f-25-jupyter-foerst-QR

May 4, 2021

[1]:
```
import numpy as np
```

[2]:
```
def house(x):
    norm_x = np.linalg.norm(x)
    if norm_x == 0:
        v = np.zeros_like(x)
        v[0] = 1
        s = 0
    else:
        u = x / norm_x
        eps = -1 if u[0] >= 0 else +1
        s = 1 + np.abs(u[0])
        v = - eps * u
        v[0] += 1
        v /= s
    return v, s

def householder_qr_data(a):
    data = np.copy(a)
    _, k = a.shape
    s = np.empty(k)
    for j in range(k):
        v, s[j] = house(data[j:, [j]])
        data[j:, j:] -= (s[j] * v) @ (v.T @ data[j:, j:])
        data[j+1:, [j]] = v[1:]
    return data, s

def householder_qr(a):
    data, s = householder_qr_data(a)
    n, k = a.shape
    r = np.triu(data[:k, :k])
    q = np.eye(n, k)
    for j in reversed(range(k)):
        x = data[j+1:, [j]]
        v = np.vstack([[1], x])
        q[j:, j:] -= (s[j] * v) @ (v.T @ q[j:, j:])
    return q, r
```

[3]:
```
a = np.array([[1., 2., 1.],
              [2., 1., 1.],
              [1., 1., 3.]])
```

[4]:
```
a.T == a
```

[4]:
```
array([[ True,  True,  True],
       [ True,  True,  True],
       [ True,  True,  True]])
```

[5]:
```
q, r = householder_qr(a)
```

[6]:
```
q, r
```

[6]:
```
(array([[-0.40824829,  0.86164044,  0.30151134],
        [-0.81649658, -0.49236596,  0.30151134],
        [-0.40824829,  0.12309149, -0.90453403]]),
 array([[-2.44948974, -2.04124145, -2.44948974],
        [ 0.        ,  1.3540064 ,  0.73854895],
        [ 0.        ,  0.        , -2.11057941]]))
```

[7]:
```
q @ r
```

[7]:
```
array([[1., 2., 1.],
       [2., 1., 1.],
       [1., 1., 3.]])
```

[8]:
```
a1 = r @ q
a1
```

[8]:
```
array([[ 3.66666667, -1.40705294,  0.86164044],
       [-1.40705294, -0.57575758, -0.25979437],
       [ 0.86164044, -0.25979437,  1.90909091]])
```

[9]:
```
q1, r1 = householder_qr(a1)
a2 = r1 @ q1
a2
```

[9]:
```
array([[ 4.34020619,  0.38439645,  0.35812246],
       [ 0.38439645, -0.97233054,  0.02577822],
       [ 0.35812246,  0.02577822,  1.63212435]])
```

[10]:
```
q2, r2 = householder_qr(a2)
a3 = r2 @ q2
a3
```

[10]:
```
array([[ 4.40668824e+00, -8.88014455e-02,  1.30836212e-01],
       [-8.88014455e-02, -9.98541492e-01, -2.14890229e-03],
```

```
       [ 1.30836212e-01, -2.14890229e-03,  1.59185325e+00]])
```

[11]:
```
q3, r3 = householder_qr(a3)
a4 = r3 @ q3
a4
```

[11]:
```
array([[ 4.41335405e+00,  2.01608034e-02,  4.70963177e-02],
       [ 2.01608034e-02, -9.99924916e-01,  1.75399502e-04],
       [ 4.70963177e-02,  1.75399502e-04,  1.58657087e+00]])
```

[12]:
```
b = np.copy(a)
for i in range(10):
    q, r = householder_qr(b)
    b = r @ q

b
```

[12]:
```
array([[ 4.41421356e+00,  2.72594426e-06,  1.01264816e-04],
       [ 2.72594427e-06, -1.00000000e+00,  5.09658823e-11],
       [ 1.01264816e-04,  5.09659469e-11,  1.58578644e+00]])
```

[13]:
```
np.linalg.eig(a)[0]
```

[13]:
```
array([ 4.41421356, -1.        ,  1.58578644])
```

[14]:
```
a = np.array([[1., 2., -1.],
              [2., -1., 1.],
              [-1., 1., 3.]])
a == a.T
```

[14]:
```
array([[ True,  True,  True],
       [ True,  True,  True],
       [ True,  True,  True]])
```

[16]:
```
b = np.copy(a)
for i in range(30):
    q, r = householder_qr(b)
    b = r @ q

b
```

[16]:
```
array([[ 3.42362157e+00,  1.84994907e-03, -1.09531882e-05],
       [ 1.84994907e-03, -2.57699406e+00,  2.11786853e-02],
       [-1.09531882e-05,  2.11786853e-02,  2.15337249e+00]])
```

[17]:
```
np.linalg.eig(a)[0]
```

[17]:
```
array([-2.57708945,  2.15346731,  3.42362214])
```

[ ]: