Sampling Methods

Sara Wade
University of Cambridge

Charles University
8-19 April 2013, Prague

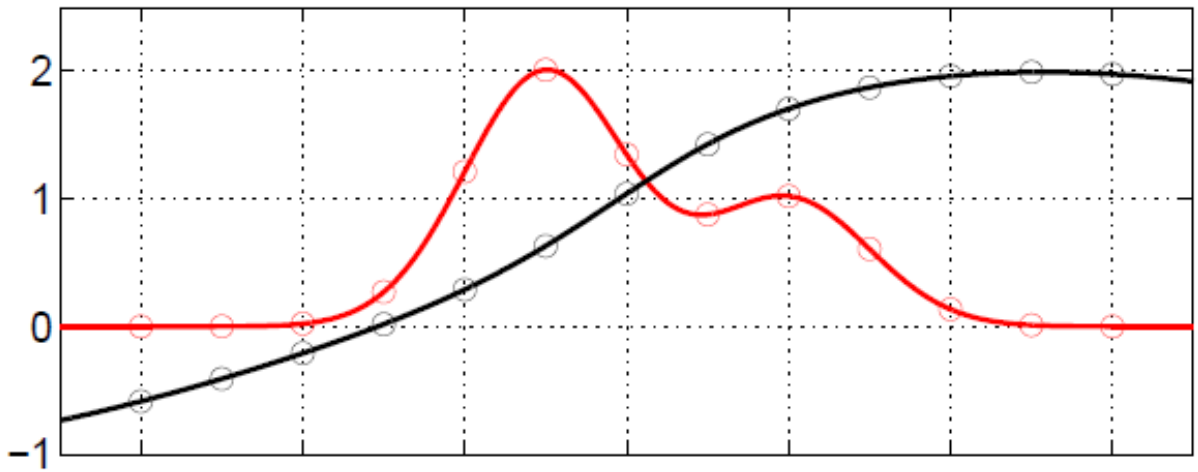
Numerical Integration

We want to compute $G = \mathsf{E}_p[g(X)] = \int g(x)p(x)dx$.

Solution 1: approximate integral by

$$\hat{G} = \sum_{t=1}^{T} g(x^t)p(x^t)\Delta x,$$

where x^t lie on an equidistant grid.

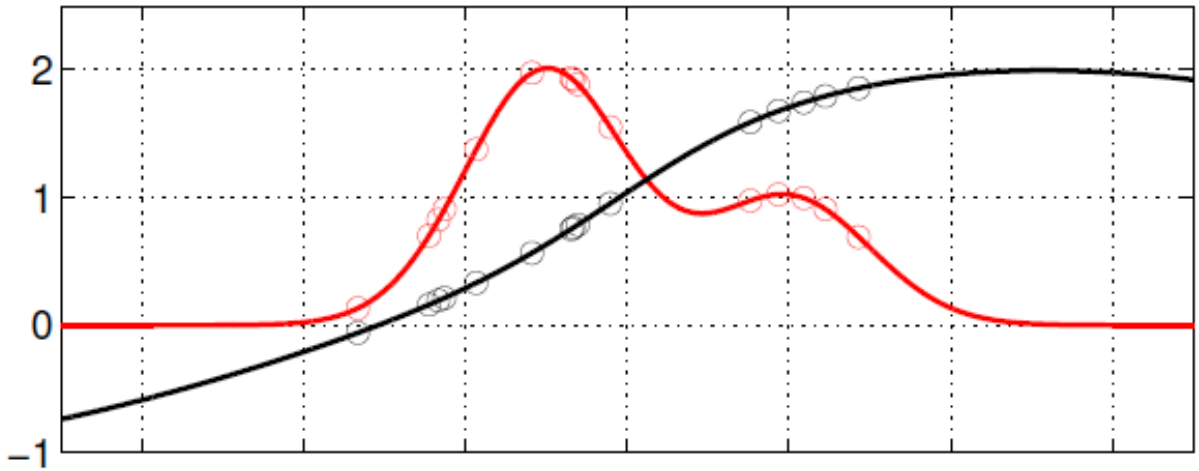


Problem: the number of grid points needed grows exponentially with the dimension of $\mathcal{X}$.

Monte Carlo

Solution 2: approximate integral by

$$\hat{G} = \frac{1}{T}\sum_{t=1}^{T} g(x^t), \quad \text{where } X^t \overset{iid}{\sim} p$$



Monte Carlo

Properties:

- **unbiased**: $\mathsf{E}[\hat{G}] = \frac{1}{T}\sum_{t=1}^{T}\mathsf{E}[g(X^t)] = G$.
- **convergence**: $\hat{G} \to \mathsf{E}_p[g(X)]$ as $T \to \infty$ (a.s.) under mild conditions (SLLN).
- **variance**: $\mathsf{V}(\hat{G}) = \frac{1}{T}\mathsf{V}(g(X)) \leftarrow$ doesn't depend on the dimension of x!

Monte Carlo

Properties:

- **unbiased**: $\mathsf{E}[\hat{G}] = \frac{1}{T}\sum_{t=1}^{T}\mathsf{E}[g(X^t)] = G$.
- **convergence**: $\hat{G} \to \mathsf{E}_p[g(X)]$ as $T \to \infty$ (a.s.) under mild conditions (SLLN).
- **variance**: $\mathsf{V}(\hat{G}) = \frac{1}{T}\mathsf{V}(g(X)) \leftarrow$ doesn't depend on the dimension of x!

But, how do we generate random samples from $p(x)$?

Probability integral transformation

Solution 1: Probability integral transformation:

If $X \sim F$, then $U = F(X) \sim \text{Unif}(0, 1)$.

$$\rightarrow X = F^{-1}(U) \sim F.$$

Note: F^{-1} is the generalized inverse $F^{-1}(u) = \inf_x\{x : F(x) \geq u\}$.

We can obtain a sample x from F by

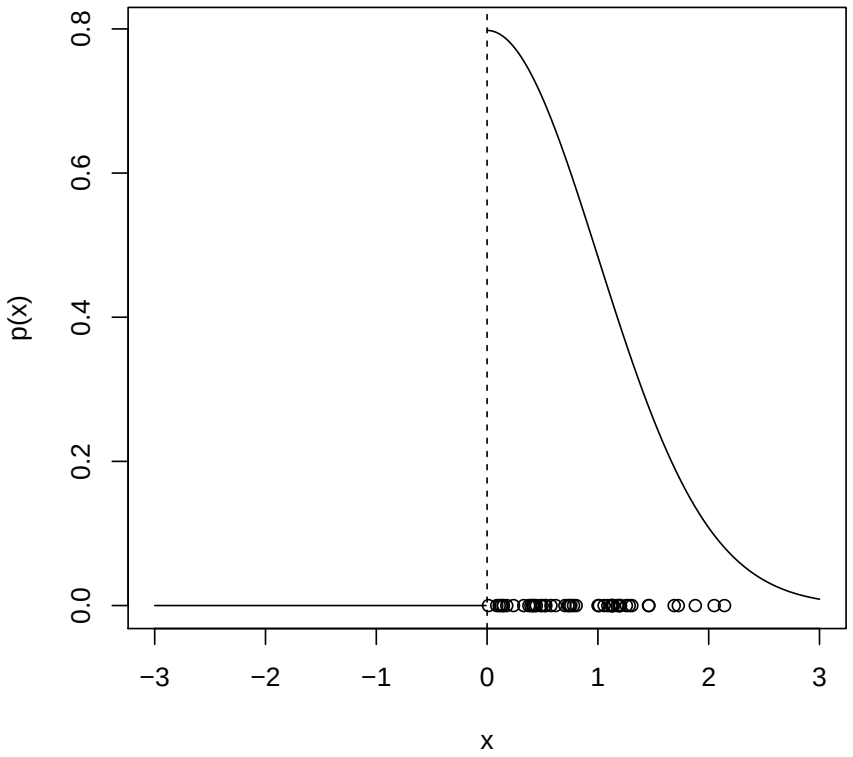
- sample $U \sim \text{Unif}(0, 1)$,
- set $x = F^{-1}(u)$.

Problems: need to compute F^{-1}, which can be expensive or unavailable.

Probability integral transformation

Ex. truncated normal $p(x) \propto \mathsf{N}(x|\mu, \sigma^2)\mathbf{1}(x \geq 0)$

- sample $U \sim \mathsf{Unif}(\Phi(-\mu/\sigma), 1)$,
- set $x = \Phi^{-1}(u)\sigma + \mu$.



Rejection sampling

Let $q(x)$ be a density (that can be easily sampled from) such that

$$p(x) \leq \frac{q(x)}{\alpha} \quad \forall\, x,$$

for some constant $\alpha \in (0, 1)$.

Solution 2: rejection sampling algorithm: obtain a sample x from $p(x)$ by

1. sample x^* from $q(x)$
2. accept $x = x^*$ with probability $\alpha(x^*) = \min(1, \frac{p(x^*)}{q(x^*)}\alpha)$, otherwise go to step 1.

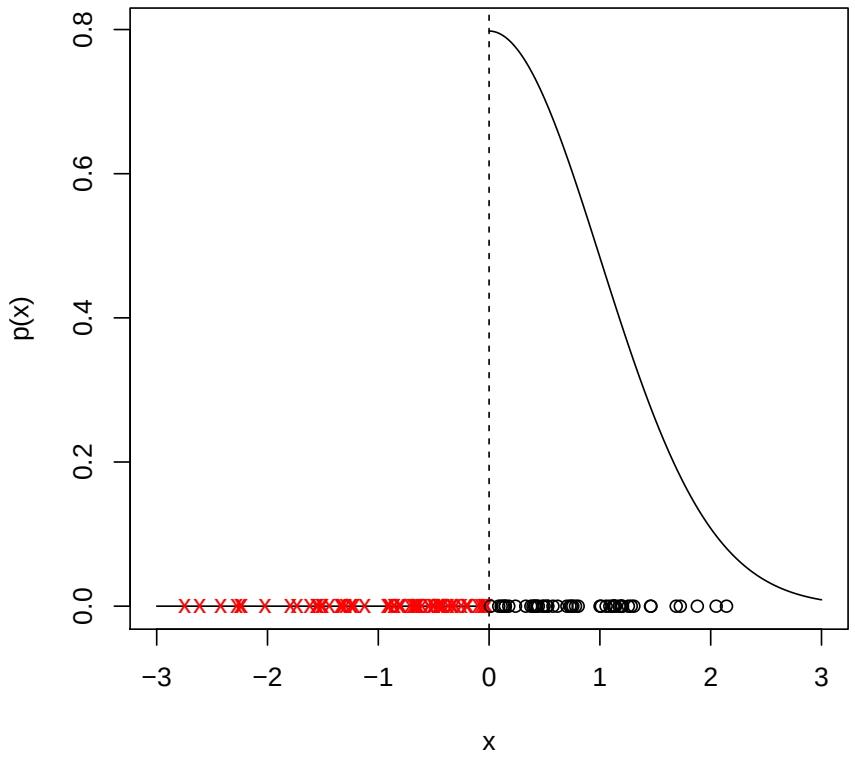
Problems: need to find $q(x)$ that can be **easily sampled** and such that the **number of rejections** is **small**.

