Seminarium magisterskie 2

Metody Monte Carlo z wykorzystaniem łańcuchów Markowa

Magdalena Trafidło

2 czerwca 2023

Opis problemu

- Problem: Chcielibyśmy próbkować z rozkładu π, którego gęstość jest skomplikowanej postaci.
- Propozycja: Metody Monte Carlo + łańcuchy Markowa.

Łańcuchy Markowa z czasem dyskretnym

- $E = \{e_1, \ldots, e_m\}$ - przestrzeń stanów,
- $\{X_n\}$ - łańcuch Markowa, jeśli:

$$\mathbb{P}(X_n = e_j | X_{n-1} = e_i, X_{n-2} = e_{i_{n-2}}, \ldots, X_0 = e_{i_0}) = \\ = \mathbb{P}(X_n = e_j | X_{n-1} = e_i) = P_{ij},$$

- $\mathbb{P} = (P_{ij})$ - macierz przejść, stochastyczna
- π - rozkład stacjonarny, jeśli $\pi\mathbb{P} = \pi$,
- $\{X_n\}$ - ergodyczny, jeśli $\{X_n\}$ - nieredukowalny i nieokresowy.

Łańcuchy Markowa c.d.

- $\{X_n\}$ - ergodyczny $\implies$ $\{X_n\}$ ma dokładnie jeden rozkład stacjonarny, dla dowolnych i, j $\lim_{t \to \infty} P_{ij}^t = \pi(j)$,
- $\{X_n\}$ - odwracalny: istnieje takie π, że dla dowolnych i, j $\pi(i)P_{ij} = \pi(j)P_{ji}$.

Markov Chain Monte Carlo (MCMC)

- Problem: Chcielibyśmy próbkować z rozkładu π, którego gęstość jest skomplikowanej postaci.
- Rozwiązanie: Konstruujemy łańcuch Markowa, którego rozkład stacjonarny to π i z niego próbkujemy.

Hard-core model

$G = (V, K)$ - graf, gdzie:

- V - zbiór wierzchołków.
- K - zbiór krawędzi.

Konfiguracja: Przypisanie każdemu z wierzchołków 0 albo 1
Konfiguracja dopuszczalna: żadne dwa wierzchołki z przypisaną 1 nie mogą być połączone ze sobą krawędzią:

$$\forall v, w \in V : (v, w) \in K \;\; e(v)e(w) = 0.$$

Niech Z_G - liczba wszystkich dopuszczalnych konfiguracji,
$e \in S = \{0, 1\}^V$,

$$\pi(e) = \begin{cases} 1/Z_G, e\text{- dopuszczalna} \\ 0, \;\text{w.p.p.} \end{cases}$$

Hard-core model

Algorytm:

1. Niech X_k będzie w jednej z dopuszczalnych konfiguracji e.
2. Wybieramy losowo wierzchołek v, z jednostajnym prawdopodobieństwem na V.
3. Generujemy $Z \sim \pi$:
    1. Dla $w \in V, w \neq v$ $Z(w) = e(w)$ $(Z_{-v} = e_{-v})$.
    2. Rzucamy monetą. Jeśli dostaniemy reszkę, i wszystkie wierzchołki sąsiadujące z v mają przypisaną wartość 0, zmieniamy wartość przy wierzchołku v na 1 $(Z(v) = 1)$. W przeciwnym wypadku $Z(v) = 0$.
4. $X_{k+1} = Z$.

Uogólnienie - próbkowanie Gibbsa

1. Niech X_k będzie w jednej z dopuszczalnych konfiguracji e.
2. Wybieramy losowo wierzchołek v, z jednostajnym prawdopodobieństwem na V.
3. Generujemy $Z \sim \pi$:
    1. Dla $w \in V, w \neq v$ $Z(w) = e(w)$ $(Z_{-v} = e_{-v})$.
    2. Przypisujemy wartość do $Z(v)$ zgodnie z warunkowym rozkładem stacjonarnym względem pozostałych wierzchołków:

    $$\mathbb{P}(Z_v = s | Z_{-v} = e_{-v}), s \in S.$$

4. $X_{k+1} = Z$.

