HW2. Start early. Otherwise you can't finish

FOR LOOP

Parallel (broadcasting) numpy

Emojis are from Pinterest.

Gaussian Mixture Model

Mahdi Roozbahani
Georgia Tech

Some of the slides are inspired based on slides from Jiawei Han Chao Zhang, and Barnabás Póczos.

Outline

- Overview
- Gaussian Mixture Model
- The Expectation-Maximization Algorithm

Recap

Conditional probabilities:

$$p(A,B) = p(A|B)p(B) = p(B|A)p(A)$$

Bayes rule:

$$p(A|B) = \frac{p(A,B)}{p(B)} = \frac{p(B|A)p(A)}{p(B)}$$

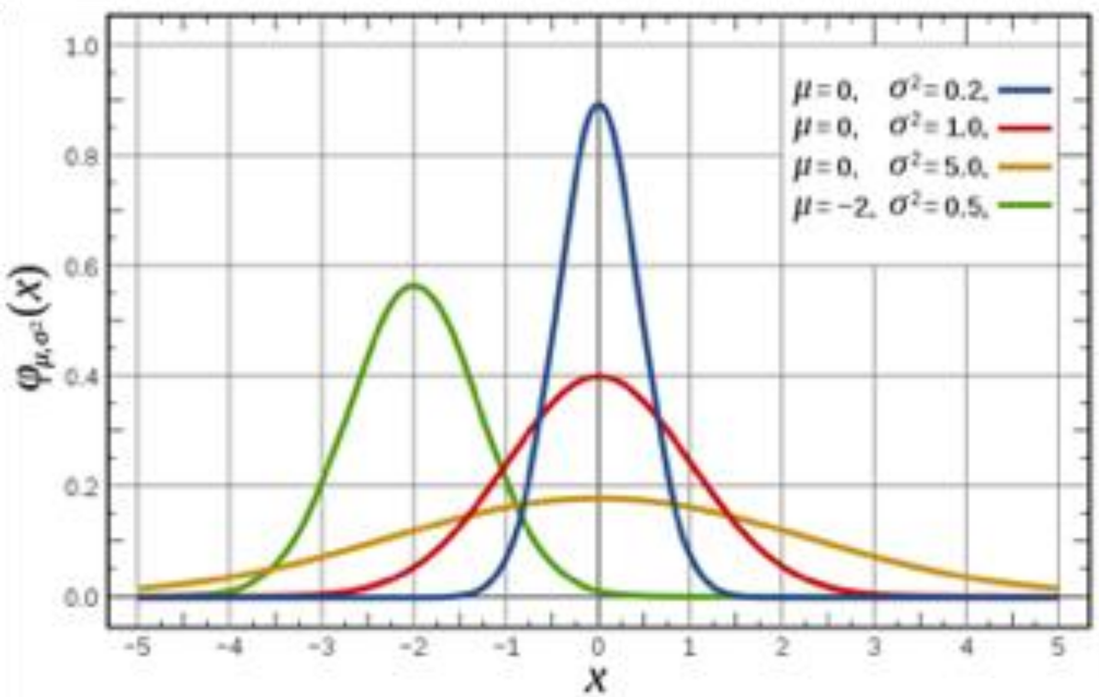
$$p(A=1) = \sum_{i=1}^{K} p(A=1,B_i) = \sum_{i=1}^{K} p(A|B_i)\,p(B_i)$$

	Tomorrow=Rainy	Tomorrow=Cold	P(Today)
Today=Rainy	4/9	2/9	[4/9 + 2/9] = 2/3
Today=Cold	2/9	1/9	[2/9 + 1/9] = 1/3
P(Tomorrow)	[4/9 + 2/9] = 2/3	[2/9 + 1/9] = 1/3	

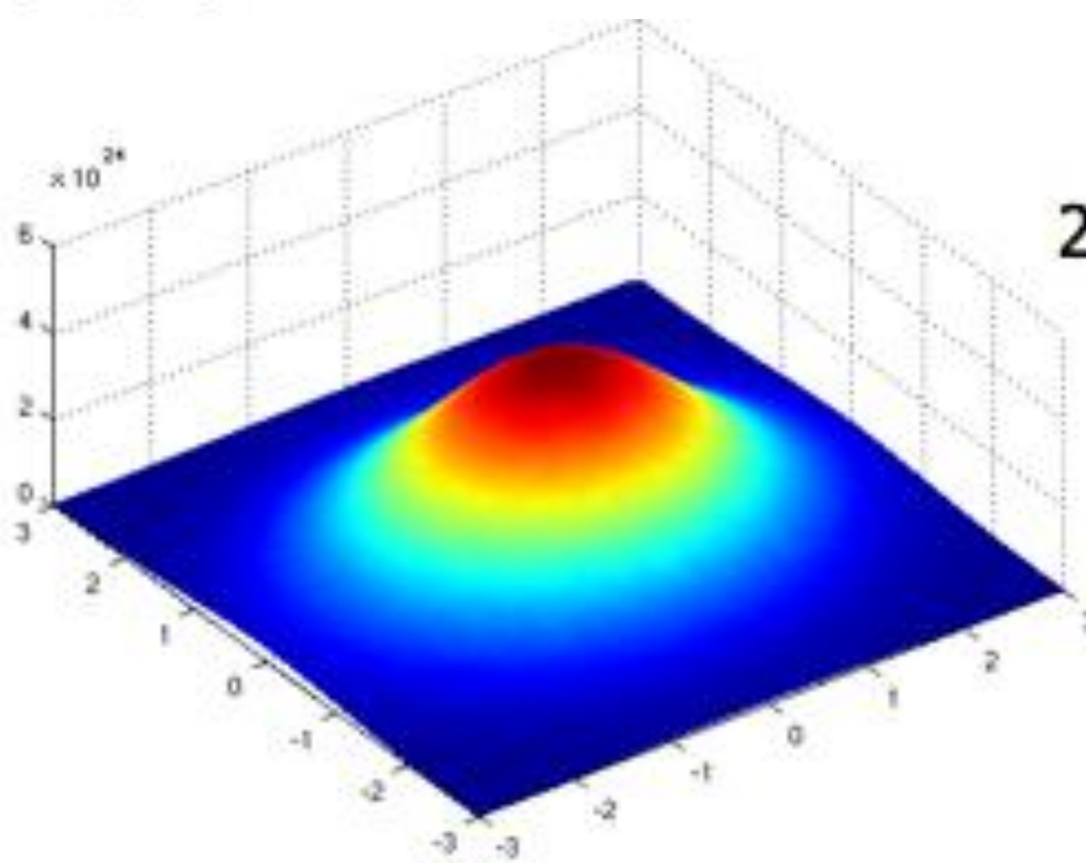
P(Tomorrow = Rainy) =

Gaussian Distribution

1-d Gaussian



$$N(\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\left(x^{\{1\}}-\mu\right)^2}{2\sigma^2}}$$



2-d Gaussian

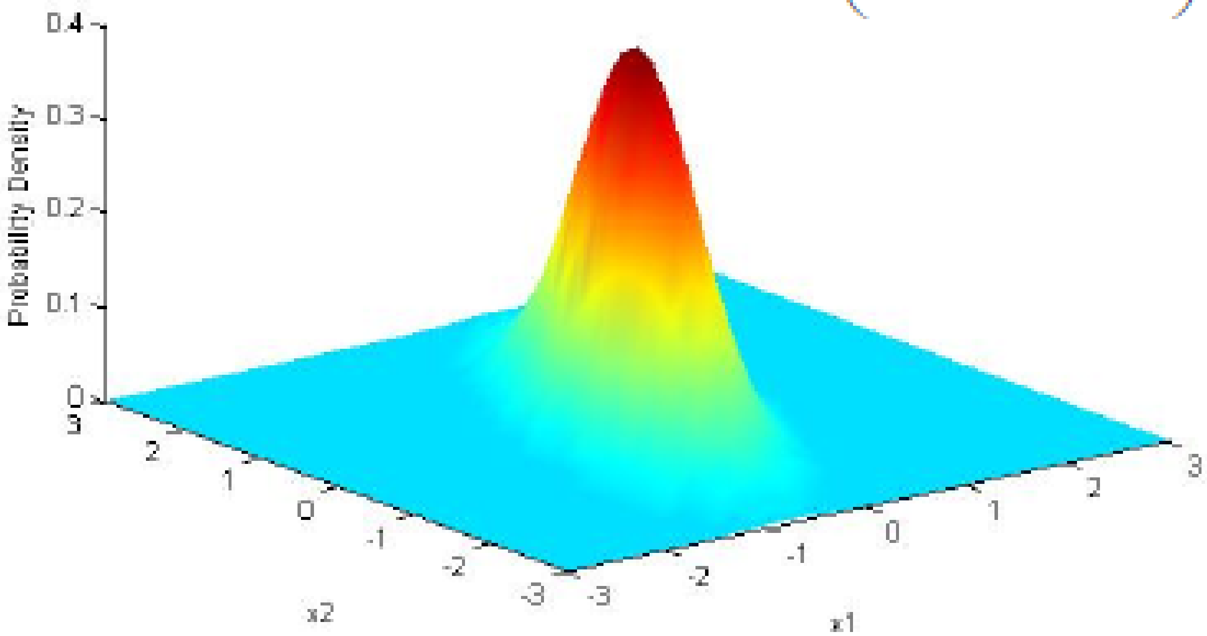
What is a Gaussian?

For ***d* dimensions**, the Gaussian distribution of a vector $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{x_3}, \dots, \boldsymbol{x_d})^{\boldsymbol{T}}$ is defined by:

$$N(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{d/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu) \right)$$

where μ is the mean and Σ is the covariance matrix of the Gaussian.