Rejection sampling

Ex. truncated normal

$$p(x) \propto \mathsf{N}(x|\mu, \sigma^2)\mathbf{1}(x \geq 0); \quad q(x) = \mathsf{N}(x|\mu, \sigma^2),$$

1. sample x^* from $\mathsf{N}(\mu, \sigma^2)$,
2. set $x = x^*$ if $x^* \geq 0$, otherwise go to step 1.



Markov chain Monte Carlo

A third solution is to generate a Markov chain $\{X^t\}_{t \geq 1}$ with an appropriately defined *transition probability* such that for large enough t, X^t is approximately sampled from $p(x)$.

- Markov chain (first order): over the state space $\mathcal{X}$, is a sequence of random variables $X_1, X_2, \ldots$ such that

$$\pi(x^{t+1}|x^1, \ldots, x^t) = \pi(x^{t+1}|x^t).$$

- Homogeneous: a Markov chain is homogeneous if the transition probabilities do not depend on t,

$$\pi(x^{t+1}|x^t) \leftarrow \text{ does not depend on } t.$$

Markov chain Monte Carlo

- Stationary distribution: $p(x)$ is stationary with respect to a homogeneous Markov chain if

$$p(x) = \sum_{x^t} \pi(x|x^t)p(x^t).$$

A sufficient condition to ensure that $p(x)$ is a stationary distribution is the **detailed balance condition**:

$$p(x)\pi(x'|x) = p(x')\pi(x|x').$$

Markov chain Monte Carlo

If $\{X^t\}_{t\geq 1}$ is a homogeneous Markov chain such that

- state space is the support of $p(x)$,
- irreducible and aperiodic,
- has stationary distribution $p(x)$,

from ergodic theory, $X^t \xrightarrow{d} X$, where X has density $p(x)$ and

$$\frac{1}{T - T_0} \sum_{t=T_0}^{T} g(x^t) \to \mathsf{E}_p[g(X)] \text{ a.s.}$$

Thus, our Markov chain produces (dependent) samples from p can be used to approximate G.
$\to$ We generate a large Markov chain and discard samples to obtain nearly independent samples from $p(x)$.

Metropolis-Hastings

How to construct such a chain?

Such a chain can be obtained by introducing a **proposal density** $q(x|x^t)$ and choose a starting value x_0.
At each iteration:

- generate x^* from $q(x|x^t)$,
- with acceptance probability

$$\alpha(x^*|x^t) = \min\left(1, \frac{p(x^*)}{p(x^t)}\frac{q(x^t|x^*)}{q(x^*|x^t)}\right),$$

  set $x^{t+1} = x^*$, otherwise set $x^{t+1} = x^t$.

The transition probability:

$$\pi(x^{t+1}|x^t) = q(x^{t+1}|x^t)\alpha(x^{t+1}|x^t) + r(x^t)\delta_{x^t}(x^{t+1}),$$

where $r(x^t) = \int q(x|x^t)\alpha(x|x^t)dx$, satisfies the detailed balance condition with respect to $p(x)$. For large enough T_0, $(x_t)_{t\geq T_0}$ are approximate samples from $p(x)$.

Independent Metropolis Hasting

The proposal density $q(x)$ is independent of x^t and such that

$$p(x) \leq \frac{q(x)}{\alpha},$$

for some constant $\alpha \in (0, 1)$.

The acceptance probability reduces to

$$\alpha(x^*|x^t) = \min\left(1, \frac{p(x^*)}{p(x^t)}\frac{q(x^t)}{q(x^*)}\right).$$

