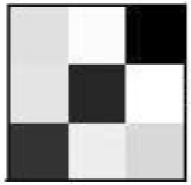
Lab 03

Lab 03

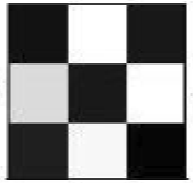
- PCA
- Cov is symmetric

PCA toy problem

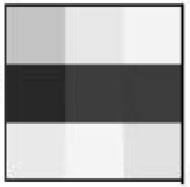
$x^1 =$



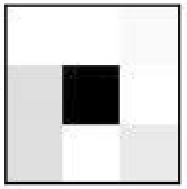
$x^2 =$



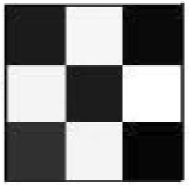
$x^3 =$



$x^4 =$

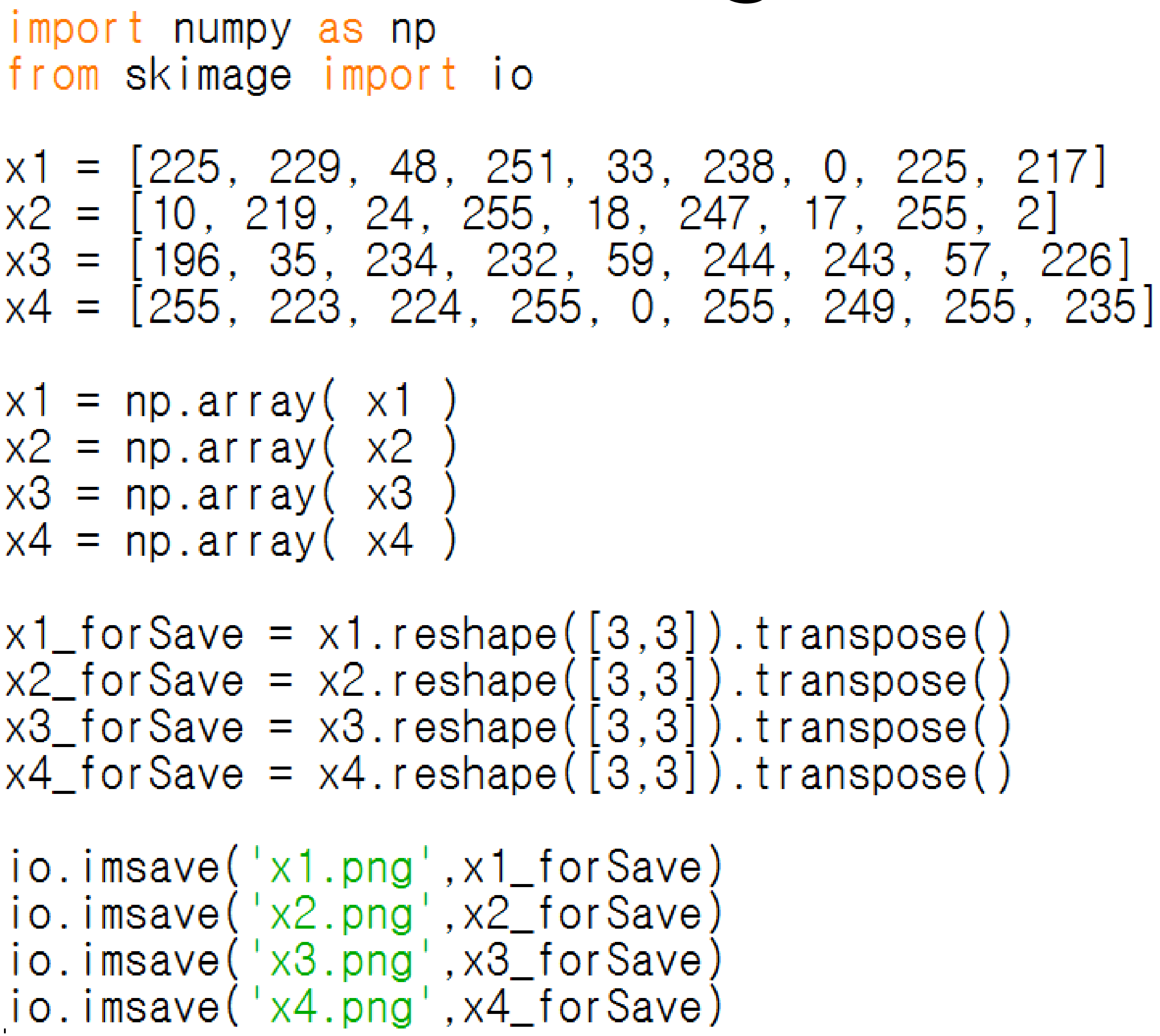


$y^1 =$



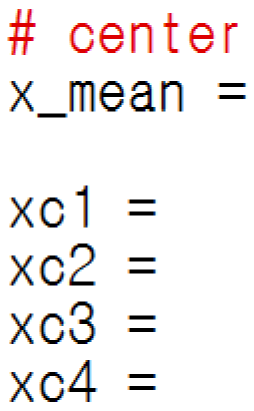
$$x^1 = \begin{bmatrix} 225 \\ 229 \\ 48 \\ 251 \\ 33 \\ 238 \\ 0 \\ 255 \\ 217 \end{bmatrix} \quad x^2 = \begin{bmatrix} 10 \\ 219 \\ 24 \\ 255 \\ 18 \\ 247 \\ 17 \\ 255 \\ 2 \end{bmatrix} \quad x^3 = \begin{bmatrix} 196 \\ 35 \\ 234 \\ 232 \\ 59 \\ 244 \\ 243 \\ 57 \\ 226 \end{bmatrix} \quad x^4 = \begin{bmatrix} 255 \\ 223 \\ 224 \\ 255 \\ 0 \\ 255 \\ 249 \\ 255 \\ 235 \end{bmatrix} \qquad y^1 = \begin{bmatrix} 20 \\ 244 \\ 44 \\ 246 \\ 21 \\ 244 \\ 4 \\ 255 \\ 2 \end{bmatrix}$$

Draw images



