Lecture 12: Markov Chain Monte Carlo

Deniz Akyildiz

MATH60047/70047 – Stochastic Simulation

November 15, 2022

Imperial College
London

Announcements

Announcements

Imperial College London

- Next Tuesday (22 Nov): HXLY 414, Maths Learning Centre 4-6pm.
- Solving exercises, some extra coding, and problem session.

Imperial College London

A brief recap of what happened:

A brief recap of what happened:

- Sampling methods: $X_i \sim p_\star$ (from now on we will call it $p_\star$)
  - Direct sampling methods (Inversion, transformation)
  - Rejection sampling

A brief recap of what happened:

- Sampling methods: $X_i \sim p_\star$ (from now on we will call it $p_\star$)
  - Direct sampling methods (Inversion, transformation)
  - Rejection sampling
- Integration: $\bar{\varphi} = \int \varphi(x) p_\star(x) \mathrm{d}x$:
  - Using i.i.d samples from $p_\star$:

$$\bar{\varphi} \approx \hat{\varphi}^N_{\mathsf{MC}} = \frac{1}{N} \sum_{i=1}^N \varphi(X_i).$$

  - Using samples from a *proposal* q:

$$\bar{\varphi} \approx \hat{\varphi}^N_{\mathsf{IS}} = \frac{1}{N} \sum_{i=1}^N \bar{\mathsf{w}}_i \varphi(X_i).$$

A brief recap of what happened:

- Sampling methods: $X_i \sim p_\star$ (from now on we will call it $p_\star$)
  - Direct sampling methods (Inversion, transformation)
  - Rejection sampling
- Integration: $\bar{\varphi} = \int \varphi(x) p_\star(x) \mathrm{d}x$:
  - Using i.i.d samples from $p_\star$:

$$\bar{\varphi} \approx \hat{\varphi}^N_{\mathsf{MC}} = \frac{1}{N} \sum_{i=1}^N \varphi(X_i).$$

  - Using samples from a *proposal* q:

$$\bar{\varphi} \approx \hat{\varphi}^N_{\mathsf{IS}} = \frac{1}{N} \sum_{i=1}^N \bar{\mathsf{w}}_i \varphi(X_i).$$

Today, we will talk about Markov chains.

But why?

But why?

- Recall we are interested in sampling from $p_\star$

But why?

- Recall we are interested in sampling from $p_\star$
- Rejection sampling has its problems (we discussed)

But why?

- Recall we are interested in sampling from $p_\star$
- Rejection sampling has its problems (we discussed)
- Importance sampling is primarily an integration technique, not a sampling technique

But why?

- Recall we are interested in sampling from $p_\star$
- Rejection sampling has its problems (we discussed)
- Importance sampling is primarily an integration technique, not a sampling technique

Yes but how can we use Markov chains?

But why?

- Recall we are interested in sampling from $p_\star$
- Rejection sampling has its problems (we discussed)
- Importance sampling is primarily an integration technique, not a sampling technique

Yes but how can we use Markov chains?

- Markov chains have *stationary* distributions

But why?

- Recall we are interested in sampling from $p_\star$
- Rejection sampling has its problems (we discussed)
- Importance sampling is primarily an integration technique, not a sampling technique

Yes but how can we use Markov chains?

- Markov chains have *stationary* distributions
- We design the chain so that the stationary distribution is $p_\star$!

But why?

- Recall we are interested in sampling from $p_\star$
- Rejection sampling has its problems (we discussed)
- Importance sampling is primarily an integration technique, not a sampling technique

Yes but how can we use Markov chains?

- Markov chains have *stationary* distributions
- We design the chain so that the stationary distribution is $p_\star$!

Properties of Markov chains

What do we need?

We need Markov chains

Properties of Markov chains

What do we need?

We need Markov chains

- With invariant distributions
- Their convergence is ensured
- Their invariant distribution is unique

Properties of Markov chains

We've done so far

- We have seen how to simulate Markov chains

Properties of Markov chains

We've done so far

- We have seen how to simulate Markov chains
- We have seen the properties of *discrete space* chains to ensure convergence

Properties of Markov chains

We've done so far

- We have seen how to simulate Markov chains
- We have seen the properties of *discrete space* chains to ensure convergence
  - Irreducibility

Properties of Markov chains

We've done so far

- We have seen how to simulate Markov chains
- We have seen the properties of *discrete space* chains to ensure convergence
  - Irreducibility
  - Recurrence

Properties of Markov chains

We've done so far

- We have seen how to simulate Markov chains
- We have seen the properties of *discrete space* chains to ensure convergence
  - Irreducibility
  - Recurrence
  - Aperiodicity (leads to ergodicity)

Properties of Markov chains

We've done so far

- We have seen how to simulate Markov chains
- We have seen the properties of *discrete space* chains to ensure convergence
  - Irreducibility
  - Recurrence
  - Aperiodicity (leads to ergodicity)

We will now look at continuous space Markov chains.

