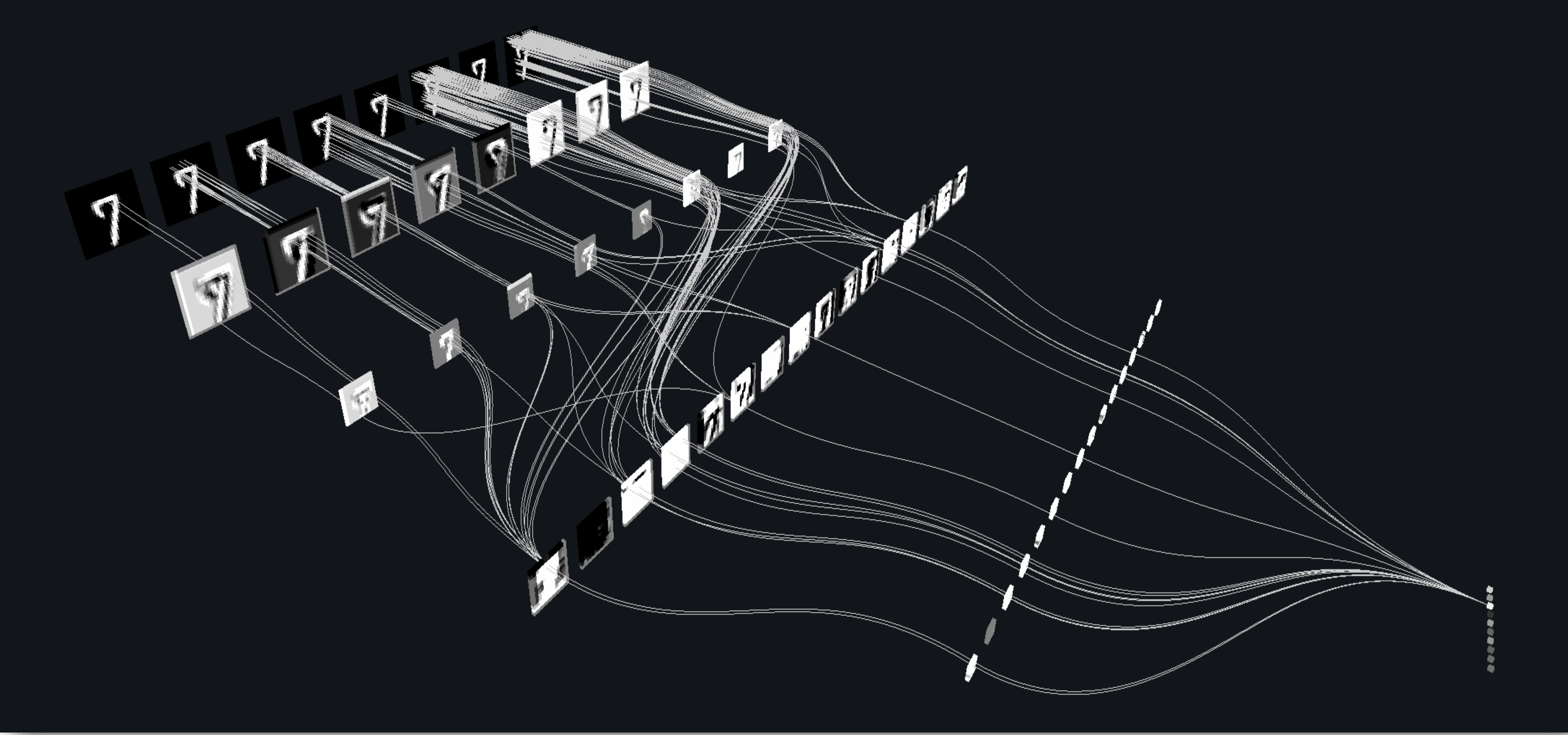
Deep Learning

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)

