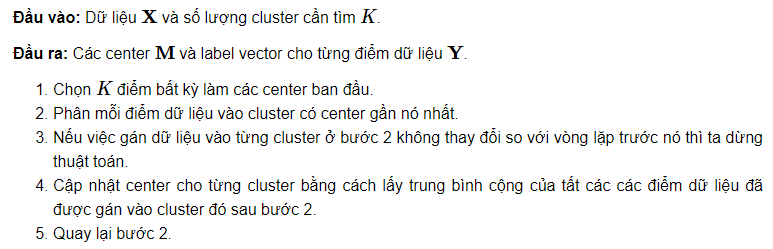
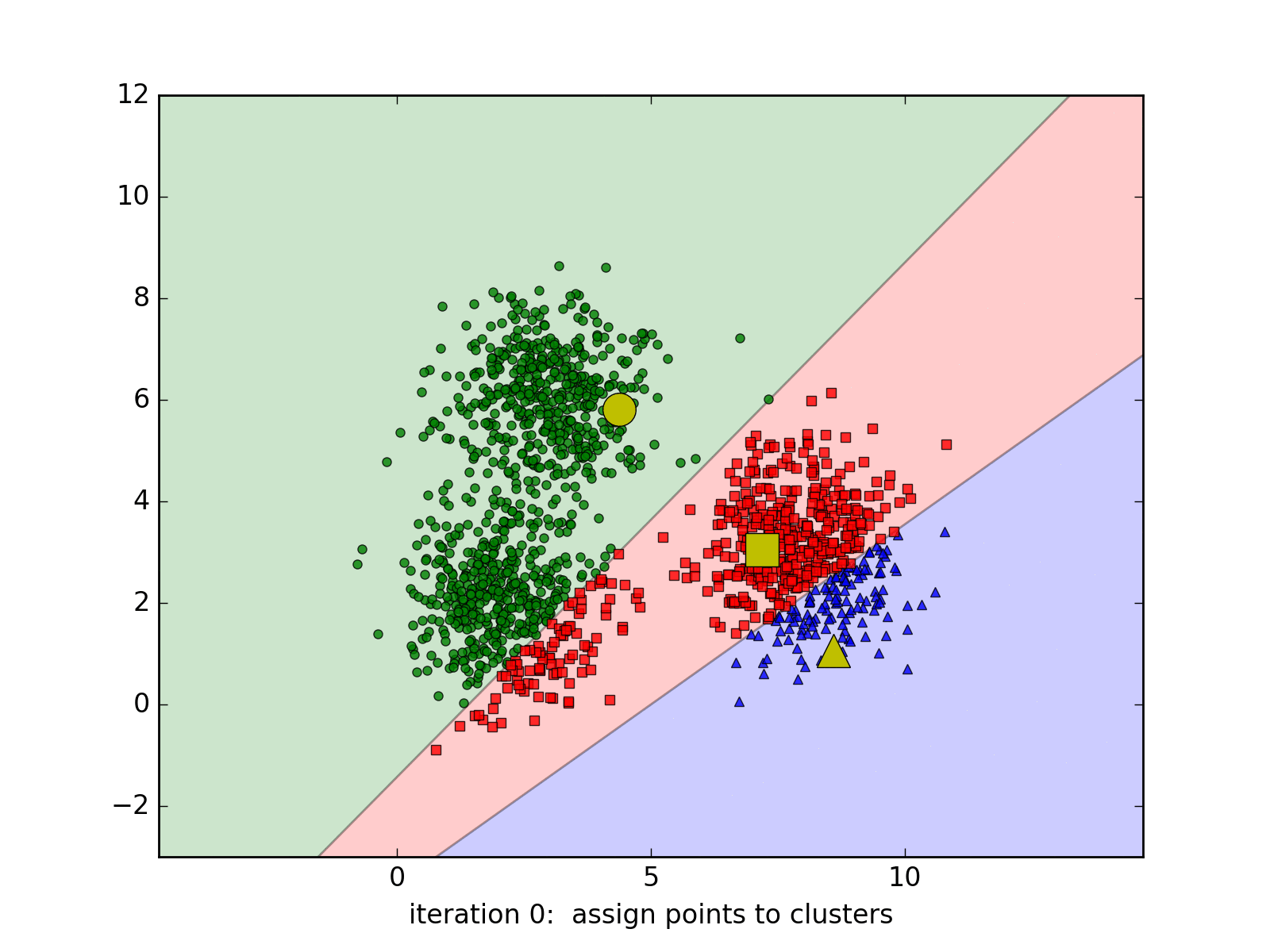
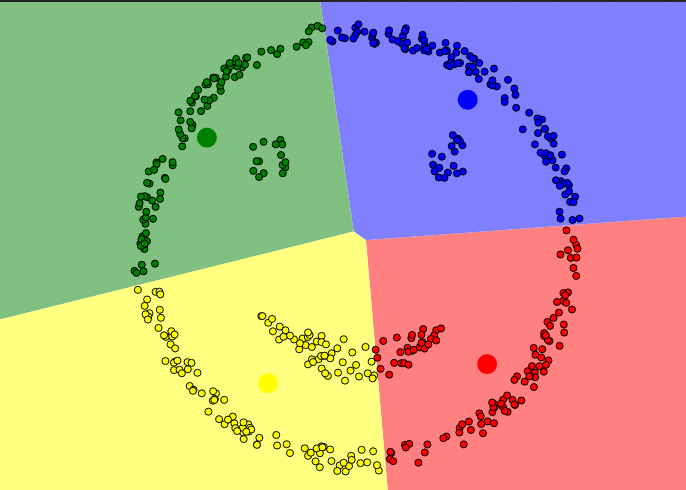
**Kmeans**

https://machinelearningcoban.com/2017/01/01/kmeans/



* Limit
* Number of K → Elbow method, Silhouette analysis
* Bad initial lead to bad output
* Highly imbalance points in each clusters (relating density)
* Non-linear data structure (vòng tròn trong vòng tròn)
* Data nên có dạng hình tròn



1. Cho các cặp điểm x,y có tọa độ như sau: (0.5; 0.5), (1; 0.5), (1; 1.5), (1.5; 1), (2.6; 2), (3; 2), (2.4; 2.5), (2.5; 3)

* Áp dụng thuật toán k-means với số centroid bằng 2.
* Khởi tạo tâm centroid tọa độ ngẫu nhiên, chạy thuật toán k-means 3 iterations và trả ra tọa độ 2 centroid.
* Vẽ 3 trường hợp khác nhau có tập dự liệu khi tiến hành phân cụm không hoạt động/ không tốt nếu sử dụng k-means. Nêu lí do.
* Đề xuất thuật toán phù hợp cho tập dữ liệu đó và giải thích tại sao thuật toán đó lại phù hợp

**GMM**

<https://towardsdatascience.com/gaussian-mixture-model-clearly-explained-115010f7d4cf>

<https://builtin.com/articles/gaussian-mixture-model#:~:text=Gaussian%20Mixture%20Model%20Defined,clusters%20in%20a%20data%20set>.

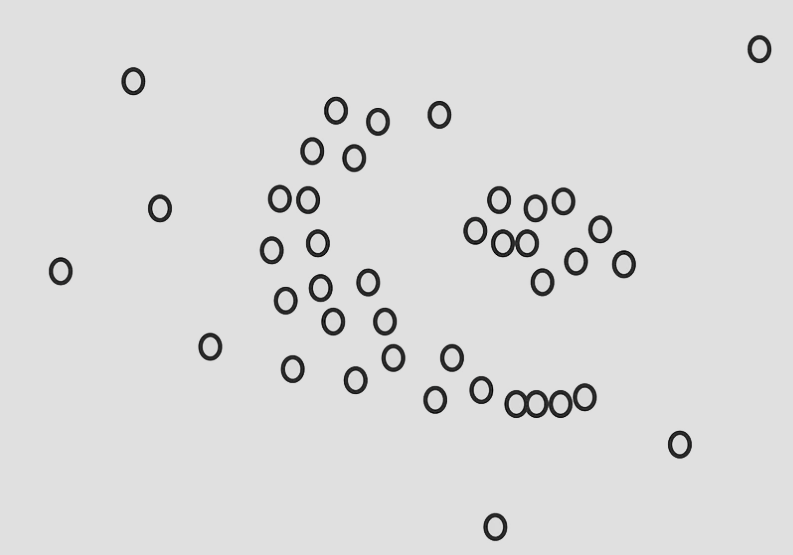
Say we want to classify datapoint into k clusters, each oh them follows its own Gauss distribution

Say we have z ∈ {0,1} → z follows Bernoulli distribution

1. **Thuật toán GMM**

* **Viết Loss function cho thuật toán**
* **Từ loss function chỉ ra hướng tiếp cận của thuật toán GMM để optimize loss này (E-M trong slide). Giải thích được ý nghĩa của E-step và M-step để làm gì (Có sử dụng công thức toán)**

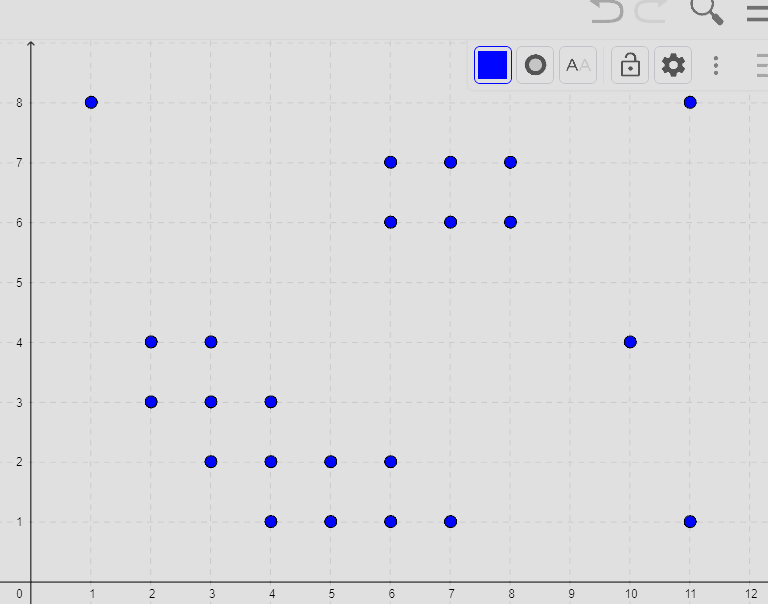
1. **Nêu các bước thực hiện của thuật toán GMM**
2. Cho hình sau:



Nêu cách thức hoạt động của dbscan theo ý hiểu (có thể dùng thêm toán để giải thích hoặc không).

Tham khảo link sau: <https://medium.com/mlearning-ai/dbscan-demystified-understanding-how-this-parameter-free-algorithm-works-89e03d7d7ab#:~:text=DBSCAN%20stands%20for%20Density-Based,their%20distance%20to%20other%20points>.

1. Cho các điểm trên hệ tọa độ OXY sau:



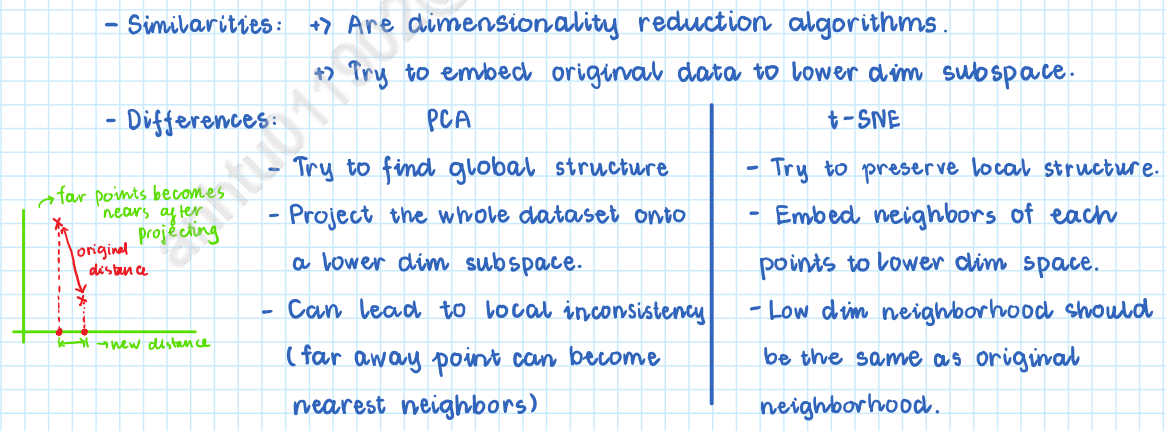
Áp dụng thuật toán DBSCAN với radius (epsilon) = 1.5 và min points = 3. Xác định các cụm và các noise points

