Markov Chain Monte Carlo Methods

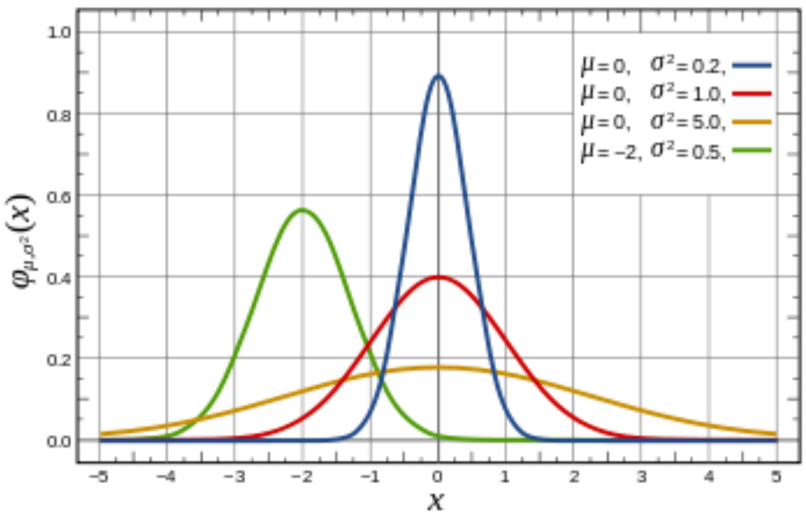
Jianlin Cheng, PhD
Computer Science Department
University of Missouri, Columbia

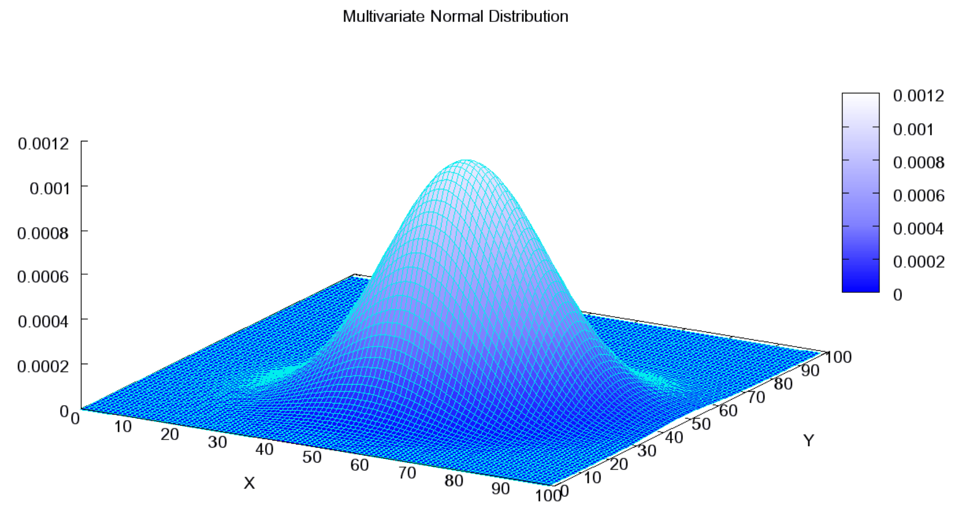
Fall, 2013

Adapted from Eric Xing's slides at CMU

Distribution of Random Variables

Random variables: GPA, wage, age, ???





$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

http://en.wikipedia.org/wiki/Normal_distribution

http://en.wikipedia.org/wiki/Multivariate_normal_distribution

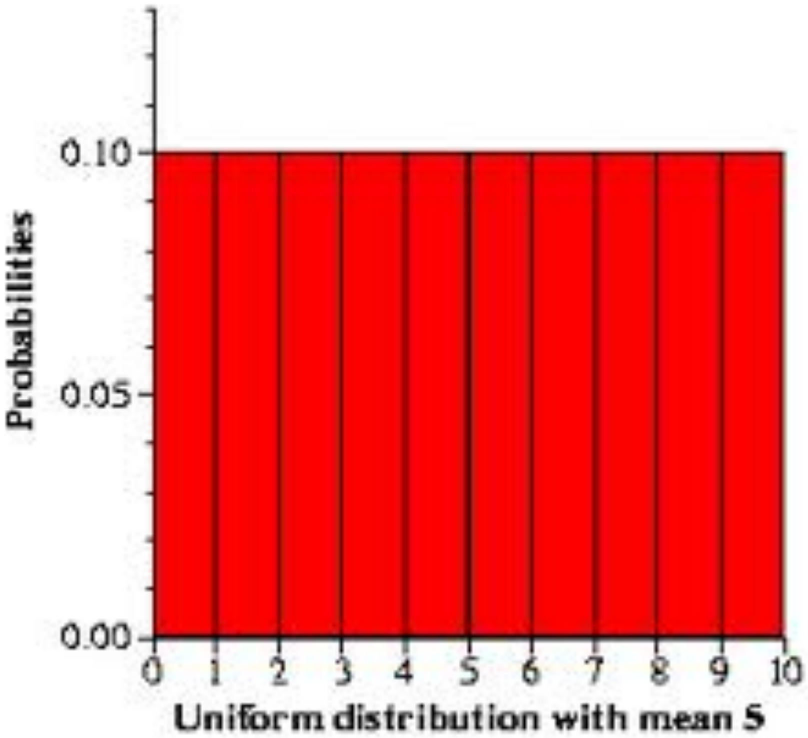
Distribution of multiple variables can be very complicated

- Fever, gender, cough, chest pain, lung cancer
- Alarm, earthquake, burglary, neighbors' call
- GRE, TOEFL, GPA, gender, ideal job offer
- Color (R, G, B) in an image
- ???

Problem: most likely values, expected values, probability / frequency

Sampling (Simulation)

- Generate data from a distribution



How to sample data from it using a computer?
How to sample a random number between 0 and 1?

