Gibbs Sampling

Johan Alenlöv, Linköping University

2023-10-30

Outline

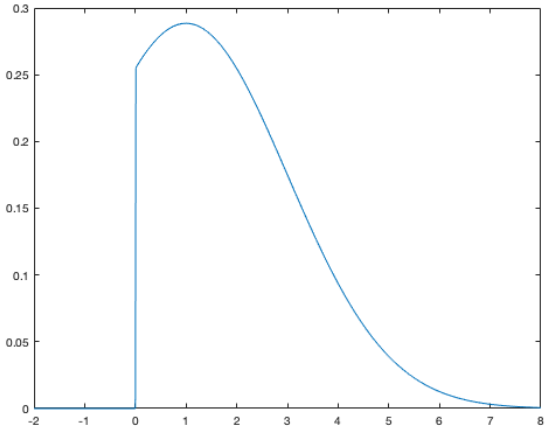
Aim: Introduce Markov chain Monte Carlo methods, in particular Gibbs sampling, for sampling from the posterior distribution.

Outline:

1. Summary of Gibbs and MCMC
2. Preparatory exercise
3. Gibbs sampling in graphical models

Motivation

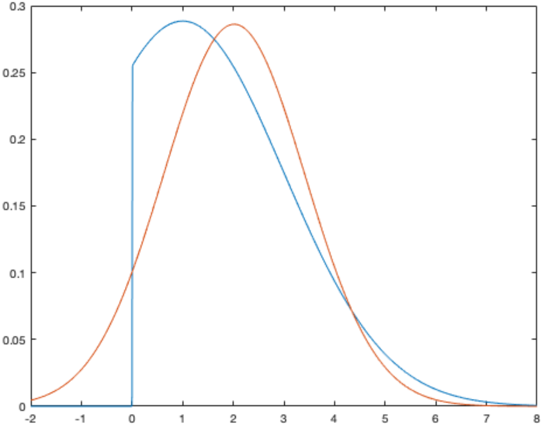
We are often tasked with sampling from some complicated distribution.



Assume now that the distribution is a truncated Gaussian distribution.

Motivation

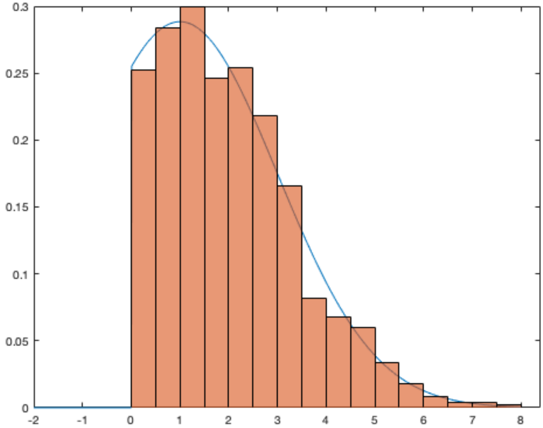
We are often tasked with sampling from some complicated distribution.



Lets approximate it using a Gaussian distribution.

Motivation

Imagine that we instead have access to some function producing random variables from the truncated Gaussian distribution and approximate the distribution using these samples.



Markov chain Monte Carlo

An MCMC sampler generates a **Markov chain** $\{x[m]\}_{m=0}^{M}$ in the following way:

- **Initialize** set $x[0]$ arbitrarily.
- **For** $m = 1, \ldots, M$: sample $x[m] \sim \kappa(x[m-1], \cdot)$

Here $\kappa(x, x^*)$ is a **Markov kernel**, i.e. a conditional distribution for the next state x^* given the current state x.

Markov chain Monte Carlo

An MCMC sampler generates a **Markov chain** $\{x[m]\}_{m=0}^{M}$ in the following way:

- **Initialize** set $x[0]$ arbitrarily.
- **For** $m = 1, \ldots, M$: sample $x[m] \sim \kappa(x[m-1], \cdot)$

Here $\kappa(x, x^*)$ is a **Markov kernel**, i.e. a conditional distribution for the next state x^* given the current state x.