Przykład - kolorowanie grafu

$G = (V, K)$ - graf, $S = \{1, \ldots, q\}$ - kolory.
Kolorujemy każdy z wierzchołków jednym z q kolorów.
Dopuszczalne kolorowanie: dwa sąsiadujące ze sobą wierzchołki nie mogą mieć tego samego koloru:

$$\forall v, w \in V : (v, w) \in K \ \ e(v) \neq e(w).$$

Przykład - Problem plecakowy

Mamy d ponumerowanych przedmiotów. Niech

- $w = (w_1, \dots, w_d)$, $w_i \geqslant 0$ - wagi przedmiotów,
- $v = (v_1, \dots, v_d)$, $v_i \geqslant 0$ - wartości przedmiotów,
- W - maksymalne obciążenie plecaka,
- $x = (x_1, \dots, x_d)$ - konfiguracja, $x_i \in \{0, 1\}$, gdzie $x_i = 1$ oznacza, że bierzemy i-ty przedmiot.

Konfiguracja jest dopuszczalna, jeśli $\sum_{i=1}^{d} x_i w_i \leqslant W$.
Nasz cel:
znalezienie x maksymalizującego $\sum_{i=1}^{d} x_i v_i$, gdzie $\sum_{i=1}^{d} x_i w_i \leqslant W$.

Przykład - Problem plecakowy

$$\pi(x) = \frac{1}{c} \exp\left(\beta \sum_{i=1}^{d} v_i x_i\right), \text{ dla } x : \sum_{i=1}^{d} x_i w_i \leqslant W$$

1. $X_k = x = (x_1, \ldots, x_d), x_i \in \{0, 1\}$, x-dopuszczalna.
2. Wybieramy $x_i \in x$ z jednostajnym prawdopodobieństwem na x.
3. Generujemy $Z \sim \pi$:
   1. $Z_{-i} = x_{-i}$,
   2. $Z_i = 0$ z pr. $\frac{1}{1+\exp(\beta v_i)}$, $Z_i = 1$ z pr. $\frac{1}{1+\exp(-\beta v_i)}$.
4. $X_{k+1} = Z$.

Generujemy nowe X, i zapamiętujemy ten, który miał największą wartość.

Przykład - Problem plecakowy

$$\mathbb{P}(Z_i = 0 | Z_{-i} = x_{-i}) = \frac{\mathbb{P}(Z_i = 0, Z_{-i} = x_{-i})}{\mathbb{P}(Z_i = 0, Z_{-i} = x_{-i}) + \mathbb{P}(Z_i = 1, Z_{-i} = x_{-i})} =$$

$$\frac{\exp\left(\beta\left(\sum_{j=1, j\neq i}^{d} v_j x_j + v_i \cdot 0\right)\right)}{\exp\left(\beta\sum_{j=1, j\neq i}^{d} v_j x_j + 0\cdot\beta v_i\right) + \exp\left(\beta\sum_{j=1, j\neq i}^{d} v_j x_j + 1\cdot\beta v_i\right)} =$$

$$\frac{\exp\left(\beta\sum_{j=1, j\neq i}^{d} v_i x_i\right)}{\exp\left(\beta\sum_{j=1, j\neq i}^{d} v_i x_i\right) + \exp\left(\beta\sum_{j=1, j\neq i}^{d} v_i x_i + \beta v_i\right)} = \frac{1}{1+\exp\left(\beta v_i\right)},$$

$$\mathbb{P}(Z_i = 1 | Z_{-i} = x_{-i}) = 1 - \frac{1}{1+\exp\left(\beta v_i\right)} = \frac{1}{1+\exp\left(-\beta v_i\right)}.$$

Przykład - Model Isinga

$G = (V, K)$ - graf, gdzie:

- V - zbiór wierzchołków.
- K - zbiór krawędzi.

