CS492: Probabilistic Programming

Markov Chain Monte Carlo

Hongseok Yang
KAIST

CS492: Probabilistic Programming

Markov Chain Monte Carlo

Hongseok Yang
KAIST

Really about: Metropolis-Hastings algorithm

```
(doquery :lmh induce-fn [ints2 outs2])
```

```
(doquery :lmh induce-fn [ints2 outs2])
```
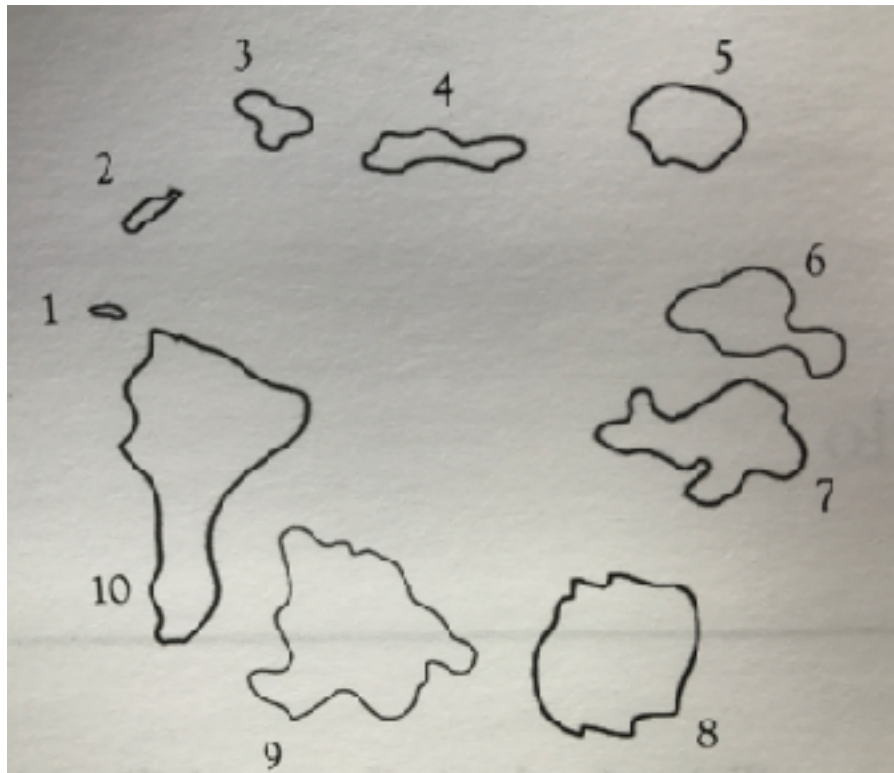
Lightweight Metropolis Hastings algorithm* (LMH).

Learning outcome

- Can explain Metropolis-Hastings algorithm.

- Can say when this algo. is correct.