Basic requirement 1: The kernel κ should admit π as a **stationary** distribution,

$$\int \pi(x)\kappa(x, x^*)\mathrm{d}x = \pi(x^*)$$

Markov chain Monte Carlo

An MCMC sampler generates a **Markov chain** $\{x[m]\}_{m=0}^{M}$ in the following way:

- **Initialize** set $x[0]$ arbitrarily.
- **For** $m = 1, \ldots, M$: sample $x[m] \sim \kappa(x[m-1], \cdot)$

Here $\kappa(x, x^*)$ is a **Markov kernel**, i.e. a conditional distribution for the next state x^* given the current state x.

Basic requirement 1: The kernel κ should admit π as a **stationary** distribution,

$$\int \pi(x)\kappa(x, x^*)\mathrm{d}x = \pi(x^*)$$

Basic requirement 2: The kernel should be **ergodic** — the chain should reach the stationary distribution no matter the initial distribution.

Convergence of MCMC

Assume that the resulting **Markov chain** $\{x[m]\}_{m=1}^{M}$ is geometrically ergodic, then

$$\sqrt{M}\left(\frac{1}{M}\sum_{m=1}^{M} h(x[m]) - \mathbb{E}_\pi(h(X))\right) \xrightarrow{d} N(0, \sigma_\infty^2(h)), \quad \text{as } M \to \infty,$$

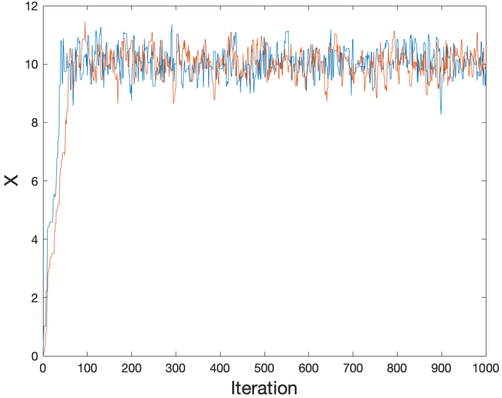
where

$$\begin{aligned}\sigma_\infty^2(h) =& r_0(h) + 2\sum_{\ell=1}^{\infty} r_\ell(h),\\ r_\ell(h) =& \mathrm{Cov}(h(X_\ell), h(X_0)), \quad \ell \geq 0.\end{aligned}$$

Where the covariance is at **stationarity**, i.e. when initialized according to π.

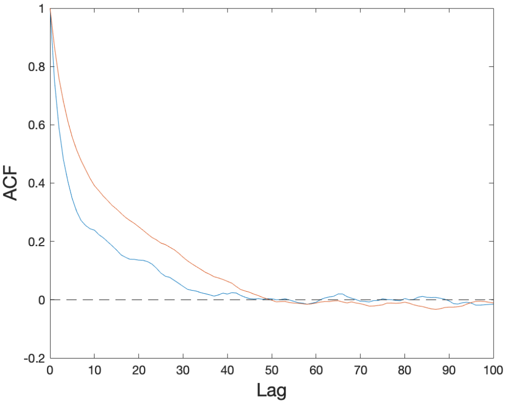
Evaluating an MCMC kernel

Assume that you have two different MCMC kernels targeting the same distribution. How do you decide which one is best?



Evaluating an MCMC kernel

Assume that you have two different MCMC kernels targeting the same distribution. How do you decide which one is best?



Looking at the autocorrelation functions can be a good help.

The Gibbs Markov kernel

We wish to sample from $\pi(\mathbf{x}) = \pi(x_1, x_2, \ldots, x_d)$.

Given a sample $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ we generate a new sample $\mathbf{x}$ in the following way:

For $j = 1, \ldots, d$

$$\text{Sample } x_j^* \sim \pi(x_j \mid x_1^*, \ldots, x_{j-1}^*, x_{j+1}, \ldots, x_d)$$

The **Gibbs kernel** defined above defines a Markov kernel with stationary distribution π.

