

파이썬 기반  
빅데이터 처리  
및 분석 기술





8-3 가우스 혼합 모델



대구가톨릭대학교  
사물인터넷(IoT)과 함께하는 빅데이터

## 빅데이터 군집 분석

3

심화: 가우스 혼합 모델

가우스 혼합 모델 등장 배경

가우스 혼합 모델 실습





## 1. 가우스 혼합 모델

### 1) 등장 배경

#### ◆ K-평균 알고리즘의 약점

#### K-평균 알고리즘의 장점

- 사용이 편리
- 직관적인 개념

#### K-평균 알고리즘의 단점

- 유연성 부족:  
→ 군집 중심 원형 경계를 사용한 군집 할당
- 확정적 군집 할당  
→ 비확률적인 군집 할당 기준







1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 표준 패키지 불러오기

```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns; sns.set()
```





1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석

```
from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=400, centers=4, cluster_std=0.60, random_state=0)
X = X[:, ::-1]
plt.scatter(X[:, 0], X[:, 1], c='gray', alpha=0.4);
```





1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석

```
from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=400, centers=4, cluster_std=0.60, random_state=0)
X = X[:, ::-1]
plt.scatter(X[:, 0], X[:, 1], c='gray', alpha=0.4);
```

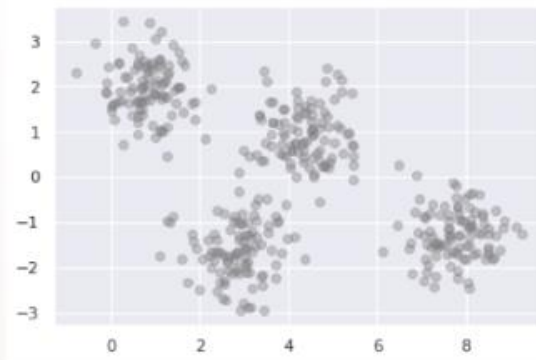


1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석

```
from sklearn.datasets.samples_generator import make_blobs  
X, y_true = make_blobs(n_samples=400, centers=4, cluster_std=0.60, random_state=0)  
X = X[:, ::-1]  
plt.scatter(X[:, 0], X[:, 1], c='gray', alpha=0.4);
```







1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석

```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=4, random_state=0)
clusters = kmeans.fit(X).predict(X)
plt.scatter(X[:, 0], X[:, 1], c=clusters, s=40, cmap='viridis')
plt.scatter(centers[:, 0], centers[:, 1], c='red');
```







1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석

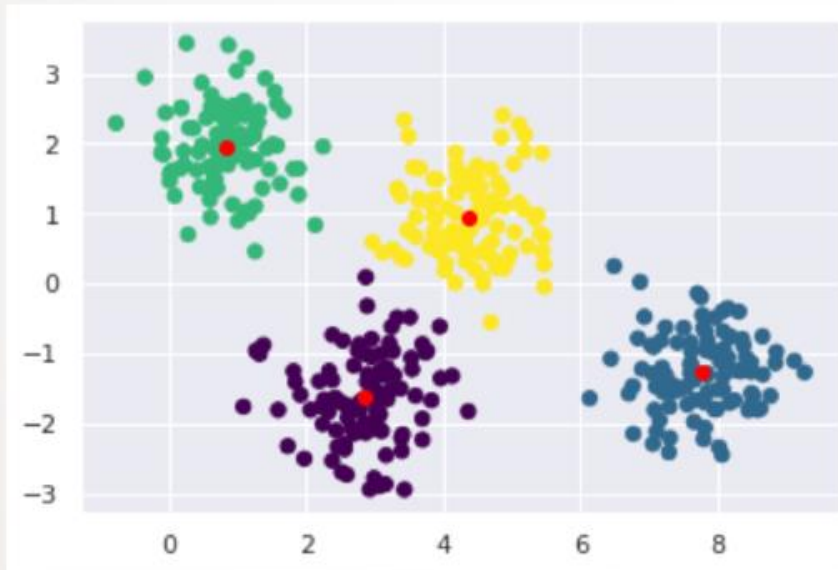
```
from sklearn.cluster import KMeans  
kmeans = KMeans(n_clusters=4, random_state=0)  
clusters = kmeans.fit(X).predict(X)  
plt.scatter(X[:, 0], X[:, 1], c=clusters, s=40, cmap='viridis')  
plt.scatter(centers[:, 0], centers[:, 1], c='red');
```



