Pattern Recognition Methods
and
Introduction to Machine Learning

Homework 4 - Report
PCA

Prepared by: Mehmet Kapson

1.05.2019

I tried to apply PCA on different pen images. I progressed a little bit; however, I could not bring back the images of the result.

Here is the code that I write:

```
import numpy as np
from matplotlib import image
from matplotlib import pyplot
import matplotlib.pyplot as plt
from PIL import Image
import os
from sklearn.decomposition import PCA
from sklearn.preprocessing import normalize
from pylab import *

loaded_images = list()
size_300 = (300,300)

for f in os.listdir('.'):
    if f.endswith('.jpg'):
        i = Image.open(f).convert(mode='L')
        i.thumbnail(size_300)
        i.save('gray/{}'.format(f))


for f in os.listdir('gray'):
        i = image.imread(f)
        loaded_images.append(i)


imgs_array = np.asarray(loaded_images)
print(imgs_array.shape)

imgs_array_reshaped = np.reshape(imgs_array,[21, 1836*3264*3])
imgs_array_norm = normalize(imgs_array_reshaped)

pca = PCA(5)
img_pca = pca.fit(imgs_array_norm)
print (np.sum(img_pca.explained_variance_ratio_))

B = img_pca.transform(imgs_array_norm)

temp = img_pca.inverse_transform(B)
temp = np.reshape(temp, [21,1836,3264,3])

fig1, ax = plt.subplots(figsize=(12,8))
ax.scatter(B[:, 0], B[:, 1])
plt.show()

"""
```

```
fig2, axa=plt.subplots(figsize=(300,300))
axa.imshow(temp[0,],cmap='gray')
plt.show()
"""
```

And this is the result for n_components = 5 in PCA.

Explained variance is %64.