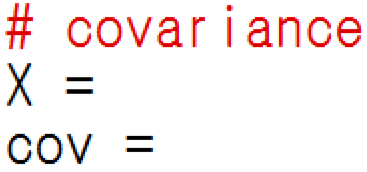
```
import numpy as np
from skimage import io

x1 = [225, 229, 48, 251, 33, 238, 0, 225, 217]
x2 = [10, 219, 24, 255, 18, 247, 17, 255, 2]
x3 = [196, 35, 234, 232, 59, 244, 243, 57, 226]
x4 = [255, 223, 224, 255, 0, 255, 249, 255, 235]

x1 = np.array( x1 )
x2 = np.array( x2 )
x3 = np.array( x3 )
x4 = np.array( x4 )

x1_forSave = x1.reshape([3,3]).transpose()
x2_forSave = x2.reshape([3,3]).transpose()
x3_forSave = x3.reshape([3,3]).transpose()
x4_forSave = x4.reshape([3,3]).transpose()

io.imsave('x1.png',x1_forSave)
io.imsave('x2.png',x2_forSave)
io.imsave('x3.png',x3_forSave)
io.imsave('x4.png',x4_forSave)
```

Center data



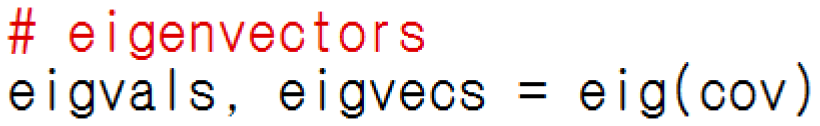
```
# center
x_mean =

xc1 =
xc2 =
xc3 =
xc4 =
```

Covariance matrix



```
# covariance
X =
cov =
```

Compute eigenvectors

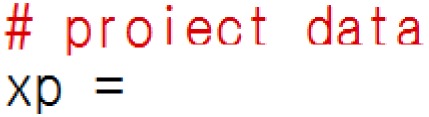


```
# eigenvectors
eigvals, eigvecs = eig(cov)
```