Let our state space X be uncountable, e.g., $\mathsf{X} = \mathbb{R}$?

Let our state space X be uncountable, e.g., $\mathsf{X} = \mathbb{R}$?

In the continuous case, however, the analogous concepts are defined in a much more complicated way.

Let our state space X be uncountable, e.g., $\mathsf{X} = \mathbb{R}$?

In the continuous case, however, the analogous concepts are defined in a much more complicated way.

We will not go into the details here (which will require measure theoretic constructions), we will just now introduce the continuous state-space notation.

We assume now our state-space is uncountable, e.g., $\mathsf{X} = \mathbb{R}$.

What is a Markov chain?

The continuous case case

We assume now our state-space is uncountable, e.g., $\mathsf{X} = \mathbb{R}$.

We denote the initial *density* of the chain by $p_0(x)$.

What is a Markov chain?

The continuous case case

We assume now our state-space is uncountable, e.g., $\mathsf{X} = \mathbb{R}$.

We denote the initial *density* of the chain by $p_0(x)$.

The transition kernel is denoted $K(x_n|x_{n-1})$.

What is a Markov chain?

The continuous case case

We assume now our state-space is uncountable, e.g., $\mathsf{X} = \mathbb{R}$.

We denote the initial *density* of the chain by $p_0(x)$.

The transition kernel is denoted $K(x_n|x_{n-1})$.

The density of the chain at time n is denoted by $p_n(x_n)$.

A discrete-time Markov chain is a process $(X_n)_{n\in\mathbb{N}}$, when X is uncountable, satisfies:

What is a Markov chain?

The continuous case

A discrete-time Markov chain is a process $(X_n)_{n \in \mathbb{N}}$, when X is uncountable, satisfies:

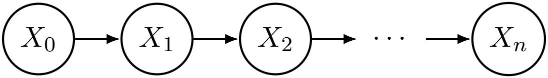
$$p(x_n|x_{1:n-1}) = p(x_n|x_{n-1}) = K(x_n|x_{n-1}).$$

What is a Markov chain?

The continuous case

A discrete-time Markov chain is a process $(X_n)_{n\in\mathbb{N}}$, when X is uncountable, satisfies:

$$p(x_n|x_{1:n-1}) = p(x_n|x_{n-1}) = K(x_n|x_{n-1}).$$



We will again consider the time-homogeneous case, i.e. the transition kernel is time-independent.

What is a Markov chain?

The continuous case

We will again consider the time-homogeneous case, i.e. the transition kernel is time-independent. A Markov chain therefore can be defined entirely by its:

- Initial state (or initial distribution)
- Transition kernel

The transition kernel is a density function $K(x_n|x_{n-1})$ for fixed x_{n-1}, i.e.,

$$\int_{\mathsf{X}} K(x_n|x_{n-1})\,\mathrm{d}x_n = 1.$$

Otherwise, it is a function of (x_n, x_{n-1}).

Consider the following Markov chain: $X_0 = 0$ and

$$K(x_n|x_{n-1}) = \mathcal{N}(x_n; ax_{n-1}, 1),$$

where $0 < a < 1$.

What is a Markov chain?

Example 1: Simulate a continuous-state Markov chain

Consider the following Markov chain: $X_0 = 0$ and

$$K(x_n | x_{n-1}) = \mathcal{N}(x_n; a x_{n-1}, 1),$$

where $0 < a < 1$.

We can simulate this chain by:

$$\begin{aligned} X_1 &\sim \mathcal{N}(0, 1) \\ X_2 &\sim \mathcal{N}(aX_1, 1) \\ X_3 &\sim \mathcal{N}(aX_2, 1) \\ &\vdots \\ X_n &\sim \mathcal{N}(aX_{n-1}, 1). \end{aligned}$$

Simulation.