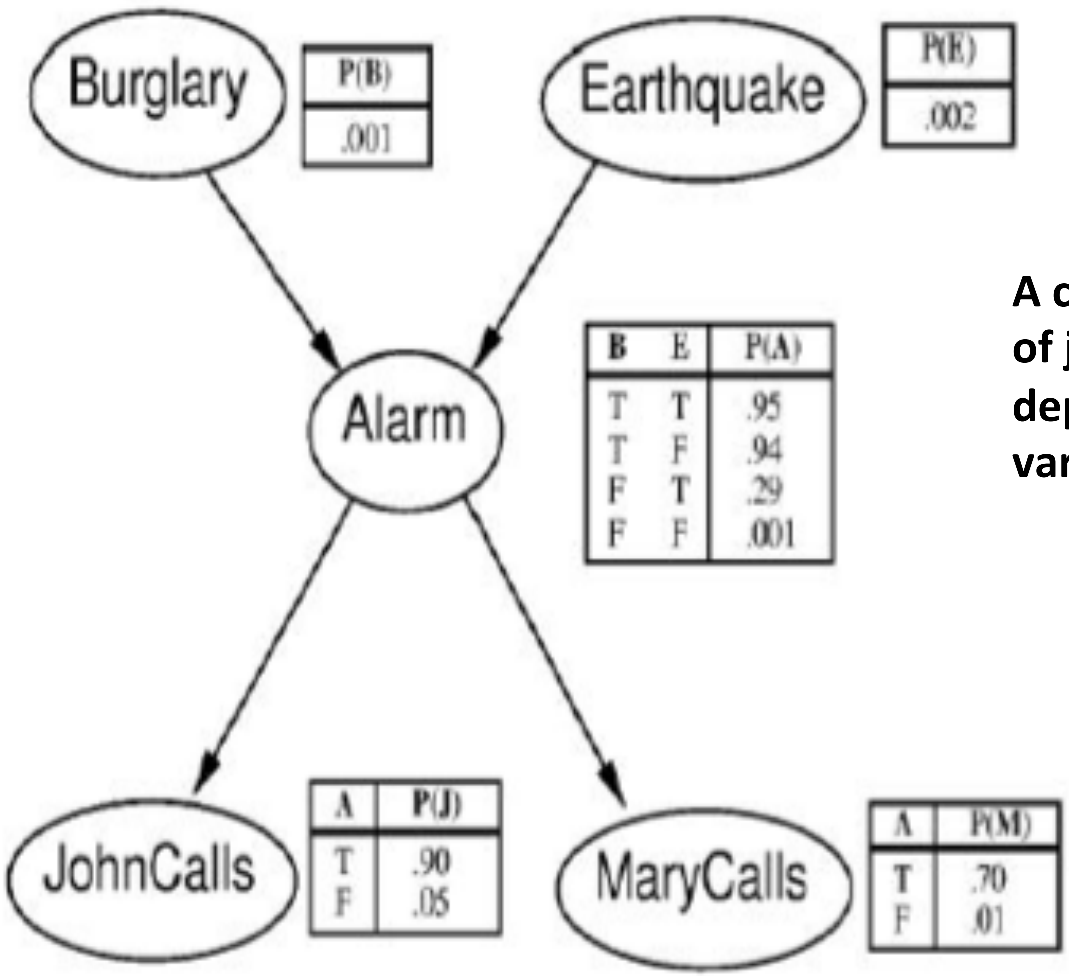
Monte Carlo Methods

- Draw random samples from the desired distribution
- Yield a stochastic representation of a complex distribution
  - marginals and other expections can be approximated using sample-based averages

$$E[f(x)] = \frac{1}{N}\sum_{t=1}^{N} f(x^{(t)})$$

- **Asymptotically** exact and easy to apply to arbitrary models
- Challenges:
  - how to draw samples from a given dist. (not all distributions can be trivially sampled)?
  - how to make better use of the samples (not all sample are useful, or eqally useful, see an example later)?
  - how to know we've sampled enough?

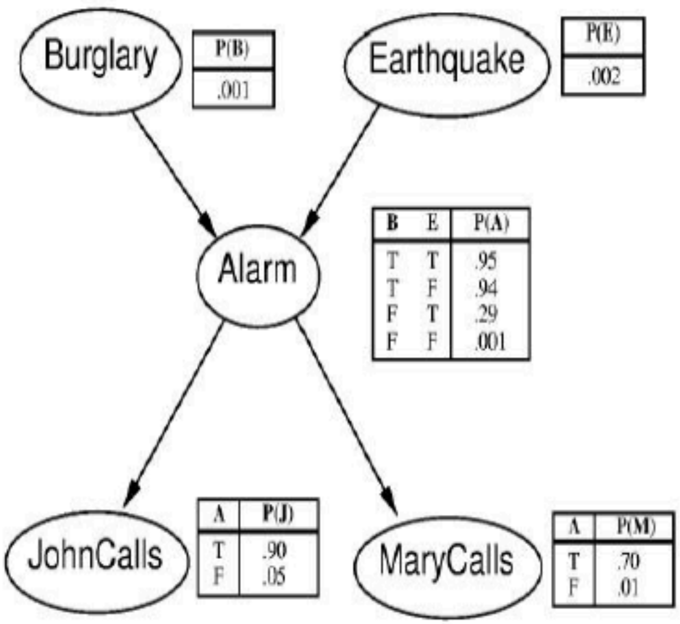
Bayesian Network (BN)



A concise, graphic representation of joint distribution and dependency of a set of variables.

Example: naïve sampling

- Construct samples according to probabilities given in a BN.



Alarm example: (Choose the right sampling sequence)
1) Sampling:P(B)=<0.001, 0.999> suppose it is false, B0. Same for E0. P(A|B0, E0)=<0.001, 0.999> suppose it is false...
2) Frequency counting: In the samples right,
P(J|A0)=P(J,A0)/P(A0)=<1/9, 8/9>.

E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E0	B0	A0	M0	J1
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E1	B0	A1	M1	J1
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0

Example: naïve sampling

- Construct samples according to probabilities given in a BN.

Alarm example: (Choose the right sampling sequence)

3) what if we want to compute P(J|A1) ?
we have only one sample ...
P(J|A1)=P(J,A1)/P(A1)=<0, 1>.

4) what if we want to compute P(J|B1) ?
No such sample available!
P(J|A1)=P(J,B1)/P(B1) can not be defined.

For a model with hundreds or more variables, rare events will be very hard to garner evough samples even after a long time or sampling ...

E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E0	B0	A0	M0	J1
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E1	B0	A1	M1	J1
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0
E0	B0	A0	M0	J0

Monte Carlo Methods

- Direct Sampling
  - We have seen it.
  - Very difficult to populate a high-dimensional state space
- Rejection Sampling
  - Create samples like direct sampling, only count samples which is consistent with given evidences.
- Likelihood weighting, ... (Importance Sampling)
  - Sample variables and calculate evidence weight. Only create the samples which support the evidences.
- Markov chain Monte Carlo (MCMC)
  - Metropolis-Hasting
  - Gibbs

Rejection Sampling

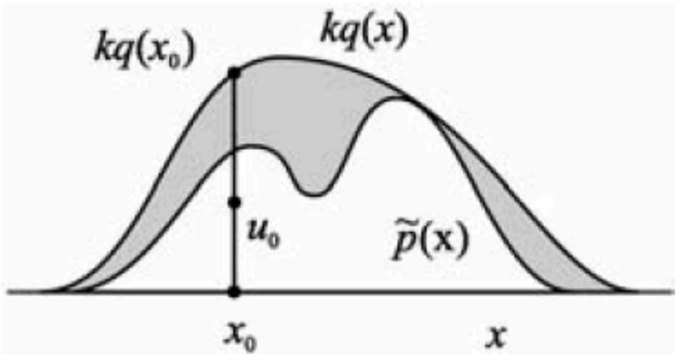
- Suppose we wish to sample from dist. $\Pi(X)=\Pi'(X)/Z$.
  - $\Pi(X)$ is difficult to sample, but $\Pi'(X)$ is easy to evaluate
  - Sample from a simpler dist $Q(X)$
  - Rejection sampling

$$x^* \sim Q(X), \qquad \text{accept } x^* \text{ w.p. } \Pi'(x^*)/kQ(x^*)$$

  - Correctness:

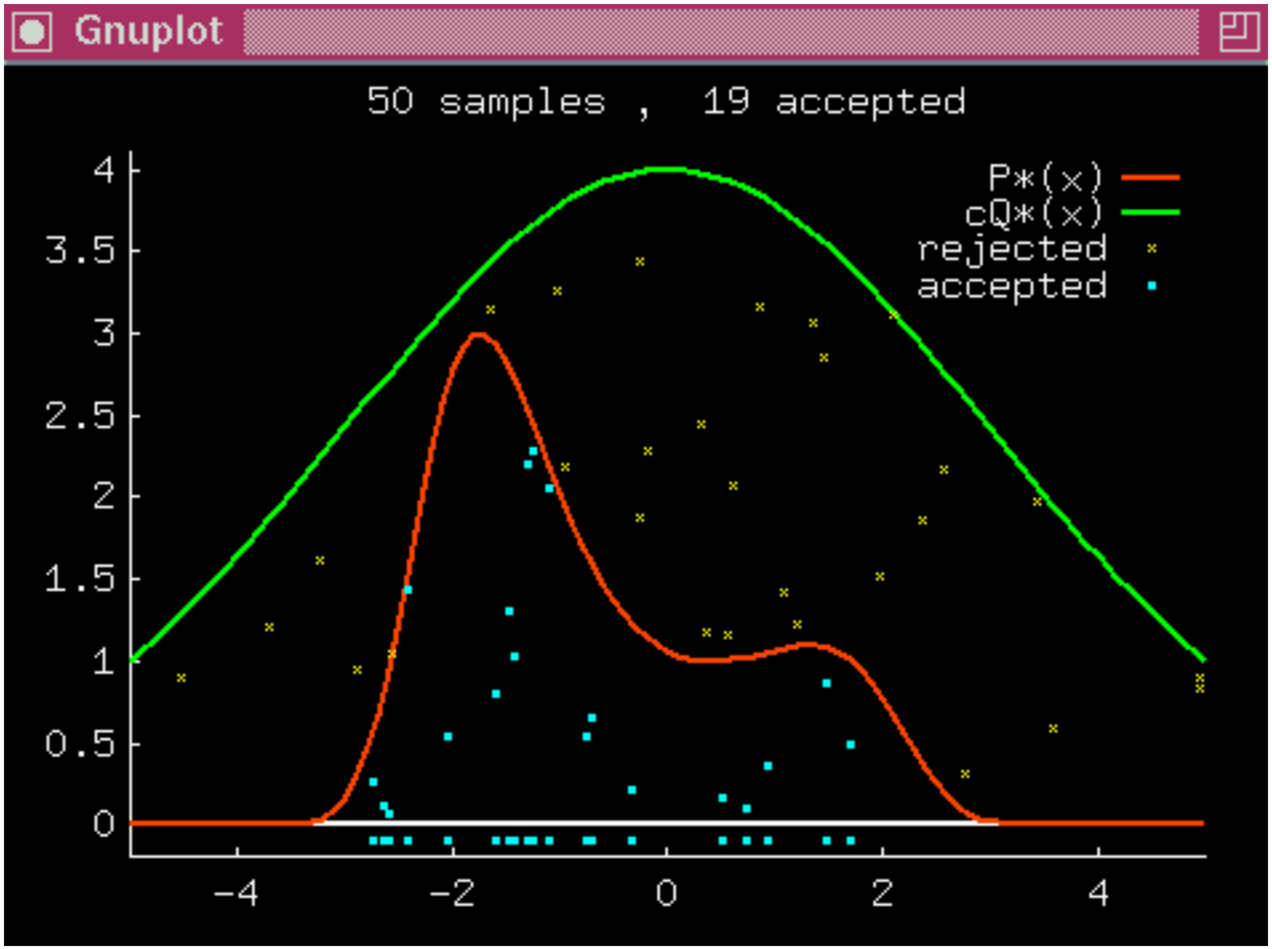
$$p(x) = \frac{[\Pi'(x)/kQ(x)]Q(x)}{\int [\Pi'(x)/kQ(x)]Q(x)dx} = \frac{\Pi'(x)}{\int \Pi'(x)dx} = \Pi(x)$$

  - Pitfall …



What kind of X is more likely accepted?

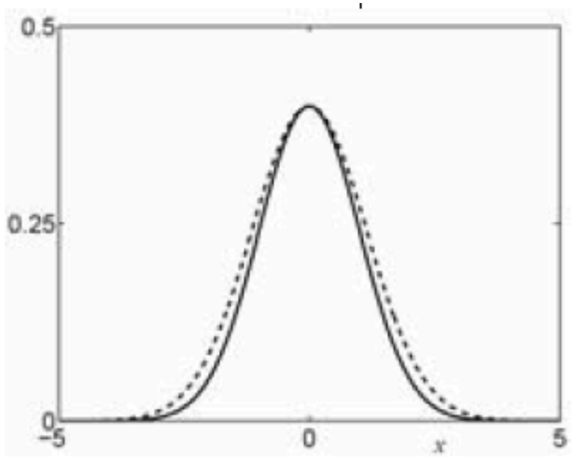
An Example of Rejection Sampling



What is the potential pitfalls of rejection sampling?

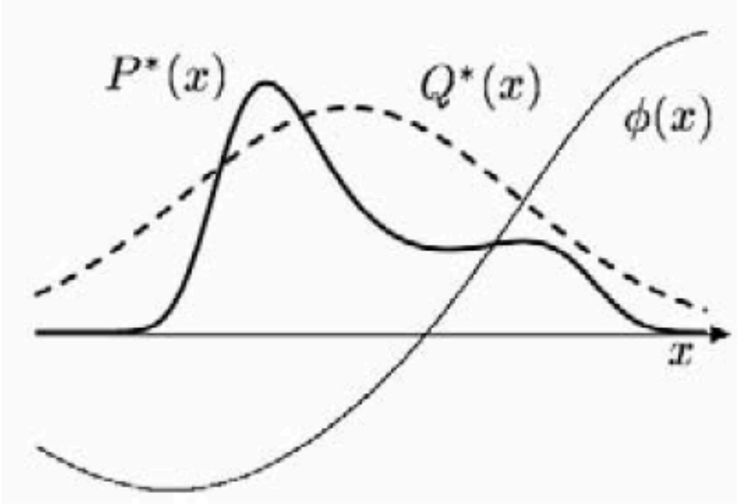
Rejection Sampling

- Pitfall:
  - Using $Q=\mathcal{N}(\mu,\sigma_q I)$ to sample $P=\mathcal{N}(\mu,\sigma_p I)$
  - If σ_q exceeds σ_p by 1%, and dimensional=1000,
  - The optimal acceptance rate $k=(\sigma_q/\sigma_p)^d \approx 1/20{,}000$
  - Big waste of samples!



