Vybrané kapitoly zo štatistickej fyziky
Príklady z cvičenia

cviko bolo 16.4.2024 *numerické metódy*

Akékoľvek otázky smelo smerujte na `juraj(a)tekel(b)gmail(c)com`

Metropolisov algortimus

Príklad 1 (Metropolisov algoritmus ako generátor náhodných čísel)**.** Majme náhodnú premennú x, ktorá je rozdelená podľa rozdelenia pravdepodobnosti $p(x)$. Zostrojme nasledujúcu postupnosť hodnôt náhodnej premennej $\{x_i\}, i = 1, \dots, N$. Začnime s nejakou hodnotou x_0, na základe algoritmu ktorý bude špecifikovaný neskôr zvoľme testovaciu hodnotu x_t a urobme pomer

$$\alpha = \frac{p(x_t)}{p(x_0)} \ . \tag{1}$$

Potom vyberieme náhdoné číslo q z intervalu $(0, 1)$ a ak $\alpha \geq q$, za x_1 zvolíme x_t, ak $\alpha < q$ za x_1 zvolíme x_0. Rovnakú postup opakujeme postupne pre ďalšie x_i. Dá sa ukázať, že rozdelenie takomto súbore hodnôt $\{x_i\}$ konverguje k $p(x)$, t.j. že vybrať náhodne z tohto súboru je to isté ako vybrať náhodne zo všetkých možných x podľa rozdelenia pravdepodobnosti $p(x)$.

a. Rozmyslite si, že pre generovanie postupnosti nepotrebujeme poznať normalizáciu $p(x)$. Prečo je to dôležité?

b. Rozmyslite si, že pre x stav v kanonickom súbore a $p(x)$ príslušné rozdelenie pravdepodobnosti takto dostaneme algoritmus z prednášky.

c. Za náhodnú premennú zvoľme celé čísla $\mathbb{Z}$ a $x_0 = 0$. Za algoritmus generovania x_t zoberme náhodný krok veľkosti 1 vpravo alebo vľavo. Takýto algoritmus vygeneruje pre $p(x) = e^{-x^2/2}$ postupnosť celých čísel rozdelených podľa Gaussovho rozdelenia. Overte to počítačovým programom. Skúste, ako váš výsledok závisí od voľby N a ako to vyzerá, keď budete čísla x voliť na číselnej osi hustejšie.

d. Naprogramujte aj dvojrozmerný prípad, pre $x_1, x_2 \in \mathbb{R}^2$. Algoritmus generovania testovacieho bodu skúste ako

- krok náhodným smerom veľkosti 1,
- krok veľkosti 1 náhodne doprava alebo doľava a potom krok veľkosti 1 náhodne hore alebo dole.

Porovnajte výsledky, fungujú oba spôsoby generovania rovnako dobre?

e. Vyskúšajte rôzne iné rozdelenia $p(x)$, napríklad aj také, kedy neviete nájsť generátor metódou inverzná k primitívnej. Kedy funguje Metropolis lepšie, ako táto metóda?

Treba upozorniť, že táto implementácia je trochu naivná a program môže mať niekoľko problémov, ktoré potom treba riešiť sofistikovanejším postupom.

Príklad 2 (Metropolisov algoritmus ako generátor náhodných čísel 2)**.** Iný postup, spomínaný v nasledujúcom príklade, je o čosi iný ako vyššie prezentovaný. Tam sa zoberie ako hodnota náhodnej premennej posledná vygenerovaná premenná x_N a ďalšia hodnota sa potom generuje novou postupnosťou. Zvoľte N_1, N_2 a upravte váš program tak, aby N_2-krát vygeneroval postupnosť N_1 náhodných čísel a za výsledný súbor zoberal iba posledné vygenerované premenné. Porovnajte z výsledkom z predchádzajúcej úlohy pre $N = N_2$.

Príklad 3 (Metropolis pre dvojhladinový systém)**.** Majme dvojhladinový systém s energiami $E_1 < E_2$.

a. Rozmyslite si, že pri výbere testovacieho stavu nie je moc čo riešiť. Nájdite transfer maticu pre tento systém.

b. Nájdite jej vlastné čísla a vlastné vektory a nájdite rozklad všeobecného rozdelenia na dvoch stavoch do týchto vektorov.

c. Ako vyzerá rozdelenie pravdepodobnosti po n Metropolisovských krokoch a ako vyzerá v limite $n \to \infty$?

Príklad 4 (■ Metropolis pre trojhladinový systém)**.** Majme trojhladinový systém s energiami $-E, 0, E$. Stavy budeme číslovať ich energiou.

a. Rozmyslite si, že pri výbere testovacieho stavu už je čo riešiť. Skúmajme dve možnosti.

   - Zo stavu 0 vieme prejsť do oboch stavov $\pm E$, zo stavu $\pm E$ vieme prejsť len do stavu 0.
   - Z každého stavu sa vieme dostať do každého iného.

   Nájdite transfer matice v oboch prípadoch.

b. Nájdite ich vlastné čísla a vlastné vektory a nájdite rozklad všeobecného rozdelenia na troch stavoch do týchto vektorov.

c. Ako vyzerá rozdelenie pravdepodobnosti po n Metropolisovských krokoch a ako vyzerá v limite $n \to \infty$? Ktorá z metód výberu testovacieho stavu vedie na správne rozdelenie? Prečo?

Návod. Skúste si napísať maticu výberu testovacieho stavu v oboch prípadoch a uvidieť v nich rozdiel. Ako vyzerá matica výberu testovacieho stavu v predchádzajúcej úlohe?

Príklad 5. Oba predchádzajúce príklady implementujte. Napíšte program, ktorý vygeneruje postupnosť stavov a zistite, aké je vo výsledných stavoch rozdelenie energie.