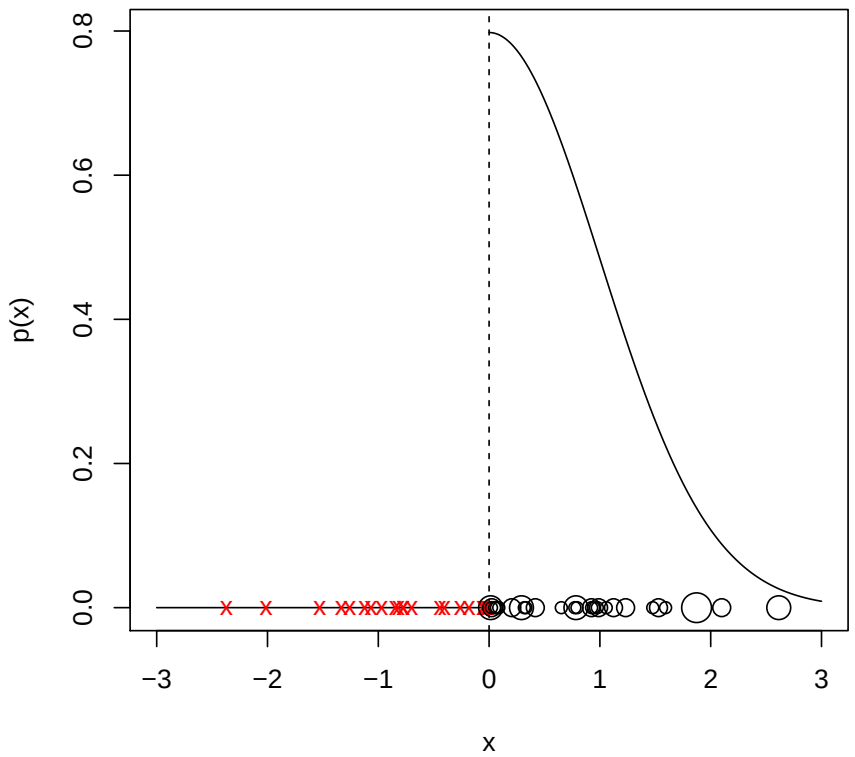
This algorithm is similar to rejection sampling but produces dependent samples, but it can be shows that on average independent MH accepts more often.

Independent MH

Ex. truncated normal

$$p(x) \propto \mathsf{N}(x|\mu, \sigma^2)\mathbf{1}(x \geq 0); \quad q(x) = \mathsf{N}(x|\mu, \sigma^2),$$

- sample x^* from $\mathsf{N}(\mu, \sigma^2)$,
- set $x^{t+1} = x^*$ if $x^* \geq 0$, otherwise set $x^{t+1} = x^t$.



Random walk

The candidate value is obtained by perturbing the previous value:

$$x^* = x^t + \epsilon,$$

where ϵ is independent of x^t with symmetric distribution centered at 0; for ex.

$$\epsilon \sim \mathsf{N}(0, \sigma^2).$$

The acceptance probability reduces to

$$\alpha(x^*|x^t) = \min\left(1, \frac{p(x^*)}{p(x^t)}\right).$$

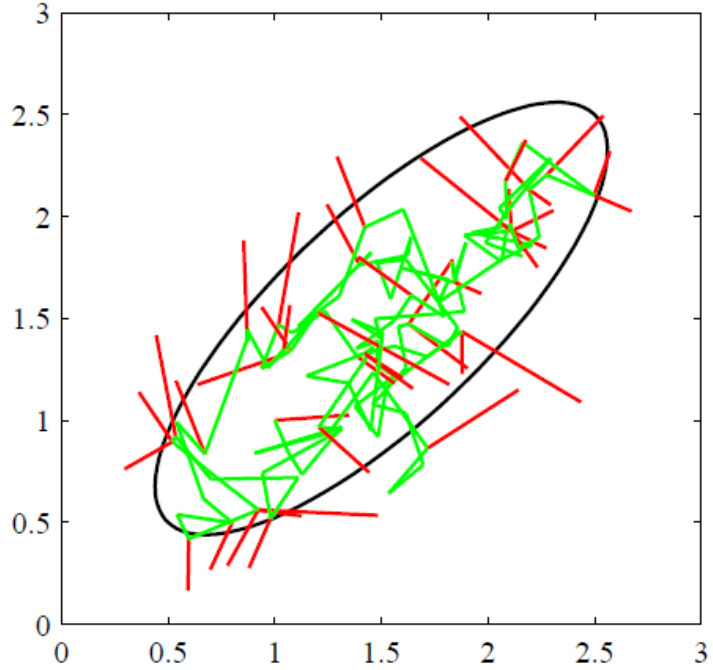
The scale of the proposal should be as large as possible without incurring high rejection rates.

Random walk

Ex. bivariate normal

$$p(x) = \mathsf{N}(x|\mu, \Sigma),$$

$$q(x|x^t) = \mathsf{N}(x_1|x_1^t, \sigma_q^2)\mathsf{N}(x_2|x_2^t, \sigma_q^2),$$



- sample x^* from

$$\mathsf{N}(x_1|x_1^t, \sigma_q^2)\mathsf{N}(x_2|x_2^t, \sigma_q^2)$$

,

- sample $U \sim \mathsf{Unif}(0, 1)$ and set

$$x^{t+1} = \begin{cases} x^* & \text{if } u < \min(1, \mathsf{N}(x^*|\mu, \Sigma)/\mathsf{N}(x^t|\mu, \Sigma)) \\ x^t & \text{otherwise} \end{cases}.$$

Gibbs sampling

Assume $\mathbf{x} = (x_1, \ldots, x_d)$. The **Gibbs sampling** algorithm iteratively samples x_i given $\mathbf{x}_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$.
Let $p(x_i|\mathbf{x}_{-i})$ be the full conditional density. The chain is initialized at x_0 and the Gibbs sampling algorithm procedes by

- sample x_1^{t+1} from $p(x_1|x_2^t, \ldots, x_d^t)$,
- sample x_2^{t+1} from $p(x_2|x_1^{t+1}, x_3^t, \ldots, x_d^t)$,