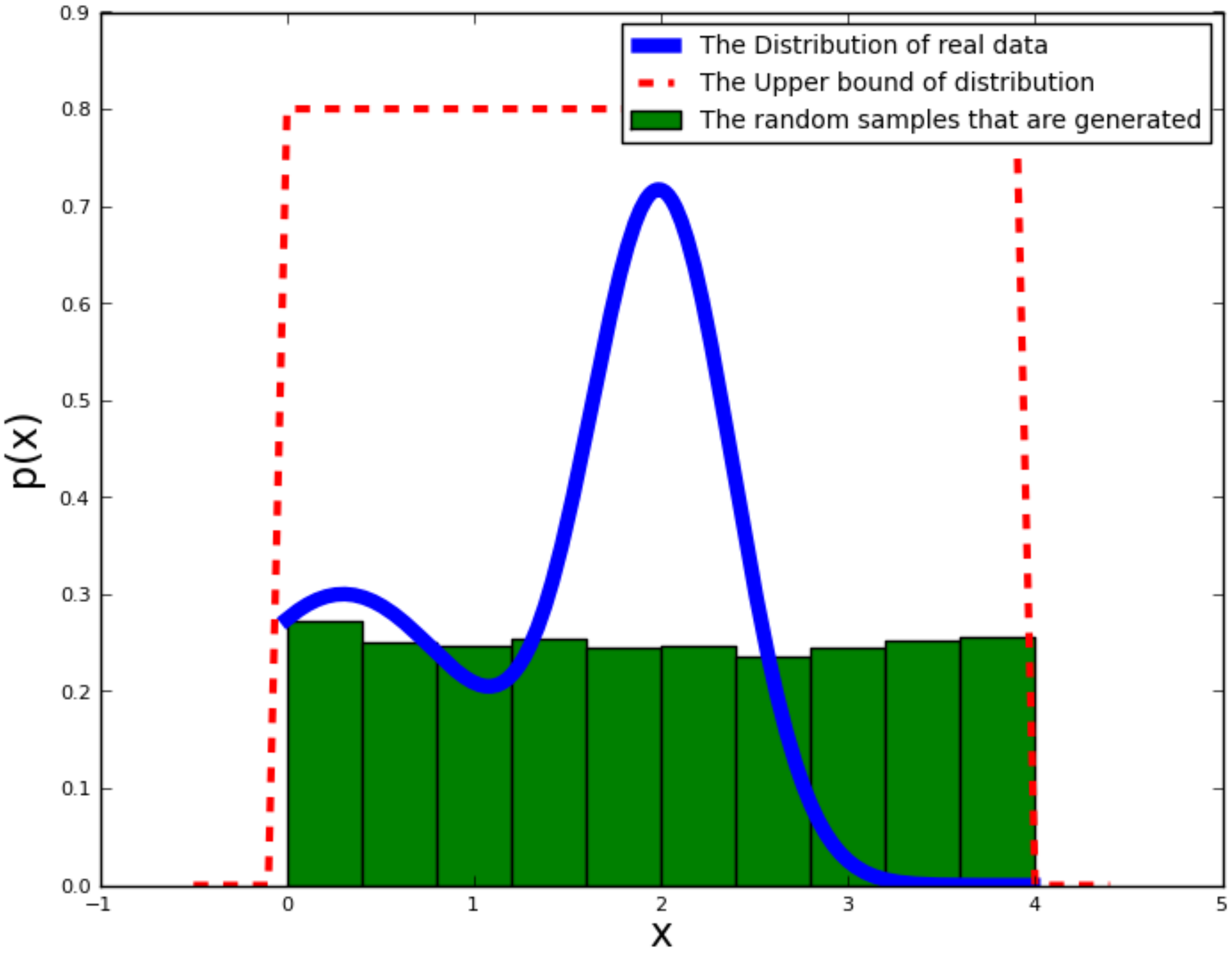
Importance sampling

- Suppose sampling from $P(\cdot)$ is hard.
- Suppose we can sample from a "simpler" proposal distribution $Q(\cdot)$ instead.
- If Q dominates P (i.e., $Q(x) > 0$ whenever $P(x) > 0$), we can sample from Q and reweight:



$$
\begin{aligned}
\langle f(X) \rangle &= \int f(x) P(x) dx \\
&= \int f(x) \frac{P(x)}{Q(x)} Q(x) dx \\
&\approx \frac{1}{M} \sum_m f(x^m) \frac{P(x^m)}{Q(x^m)} \quad \text{where } x^m \sim Q(X) \\
&= \frac{1}{M} \sum_m f(x^m) w^m
\end{aligned}
$$

Importance Sampling



Question

- What is the main difference between rejection sampling and importance sampling?

Markov Chain Monte Carlo (MCMC)

- Importance sampling does not scale well to high dimension
- MCMC is an alternative
- Construct a Markov chain whose *stationary distribution* is the *target density = P(X)*
- Run for T samples until the chain converges / mixes / reaches stationary distribution
- Then collect M samples.
- Key issues: designing proposals so that the chain mixes rapidly, diagnosing convergence.

Markov Chains

- **Definition:**
  - Given an n-dimensional state space
  - Random vector $\mathbf{X} = (x_1,\ldots,x_n)$
  - $\mathbf{x}^{(t)} = \mathbf{x}$ at time-step t
  - $\mathbf{x}^{(t)}$ transitions to $\mathbf{x}^{(t+1)}$ with prob

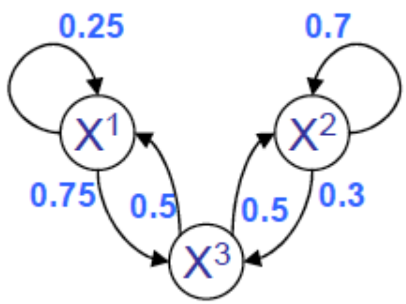
$$P(\mathbf{x}^{(t+1)} \mid \mathbf{x}^{(t)},\ldots,\mathbf{x}^{(1)}) = T(\mathbf{x}^{(t+1)} \mid \mathbf{x}^{(t)}) = T(\mathbf{x}^{(t)} \rightarrow \mathbf{x}^{(t+1)})$$

- **Homogenous**: chain determined by state $\mathbf{x}^{(0)}$, fixed *transition kernel* T (rows sum to 1)
- **Equilibrium**: $\pi(\mathbf{x})$ is a *stationary (equilibrium) distribution* if

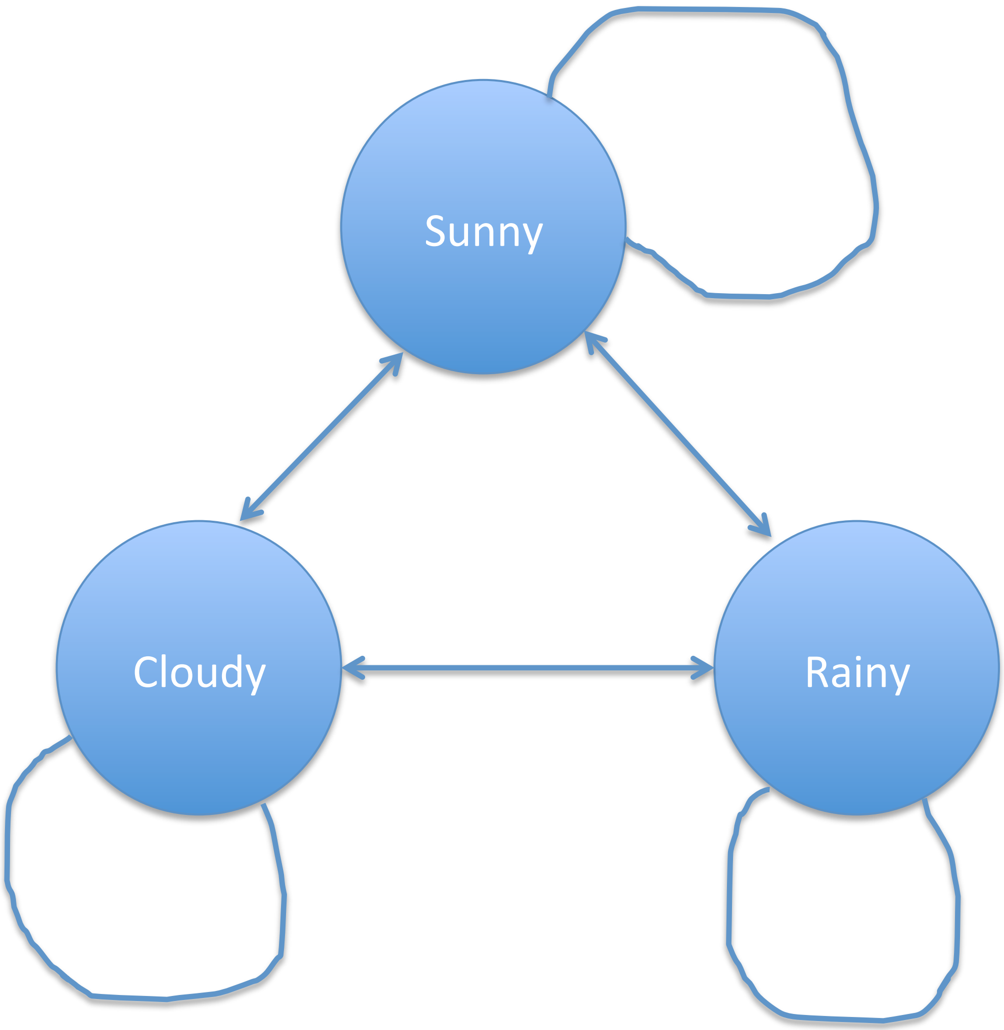
$$\pi(\mathbf{x}') = \Sigma_{\mathbf{x}} \pi(\mathbf{x})\, T(\mathbf{x} \rightarrow \mathbf{x}').$$

i.e., is a left eigenvector of the transition matrix $\pi^T T = \pi^T T$.