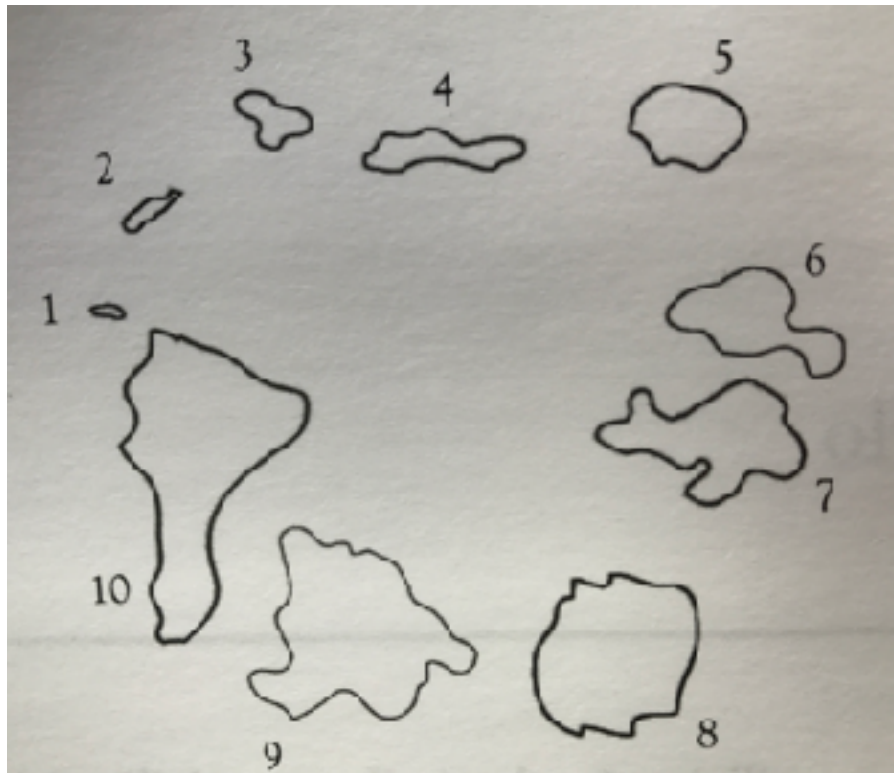- Can develop an instance of the algorithm.

Good king Markov puzzle*

Markov rules 10 islands.

100i people live in island i.

Good king Markov puzzle*

Markov rules 10 islands.

100i people live in island i.

King loves his people and wants to visit each island as often as its population size.

[Q] Find an algorithm. No scheduling nor bookkeeping. Can move adjacent islands only.

* Borrowed from McElreath's book "Statistical Rethinking"

Good king Markov puzzle*

Bad!!

1 day

10 days

9 days   8 days

Markov rules 10 islands.

100i people live in island i.

King loves his people and wants to visit each island as often as its population size.

[Q] Find an algorithm. No scheduling nor bookkeeping. Can move adjacent islands only.

* Borrowed from McElreath's book "Statistical Rethinking"

Good king Markov puzzle*

Markov rules 10 islands.

100i people live in island i.
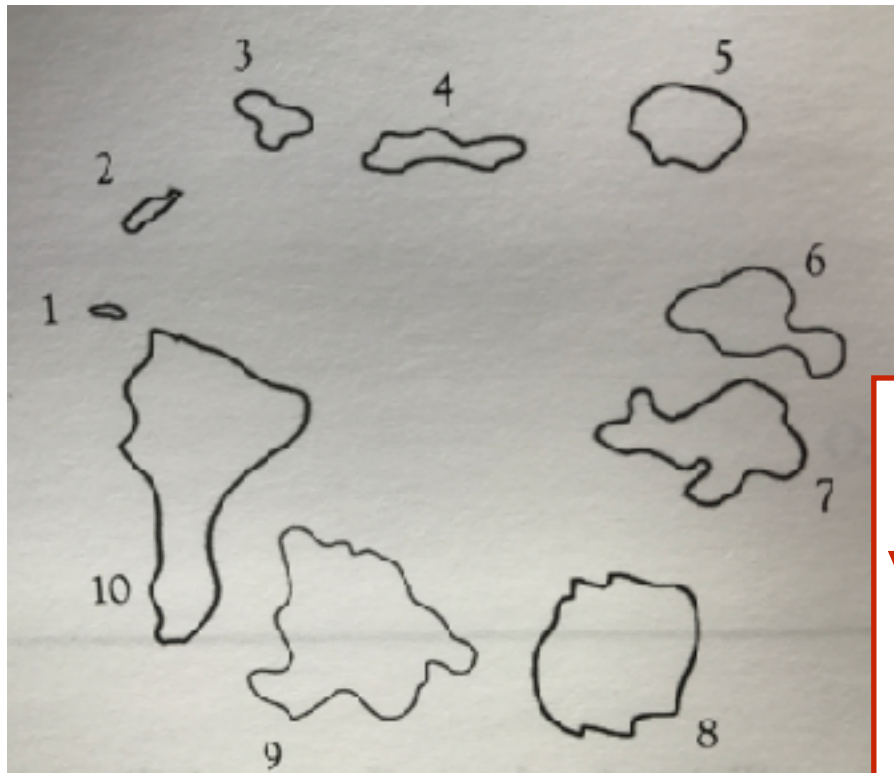
i ~ discrete(1,2,…,10). Bad!!
Visit i.
Repeat.

