

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ     
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего профессионального образования

**ДАЛЬНЕВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ**

**Институт математики и компьютерных технологий**

**Департамент математического и компьютерного моделирования**

**ОТЧЕТ**

**по лабораторной работе** **№ 7**

«Метод вращений для нахождения собственных значений матрицы»

Выполнил: студент гр. Б9120-02.03.01сцт

Пограничный К. О.

Проверил: преподаватель

Кузнецов К. С.

Владивосток 2023

**Постановка задачи**

Дана симметричная положительно определенная матрица А размерности n. Требуется найти её собственные значения методом вращений и провести эксперимент, чтобы узнать сколько требуется шагов методу вращений при разных

Возьмём следующие входные данные, где точность и матрицы:

**Метод решения**

Метод вращения, также известный как метод Якоби или метод вращений Якоби, является итерационным методом для нахождения собственных значений и собственных векторов симметричной матрицы. Основан на итеративном процессе вращений матрицы с целью приближенно привести её к диагональному виду, где собственные значения находятся на главной диагонали.

**Алгоритм решения**

1. Задать точность нахождения собственных значений, .
2. Выделить в верхней треугольной над диагональной частью матрицы А максимальный по модулю элемент
3. Найти по формуле угол поворота
4. Составить матрицу вращения , где:

Остальные элементы матрицы вращений равны нулю.

1. Вычислить новое приближение
2. Повторять шаги 2–5 до тех пор, пока не будет выполняться условие где, задано на первом шаге.

**Подробный анализ кода**

Для реализации алгоритмов решения используется язык программирования python и библиотека numpy.

np.linalg.norm, использующаяся для нахождения нормы разности – встроенная функция numpy, находящая Евклидову норму без указания параметра

**Результаты тестов**

После проведения численного эксперимента с заданными матрицами и точностью, были получены следующие данные:

1. Рассмотрим матрицу :

Вычисленное встроенной функцией максимальное значение: 10.229815082953518

Значения, вычисленные методом вращений:

|  |  |  |
| --- | --- | --- |
|  | Результат | Кол-во шагов |
|  | 10.331140350877194 | 5 |
|  | 10.230497724600053 | 13 |
|  | 10.229823854105094 | 20 |
|  | 10.229815195668705 | 27 |

1. Рассмотрим матрицу :

Вычисленное встроенной функцией максимальное значение: 17.05945859653546

Значения, вычисленные методом вращений:

|  |  |  |
| --- | --- | --- |
|  | Результат | Кол-во шагов |
|  | 17.16936870914093 | 6 |
|  | 17.060544233775644 | 15 |
|  | 17.05947103093332 | 24 |
|  | 17.059458741144795 | 33 |

1. Рассмотрим матрицу :

Вычисленное встроенной функцией максимальное значение: 16.95085232231957

Значения, вычисленные методом вращений:

|  |  |  |
| --- | --- | --- |
|  | Результат | Кол-во шагов |
|  | 16.991761802552855 | 6 |
|  | 16.951275786205073 | 11 |
|  | 16.95085652684473 | 16 |
|  | 16.950852364423717 | 21 |

**Вывод**

Метод вращений является эффективным алгоритмом для нахождения собственных значений матрицы. Он позволяет получить точные численные результаты при решении задачи нахождения собственных значений.

Метод вращений обладает высокой эффективностью при работе с матрицами большой размерности. Это связано с тем, что алгоритм имеет сложность, зависящую линейно от размерности матрицы. Таким образом, метод вращений позволяет обрабатывать большие матрицы с достаточно высокой скоростью и минимальными требованиями к вычислительным ресурсам.

Увеличение точности нахождения собственных значений матрицы влечет за собой увеличение количества итераций метода вращений. Это связано с тем, что увеличение точности требует более точного приближения диагонального элемента матрицы и требует проведения дополнительных итераций для достижения необходимой точности. При выборе значения   
 следует учитывать требуемую точность результата и вычислительные возможности доступной вычислительной системы.

В целом, метод вращений представляет собой полезный и эффективный инструмент для нахождения собственных значений матрицы, и его применение может быть расширено на практике для решения различных задач в различных областях науки и инженерии.

**Приложение**

Листинг программы:

import numpy as np  
import warnings  
  
  
warnings.simplefilter(action="ignore", category=RuntimeWarning)  
  
  
def iterate\_method(\_a, \_n):  
 eps = 10e-4  
 x\_prev = np.full(\_n, 1.)  
 lambda\_prev = -1  
 k = 0  
 while True:  
 x\_next = \_a.dot(x\_prev)  
 lambda\_next = max(x\_next[i] / x\_prev[i] for i in range(\_n))  
 k += 1  
 if np.linalg.norm(lambda\_next - lambda\_prev) <= eps:  
 return lambda\_next, k  
 lambda\_prev = lambda\_next  
 x\_prev = x\_next  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 a3 = np.array([  
 [5, 2, 1],  
 [2, 4, 2],  
 [1, 2, 9]  
 ])  
  
 a5 = np.array([  
 [10, 1, 2, 3, 1],  
 [1, 2, -3, 1, 3],  
 [2, -3, 5, 7, 2],  
 [3, 1, 7, 9, 1],  
 [1, 3, 2, 1, 10]  
 ])  
  
 a7 = np.array([  
 [10, 1, 2, 3, 2, 1, 1],  
 [1, 4, -3, 1, 1, 1, 1],  
 [2, -3, 3, 2, 1, 1, 2],  
 [3, 1, 2, 10, 1, 1, 3],  
 [2, 1, 1, 1, 5, 1, 2],  
 [1, 1, 1, 1, 1, 4, 1],  
 [1, 1, 2, 3, 2, 1, 10]  
 ])  
  
 print("a:")  
 print(a3)  
 print("a:")  
 print(a5)  
 print("a:")  
 print(a7)  
  
 print("n = 3:")  
 print("непосредственное решение: ", max(np.linalg.eigvals(a3)))  
 answer, k = iterate\_method(a3, len(a3))  
 print("решение, полученное методом итераций: ", answer)  
 print("количество шагов: ", k)  
  
 print("n = 5:")  
 print("непосредственное решение: ", max(np.linalg.eigvals(a5)))  
 answer, k = iterate\_method(a5, len(a5))  
 print("решение, полученное методом итераций: ", answer)  
 print("количество шагов: ", k)  
  
 print("n = 7:")  
 print("непосредственное решение: ", max(np.linalg.eigvals(a7)))  
 answer, k = iterate\_method(a7, len(a7))  
 print("решение, полученное методом итераций: ", answer)  
 print("количество шагов: ", k)