1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석 결과

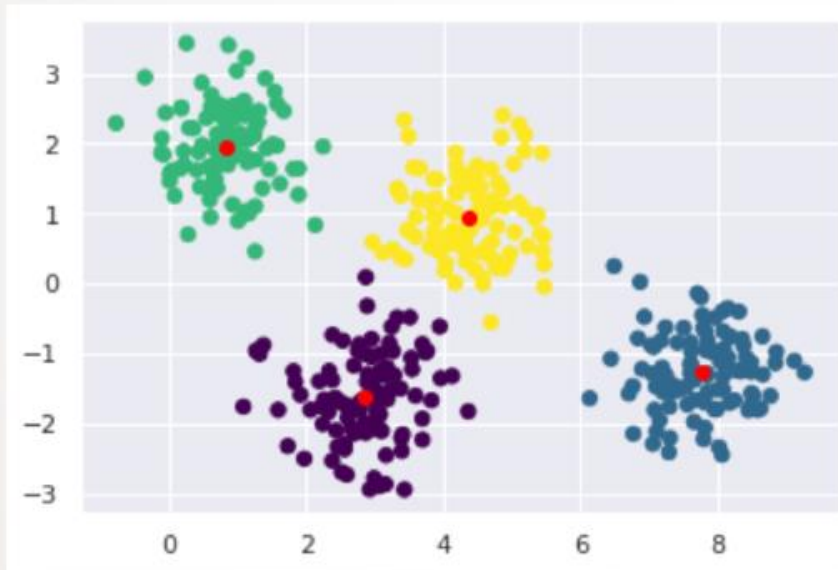


군집의 중심에 가까운 점일수록 군집 할당이 명확

1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ 정상적인 K-평균 군집 분석 결과



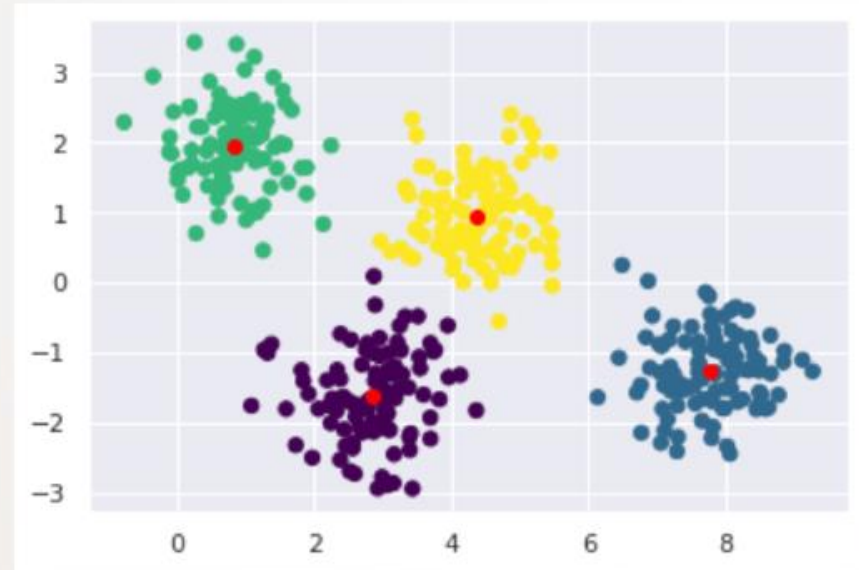
K-평균 모델에는 군집 할당의 확률, 불확실성에 대한 기준이 없음



1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ K-평균 군집의 원형 경계

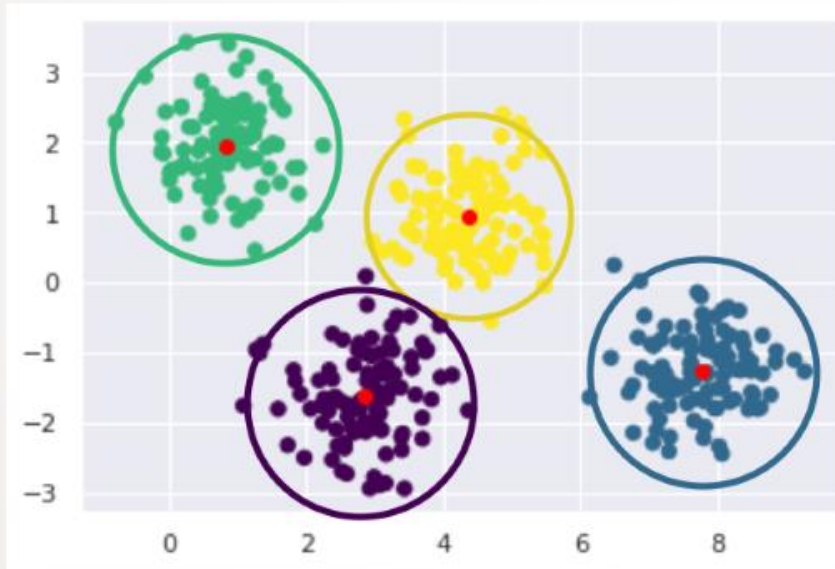




1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ K-평균 군집의 원형 경계

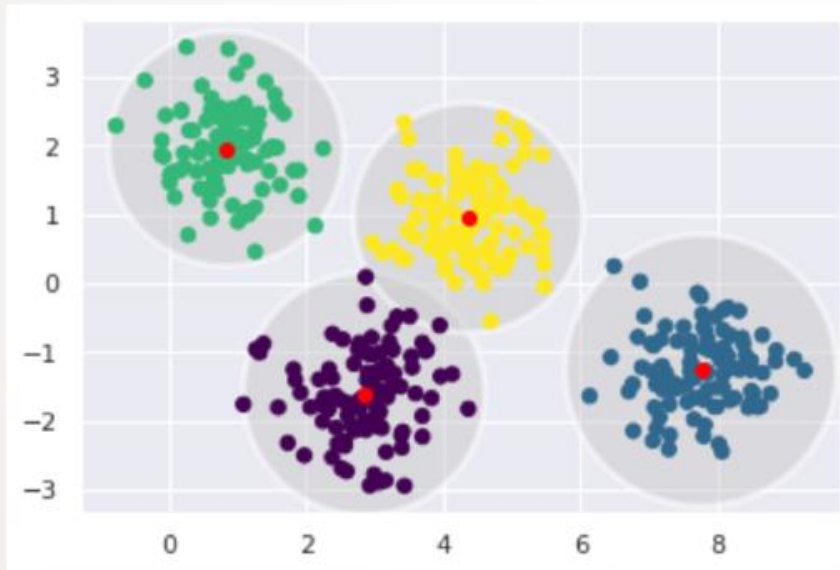




1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

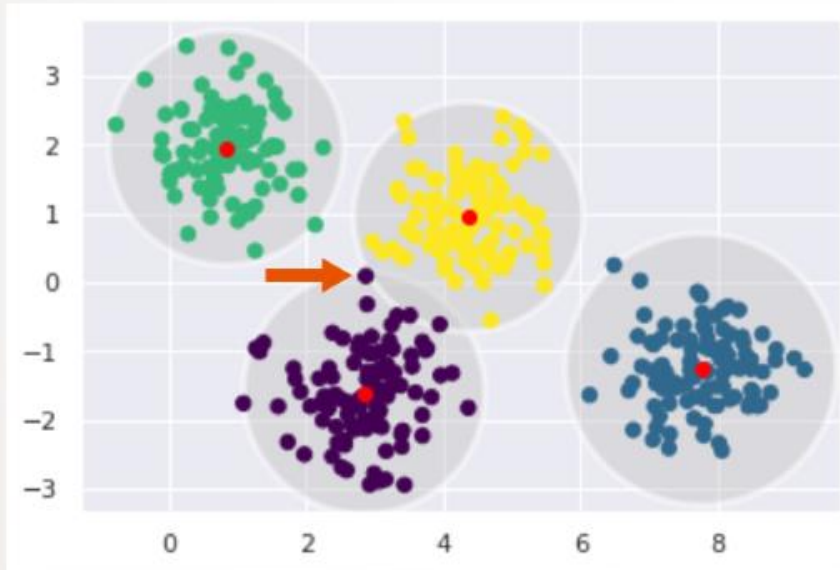
◆ K-평균 군집의 원형 경계



1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ K-평균 군집의 원형 경계

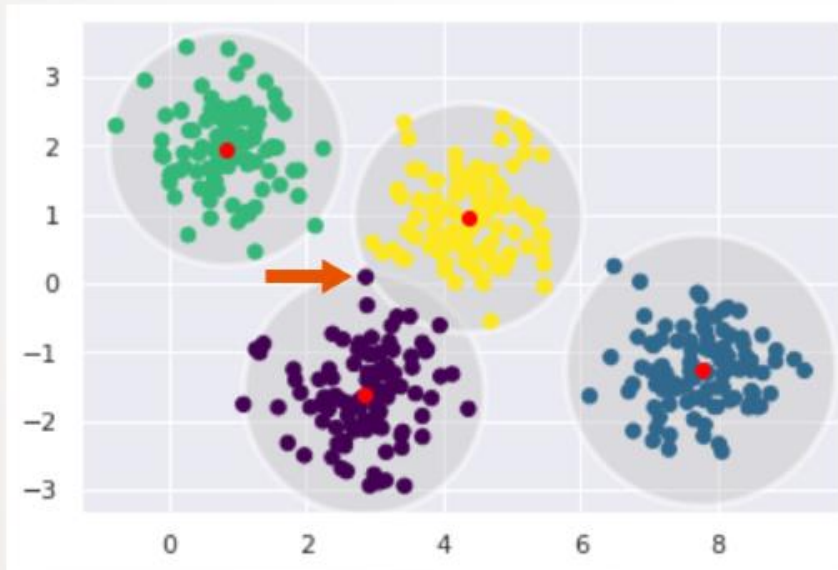




1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ K-평균 군집의 원형 경계



