Лабораторная работа №3

Тема: «Итерационный степенной метод»

Самоховец Давид

2 курс,1 группа

Постановка задачи

С помощью итерационного степенного метода найти с точностью 610наибольшее по модулю собственное значение и соответствующий ему собственный вектор симметричной матрицы. Вычислительный процесс проводить с нормировкой векторов итерационной последовательности.

Краткие теоретические сведения

Итерационный степенной метод (называемый также степенным

методом) предназначен для нахождения одного или нескольких собственных

значений и соответствующих собственных векторов.

Пусть y(0) – произвольный ненулевой вектор (например, y(0) =[1, 0, …, 0]).



Основные вычисления метода – это реализация итерационного процесса

Пусть наибольшее по модулю собственное значение вещественное и

простое (т.е. не кратное):



Будем предполагать, что

α1≠0.

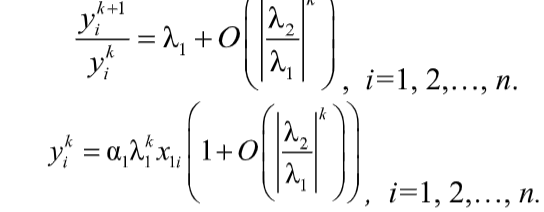
**Замечание 1**. Условие α1≠0 нельзя проверить заранее. Поэтому,

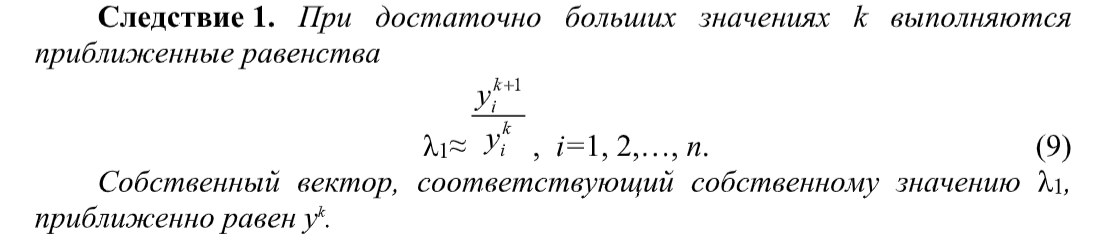
полученные далее приближения собственного значения верны, вообще

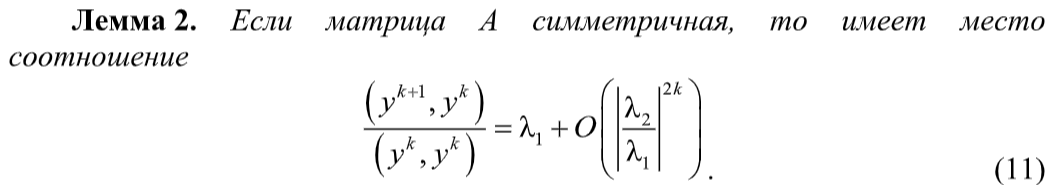
говоря, не для всех y 0 . Если возникают сомнения в правильности

приближения, следует произвести вычисления еще для одного y 0 .

**Лемма 1**. Имеют место соотношения

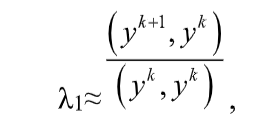


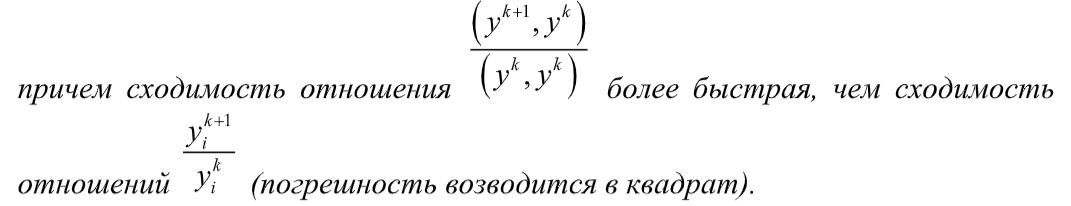


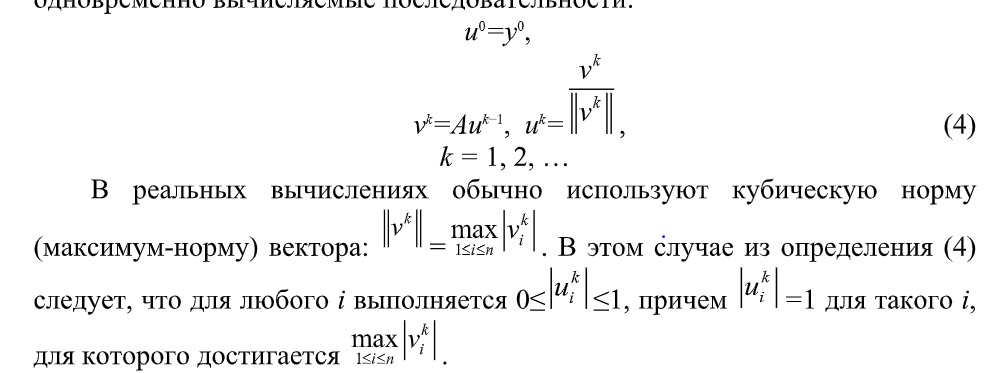


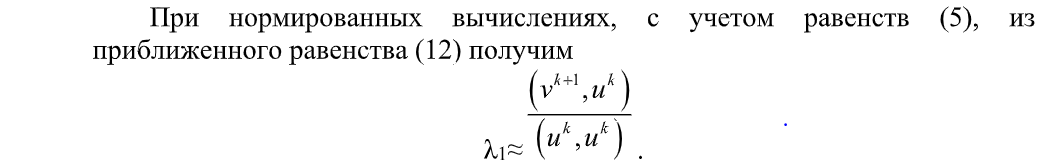
**Следствие 2**. При достаточно больших значениях k выполняются