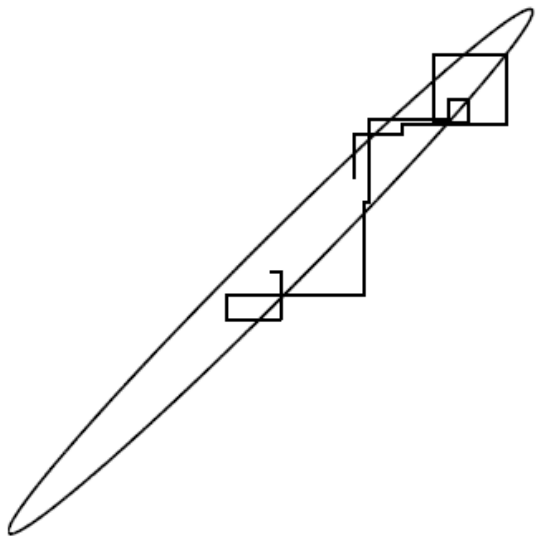
  $\vdots$
- sample x_d^{t+1} from $p(x_d|x_1^{t+1}, \ldots, x_{d-1}^{t+1})$,

The Gibbs sampling procedure is a special case of MH where the proposal is always accepted.

Gibbs sampling

Ex. bivariate normal $p(x) = \mathsf{N}(x|\mu, \Sigma)$:

- sample x_1^{t+1} from $\mathsf{N}(x_1|\beta_{0,1} + \beta_{1,1}x_2^t, \sigma^2_{1|2})$,
- sample x_2^{t+1} from $\mathsf{N}(x_2|\beta_{0,2} + \beta_{1,2}x_1^{t+1}, \sigma^2_{2|1})$.



Strong correlations can slow down Gibbs sampling.

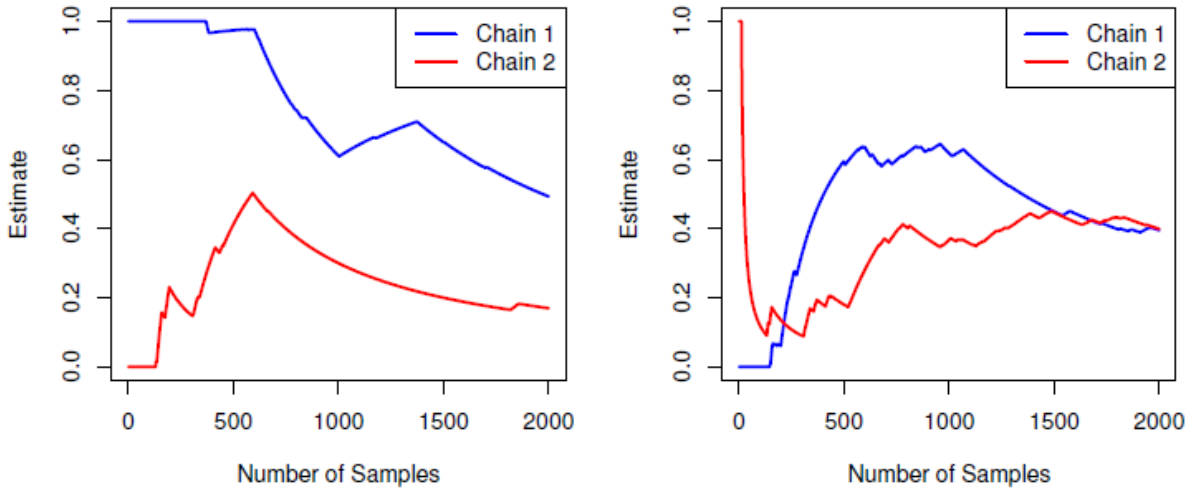
If groups of variables can be updated jointly, we can improve the mixing; this is known as **blocked Gibbs sampling**.

Convergence of MCMC

Samples need to be discarded so that $(x^t)_{t=1}^T$ are independent samples from $p(x)$:

- **burn in**: to overcome poor initialization, we may need to throw away the first T_0 samples.
- **thinning**: to reduce dependency, we may need to keep only every k^{th} sample.

It is difficult to monitor convergence, but we can determine lack of convergence via trace plots, autocorrelations, and comparing chains with different initializations.



Summary of MCMC

Advantages:

- general, applicable to many problems,
- easy to implement,
- theoretical guarantees as $T \to \infty$.

Disadvantages:

- can be slow and expensive to compute,
- difficult to assess convergence.

MCMC in Bayesian inference

In Bayesian inference, a closed form for the posterior distribution is often unavailable.

