

مقدمه ای بر الگوریتم Mean-shift :

همانطور که پیش تر گفته شد، این یکی دیگر از الگوریتم های خوشه بندی قدرتمند است که در یادگیری بدون نظارت مورد استفاده قرار می گیرد. بر خلاف خوشه بندی K-means، از آنجایی که Mean-shift یک الگوریتم بدون پارامتر است، هیچ فرضیاتی در نظر نمی گیرد.

اساسا الگوریتم Mean-shift ، با انتقال نقاط به سمت نقاط داده با بیشترین تراکم (مانند نقطه مرکزی خوشه) ، نقاط داده را مکررا به خوشه ها تخصیص می دهد .

تفاوت بین الگوریتم های K-Means و Mean-Shift این است که ، Mean-Shift نیازی ندارد تا از قبل تعداد خوشه ها را مشخص کند، زیرا تعداد خوشه ها توسط الگوریتم داده w.r.t مشخص می شود.

عملکرد الگوریتم Mean-Shift: با کمک مراحل زیر می توان نحوه عملکرد الگوریتم Mean-Shift را فهمید.

مرحله ۱- ابتدا، با نقاط داده تخصیص یافته به خوشه خود، شروع کنید.

مرحله ۲- سپس، این الگوریتم نقاط مرکزی را محاسبه خواهد کرد.

مرحله ۳- در این مرحله، مکان نقاط مرکزی جدید به روز رسانی خواهد شد.

مرحله ۴- حال، پردازش تکرار خواهد شد و به ناحیه با تراکم بالاتر منتقل خواهد شد.

مرحله ۵- در انتها، زمانی که نقاط مرکزی به مکانی برسند که نتوانند پیش بروند، متوقف خواهند شد.

پیاده سازی در پایتون: این یک مثال ساده برای فهم نحوه عملکرد الگوریتم Mean-Shift است. در این

مثال، ابتدا یک مجموعه داده 2D شامل ۴ blob خواهیم ساخت. سپس الگوریتم Mean-Shift را اعمال

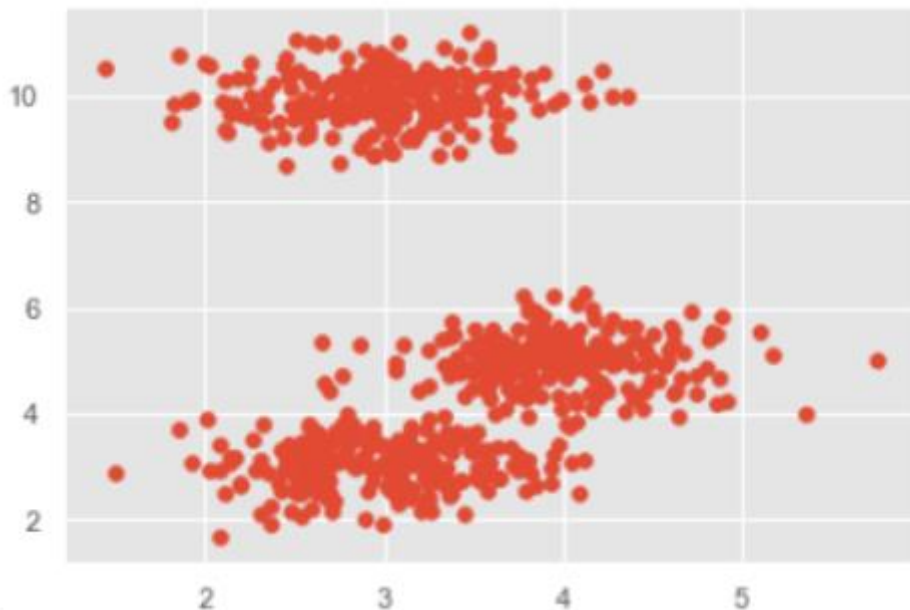
خواهیم کرد تا نتیجه را ببینیم.

```
%matplotlib inline
import numpy as np
```

```

from sklearn.cluster import MeanShift
import matplotlib.pyplot as plt
from matplotlib import style
style.use("ggplot")
from sklearn.datasets.samples_generator import
make_blobs
centers = [[3,3,3],[4,5,5],[3,10,10]]
X, _ = make_blobs(n_samples = 700, centers = centers,
cluster_std = 0.5)
plt.scatter(X[:,0],X[:,1])
plt.show()

