```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from sklearn.datasets.samples_generator import make_blobs
# X 为样本特征，Y 为样本簇类别， 共 1000 个样本，每个样本 3 个特征，共 4 个簇
X, y = make_blobs(n_samples=10000, n_features=3, centers=[[3,3, 3], [0,0,0], [1,1,1], [2,2,2]],
cluster_std=[0.2, 0.1, 0.2, 0.2], random_state =9)
fig = plt.figure()
ax = Axes3D(fig, rect=[0, 0, 1, 1], elev=30, azim=20)
plt.scatter(X[:, 0], X[:, 1], X[:, 2],marker='o')
```

```
from sklearn.decomposition import PCA
pca = PCA(n_components=3)
pca.fit(X)
print pca.explained_variance_ratio_
print pca.explained_variance_
```

```
pca = PCA(n_components=2)
pca.fit(X)
print pca.explained_variance_ratio_
print pca.explained_variance_

X_new = pca.transform(X)
plt.scatter(X_new[:, 0], X_new[:, 1],marker='o')
plt.show()
```

```
pca = PCA(n_components=0.95)
pca.fit(X)
print pca.explained_variance_ratio_
print pca.explained_variance_
print pca.n_components_
```

```
pca = PCA(n_components=0.99)
pca.fit(X)
print pca.explained_variance_ratio_
print pca.explained_variance_
print pca.n_components_
```

运用 pca.py 的代码

```
dataMat=pca.loadDataSet('testSet.txt')
```

```
lowDMat,reconMat=pca.pca(dataMat,1)
import matplotlib
import matplotlib.pyplot as plt
fig=plt.figure()
ax=fig.add_subplot(111)
ax.scatter(dataMat[:,0].flatten().A[0],dataMat[:,1].flatten().A[0],marker='^',s=90)
ax.scatter(reconMat[:,0].flatten().A[0],reconMat[:,1].flatten().A[0],marker='o',s=50,c='red')
```