

# Numerical Algebra: Homework 15

HUANG Wenchong

3200100006

See my codes in my [Github!](#)

June 20, 2022

# 1 上机题报告

## 1.1 二分法求非退化对称三对角阵指定特征值及对应特征向量的通用子程序

以下是通用子程序 (dichotomy.m)

```
function [lambda,v] = dichotomy(T,m)
% 利用二分法求非退化对称三对角阵T的第m大特征值
    r = norm(T,inf);
    l = -r;
    eps = 1e-15;
    while r-l > eps
        mid = (l+r)/2;
        if variant(T,mid) >= m, r=mid;
        else, l=mid; end
    end
    lambda = l;
    n = size(T,1);
    v = inversePM(T,lambda,ones(n,1));
end

function m = variant(T,x)
% 求变号数
    m = 0;
    q = T(1,1) - x;
    if q<0, m=m+1; end
    n = size(T,1);
    for i = 2:n
        q = T(i,i) - x - T(i,i-1)^2/q;
        if q<0, m=m+1; end
    end
end
```

程序中用到反幂法，其代码如下 (inversePM.m)

```
function v = inversePM(A,lambda,z)
% 反幂法计算特征向量
    n = size(A,1);
    eps = 1e-15;
    v = z;
    while vecnorm(A*v-lambda*v)>=eps
        y = (A-lambda*eye(n))\v;
        v = y/vecnorm(y);
    end
end
```

## 1.2 上机习题 7.2

调用如下程序 (ex\_7\_2.m) 生成一个题中描述的 100 阶对称三对角阵, 并输出最小和最大的特征值及对应的特征向量, 及其数值误差:

```
n = 100;
A = diag(repmat([2], 1, n))+diag(repmat([-1], 1, n-1), 1)+diag(repmat([-1], 1, n-1), -1);
[lambda_min, v_min] = dichotomy(A,1)
err_min = vecnorm(A*v_min-lambda_min*v_min)
[lambda_max, v_max] = dichotomy(A,n)
err_max = vecnorm(A*v_max-lambda_max*v_max)
```

由于特征向量太长, 这里仅给出程序求解的最小、最大特征值, 及对应特征向量的数值误差, 请运行上述程序以查看输出的特征向量

```
lambda_min =

    0.000967435416023399

err_min =

    5.74279407790677e-16

lambda_max =

    3.99903256458398

err_max =

    5.44692122150535e-16
```