11. Sampling Methods: Markov Chain Monte Carlo

Markov Chain Monte Carlo

- In high-dimensional spaces, rejection sampling and importance sampling are very inefficient
- An alternative is Markov Chain Monte Carlo (MCMC)
- It keeps a record of the current state and the proposal depends on that state
- Most common algorithms are the Metropolis-Hastings algorithm and Gibbs Sampling

Markov Chains Revisited

A Markov Chain is a distribution over discrete-state random variables $\mathbf{x}_1, \dots, \mathbf{x}_M$ so that

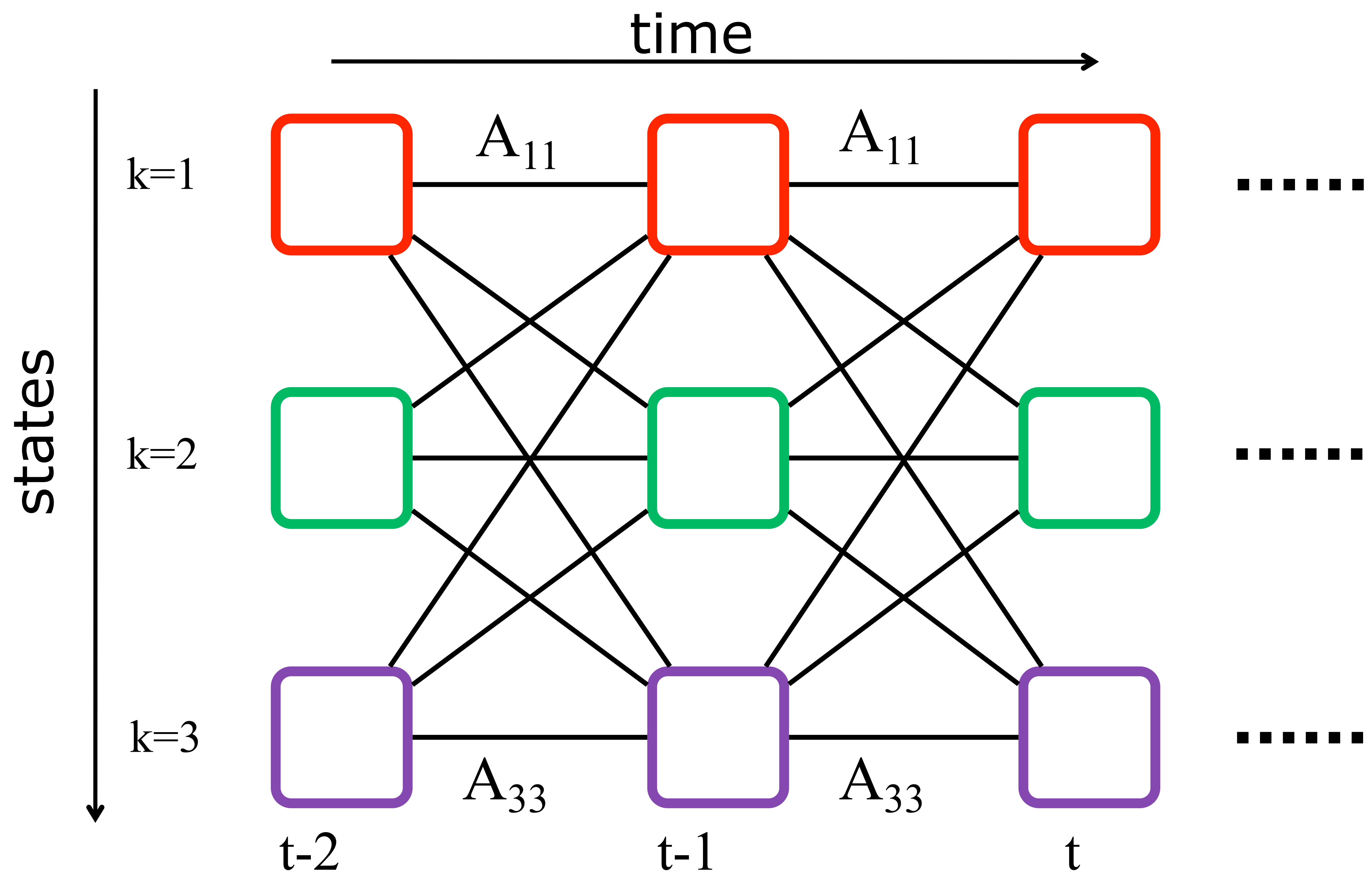
$$p(\mathbf{x}_1, \dots, \mathbf{x}_T) = p(\mathbf{x}_1)p(\mathbf{x}_2 \mid \mathbf{x}_1) \cdots = p(\mathbf{x}_1) \prod_{t=2}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$$

The graphical model of a Markov chain is this:

We will denote $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ as a row vector $\boldsymbol{\pi}_t$

A Markov chain can also be visualized as a **state transition diagram.**

The State Transition Diagram



Some Notions

- The Markov chain is said to be **homogeneous** if the transitions probabilities are all the same at every time step t (here we only consider homogeneous Markov chains)
- The transition matrix is **row-stochastic**, i.e. all entries are between 0 and 1 and all rows sum up to 1
- Observation: the probabilities of reaching the states can be computed using a vector-matrix multiplication

The Stationary Distribution

The probability to reach state k is $\pi_{k,t} = \sum_{i=1}^{K} \pi_{i,t-1} A_{ik}$

Or, in matrix notation: $\boldsymbol{\pi}_t = \boldsymbol{\pi}_{t-1} A$

We say that $\boldsymbol{\pi}_t$ is **stationary** if $\boldsymbol{\pi}_t = \boldsymbol{\pi}_{t-1}$

Questions:

- How can we know that a stationary distributions exists?
- And if it exists, how do we know that it is unique?