Example: $\mu = (0,0)^T$ $\Sigma = \begin{pmatrix} 0.25 & 0.30 \\ 0.30 & 1.00 \end{pmatrix}$



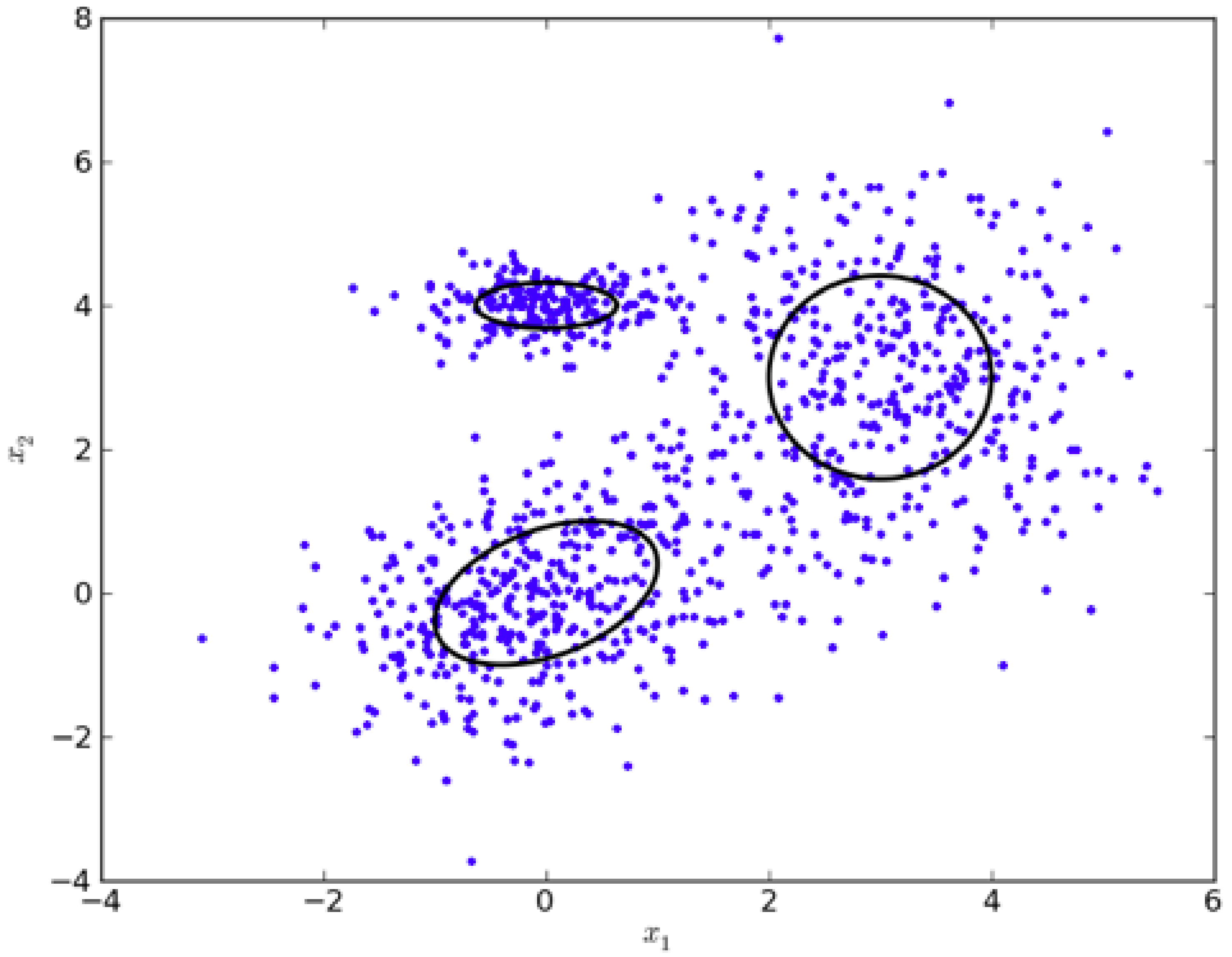
Is Gaussian Concave?

