FICHA LABORATORIAL LAB3 - Análise de componentes principais

Ex1. Corre o código abaixo observando o propósito de cada instrução. Coloca a localização no teu computador do ficheiro 'MINISTsmall.p' que se encontra no Moodle

```
import numpy as np
from matplotlib import pyplot as plt
import pickle

```

```
s=pickle.load(open('/Users/gxufre/Dropbox/ARQUIVO/0
s.keys()
```

Visualizar os dados

```
digitos=dados["X"]
trueClass=dados["trueClass"]
f1=dados["foldTrain"]
f2=dados["foldTest"]
```

```
train3=digitos[:,(trueClass==3) & f1]
I3_8=np.reshape(train3[:,7],(28,28))
plt.imshow(255-I3_8, cmap='gray',interpolation=None)
```

```
I3_m=np.reshape(np.mean(train3,axis=1),(28,28))
plt.imshow(255-I3_m, cmap='gray',interpolation=None)
```

Ex.2. Corre o código abaixo observando o propósito de cada instrução

```
C0=np.cov(X[:,trueClass==0])
C1=np.cov(X[:,trueClass==1])
C2=np.cov(X[:,trueClass==2])
C3=np.cov(X[:,trueClass==3])
C4=np.cov(X[:,trueClass==4])
C5=np.cov(X[:,trueClass==5])
C6=np.cov(X[:,trueClass==6])
C7=np.cov(X[:,trueClass==7])
C8=np.cov(X[:,trueClass==8])
C9=np.cov(X[:,trueClass==9])
```

```
v=np.diag(C7)
z7=(v==0)*1
plt.imshow(1-z7.reshape((28,28)),cmap='gray')
```

```
z0=(np.diag(C0)!=0)
plt.imshow(z0.reshape((28,28)),cmap='gray')
```

```
z9=(np.diag(C9)!=0)
plt.imshow(z9.reshape((28,28)),cmap='gray')
```

Encontra uma justificação para as imagens obtidas.

ISEL INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Rua Conselheiro Emídio Navarro 1
1959-007 Lisboa- Portugal
Tel: (+351) 218 317 000
Fax: (+351) 218 317 162
Web: www.isel.pt

Ex.3. Corre o código abaixo observando o propósito de cada instrução

Encontrar os valores e vetores próprios

```
X=digitos
Cx=np.cov(X) #no conjunto total de dados
(v,W)=np.linalg.eig(Cx) #v valores proprios, W vetores próprios
```

```
np.imag(v).max() #verificar máximo imaginário
v=v.real #descartar a parte imaginária dos valores próprios
```

```
idx=np.argsort(-v) #vamos ordenar os valores próprios por ordem descrescente
v=v[idx] #ordenar os valores próprios
W=W[:,idx] #ordenaro os vetores próprios
W=W[:,v>=1e-10] #remover componetes com valores próprios próximo de zero
W=W.real #retirar a parte imaginária dos vetores próprios


```

```
plt.plot(np.log(v))
```

Observa a importância da aplicação do logaritmos e Identifica o problema com o gráfico obtido. Apresenta uma solução para resolver o problema-

Ex. 4. Corre o código abaixo observando o propósito de cada instrução

Selecionar as componentes principais

```
v=v/np.sum(v) #Peso relativo de cada valor próprio
L=np.cumsum(v) #Peso relativo acumulado para cada valor próprio
plt.plot(L)
```

```
np.sum(L<=0.99) #número de componentes com 99% da variância
```

```
np.sum(L<=0.95) #número de componentes com 95% da variância
```

```
k=150
```

```
m=np.mean(X,axis=1)[:,np.newaxis]
Xn=X-m
```

```
W2=W[:,:k]
Xp=np.dot(W2.T,X)
Xr=np.dot(W2,Xp)+m
```

```
x=Xr[:,4508].reshape(28,28)
print(x.min(),x.max())
x=x-x.min()
x=x/x.max()*255
plt.imshow(x.astype('uint8'),cmap='binary')
```

```
x=Xr[:,4508].reshape(28,28)
x=np.clip(x,0,255)
plt.imshow(x.astype('uint8'),cmap='binary')
```

ISEL INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Rua Conselheiro Emídio Navarro 1
1959-007 Lisboa- Portugal
Tel: (+351) 218 317 000
Fax: (+351) 218 317 162
Web: www.isel.pt

Ex. 5. Corre o código abaixo observando o propósito de cada instrução

Usar o classificador k-NN

```
X1=X[:,f1]
X2=X[:,f2]
y1=trueClass[f1]
y2=trueClass[f2]
print(X1.shape,X2.shape)
```

Calcular as componentes principais

```
import scipy.linalg as la
Cx=np.cov(X1) #no conjunto de treino!!!!
u,V=la.eig(Cx)
u=u.real
V=V.real
```

```
idx=np.argsort(-u)
u=u[idx]
V=V[:,idx]
```

```
print(np.sum(u>=1e-6))
```

```
W=V[:,:664]
```

```
m=np.mean(X1,axis=1)[:,np.newaxis]
X1n=X1-m
X1p=np.dot(W.T,X1n)

```

```
print(X1p.shape)
```

```
#Projetar nas componentes e fazer a mesma normalização para os dados de teste
X2n=X2-m
X2p=np.dot(W.T,X2n)
X2p=np.dot(S,X2p)
```

ISEL INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Rua Conselheiro Emídio Navarro 1
1959-007 Lisboa- Portugal
Tel: (+351) 218 317 000
Fax: (+351) 218 317 162
Web: www.isel.pt

Ex. 6. Corre o código abaixo observando o propósito de cada instrução

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix
from plotMatriz import plotMatriz

#Dados originais

knn=KNeighborsClassifier(n_neighbors=1).fit(X1.T,y1)
y2e=knn.predict(X2.T)
CM=confusion_matrix(y2,y2e)
plotMatriz(CM)
print(np.sum(y2!=y2e)/5000*100)
```

```
# Dados projetados nas componentes principais
knn_pca=KNeighborsClassifier(n_neighbors=1).fit(X1p.T,y1)
y2e=knn_pca.predict(X2p.T)
CM=confusion_matrix(y2,y2e)X
plotMatriz(CM)
print(np.sum(y2!=y2e)/5000*100)
```

Ex. 7. Corre o código abaixo observando o propósito de cada instrução. Experimenta vários valores de número de componentes a utilizar e diferentes números de vizinhos no Knn. Tenta descobri o conjunto de melhores pares nº de componentes/vizinhos para classificar os dígitos.

Fazer experiências

```
n_pca=???
vizinhos=???

W=V[:,:n_pca]
m=np.mean(X1,axis=1)[:,np.newaxis]
X1n=X1-m
X1p=np.dot(W.T,X1n)
X2n=X2-m
X2p=np.dot(W.T,X2n)

knn_pca=KNeighborsClassifier(n_neighbors=vizinhos).fit(X1p.T,y1)
y2e=knn_pca.predict(X2p.T)
CM=confusion_matrix(y2,y2e)
plotMatriz(CM)
print(np.sum(y2!=y2e)/5000*100)
```


Rua Conselheiro Emídio Navarro 1
1959-007 Lisboa- Portugal
Tel: (+351) 218 317 000
Fax: (+351) 218 317 162
Web: www.isel.pt