MCMC can be used to obtain samples $(\theta^t)_{t=1}^T$ from the posterior $p(\theta|\mathbf{x})$, and

$$\mathsf{E}[g(\theta)|\mathbf{x}] \approx \frac{1}{T}\sum_{t=1}^{T} g(\theta^t).$$

For example, the posterior distribution can be approximated by

$$P(A|\mathbf{x}) \approx \frac{1}{T}\sum_{t=1}^{T} \delta_{\theta^t}(A),$$

for any measurable set $A \subseteq \Theta$, and the predictive density at x_* is approximately

$$p(x_*|\mathbf{x}) \approx \frac{1}{T}\sum_{t=1}^{T} p(x_*|\theta^t, \mathbf{x}).$$

Probit regression

Model:

$$Y_i|x_i, w \overset{ind}{\sim} \text{Bern}(p(x_i)),$$
$$p(x_i) = p(Y_i = 1|x_i, w) = \Phi(w_0 + w_1 x_i).$$

Prior:

$$w \sim \text{N}(\mu_0, \Sigma_0).$$

Latent model: Introduce a latent variable $\tilde{Y}_i$ where

$$\tilde{Y}_i|x_i, w \overset{ind}{\sim} \text{N}(w_0 + w_1 x_i, 1),$$
$$y_i|\tilde{y}_i, x_i, w = \mathbf{1}(\tilde{y}_i \geq 0)$$

Note: if we integrate out $\tilde{Y}_i$, we recover the original model.

Probit regression - Gibbs sampler

Gibbs sampling algorithm: Initialize $w, \tilde{\mathbf{y}}$

- sample $w|\tilde{\mathbf{y}}, \mathbf{x} \sim \mathsf{N}(\hat{w}, \hat{\Sigma})$, where

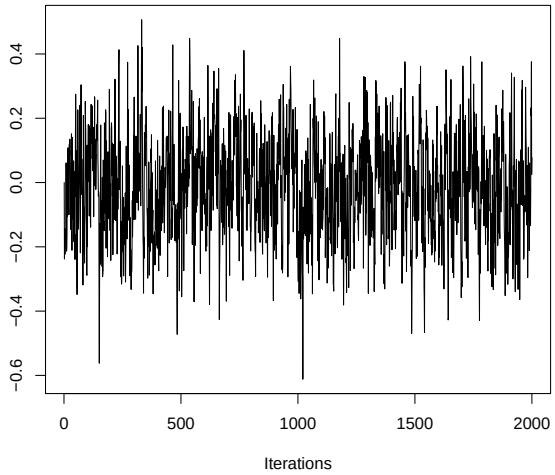
$$\hat{\Sigma} = (\Sigma_0^{-1} + X'X)^{-1},$$

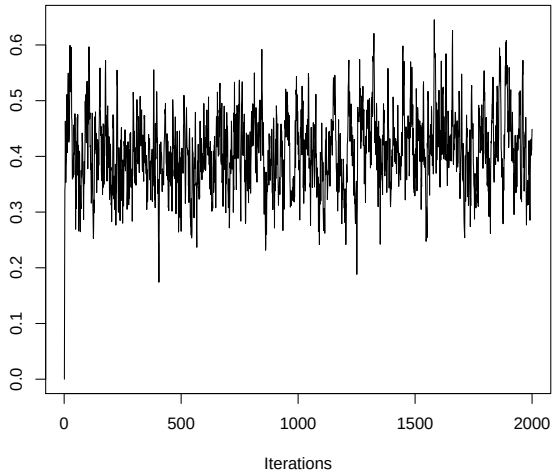
$$\hat{w} = \hat{\Sigma}(\Sigma_0^{-1}\mu_0 + X'\mathbf{y}).$$

- sample $\tilde{Y}_i|w, x_i, y_i$ independently with density

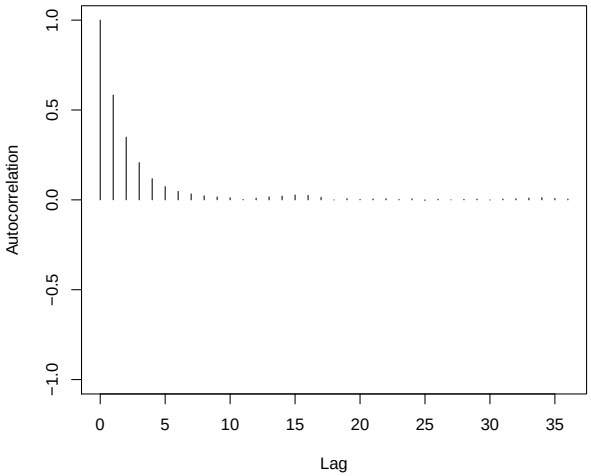
$$\frac{1}{Z}\mathsf{N}(\tilde{y}_i|w_0 + w_1 x_i, 1)(\mathbf{1}(\tilde{y}_i \geq 0))^{y_i}(\mathbf{1}(\tilde{y}_i < 0))^{1-y_i}.$$