The Stationary Distribution (Existence)

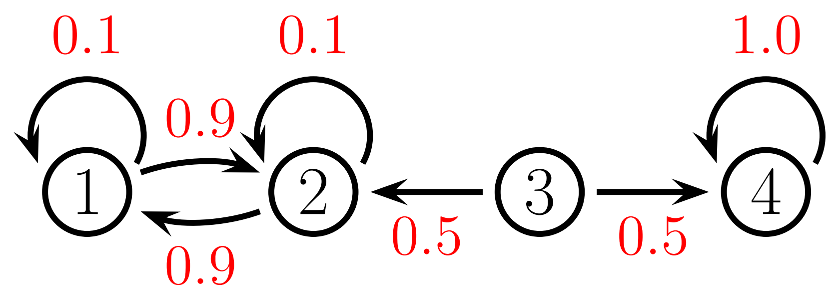
To find a stationary distribution we need to solve the eigenvector problem $A^T \mathbf{v} = \mathbf{v}$

The stationary distribution is then $\boldsymbol{\pi} = \mathbf{v}^T$ where $\mathbf{v}$ is the eigenvector for which the eigenvalue is 1.

This eigenvector needs to be normalized so that it is a valid distribution.

Theorem (Perron-Frobenius): Every row-stochastic matrix has such an eigen vector, but this vector may not be unique.

Stationary Distribution (Uniqueness)



- A Markov chain can have many stationary distributions
- Sufficient for a unique stationary distribution: we can reach every state from any other state in finite steps at non-zero probability (i.e. the chain is **ergodic**)
- This is equivalent to the property that the transition matrix is **irreducible:**

$$\forall i, j \; \exists m \quad (A^m)_{ij} > 0$$

Main Idea of MCMC

- So far, we specified the transition probabilities and analysed the resulting distribution
- This was used, e.g. in HMMs

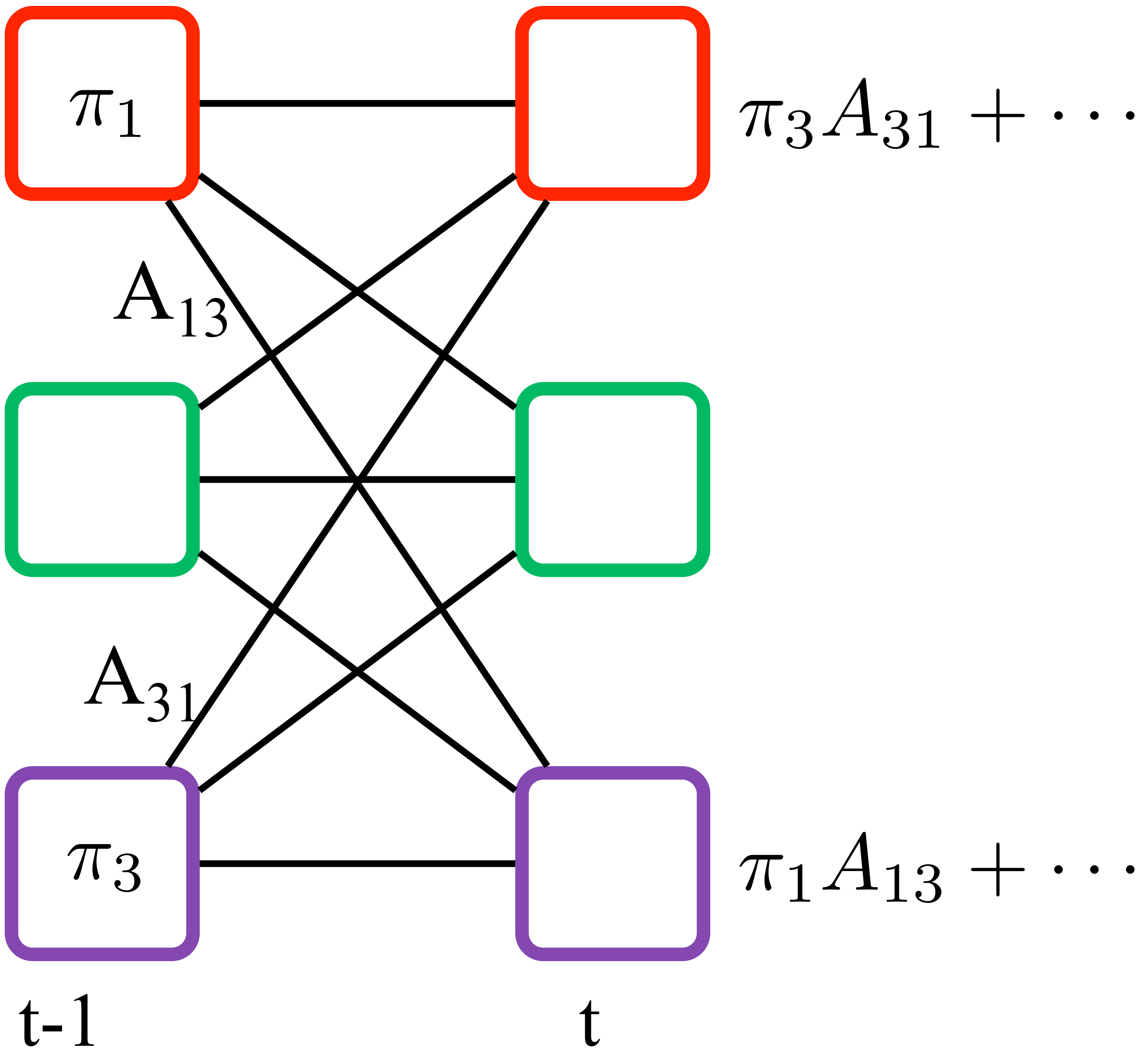
Now:

- We want to sample from an arbitrary distribution
- To do that, we design the transition probabilities so that the resulting stationary distribution is our desired (target) distribution!

Detailed Balance

Definition: A transition distribution π_t satisfies the property of **detailed balance** if $\pi_i A_{ij} = \pi_j A_{ji}$

The chain is then said to be **reversible.**



Making a Distribution Stationary

Theorem: If a Markov chain with transition matrix A is irreducible and satisfies detailed balance wrt. the distribution π, then π is a stationary distribution of the chain.