단점 1. 확정적 군집 할당 → 비확률적인 군집 할당 기준

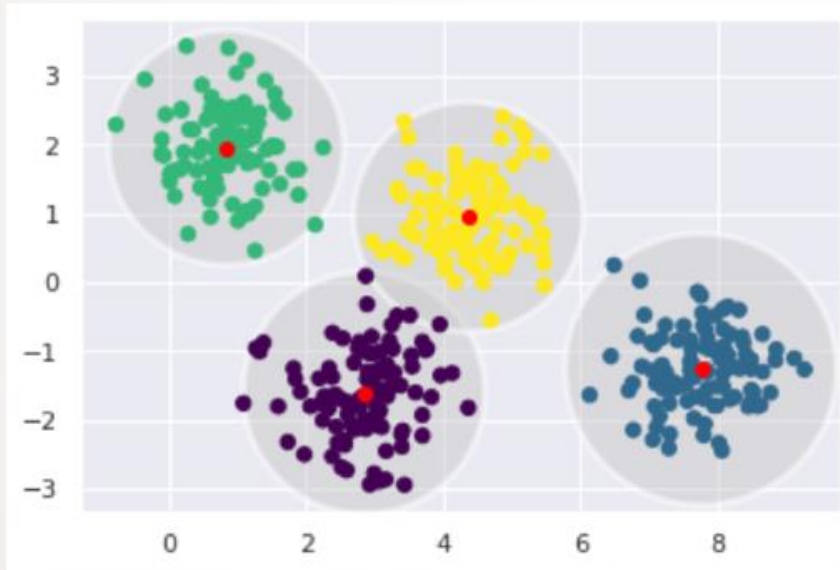




1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

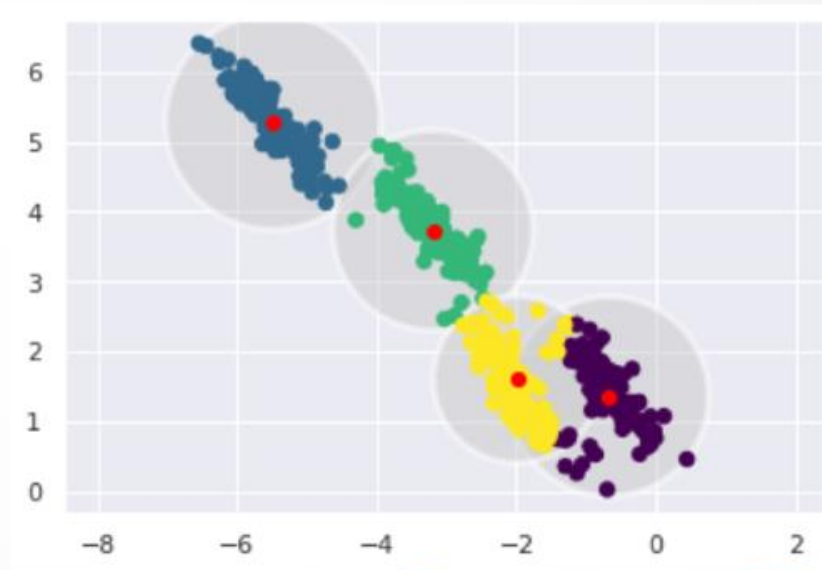
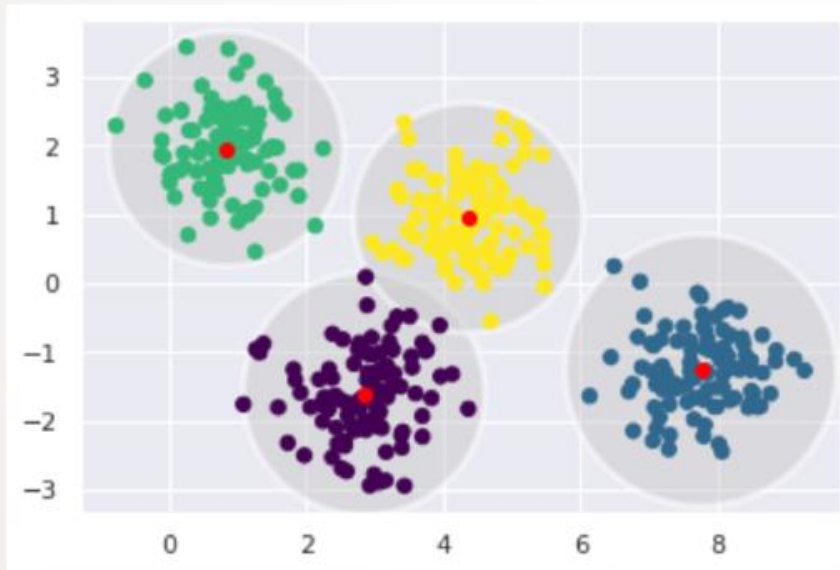
◆ K-평균 군집의 잘못된 결과



1. 가우스 혼합 모델

2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

◆ K-평균 군집의 잘못된 결과



단점 2. 유연성 부족 → 군집 중심 원형 경계를 사용한 군집 할당



1. 가우스 혼합 모델



## 2) 정상적인 적용 결과 Vs. 비정상적인 적용 결과

### ◆ K-평균 군집 분석 단점 개선

- 각 점과 모든 군집까지의 거리를 비교하여 확률적으로 군집 할당의 가능성 측정
- 원형 대신 타원형을 군집 경계로 허용

가우스  
혼합 모델





1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

```
from sklearn.mixture import GaussianMixture
gmm = GaussianMixture(n_components=4).fit(X)
labels = gmm.predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels, s=40, cmap='viridis') c='red');
```