The Chapman-Kolmogorov equation for the continuous case

$$p(x_n|x_{n-k}) = \int_{\mathsf{X}} K(x_n|x_{n-1})p(x_{n-1}|x_{n-k})\,\mathrm{d}x_{n-1},$$

for $k > 1$.

What is a Markov chain?

The continuous case: The evolution of the density of the chain

Let $p_0(x)$ be the initial density such that $X_0 \sim p_0(x)$.

Then, the density of the chain at time n is given by

$$p_n(x_n) = \int_{\mathsf{X}} K(x_n|x_{n-1}) p_{n-1}(x_{n-1}) \, \mathrm{d}x_{n-1}.$$

What is a Markov chain?

The continuous case: m-step transition kernel

It is useful for us to define the m-step transition kernel:

$$
\begin{aligned}
p(x_{m+n}|x_n) &= K^m(x_{m+n}|x_n), \\
&= \int_{\mathsf{X}} K(x_{m+n}|x_{m+n-1}) \cdots K(x_{n+1}|x_n)\, \mathrm{d}x_{m+n-1} \cdots \mathrm{d}x_{n+1}.
\end{aligned}
$$

We have the similar conditions of aperiodicity and irreducibility as in the discrete case, but,

- These are defined over *sets* rather than states.
- irreducibility is replaced by ϕ-irreducibility.
- aperiodicity is defined for sets

What is a Markov chain?

Properties

We have the similar conditions of aperiodicity and irreducibility as in the discrete case, but,

- These are defined over *sets* rather than states.
- irreducibility is replaced by ϕ-irreducibility.
- aperiodicity is defined for sets

We will not go into the details of these conditions for continuous space case.

A probability distribution $p_\star$ is called K-invariant if

$$p_\star(x) = \int_{\mathsf{X}} p_\star(x') K(x|x') \, \mathrm{d}x'.$$

Similar to the discrete case.

The detailed balance condition for the continuous case takes a similar form:

$$p_\star(x)K(x'|x) = p_\star(x')K(x|x').$$

The detailed balance condition for the continuous case takes a similar form:

$$p_\star(x)K(x'|x) = p_\star(x')K(x|x').$$

Note that this is a sufficient condition for stationarity of $p_\star$:

$$\begin{aligned}\int p_\star(x)K(x'|x)\mathrm{d}y &= \int p_\star(x')K(x|x')\mathrm{d}x', \\ \implies p_\star(x) &= \int K(x|x')p_\star(x')\mathrm{d}x',\end{aligned}$$

which implies $p_\star$ is K-invariant.

What is a Markov chain?

Example: Go back to Gaussian model

Consider the following Markov chain: $X_0 = 0$ and

$$K(x_n | x_{n-1}) = \mathcal{N}(x_n; a x_{n-1}, 1),$$

where $0 < a < 1$.

Consider the following Markov chain: $X_0 = 0$ and

$$K(x_n|x_{n-1}) = \mathcal{N}(x_n; ax_{n-1}, 1),$$

where $0 < a < 1$. Note that we can also write this as

$$X_n = aX_{n-1} + \epsilon_n,$$

where $\epsilon_n \sim \mathcal{N}(0, 1)$.

What is a Markov chain?

Example: Go back to Gaussian model

Prove that for

$$p_\star(x) = \mathcal{N}\left(x; 0, \frac{1}{1-a^2}\right),$$

the detailed balance condition is satisfied for the kernel

$$K(x_n|x_{n-1}) = \mathcal{N}(x_n; ax_{n-1}, 1),$$

where $0 < a < 1$.

Prove that $K^m(x_{m+n}|x_n)$ is given by

$$K^m(x_{m+n}|x_n) = \mathcal{N}\left(x_{m+n}; a^m x_n, \frac{1-a^{2m}}{1-a^2}\right).$$

Then prove that

$$p_\star(x) = \lim_{m \to \infty} K^m(x|x'),$$

independent of x'.

Hint: Use $X_n = aX_{n-1} + \epsilon_n$.

We will first look at a surprisingly simple approach: the Metropolis-Hastings algorithm.

We will first look at a surprisingly simple approach: the Metropolis-Hastings algorithm.

This approach relies on the following idea:

We will first look at a surprisingly simple approach: the Metropolis-Hastings algorithm.

This approach relies on the following idea:

- We can sample from a proposal $q(x|x')$ (that is a Markov kernel)

We will first look at a surprisingly simple approach: the Metropolis-Hastings algorithm.