Outline

- Overview
- Gaussian Mixture Model
- The Expectation-Maximization Algorithm

Hard Clustering Can Be Difficult

- Hard Clustering: K-Means, Hierarchical Clustering, DBSCAN

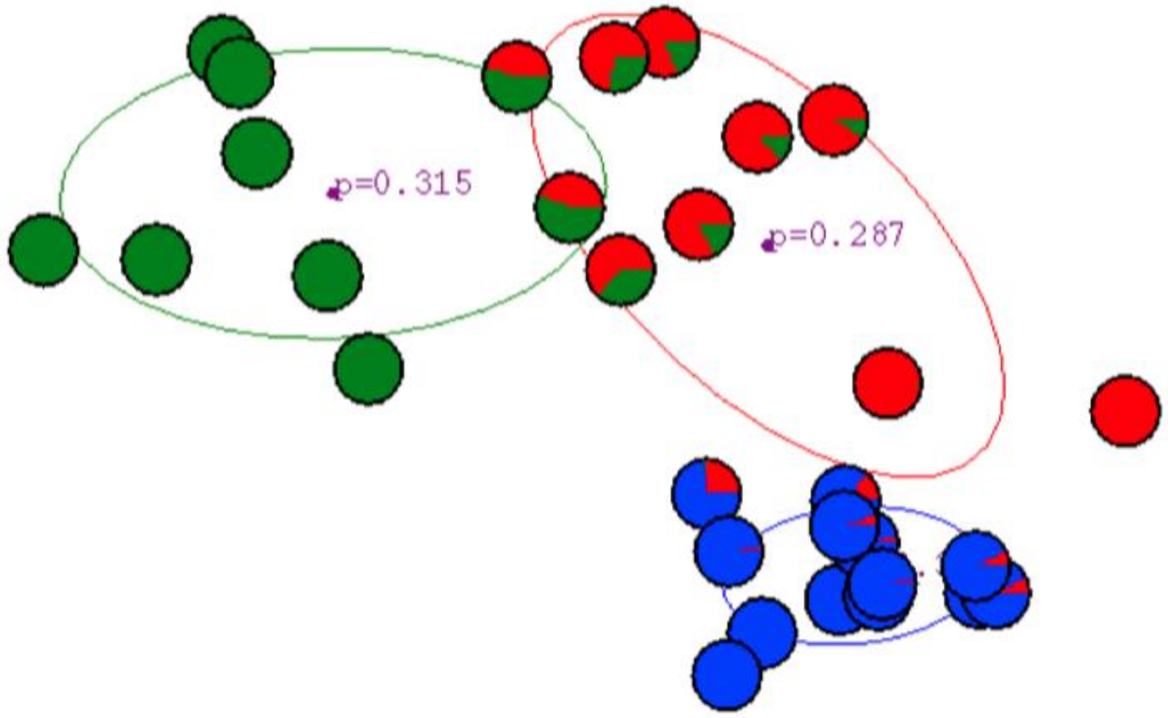


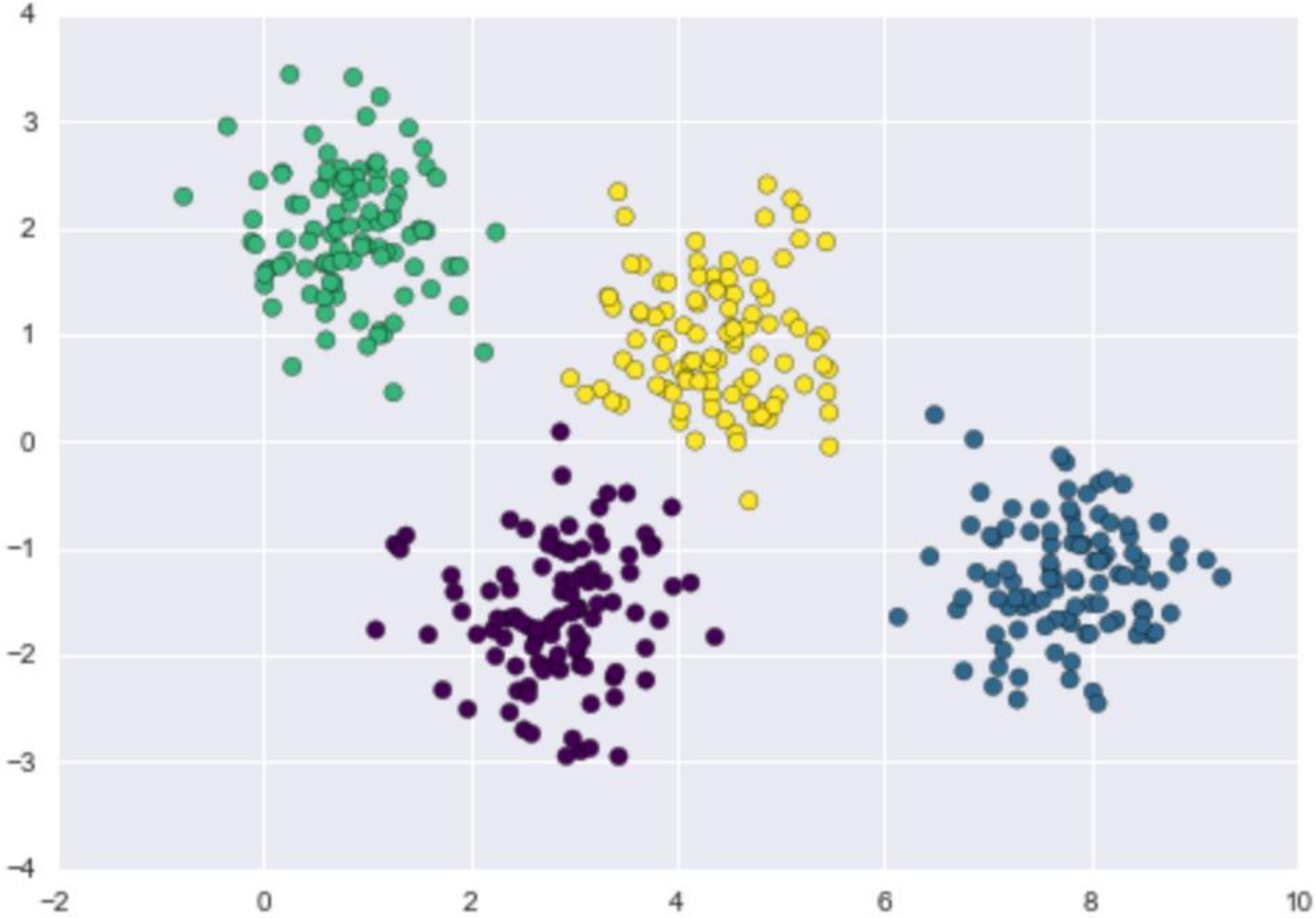
Towards Soft Clustering

- **K-means**
  - **hard assignment**: each object belongs to only one cluster

$$\theta_i \in \{\theta_1, \ldots, \theta_K\}$$

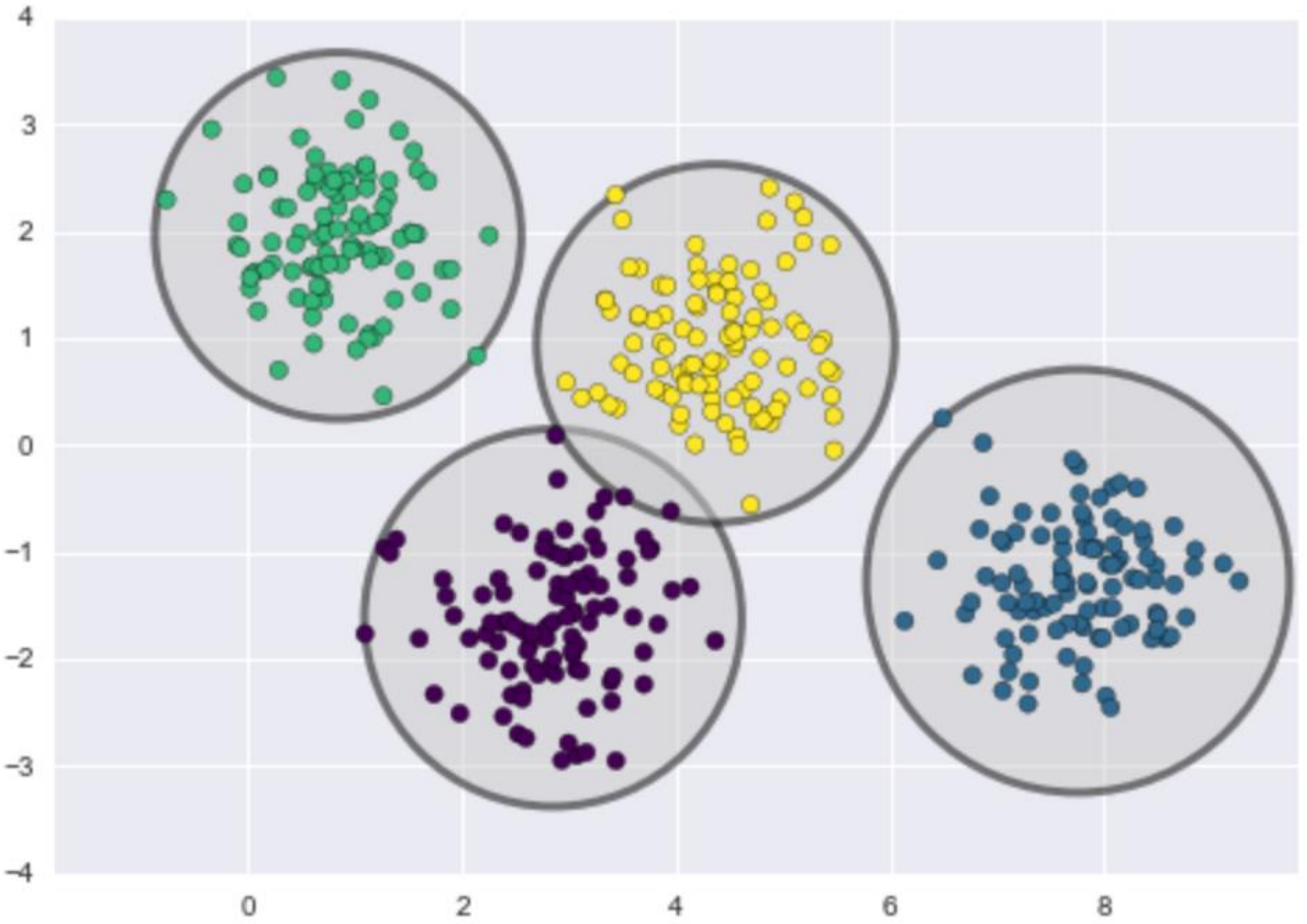
- **Mixture modeling**
  - **soft assignment**: probability that an object belongs to a cluster



