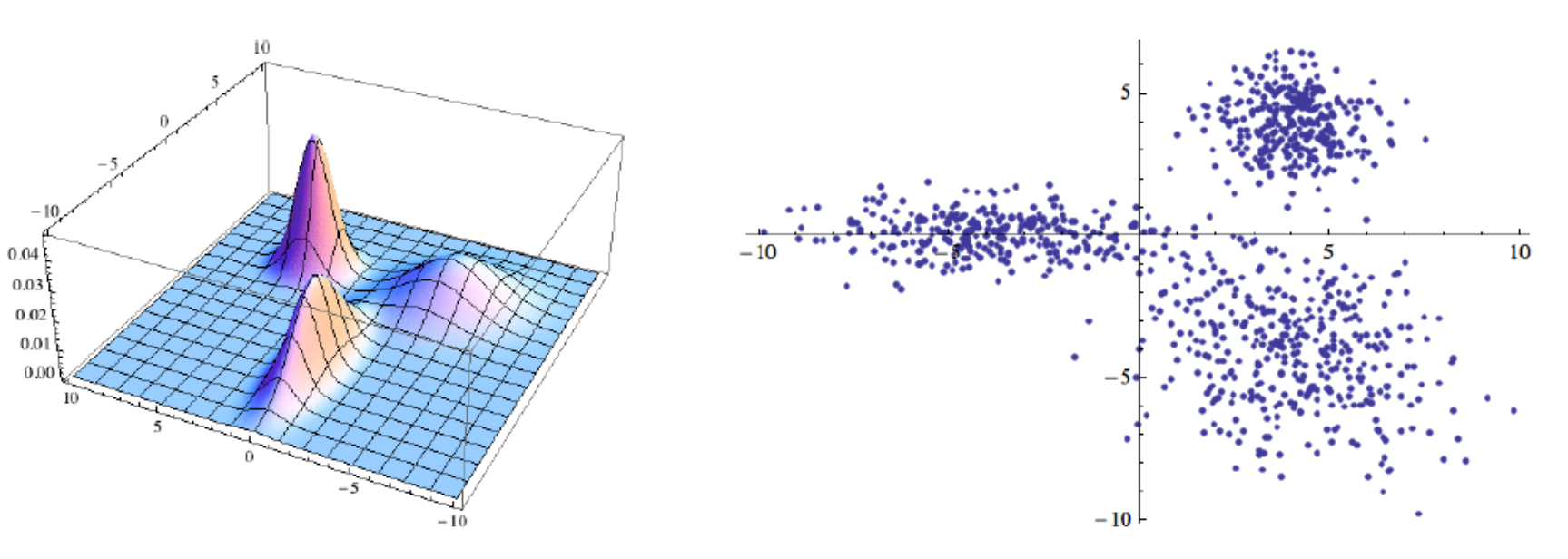
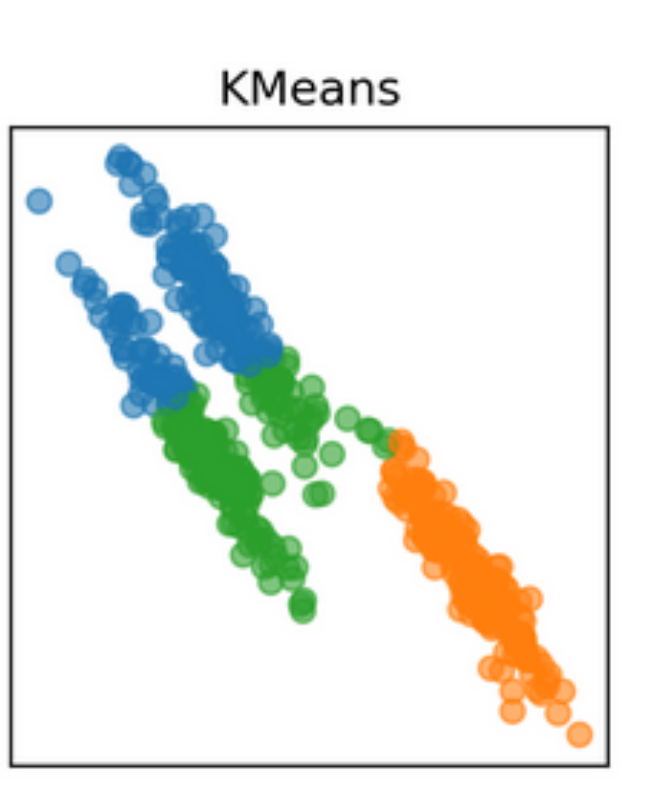
# Gaussian Mixture

