

K-Means Clustering Algorithm Example-2

Unsupervised machine learning models ထဲမှ တစ်ခုဖြစ်သည့် clustering algorithms အကြောင်းဥပမာကို ရှင်းပြထားသည်။

Scikit-Learn ထဲတွင် clustering algorithm များစွာပါဝင်သည်။ အရိုးရှင်းဆုံးနည်းမှာ k-means clustering algorithm ဖြစ်သည်။ sklearn.cluster.KMeans ကို အသုံးပြုထားသည်။
ပထမဦးစွာ လိုအပ်သည့် package များကို import လုပ်သည်။

```
In [1]: %matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns; sns.set() # for plot styling
import numpy as np
```

Introducing k-Means

k-means algorithm ၏ အလုပ်မှာ unlabeled multidimensional dataset များထဲမှ ကြိုတင်သတ်မှတ်ထားသည့် အစုငယ်(cluster)များ ဖွဲ့ပေးရန် ဖြစ်သည်။

အစုငယ်(cluster)တိုင်းတွင် center တစ်ခုစီ ရှိသည်။"cluster center" သည် cluster ထဲရှိ point တစ်ခုချင်းစီနှင့် center အကြားအကွာအဝေးတို့၏ ပျမ်းမျှတန်ဖိုး (arithmetic mean of all the points belonging to the cluster) ဖြစ်သည်။

sklearn.datasets.samples_generator မှ make_blobs ဖြင့် ဒေတာများကို ထုတ်ယူ(generate)သည်။ ဒေတာ point ပေါင်း (၁၀၀) ပါသည့် data set ဖြစ်သည်။ (n_samples=100)

ဒေတာ များကို ကြည့်လိုပါက print(X) ဖြင့် ကြည့်နိုင်သည်။

```
In [2]: from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=100, centers=4,
                        cluster_std=0.60, random_state=0)
# print(X)
print(y_true)
```

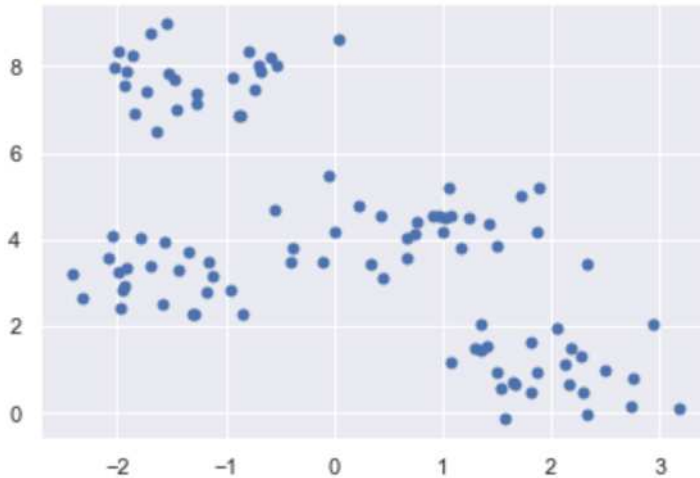
```
[0 3 0 0 0 0 2 3 0 3 3 3 3 3 1 1 2 2 1 0 3 2 1 0 2 2 0 1 1 1 3 1 1 2 0 3
 1 3 2 0 2 3 2 2 3 1 2 0 0 0 1 2 2 2 3 3 1 1 3 3 1 1 0 1 3 2 2 1 0 3 1 0 3
 0 0 2 2 1 1 1 3 2 0 1 2 1 1 0 0 0 2 0 2 2 3 3 2 3 0]
```

Plotting

plt.scatter ဖြင့် scatter ဂရပ်ပုံ ဆွဲသည်။

X[:, 0] သည် ဂရပ်ဆွဲရန်အတွက် X ဝင်ရိုးတန်းဖိုးများဖြစ်ကြပြီး X[:, 1] သည် Y ဝင်ရိုးတန်းဖိုးများဖြစ်ကြသည်။ s=25 သည် ဂရပ် ၏ marker အရွယ်အစား ဖြစ်သည်။

In [3]: `plt.scatter(X[:, 0], X[:, 1], s=25);`



`sklearn.cluster` မှ `KMeans` ကို အသုံးပြုရန် `import` လုပ်သည်။

In [4]: `from sklearn.cluster import KMeans`
`kmeans = KMeans(n_clusters=4)`
`kmeans`

Out[4]: `KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300, n_clusters=4, n_init=10, n_jobs=None, precompute_distances='auto', random_state=None, tol=0.0001, verbose=0)`