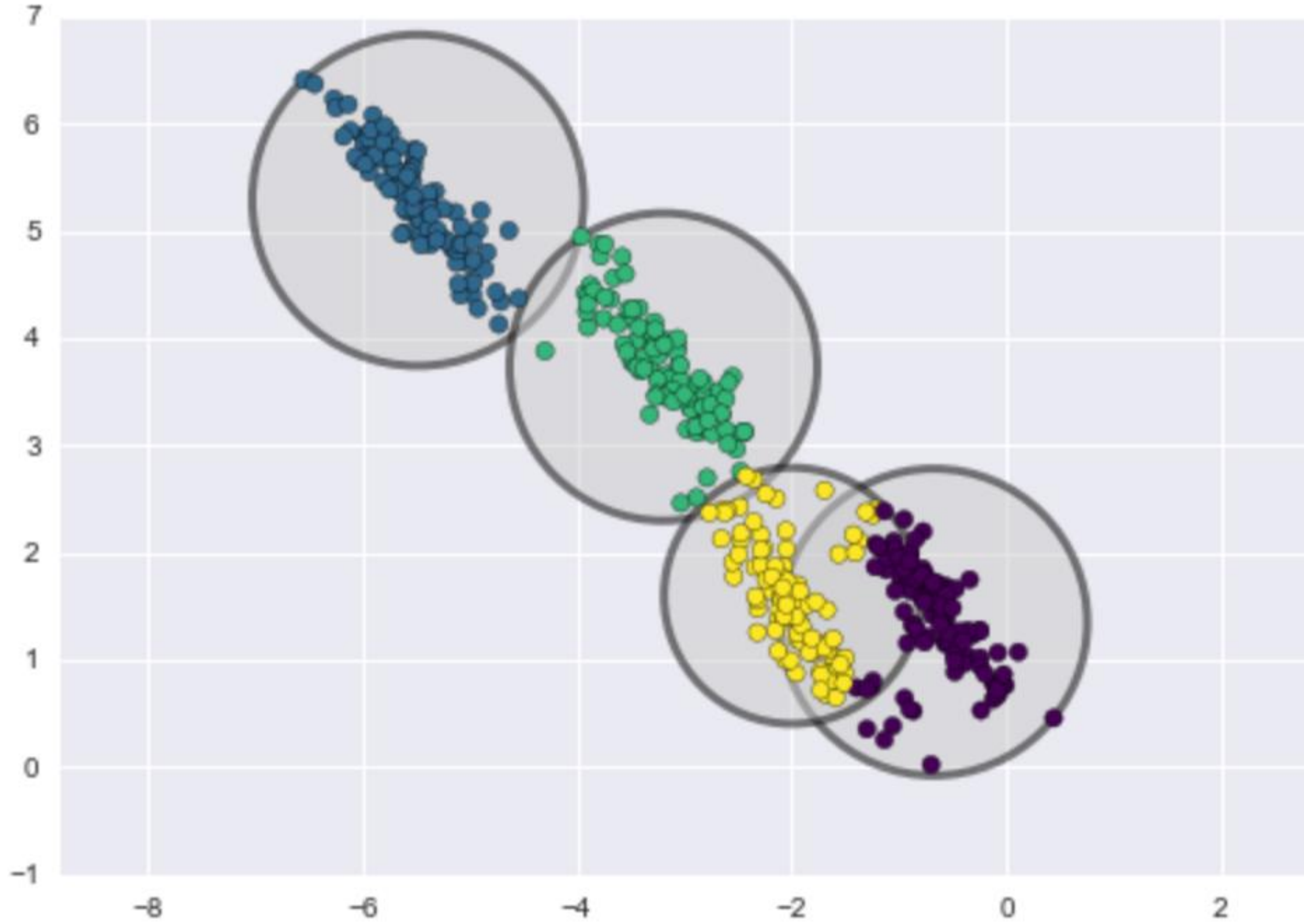


This is an excerpt from the [Python Data Science Handbook]() by Jake VanderPlas

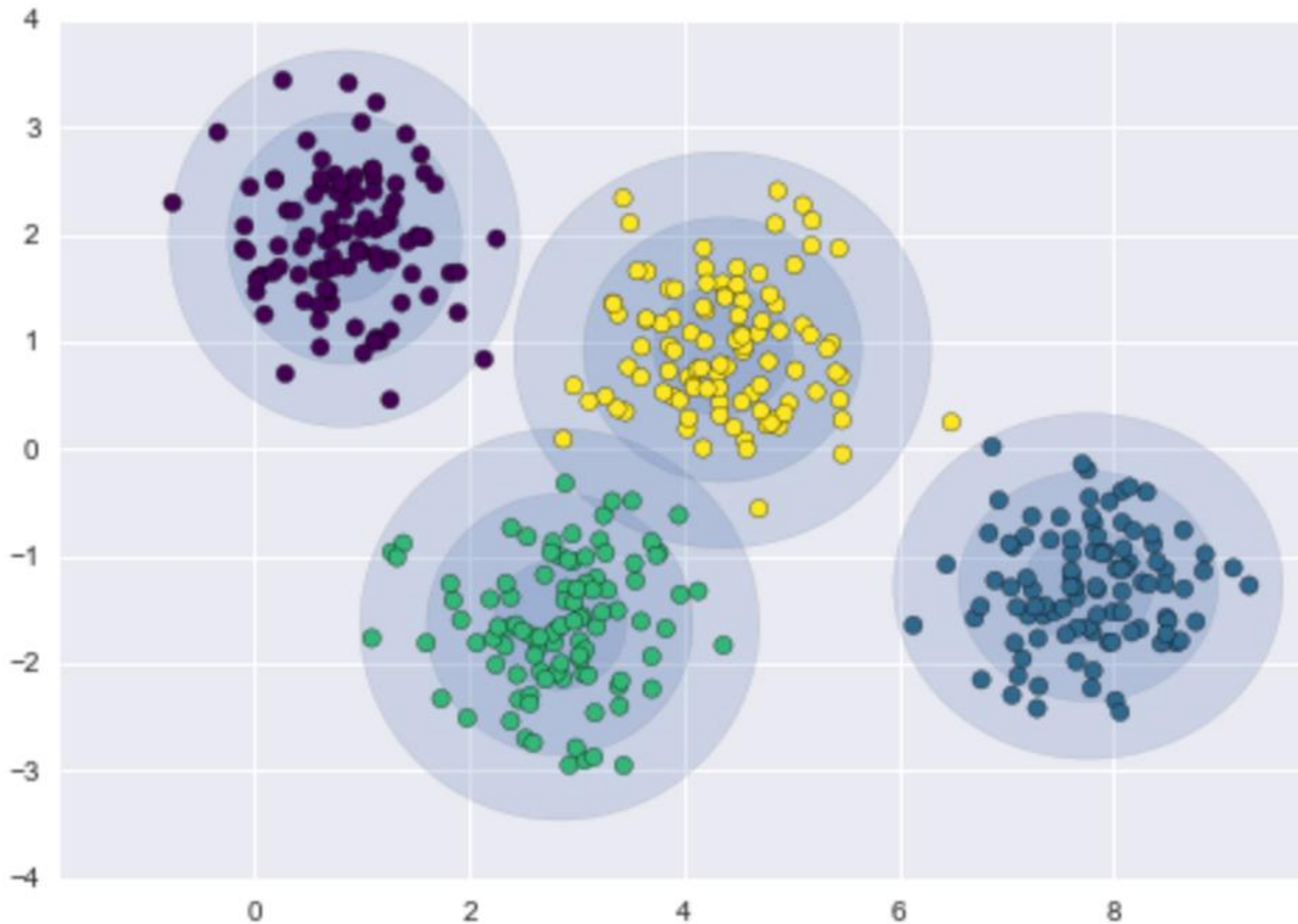
Let’s run K-Means on the dataset

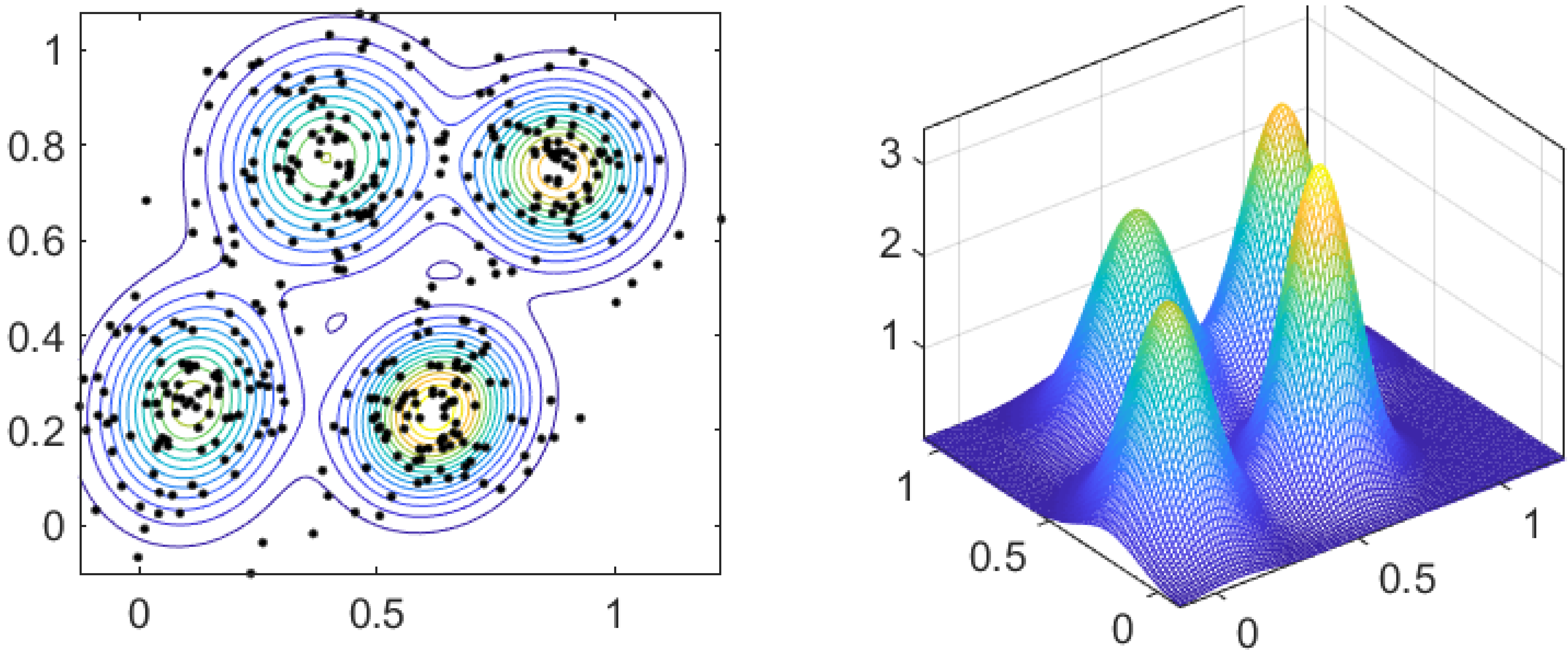


Let's generate a new dataset and run K-Means



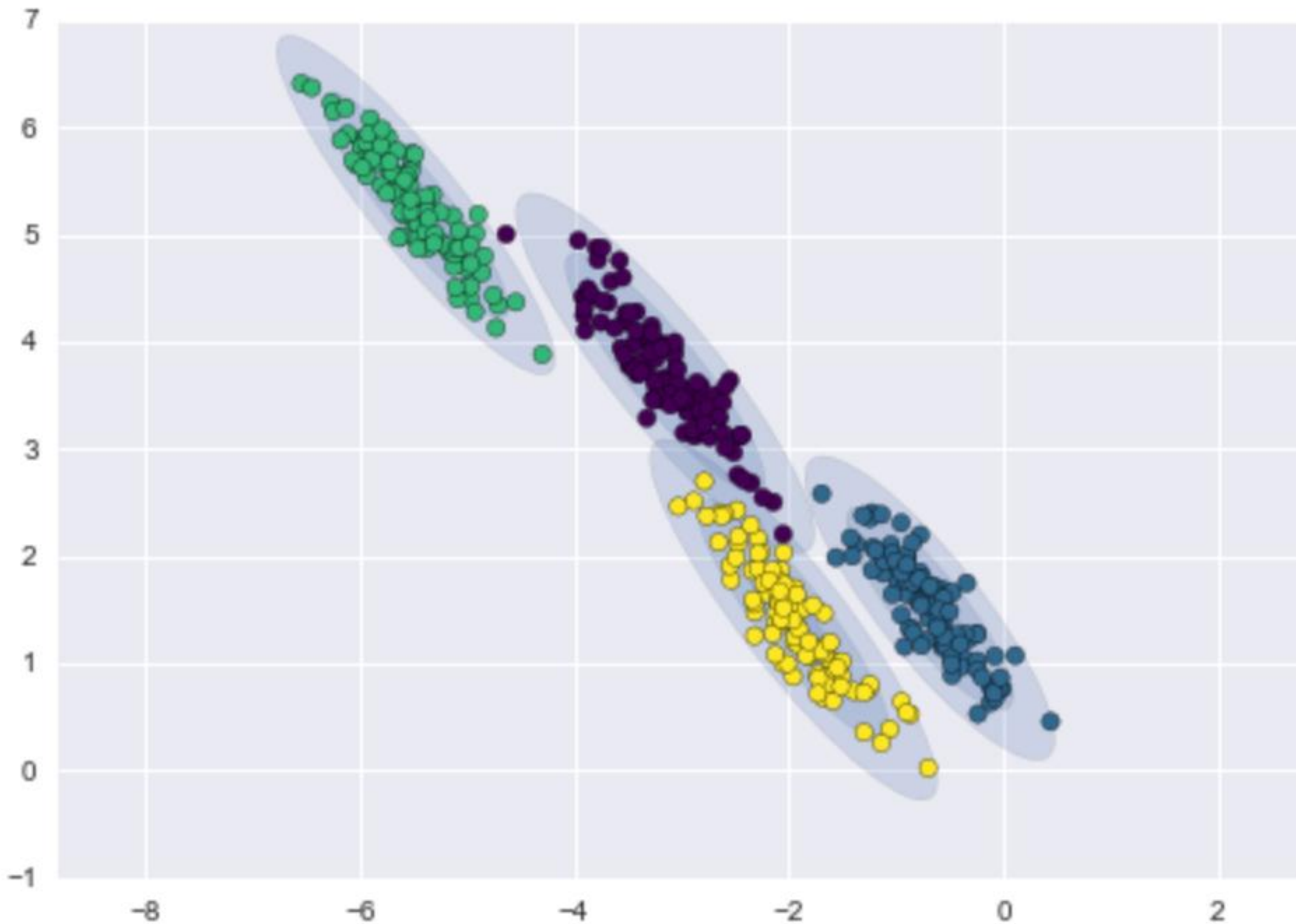
Let's run GMM on the first dataset

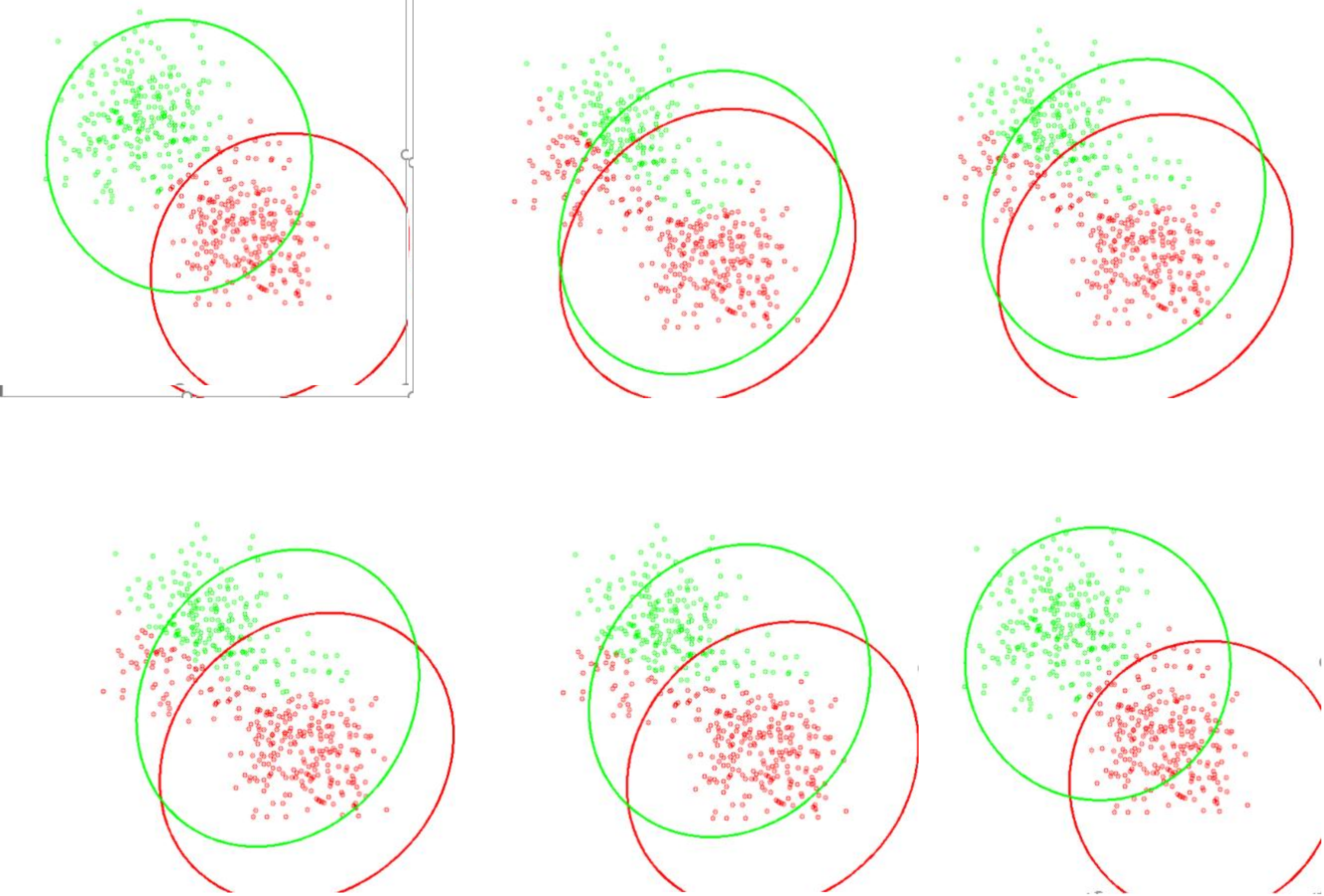


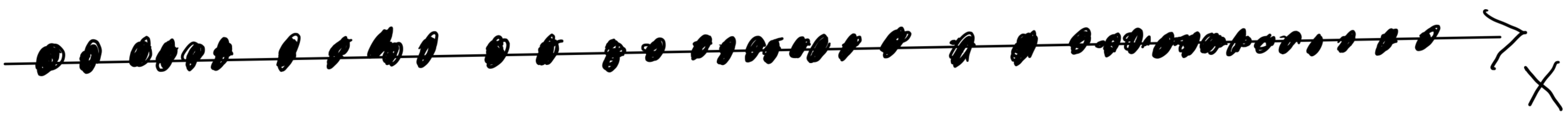


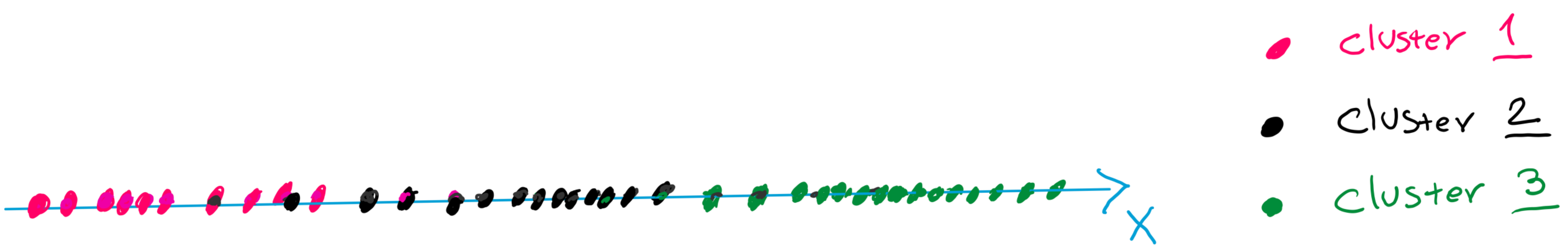
1
0.8
0.6
0.4
0.2
0
0
0.5
1
3
2
1
1
0.5
0
0
0.5
1

Let’s do GMM on the second dataset

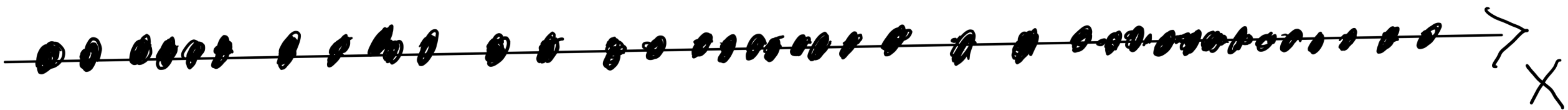


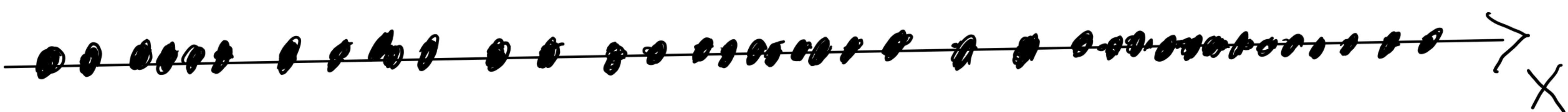



X

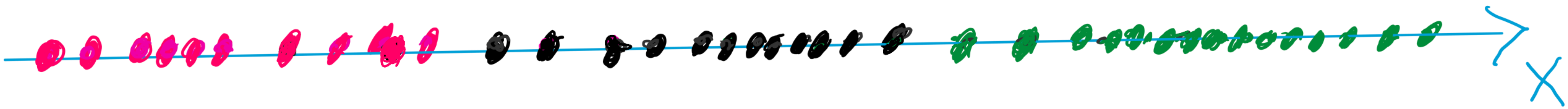

X
cluster 1
cluster 2
cluster 3

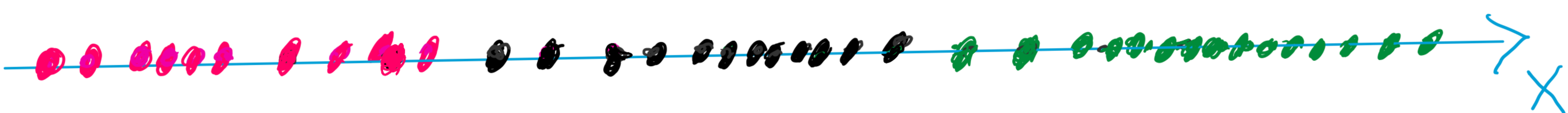
Initial Step





Final Step





Gaussian Recap

Mixture perspective – soft assignment

Let’s create a **SINGLE** pdf that combines all three Gaussians!!!!

Mixture perspective – soft assignment

Mixture perspective – Initialization

$$p(x) = \pi_0 N(X|\mu_0, \sigma_0) + \pi_1 N(X|\mu_1, \sigma_1) + \pi_2 N(X|\mu_2, \sigma_2)$$

Initial Step

Final Step

Maximizing Likelihood $\{\pi, \mu, \sigma\} \in \theta$

$$\mathrm{p}(\mathrm{x}|\pi, \mu, \sigma) = \mathrm{p}(\mathrm{x}|\theta) = p(x)$$
$$= \pi_0 N(X|\mu_0, \sigma_0) + \pi_1 N(X|\mu_1, \sigma_1) + \pi_2 N(X|\mu_2, \sigma_2)$$

$\boldsymbol{z_k}$ Math Notation

$$\mathrm{p(x)} = \pi_0 N(X|\mu_0, \sigma_0) + \pi_1 N(X|\mu_1, \sigma_1) + \pi_2 N(X|\mu_2, \sigma_2)$$

How can we find the probability of each data point in each cluster now?