1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

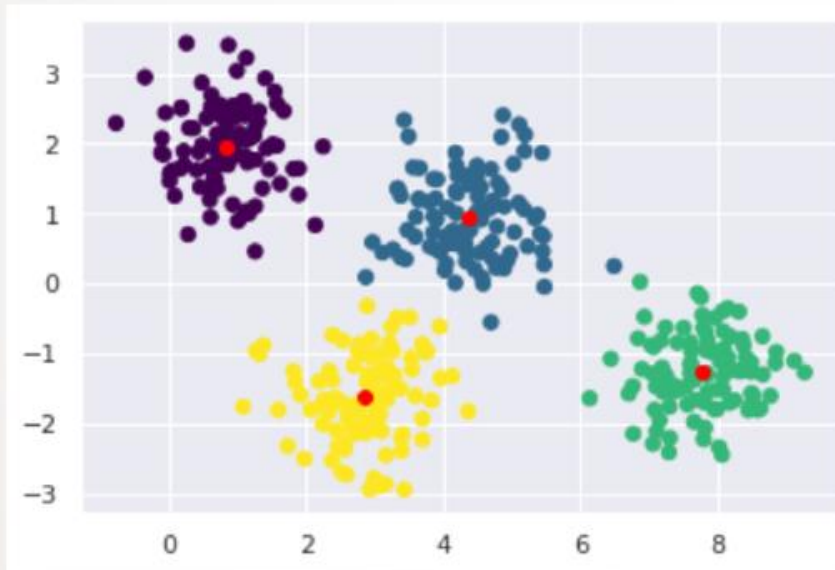
```
from sklearn.mixture import GaussianMixture
gmm = GaussianMixture(n_components=4).fit(X)
labels = gmm.predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels, s=40, cmap='viridis') c='red');
```



1. 가우스 혼합 모델

3) 가우스 혼합 모델 사용법

◆ 결과



확률적 군집 할당 기능 표시 전에는 K-평균 군집 분석 결과와 동일



1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 확률적 군집 할당

- `probs = gmm.predict_proba(X)`  
`print(probs[:5].round(3))`





1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 확률적 군집 할당

- `probs = gmm.predict_proba(X)`  
`print(probs[:5].round(3))`

[	[	0.	0.531	0.469	0.	]
	[	0.	0.	0.	1.	]
	[	0.	0.	0.	1.	]
	[	0.	1.	0.	0.	]
	[	0.	0.	0.	1.	]]





1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 확률적 군집 할당

```
size = 50 * probs.max(1) ** 2  
plt.scatter(X[:, 0], X[:, 1], c=labels, cmap='viridis', s=size)  
plt.scatter(centers[:, 0], centers[:, 1], c='red');
```



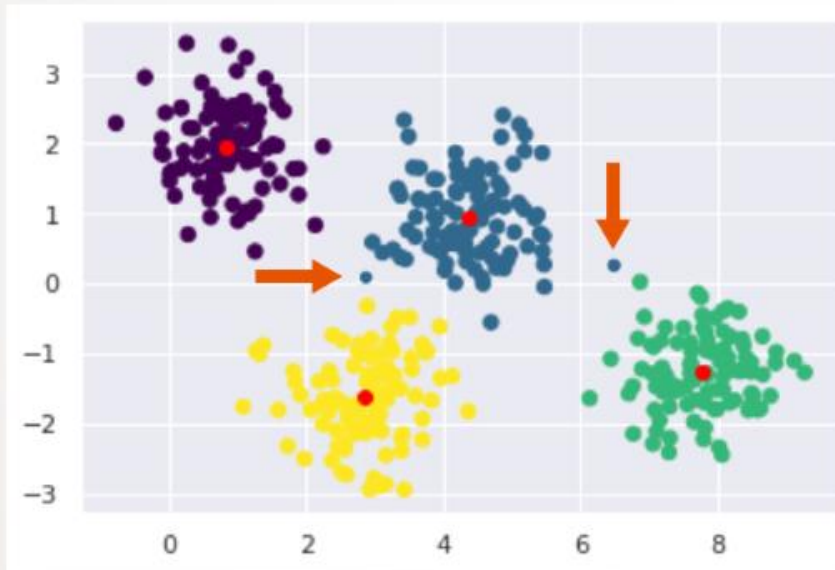


1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 확률적 군집 할당



할당된 군집이 잘못되었을 수도 있으므로 주의



1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 타원형 군집 할당

```
from sklearn.mixture import GaussianMixture  
gmm = GaussianMixture(n_components=4, covariance_type='full', random_state=42)  
plot_gmm(gmm, X_new)
```



