CSE 455 Homework 1

Heyi Sun(50198052)

1.

a. Use np.mean() to calculate the mean matrix

b. Use np.cov() to calculate the covariance matrix, note that we need the diagonal to output the images

c. Use the the PIL library to output the images

Mean:  (.png)

Varience:  (.bmp)

2.

a. Implement the discriminant function

```
def discriminant(x,i,args):
    avg,cova,prob=args
    first=np.diag(np.dot(np.dot(x,bigw(cova[i])),np.transpose(x)))
    second=np.dot(np.transpose(smallw(cova[i],np.transpose([avg[i]]))),np.transpose(x))
    third1=-1.0/2.0*np.dot(np.dot(avg[i],np.linalg.pinv(cova[i])),np.transpose(avg[i]))
    third2=np.log(prob[i]/np.sum(prob))
    return first+second+third1+third2
```

Use the discriminant function to make decision and calculate the accuracy

```
a=np.argmax(classify(t_images,args).reshape(10,10000),0)
for i in range(len(t_images)):
    if(a[i]==t_labels[i]):
        true=true+1
    else:
        false=false+1
```

As the outcome of my code, the accuracy is 85.72%

```
print('accuracy='+str(100*true/(true+false))+'%')
```

```
In [34]: runfile('E:/!Cse 455/Hw1/TEST2py.py', wdir='E:/!
Cse 455/Hw1')
accuracy=85.72%
```

The performance is not as good as other methods because this method is lack of the update of its parameters. It will get better it optimizes the parameters after process every training sample.