Proof:

$$\sum_{i=1}^{K} \pi_i A_{ij} = \sum_{i=1}^{K} \pi_j A_{ji} = \pi_j \sum_{i=1}^{K} A_{ji} = \pi_j \qquad \forall j$$

it follows $\boldsymbol{\pi} = \boldsymbol{\pi} A$.

This is a sufficient, but not necessary condition.

Sampling with a Markov Chain

The idea of MCMC is to sample state transitions based on a **proposal distribution** q.

The most widely used algorithm is the Metropolis-Hastings (MH) algorithm.

In MH, the decision whether to stay in a given state is based on a given probability.

If the proposal distribution is $q(\mathbf{x}' \mid \mathbf{x})$, then we move to state $\mathbf{x}'$ with probability

$$\min\left(1, \frac{\tilde{p}(x')q(x \mid x')}{\tilde{p}(x)q(x' \mid x)}\right)$$

Unnormalized target distribution

The Metropolis-Hastings Algorithm

- Initialize x^0
- for $s = 0, 1, 2, \ldots$
  - define $x = x^s$
  - sample $x' \sim q(x' \mid x)$
    - compute acceptance probability
$$\alpha = \frac{\tilde{p}(x')q(x \mid x')}{\tilde{p}(x)q(x' \mid x)}$$
    - compute $r = \min(1, \alpha)$
    - sample $u \sim U(0, 1)$
    - set new sample to
$$x^{s+1} = \begin{cases} x' & \text{if} \quad u < r \\ x^s & \text{if} \quad u \geq r \end{cases}$$

Why Does This Work?

We have to prove that the transition probability of the MH algorithm satisfies detailed balance wrt the target distribution.

Theorem: If $p_{MH}(\mathbf{x}' \mid \mathbf{x})$ is the transition probability of the MH algorithm, then

$$p(\mathbf{x})p_{MH}(\mathbf{x}' \mid \mathbf{x}) = p(\mathbf{x}')p_{MH}(\mathbf{x} \mid \mathbf{x}')$$

Proof:

Why Does This Work?

We have to prove that the transition probability of the MH algorithm satisfies detailed balance wrt the target distribution.

Theorem: If $p_{MH}(\mathbf{x}' \mid \mathbf{x})$ is the transition probability of the MH algorithm, then

$$p(\mathbf{x})p_{MH}(\mathbf{x}' \mid \mathbf{x}) = p(\mathbf{x}')p_{MH}(\mathbf{x} \mid \mathbf{x}')$$

Note: All formulations are valid for discrete and for continuous variables!