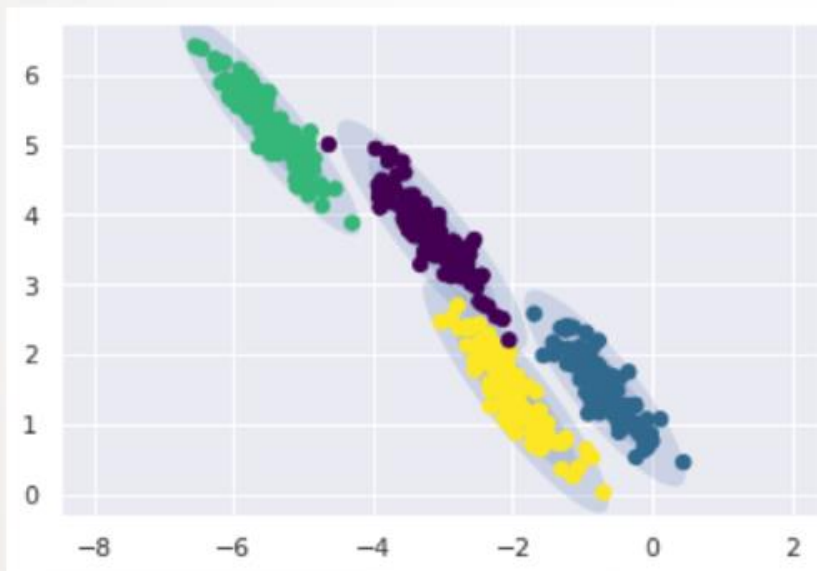


1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 타원형 군집 할당





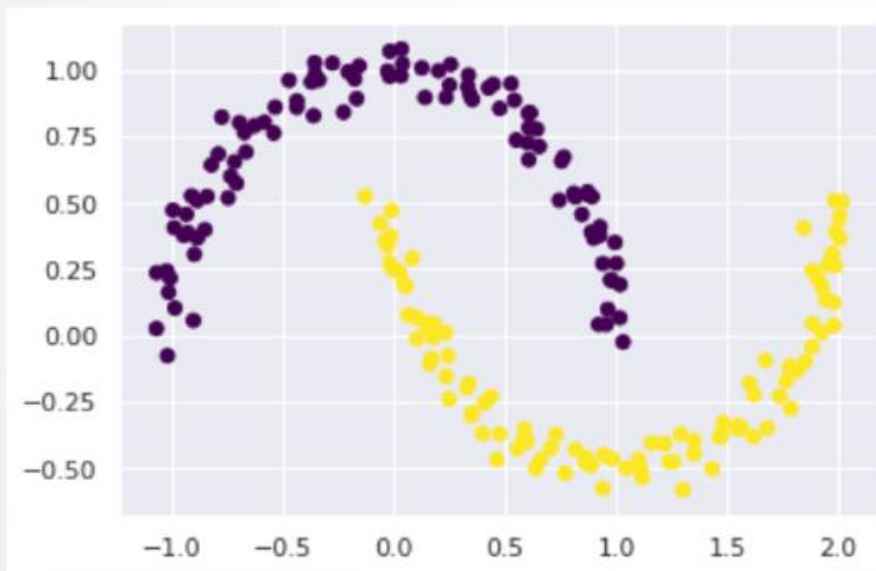


1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 복잡한 군집 할당



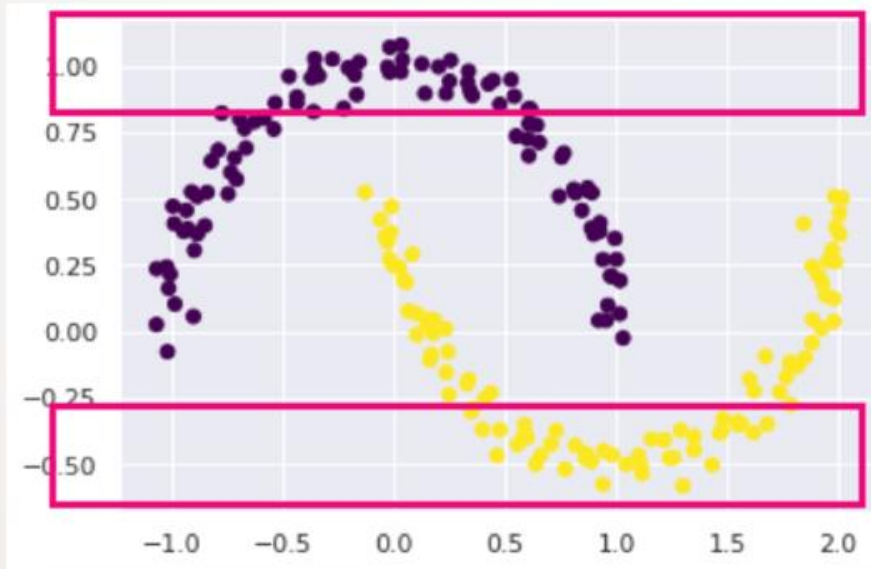


1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 복잡한 군집 할당





1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 복잡한 군집 할당

```
from sklearn.mixture import GaussianMixture  
gmm = GaussianMixture(n_components=16, covariance_type='full', random_state=0)  
plot_gmm(gmm, X_new)
```

16개의 가우스 모델을 혼합하여 군집을 할당하라!