Extensions and Improvements

So far we have discussed the Gibbs Markov kernel, there are many extensions and improvements compared with the standard algorithm.

- **Blocking**, instead of sampling one variable at the time we sample a block of variables. These blocks may overlap.

Extensions and Improvements

So far we have discussed the Gibbs Markov kernel, there are many extensions and improvements compared with the standard algorithm.

- **Blocking**, instead of sampling one variable at the time we sample a block of variables. These blocks may overlap.
- **Collapsing**, if we can analytically integrate out some variables and then use Gibbs on the remaining.

Extensions and Improvements

So far we have discussed the Gibbs Markov kernel, there are many extensions and improvements compared with the standard algorithm.

- **Blocking**, instead of sampling one variable at the time we sample a block of variables. These blocks may overlap.
- **Collapsing**, if we can analytically integrate out some variables and then use Gibbs on the remaining.
- **Random scan** Select the components/blocks to sample randomly (with or without replacement)

Extensions and Improvements

So far we have discussed the Gibbs Markov kernel, there are many extensions and improvements compared with the standard algorithm.

- **Blocking**, instead of sampling one variable at the time we sample a block of variables. These blocks may overlap.
- **Collapsing**, if we can analytically integrate out some variables and then use Gibbs on the remaining.
- **Random scan** Select the components/blocks to sample randomly (with or without replacement)
- Use other **MCMC kernels within Gibbs**.

Preparatory exercise

Given data $x_{1:n}$ from distribution

$$p(x_i|\pi, \boldsymbol{\mu}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x_i|\mu_k, I),$$

with prior distributions

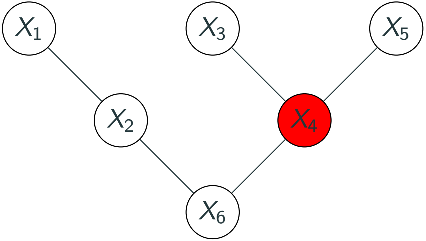
$$\begin{aligned} p(\pi) &= \mathrm{Dir}(\pi|\alpha_1, \ldots, \alpha_K) \\ p(\mu_k) &= \mathcal{N}(\mu_k|m, S). \end{aligned}$$

Construct a Gibbs sampler to sample from the posterior $p(\pi, \boldsymbol{\mu}|x_{1:n})$.

Gibbs sampling in graphical models

Assume that we are supposed to sample from the conditional of the node X_4 marked in red. Which other nodes will be included in the full conditional?

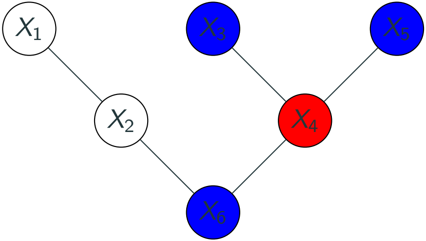
Undirected graph



Gibbs sampling in graphical models

Assume that we are supposed to sample from the conditional of the node X_4 marked in red. Which other nodes will be included in the full conditional?

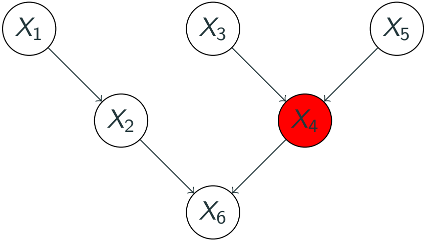
Undirected graph



Gibbs sampling in graphical models

Assume that we are supposed to sample from the conditional of the node X_4 marked in red. Which other nodes will be included in the full conditional?

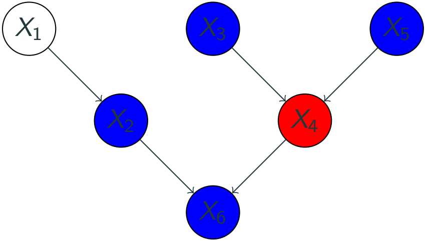
Directed graph



Gibbs sampling in graphical models

Assume that we are supposed to sample from the conditional of the node X_4 marked in red. Which other nodes will be included in the full conditional?