**PCA**

1. Steps + chose k

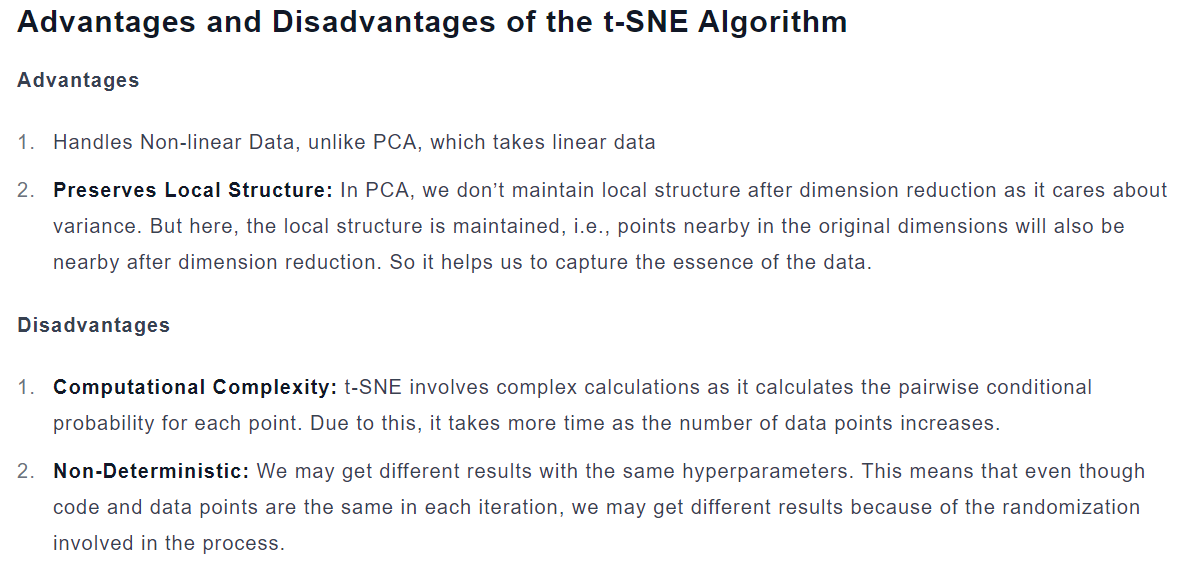
https://medium.com/analytics-vidhya/understanding-principle-component-analysis-pca-step-by-step-e7a4bb4031d9

1. Compare



**t-SNE**

<https://www.youtube.com/watch?v=NEaUSP4YerM>



5,

* Points should be spread randomly on a new space.
* Axes in t-SNE has no interpretable meaning, that is why t-SNE is only used for visualization and not for prediction.

0,

Say we embed 10 dim data into 2 dim, in high dim, we can have 11 point with the same distance between each other, but there no way to put these 11 points into lower dimension and still remain this characteristics. This is crowding problem, occuring when lower-dimensional representation fail to remain the relative distances and neighborhood between points in high dim due to the limited space. The consequence is  the difficulty in distinguishing individual data points or uncover structures and relationships within the data.

1,

The using of Gauss in lower dimensiona of SNE cause the crowding problem → t-SNE using T-distribution, which have a heavy tail, meaning far points can have higher possibility and center-around points have lower possibility. So that in cost function, less similar pair are penalized less heavily on the lower-dimensional embedding. Beside the new cost function is in a simpler form and is much easier and faster to compute, improving performance.

the Student t-distribution is that it is closely related to the Gaussian distribution, as the  
Student t-distribution is an infinite mixture of Gaussians. A computationally convenient property is that it is much faster to evaluate the density of a point under a Student t-distribution than under a Gaussian because it does not involve an exponential

2,

If two points are close together in high dimensional space, their dissimilarities are low and the probability p should be high (*p* ~ 1). Then, if they were mapped far away, the low dimensional probability would be low (*q* ~ 0). In this scenario we can see that the loss function takes very high values, severely penalizing that mistake. On the other hand, if two points are far from each other in high dimensional space, their dissimilarities are high and the probability p should be low (*p* ~ 0). Then, if they were mapped near each other, the low dimensional probability would be high (*q* ~ 1). We can see that the KL divergence is not penalizing this mistake as much as we would want.

***Intuition:****Since the KL divergence function does not penalize the misplacement in low dimensional space of points that are far away in high dimensional space, we can conclude that the global structure is not well preserved. t-SNE will group similar data points together into clusters, but distances between clusters might not mean anything.*

It is clear that in the high variance limit, the contribution of to the SNE cost function is just as important for distant neighbors as for close ones. When is very large, it can be shown that SNE is equivalent to minimizing the mismatch between squared distances in the two spaces, provided all the squared distances

3,

In t-SNE, perplexity can be interpreted as a sets of effective nearest neighbors. It have a positive correlation with the Gauss variance → we use it to represent variance of each point. It is not likely that there is a single value of σ*i* that is optimal for all datapoints in the data set because the density of the data is likely to vary. In dense regions, a smaller value of σ*i* is usually more appropriate than in sparser regions. If we set it too large, Gauss tend to be uniform, which not cover the feature of distance between points.

The performance of SNE is fairly robust to changes in the perplexity, and typical values are between 5 and 50.