KMeans 演示

```python
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
import numpy as np
import matplotlib.pyplot as plt

# 构造数据集,分别来自两个正态总体
X1 = np.random.normal(loc=0.2,scale=0.2,size=(100,100))
X2 = np.random.normal(loc=0.9,scale=0.1,size=(100,100))
X = np.r_[X1,X2]

plt.figure(1)
plt.title("Dataset")
plt.scatter(X[:,0],X[:,1])
plt.show()
```
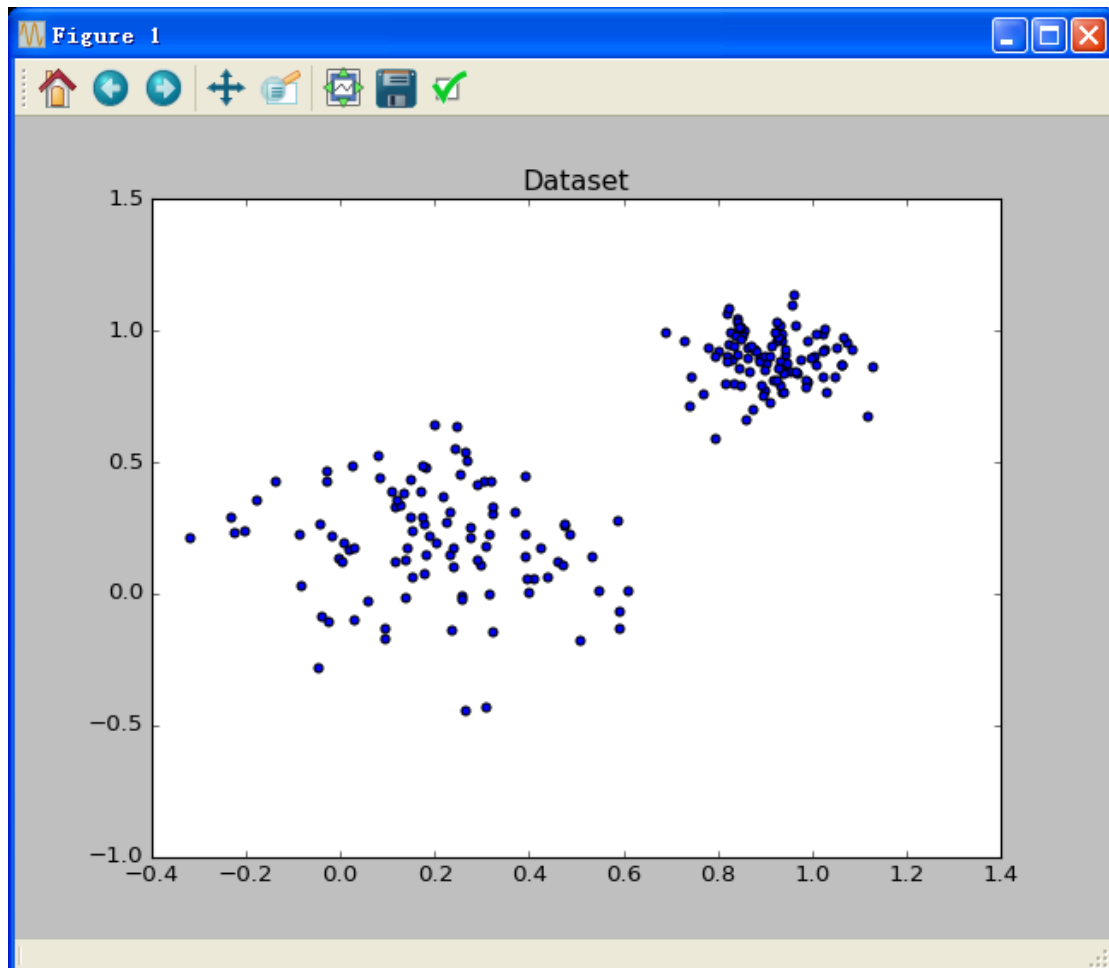


```python
# 聚类个数 k = 2/3/4/5
ss_list = []
```

```
for k in range(2,6):
    model = KMeans(n_clusters=k,init='random')
    model.fit(X)
    labels = model.labels_
    print labels
    ss = silhouette_score(X,labels)
    ss_list.append(ss)
```

[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

[1 0 1 0 0 1 0 1 0 0 0 0 0 1 1 0 0 0 1 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0
 1 0 1 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 1 0 0 1 0 0 0 0 1 0 0 1 0 0
 0 0 0 1 1 0 0 0 1 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0 1 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2]

[1 2 1 3 3 2 3 3 2 3 1 3 2 3 3 2 1 3 1 3 2 3 3 2 3 3 3 3 2 1 3 2 3 3 3 3 2 1 1
 3 3 3 2 2 3 2 3 3 2 2 1 2 1 3 2 2 2 3 1 2 2 3 3 2 2 3 1 2 1 1 3 3 3 3 3 2
 3 3 2 1 1 2 1 3 3 3 3 2 3 2 2 3 3 2 3 3 3 3 3 3 2 2 3 3 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

[0 0 3 1 3 3 2 2 3 2 2 3 2 2 1 0 2 3 2 3 2 2 2 2 3 3 0 0 3 3 0 0 1 2 1 0 3
 3 3 0 3 0 0 2 2 2 2 0 2 2 3 0 0 0 2 2 3 0 3 2 3 2 3 2 1 0 2 2 2 2 2 2 2
 0 2 0 3 3 0 0 2 0 0 0 3 0 1 2 2 3 2 0 2 2 2 0 0 0 3 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 4 4]

```
# 绘制 k 的各种值对应的轮廓系数,可见 k=2 时轮廓系数最高.
plt.figure(2)
plt.plot(range(2,6),ss_list)
plt.title("Cluster Quality")
plt.xlabel("No of clusters k")
plt.ylabel("Silhouette Coefficient")
plt.show()
```