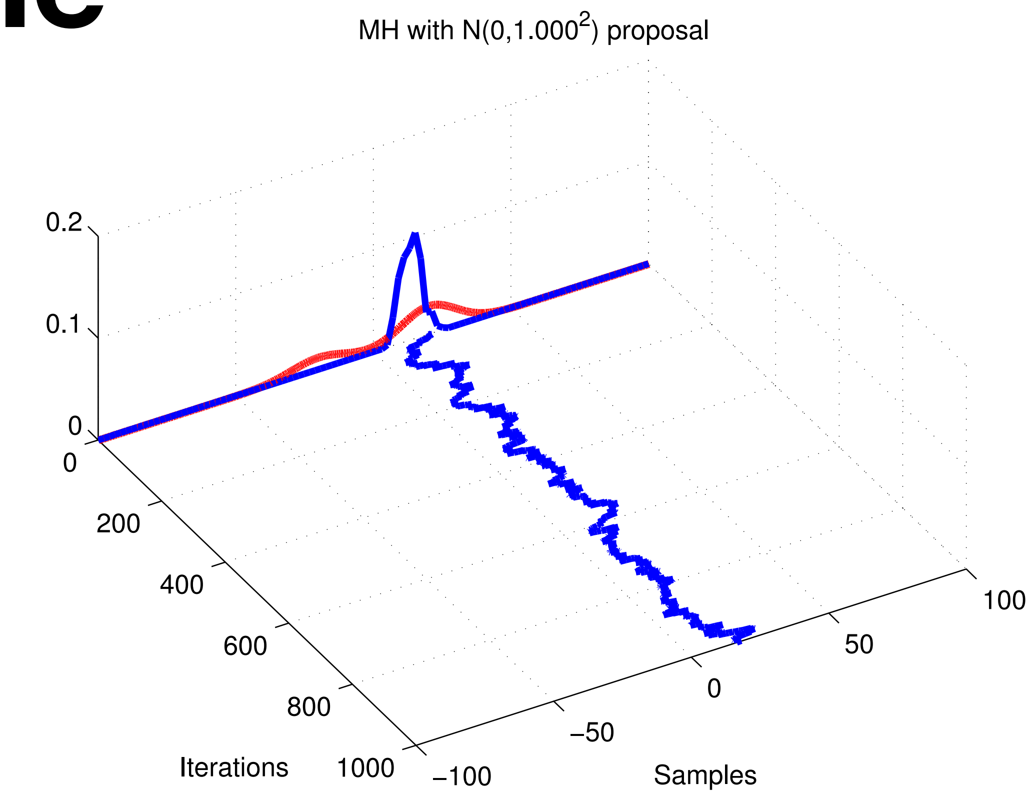
Choosing the Proposal

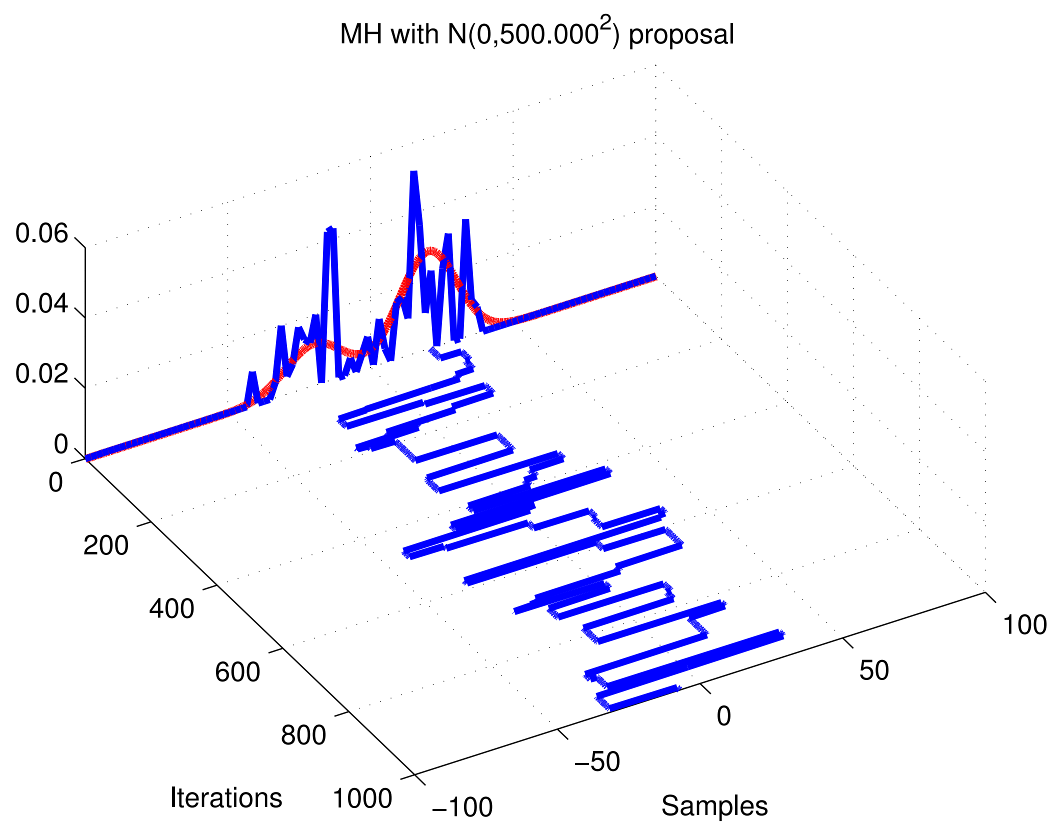
- A proposal distribution is valid if it gives a non-zero probability of moving to the states that have a non-zero probability in the target.
- A good proposal is the Gaussian, because it has a non-zero probability for all states.
- **However:** the variance of the Gaussian is important!
  - with low variance, the sampler does not explore sufficiently, e.g. it is fixed to a particular mode
  - with too high variance, the proposal is rejected too often, the samples are a bad approximation

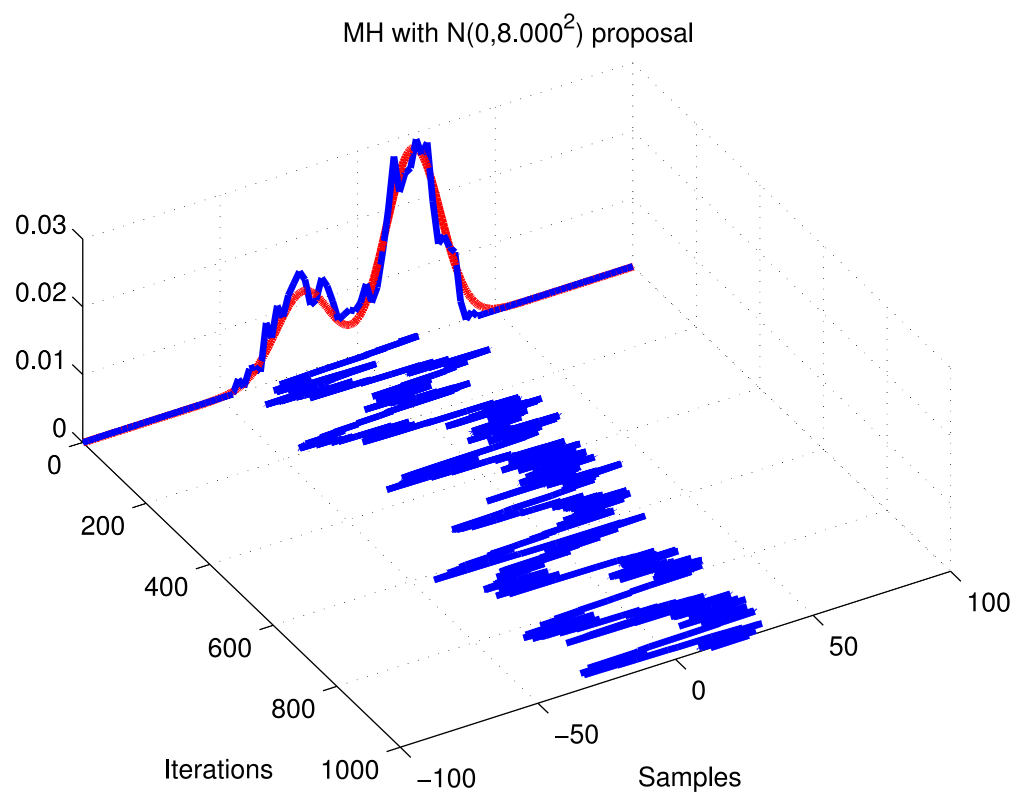
Example

Target is a mixture of 2 1D Gaussians.

Proposal is a Gaussian with different variances.







