Approximate probabilistic inference:
- Markov Chain Monte Carlo (MCMC)
- Variational methods

Markov chain Monte Carlo
- Importance sampling: samples are generated according to Q and every sample from Q is reweighted according to w, but the Q distribution may be very far from the target
- MCMC is a strategy for generating samples from the target distribution, including conditional distributions
- MCMC:
  - Markov chain defines a sampling process that
  - initially generates samples very different from the target distribution (e.g. posterior)
  - but gradually refines the samples so that they are closer and closer to the posterior.
MCMC

- The construction of a Markov chain requires two basic ingredients
  - a transition matrix $P$
  - an initial distribution $\pi_0$
- Assume a finite set $S=\{1,\ldots,m\}$ of states, then a transition matrix is
  \[
P = \begin{pmatrix}
p_{11} & p_{12} & \cdots & p_{1m} \\
p_{21} & p_{22} & \cdots & p_{2m} \\
\vdots & \vdots & \ddots & \vdots \\
p_{m1} & p_{m2} & \cdots & p_{mm}
\end{pmatrix}
\]
  Where $p_{ij} \geq 0 \quad \forall (i,j) \in S^2$ and $\sum_{j \in S} p_{ij} = 1 \quad \forall i \in S$

Markov Chain

- Markov chain defines a random process of selecting states
  $X^{(0)}, X^{(1)}, \ldots, X^{(m)}, \ldots$

  Initial state selected based on $\pi_0$

  Subsequent states selected based on the previous state and the transition matrix

  $X^{(t)} \rightarrow X^{(t+1)}$

- Chain Dynamics
  \[P^{(t+1)}(X^{(t+1)} = x') = \sum_{x \in \text{Dom}(X)} P^{(t)}(X^{(t)} = x) T(x \rightarrow x')\]
  Probability of a state $x'$ being selected at time $t+1$
MCMC

• **Markov chain** satisfies
  \[ P(X_{n+1} = j \mid X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n) = P(X_{n+1} = j \mid X_n = i_n) \]

• **Irreducibility**: A MC is called irreducible (or un-decomposable) if there is a positive transition probability for all pairs of states within a limited number of steps.
  
  • In irreducible chains there may still exist a periodic structure such that for each state \( i \in \mathcal{S} \), the set of possible return times to \( i \) when starting in \( i \) is a subset of the set \( \mathcal{P} = \{ p, 2p, 3p, \ldots \} \) containing all but a finite set of these elements. The smallest number \( p \) with this property is the so-called **period of the chain**
    
  \[ p = \gcd\{n \in \mathbb{N} : p^{(n)}_{ii} > 0\} \]

MCMC

• **Aperiodicity**: An irreducible chain is called aperiodic (or acyclic) if the period \( p \) equals 1 or, equivalently, if for all pairs of states there is an integer \( n_{ij} \) such that for all \( n \geq n_{ij} \), the probability \( p^{(n)}_{ij} > 0 \).
  
  • If a Markov chain satisfies both **irreducibility and aperiodicity**, then it converges to an invariant distribution \( q(x) \)

  • A Markov chain with transition matrix \( P \) will have an **equilibrium distribution** \( q \) **iff** \( q = qP \).
  
  • A sufficient, but not necessary, condition to ensure a particular \( q(x) \) is the invariant distribution of transition matrix \( P \) is the following **reversibility (detailed balance) condition**
    
  \[ q(x^i)P(x^{i-1} \mid x^i) = q(x^{i-1})P(x^i \mid x^{i-1}) \]
**Objective:** generate samples from the posterior distribution

- **Idea:**
  - Markov chain defines a sampling process that
  - initially generates samples very different from the target posterior
  - but gradually refines the samples so that they are closer and closer to the posterior.

---

**MCMC**

- $P(X|e)$ — the query we want to compute
- $e_1$ & $e_2$ are known evidence
- Sampling from the distribution $P(X)$ is very different from the desired posterior $P(X|e)$
Markov Chain Monte Carlo (MCMC)

- **Goal**: a sample from $P(X|e)$
- Start from some $P(X)$ and generate a sample $x_1$
MCMC (Cont.)

- **Goal:** a sample from \( P(X|e) \)
- Start from some \( P(X) \) and generate a sample \( x_1 \)

\[ X_1 \]

\[ \text{Apply T} \]

MCMC (Cont.)