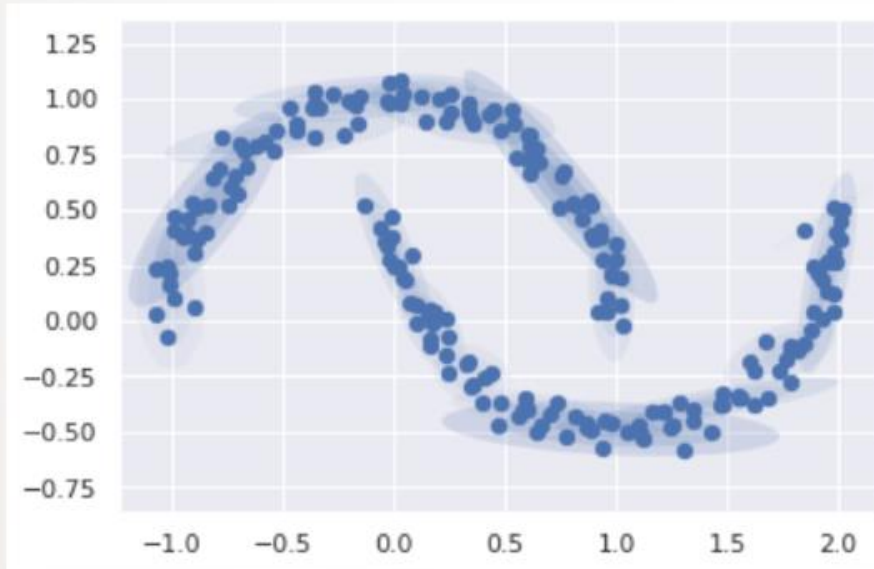


1. 가우스 혼합 모델



### 3) 가우스 혼합 모델 사용법

◆ 복잡한 군집 할당



분리된 데이터 군집 발견 X → 입력 데이터의 일부분을 파편적으로 군집 할당







## 이번 시간에는

3

심화: 가우스 혼합 모델

가우스 혼합 모델 등장 배경

가우스 혼합 모델 실습





## 이번 시간에는

3

심화: 가우스 혼합 모델

## 머신러닝

지도 학습

회귀 분석  
분류 분석

비지도 학습

군집 분석

Scikit-Learn 패키지 사용





## 이번 시간에는

### 실습 참고 자료

- Colab 노트북 파일
- Scikit-Learn 공식 사이트 자료  
- [https://scikit-learn.org/stable/user\\_guide.html](https://scikit-learn.org/stable/user_guide.html)





## 이번 시간에는

### 과제 안내

- 과 제 : 퀴즈
- 제출 방법 : 과제 게시판 제출 방법 안내 참조

### 질의 응답 게시판

- 학습 내용, 퀴즈, 과제 등에 대한 질의응답 게시판을 통한 질의응답