VanThiep - August 12, 2020 - Uncategorized - 0 Comments



(a) A probability distribution on $\mathbb{R}^2$. (b) Data sampled from this distribution.

Như chúng ta đã biết, mặc dù Kmeans là một thuật toán mạnh mẽ, áp dụng được trong khá nhiều trường hợp. Tuy nhiên, nó vẫn có một vài hạn chế nhất định. Một trong số đó là Kmeans thường chỉ phù hợp với cluster có dạng hình tròn. Vậy nếu cluster không có dạng hình tròn thì sao?

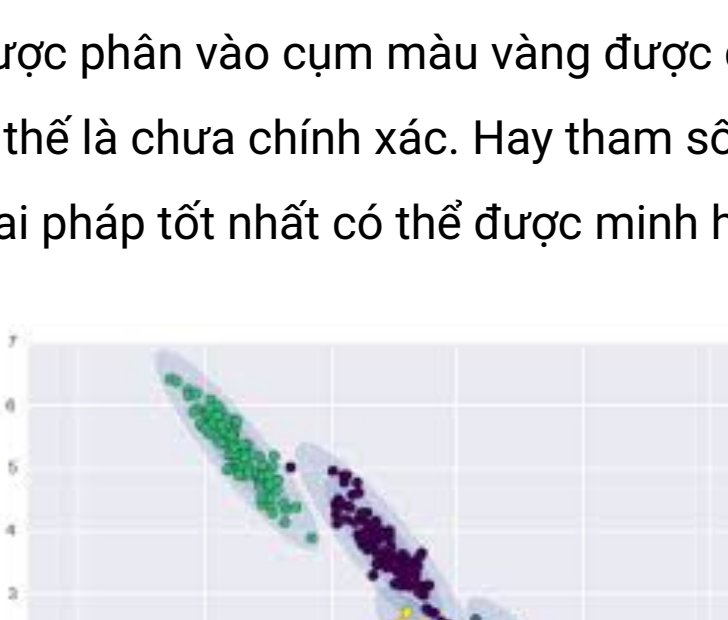


Như hình trên có thể thấy, cluster có dạng hình elip dẹt và khi sử dụng Kmeans để phân cụm thì cho kết quả khá tệ. Lúc này thuật toán Gaussian Mixture sẽ phát huy tác dụng của nó. Nếu Kmeans là thuật toán phân cụm dựa trên khoảng cách thì Gaussian Mixture lại dựa trên phân phối xác suất. Thực chất, Kmeans chỉ là một trường hợp cụ thể của Gaussian Mixture. Thuật toán Kmeans chỉ cập nhật giá trị trung bình. Trong khi đó, Gaussian Mixture cập nhật cả trung bình và phương sai. Tiếp theo ta sẽ cùng tìm hiểu kĩ hơn về thuật toán này.

