Paweł Piwowarczyk 1, 2, 5, 5

................................................................... ---------------------------------

(Imię i nazwisko) (A, B, C, D)

Parametry:

M = 2

N = 15

norma = 1 (zadanie 1)

norma = 0 (zadanie 2)

**Raport z Pracowni nr 2**

**Zadanie 1.**

1. Cel zadania

Celem zadania było zbadanie zbieżności algorytmu przybliżającego rozwiązanie macierzy metodą iteracji Seidela w zależności od rozmiaru macierzy dla normy kolumnowej.

1. Metody

W doświadczeniu wykorzystano kilka klas stworzonych w języku Python. Odpowiedni projekt stworzono w środowisku Visual Studio Code i kompilowano w zintegrowanym ze środowiskiem terminalu PowerShell na komputerze personalnym o procesorze AMD Ryzen 7 3700X.

1. Przyjęte parametry

N = 15

norma = 1 (kolumnowa)

k określone w trakcie doświadczenia

alfa określona w trakcie doświadczenia

zakres wartości n określony w trakcie doświadczenia

1. Przebieg doświadczenia i wyniki

Doświadczenie rozpoczęto od modyfikowania metody *testy()* w pliku *zadanie1.py*, tworząc w niej algorytm obliczający średnie wyniki iteracji Seidela dla macierzy o różnych rozmiarach i prezentujący je w formie wykresu, następnie kilkukrotnie edytowano parametr alfa, parametr definiujący ilość powtórzeń losowania i pomiaru nowych wartości macierzy tego samego rozmiaru dla uzyskania średniej i zakres N rozmiarów (n) macierzy, do momentu dobrania kombinacji wartości parametrów pozwalającej na wyraźne zaprezentowanie zbieżności metody na wykresie.

Następnie, na bazie kodu metody *testy()* i wniosków wyciągniętych z pracy nad jej utworzeniem i modyfikacją opracowano metodę *badaj\_zbieznosc*. Metoda, tak jak metoda *testy()*, wywoływana była w instancji klasy *Zadanie1* z pliku *zadanie1.py*.

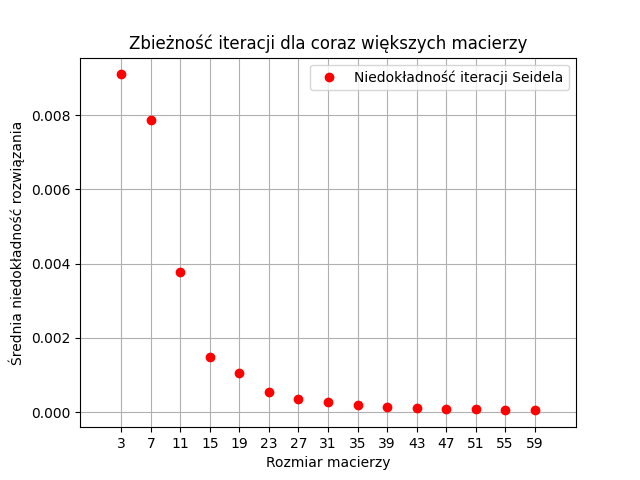
Na bazie wniosków wyciągniętych z opracowania metody testy() inicjalizując instancje klasy *Zadanie1* przyjęto parametry:

k = 5

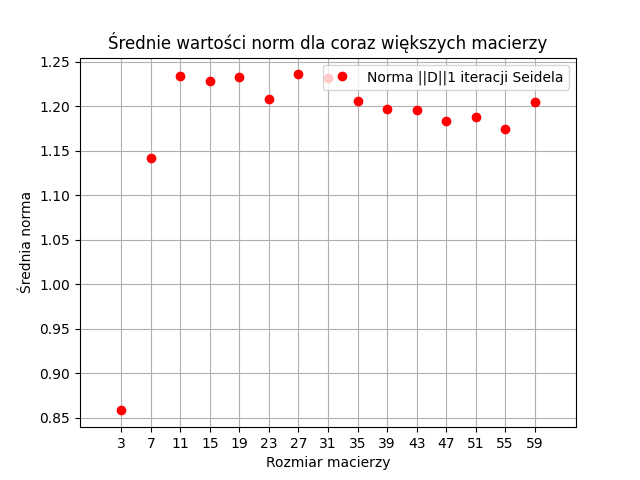
alfa = 0.01

Zakres wartości n = [3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59]

W wyniku wykonania kodu otrzymano wyniki, zilustrowane na poniżej zamieszczonych wykresach:



Wykres 1. Zależność dokładności rozwiązania po k = 5 iteracjach od rozmiaru macierzy n.