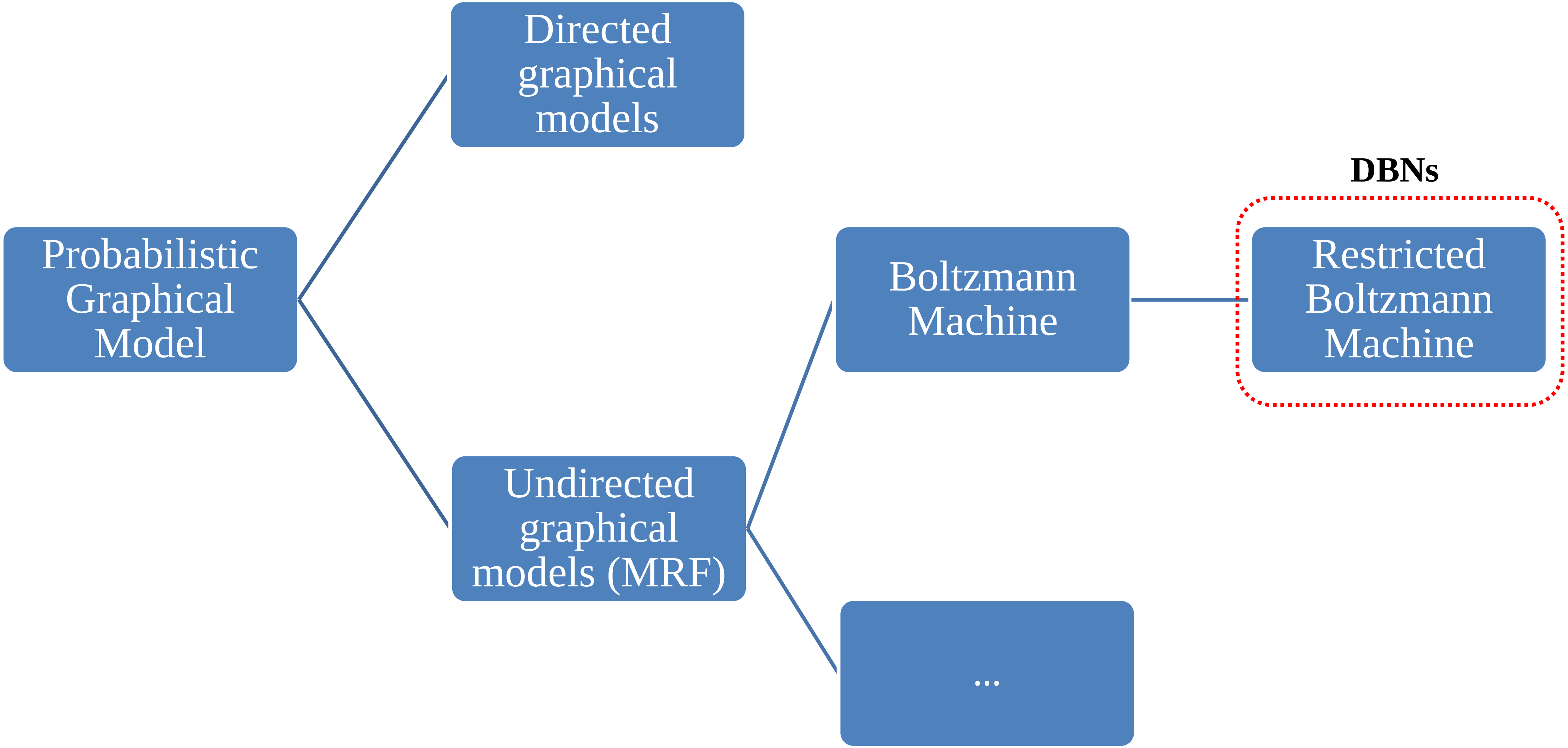
DBN

- DBN’s prerequisite
  - MRF
  - Sampling
  - RBMs

Restricted Boltzmann Machine (RBM)

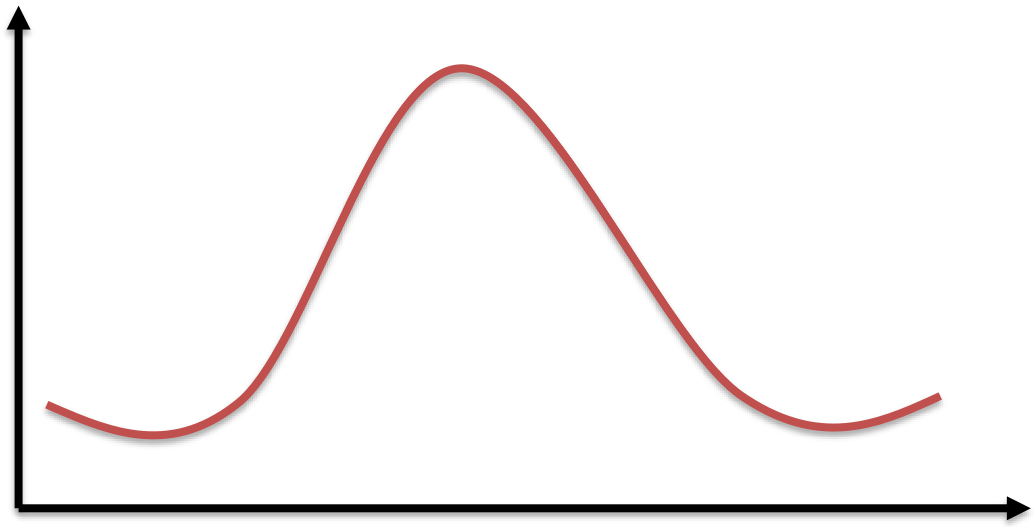
- RBMs are building blocks for the multi-layer leaning architectures, e.g. DBNs.
- RBMs are a special case of general **Boltzmann Machines** (BMs).
- BMs are a particular form of **Markov Random Field (MRF)**, a.k.a. **Markov networks** or **undirected graph models**.