```



```

ms = MeanShift()
ms.fit(X)
labels = ms.labels_
cluster_centers = ms.cluster_centers_
print(cluster_centers)
n_clusters_ = len(np.unique(labels))
print("Estimated clusters:", n_clusters_)
colors = 10*['r.','g.','b.','c.','k.','y.','m.']

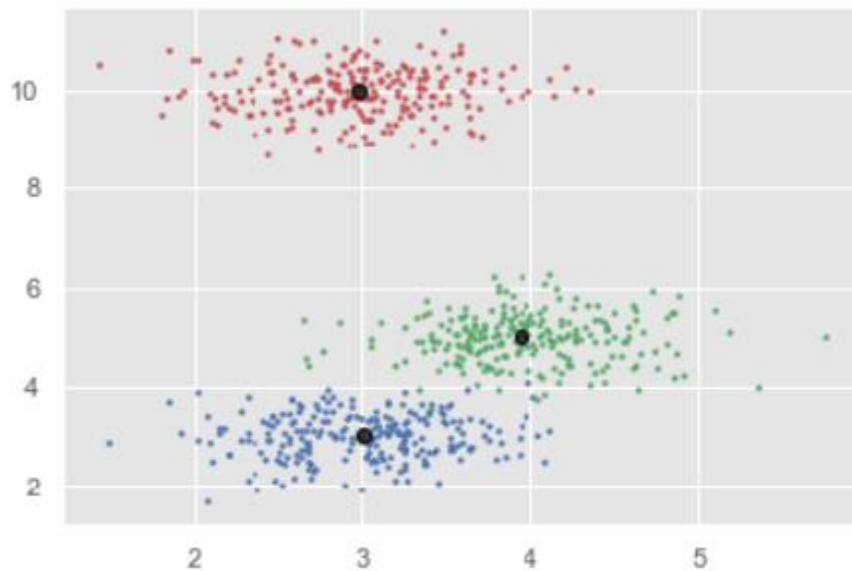
for i in range(len(X)):
    plt.plot(X[i][0], X[i][1], colors[labels[i]],
markersize = 3)
plt.scatter(cluster_centers[:,0],cluster_centers[:,1]
,
marker = ".",color = 'k', s = 20, linewidths = 5,
zorder = 10)

```

```
plt.show()
```

خروجی:

```
[[ 2.98462798  9.9733794 10.02629344]  
 [ 3.94758484  4.99122771  4.99349433]  
 [ 3.00788996  3.03851268  2.99183033]]  
Estimated clusters: 3
```



مزایا و معایب:

مزایا: موارد زیر برخی از مزایای الگوریتم خوشه بندی Mean-Shift است. این الگوریتم برخلاف k-means یا ماتریس گوسی، نیازی به فرض مدل ندارد. همچنین می تواند خوشه های پیچیده که شکل غیر محدب (nonconvex) دارند را مدل کند. تنها نیاز به یک پارامتر به نام پهنای باند (bandwidth) دارد که به طور خودکار تعداد خوشه ها را مشخص می کند. مشکل کمینه محلی موجود در K-means، در این الگوریتم وجود ندارد. هیچ مشکلی از داده های خارج از محدوده به وجود نیامده است.

معایب: موارد زیر برخی از معایب الگوریتم خوشه بندی Mean-Shift است. الگوریتم Mean-shift در ابعاد بزرگ، جایی که تعداد خوشه ها ناگهانی تغییر می کند، به خوبی کار نمی کند. ما هیچ کنترل مستقیمی روی تعداد خوشه ها نداریم، اما در برخی از برنامه های کاربردی به تعداد معینی از خوشه ها نیاز داریم. این الگوریتم نمی تواند بین حالات معنا دار و بدون معنا تفاوت قائل شود.