- **Goal:** a sample from \( P(X|e) \)
- Start from some \( P(X) \) and generate a sample \( x_1 \)
- From \( x_1 \) and transition generate \( x_2 \)

\[ X_1 \rightarrow X_2 \]

\[ \text{Apply T} \]

\[ \text{Apply T} \]
MCMC (Cont.)

- **Goal**: a sample from $P(X|e)$
- Start from some $P(X)$ and generate a sample $x_1$
- From $x_1$ and transition generate $x_2$

\[ X_1 \xrightarrow{\text{Apply T}} X_2 \]

\[ X_2 \xrightarrow{\text{Apply T}} X_3 \]

... 

\[ X_n \xrightarrow{\text{Apply T}} \]

---

MCMC (Cont.)

- **Goal**: a sample from $P(X|e)$
- Start from some $P(X)$ and generate a sample $x_1$
- From $x_1$ and transition generate $x_2$
- Repeat for $n$ steps

\[ P'(X|e) \]

\[ X_1 \xrightarrow{\text{Apply T}} X_2 \]

... 

\[ X_n \xrightarrow{\text{Apply T}} \]
**MCMC (Cont.)**

- **Goal:** a sample from $P(X|e)$
- Start from some $P(X)$ and generate a sample $x_1$
- From $x_1$ and transition generate $x_2$
- Repeat for $n$ steps

\[ P'(X|e) \]

\[ X_1 \rightarrow X_2 \rightarrow \cdots \rightarrow X_n \]

Apply T  Apply T  Apply T

---

**MCMC (Cont.)**

- **Goal:** a sample from $P(X|e)$
- Start from some $P(X)$ and generate a sample $x_1$
- From $x_1$ and transition generate $x_2$
- Repeat for $n$ steps

\[ P'(X|e) \]

\[ X_1 \rightarrow X_2 \rightarrow \cdots \rightarrow X_n \rightarrow X_{n+1} \rightarrow X_{n+2} \rightarrow \cdots \]

Apply T  Apply T  Apply T

---
MCMC

- In general, an MCMC sampling process doesn’t have to converge to a stationary distribution
- A finite state Markov Chain has a unique stationary distribution iff the Markov chain is regular
  - regular: exist some \( k \), for each pair of states \( x \) and \( x' \), the probability of getting from \( x \) to \( x' \) in exactly \( k \) steps is greater than 0
- We want Markov chains that converge to a unique target distribution from any initial state

**Big question:**
- How to build such Markov chains?

Gibbs Sampling

- A simple method to define MC for BBN can benefit from the structure (independences) in the network

- **Evidence:**
  - \( x_5 = T \)
  - \( x_6 = T \)
- all variables have binary values T or F
Gibbs Sampling

Initial state

$x_0$

Update Value of $x_4$

$x_1 = F, x_2 = T$

$x_3 = T, x_4 = T$

$x_5 = x_6 = T$ (Fixed)

$x_1 = F, x_2 = T$

$x_3 = T, x_4 = T$

$x_5 = x_6 = T$ (Fixed)
Gibbs Sampling

\[ x_0 \rightarrow x_1 \]

\[ x_1 = F, \ x_2 = T, \ x_3 = T, \]
\[ x_4 = F, \ x_5 = T, \ x_6 = T \]

CS 3750 Advanced Machine Learning
Gibbs Sampling

After many reassignments

Samples from desired $P(X_{rest} \mid e)$
**Gibbs Sampling**

Keep resampling each variable using the value of variables in its local neighborhood (Markov blanket).

\[ P(X_4 \mid x_2, x_3, x_5, x_6) \]

**Gibbs Sampling**

- Gibbs sampling takes advantage of the graphical model structure
- Markov blanket makes the variable independent from the rest of the network

\[ P(X_4 \mid x_2, x_3, x_5, x_6) \]
Building a Markov Chain

• **A reversible Markov chain:**
  - A sufficient, but not necessary, condition to ensure a particular $q(x)$ is the invariant distribution of transition matrix $P$ is the following reversibility (detailed balance) condition
    \[ q(x^i)P(x^{i-1}|x^i) = q(x^{i-1})P(x^i|x^{i-1}) \]

• **Metropolis-Hastings algorithm**
  - builds a reversible Markov Chain
  - Uses a proposal distribution to generate candidate states
    - Either accept it and take a transition to state $x'$
    - Or reject it and stay at current state $x$