King loves his people and wants to visit each island as often as its population size.

[Q] Find an algorithm. No scheduling nor bookkeeping. Can move adjacent islands only.

* Borrowed from McElreath's book "Statistical Rethinking"

Good king Markov puzzle*
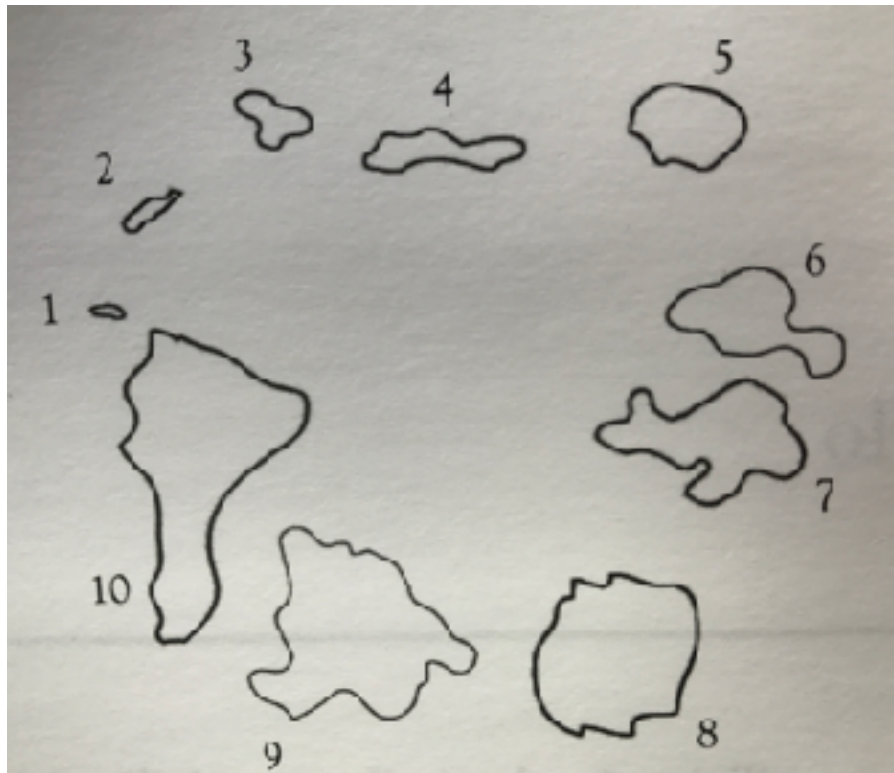

Markov rules 10 islands.

100i people live in island i.

King loves his people and wants to visit each island as often as its population size.

[Q] Find an algorithm. No scheduling nor bookkeeping. Can move adjacent islands only.

Solution

k_n — King's current island at step n.

Repeat the following steps:

1. Flip a coin with prob. 0.5. If head, move clockwise. If tail, move counterclockwise.

2. k' := target island. α := $\min(1, k'/k_n)$.

3. Flip a coin with prob. α. If head, k_{n+1} := k'. Otherwise, k_{n+1} := k_n.

Solution

k_n — King's current island at step n.

Repeat the following steps:

1. Flip a coin with prob. 0.5. If head, move clockwise. If tail, move counterclockwise.

2. $k' :=$ target island. $\alpha := \min(1, k'/k_n)$.

3. Flip a coin with prob. α. If head, $k_{n+1} := k'$. Otherwise, $k_{n+1} := k_n$.

[Q] Why correct? What does correctness even mean?