$$(0.2 \quad 0.5 \quad 0.3) = (0.2 \quad 0.5 \quad 0.3)\begin{pmatrix} 0.25 & 0 & 0.75 \\ 0 & 0.7 & 0.3 \\ 0.5 & 0.5 & 0 \end{pmatrix}$$



Markov Chain Example



Another example of Markov Chain?

Markov Chain Examples

Markov Chains

- An MC is ***irreducible*** if transition graph connected
- An MC is ***aperiodic*** if it is not trapped in cycles
- An MC is ***ergodic*** (regular) if you can get from state x to x' in a finite number of steps.
- **Detailed balance**: prob($\mathbf{x}^{(t)} \rightarrow \mathbf{x}^{(i-1)}$) = prob($\mathbf{x}^{(t-1)} \rightarrow \mathbf{x}^{(t)}$)

$$p(\mathbf{x}^{(t)})T(\mathbf{x}^{(t-1)} \mid \mathbf{x}^{(t)}) = p(\mathbf{x}^{(t-1)})T(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-1)})$$

summing over $\mathbf{x}^{(t-1)}$

$$p(\mathbf{x}^{(t)}) = \sum_{\mathbf{x}^{(t-1)}} p(\mathbf{x}^{(t-1)})T(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-1)})$$

- Detailed bal → stationary dist exists

Markov Chain Examples

Irreducible?

Aperiodic?

Ergodic?

Detailed balance?

Metropolis-Hastings

- Treat the target distribution as stationary distribution
- Sample from an easier proposal distribution, followed by an acceptance test
- This induces a transition matrix that satisfies detailed balance

  - MH proposes moves according to $Q(x'|x)$ and accepts samples with probability $A(x'|x)$.
  - The induced transition matrix is $T(x \to x') = Q(x'|x)A(x'|x)$
  - Detailed balance means

$$\pi(x)Q(x'|x)A(x'|x) = \pi(x')Q(x|x')A(x|x')$$

  - Hence the acceptance ratio is

$$A(x'|x) = \min\left(1, \frac{\pi(x')Q(x|x')}{\pi(x)Q(x'|x)}\right)$$

MCMC algorithm

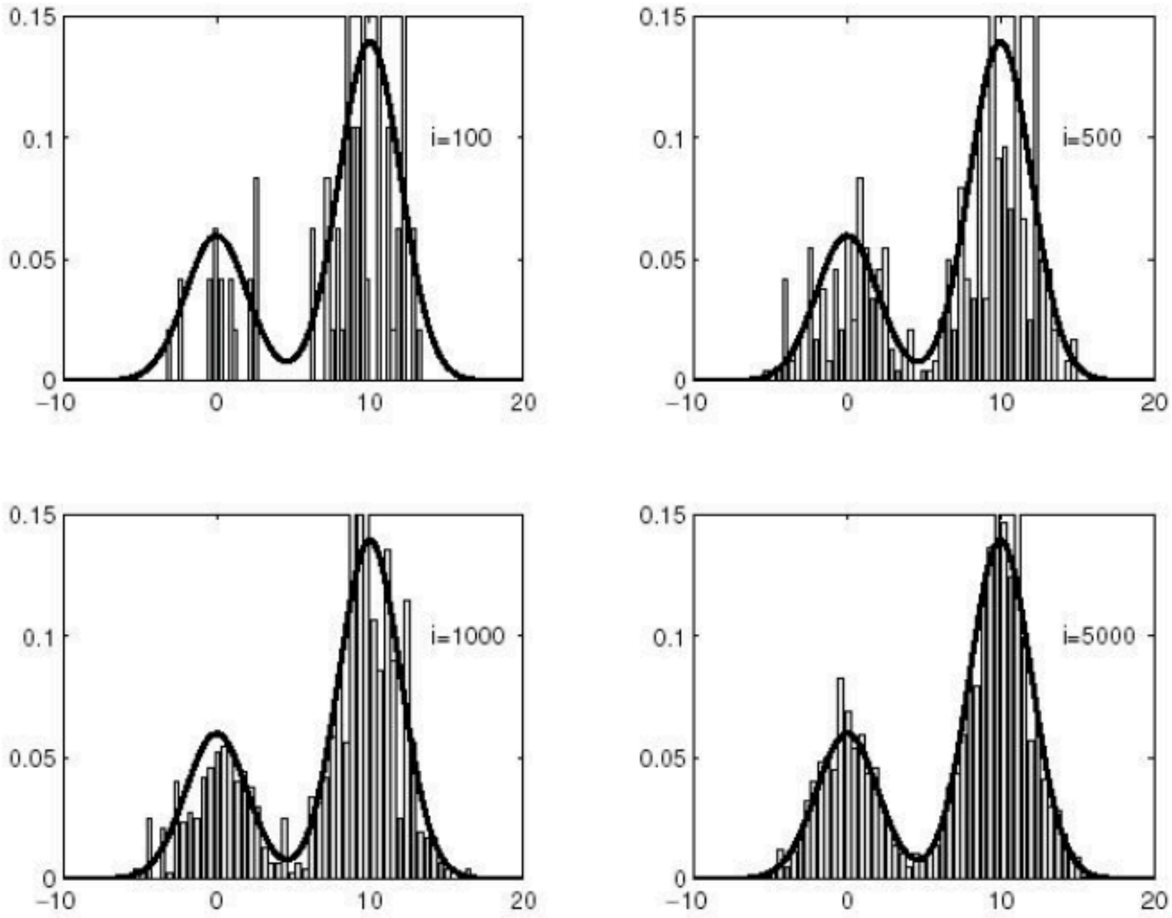
1. Initialize $x^{(0)}$
2. While not mixing // burn-in
   - $x = x^{(t)}$
   - t += 1,
   - sample $u \sim \text{Unif}(0,1)$
   - sample $x^* \sim Q(x^*|x)$

     - if $u < A(x^* | x) = \min\left(1, \frac{\pi(x^*)Q(x | x^*)}{\pi(x)Q(x^* | x)}\right)$
       - $x^{(t)} = x^*$ // transition
     - else
       - $x^{(t)} = x$ // stay in current state

Function
Draw sample (x(t))

- Reset t=0, for $t = 1{:}N$
  - x(t+1)) ← Draw sample (x(t))

MCMC Example



$q(x^*|x) \sim N(x^{(i)},100)$

$p(x) \sim 0.3 \exp(-0.2x^2) + 0.7 \exp(-0.2(x-10)^2)$

Summary of MH

- Random walk through state space
- Can simulate multiple chains in parallel
- Much hinges on proposal distribution Q
  - Want to visit state space where $p(X)$ puts mass
  - Want $A(x^*|x)$ high in modes of $p(X)$
  - Chain mixes well
- Convergence diagnosis
  - How can we tell when burn-in is over?
  - Run multiple chains from different starting conditions, wait until they start "behaving similarly".
  - Various heuristics have been proposed.