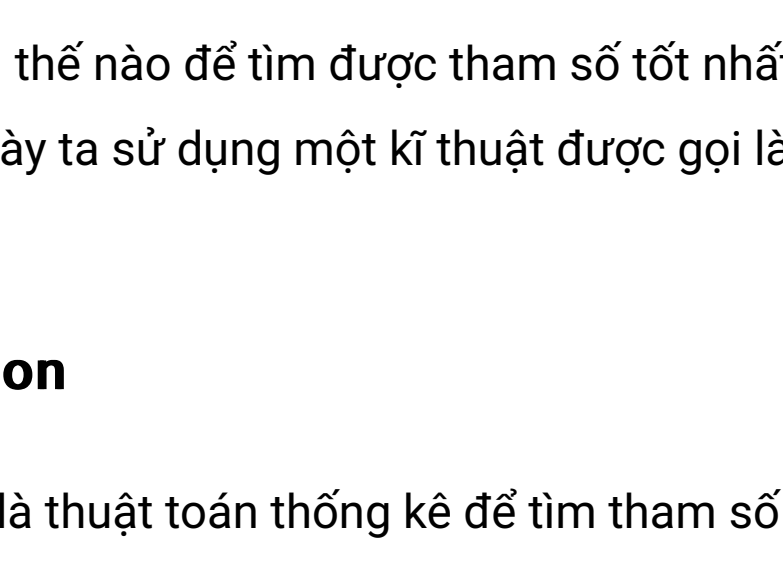
## Gaussian Mixture Models (GMMs)

Xét dưới góc độ xác suất, ta có thể coi một tập dữ liệu là một biến ngẫu nhiên. Mà bất kì một biến ngẫu nhiên nào cũng có thể được mô tả bởi một phân phối xác suất. Hàm phân phối xác suất biểu diễn mối quan hệ giữa biến cố với xác suất xảy ra biến cố đó. Do đó, ta có thể nói một tập dữ liệu X được tạo ra từ phân phối xác suất nào đó.

Gaussian mixture là mô hình xác suất với giả định rằng tất cả các điểm dữ liệu được tạo ra từ một tập hữu hạn k các phân phối xác suất chuẩn( Gaussian distribution- GD ) chưa biết tham số. Thuật toán này coi mỗi cụm là một phân phối xác suất chuẩn, k GD tương ứng với k cụm. Đối với một tập dữ liệu cụ thể, Gaussian mixture model sẽ xác định xác suất điểm dữ liệu thuộc một phân phối xác suất cụ thể. Để xác định xác suất này ta cần biết tham số của các phân phối xác suất. Như vậy, mục tiêu của GMM là xác định tham số của các phân phối xác suất. Ta cùng xem xét ví dụ sau



Như trên hình vẽ có thể thấy, $x_i$ thuộc $GD_3(x|\mu_3, \sigma_3)$ và $GD_4((x|\mu_4, \sigma_4)$. Nhưng xác suất $x_i$ thuộc $GD_3$ cao hơn nên nó được phân vào cụm màu vàng được đại diện bởi $GD_3$. Tuy nhiên ta có thể thấy việc phân loại như thế là chưa chính xác. Hay tham số $(\mu_3, \sigma_3)$, $(\mu_4, \sigma_4)$ của $GD_3$, $GD_4$ chưa phải là tốt nhất. Giai pháp tốt nhất có thể được minh họa bởi hình bên dưới.



Ngoài ra cần lưu ý, đối với phân phối xác suất chuẩn của dữ liệu một chiều thì tham số là trung bình và phương sai. Tuy nhiên, nếu là của dữ liệu nhiều chiều thì tham số là vecto trung bình và ma trận hiệp phương sai.

Câu hỏi đặt ra tiếp theo là làm thế nào để tìm được tham số tốt nhất của các phân phối xác suất chuẩn? Để giải quyết vấn đề này ta sử dụng một kĩ thuật được gọi là **Expectation-Maximization**

## Expectation-Maximization

**Expectation-Maximization** là thuật toán thống kê để tìm tham số của phân phối xác suất. Kĩ thuật này thường được sử dụng khi dữ liệu có missing values hay nói cách khác dữ liệu là không đầy đủ. Trong bài viết này, missing value có thể được hiểu là biến mục tiêu hay cụm. Vì ta không biết điểm dữ liệu sẽ thuộc cụm nào.

