activity_15_starter

March 23, 2023

[ ]:
```
import numpy as np
from scipy.io import loadmat
```

[ ]:
```
Xtrue = loadmat("incomplete.mat")["Xtrue"]
Y1 = loadmat("incomplete.mat")["Y1"]
Y2 = loadmat("incomplete.mat")["Y2"]
Y3 = loadmat("incomplete.mat")["Y3"]
```

[ ]:
```
def ItSingValThresh(Y, r):
    """
    Iterative Singular Value Thresholding function for Matrix Completion
    """
    tol = 10**(-3)  # difference between iterates at termination
    max_its = 100
    n,p = Y.shape
    X = np.array(Y) #make a copy so operations do not mutate the original
    X[np.isnan(X)] = 0 # Fill in missing entries with zeros

    err = 10**6
    itt = 0

    while err > tol and itt < max_its:
        U,s,VT = np.linalg.svd(X, full_matrices=False)
        V, S = VT.T, np.diag(s)
        Xnew = np.dot(U[:,:r],np.dot(S[:r,:r],V[:,:r].T))
        for i in range(n):
            for j in range(p):
                if ~np.isnan(Y[i,j]):  #replace Xnew with known entries
                    Xnew[i,j] = Y[i,j]
        err = np.linalg.norm(X-Xnew,'fro')
        X = Xnew
        itt += 1
    return X
# compare error with Xtrue by using rank 2 approximation
print("Y1 with rank 2")
print(np.linalg.norm(Xtrue-ItSingValThresh(Y1,2), 'fro'))
print("Y2 with rank 2")
```

```
print(np.linalg.norm(Xtrue-ItSingValThresh(Y2,2), 'fro'))
print("Y3 with rank 2")
print(np.linalg.norm(Xtrue-ItSingValThresh(Y3,2), 'fro'))

# compare error with Xtrue by using rank 3 approximation
print("Y1 with rank 3")
print(np.linalg.norm(Xtrue-ItSingValThresh(Y1,3), 'fro'))
print("Y2 with rank 3")
print(np.linalg.norm(Xtrue-ItSingValThresh(Y2,3), 'fro'))
print("Y3 with rank 3")
print(np.linalg.norm(Xtrue-ItSingValThresh(Y3,3), 'fro'))
```

```
Y1 with rank 2
87.24667705099665
Y2 with rank 2
0.00473559952738469
Y3 with rank 2
0.000715321865523942
Y1 with rank 3
128.7780484677201
Y2 with rank 3
48.97940976510764
Y3 with rank 3
20.785069891602074
```

0.0.1 b)

The rank is incorrect and thus the error increases