This approach relies on the following idea:

- We can sample from a proposal $q(x|x')$ (that is a Markov kernel)
- We can use accept/reject

We will first look at a surprisingly simple approach: the Metropolis-Hastings algorithm.

This approach relies on the following idea:

- We can sample from a proposal $q(x|x')$ (that is a Markov kernel)
- We can use accept/reject

We can design the process so that the stationary distribution of the chain is the target distribution.

We will first look at a surprisingly simple approach: the Metropolis-Hastings algorithm.

This approach relies on the following idea:

- We can sample from a proposal $q(x|x')$ (that is a Markov kernel)
- We can use accept/reject

We can design the process so that the stationary distribution of the chain is the target distribution.

This is however very different from the rejection sampling approach.

Consider the following method:

- Sample $X' \sim q(x'|X_{n-1})$
- Set $X_n = X'$ with probability

$$\alpha(X_{n-1}, X') = \min\left\{1, \frac{p_\star(X')q(X_{n-1}|X')}{p_\star(X_{n-1})q(X'|X_{n-1})}\right\}.$$

- Otherwise, set $X_n = X_{n-1}$.

Consider the following method:

- Sample $X' \sim q(x'|X_{n-1})$
- Set $X_n = X'$ with probability

$$\alpha(X_{n-1}, X') = \min\left\{1, \frac{p_\star(X')q(X_{n-1}|X')}{p_\star(X_{n-1})q(X'|X_{n-1})}\right\}.$$

- Otherwise, set $X_n = X_{n-1}$.

Note the last step: we discard the sample X' if rejected BUT set $X_n = X_{n-1}$.

The ratio

$$\mathsf{r}(x, x') = \frac{p_\star(x')q(x|x')}{p_\star(x)q(x'|x)},$$

is called acceptance ratio.

We have discussed explicit kernels in the discrete and continuous cases.

We have discussed explicit kernels in the discrete and continuous cases.

But the MH algorithm automatically gives us a kernel.

We have discussed explicit kernels in the discrete and continuous cases.

But the MH algorithm automatically gives us a kernel.

How to prove that the stationary distribution is the target distribution?

Let us figure out the kernel: First question

- What is the probability of being at x_{n-1} and getting accepted?

$$a(x_{n-1}) = \int_{\mathsf{X}} \alpha(x|x_{n-1}) q(x|x_{n-1}) \mathrm{d}x.$$

- Therefore, the probability of being at x_{n-1} and getting rejected is $1 - a(x_{n-1})$.

We can see that the kernel is

$$K(x_n|x_{n-1}) = \alpha(x_n|x_{n-1}) q(x_n|x_{n-1}) + (1 - a(x_{n-1}))\delta_{x_{n-1}}(x_n).$$

We can now prove that the kernel satisfies the detailed balance condition:

$$K(x'|x)p_\star(x) = K(x|x')p_\star(x').$$

$$
\begin{aligned}
p_\star(x)K(x'|x) &= p_\star(x)q(x'|x)\alpha(x',x) + p_\star(x)(1-a(x))\delta_x(x') \\
&= p_\star(x)q(x'|x)\min\left\{1, \frac{p_\star(x')q(x|x')}{p_\star(x)q(x'|x)}\right\} + p_\star(x)(1-a(x))\delta_x(x') \\
&= \min\left\{p_\star(x)q(x'|x), p_\star(x')q(x|x')\right\} + p_\star(x)(1-a(x))\delta_x(x') \\
&= \min\left\{\frac{p_\star(x)q(x'|x)}{p_\star(x')q(x|x')}, 1\right\} p_\star(x')q(x|x') + p_\star(x')(1-a(x'))\delta_{x'}(x) \\
&= K(x|x')p_\star(x').
\end{aligned}
$$

Metropolis-Hastings

Unnormalised density

Imperial College London

Assume we are given an unnormalised density to sample $\bar{p}_\star$ where

$$p_\star(x) = \frac{\bar{p}_\star(x)}{Z},$$

where Z is the normalisation constant.

Metropolis-Hastings

Unnormalised density

- Sample $X' \sim q(x'|X_{n-1})$
- Set $X_n = X'$ with probability

$$\alpha(X_{n-1}, X') = \min\left\{1, \frac{\bar{p}_\star(X')q(X_{n-1}|X')}{\bar{p}_\star(X_{n-1})q(X'|X_{n-1})}\right\}.$$

- Otherwise, set $X_n = X_{n-1}$.

as the normalising constants of $p_\star$ would cancel out.