Gibbs Sampling is a Special Case of MH

- Gibbs sampling is a special case of MH
- The transition matrix updates each node one at a time using the following proposal:

$$Q((x_i, \mathbf{x}_{-i}) \to (x_i', \mathbf{x}_{-i})) = p(x_i' | \mathbf{x}_{-i})$$

- This is efficient since for two reasons
  - It leads to samples that is always accepted

$$A((x_i, \mathbf{x}_{-i}) \to (x_i', \mathbf{x}_{-i})) = \min\left(1, \frac{p(x'_i, \mathbf{x}_{-i}) Q((x'_i, \mathbf{x}_{-i}) \to (x_i, \mathbf{x}_{-i}))}{p(x_i, \mathbf{x}_{-i}) Q((x_i, \mathbf{x}_{-i}) \to (x'_i, \mathbf{x}_{-i}))}\right)$$

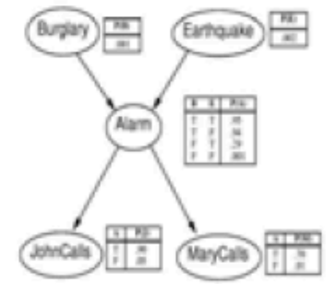
$$= \min\left(1, \frac{p(x'_i | \mathbf{x}_{-i}) p(\mathbf{x}_{-i}) p(x_i | \mathbf{x}_{-i})}{p(x_i | \mathbf{x}_{-i}) p(\mathbf{x}_{-i}) p(x'_i | \mathbf{x}_{-i})}\right) = \min(1,1)$$

  Thus $$T((x_i, \mathbf{x}_{-i}) \to (x_i', \mathbf{x}_{-i})) = p(x_i' | \mathbf{x}_{-i})$$

  - It is efficient since $p(x_i' | \mathbf{x}_{-i})$ only depends on the values in X_i's Markov blanket

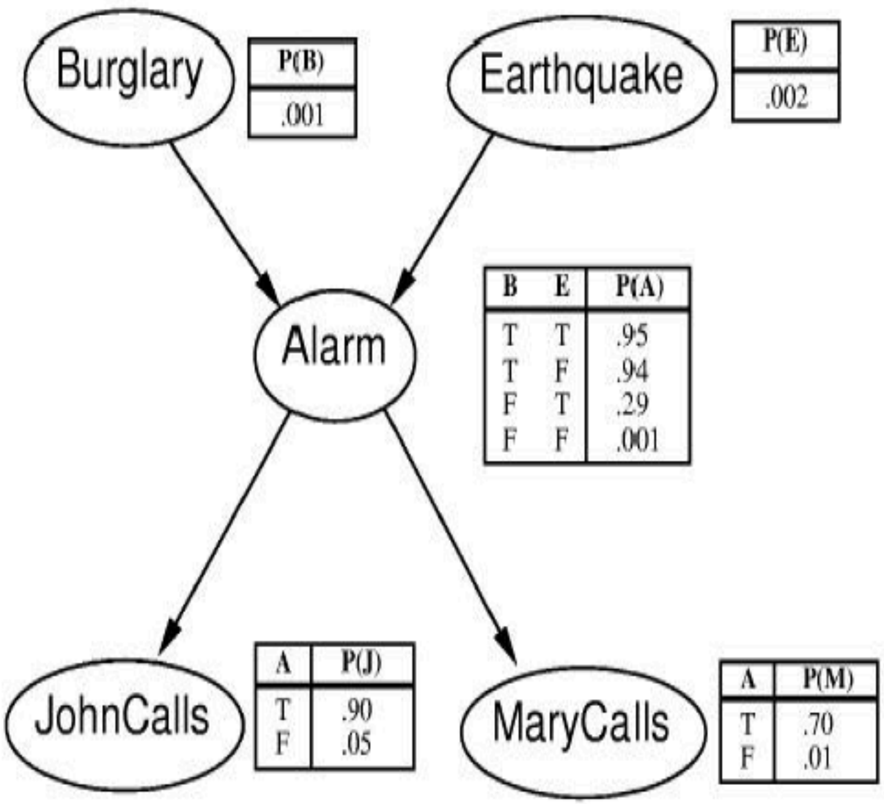
Gibbs Sampling

- Gibbs sampling is an MCMC algorithm that is especially appropriate for inference in graphical models.



- The procedue
  - we have variable set $\boldsymbol{X}=\{x_1, x_2, x_3, \ldots x_N\}$ for a GM
  - at each step one of the variables X_i is selected (at random or according to some fixed sequences), denote the remaining variables as $\boldsymbol{X}_{-i}$, and its current value as $\boldsymbol{x}_{-i}^{(t-1)}$
    - Using the "alarm network" as an example, say at time t we choose X_E, and we denote the current value assignments of the remaining variables, $\boldsymbol{X}_{-E}$, obtained from previous samples, as $x_{-E}^{(t-1)} = \left\{x_B^{(t-1)}, x_A^{(t-1)}, x_J^{(t-1)}, x_M^{(t-1)}\right\}$
  - the conditonal distribution $p(X_i | \boldsymbol{x}_{-i}^{(t-1)})$ is computed
  - a value $\boldsymbol{x}_i^{(t)}$ is sampled from this distribution
  - the sample $\boldsymbol{x}_i^{(t)}$ replaces the previous sampled value of X_i in $\boldsymbol{X}$.
    - i.e., $x^{(t)} = x_{-E}^{(t-1)} \cup x_E^{(t)}$

Gibbs Sampling of an Alarm Network



MB(A)={B, E, J, M}

MB(E)={A, B}

- To calculate P(J|B1,M1)
- Choose (B1,E0,A1,M1,J1) as a start
- **Evidences** are B1, M1, **variables** are A, E, J.
- Choose next variable as A
- Sample A by P(A|MB(A))=P(A|B1, E0, M1, J1) suppose to be false.
- (B1, E0, A0, M1, J1)
- Choose next random variable as E, sample E~P(E|B1,A0)
- ...

A General Gibbs Sampling Algorithm

- Given a target distribution p(X), where X = (x_1, x_2, …, x_D).
- Criterion: (1) have an analytic (mathematical) expression for the conditional distribution of each variable given all other variables. P(x_i | x_1, x_2, …, x_{i-1}, x_{i+1}, …, x_D).
- (2) Be able to sample a variable from each conditional distribution

Algorithm

- Set t = 0
- Generate an initial state $X^{(0)}$
- Repeat until t = M

  set t = t + 1

  for each dimension i = 1 .. D

  draw x_i from $P(x_i \mid x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_D)$.