Mixture Models

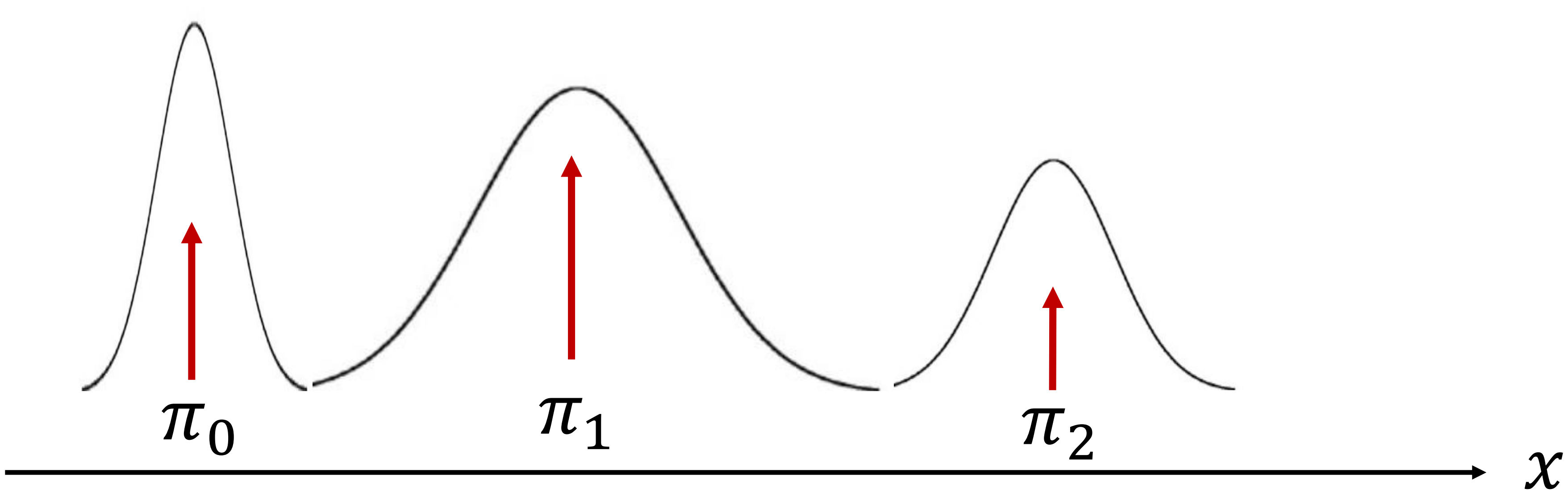
- Formally a Mixture Model is the weighted sum of a number of pdfs where the weights are determined by a distribution, π

$$p(x) = \pi_0 f_0(x) + \pi_1 f_1(x) + \pi_2 f_2(x) + \ldots + \pi_k f_k(x)$$

$$\text{where } \sum_{i=0}^{k} \pi_i = 1$$

$$\boxed{p(x) = \sum_{i=0}^{k} \pi_i f_i(x)}$$

What is **f** in GMM?



Mixture Models are Generative

- Generative simply means dealing with joint probability $p(x, z)$

$$\mathrm{p(x)} = \pi_0 f_0(x) + \pi_1 f_1(x) + \cdots + \pi_k f_k(x)$$

Let's say $f(.)$ is a Gaussian distribution

$$\mathrm{p(x)} = \pi_0 N(X|\mu_0, \sigma_0) + \pi_1 N(X|\mu_1, \sigma_1) + \cdots + \pi_k N(X|\mu_k, \sigma_k)$$

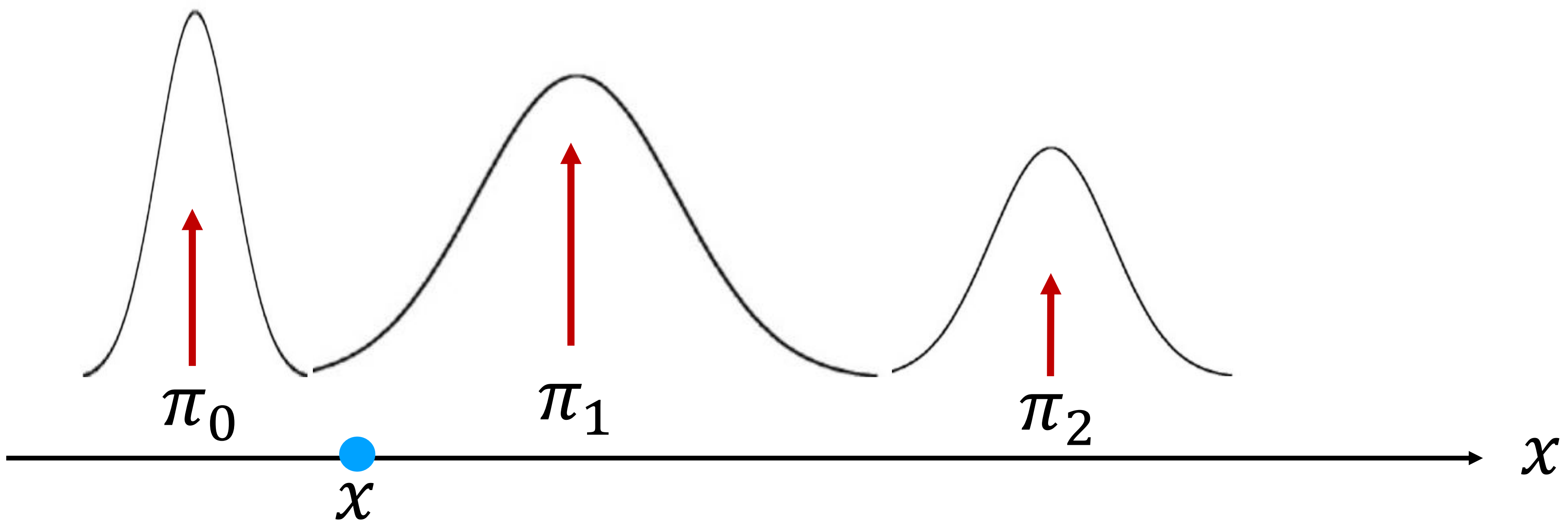
$$p(x) = \sum_k N(x|\mu_k, \sigma_k)\pi_k$$

$$p(x) = \sum_k p(x|z_k)p(z_k)$$

z_k is component k

$$p(x) = \sum_k p(x, z_k)$$

What is soft assignment?



What is the probability of a datapoint x in each component?

How many components we have here? **3**

How many probability? **3**

What is the sum value of the 3 probabilities for each datapoint? **1**

Inferring Cluster Membership

- We have representations of the joint $p(x, z_{nk}|\theta)$ and the marginal, $p(x|\theta)$
- The conditional of $p(z_{nk}|x,\theta)$ can be derived using Bayes rule.
  - The **responsibility** that **a** mixture component takes for explaining an observation x.

$$\tau(z_k) = p(z_k|x) = \frac{p(z_k)p(x|z_k)}{\sum_{j=1}^{K} p(z_j)p(x|z_j)}$$

$$= \frac{\pi_k N(x|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x|\mu_j, \Sigma_j)}$$

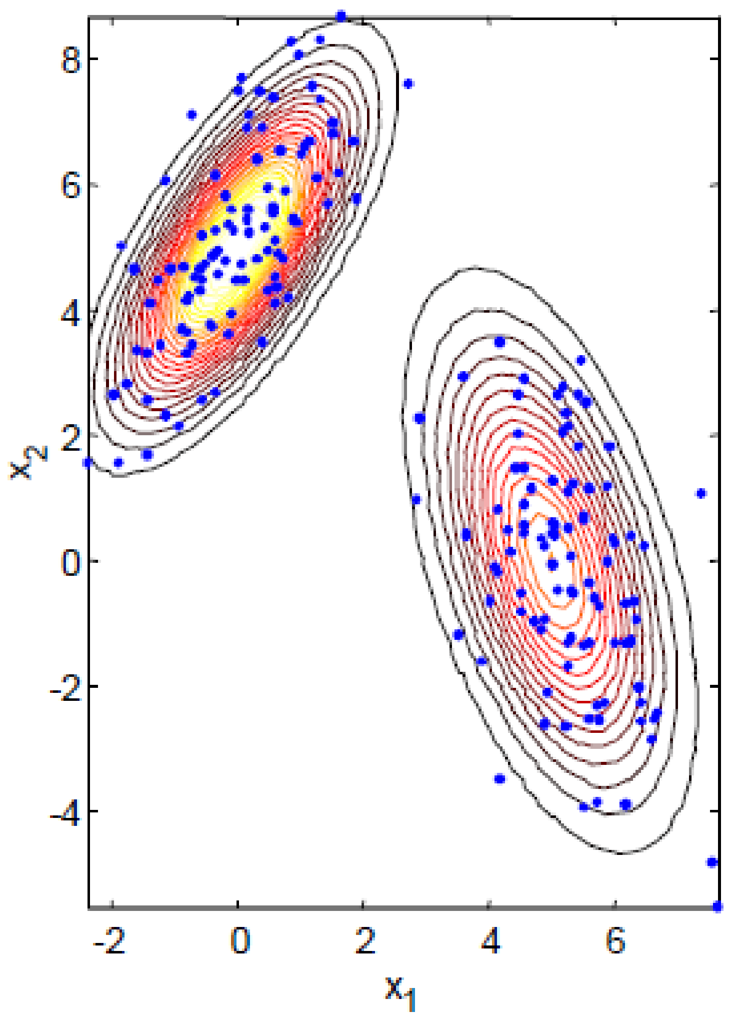
Why having “Latent variable”