DBNs


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

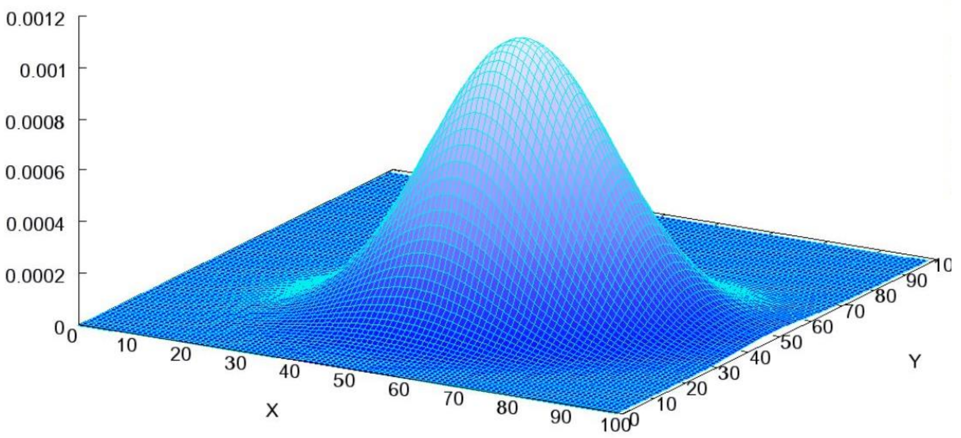
Sampling

- **Methods**
  - **Exact inference**
  - **Approximated inference**
    - Transformation method, rejection sampling, importance sampling, sampling-importance-resampling
    - Gibbs sampling

