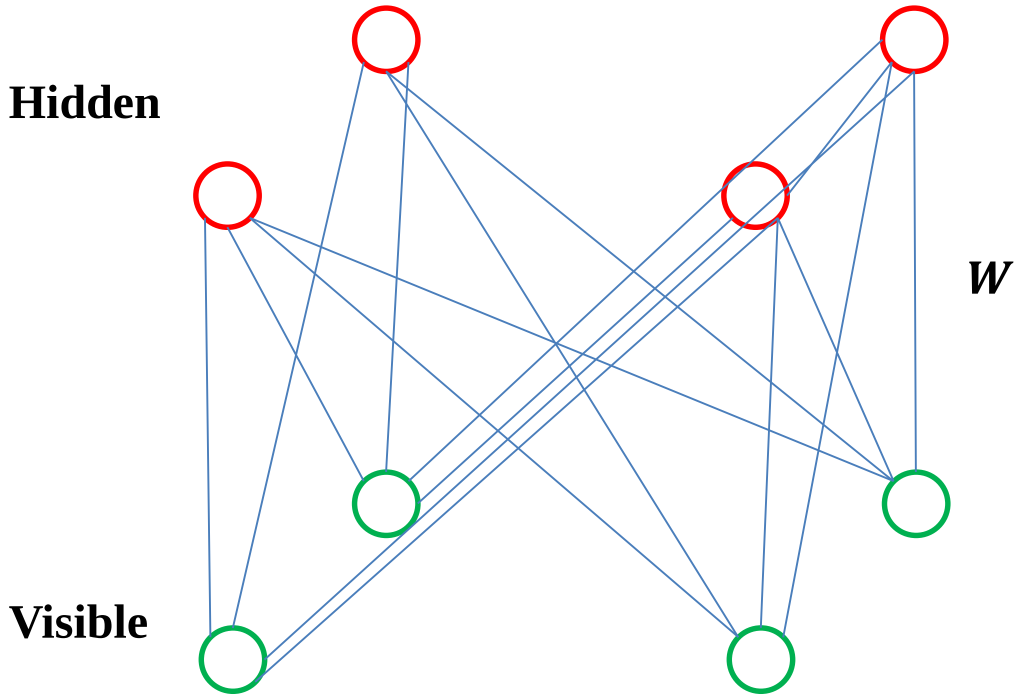
DBNs
Probabilistic Graphical Model
Directed graphical models
Undirected graphical models (MRF)
Boltzmann Machine
Restricted Boltzmann Machine
...

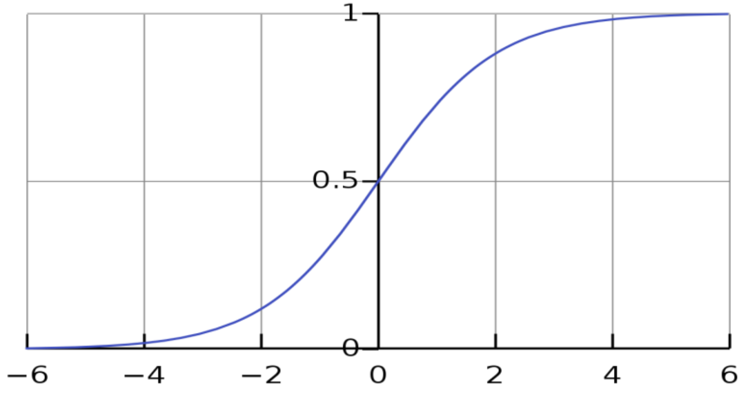
… topic covered so far

- BMs
- RMBs
- Joint distribution
- Potential functions
- Cliques
- Maximal cliques
- Energy function
- Energy function.
- Conditional independent
- Ascending gradient
- Transformation method
- Rejection sampling
- Gibbs sampling

RBM

- RBM: an special type MRF
  - e.g. undirected graph.
  - **Components**:
    - **Visible units** $v \in \{0,1\}^m$
    - **Hidden units** $h \in \{0,1\}^n$
    - The conditional probability of a single variable being one can be interpreted as the firing rate of a neuron with sigmoid activation function

