exp5

March 4, 2022

[ ]:
```
import numpy as np
from sklearn.decomposition import TruncatedSVD
```

[ ]:
```
A = np.array([[3,4,3],[1,2,3],[4,2,1]])

U, D, VT = np.linalg.svd(A)
U,D,VT

print("A",A)


print(f"\nU\n{U}\n\nD\n{D}\n\nVT\n{VT}")
```

```
A [[3 4 3]
 [1 2 3]
 [4 2 1]]

U
[[-0.73553325 -0.18392937 -0.65204358]
 [-0.42657919 -0.62196982  0.65664582]
 [-0.52632788  0.76113306  0.37901904]]

D
[7.87764972 2.54031671 0.69958986]

VT
[[-0.60151068 -0.61540527 -0.5093734 ]
 [ 0.73643349 -0.18005275 -0.65210944]
 [ 0.30959751 -0.76737042  0.5615087 ]]
```

[ ]:
```
op = U.dot(np.diag(D)).dot(VT)

print(f"U . diag(D) . VT \n {op}")
```

```
U . diag(D) . VT
 [[3. 4. 3.]
 [1. 2. 3.]
 [4. 2. 1.]]
```

[ ]:
```
# transformation matrix with skelarn.decompose.TruncatedSVD
TruncatedSVD(n_components=2).fit_transform(A)
```

[ ]:
```
array([[ 5.7942733 , -0.46723885],
       [ 3.3604414 , -1.58000033],
       [ 4.14622666,  1.93351902]])
```

[ ]:

[ ]: