Algorytmy Numeryczne – Zadanie 2
Niezdecydowany wędrowiec

14 marca 2024

1 Wersja uproszczona

Niezdecydowany wędrowiec spacerując parkową aleją położoną w linii NS porusza się jeden krok na północ z prawdopodobieństwem 1/2 i z prawdopodobieństwem 1/2 jeden krok na południe. Proszę przyjąć, że wędrowiec robi równe kroki w obu kierunkach a kolejne losowania są niezależne.

W odległości n kroków na północ od początkowego położenia (start) wędrowca znajduje się odkryta studzienka kanalizacyjna (OSK) do której niezdecydowany wędrowiec z pewnością wpadnie, złamie nogę i tak skończy się ten spacer jeśli tylko do niej dotrze.

W odległości s kroków na południe od początkowego położenia wędrowca znajduje się wyjście z parku. Po dojściu do wyjścia wędrowiec przestaje być niezdecydowany i dalej pewnym krokiem bezpiecznie wraca do domu.

Interesuje nas prawdopodobieństwo, że wędrowiec bezpiecznie wróci do domu.

Przykład

Niech $n = 2$ i $s = 3$, a więc OSK jest 5 kroków od wejścia, a wędrowiec stoi 3 kroki od wejścia.

Rozwiązanie przy pomocy układu równań liniowych

Nazwijmy miejsca, w których może stać wędrowiec: m_0 (wyjście), m_1, m_2, m_3 (start), m_4 i m_5 (OSK). Niech p_i oznacza prawdopodobieństwo bezpiecznego powrotu do domu, gdy wędrowiec stoi w punkcie m_i. Zauważmy, że nie jest istotne, czy wędrowiec zrobił krok na północ i po chwili na południe by stanąć w tym samym miejscu. Losowania są niezależne, proces jest bezpamięciowy, sytuacja w parku się nie zmienia z kolejnymi krokami wędrowca i prawdopodobieństwo zależy tylko od tego, gdzie się on aktualnie znajduje. Możemy więc zapisać następujące równania:

$$p_0 = 1,$$

gdyż wędrowiec od wyjścia wraca bezpiecznie do domu

$$p_5 = 0,$$

gdyż wędrowiec wpada w studzienkę

$$p_1 = 0.5 \cdot p_0 + 0.5 \cdot p_2,$$

gdyż z równym prawdopodobieństwem wędrowiec idzie w stronę wyjścia lub w stronę OSK. Podobne równania możemy zapisać dla innych miejsc, w postaci macierzowej otrzymamy następujący układ równań:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -0.5 & 1 & -0.5 & 0 & 0 & 0 \\ 0 & -0.5 & 1 & -0.5 & 0 & 0 \\ 0 & 0 & -0.5 & 1 & 0 & 0 \\ 0 & 0 & 0 & -0.5 & 1 & -0.5 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Rozwiązaniem tego układu jest wektor prawdopodobieństw P, który dla przypadku powyżej wynosi:

$$P = \begin{bmatrix} 1 \\ 0.8 \\ 0.6 \\ 0.4 \\ 0.2 \\ 0 \end{bmatrix}$$

A więc obliczyliśmy, że dla wskazanego w przykładzie punktu startowego wędrowiec wraca bezpiecznie do domu z prawdopodobieństwem 0.4. Zauważmy że, niejako przy okazji, obliczyliśmy prawdopodobieństwa dla wszystkich możliwych punktów startowych.

Metoda Monte Carlo

Stosując metodę Monte Carlo możemy zweryfikować, czy wyniki naszych obliczeń znajdą potwierdzenie w symulacji.

Postąpimy w następująco:

- ustawiamy wędrowca w miejscu startowym
- prowadzimy symulację wędrówki aż wędrowiec trafi w OSK lub do wyjścia
- powyższe powtarzamy N razy zliczając i – ile razy trafiliśmy do wyjścia
- za oszacowanie prawdopodobieństwa powrotu do domu bierzemy iloraz i/N

2 Wersja Podstawowa

Park składa się z m alejek, które krzyżują się w punktach v_1, v_2, …, v_n. Początek i koniec alejki uważamy za skrzyżowanie. Jest co najwyżej jedna alejka łącząca v_i z v_j o długości d_{ij} kroków, $d_{ij} \in \mathbb{N}^+$.