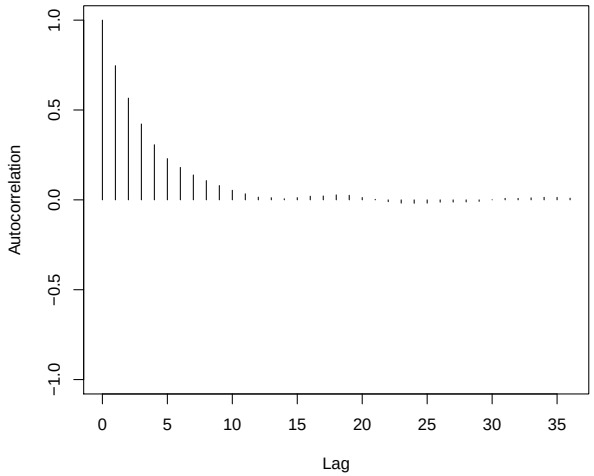
Probit regression





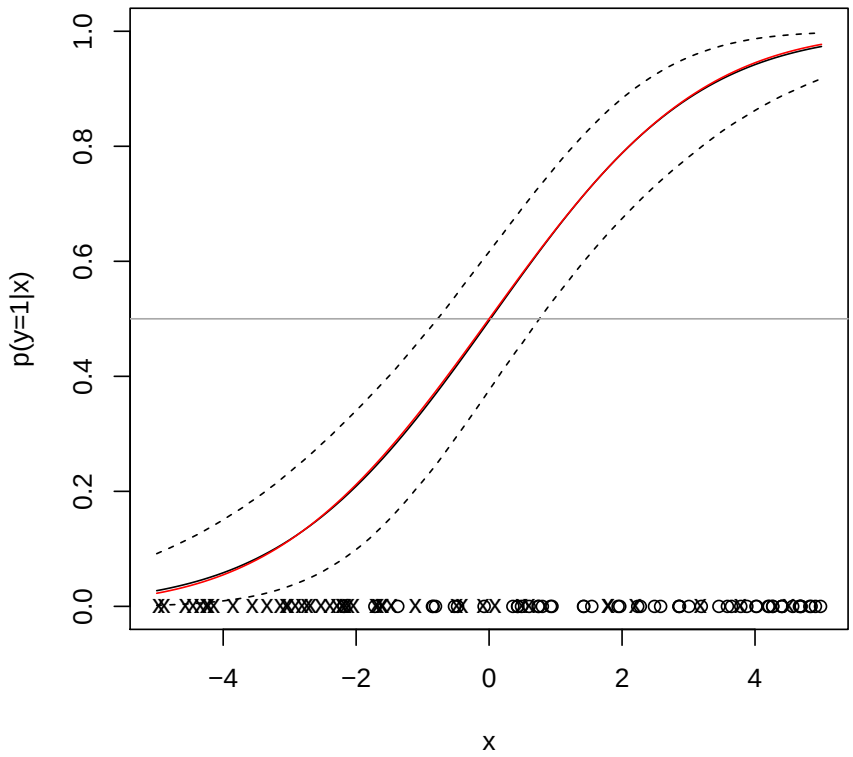
← Trace plot of first 2000 samples for w_0 (left) and w_1 (right)





← Autocorrelation for w_0 (left) and w_1 (right)

Probit regression



Posterior mean of $p(Y_* = 1|x_*, w)$ at a grid of new values x_* with 95% pointwise credible intervals (and the true curve in red). The $n = 100$ data points are represented as crosses for $y_i = 0$ and circles for $y_i = 1$.

Logit regression

Model:

$$Y_i|x_i, w \overset{ind}{\sim} \mathsf{Bern}(p(x_i)),$$

$$p(x_i) = p(Y_i = 1|x_i, w) = \frac{\exp(w_0 + w_1 x_i)}{1 + \exp(w_0 + w_1 x_i)}.$$

Prior:

$$w \sim \mathsf{N}(\mu_0, \Sigma_0).$$

Logit regression - MH random walk

MH random walk algorithm: Initialize w

- sample $w^*|w^t$ with proposal density

$$q(w|w^t) = \mathsf{N}(w|w^t, V(\Sigma_0^{-1} + \hat{C}^{-1})^{-1}V),$$

  where $\hat{C}$ is variance-covariance matrix of the MLEs and V is a diagonal matrix of tuning parameters.

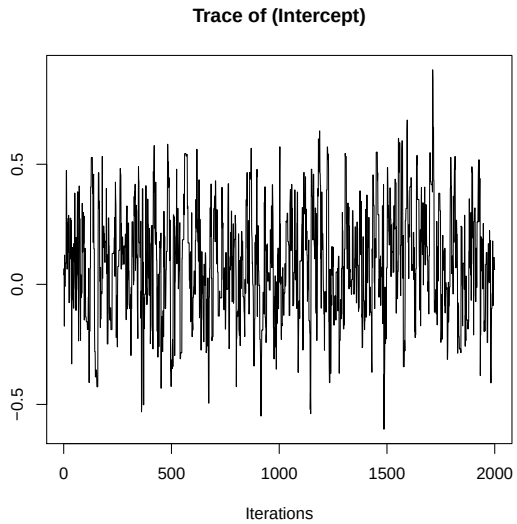
- accept $w^{t+1} = w^*$ with probability

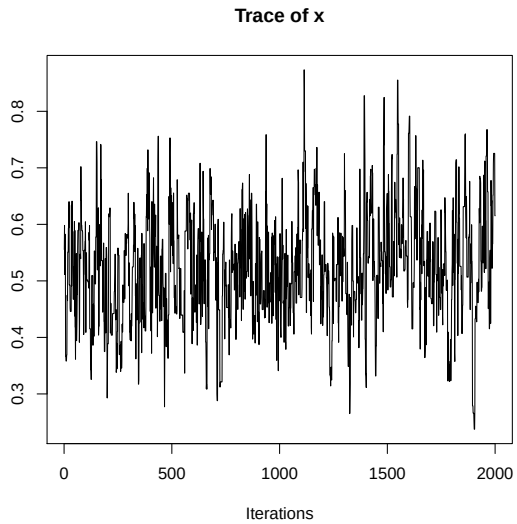
$$\alpha(x^*|x^t) = \min\left(1, \prod_{i=1}^{n}\left(\frac{\exp(w_0^* + w_1^* x_i)}{\exp(w_0^t + w_1^t x_i)}\right)^{y_i}\left(\frac{1+\exp(w_0^t + w_1^t x_i)}{1+\exp(w_0^* + w_1^* x_i)}\right)\right)$$

  otherwise set $w^{t+1} = w^t$.