Thuật toán này bao gồm 2 bước như sau:

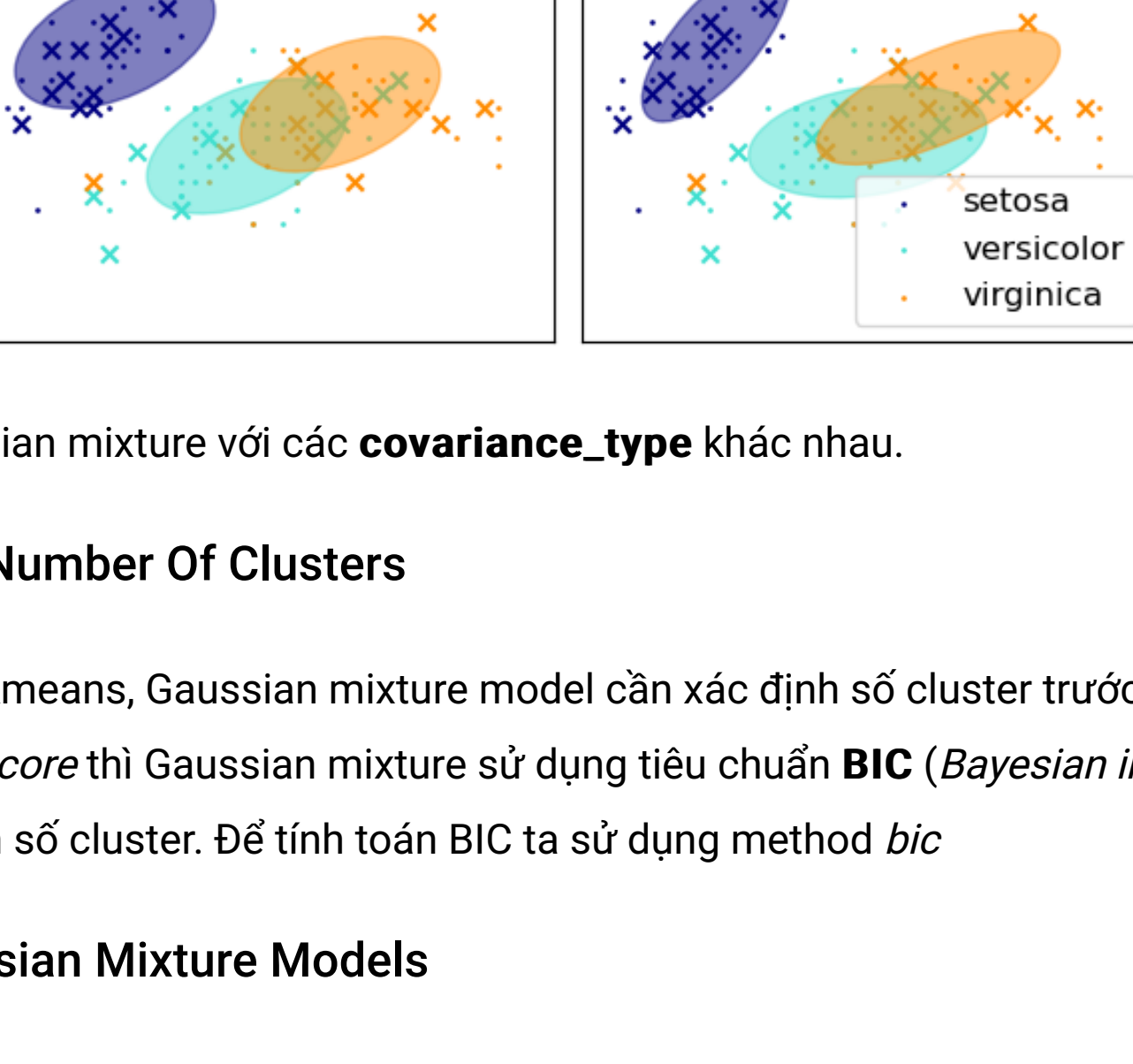
- E-step: ban đầu thuật toán sẽ khởi tạo các tham số một cách ngẫu nhiên cho các phân phối xác suất. Sau đó sử dụng các tham số này để tính xác suất điểm dữ liệu thuộc một phân phối xác suất cụ thể. Qua đó có thể gán được từng điểm dữ liệu vào một phân phối xác suất cụ thể.
- M-step: sử dụng các điểm dữ liệu đã được gán để cập nhật tham số của các phân phối xác suất