---

**Building a Markov Chain**

• **Metropolis-Hastings algorithm**
  - builds a reversible Markov Chain
  - uses the **proposal distribution** (similar to proposal the distribution in importance sampling) to generate candidates for $x'$
    - A proposal distribution $Q$: $T^Q (x \rightarrow x')$
    - Example: Uniform over the values of variables
  - Either accept a proposal and take a transition to state $x'$
  - Or reject it and stay at current state $x$
    - Acceptance probability
      \[ A(x \rightarrow x') \]
Building a Markov Chain

• Transition for the MH:

\[ T(x \rightarrow x') = T^Q(x \rightarrow x')A(x \rightarrow x') \quad \text{if} \quad x \neq x' \]

\[ T(x \rightarrow x) = T^Q(x \rightarrow x) + \sum_{x \neq x} T^Q(x \rightarrow x')(1 - A(x \rightarrow x')) \quad \text{otherwise} \]

• From reversibility condition:

\[ q(x)T(x \rightarrow x') = q(x')T(x' \rightarrow x) \]

• We get

\[ A(x \rightarrow x') = \min[1, \frac{q(x')T^Q(x' \rightarrow x)}{q(x)T^Q(x \rightarrow x')} ] \]

Building a Markov Chain

• Comparing Metropolis Hastings with Gibbs sampling
  – For Gibbs

\[ A(u_i, x_i \rightarrow u_i, x_i') = \min[1, \frac{P(x_i' | u_i)T^Q(u_i, x_i' \rightarrow u_i, x_i)}{P(x_i | u_i)T^Q(u_i, x_i \rightarrow u_i, x_i)}] \]

\[ = \min[1, \frac{P(x_i' | u_i)P(x_i | u_i)}{P(x_i | u_i)P(x_i' | u_i)}] = \min[1, 1] = 1 \]

  – Special MH, for which acceptance probability is 1.
**Metropolis Hastings algorithm**

- **Assumptions:**
  - We can’t draw the samples from \( q(x) \)
  - We can evaluate \( q(x) \) for any \( x \)
- We use a Markov chain that moves towards \( x^* \) with acceptance probability

\[
A(x, x^*) = \min \left[ 1, \frac{q(x^*) p(x | x^*)}{q(x) p(x^* | x)} \right]
\]

- The transition kernel defined by this process satisfies the detailed balance condition

---

**Mixing Time in Using Markov Chain**

- **Mixing Time**
  - The number of steps we take until we collect a sample from the target distribution. \((# = n)\)
Summary

- **Markov Chain Monte Carlo method** attempts to generate samples from posterior distribution.
- **Metropolis Hastings algorithm** is a general scheme for specifying a Markov chain.
- **Gibbs sampling** is a special case that takes advantage of the network structure (Markov Blanket).

Variational approximations
Variational approximation

Assume we have a function \( f(Z) \) that is hard to calculate.

**Example:** posterior probability in a complex BBNs

\[ P(Z \mid X) \]

- this inference can be very hard

**Idea:** replace calculations of \( f(Z) \) with an optimization over a simpler parametric function \( q(Z \mid \lambda) \)

\[ f(Z) \sim \max_{\lambda} q(Z \mid \lambda) \]

---

Variational lower bound

Let \( X \) denote observed variables and \( Z \) denote target variables

\[ P(Z \mid X) = \frac{P(X,Z)}{P(X)} \]

\[ \log P(Z \mid X) = \log P(X,Z) - \log P(X) \]

\[ \log P(X) = \log P(X,Z) - \log P(Z \mid X) \]

Assume some distribution: \( Q_\theta(Z \mid X) \) defined by parameters \( \theta \)

**Average both sides** with \( E_{Q_\theta} \)

\[
\sum_z Q_\theta(Z \mid X) \log P(X) = \sum_z Q_\theta(Z \mid X) \log P(X,Z) - \sum_z Q_\theta(Z \mid X) \log P(Z \mid X) \\
\log P(X) = E_{Q_\theta} (\log P(X,Z)) - E_{Q_\theta} (\log P(Z \mid X))
\]
Variational lower bound

\[ \log P(X) = E_{Q_\theta} \left( \log P(X, Z) \right) - E_{Q_\theta} \left( \log P(Z \mid X) \right) \]

\[ \log P(X) = \sum_z Q_\theta(Z \mid X) \log P(X, Z) - \sum_z Q_\theta(Z \mid X) \log P(Z \mid X) \]

\[ \log P(X) = \sum_z Q_\theta(Z \mid X) \log P(X, Z) - \sum_z Q_\theta(Z \mid X) \log P(Z \mid X) \]

\[ + \sum_z Q_\theta(Z \mid X) \log Q_\theta(Z \mid X) - \sum_z Q_\theta(Z \mid X) \log Q_\theta(Z \mid X) \]