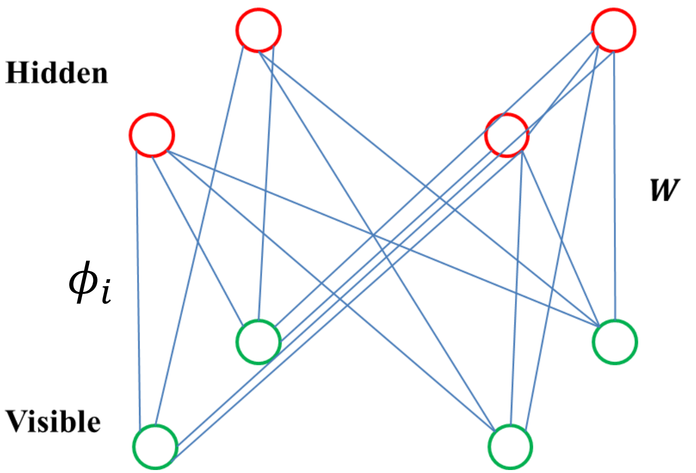




Sigmoid Function

RBM

- **Factors**: the links between each visible and hidden units.
- **Joint distribution**: multiplying factors and normalization



- **Log-linear models**

$$\phi_i(\boldsymbol{D}) = \exp\left(-f_i(\boldsymbol{D})\right)$$

$$\tilde{P} \propto \prod_j \exp\left(-f_i(\boldsymbol{D})\right) = \exp\left(-\sum_j f_j(\boldsymbol{D})\right)$$

features

RBM

- **Log-linear models**

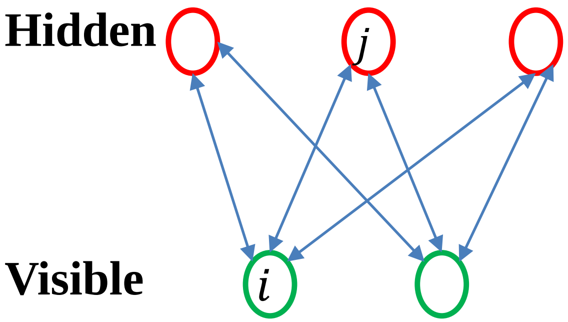
$$\phi_i(\boldsymbol{D}) = \exp\big(-f_i(\boldsymbol{D})\big)$$

$$\tilde{P} \propto \prod_j \exp\big(-f_i(\boldsymbol{D})\big) = \exp\left(-\sum_j f_j(\boldsymbol{D})\right)$$

- $D = v_i, h_j$
- $f(v_i, h_i) = E\big(v_i, h_j\big) = v_i w_{ij} h_j + a_i v_i + b_j h_j$
- $\phi_{ij}\big(v_i, h_j\big) = \exp\Big(-E\big(v_i, h_j\big)\Big)$
- $P(v, h) = \frac{1}{Z}\Pi_{i,j}\phi_{ij}\big(v_i, h_j\big) = \frac{1}{Z}\Pi_{i,j}\exp\Big(-E\big(v_i, h_j\big)\Big) = \frac{1}{Z}\exp(-\sum_{ij} E(v_i, h_j))$

RBM

- In an RBM, the hidden units are **conditionally independent** given the visible states.
- So, an unbiased sample from the posterior distribution is possible given a data-vector.



$$P(\boldsymbol{h}|\boldsymbol{v};\theta) = \prod_j p(h_j|v), \qquad P(h_j = 1|v) = g\left(\sum_i W_{ij} v_i + a_j\right)$$

$$P(\boldsymbol{v}|\boldsymbol{h};\theta) = \prod_i p(v_i|h), \qquad P(v_i = 1|h) = g\left(\sum_j W_{ij} h_j + b_j\right)$$

RBM

- **The energy of a joint configuration**
  - Just ignore the biases for simplicity.

$$E(v,h) = -\sum_{i,j} v_i h_j w_{ij}$$

- $E(v,h)$: **energy** with configuration v on the visible units and h on the hidden units
- v_i: binary state of **visible unit** i
- h_j: binary state of **hidden unit** j
- w_{ij}: **weight** between units i and j

$$-\frac{\partial E(v,h)}{\partial w_{ij}} = v_i h_j$$

RBM

- **Using energies to define probabilities**
- The probability of a joint configuration over both visible and hidden units

$$p(v,h) = \frac{e^{-E(v,h)}}{\sum_{u,g} e^{-E(u,g)}}$$

Partition function

$$p(v,h) = \frac{1}{Z}\exp(-\sum_{ij} E(v_i,h_j))$$

- The probability of a configuration of the visible units is the sum of the probabilities of all the joint configuration that contain it.