Wykres 2. Zależność wartości normy macierzy D po k = 5 iteracjach od rozmiaru macierzy n.

|  |  |  |
| --- | --- | --- |
| n | średnia norma | średnia niedokładność |
| 3 | 0.8583878246686126 | 0.009097933490743665 |
| 7 | 1.141947036684496 | 0.007878809607653766 |
| 11 | 1.233759540183447 | 0.0037862210009352486 |
| 15 | 1.2284336728306595 | 0.0014852075704223282 |
| 19 | 1.2331110793625684 | 0.0010538452035147423 |
| 23 | 1.2082224697528745 | 0.0005483423718025024 |
| 27 | 1.2360617211800005 | 0.0003628217796030831 |
| 31 | 1.2325242940359646 | 0.0002806636011253642 |
| 35 | 1.2061050889926992 | 0.00019939936823508729 |
| 39 | 1.196608582183106 | 0.00014425680641601157 |
| 43 | 1.1959711146946712 | 0.00010987646867420008 |
| 47 | 1.1839266215378257 | 7.66305800001385e-05 |
| 51 | 1.187971399598804 | 7.656068469862982e-05 |
| 55 | 1.1745199947024934 | 4.5701521178911777e-05 |
| 59 | 1.204621019017632 | 4.532362491181379e-05 |

Tabela 1. Uzyskane dokładne średnie wartości zawarte w wykresach 1 i 2.

1. Wnioski

W wyniku przeprowadzonego eksperymentu udało się wykazać, że metoda iteracji Seidela stosowana na macierzy jest zbieżna, i w miarę stosowania na macierzach coraz większych rozmiarów osiąga bliską zbieżność do dokładnego rozwiązania macierzy po wykonaniu coraz mniejszej ilości iteracji.

**Zadanie 2.**

1. Cel zadania

Celem zadania było zbadanie wpływu parametru epsilon na efektywność uzyskiwania rankingu stron Google PageRank za pomocą iteracji Seidela i metody potęgowej, wykorzystując metodę *iteruj\_roznica*.

1. Metody

W doświadczeniu wykorzystano kilka klas stworzonych w języku Python. Odpowiedni projekt stworzono w środowisku Visual Studio Code i kompilowano w zintegrowanym ze środowiskiem terminalu PowerShell na komputerze personalnym o procesorze AMD Ryzen 7 3700X.

1. Przyjęte parametry

N = 15

norma = 0 (wierszowa)

gamma, n, k, epsilony - określone w trakcie doświadczenia

1. Przebieg doświadczenia i wyniki

Doświadczenie rozpoczęto od modyfikowania metody *testy()* w pliku *zadanie2.py*, tworząc w niej algorytm obliczający liczbę iteracji wymaganych do uzyskania przybliżenia ostatecznego rankingu stron macierzy Google PageRank, przyjmując różny parametr epsilon dla iteracji z warunkiem stopu opartym sprawdzaniu, czy różnica niedokładności ostatnich dwóch przybliżeń jest mniejsza od parametru epsilon, następnie dobierając połączenie parametrów pozwalające na relatywnie szybkie zobrazowanie przejrzystych wyników.

W wyniku testowania wybrano następujące wartości parametrów:

- gamma = 0.1

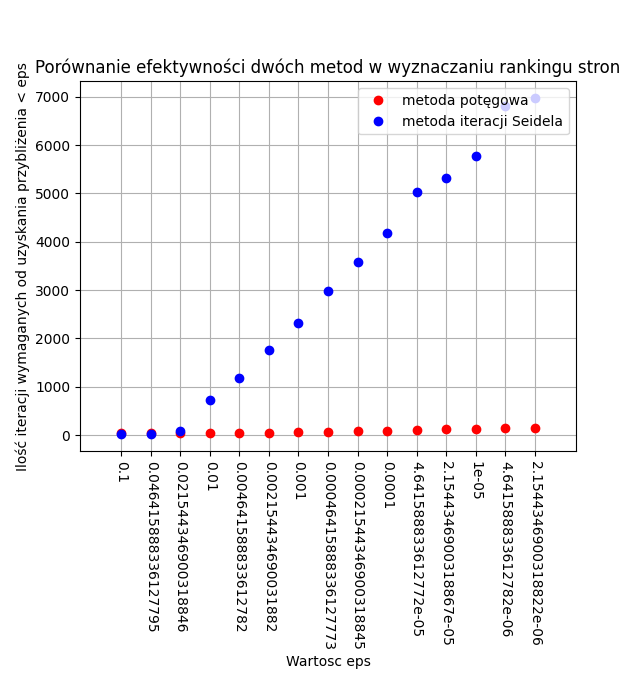
- n = 150

- k = 20

- wartości epsilonów = [0.1, 0.046415888336127795, 0.021544346900318846, 0.01, 0.004641588833612782, 0.002154434690031882, 0.001, 0.00046415888336127773, 0.00021544346900318845, 0.0001, 4.641588833612772e-05, 2.1544346900318867e-05, 1e-05, 4.641588833612782e-06, 2.1544346900318822e-06]

