In [1]:

```
#导入包和必要的工具
import pandas as pd
import numpy as np
from sklearn.decomposition import PCA
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
```

In [2]:

```
#读取训练集和测试集
train = pd.read_csv('./data/MNIST_train.csv')
test = pd.read_csv('./data/MNIST_test.csv')
y_train = train.label.values
y_train_pca = train.label.values
x_train = train.drop("label",axis = 1).values
x_test = test.values
#将像素值[0，255]——>[0,1]
x_train = x_train / 255.0
x_test = x_test / 255.0
```

In [3]:

```
#原始输入的特征维数和样本数目
print('the shape of train_image:{}'.format(x_train.shape))
print('the shape of test_image:{}'.format(x_test.shape))
#PCA降维
pca = PCA(n_components = 0.95,svd_solver = 'full')
pca.fit(x_train)
```

```
the shape of train_image:(42000, 784)
the shape of test_image:(28000, 784)
```

Out[3]:

```
PCA(n_components=0.95, svd_solver='full')
```

In [4]:

```
#在训练集和测试集上降维
x_train_pca = pca.transform(x_train)
x_test_pca = pca.transform(x_test)
x_train_pca.shape
```

Out[4]:

```
(42000, 154)
```

In [5]:

```
#LAD分类器(降维前的数据)
lda = LinearDiscriminantAnalysis()
lda.fit(x_train,y_train)
lda.predict(x_test)
```

Out[5]:

```
array([2, 0, 9, ..., 3, 9, 2], dtype=int64)
```

In [6]:

```
#LAD分类器(降维后的数据)
lda2 = LinearDiscriminantAnalysis()
lda2.fit(x_train_pca,y_train_pca)
lda2.predict(x_test_pca)
```

Out[6]:

```
array([2, 0, 9, ..., 3, 9, 2], dtype=int64)
```

In [7]:

```
#用降维前的全体训练数据集上的训练的模型对测试集进行测试
y_predict = lda.predict(x_test)
```

In [8]:

```
#用降维后的全体训练数据集上的训练的模型对测试集进行测试
y_predict2 = lda2.predict(x_test_pca)
```

In [9]:

```
#生成提交测试结果(未降维)
df = pd.DataFrame(y_predict)
df.columns = ['Lable']
df.index += 1
df.index.name = 'imageid'
df.to_csv('SVC_Minist_submission.csv',header = True)
#生成提交测试结果(降维)
df = pd.DataFrame(y_predict2)
df.columns = ['Lable']
df.index += 1
df.index.name = 'imageid'
df.to_csv('SVC_Minist_submission_pca.csv',header = True)
```

In [10]:

```
#交叉验证(未降维)
from sklearn.model_selection import cross_val_score
loss = cross_val_score(lda,x_train,y_train,cv = 5)
print('accuracy of each fold is :',loss)
print('cv accuracy is :',loss.mean())
loss.mean()
```

```
accuracy of each fold is : [0.86404762 0.86559524 0.86369048 0.86083333 0.8677381 ]
cv accuracy is : 0.8643809523809525
```

Out[10]:

```
0.8643809523809525
```

In [11]:

```
#交叉验证(降维)
from sklearn.model_selection import cross_val_score
loss = cross_val_score(lda2,x_train_pca,y_train_pca,cv = 5)
print(’accuracy of each fold is :’,loss)
print(’cv accuracy is :’,loss.mean())
loss.mean()
```

```
accuracy of each fold is : [0.87071429 0.86869048 0.8652381  0.86547619 0.8727381 ]
cv accuracy is : 0.8685714285714287
```

Out[11]:

```
0.8685714285714287
```