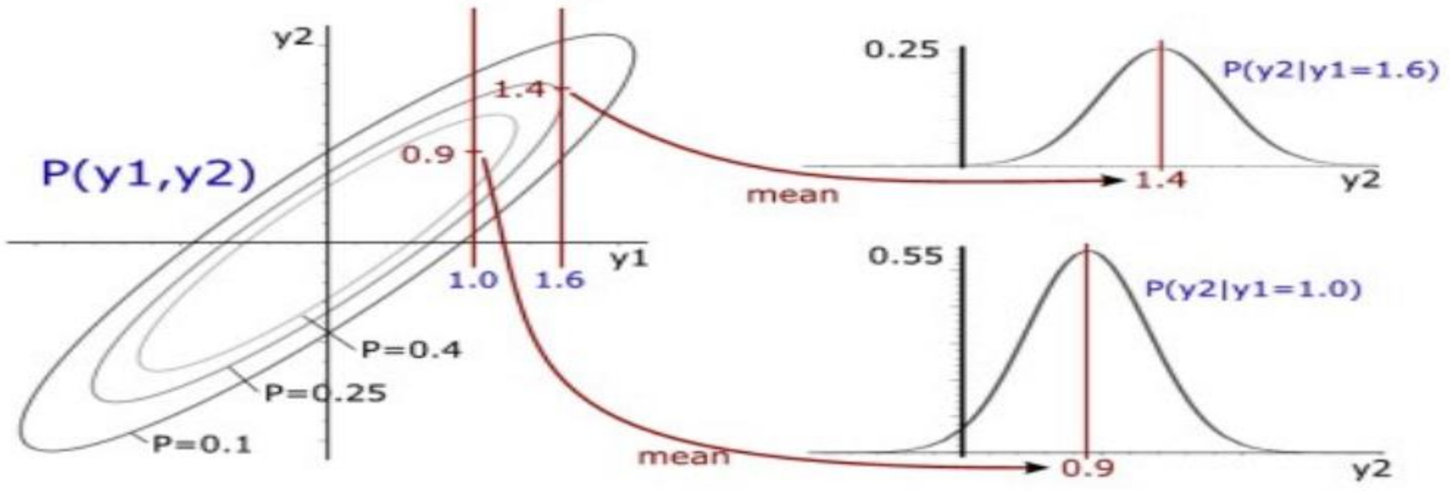


Sampling

- **Posterior $\boldsymbol{p(Z|X)}$**
  - Z: latent variables
  - X: observed data
- Evaluation of expectation (make predictions
- **Methods**:
  - **Exact inference**
  - **Approximated inference**
    - Deterministic variational inference
    - Stochastic (sampling)



Exact Inference



- **Disadvantages**:
  - For most probabilistic models, exact inference is intractable
  - High dimensional latent space
  - Posterior distribution is highly complex (analytical intractable)
  - Summing over all possible configuration of hidden variables (exponentially)

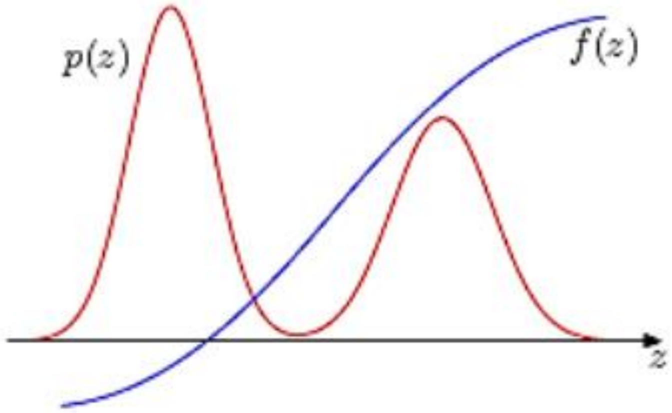
Approximated Inference

- **Sampling**
  - **Problem**: to find the expectation of function $f(x)$ w.r.t. a probability distribution $p(z)$

$$\mathbb{E}[f] = \int f(z)p(z)dz$$

- **Idea**: if we obtain a set of samples $z(l),\ l = 1 \ldots L$ drawn independently from $p(z)$,
  - Then, the expectation may be approximated by:

$$\hat{f} = \frac{1}{L}\sum_{l=1}^{L} f(z^{(l)})$$

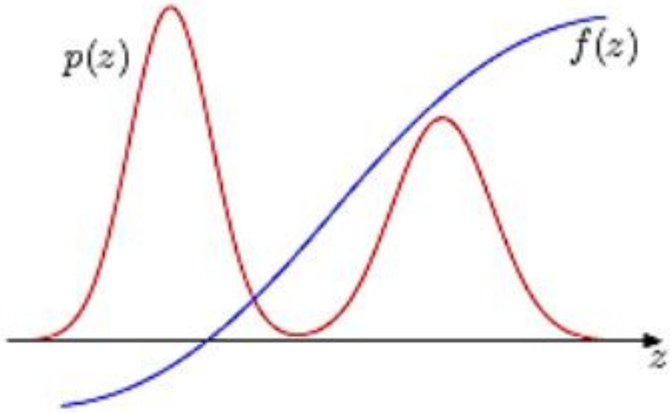


Approximated Inference

- **Sampling**
  - **New problem**: to obtain independent samples from a distribution $p(z)$

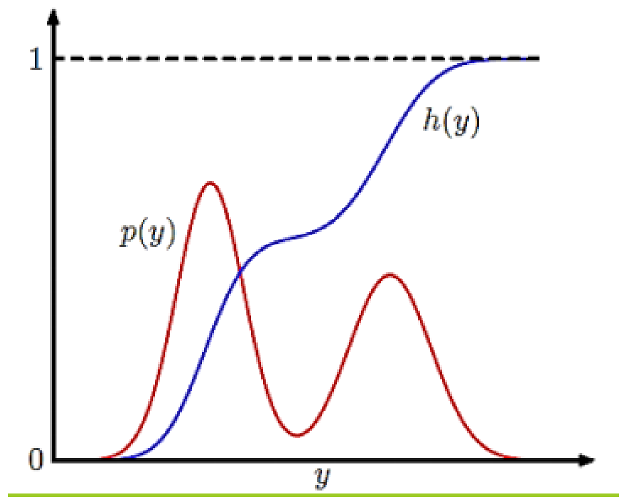
$$\hat{f} = \frac{1}{L}\sum_{l=1}^{L} f(z^{(l)})$$