Konfiguracja: Przypisanie każdemu z wierzchołków tzw spina $+1$ albo -1

Energia konfiguracji: $H(e) = \sum_{(v,w)\in K} e(v)e(w)$

$$\pi(e) = \frac{1}{c} \exp\left(\beta \sum_{(v,w)\in K} e(v)e(w)\right)$$

Przykład - Model Isinga

Niech $I = \{(u, w) \in K, u, w \neq v\}, J = \{w : (v, w) \in K\}$.

$$\mathbb{P}(Z(v) = 1 | Z_{-v} = e_{-v}) =$$
$$\frac{\mathbb{P}(Z(v) = 1, Z_{-v} = e_{-v})}{\mathbb{P}(Z(v) = 1, Z_{-v} = e_{-v}) + \mathbb{P}(Z(v) = -1, Z_{-v} = e_{-v})} =$$
$$\frac{\exp\left(\beta[\sum_I e(u)e(w) + \sum_J 1 \cdot e(w)]\right)}{\exp\left(\beta[\sum_I e(u)e(w) + \sum_J e(w)]\right) + \exp\left(\beta[\sum_I e(u)e(w) - \sum_J e(w)]\right)} =$$
$$\frac{1}{1 + exp(-2\beta \sum_{w:(v,w)\in K} e(w))},$$

$$\mathbb{P}(Z(v) = -1 | Z_{-v} = e_{-v}) = \frac{1}{1 + exp(2\beta \sum_{w:(v,w)\in K} e(w))}$$

Algorytm Metropolisa

Niech

- $E = \{e_1, \ldots, e_M\}$ - przestrzeń stanów, konfiguracje,
- π - rozkład stacjonarny.

Chcemy generować z rozkładu π, ale nie znamy macierzy $\mathbb{P}$, która spełniałaby $\pi\mathbb{P} = \pi$.
Do znalezienia $\mathbb{P}$ możemy użyć tzw rozkładu proponującego, opisanego macierzą przejść Q na przestrzeni stanów E, nieredukowalnego i nieokresowego łańcucha Markowa:

$$\mathbb{P}(e, e') = \begin{cases} Q(e, e') \min\left(1, \frac{\pi(e')}{\pi(e)}\right), e' \neq e, \\ 1 - \sum_{e'' \in E, e'' \neq e} Q(e, e'') \min\left(1, \frac{\pi(e'')}{\pi(e)}\right), e' = e. \end{cases}$$

Algorytm Metropolisa

Q - symetryczna macierz przejść,
Algorytm:

1. Niech X_k będzie w jednej z dopuszczalnych konfiguracji e.
2. Generujemy $Z \sim (Q(e, e_1), Q(e, e_2), \ldots, Q(e, e_M))$ (lub w skrócie: $Z \sim Q(e, \cdot)$), powiedzmy że $Z = e'$.
3. Niech $\alpha = \min(1, \frac{\pi(e')}{\pi(e)})$.
4. Generujemy $U \sim U(0, 1)$. Jeśli $U < \alpha$, to $X_{k+1} = e'$, w przeciwnym przypadku $X_{k+1} = X_k$.

Algorytm Metropolisa-Hastingsa

Q - niesymetryczna macierz przejść,
Algorytm:

1. Niech X_k będzie w jednej z dopuszczalnych konfiguracji e.
2. Generujemy $Z \sim (Q(e, e_1), Q(e, e_2), \ldots, Q(e, e_M))$ (lub w skrócie: $Z \sim Q(e, \cdot)$), powiedzmy że $Z = e'$.
3. Niech $\alpha = \min\left(1, \frac{Q(e', e)\pi(e')}{Q(e, e')\pi(e)}\right)$.
4. Generujemy $U \sim U(0, 1)$. Jeśli $U < \alpha$, to $X_{k+1} = e'$, w przeciwnym przypadku $X_{k+1} = X_k$.

Problem plecakowy - algorytm Metropolisa

- $x = (x_1, \dots, x_d) \in \{0,1\}^d$,
- $\pi(x) = \frac{1}{c} \exp\left(\beta \sum_{i=1}^{d} v_i x_i\right)$, dla $x : \sum_{i=1}^{d} x_i w_i \leqslant W$.