Gibbs Sampling for Gaussian Distribution

$$f_{\mathbf{x}}(x_1, \ldots, x_k) = \frac{1}{\sqrt{(2\pi)^k |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right),$$

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(x-\mu_x)^2}{\sigma_x^2} + \frac{(y-\mu_y)^2}{\sigma_y^2} - \frac{2\rho(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y}\right]\right)$$

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix}.$$

$$p(\mathbf{x}) = \mathcal{N}(\mu, \Sigma)$$

with mean

$$\mu = [\mu_1, \mu_2] = [0, 0]$$

and covariance

$$\Sigma = \begin{bmatrix} 1 & \rho_{12} \\ \rho_{21} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

Conditional Sampling

$$p(x_1 | x_2^{(t-1)}) = \mathcal{N}(\mu_1 + \rho_{21}(x_2^{(t-1)} - \mu_2), \sqrt{1 - \rho_{21}^2})$$

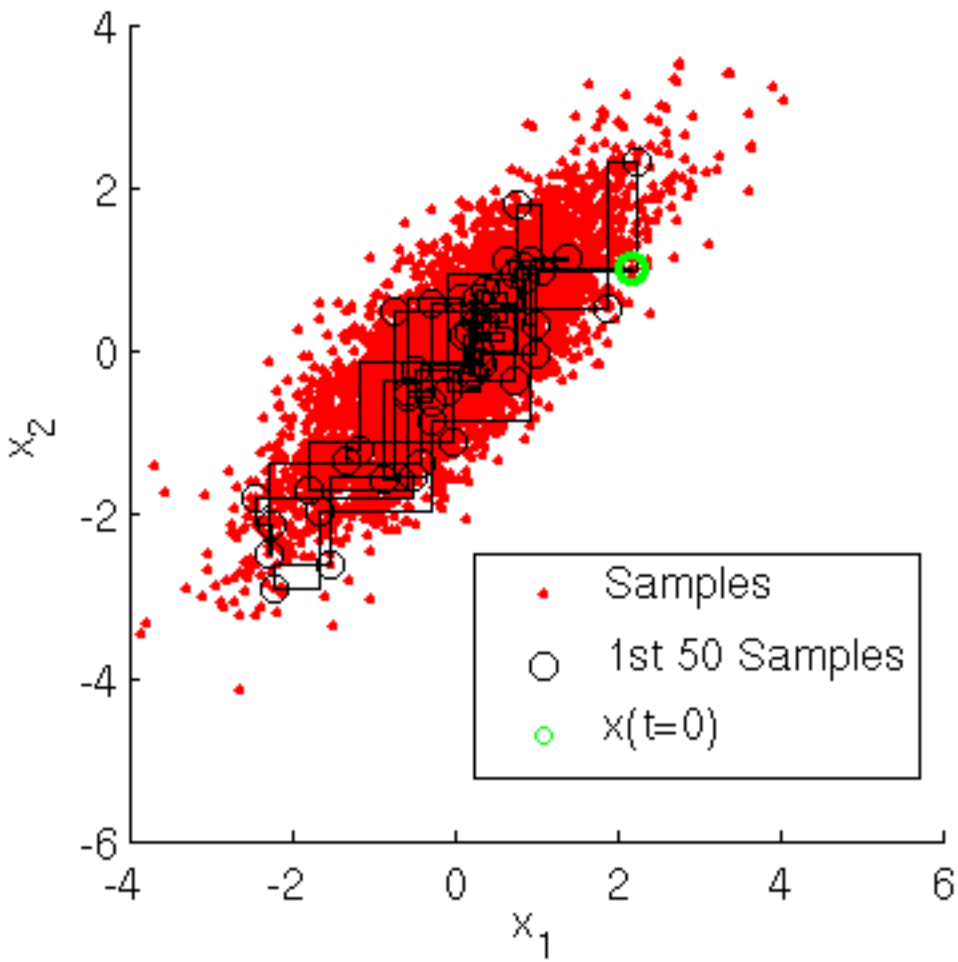
and

$$p(x_2 | x_1^{(t)}) = \mathcal{N}(\mu_2 + \rho_{12}(x_1^{(t)} - \mu_1), \sqrt{1 - \rho_{12}^2}),$$

Matlab Implementation

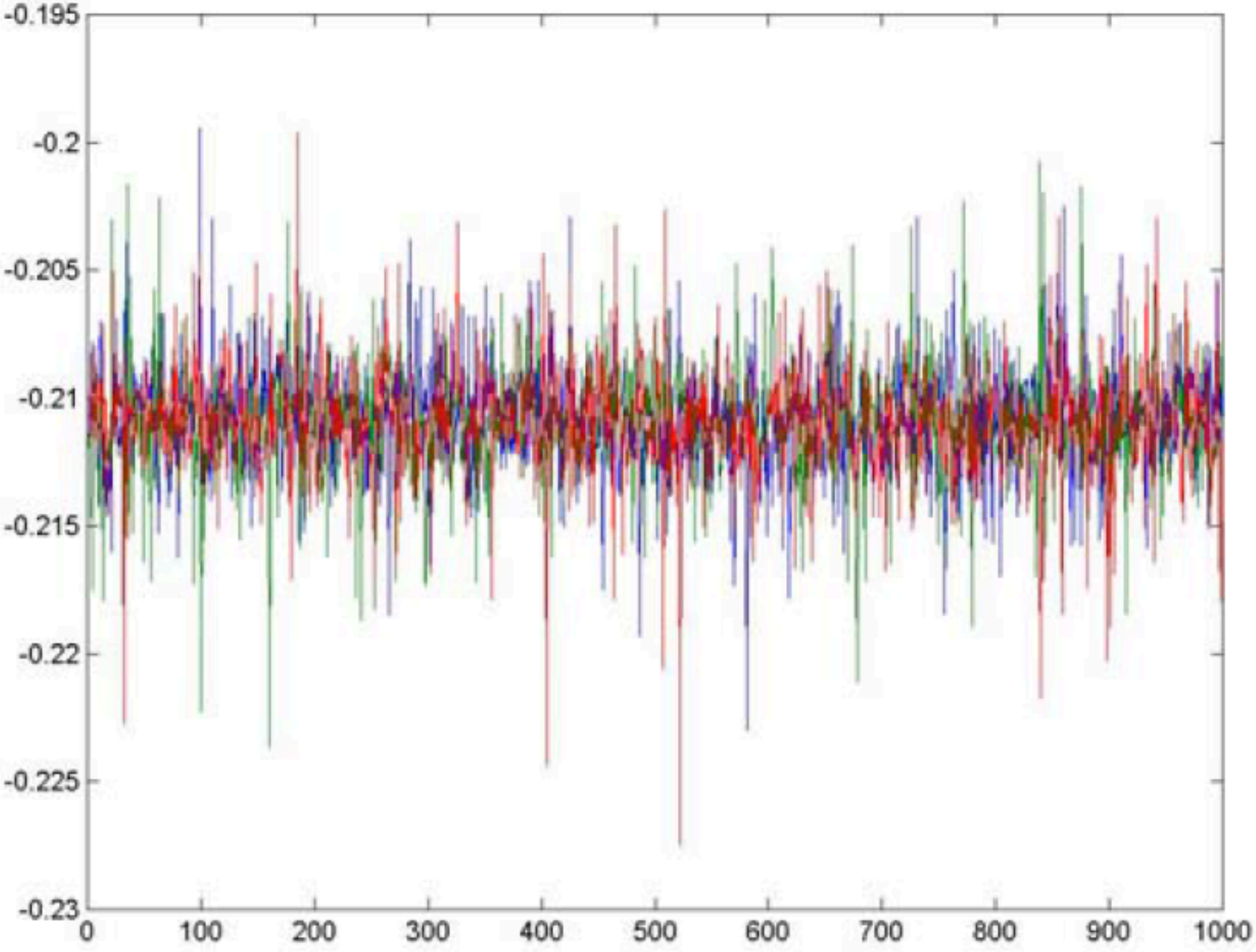
```matlab
% EXAMPLE: GIBBS SAMPLER FOR BIVARIATE NORMAL
rand('seed' ,12345);
nSamples = 5000;

mu = [0 0]; % TARGET MEAN
rho(1) = 0.8; % rho_21
rho(2) = 0.8; % rho_12

% INITIALIZE THE GIBBS SAMPLER
propSigma = 1; % PROPOSAL VARIANCE
minn = [-3 -3];
maxx = [3 3];

% INITIALIZE SAMPLES
x = zeros(nSamples,2);
x(1,1) = unifrnd(minn(1), maxx(1));
x(1,2) = unifrnd(minn(2), maxx(2));

dims = 1:2; % INDEX INTO EACH DIMENSION

% RUN GIBBS SAMPLER
t = 1;
while t < nSamples
    t = t + 1;
    T = [t-1,t];
    for iD = 1:2 % LOOP OVER DIMENSIONS
        % UPDATE SAMPLES
        nIx = dims~=iD; % *NOT* THE CURRENT DIMENSION
        % CONDITIONAL MEAN
        muCond = mu(iD) + rho(iD)*(x(T(iD),nIx)-mu(nIx));
        % CONDITIONAL VARIANCE
        varCond = sqrt(1-rho(iD)^2);
        % DRAW FROM CONDITIONAL
        x(t,iD) = normrnd(muCond,varCond);
    end
end

% DISPLAY SAMPLING DYNAMICS
figure;
h1 = scatter(x(:,1),x(:,2),'r.');

% CONDITIONAL STEPS/SAMPLES
hold on;
for t = 1:50
    plot([x(t,1),x(t+1,1)],[x(t,2),x(t,2)],'k-');
    plot([x(t+1,1),x(t+1,1)],[x(t,2),x(t+1,2)],'k-');
    h2 = plot(x(t+1,1),x(t+1,2),'ko');
end

h3 = scatter(x(1,1),x(1,2),'go','Linewidth',3);
legend([h1,h2,h3],{'Samples','1st 50 Samples','x(t=0)'},'Location','Northwest')
hold off;
xlabel('x_1');
ylabel('x_2');
axis square
```