приближенные равенства









Листинг программы

Итерационный процесс с нормировкой:

void Solve1(double A[N][N])

{

int it = 0;

double lambd2[N] = {1, 1, 1, 1, 1, 1};

double u2[N] = {1, 1, 1, 1, 1, 1};

double v1[N];

double u1[N];

double lambd1[N];

Multiply(A, u2, v1);

double maxValue = MaxNorm(v1);

for(int i = 0; i < N; ++i)

u1[i] = v1[i] / maxValue;

for(int i = 0; i < N; ++i)

lambd1[i] = v1[i] / u2[i];

while (MaxNorm(lambd1, lambd2) > Eps) {

it++;

for(int i = 0; i < N; ++i)

u2[i] = u1[i];

for(int i = 0; i < N; ++i)

lambd2[i] = lambd1[i];

Multiply(A, u2, v1);

maxValue = MaxNorm(v1);

for(int i = 0; i < N; ++i)

u1[i] = v1[i] / maxValue;

for(int i = 0; i < N; ++i)

lambd1[i] = v1[i] / u2[i];

}

cout << lambd1[0] << endl;

cout << "Vector: " << endl;

OutVector(u1);

cout << "Iterations: " << it << "\n";

return;

}

Итерационный процесс для симметричных матриц с нормировкой:

void Solve2(double A[N][N])

{

int it = 0;

double vallambd1;

double vallambd2 = 0;

double u2[N] = {1, 1, 1, 1, 1, 1};

double v1[N];

double u1[N];

Multiply(A, u2, v1);

double maxValue = MaxNorm(v1);

for(int i = 0; i < N; ++i)

u1[i] = v1[i] / maxValue;

vallambd1 = GetLambda(v1, u2);

while (abs(vallambd1 - vallambd2) > Eps)

{

it++;

for(int i = 0; i < N; ++i)

u2[i] = u1[i];

vallambd2 = vallambd1;

Multiply(A, u2, v1);

maxValue = MaxNorm(v1);

for(int i = 0; i < N; ++i)

u1[i] = v1[i] / maxValue;

vallambd1 = GetLambda(v1, u2);

}

cout << vallambd1 << endl;

cout << "Vector: " << endl;

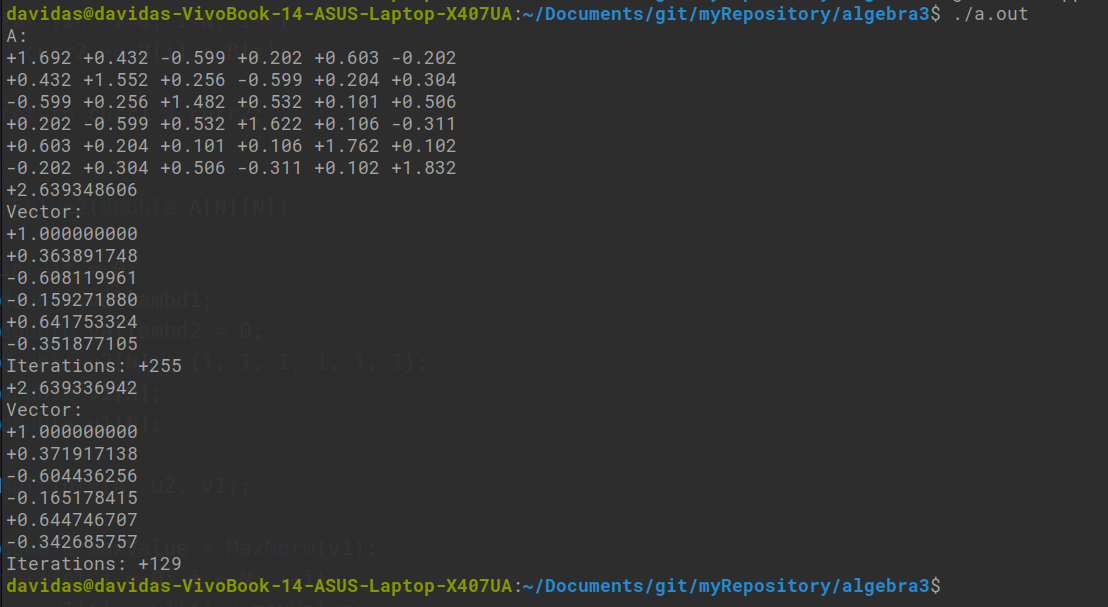
OutVector(u1);

cout << "Iterations: " << it << "\n";

return;

}

Результаты



Вывод:

Итерационно степенной метод является одним из самых популярных и эффективных итерационных методов для нахождения наибольшего по модулю собственного значения и соответствующий ему собственного вектора. Его популярность объясняется простотой в понимании и реализации. Существует несколько вариаций данного метода. Для симметричной матрицы используется специальная вариация метода, обеспечивающая более быструю сходимость(примерно в 2 раза быстрее простой).