**Kullback-Leibler divergence:** distance between 2 distributions

\[ KL(Q \mid P) = \sum_z Q_\theta(Z \mid X) \log Q_\theta(Z \mid X) - \sum_z Q_\theta(Z \mid X) \log P(Z \mid X) \]

**Functional (Evidence lower bound or ELBO):**

\[ F(Q, P) = \sum_z Q_\theta(Z \mid X) \log P(X, Z) - \sum_z Q_\theta(Z \mid X) \log Q_\theta(Z \mid X) \]

\[ \log P(X) = F(Q, P) + KL(Q \mid P) \]

**Variational lower bound**

\[ \log P(X) = F(Q, P) + KL(Q \mid P) \]

\[ \downarrow \]

**distance between** \( Q_\theta(Z \mid X), P(Z \mid X) \)

Always \( \geq 0 \)

Equals 0 if \( Q_\theta(Z \mid X) = P(Z \mid X) \)

We can optimize the approximation \( Q_\theta(Z \mid X) \) by minimizing

\[ \min_\theta KL(Q_\theta \mid P) \]

We can also do this by maximizing \( F(Q_\theta, P) \)

\[ \max_\theta F(Q_\theta, P) \quad \text{Often much easier} \]

\[ F(Q, P) = \sum_z Q_\theta(Z \mid X) \log P(X, Z) - \sum_z Q_\theta(Z \mid X) \log Q_\theta(Z \mid X) \]
Latent variable models

Let $X$ denote observed variables and $Z$ denote hidden (latent variables)

Inference opposite the links is hard: $P(Z \mid X)$

**Solution:** Define a simpler distribution: $Q_{\theta}(Z \mid X)$ to approximate $P(Z \mid X)$

**Optimize:**

$$
\max_{\theta} ~ F(Q_{\theta}, P)
$$

$$
F(Q, P) = \sum_{z} Q_{\theta}(Z \mid X) \log P(X \mid Z) + \sum_{z} Q_{\theta}(Z \mid X) \log P(Z) - \sum_{z} Q_{\theta}(Z \mid X) \log Q_{\theta}(Z \mid X)
$$

Mean field approximation

How to construct approximation of $Q_{\theta}(Z \mid X)$

Mean field approximation:

$$
Q_{\theta}(Z \mid X) = \prod_{i} Q_{i}(Z_{i} \mid \theta_{i})
$$

**Optimize:**

$$
\max_{\theta} ~ \sum_{z_{1}, z_{2}, \ldots, z_{i}} Q_{\theta}(Z \mid X) \log P(X, Z) - \sum_{z_{1}, z_{2}, \ldots, z_{i}} Q_{\theta}(Z \mid X) \log Q_{\theta}(Z \mid X)
$$

$$
\max_{\theta_{1}, \theta_{2}, \ldots, \theta_{i}} ~ \sum_{z_{1}, z_{2}, \ldots, z_{i}} \prod_{i} Q_{i}(Z_{i} \mid \theta_{i}) \log P(X, Z)
$$

$$
- \sum_{z_{1}, z_{2}, \ldots, z_{i}} \prod_{i} Q_{i}(Z_{i} \mid \theta_{i}) \log \prod_{i} Q_{i}(Z_{i} \mid \theta_{i})
$$
Latent variable models

Let $X$ denote observed variables and $Z$ denote hidden (latent variables)

\[ Q_\theta(Z \mid X) = \prod_i Q_i(Z_i \mid \theta_i) \]

\[
\max_\theta F(Q_\theta, P) \]

\[
\max_{\theta_1, \theta_2, \ldots, \theta_l} \sum_{Z_1, Z_2, \ldots, Z_l} \prod_i Q_i(Z_i \mid \theta_i) \log P(X \mid Z) \\
+ \sum_{Z_i} \prod_i Q_i(Z_i \mid \theta_i) \log P(Z_i) \\
- \sum_{Z_i} \prod_i Q_i(Z_i \mid \theta_i) \log \prod_i Q_i(Z_i \mid \theta_i)
\]

Express analytically $F$, differentiate wrt parameters and set to 0

\[ \Rightarrow \text{Mean field equations that can be used to get optimal set } \theta \]