Następnie, na bazie kodu i wniosków wyciągniętych w ramach pracy nad metodą *testy()* opracowano metodę *badaj\_zbieznosc*. Metoda ta, wywoływana jako obiekt klasy Zadanie2 z pliku Zadanie2.py, ponad funkcjonalność opisaną w metodzie *testy()*, powtarzała pomiar k razy dla każdego badanego epsilona, notując średnie: ilość iteracji, liczbę linków na stronie i niedokładność rozwiązania.

W wyniku wykonania kodu otrzymano wyniki, zilustrowane na poniżej zamieszczonym wykresie oraz tabeli:



Wykres 3. Porównanie prędkości zbliżania się

|  |  |  |  |
| --- | --- | --- | --- |
|  | metoda potęgowa | | |
| epsilon | śr niedokładność | średnia ilość iteracji | średnia ilość linków |
| 0.1 | 0.11394133160461235 | 40.0 | 298.06666666666666 |
| 0.046415888336127795 | 0.11549677953594677 | 40.0 | 296.81333333333333 |
| 0.021544346900318846 | 0.11032044064176987 | 40.0 | 298.97999999999996 |
| 0.01 | 0.11107160733864596 | 40.0 | 299.46000000000004 |
| 0.004641588833612782 | 0.11772936152694553 | 40.0 | 301.41999999999996 |
| 0.002154434690031882 | 0.10040277981337498 | 43.0 | 299.64000000000004 |
| 0.001 | 0.03813749061157777 | 67.0 | 302.59999999999997 |
| 0.00046415888336127773 | 0.051863998992197236 | 60.0 | 299.6 |
| 0.00021544346900318845 | 0.024878851014482492 | 83.0 | 297.8066666666667 |
| 0.0001 | 0.014096359496009219 | 93.0 | 300.8066666666666 |
| 4.641588833612772e-05 | 0.008263260800653465 | 100.0 | 300.5466666666667 |
| 2.1544346900318867e-05 | 0.0007897970974183619 | 123.0 | 300.37333333333333 |
| 1,00E-05 | 0.0021163165484855582 | 125.0 | 300.88 |
| 4.641588833612782e-06 | 0.00011579748880057161 | 147.0 | 300.18 |
| 2.1544346900318822e-06 | 9.385077878463425e-05 | 153.0 | 301.4933333333333 |

Tabela2 . Ukazująca dokładne wartości parametrów wymienione w podpunkcie 4.

|  |  |  |  |
| --- | --- | --- | --- |
|  | metoda iteracji Seidela | | |
| epsilon | śr niedokładność | średnia ilość iteracji | średnia iośćlinków |
| 0.1 | 0.01758565535631092 | 20.0 | 300.58666666666664 |
| 0.046415888336127795 | 0.018032131777107992 | 21.0 | 296.5933333333333 |
| 0.021544346900318846 | 0.01791201625734044 | 96.0 | 298.43333333333334 |
| 0.01 | 0.00784916897968418 | 721.0 | 301.09999999999997 |
| 0.004641588833612782 | 0.003813574672941632 | 1189.0 | 299.9866666666667 |
| 0.002154434690031882 | 0.001785125341080802 | 1771.0 | 302.73999999999995 |
| 0.001 | 0.0008301878691885517 | 2314.0 | 300.0 |
| 0.00046415888336127773 | 0.0003773241827546254 | 2980.0 | 299.36 |
| 0.00021544346900318845 | 0.000176116178848317 | 3583.0 | 298.67999999999995 |
| 0.0001 | 8.506151047825948e-05 | 4190.0 | 299.2133333333333 |
| 4.641588833612772e-05 | 3.773591830660336e-05 | 5037.0 | 297.58666666666664 |
| 2.1544346900318867e-05 | 1.754510350436972e-05 | 5326.0 | 298.8333333333333 |
| 1,00E-05 | 8.145752235302027e-06 | 5782.0 | 300.14666666666665 |
| 4.641588833612782e-06 | 3.6626972223032657e-06 | 6814.0 | 299.0466666666666 |
| 2.1544346900318822e-06 | 1.8164208655776018e-06 | 6967.0 | 300.1066666666667 |

Tabela 2 c.d.

1. Wnioski

W wyniku przeprowadzonego eksperymentu udało się wykazać, że wraz ze zmniejszaniem się parametru epsilon, stosując metodę iteracji Seidela będzie ona stopniowo wymagała znacznie większej ilości iteracji do osiągnięcia różnicowego warunku stop iteracji.