Các bước trên cũng tương tự như các bước tìm centroids trong thuật toán Kmeans vây. Tuy nhiên ở bước cập nhật tham số, nếu Kmeans chỉ đơn giản là thay centroids cũ bằng trung bình của các sample trong cluster thì **Expectation-Maximization** sử dụng kĩ thuật được gọi là *Maximum lilklihood estimation*

### Drawbacks Of Gaussian Mixture

Một trong những hạn chế của Gaussian Mixture model là nếu mô hình không có đủ nhiều dữ liệu trong mỗi mixture( GD/ cluster) thì ước lượng ma trận hiệp phương sai sẽ trở nên khó khăn. Dấn đến mô hình không tìm được giải pháp tối ưu. Ngoài ra khi dữ liệu có nhiều chiều hoặc nhiều cluster thì cũng ảnh hưởng tới việc expectation-maximization tìm giải pháp tơi ưu. Chúng ta có thể giảm thiểu khó khăn này bằng việc giới hạn số tham số cần học. Một cách để thực hiện điều đó là giới hạn hình dáng và hướng mà các cluster có thể có, hay nói cách khác là giời hạn ma trận hiệp phương sai. Để thực hiện điều đó ta thiết lập tham số **covariance_type** là một trong các giá trị sau:

- “spherical”: Tất cả các cluster phải là hình cầu, nhưng chúng có thể có đường kính, kích thước khác nhau
- “diag”: clusters có thể là hình elip với bất kì kích thước nào. Tuy nhiên trục của hình elip phải song song với hệ trục tọa độ( ví dụ: ma trận hiệp phương sai phải là ma trận đường chéo)
- “tied”: Các clusters là các hình elip có cùng kích thước, hình dáng và hướng ( các cluster có cùng ma trận hiệp phương sai)
- “full”- được set mạc định: mỗi cluster có thể có bất kì hình dáng, kích thước và phương hướng nào. Hay nói cách khác ma trân hiệp phương sai sẽ không bị giới hạn.



Bên trên là Gaussian mixture với các **covariance_type** khác nhau.

### Selecting The Number Of Clusters

Cũng giống như kmeans, Gaussian mixture model cần xác định số cluster trước. Nếu Kmeans sử dụng *silhouette score* thì Gaussian mixture sử dụng tiêu chuẩn **BIC** (*Bayesian information criterion*) để chọn số cluster. Để tính toán BIC ta sử dụng method *bic*

### Bayesian Gaussian Mixture Models

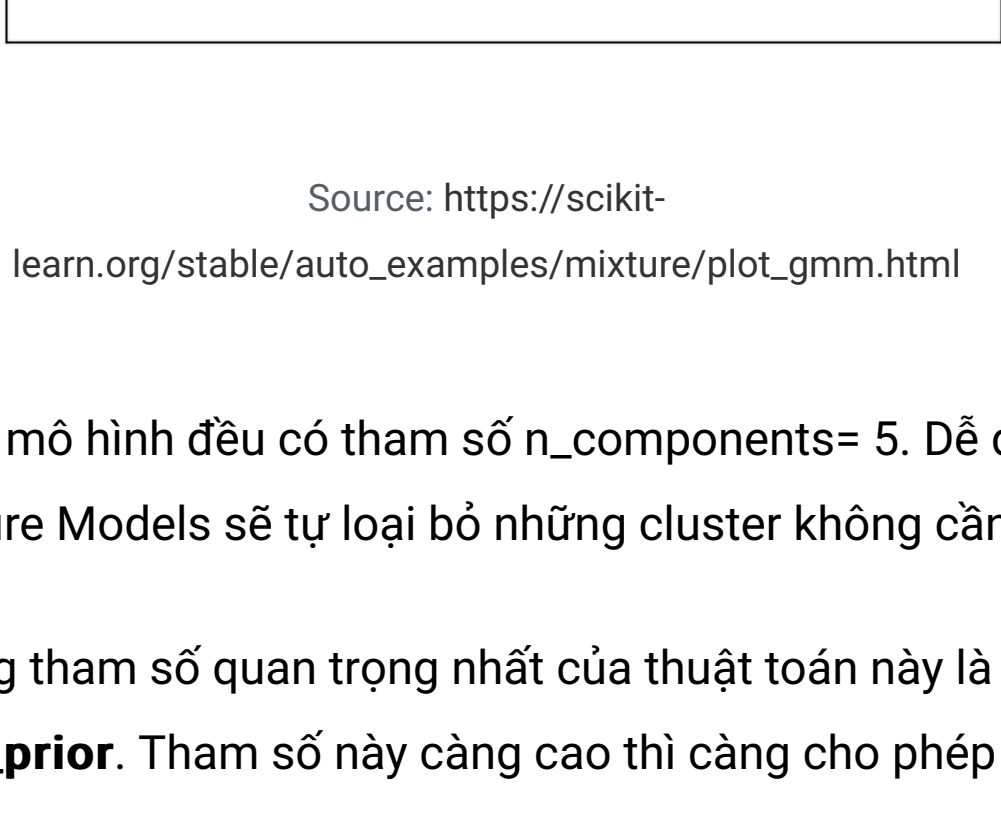
Thay vì việc tìm kiếm số cluster tối ưu một cách thủ công , ta đơn giản có thể sử dụng **BayesianGaussianMixture** model. Model này có thể đưa trọng số bằng 0 hoặc gần bằng 0 đối với các cluster không cần thiết. Ta chỉ cần đặt tham số **n_components** là một số mà ta tin rằng số đó lớn hơn số cluster tối ưu. Sau đó thuật toán sẽ tự động loại bỏ các cluster không cần thiết. Ví dụ:

```
from sklearn.mixture import BayesianGaussianMixture
bgm = BayesianGaussianMixture(n_components=10, n_init=10, random_s
tate=42)

bgm.fit(X)

np.round(bgm.weights_, 2)
array([0.4 , 0.21, 0.4 , 0. , 0. , 0. , 0. , 0. , 0. , 0. ])
```