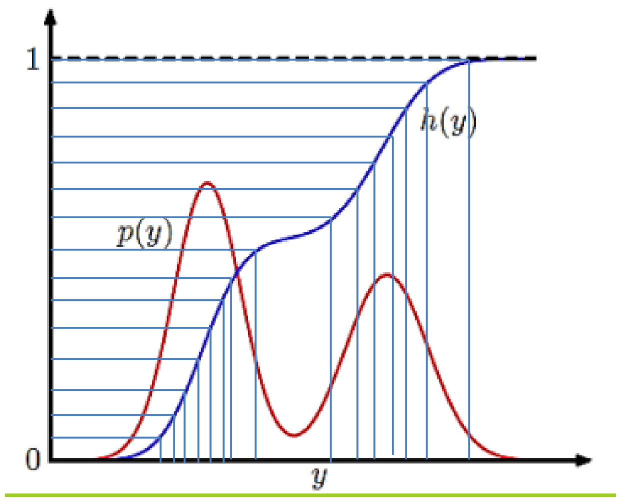
  - **Solutions**:
    - Transformation method
    - Rejection sampling
    - Importance sampling
    - Sampling-importance-resampling



Transformation method

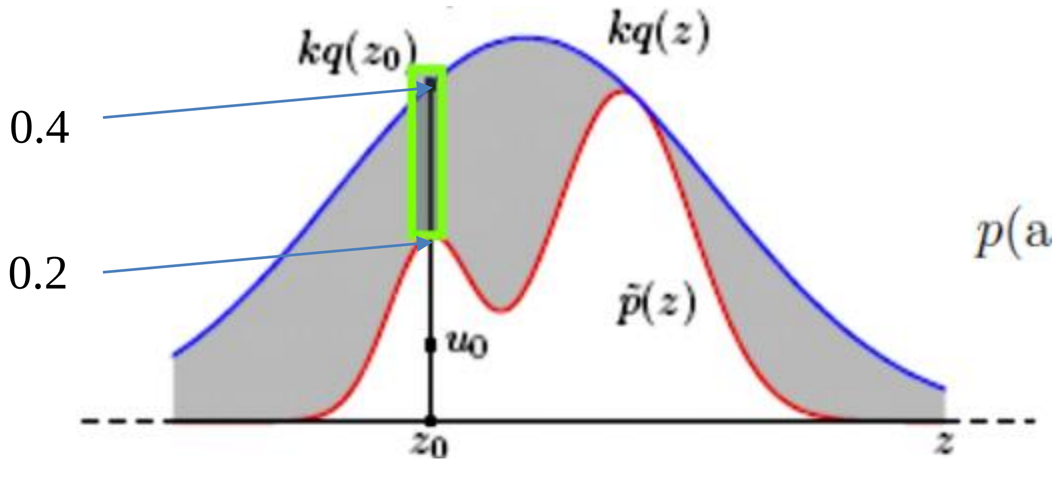
- **Goal**: sampling from $p(.)$
- Suppose the available sample z uniformly distributed over the interval (0,1)
- Transforming z into y using $y = h^{-1}(z)$
  - h: the **cumulative distribution function (CDF)** of p





Rejection Sampling

- Goal: to obtain independent samples from $p(.)$
  - Generate a number of z_0 from $q(.)$
  - Generate a number u_0 from the uniform distribution over $[0, kq(z_0)]$
  - If $u_0 > \tilde{p}(z_0)$ then the sample is rejected, otherwise z_0 is kept
  - The set of kept z are distributed according to $p(.)$



$$p(z) = \frac{1}{Z_p}\tilde{p}(z)$$

$$p(accept) = \int \left\{\frac{\tilde{p}(z)}{k\cancel{q(z)}}\right\} \cancel{q(z)}dz$$

$$= \frac{1}{k}\int \tilde{p}(z)dz$$

Gibbs Sampling

- **Goal**: to sample from $p(z) = p(z_1, \dots, z_M)$
- Prior Knoledge: $p\left(z_i \middle| z_{\setminus \mathrm{i}}\right)$
- In every step:
  - Replace the value of one of the variables z_i by a value drawn from the distribution of that variable conditioned on the values of the remaining variables; $p\left(z_i \middle| z_{\setminus \mathrm{i}}\right)$

