



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

Департамент математического и компьютерного
моделирования

ОТЧЕТ

к лабораторной работе №13 по дисциплине
«Вычислительная математика»

Направление подготовки
01.03.02 «Прикладная математика и информатика»

Выполнил студент
гр. Б9119-01.03.02 систпро
Нагорнов С.С.

(ФИО)

(Подпись)

« 20 » _____ июня 2021 г.

г. Владивосток
2021

Содержание

Введение	2
Постановка задачи	2
Алгоритм работы	2
Приложения	3
Вывод	4

Введение

В данной лабораторной работе необходимо найти собственные значения матрицы с помощью метода вращений.

Постановка задачи

Дана матрица A . Необходимо найти все собственные значения λ данной матрицы методом вращений.

Алгоритм работы

1. Найдем коэффициенты максимального абсолютно элемента матрицы, учитывая $i < j$. Коэффициенты обозначим за l и m .
2. Построим единичную матрицу такого же размера, как и матрица A , но при этом определим на l и m элементы. Матрица примет такой вид:

$$T = \begin{pmatrix} 1 & \dots & \dots & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \cos \varphi & \dots & -\sin \varphi & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \sin \varphi & \dots & \cos \varphi & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & \dots & \dots & 1 \end{pmatrix}$$

3. Найдем угол $\varphi = \frac{\arctan\left(\frac{2a_{lm}}{a_{ll}-a_{mm}}\right)}{2}$.
4. Вычислим $A^{(k+1)} = T^T A^{(k)} T$.
5. По итогу, через некоторое количество итераций $k = 1, 2, 3, \dots$ (В нашем случае $k = 15$) на главной диагонали матрицы $A^{(k)}$ бу-

дуг располагаться приближенные к точному собственные значения матрицы A .

Приложения

```
1  import numpy as np
2
3
4  def get_max(A):
5      index = (1, 0)
6      for i in range(A.shape[0]):
7          for j in range(i):
8              if abs(A[index]) < abs(A[i, j]):
9                  index = (i, j)
10     return index
11
12
13 def rotation_method(A):
14
15     for k in range(15):
16         l, m = get_max(A)
17         phi = np.arctan(2 * A[l, m] / (A[l, l] - A[m, m])) / 2
18         T = np.eye(A.shape[0])
19         T[l, l] = np.cos(phi)
20         T[l, m] = -np.sin(phi)
21         T[m, l] = np.sin(phi)
22         T[m, m] = np.cos(phi)
23
24         A = T.T * A * T
25
26     return A
27
28
29 def main():
30     vec = np.matrix([
31         [-0.168700, 0.353699, 0.008540, 0.733624],
32         [0.353699, 0.056519, 0.723182, -0.076440],
33         [0.008540, 0.723182, 0.015938, 0.342333],
34         [0.733624, -0.076440, 0.342333, -0.045744]
35     ])
36     np.set_printoptions(precision=4, suppress=True)
37
38     print(rotation_method(vec).diagonal())
39
40
41 if __name__ == '__main__':
```

```
42     main()
43
```

Листинг 1: Компьютерная реализация алгоритма

Вывод

В данной лабораторной работе был реализован поиск собственных значений с помощью метода вращений.