Metropolis-Hastings

How do we choose proposals?

- Independent proposals
- Symmetric (random walk) proposals
- Gradient-based proposals
- Adaptive proposals

Choose the proposal $q(x)$ independently of the current state X_{n-1}. Leads to

- $X' \sim q(x')$
- Accept with probability

$$\alpha(X_{n-1}, X') = \min\left\{1, \frac{p_\star(X')q(X_{n-1})}{p_\star(X_{n-1})q(X')}\right\}.$$

- Otherwise, set $X_n = X_{n-1}$.

Let us say

$$p_\star(x) = \mathcal{N}(x; \mu, \sigma^2)$$

For the example, assume we want to use MH to sample from it. Choose a proposal

$$q(x) = \mathcal{N}(x; \mu_q, \sigma_q^2).$$

How to compute the acceptance ratio?

Metropolis-Hastings

Independent proposals

$$
\begin{aligned}
\mathsf{r}(x, x') &= \frac{p_\star(x')q(x)}{p_\star(x)q(x')} \\
&= \frac{\mathcal{N}(x'; \mu, \sigma^2)\mathcal{N}(x; \mu_q, \sigma_q^2)}{\mathcal{N}(x; \mu, \sigma^2)\mathcal{N}(x'; \mu_q, \sigma_q^2)} \\
&= \frac{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x'-\mu)^2}{2\sigma^2}\right) \frac{1}{\sqrt{2\pi\sigma_q^2}} \exp\left(-\frac{(x-\mu_q)^2}{2\sigma_q^2}\right)}{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \frac{1}{\sqrt{2\pi\sigma_q^2}} \exp\left(-\frac{(x'-\mu_q)^2}{2\sigma_q^2}\right)} \\
&= \frac{\exp\left(-\frac{(x'-\mu)^2}{2\sigma^2}\right) \exp\left(-\frac{(x-\mu_q)^2}{2\sigma_q^2}\right)}{\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \exp\left(-\frac{(x'-\mu_q)^2}{2\sigma_q^2}\right)} \\
&= e^{\left(-\frac{1}{2\sigma^2}\left[(x'-\mu)^2-(x-\mu)^2\right]\right)} e^{\left(-\frac{1}{2\sigma_q^2}\left[(x-\mu_q)^2-(x'-\mu_q)^2\right]\right)}
\end{aligned}
$$

Simulation.

Metropolis-Hastings

Random walk proposal

We can choose:

$$q(x'|x) = \mathcal{N}(x'; x, \sigma_q^2)$$

The proposal looks at where we are and take a random step (random walk).

We can choose:

$$q(x'|x) = \mathcal{N}(x'; x, \sigma_q^2)$$

The proposal looks at where we are and take a random step (random walk).

Note that $q(x'|x)$ is symmetric, i.e. $q(x|x') = q(x'|x)$.

Acceptance ratio:

$$
\begin{aligned}
\mathsf{r}(x, x') &= \frac{p_\star(x')q(x|x')}{p_\star(x)q(x'|x)} \\
&= \frac{p_\star(x')}{p_\star(x)}, \\
&= \frac{\mathcal{N}(x'; \mu, \sigma^2)}{\mathcal{N}(x; \mu, \sigma^2)} \\
&= e^{\left(-\frac{1}{2\sigma^2}\left[(x'-\mu)^2-(x-\mu)^2\right]\right)}.
\end{aligned}
$$

Simulation.

Metropolis-Hastings

Random walk proposal

Set a burnin period:

- Run the sampler for fixed number of iterations and discard the first n samples.
- This accounts for the convergence to the stationary measure.

We can *inform* the proposal by using the gradient of the target distribution.

$$q(x'|x) = \mathcal{N}(x'; x + \gamma \nabla \log p_\star(x), \sigma_q^2),$$

This tends to behave really well.

We can *inform* the proposal by using the gradient of the target distribution.

$$q(x'|x) = \mathcal{N}(x'; x + \gamma \nabla \log p_\star(x), \sigma_q^2),$$

This tends to behave really well.

This approach is called *Metropolis adjusted Langevin algorithm* (MALA). (more on these later)

- One has to be careful that $p/q < \infty$ (while no theoretical reason, the performance tends to be quite bad).