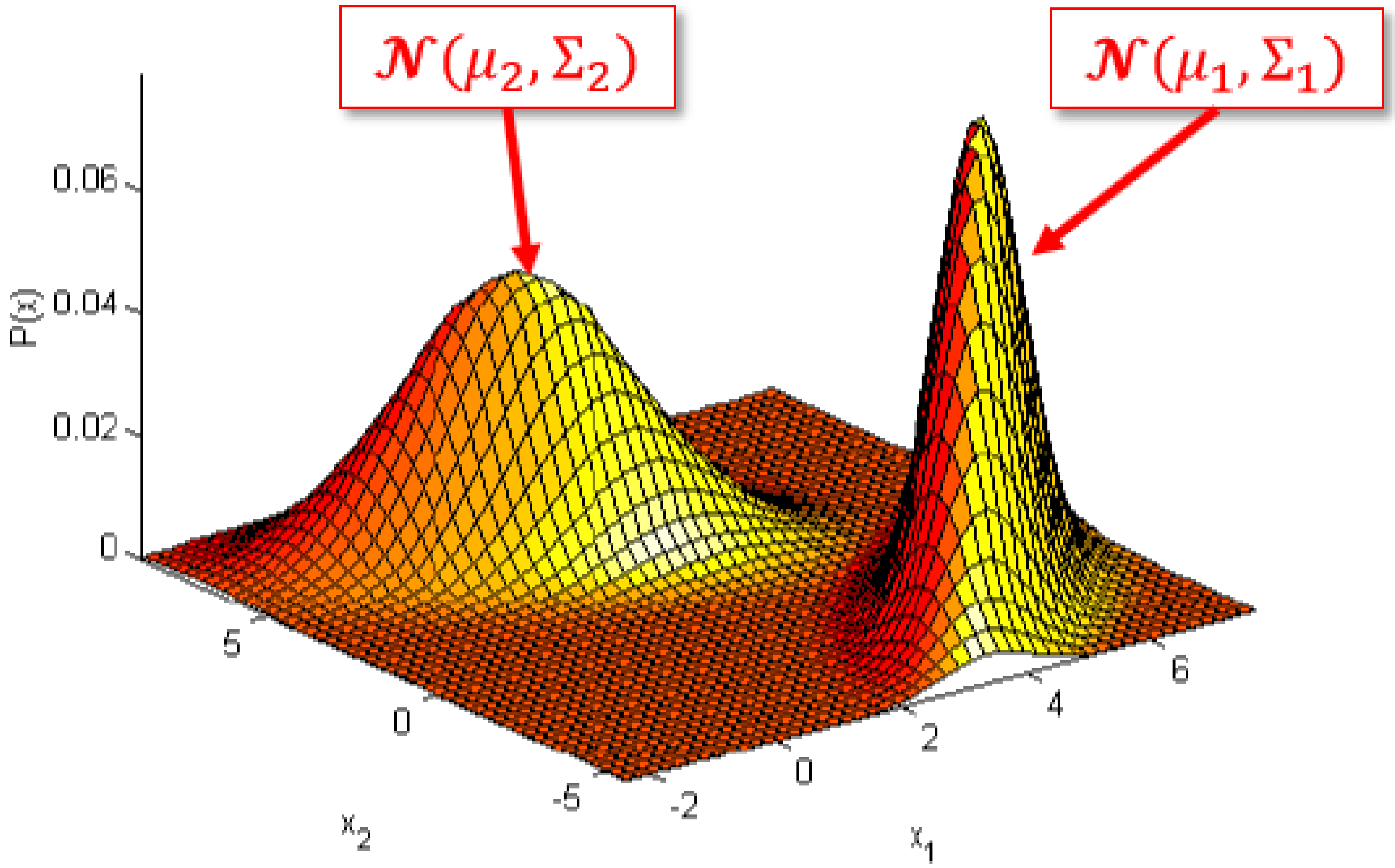
- A variable can be unobserved (latent) because:
  - it is an imaginary quantity meant to provide some simplified and abstractive view of the data generation process.
    - e.g., speech recognition models, mixture models (soft clustering)…
  - it is a real-world object and/or phenomena, but difficult or impossible to measure
    - e.g., the temperature of a star, causes of a disease, evolutionary ancestors …
  - it is a real-world object and/or phenomena, but sometimes wasn’t measured, because of faulty sensors, etc.

- Discrete latent variables can be used to partition/cluster data into sub-groups.
- Continuous latent variables (factors) can be used for dimensionality reduction (factor analysis, etc).

How about GMM for multimodal distribution?

- What if we know the data consists of a few Gaussians
- What if we want to fit parametric models





Gaussian Mixture Model

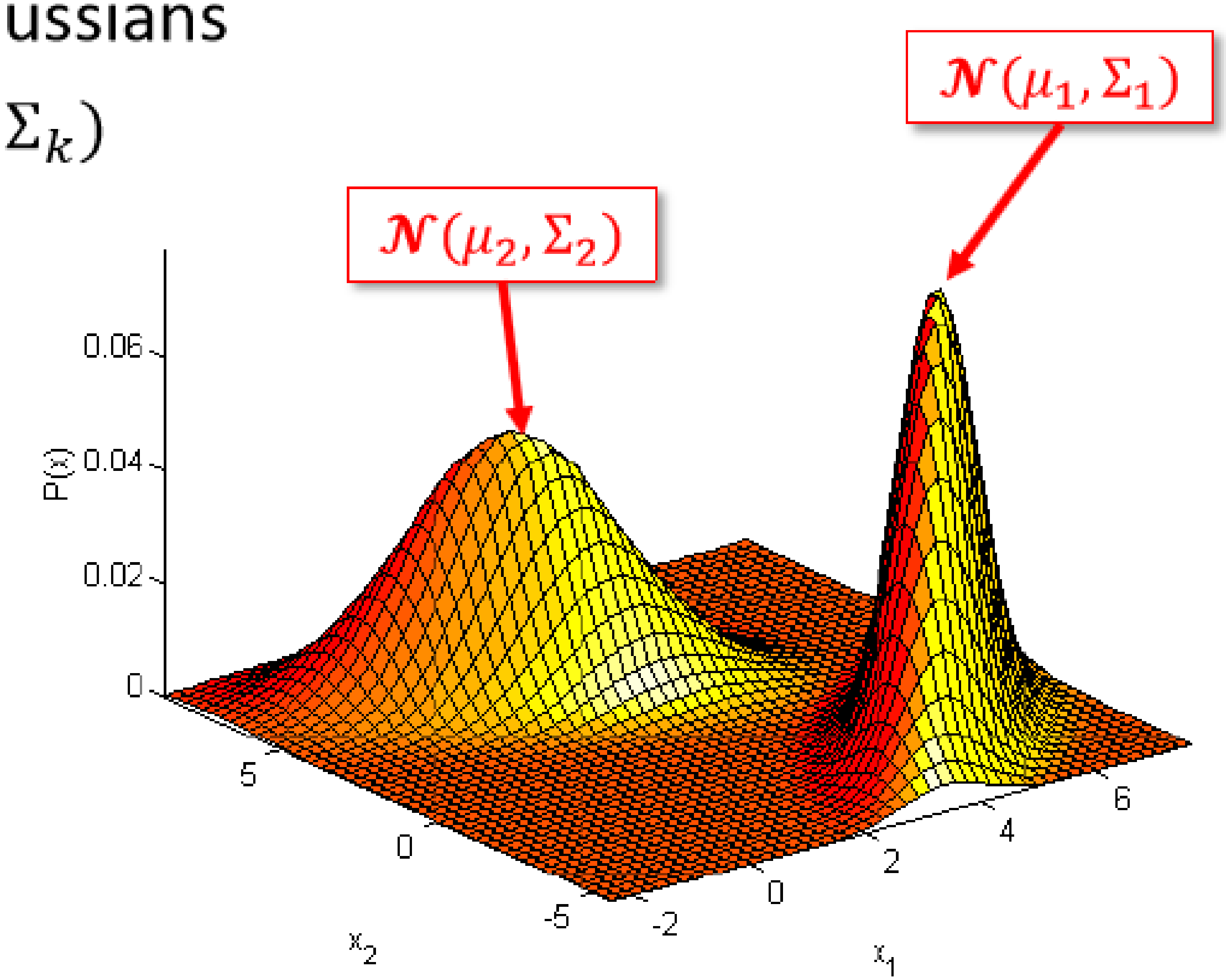
- A density model $p(X)$ may be multi-modal: model it as a mixture of uni-modal distributions (e.g. Gaussians)

- Consider a mixture of K Gaussians
  - $p(X) = \sum_{k=1}^{K} \pi_k \mathcal{N}(X|\mu_k, \Sigma_k)$

mixing proportion

mixture Component

- Learn $\pi_k \in (0,1), \mu_k, \Sigma_k$;



What are GMM parameters?

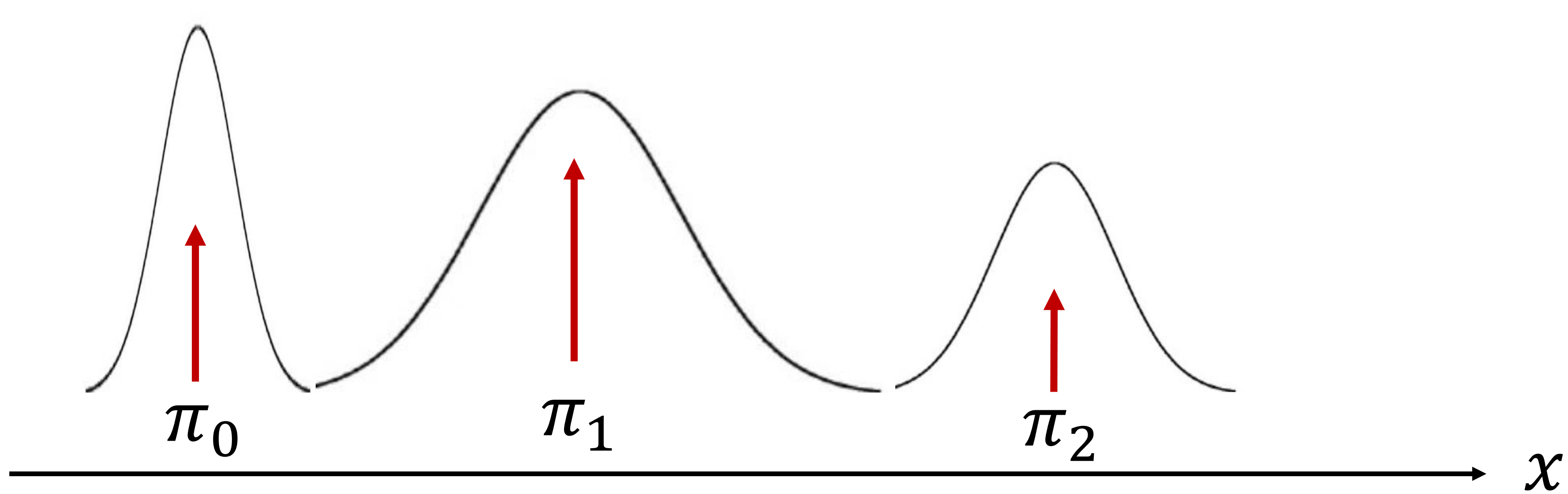
Mean μ_k Variance σ_k Size π_k

Marginal probability distribution

$$\mathrm{p}(\mathrm{x}|\theta) = \sum_k p(x, z_k|\theta) = \sum_k \underbrace{p(x|z_k, \theta)}_{f_k(x)} \underbrace{p(z_k|\theta\ )}_{\pi_k} = \sum_k N(x|\mu_k, \sigma_k)\pi_k$$

$p(z_k|\theta) = \pi_k$ Select a mixture component with probability π

$p(x|z_k, \theta) = N(x|\mu_k, \sigma_k)$ Sample from that component's Gaussian



Parameters' definition

- Purpose: GMM is a clustering algorithm derived from probabilistic theory that uses soft-assignment, meaning that data points have probability of being associated/generated from K gaussians/clusters. This is as opposed to K-means where data points definitively are either from a cluster or they're not.