Niezdecydowany wędrowiec spacerując parkową aleją porusza się podobnie jak w wersji uproszczonej z prawdopodobieństwem 1/2 w jedną lub drugą stronę. Jeśli trafi na skrzyżowanie, to wybiera z równym prawdopodobieństwem jedną z alejek (łącznie z tą z której trafił na skrzyżowanie).

Jedno ze skrzyżowań jest punktem startowym, jedno ze skrzyżowań jest wyjściem. W jednym ze skrzyżowań znajduje się OSK (ale nie w wyjściu i nie w punkcie startowym).

Podobnie jak w wersji uproszczonej interesuje nas prawdopodobieństwo, że wędrowiec bezpiecznie wróci do domu.

Rozszerzenia wersji podstawowej

R0: Może się zdarzyć, że w parku jest więcej niż jedno wyjście (+5%).

R1: Może się zdarzyć, że w parku jest więcej niż jedna OSK (+5%).

R2: W parku może być dwóch wędrowców, pytamy czy obaj wrócą do domu. Jeśli w trakcie wędrówki zdarzy się, że wędrowcy się miną w tej samej alejce, staną w tym samym miejscu alejki lub na tym samym skrzyżowaniu, to dochodzi między nimi do awantury i incydentu, który nie kończy się bezpiecznym powrotem do domu (+5%).

R3: Może się zdarzyć, że kilka alejek, być może różnej długości, łączy te same dwa skrzyżowania (+5%).

R4: Przy wybranych skrzyżowaniach stoją pojemniki na śmieci. Wędrowiec nie lubi śmietników i prawdopodobieństwo, że zrobi krok do miejsca, gdzie pojemnik się znajduje jest dwa razy mniejsze niż w stronę, gdzie pojemnika nie ma. Należy uwzględnić przypadek bardzo krótkich alejek, np. o długości jednego lub dwóch kroków (+5%).

Format danych wejściowych

W pierwszej linii dwie liczby n i m oznaczające liczbę skrzyżowań i liczbę alejek. Następnie w kolejnych m liniach po trzy liczby: i, j, d_{ij} opisujące jedną alejkę, odpowiednio numery skrzyżowań do których prowadzi i jej długość.

Następnie w osobnych liniach dane o OSK, wyjściach, punktach startowych wędrowców i śmietnikach (zobacz przykłady).

Przykład

```
4 5
1 2 4
2 3 4
3 4 4
4 1 4
1 3 6

1 1
2 2 4
1 3
0
```

W powyższym przykładzie mamy 5 alejek krzyżujących się w 4 miejscach. Alejka łącząca 1 i 3 ma 6 kroków a pozostałe po 4. Jest jedna OSK na skrzyżowaniu 1, dwa wyjścia w skrzyżowaniach 2 i 4 oraz jednego wędrowca startującego w 3. Nie ma śmietników.

Przykład II

```
5 4
1 2 2
```

```
1 3 2
1 4 2
1 5 2

1 1
2 2
1 3
1 4 5
```

W powyższym przykładzie mamy 4 alejki krzyżujące się w 5 miejscach. Od skrzyżowania oznaczonego 1 prowadzą cztery alejki do pozostałych miejsc i wszystkie mają po 2 kroki długości. Jest jedna OSK na skrzyżowaniu 1, jedno wyjście w miejscu 2 oraz jednego wędrowca startującego w 3. Są dwa śmietniki znajdujące się w miejscach 4 i 5.

3 Zadanie

Proszę napisać program, który buduje i rozwiązuje układ równań dla wybranego wariantu zadania (wersja podstawowa i wybrane rozszerzenia). Poprawność rozwiązania układu równań proszę sprawdzić podstawiając otrzymane rozwiązanie. Poprawność budowania układu równań proszę sprawdzić metodą Monte Carlo.