Metropolis-Hastings

Caveats, design rules

- One has to be careful that $p/q < \infty$ (while no theoretical reason, the performance tends to be quite bad).
- The proposal should attain a balance of acceptance rate and efficiency.

- One has to be careful that $p/q < \infty$ (while no theoretical reason, the performance tends to be quite bad).
- The proposal should attain a balance of acceptance rate and efficiency.
- Too high acceptance rate is **not** necessarily good: You might be taking too small steps and getting stuck in some regions

Let us look at now the Bayesian inference problem.

We can solve it in full generality (in theory) using MH.

Recall the general formulation

$$p(x|y_{1:n}) = \frac{p(y_{1:n}|x)p(x)}{p(y_{1:n})} = \frac{\prod_{i=1}^n p(y_i|x)p(x)}{p(y_{1:n})},$$

when $y_1, \ldots, y_n$ are conditionally independent given x.

We write

$$p(x|y_{1:n}) \propto \prod_{i=1}^{n} p(y_i|x)p(x),$$

and set

$$\bar{p}_\star(x) = \prod_{i=1}^{n} p(y_i|x)p(x),$$

as our unnormalised posterior.

The generic MH for Bayesian inference, given x_{n-1}

- Sample $X' \sim q(x'|x_{n-1})$.
- Accept $x_n = x'$ with probability

$$\alpha(x_{n-1}, x') = \min\left\{1, \frac{\bar{p}_\star(x')q(x_{n-1}|x')}{\bar{p}_\star(x_{n-1})q(x'|x_{n-1})}\right\}.$$

- Otherwise, $X_n = x_{n-1}$.

Recall our example about localising a source using observations from a sensor network.

We can now formalise this problem. Assume that the source is located at $x \in \mathbb{R}^2$ and the sensor network is located at $s_1, \ldots, s_3 \in \mathbb{R}^2$ (3 sensors).

Assume that these three sensors "observe" the source according to:

$$p(y_i|x, s_i) = \mathcal{N}(y_i; \|x - s_i\|, R),$$

where y_i is the observation from sensor i.

Metropolis-Hastings

Example: Source localisation

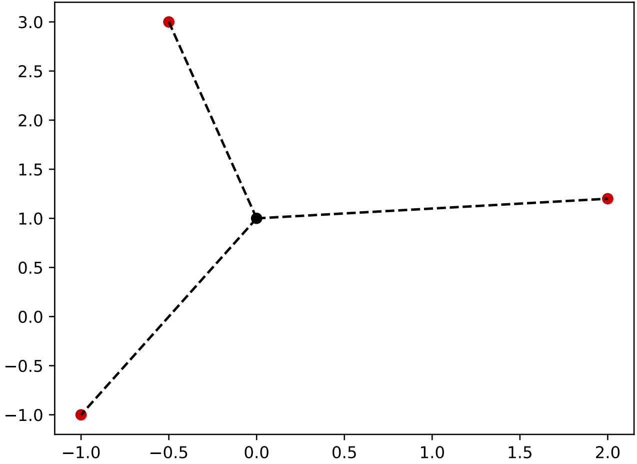


Figure: Source localisation

Assume that you are asked to estimate the location of the source given the observations y_1, y_2, y_3. What is the model?

Assume that you are asked to estimate the location of the source given the observations y_1, y_2, y_3. What is the model?

We first need a prior on the source location:

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

where μ is the prior mean and Σ is the prior covariance. We already have the likelihoods for each y_i.

The posterior is given by

$$p(x|y_1, y_2, y_3, s_1, s_2, s_3) \propto p(x) \prod_{i=1}^{3} p(y_i|x, s_i).$$

We choose a random walk proposal:

$$q(x'|x) = \mathcal{N}(x'; x, \sigma^2 I).$$

This is symmetric so the acceptance ratio is:

$$\mathsf{r}(x, x') = \frac{p(x')p(y_1|x', s_1)p(y_2|x', s_2)p(y_3|x', s_3)}{p(x)p(y_1|x, s_1)p(y_2|x, s_2)p(y_3|x, s_3)}.$$

Assume that we observe

$$Y_1, \ldots, Y_n | z, s \sim \mathcal{N}(y_i; z, s)$$

where we do not know z and s. Assume we have an independent prior on z and s:

$$p(z)p(s) = \mathcal{N}(z; m, \kappa^2)\mathcal{IG}(s; \alpha, \beta).$$

where $\mathcal{IG}(s; \alpha, \beta)$ is the inverse Gamma distribution

$$\mathcal{IG}(s; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} s^{-\alpha-1} \exp\left(-\frac{\beta}{s}\right).$$