Gibbs Sampling Example



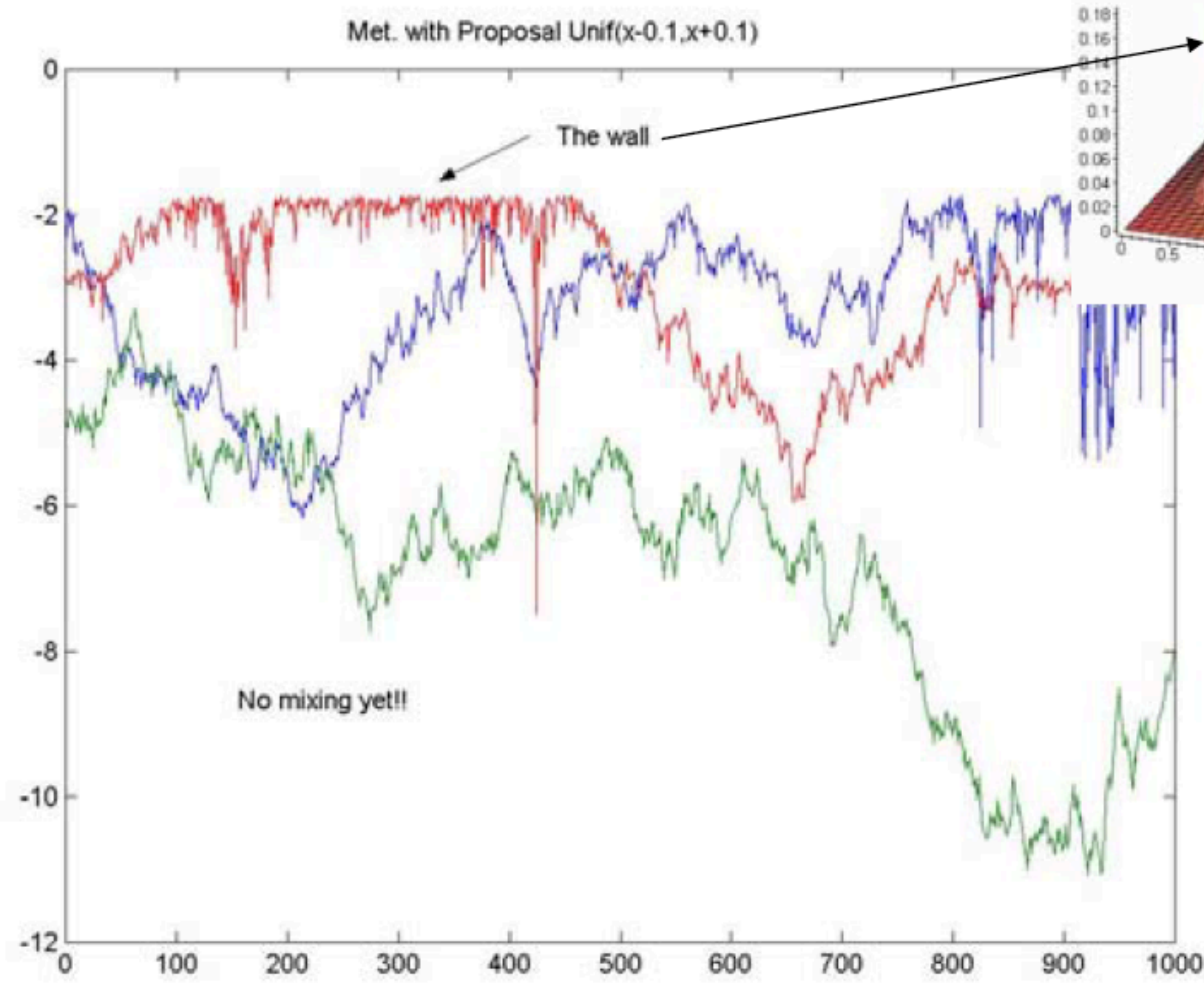
http://theclevermachine.wordpress.com/2012/11/05/mcmc-the-gibbs-sampler/

Good Chains

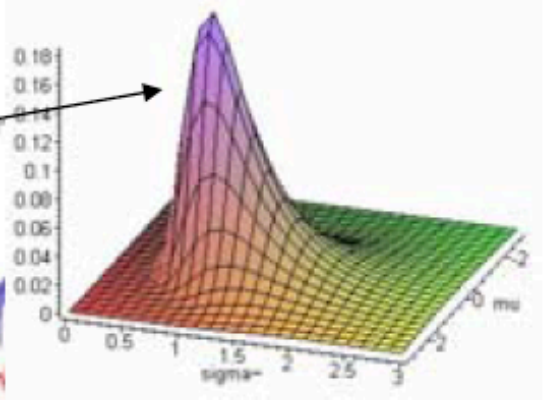


-0.195
-0.2
-0.205
-0.21
-0.215
-0.22
-0.225
-0.23
0
100
200
300
400
500
600
700
800
900
1000

Bad Chains



Met. with Proposal Unif(x-0.1,x+0.1)
The wall
No mixing yet!!
0
-2
-4
-6
-8
-10
-12
0
100
200
300
400
500
600
700
800
900
1000



0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
0
0.5
1
1.5
2
2.5
3
sigma
mu

Reading Assignment

- C. Andrieu et al. An Introduction of MCMC for machine learning.
- http://www.cs.princeton.edu/courses/archive/spr06/cos598C/papers/AndrieuFreitasDoucetJordan2003.pdf
- Write a half-page summary
- Due August 28 (Wednesday)

A Real-World Optimization Problem

- Find the common substring in multiple DNA sequences
- Gibbs sampling approach
- Your group info (5 – 6 students) to me by August 30 (Friday).