

Лабораторная работа No 9 «Изучение скорости сходимости однопараметрического метода»

Цель работы

Изучить зависимость скорости сходимости однопараметрического метода в зависимости от значения τ .

Задание

1. Реализовать однопараметрический метод для положительной симметричной матрицы.
2. Используя готовые библиотеки вычислить спектр матрицы A и получить минимальное и максимальное значение спектра λ_{\min} и λ_{\max} . После чего вычислить $\tau_{\text{opt}} = 2/(\lambda_{\min} + \lambda_{\max})$.
3. Построить график зависимости количества итераций решения уравнения $Ax=b$ однопараметрическим методом в зависимости от значения τ лежащего в пределах от 0 до $2/\lambda_{\max}$. Определить τ_{opt} из графика и сравнить с теоретическим значением полученным в пункте 2.
4. Для каждого эксперимента пункта 3 вывести условие сходимости посчитанное по формуле:

```
import numpy
import numpy as np
import matplotlib.pyplot as plt

epsilon = 0.001

def fill_matrix(n, a, b):
    m = np.random.random((n, n)) * (b - a) + a
    for i in range(0, n):
        for j in range(0, i):
            m[i][j] = m[j][i]
            m[i][i] = m[i][i] * 10
    return m

def fill_vector(n, a, b):
    return (np.random.random((n, n)) * (b - a) + a)[0]

def norma_vec(vec):
    max = -float('inf')
```

```
for x in vec:
    if abs(x) > max:
        max = abs(x)

return max


def calc_evklid_diff(x1, x2):
    x = []
    for i in range(0, len(x1)):
        x.append(abs(x1[i] - x2[i]))

    sum = 0
    for i in range(0, len(x)):
        sum += x[i] ** 2

    return numpy.sqrt(sum)


def own_vals(a):
    return min(numpy.linalg.eigh(a)[0]), max(numpy.linalg.eigh(a)[0])


def draw_charts(ts, iters, min_lamb, max_lamb):
    plt.plot(ts, iters, color='blue')
    plt.plot([2 / (min_lamb + max_lamb) for _ in range(len(iters))], iters, color='
    plt.show()


def get_max_mu(own_vals, t):
    mu = []
    for i in range(0, len(own_vals)):
        mu.append((1 - t * own_vals[i]) ** 2)
    return max(mu)


def one_param_method(a, b, n, t):
    x = [1 for i in range(n)]

    p = np.eye(n) - a * t
    g = t * b

    own_vals = numpy.linalg.eigh(a)[0]
    print(f"OWN: {own_vals}")

    max_mu = get_max_mu(own_vals, t)

    iterr = 0
    e_cur = 0
    while True:
        iterr = iterr + 1
        x_temp = p.dot(x) + g

        if norma_vec(x - x_temp) < epsilon:
            return x_temp, iterr
```

```
error_temp = calc_evklid_diff(x_temp, x) ** 2
print(f"ERROR: {error_temp - max_mu * e_cur}")

e_cur = error_temp
x = x_temp

def test_algo():
    n = 5
    left, right = 0.0, 1.0

    b = fill_vector(n, left, right)
    a = fill_matrix(n, left, right)

    min_lamb, max_lamb = own_vals(a)

    t_opt = 2 / (min_lamb + max_lamb)
    t = 0.01
    min_iter = float("inf")
    t_min = 0
    ts = []
    iters = []

    while t < 2 / max_lamb:
        iter_count = one_param_method(a, b, n, t)[1]
        ts.append(t)
        iters.append(iter_count)

        if iter_count < min_iter:
            min_iter = iter_count
            t_min = t

        t += 0.01

    draw_charts(ts, iters, min_lamb, max_lamb)

    print(f"t opt = {t_opt}, calc t = {t_min}")

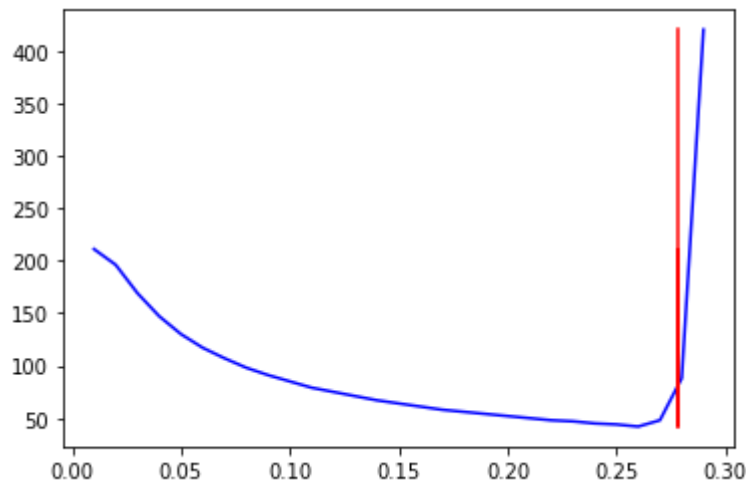
if __name__ == '__main__':
    test_algo()
```



```

ERROR: 1.1104200000000000e-19
ERROR: -8.809142651444724e-20
ERROR: 5.166900978251232e-20
ERROR: 1.1943164556285635e-19
ERROR: 7.623296525288703e-21
ERROR: -1.1180834903756764e-19
ERROR: -5.421010862427522e-20
ERROR: -5.082197683525802e-21
ERROR: 3.4728350837426314e-20
ERROR: 3.3881317890172014e-20
ERROR: -5.251604272976662e-20
ERROR: 0.0
ERROR: 5.336307567702092e-20
ERROR: 2.837560373301906e-20
ERROR: 4.235164736271502e-21
ERROR: -6.395098751769968e-20
ERROR: 1.6940658945086007e-21
ERROR: 7.115076756936123e-20
ERROR: 6.691560283308973e-20
ERROR: 1.1265538198482195e-19
ERROR: 8.004461351553138e-20
ERROR: 6.437450399132683e-20
ERROR: 2.668153783851046e-20
ERROR: -1.5670109524204556e-20
ERROR: 1.2281977735187355e-20
ERROR: 1.8634724839594607e-20
ERROR: 1.7787691892340307e-20
ERROR: -3.8116482626443515e-21
ERROR: -4.1081097941833566e-20
ERROR: -4.362219678359647e-20
ERROR: -1.4399560103323106e-20
ERROR: -6.3527471044072525e-21
ERROR: -2.3505164286306834e-20
ERROR: -2.9434394917086937e-20
ERROR: -4.256340559952859e-20
ERROR: -5.336307567702092e-20
ERROR: 7.199780051661553e-21
ERROR: 6.564505341220828e-20
ERROR: -2.117582368135751e-22
ERROR: -1.9481757786848908e-20
ERROR: 4.129285617864714e-20
ERROR: 2.117582368135751e-22
ERROR: -1.291725244562808e-20
ERROR: 7.623296525288703e-21

```



$t_{\text{opt}} = 0.27834291426283264$, $\text{calc } t = 0.26000000000000006$