In other words, we have

$$p(z)p(s) = \frac{1}{\sqrt{2\pi\kappa^2}} \exp\left(-\frac{(z-m)^2}{2\kappa^2}\right) \frac{\beta^\alpha}{\Gamma(\alpha)} s^{-\alpha-1} \exp\left(-\frac{\beta}{s}\right).$$

We are after the posterior distribution

$$\begin{aligned} p(z, s | y_1, \dots, y_n) &\propto p(y_1, \dots, y_n | z, s) p(z) p(s), \\ &= \prod_{i=1}^n \mathcal{N}(y_i; z, s) \mathcal{N}(z; m, \kappa^2) \mathcal{IG}(s; \alpha, \beta). \end{aligned}$$

Let us call our unnormalised posterior as $\bar{p}_\star(z, s|y_{1:n})$.

In order to do this, we need to design proposals over z and s. We choose a random walk proposal for z:

$$q(z'|z) = \mathcal{N}(z'; z, \sigma_q^2).$$

and an independent proposal for s:

$$q(s') = \mathcal{IG}(s'; \alpha, \beta).$$

The joint proposal therefore is

$$q(z', s'|z, s) = \mathcal{N}(z'; z, \sigma_q^2)\mathcal{IG}(s'; \alpha, \beta).$$

Design the MH algorithm.

The acceptance ratio is

$$\mathsf{r}(z, s, z', s') = \frac{\bar{p}(z', s'|y_{1:n})q(z, s|z', s')}{p(z, s|y_{1:n})q(z', s'|z, s)}$$
$$= \frac{p(z')p(s')\left[\prod_{k=1}^{n}\mathcal{N}(y_k; z', s')\right]\mathcal{N}(z; z', \sigma_q^2)p(s)}{p(z)p(s)\left[\prod_{k=1}^{n}\mathcal{N}(y_k; z, s)\right]\mathcal{N}(z'; z, \sigma_q^2)p(s')}$$
$$= \frac{\mathcal{N}(z'; m, \kappa^2)\left[\prod_{k=1}^{n}\mathcal{N}(y_k; z', s')\right]}{\mathcal{N}(z; m, \kappa^2)\left[\prod_{k=1}^{n}\mathcal{N}(y_k; z, s)\right]}$$

Consider the 2D density

$$p(x, y) \propto \exp\left(-\frac{x^2}{10} - \frac{y^4}{10} - 2(y - x^2)^2\right).$$

Assume we would like to sample from it.

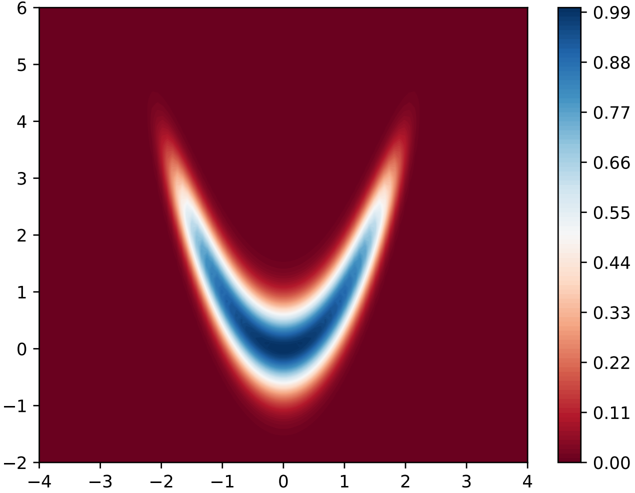


Figure: The banana density (unnormalised)

We have

$$\bar{p}_\star(x, y) = \exp\left(-\frac{x^2}{10} - \frac{y^4}{10} - 2(y - x^2)^2\right).$$

and let us choose two alternative proposals

- The random walk proposal:

$$q(x', y' | x, y) = \mathcal{N}(x'; x, \sigma_q^2)\mathcal{N}(y'; y, \sigma_q^2).$$

- and the gradient-based proposal (MALA):

$$q(x', y' | x, y) = \mathcal{N}(z; z + \gamma \nabla \log \bar{p}_\star(z), \sqrt{2\gamma}\mathbf{I}).$$

where $z = (x, y)$ and γ is a step size.

See you next week!

References I

Imperial College London