Construct d-dim eigenspace

```
# eigenspace
```

Project data



```
# proiect data
xp =
```

Test

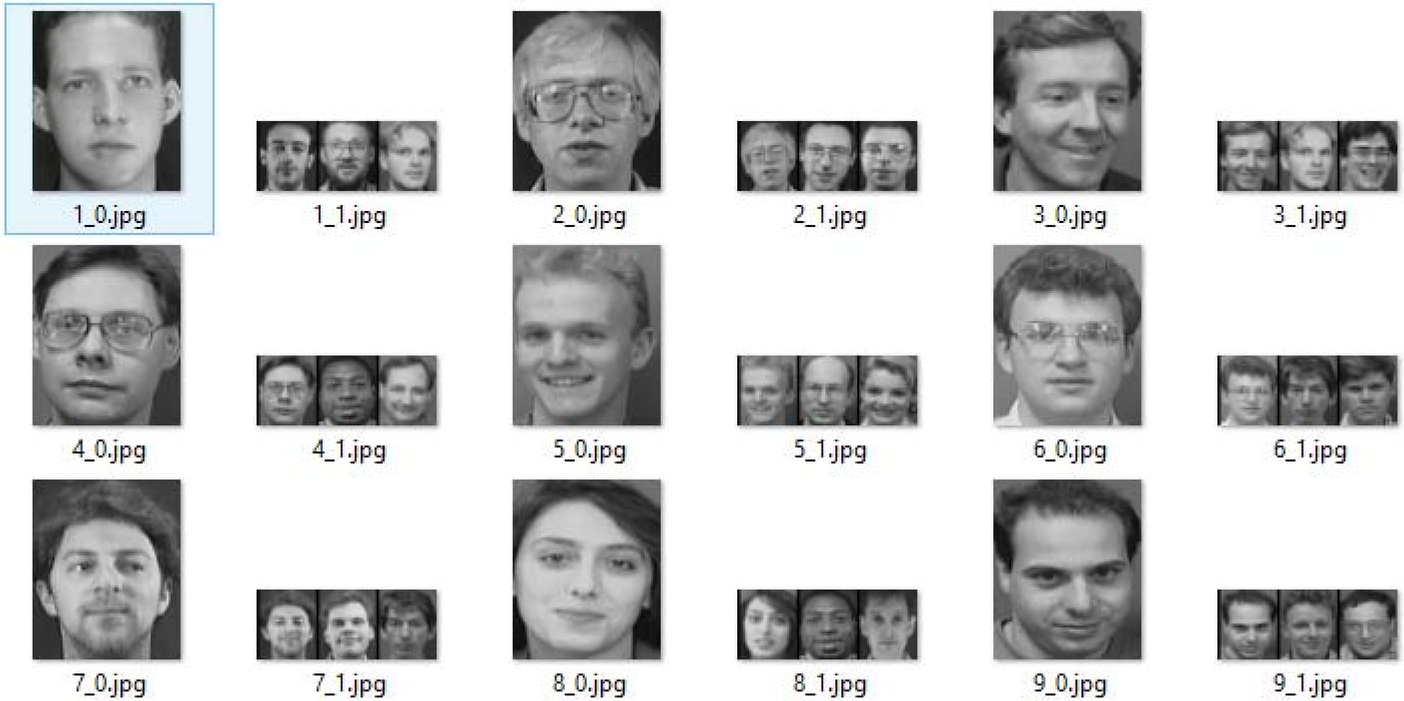
```
# test
y = [20, 244, 44, 246, 21, 244, 4, 255, 2]

yp =
diff = xp-                          yp
dist =
```

homework

- Implement Eigenfaces for Recognition i.e., do the same thing with images
- Use
  1) Original algorithm
  2) Trick with AA^T
  3) SVD
- And compare time consumption and accuracy

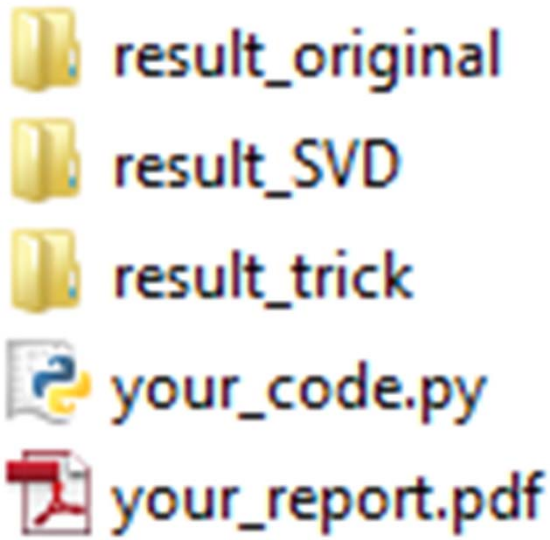
Homework - report

- Display eigenspace as eigenfaces
- Show three closest images for each 40 train image and 40 test image



Homework - submission

- Zip your
  - Code
  - Report
  - Result images
    - Result_original
    - Result_trick
    - Result_SVD



result_original
result_SVD
result_trick
your_code.py
your_report.pdf