$$p(v) = \frac{\sum_h e^{-E(v,h)}}{\sum_{u,g} e^{-E(u,g)}}$$

RBM

- Summary

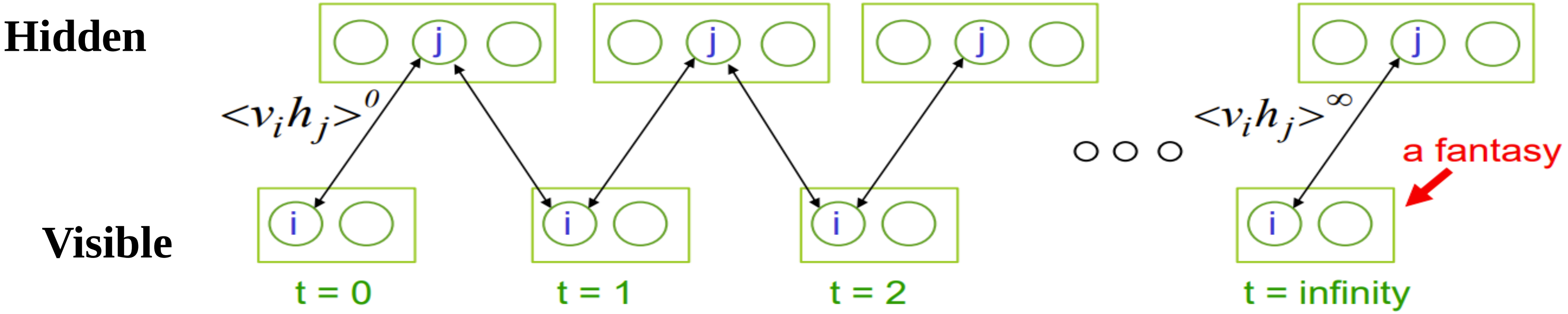
Joint distribution	$p(v,h) = \frac{1}{Z} exp(-E(v,h))$
Energy function	$E(v,h) = -v^T Wh - a^T v - b^T h$
Probability of visible units	$p(v) = \boxed{\sum_h p(v,h)}$
Likelihood	$maximize_{\{w_{ij},a_i,b_j\}} \frac{1}{\text{m}} \sum_{l=1}^{m} \log\left(\boxed{\sum_h P\left(\boldsymbol{v}^{(l)}, \boldsymbol{h}^{(l)}\right)}\right)$
Derivative	$\frac{\partial}{\partial w_{ij}}\left(\frac{1}{m}\sum_{l=1}^{m} \log\left(\sum_h P\left(\boldsymbol{v}^{(l)}, \boldsymbol{h}^{(l)}\right)\right)\right)$

RBM

- Maximum likelihood learning algorithm for an RBM
  - Start with a training vector on the visible units.
  - Then, alternate between updating all the hidden units in parallel and updating all the visible units in parallel.

$$\frac{\partial \log p(v)}{\partial w_{ij}} = \langle v_i h_j \rangle^0 - \langle v_i h_j \rangle^\infty$$

- $\langle v_i h_j \rangle^0$: from fixing $\boldsymbol{v}$ to observed value, and sampling $\boldsymbol{h}$ from $P(\boldsymbol{h}|\boldsymbol{v})$
- $\langle v_i h_j \rangle^\infty$: from running Gibbs sampling to convergence.