Gibbs Sampling

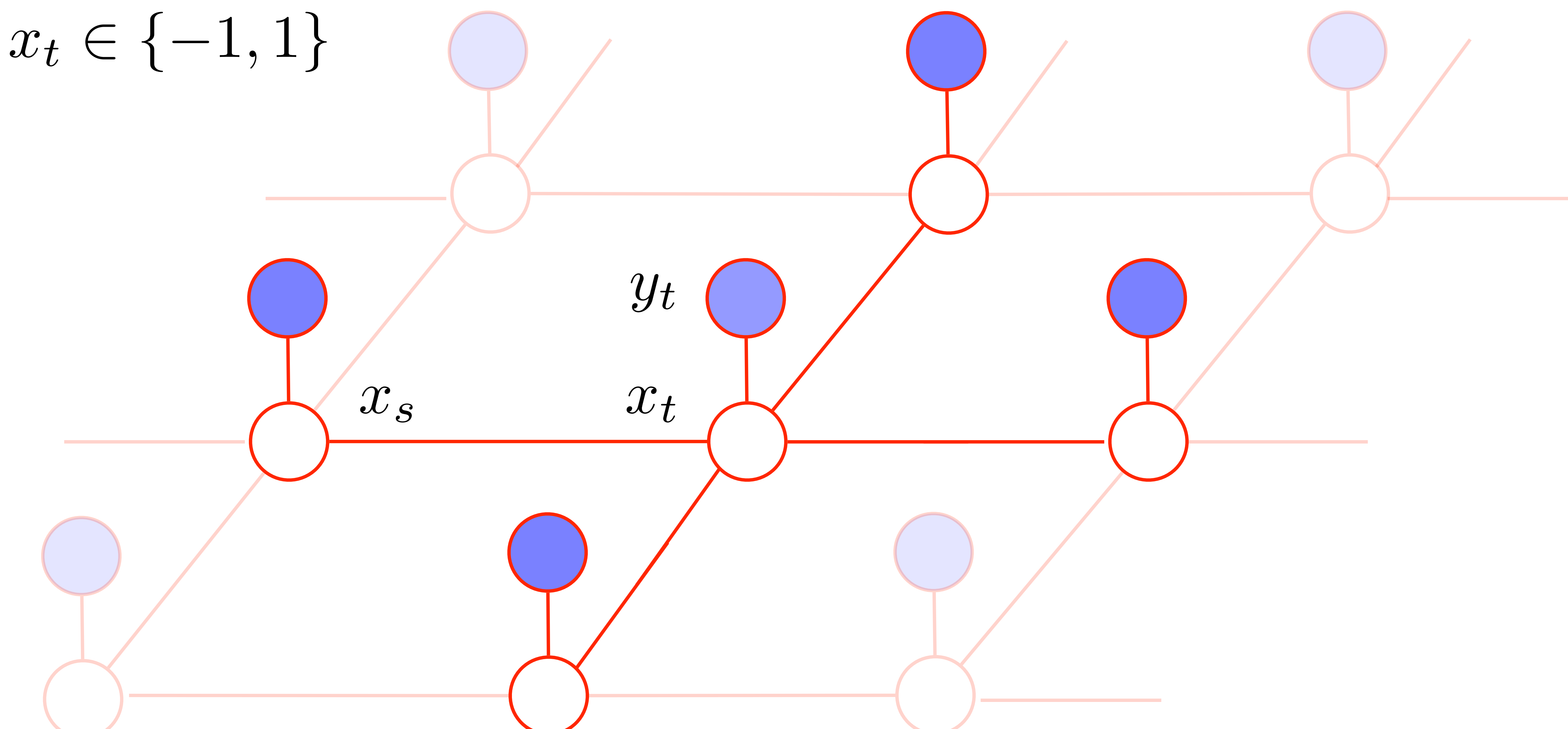
- Initialize $\{z_i : i = 1, \ldots, M\}$
- For $\tau = 1, \ldots, T$
  - Sample $z_1^{(\tau+1)} \sim p(z_1 \mid z_2^{(\tau)}, \ldots, z_M^{(\tau)})$
  - Sample $z_2^{(\tau+1)} \sim p(z_2 \mid z_1^{(\tau+1)}, \ldots, z_M^{(\tau)})$
  - ...
  - Sample $z_M^{(\tau+1)} \sim p(z_M \mid z_1^{(\tau+1)}, \ldots, z_{M-1}^{(\tau+1)})$

Idea: sample from the full conditional

This can be obtained, e.g. from the Markov blanket in graphical models.

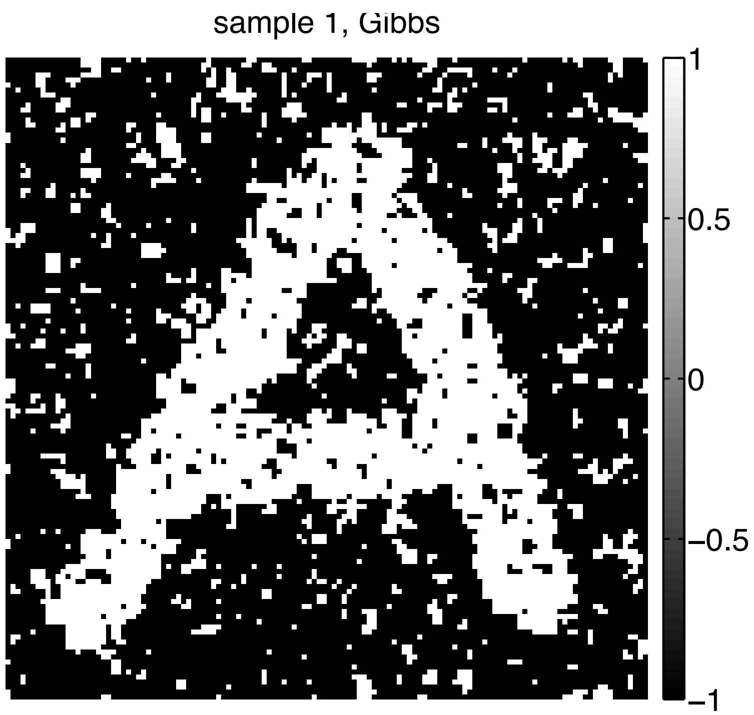
Gibbs Sampling: Example

- Use an MRF on a binary image with edge potentials $\psi(x_s, x_t) = \exp(J x_s x_t)$ ("Ising model") and node potentials $\psi(x_t) = \mathcal{N}(y_t \mid x_t, \sigma^2)$