Sequence generated by the algorithm:
k_1, k_2, k_3, k_4, k_5, ..., k_n, ...

[Strong convergence] For any $f : \{1,..,10\} \to \mathbb{R}$,
$(\sum_{j \leq n} f(k_j))/n \longrightarrow \mathbb{E}_{p(i)}[f(i)]$ as $n \longrightarrow \infty$ with prob. 1

$p(i) = i/55$, the target prob. for the visit of i.

Holds because 1) the random move of the algo. has p as invariant; 2) the algo. may reach from one island to another in finitely many moves.

Metropolis algorithm

Goal: Generate a sample from target $r(x)/Z$, where $Z = \int r(x)dx$, the normalising constant.

Parameter: Conditional distribution $q(x'|x)$.

- Should be symmetric $q(x'|x) = q(x|x')$.

- Represents a random move.

- Called proposal kernel.

Metropolis algorithm

Goal: Generate a sample from target r(x)/Z, where Z= ∫ r(x)dx, the normalising constant.

Parameter: Conditional distribution q(x'|x).

- Should be symmetric q(x'|x) = q(x|x').

- Represents a random move.

- Called proposal kernel.

[Q] What are r(-) and q(-|-) in King Markov?

Metropolis algorithm

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

Metropolis

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   ≥ 1 for better x'
   < 1 for worse x'

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

Metropolis

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly; n:=1

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; n:=n+1

Metropolis

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

[Q1] How is it related to our sol. for King Markov?

Metropolis

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly;   n:=1

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim$ uniform$(0,1)$

   c) $x_{n+1} := $ (if $(u \leq \alpha)$ then x' else x_n);   n:=n+1

[Q2] Use it & do posterior inference. Assume $x \in \mathbb{R}^m$.

$r(x) = p(x|y)p(x)$
$q(x) = normal(x, \varepsilon \times ID)$ **s**

Noisy greedy exploration.
No need to know Z.

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly; n:=1

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := min(1, r(x')/r(x_n))$

   b) $u \sim uniform(0,1)$

   c) $x_{n+1} := (if\ (u \leq \alpha)\ then\ x'\ else\ x_n)$; n:=n+1

[Q2] Use it & do posterior inference. Assume $x \in \mathbb{R}^m$.

Metropolis

Target $r(x)/Z$. Symmetric proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

[Q3] Does each step preserve $r(x)/Z$ as invariant?

~~Hastings~~ Metropolis algorithm

Target $r(x)/Z$. ~~Symmetric~~ proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

$$\frac{r(x') \times q(x_n|x')}{r(x_n) \times q(x'|x_n)}$$

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

Hastings Metropolis algorithm

Target $r(x)/Z$. ~~Symmetric~~ proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

   $$\frac{r(x') \times q(x_n|x')}{r(x_n) \times q(x'|x_n)}$$

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, \cancel{r(x')/r(x_n)})$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

[Q1] Develop q for the King Markov puzzle.

Metropolis Hastings algorithm

Target $r(x)/Z$. ~~Symmetric~~ proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

$$\frac{r(x') \times q(x_n|x')}{r(x_n) \times q(x'|x_n)}$$

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

[Q2] Noisy greedy exploration. Find a (relative) obj.

Metropolis ~~Hastings~~ algorithm

Target $r(x)/Z$. ~~Symmetric~~ proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

$$\frac{r(x') \times q(x_n|x')}{r(x_n) \times q(x'|x_n)}$$

    a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, r(x')/r(x_n))$

    b) $u \sim \text{uniform}(0,1)$

    c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

[Q2] Noisy greedy exploration. Find a (relative) obj.

Hastings
Metropolis algorithm

Target $r(x)/Z$. ~~Symmetric~~ proposal $q(x'|x)$.