Gibbs Sampling

- Example
  - Goal: $p(z_1, z_2, z_3)$
  - For some values for $z_1^{(i)}, z_2^{(i)}, z_3^{(i)}$, Compute $z_1^{(i+1)}, z_2^{(i+1)}, z_3^{(i+1)}$

$$z_1^{(i+1)} \sim p\left(z_1 \middle| z_2^{(i)}, z_3^{(i)}\right)$$

$$z_2^{(i+1)} \sim p\left(z_2 \middle| z_1^{(i+1)}, z_3^{(i)}\right)$$

$$z_3^{(i+1)} \sim p\left(z_3 \middle| z_1^{(i+1)}, z_2^{(i+1)}\right)$$

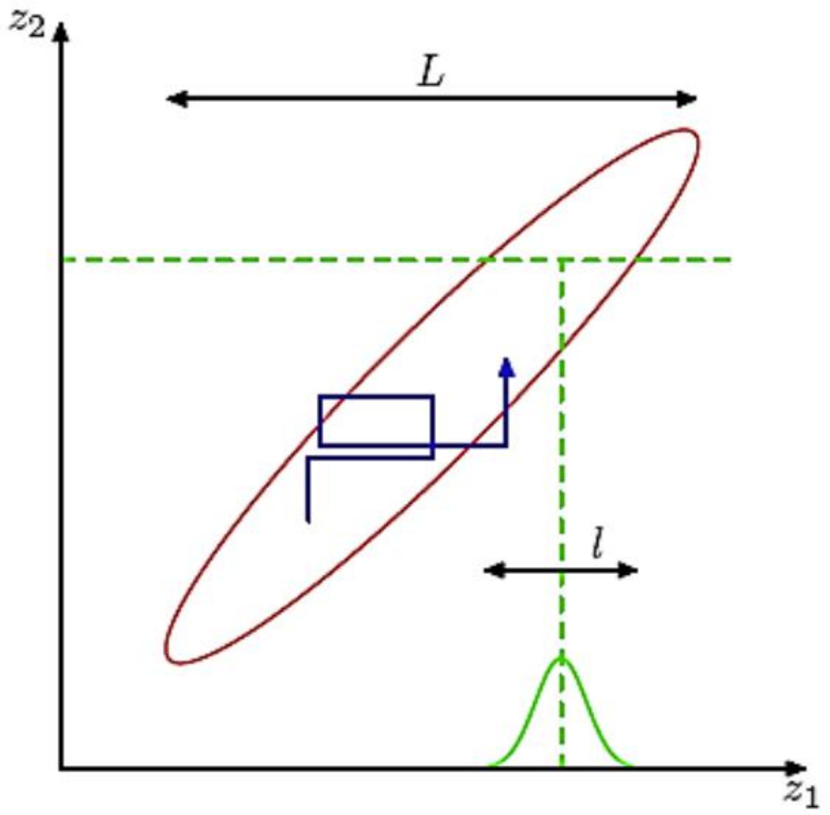
- And repeat till achieve the desired distribution

Gibbs Sampling

1. Initialize $\{z_i : i = 1, \dots, M\}$
2. For $\tau = 1, \dots, T$
   - Sample $z_1^{(\tau+1)} \sim p\left(z_1 \middle| z_2^{(\tau)}, z_3^{(\tau)}, \dots, z_M^{(\tau)}\right)$
   - Sample $z_2^{(\tau+1)} \sim p\left(z_2 \middle| z_1^{(\tau+1)}, z_3^{(\tau)}, \dots, z_M^{(\tau)}\right)$
   - $\vdots$
   - Sample $z_j^{(\tau+1)} \sim p\left(z_j \middle| z_1^{(\tau+1)}, \dots, z_{j-1}^{(\tau+1)}, z_{j+1}^{(\tau)}, \dots, z_M^{(\tau)}\right)$
   - $\vdots$
   - Sample $z_M^{(\tau+1)} \sim p\left(z_M \middle| z_1^{(\tau+1)}, z_2^{(\tau+1)}, \dots, z_{M-1}^{(\tau+1)}\right)$

Gibbs Sampling

- Alternate updated of two variables (blue step) whose distribution is a correlated Gaussian (red)
- The conditional distributions are Gaussian (green Curve)



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