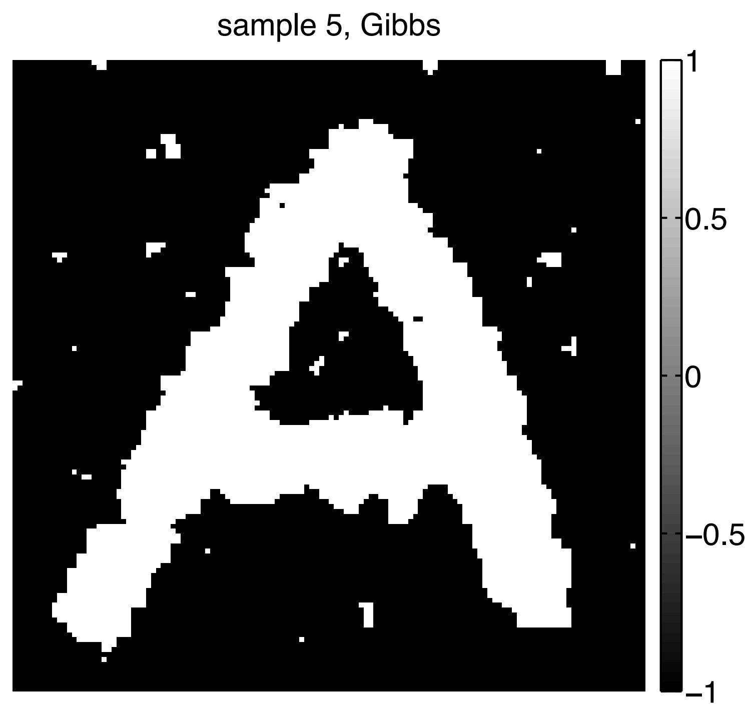


Gibbs Sampling: Example

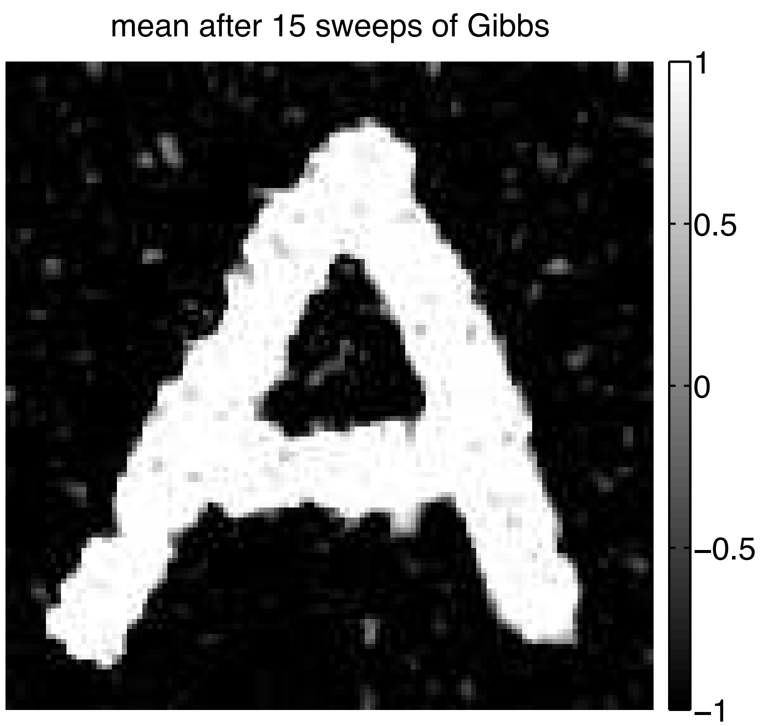
- Use an MRF on a binary image with edge potentials $\psi(x_s, x_t) = \exp(J x_s x_t)$ ("Ising model") and node potentials $\psi(x_t) = \mathcal{N}(y_t \mid x_t, \sigma^2)$
- Sample each pixel in turn



After 1 sample



After 5 samples



Average after 15 samples

Gibbs Sampling for GMMs

- Again, we start with the full joint distribution:

$$p(X, Z, \boldsymbol{\mu}, \Sigma, \boldsymbol{\pi}) = p(X \mid Z, \boldsymbol{\mu}, \Sigma)p(Z \mid \boldsymbol{\pi})p(\boldsymbol{\pi}) \prod_{k=1}^{K} p(\boldsymbol{\mu}_k)p(\Sigma_k)$$

- It can be shown that the full conditionals are:

$$p(z_i = k \mid \mathbf{x}_i, \boldsymbol{\mu}, \Sigma, \boldsymbol{\pi}) \propto \pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \Sigma_k)$$