1. initialise x_1 randomly; $n:=1$

2. repeat:

$$\frac{r(x') \times q(x_n|x')}{r(x_n) \times q(x'|x_n)}$$

   a) $x' \sim q(x'|x_n)$; $\alpha := \min(1, \frac{r(x')}{r(x_n)})$

   b) $u \sim \text{uniform}(0,1)$

   c) $x_{n+1} := (\text{if } (u \leq \alpha) \text{ then } x' \text{ else } x_n)$; $n:=n+1$

[Q3] Does each step preserve $r(x)/Z$ as invariant?

Metropolis-Hastings algo.

- Generate samples from unnormalised r(x).

- Noisy greedy exploration using q(x'|x).

Recap of the MH algo.

- Generate samples from unnormalised r(x).

- Noisy greedy exploration using q(x'|x).

- Each step of MH has r(x)/Z as inv. dist.

Guarantees informally

[Thm1] Suppose $q(x|x') > 0$ iff $q(x'|x) > 0$. Then, each step of MH has r/Z as inv. dist.

Guarantees informally

MH samples: x_1, x_2, x_3, …, x_n, …

[Thm2] For all $f:X \rightarrow \mathbb{R}$ with $\mathbb{E}_{r(x)/Z}[f(x)]$ defined,

$\sum_{i \leq n} f(x_i)/n \longrightarrow \mathbb{E}_{r(x)/Z}[f(x)]$ as $n \longrightarrow \infty$ with prob. 1,

if the MH with q is r/Z-irreducible.

Guarantees informally

The estimate converges to the right value.

MH samples: x_1, x_2, x_3, ..., x_n, ...

[Thm2] For all $f : X \to \mathbb{R}$ with $\mathbb{E}_{r(x)/Z}[f(x)]$ defined,

$$\sum_{i \leq n} f(x_i)/n \longrightarrow \mathbb{E}_{r(x)/Z}[f(x)] \text{ as } n \longrightarrow \infty \text{ with prob. 1,}$$

if the MH with q is r/Z-irreducible.

Guarantees informally

The estimate converges to the right value.

MH samples: x_1, x_2, x_3, …, x_n, …

[Thm2] For all $f: X \rightarrow \mathbb{R}$ with $\mathbb{E}_{r(x)/Z}[f(x)]$ defined,

$\sum_{i \leq n} f(x_i)/n \longrightarrow \mathbb{E}_{r(x)/Z}[f(x)]$ as $n \longrightarrow \infty$ with prob. 1,

if the MH with q is r/Z-irreducible.

MH moves well. For any r/Z-legal x, x', the MH can go from x to x' with non-zero prob.

Guarantees informally

The estimate converges to the right value.

MH samples: x_1, x_2, x_3, ..., x_n, ...

[Thm2] For all $f : X \rightarrow \mathbb{R}$ with $\mathbb{E}_{r(x)/Z}[f(x)]$ defined,

$\sum_{i \leq n} f(x_i)/n \longrightarrow \mathbb{E}_{r(x)/Z}[f(x)]$ as $n \longrightarrow \infty$ with prob. 1,

if the MH with q is r/Z-irreducible.

MH moves well. For any r/Z-legal x, x', the MH can go from x to x' with non-zero prob.

Guarantees informally

MH samples: x_1, x_2, x_3, ..., x_n, ...

[Thm2] For all $f : X \to \mathbb{R}$ with $\mathbb{E}_{r(x)/Z}[f(x)]$ defined,

$$\sum_{i \leq n} f(x_i)/n \longrightarrow \mathbb{E}_{r(x)/Z}[f(x)] \text{ as } n \longrightarrow \infty \text{ with prob. } 1,$$

if the MH with q is r/Z-irreducible.

Consequence of a general result in ergodic theory. Thm1 plays a crucial role in the proof.

Reference

I looked at Chapters 5 and 6 of Robert & Casella's "Monte Carlo Statistical Methods".

Not recommended for general reading.

But details and pointers can be found there.