RBM

- **Maximum likelihood learning**

$$\frac{\partial \log P(v)}{\partial \theta} = -\sum_{h} P(h|v)\frac{\partial E(v,h)}{\partial \theta} + \sum_{v',h'} P(v',h')\frac{\partial E(v',h')}{\partial \theta}$$

$$E(v,h) = -v^T W h - a^T v - b^T$$

$$= \sum_{i=1}^{g_v}\sum_{j=1}^{g_h} W_{ij} v_i h_j - \sum_{i=1}^{g_v} a_i v_i - \sum_{j=1}^{g_h} b_j h_j$$

$$-\frac{\partial \log P(v)}{\partial W_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}$$

$$-\frac{\partial \log P(v)}{\partial b_j} = \langle h_j \rangle_{data} - \langle h_j \rangle_{model}$$

RBM

- **Maximum likelihood learning**

Gibbs Sampling

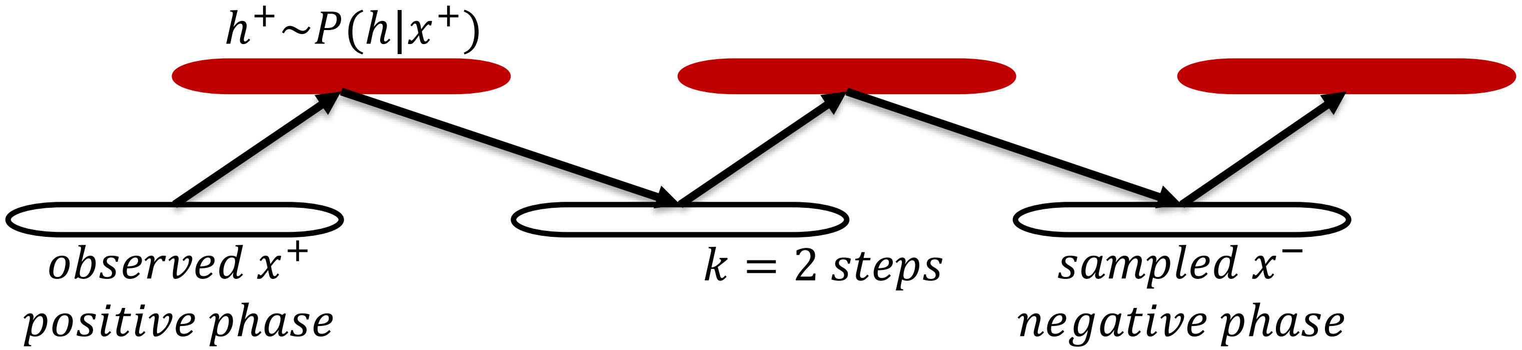
Contrastive Divergence (CD)

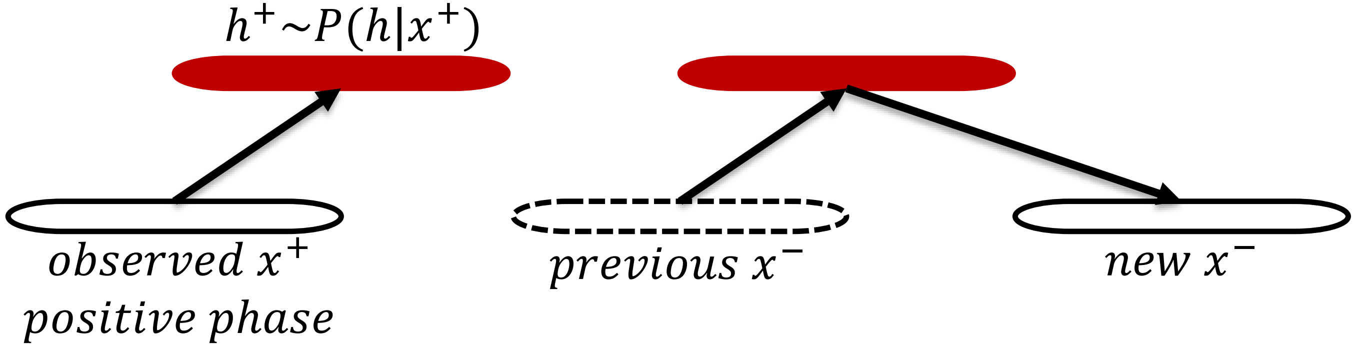
Persistent Contrastive Divergence (PCD)

PCD with Partial Smoothing (PCD PS)

Fast PCD (FPCD)

Free Energy in PCD (FECD)





RBM

- Different types of units
- RBM’s were developed using **binary** visible and hidden units
- Many other types of units can be used:
  - **Softmax** and **multinomial** units
  - **Gaussian** visible units
  - **Rectified** linear units

Office: #432, Dayang AI Center, **Phone**: 010-8999-8586, **email**: *piran@sejong.ac.kr*