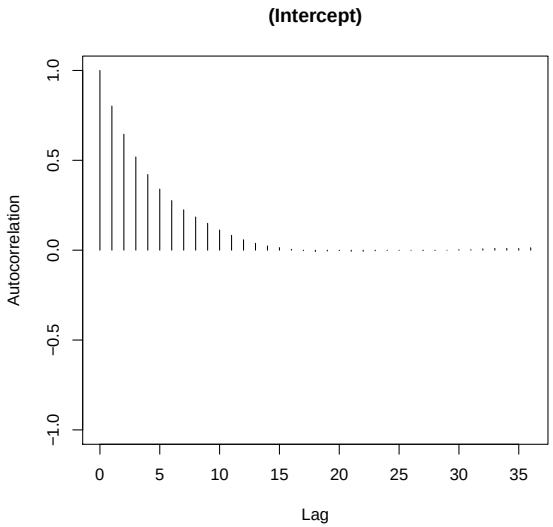
Implemented in *MCMClogit* function of the R package *MCMCpack*.

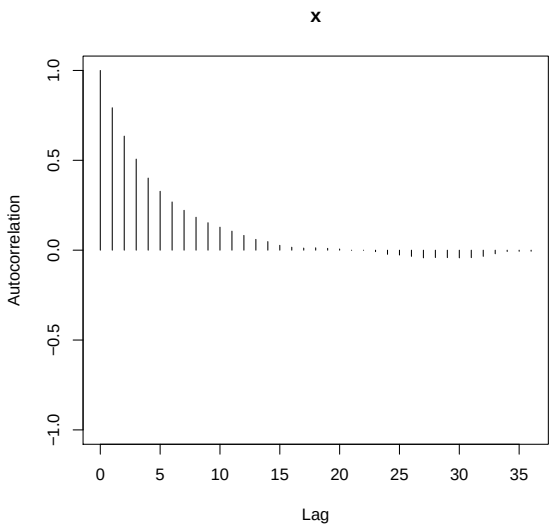
Logit regression





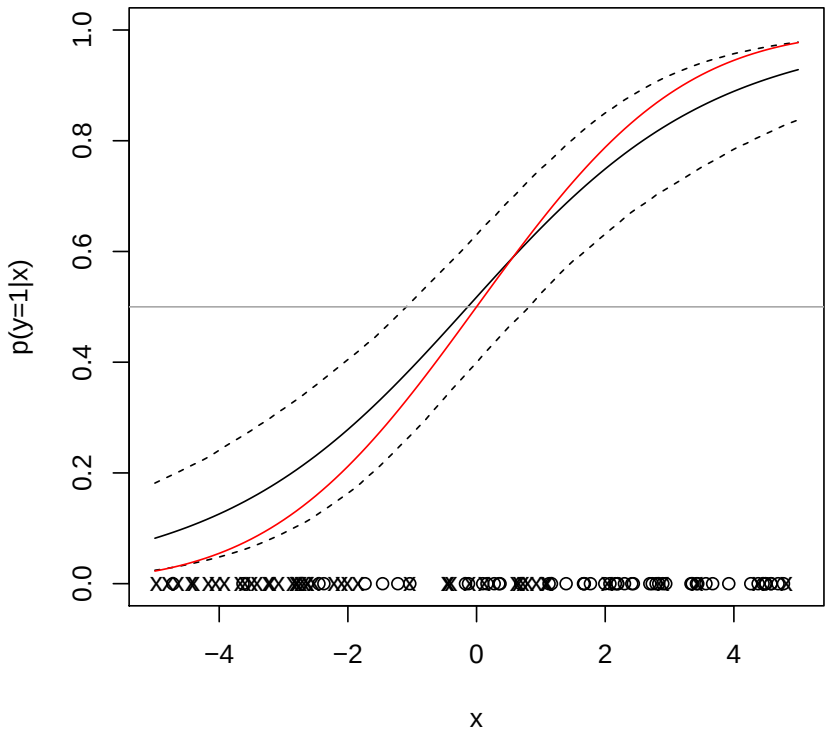
← Trace plot of first 2000 samples for w_0 (left) and w_1 (right)





← Autocorrelation for w_0 (left) and w_1 (right)

Logit regression



Posterior mean of $p(Y_* = 1|x_*, w)$ at a grid of new values x_* with 95% pointwise credible intervals (and the true curve in red). The $n = 100$ data points are represented as crosses for $y_i = 0$ and circles for $y_i = 1$.

References

- Chib, S. and Greenberg, E. (1995). Understanding the Metropolis-Hasting algorithm. *The American Statistician*, 49, 4:327-335.
- Casella, G. and George, E. (1992). Explaining Gibbs sampler. *The American Statistician*, 46, 3:167-174.