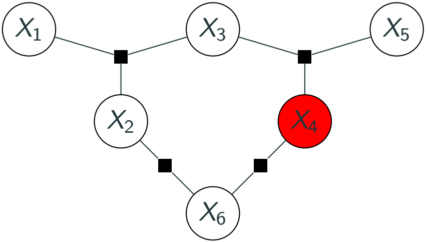
Directed graph



Gibbs sampling in graphical models

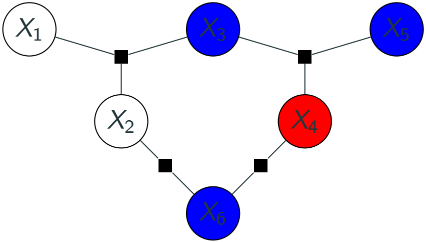
Assume that we are supposed to sample from the conditional of the node X_4 marked in red. Which other nodes will be included in the full conditional?

Factor graph



Assume that we are supposed to sample from the conditional of the node X_4 marked in red. Which other nodes will be included in the full conditional?

Factor graph



Gibbs sampling in the Trueskill model

We are interested in sampling from the posterior

$$p(\mathbf{w}, \mathbf{t} \mid \mathbf{y}) \propto \prod_{i=1}^{M} f_i(w_i) \prod_{k=1}^{N} g_k(t_k, w_{I_k}, w_{J_k}) h_k(t_k),$$

using **Gibbs sampling**.

Gibbs sampling in the Trueskill model

We are interested in sampling from the posterior

$$p(\mathbf{w}, \mathbf{t} \mid \mathbf{y}) \propto \prod_{i=1}^{M} f_i(w_i) \prod_{k=1}^{N} g_k(t_k, w_{I_k}, w_{J_k}) h_k(t_k),$$

using **Gibbs sampling**.

1. Initialize $\mathbf{w}$, e.g. from the prior $f(\mathbf{w})$.

Gibbs sampling in the Trueskill model

We are interested in sampling from the posterior

$$p(\mathbf{w}, \mathbf{t} \mid \mathbf{y}) \propto \prod_{i=1}^{M} f_i(w_i) \prod_{k=1}^{N} g_k(t_k, w_{I_k}, w_{J_k}) h_k(t_k),$$

using **Gibbs sampling**.

1. Initialize $\mathbf{w}$, e.g. from the prior $f(\mathbf{w})$.
2. For each iteration:

Gibbs sampling in the Trueskill model

We are interested in sampling from the posterior

$$p(\mathbf{w}, \mathbf{t} \mid \mathbf{y}) \propto \prod_{i=1}^{M} f_i(w_i) \prod_{k=1}^{N} g_k(t_k, w_{I_k}, w_{J_k}) h_k(t_k),$$

using **Gibbs sampling**.

1. Initialize $\mathbf{w}$, e.g. from the prior $f(\mathbf{w})$.
2. For each iteration:
    1. Sample the performance for each game from

       $$p(t_k \mid w_{I_k}, w_{J_k}, y_k) \propto g_k(t_k, w_{I_k}, w_{J_k}) h_k(t_k) = \delta_{\mathrm{sign}(t_k)}(y_k) N(t_k \mid w_{I_k} - w_{J_k}, 1)$$

    2. Jointly sample the skills

       $$p(\mathbf{w} \mid \mathbf{t}, \mathbf{y}) = \underbrace{p(\mathbf{w} \mid \mathbf{t})}_{N(\mathbf{w} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})} \propto \prod_{i=1}^{M} \underbrace{f_i(w_i)}_{N(w_i \mid 0, \sigma_0^2)} \prod_{k=1}^{N} \underbrace{g_k(t_k, w_{I_k}, w_{J_k})}_{N(t_k \mid w_{I_k} - w_{J_k}, 1)}$$