Niech

$$Q(x, x') = \begin{cases} \frac{1}{d}, & \sum_i |x_i - x_i'| = 1 \\ 0 \text{ w.p.p} \end{cases}$$

Algorytm:

1. Niech $X_k = x, x : \sum_{i=1}^{d} x_i w_i$.
2. Generujemy $Z \sim Q(x, \cdot)$, powiedzmy że $Z = x'$.
3. Niech $\alpha = \min(1, \frac{\pi(x')}{\pi(x)})$.
4. Generujemy $U \sim U(0,1)$. Jeśli $U < \alpha$, to $X_{k+1} = x'$, w przeciwnym przypadku $X_{k+1} = X_k$.

Problem plecakowy - algorytm Metropolisa

$$\frac{Q(x',x)\pi(x')}{Q(x,x')\pi(x)} = \frac{\exp\left(\beta\sum_{i=1}^{d} v_i x_i'\right)}{\exp\left(\beta\sum_{i=1}^{d} v_i x_i\right)} = \frac{\exp\left(\beta\sum\limits_{i=1,i\neq j}^{d} v_i x_i' + \beta v_j x_j'\right)}{\exp\left(\beta\sum\limits_{i=1,i\neq j}^{d} v_i x_i + \beta v_j x_j\right)} =$$

$$\frac{\exp\left(\beta\sum\limits_{i=1,i\neq j}^{d} v_i x_i + \beta v_j x_j'\right)}{\exp\left(\beta\sum\limits_{i=1,i\neq j}^{d} v_i x_i + \beta v_j x_j\right)} = \frac{\exp\left(\beta v_j x_j'\right)}{\exp\left(\beta v_j x_j\right)} = \frac{\exp\left(\beta v_j(1-x_j)\right)}{\exp\left(\beta v_j x_j\right)} =$$

$$\exp(-\beta v_j(2x_j-1))$$

Miara monotoniczna

Miara probabilistyczna π jest monotoniczna na $E = S^V$ jeśli $\forall e \leqslant e', \forall v \in V : \pi(e_{-v}) > 0, \pi(e'_{-v}) > 0$ mamy:

$$\mathbb{P}(z(v) \leqslant s | z_{-v} = e_{-v}) \geqslant \mathbb{P}(z(v) \leqslant s | z_{-v} = e'_{-v}) \ \forall s \in S$$

natomiast jest antymonotoniczna jeśli zachodzi:

$$\mathbb{P}(z(v) \leqslant s | z_{-v} = e_{-v}) \leqslant \mathbb{P}(z(v) \leqslant s | z_{-v} = e'_{-v}) \ \forall s \in S.$$

Funkcja aktualizująca monotoniczna

Mówimy, że funkcja aktualizująca φ jest monotoniczna względem $\leqslant$ jeśli:

$$\forall i, j \in E : (i \leqslant j) \; \forall U \in [0,1] \; \varphi(i,u) \leqslant \varphi(j,u).$$

Coupling from the past(CFTP)

1. Niech n=1.
2. Startujemy dwa łańcuchy w chwili $-N_n$, jeden w minimalnym stanie (0), drugi w maksymalnym stanie (M), do chwili 0, używając takiej samej funkcji aktualizującej φ, i używając takich samych $U_{-N-1}, \dots, U_1, U_0 \sim U(0,1)$.
3. Jeśli w chwili 0 oba łańcuchy znajdą się w tym samym stanie e, zwracamy e i kończymy. W przeciwnym przypadku: $n = n + 1$, idziemy do kroku 2 i używamy poprzednio wygenerowanych U_i dla nowego n.

Źródła

- Andrzej Ruciński, Wykład 9: Markov Chain Monte Carlo
- Maciej Romaniuk, Metody Monte-Carlo i MCMC
- Bartosz Chmiela, Locally-informed proposals in Metropolis-Hastings algorithm with applications