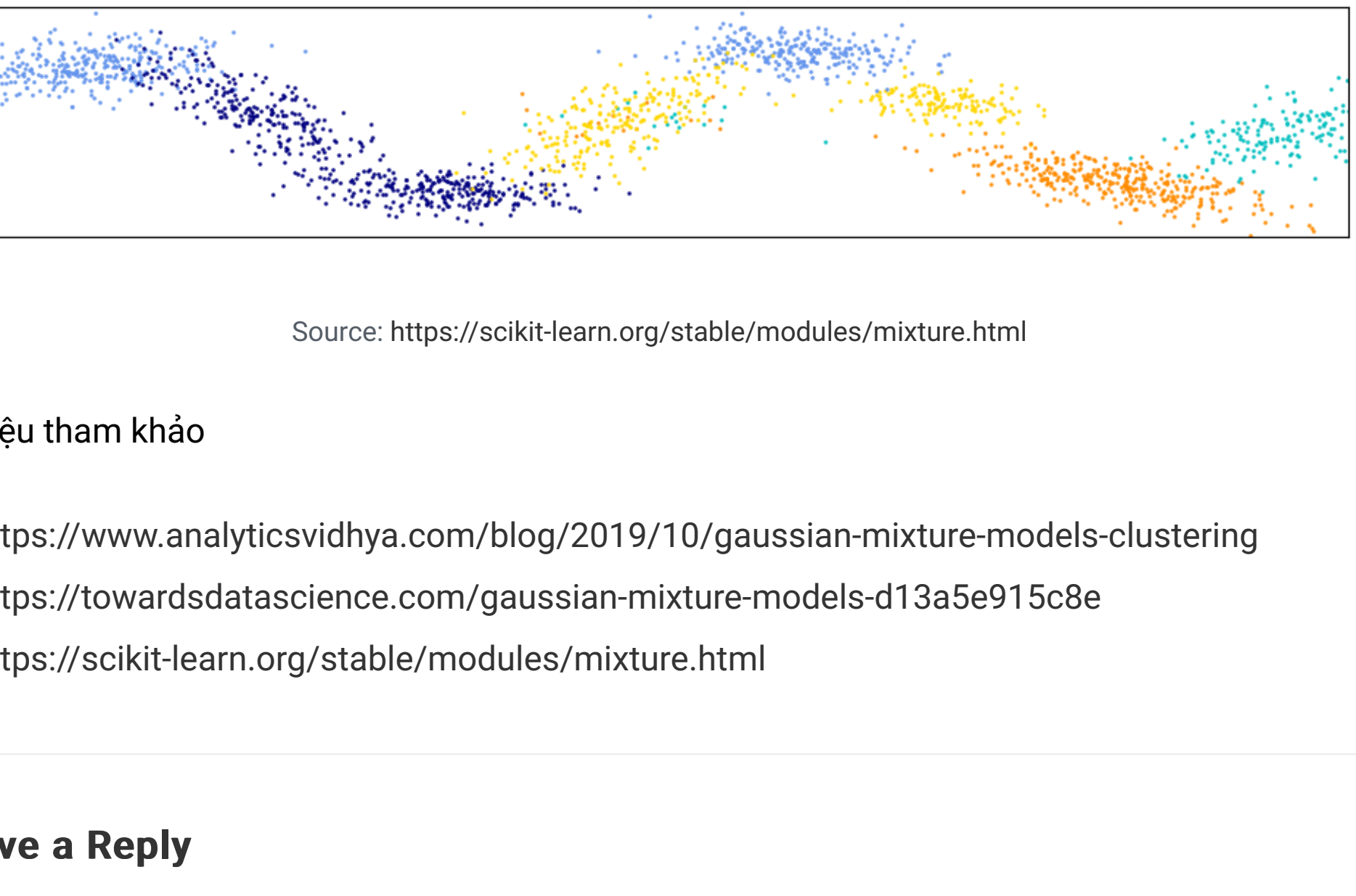
Ta cùng xem xét một ví dụ khác



Source: https://scikit-learn.org/stable/auto_examples/mixture/plot_gmm.html

Bên trên là kết quả của 2 mô hình đều có tham số n_components= 5. Dễ dàng nhận thấy, Bayesian Gaussian Mixture Models sẽ tự loại bỏ những cluster không cần thiết.

Ngoài ra một trong những tham số quan trọng nhất của thuật toán này là **weight_concentration_prior**. Tham số này càng cao thì càng cho phép nhiều cluster active. Hay nói cách khác, giá trị tham số này càng lớn thì dữ liệu được phân vào nhiều cụm hơn. Bên dưới là kết quả của Bayesian Gaussian Mixture Models với các giá trị weight_concentration_prior khác nhau.



Source: https://scikit-learn.org/stable/modules/mixture.html

Tài liệu tham khảo

- https://www.analyticsvidhya.com/blog/2019/10/gaussian-mixture-models-clustering
- https://towardsdatascience.com/gaussian-mixture-models-d13a5e915c8e
- https://scikit-learn.org/stable/modules/mixture.html

## Leave a Reply

Your comment here...

Name (required) Email (required) Website

☐ Save my name, email, and website in this browser for the next time I comment.

POST COMMENT

Search

Recent Posts

- XGBoost with math formulation
- Ensemble learning and random forest
- Gaussian mixture
- K-mean clustering and DBSCAN
- Support vector machines

Recent Comments

- torrent on Support vector machines