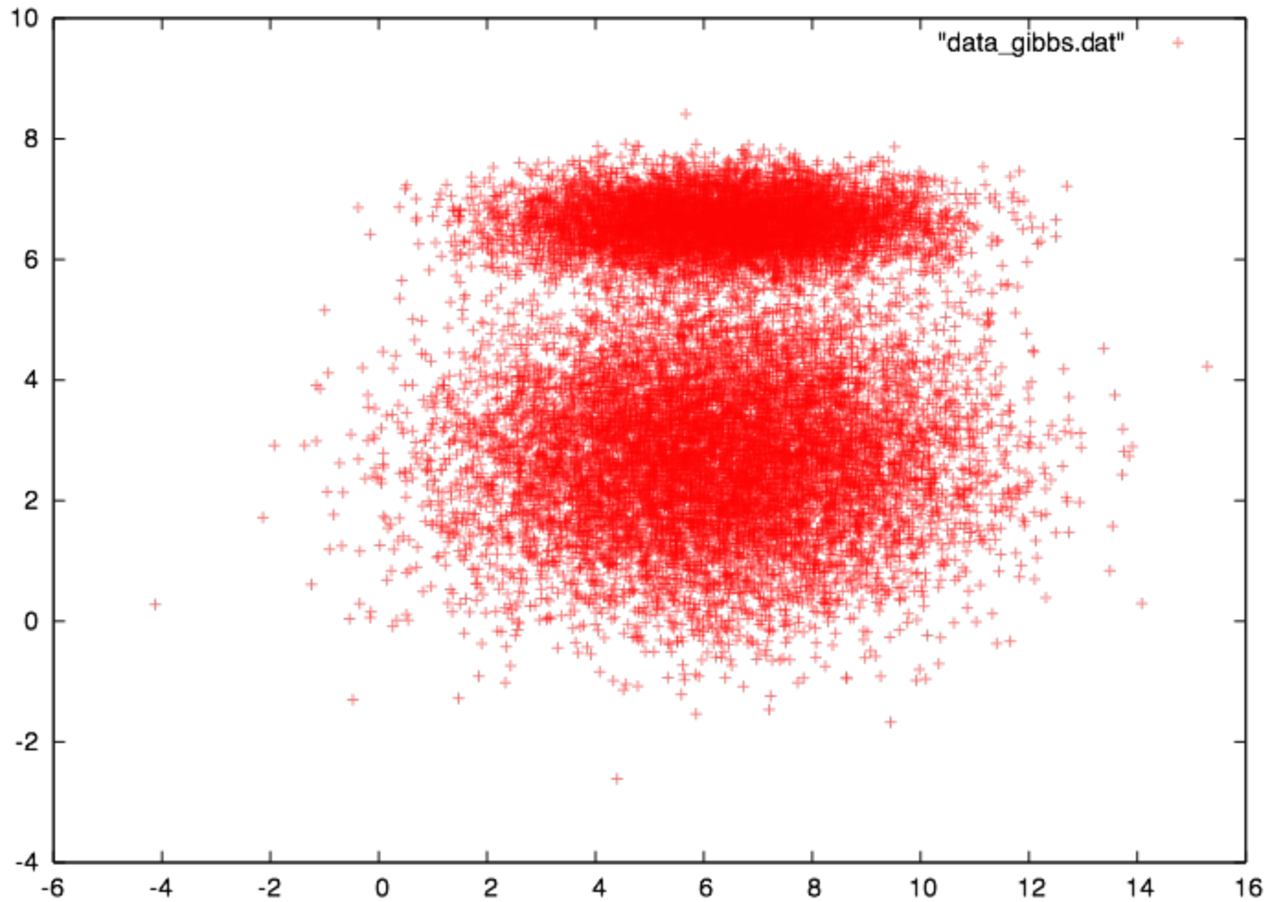
$$p(\boldsymbol{\pi} \mid \mathbf{z}) = \mathrm{Dir}(\{\alpha_k + \sum_{i=1}^{N} z_{ik}\}_{k=1}^{K})$$

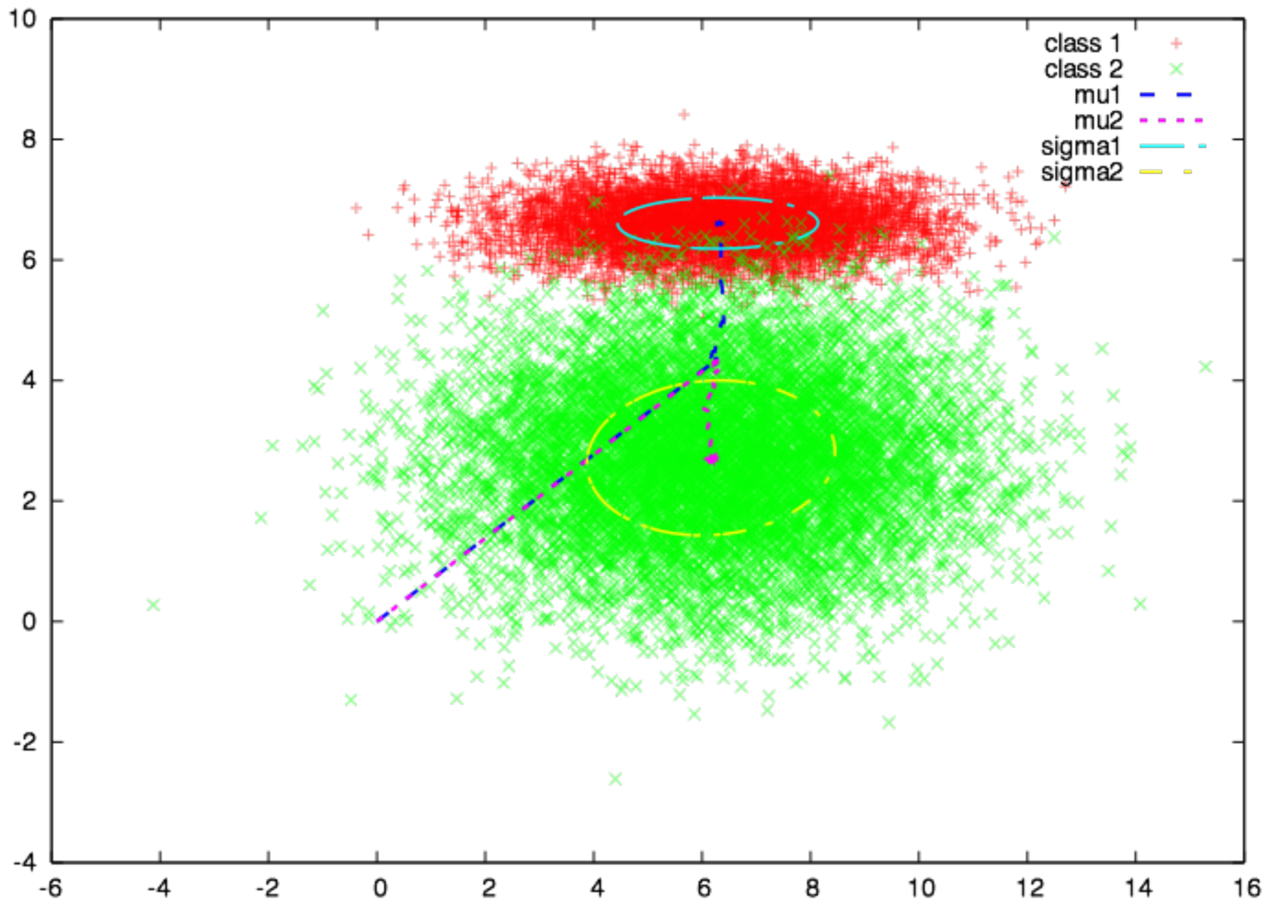
$$p(\boldsymbol{\mu}_k \mid \Sigma_k, Z, X) = \mathcal{N}(\boldsymbol{\mu}_k \mid \mathbf{m}_k, V_k)$$ (linear-Gaussian)