Struktury danych

W przypadku tego zadania okaże się, że wiele elementów macierzy układu to zera, a stosunkowo niewiele elementów jest od zera różnych. Takie macierze nazywamy rzadkimi (ang. sparse). Dla macierzy rzadkich mamy zarówno możliwość optymalizacji użycia pamięci jak i czasu działania algorytmów.

W celu oszczędności pamięci elementy macierzy mogą być zapamiętane tylko wtedy, gdy są różne od zera. Można to zrobić na wiele sposobów, na potrzeby tego zadania można przyjąć:

- DS1 – słownik (ang. map), w którym kluczem jest para (r, c) opisująca numer wiersza i kolumny a wartością jest wartość elementu
- DS2 – pojemnik (np. lista lub tablica), w którym każdy element opisuje jeden wiersz macierzy. Przy czym wiersz macierzy może być kolejnym pojemnikiem
- DS3 – pojemnik (np. lista lub tablica), w którym każdy element opisuje jedną kolumnę macierzy. Przy czym kolumna macierzy może być kolejnym pojemnikiem
- DS4 – inne wybrane, w tym połączenia metod powyższych.

Algorytmy

Zaimplementuj:

- A1 – algorytm eliminacji Gaussa bez wyboru elementu podstawowego,
- A2 – algorytm eliminacji Gaussa z częściowym wyborem elementu podstawowego,
- A3 – metodę iteracyjną Gaussa-Seidela.

Hipotezy

Zweryfikuj następujące hipotezy:

H1: Algorytm A2 zwykle daje dokładniejsze wyniki niż A1. Różnica w dokładności rośnie wraz z rozmiarem macierzy i liczbą niezerowych współczynników.

H2: Algorytm A3 działa dla postawionego zadania (jeśli nie działa, tzn. proces nie zawsze jest zbieżny do rozwiązania, to wskaż przykłady, gdy jest rozbieżny).

H3: Jeśli Algorytm A3 jest zbieżny do rozwiązania, to wyniki otrzymujemy istotnie szybciej niż dla A1 i A2.

Punktacja

Maksymalna punktacja dla wersji podstawowej wynosi 90%. Można zwiększyć maksymalną liczbę punktów implementując rozszerzenia.

Praca zespołowa

Zadanie można wykonać w zespole dwu lub trzy osobowym. W takim przypadku proszę dokładnie oznaczyć jaki był zakres pracy członków zespołu. W oddaniu projektu musi uczestniczyć cały zespół.

Wskazówki

- Zacznij od wersji uproszczonej
- Korzystaj z systemu kontroli wersji i z testów jednostkowych (ang unit tests).
- Wszystkie testy wykonuj w taki sposób, aby można było je łatwo odtworzyć i powtórzyć (wykonywane eksperymenty zachowaj w kodzie programu lub w skryptach uruchomieniowych).
- Dane do testów należy generować raz, w taki sposób aby porównywane operacje wykonywać na tych samych danych – jest to istotne zwłaszcza przy porównywaniu dokładności obliczeń.
- Można zerknąć na A Gentle Introduction to Sparse Matrices for Machine Learning – wprowadzenie w świat rzadkich macierzy przyjazne dla programistów Pythona – wraz z dalszymi odnośnikami do literatury.
- Proszę nie zwlekać do terminu oddania projektu

4 Dla zainteresowanych

Koncept niezdecydowanego wędrowca pochodzi wprost z błądzenia losowego (ang. random walk) w grafach. Błądzenie losowe ma związek z obwodami elektrycznymi, o czym można poczytać na przykład tutaj (https://math.dartmouth.edu/ doyle/docs/walks/walks.pdf) oraz z konstrukcją algorytmu PageRank, o czym można poczytać na przykład tutaj (https://www.math.cmu.edu/users/pmelsted/papers/pagerank.pdf). Zachęcam do lektury.