`n_clusters=4` သည် cluster (၄)ခု ဖွဲ့စည်းပေးရန် သို့မဟုတ် cluster (၄)ခု ပိုင်းခြားပေးရန် ဖြစ်သည်။
`kmeans.fit(X)` ဖြင့် ဒေတာများကို models ထဲတွင် fit လုပ်သည်။

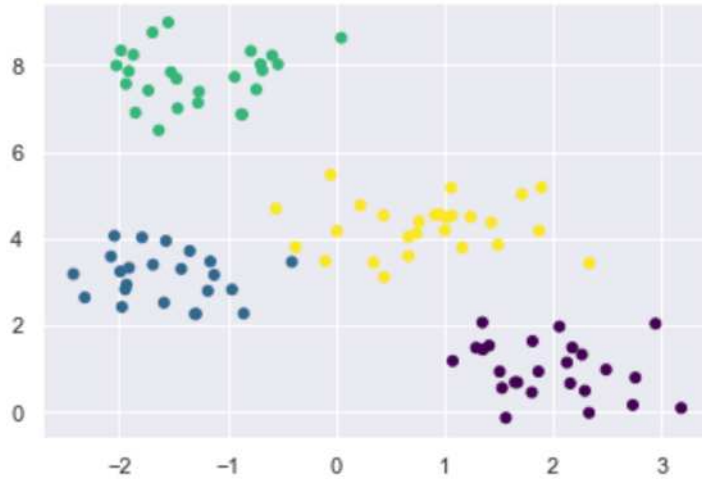
In [5]: `kmeans.fit(X)`
`y_kmeans = kmeans.predict(X)`
`y_kmeans`

Out[5]: `array([3, 2, 3, 3, 3, 3, 1, 2, 3, 2, 2, 2, 2, 2, 0, 0, 3, 1, 0, 3, 2,`
`1, 0, 3, 1, 3, 3, 0, 0, 0, 2, 0, 0, 1, 3, 2, 0, 2, 1, 3, 1, 2, 1,`
`1, 2, 0, 1, 3, 3, 3, 0, 1, 1, 1, 2, 2, 0, 0, 2, 2, 0, 0, 3, 0, 2,`
`1, 1, 0, 3, 2, 0, 3, 2, 3, 3, 1, 1, 0, 0, 0, 2, 1, 3, 0, 1, 0, 0,`
`3, 3, 3, 1, 3, 1, 1, 2, 2, 1, 2, 3])`

Cluster (၄)ခု ဖွဲ့ထားပြီးပုံကို ဂရပ်ဆွဲသည်။

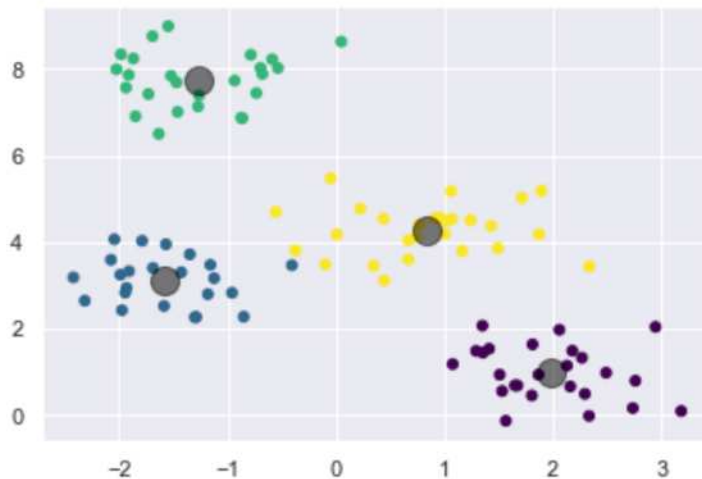
```
In [6]: plt.scatter(X[:, 0], X[:, 1], c=y_kmeans, s=25, cmap='viridis')
```

```
Out[6]: <matplotlib.collections.PathCollection at 0x52e516d68>
```



ထို ဂရပ်ထဲတွင် center or centroid များကို ပါထည့်သွင်း ဖော်ပြသည်။

```
In [7]: plt.scatter(X[:, 0], X[:, 1], c=y_kmeans, s=25, cmap='viridis')
centers = kmeans.cluster_centers_
plt.scatter(centers[:, 0], centers[:, 1], c='black', s=200, alpha=0.5);
```



Center or centroid များ တည်ရှိရာ နေရာကို ဖော်သည့် array ကို ကြည့်ရန်

```
In [8]: centers
```

```
Out[8]: array([[ 1.97918933,  0.97920012],
               [-1.5772186 ,  3.11456071],
               [-1.27208964,  7.74944718],
               [ 0.83044547,  4.27831711]])
```

Ref : Python Data Science Handbook by Jake VanderPlas