$$p(\Sigma_k \mid \boldsymbol{\mu}_k, Z, X) = \mathcal{IW}(\Sigma_k \mid S_k, \nu_k)$$

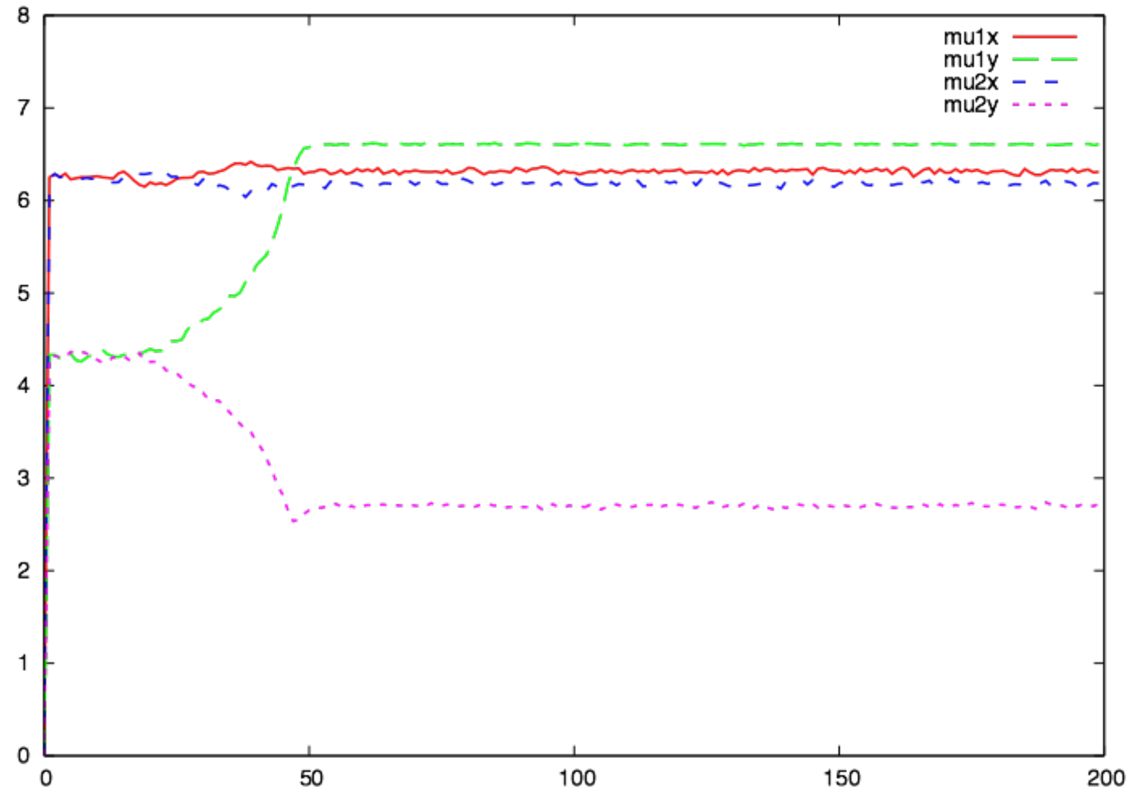
Gibbs Sampling for GMMs

- First, we initialize all variables
- Then we iterate over sampling from each conditional in turn
- In the end, we look at $\boldsymbol{\mu}_k$ and Σ_k





How Often Do We Have To Sample?



- Here: after 50 sample rounds the values don't change any more
- In general, the **mixing time** τ_ϵ is related to the **eigen gap** $\gamma = \lambda_1 - \lambda_2$ of the transition matrix:

$$\tau_\epsilon \leq O(\frac{1}{\gamma} \log \frac{n}{\epsilon})$$

Gibbs Sampling is a Special Case of MH

- The proposal distribution in Gibbs sampling is

$$q(\mathbf{x}' \mid \mathbf{x}) = p(x_i' \mid \mathbf{x}_{-i})\mathbb{I}(\mathbf{x}'_{-i} = \mathbf{x}_{-i})$$

- This leads to an acceptance rate of:

$$\alpha = \frac{p(\mathbf{x}')q(\mathbf{x} \mid \mathbf{x}')}{p(\mathbf{x})q(\mathbf{x}' \mid \mathbf{x})} = \frac{p(x_i' \mid \mathbf{x}'_{-i})p(\mathbf{x}'_{-i})p(x_i \mid \mathbf{x}'_{-i})}{p(x_i \mid \mathbf{x}_{-i})p(\mathbf{x}_{-i})p(x_i' \mid \mathbf{x}_{-i})} = 1$$

- Although the acceptance is 100%, Gibbs sampling does not converge faster, as it only updates one variable at a time.

Summary

- Markov Chain Monte Carlo is a family of sampling algorithms that can sample from arbitrary distributions by moving in state space
- Most used methods are the Metropolis-Hastings (MH) and the Gibbs sampling method
- MH uses a proposal distribution and accepts a proposed state randomly
- Gibbs sampling does not use a proposal distribution, but samples from the full conditionals