- Gaussian Parameters
  - μ: Mean of each gaussian, can be compared to the K-means cluster centers
  - Σ: Covariance matrix of each gaussian, which represents how dimensions vary between each other. If it's the covariance of a specific dimension with itself, it is just the standard deviation of that variable/dimension. This is in a DxD matrix (for each gaussian and every element Σi,j represents the covariance of dimension I with j. If you assume the dimensions are independent, then only the diagonals are non-zero.
  - z_{nk}: Latent variable which isn't explicitly known, but tells us which gaussian each datapoint was generated from. z is binary, it either's 1 (point x came from gaussian k) or 0 (point x did not come from gaussian k)
  - $p(z_{nk}) = \pi$: Mixing proportions/weights, which represent the fraction of data points that are generated from/associated with each gaussian. These sum to 1.

- $N(X_n \mid \mu_k, \Sigma_k)$: This term is the probability of some data point X_n occurring based on the assumption that it is generated from gaussian k. Mathematically, this Is equal to the likelihood ($p(x|z,\mu_k, \Sigma_k)$). Multiply this with the mixing weight π, and you get the joint distribution p(x=X_n,z=k).

- P(X): If you sum up all the $N(X_n \mid \mu_k, \Sigma_k)*\pi$ terms for each cluster, you get the probability of the entire data set occurring.

- $\gamma(z_{nk})$ or $P(z_{nk} \mid x_n)$: We call this term the "responsibility." It is the probability of z for "A" data point x, meaning that this is probability that point n is generated from gaussian k normalized by P(X), the probability of the entire data set occurring.

Well, we don't know $\pi_k, \mu_k, \Sigma_\mathrm{k}$ What should we do?

We use a method called "Maximum Likelihood Estimation" (MLE) to solve the problem.

$$\mathrm{p(x)} = \mathrm{p(x}|\theta) = \sum_k p(x, z_k|\theta) = \sum_k p(z_k|\theta)p(x|z_k, \theta) = \sum_{k=0}^{K} \pi_k N(x|\mu_k, \Sigma_k)$$

Let's identify a likelihood function, why?

Because we use likelihood function to optimize the probabilistic model parameters!

$$\arg\max p(x|\theta) = p(x|\pi, \mu, \Sigma) = \prod_{n=1}^{N} p(x_n|\theta) = \prod_{n=1}^{N} \sum_{k=0}^{K} \pi_k N(x_n|\mu_k, \Sigma_k)$$

$$\arg\max p(x) = p(x|\pi,\mu,\Sigma) = \prod_{n=1}^{N} p(x_n|\theta) = \prod_{n=1}^{N}\sum_{k=0}^{K} \pi_k N(x_n|\mu_k,\Sigma_k)$$

$$\ln[p(x)] = \ln[p(x|\pi,\mu,\Sigma)]$$

- As usual: Identify a likelihood function

$$\ln p(x|\pi,\mu,\Sigma) = \sum_{n=1}^{N} \ln\left\{\sum_{k=1}^{K} \pi_k N(x_n|\mu_k,\Sigma_k)\right\}$$

- And set partials to zero…

Maximum Likelihood of a GMM

- Optimization of means.

$$\ln p(x|\pi,\mu,\Sigma) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k N(x_n|\mu_k,\Sigma_k) \right\}$$

$$\frac{\partial \ln p(x|\pi,\mu,\Sigma)}{\partial \mu_k} = \sum_{n=1}^{N} \frac{\pi_k N(x_n|\mu_k,\Sigma_k)}{\sum_j \pi_j N(x_n|\mu_j,\Sigma_j)} \Sigma_k^{-1}(x_k - \mu_k) = 0$$

$$= \sum_{n=1}^{N} \tau(z_{nk})\Sigma_k^{-1}(x_k - \mu_k) = 0$$

$$\boxed{\mu_k = \frac{\sum_{n=1}^{N} \tau(z_{nk})x_n}{\sum_{n=1}^{N} \tau(z_{nk})}}$$

Maximum Likelihood of a GMM

- Optimization of covariance

$$\ln p(x|\pi,\mu,\Sigma) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k N(x_n|\mu_k,\Sigma_k) \right\}$$

$$\boxed{\Sigma_k = \frac{1}{\sum_{n=1}^{N} \tau(z_{nk})} \sum_{n=1}^{N} \tau(z_{nk})(x_n-\mu_k)(x_n-\mu_k)^T}$$

Maximum Likelihood of a GMM

- Optimization of mixing term

$$\ln p(x|\pi, \mu, \Sigma) + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

$$0 = \sum_{n=1}^{N} \frac{N(x_n|\mu_k, \Sigma_k)}{\sum_j \pi_j N(x_n|\mu_j, \Sigma_j)} + \lambda$$

$$\boxed{\pi_k = \frac{\sum_{n=1}^{N} \tau(z_{nk})}{N}}$$

MLE of a GMM

$$\boxed{\mu_k = \frac{\sum_{n=1}^{N} \tau(z_{nk})x_n}{N_k}}$$

$$\boxed{\Sigma_k = \frac{1}{N_k}\sum_{n=1}^{N} \tau(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T}$$

$$\boxed{\pi_k = \frac{N_k}{N}}$$

$$N_k = \sum_{n=1}^{N} \tau(z_{nk})$$

Outline

- Overview
- Gaussian Mixture Model
- The Expectation-Maximization Algorithm

EM for GMMs

- E-step: Evaluate the Responsibilities

$$\boxed{\tau(z_k) = \frac{\pi_k N(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x_n|\mu_j, \Sigma_j)}}$$

EM for GMMs

- M-Step: Re-estimate Parameters

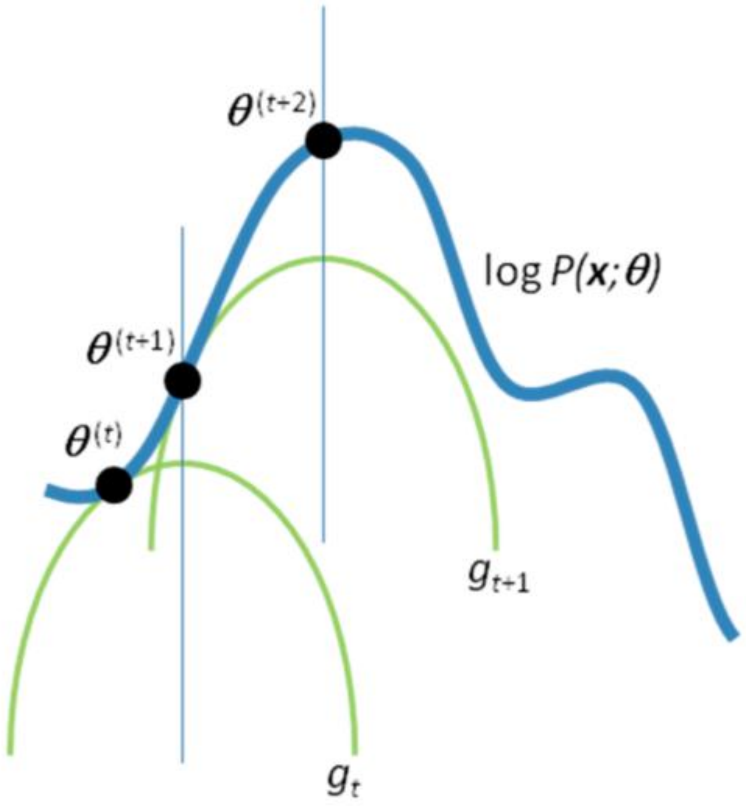
$$\boxed{\mu_k^{new} = \frac{\sum_{n=1}^{N} \tau(z_{nk})x_n}{N_k}}$$

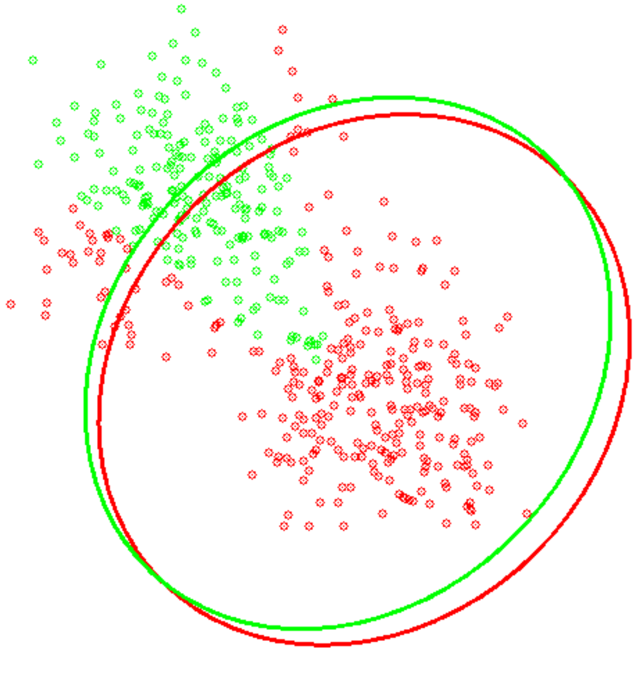
$$\boxed{\Sigma_k^{new} = \frac{1}{N_k}\sum_{n=1}^{N} \tau(z_{nk})(x_n - \mu_k^{new})(x_n - \mu_k^{new})^T}$$

$$\boxed{\pi_k^{new} = \frac{N_k}{N}}$$

Expectation Maximization

- Expectation Maximization (EM) is a general algorithm to deal with hidden variables.
- Two steps:
  - E-Step: Fill-in hidden values using inference
  - M-Step: Apply standard MLE method to estimate parameters
- EM always converges to a local minimum of the likelihood.





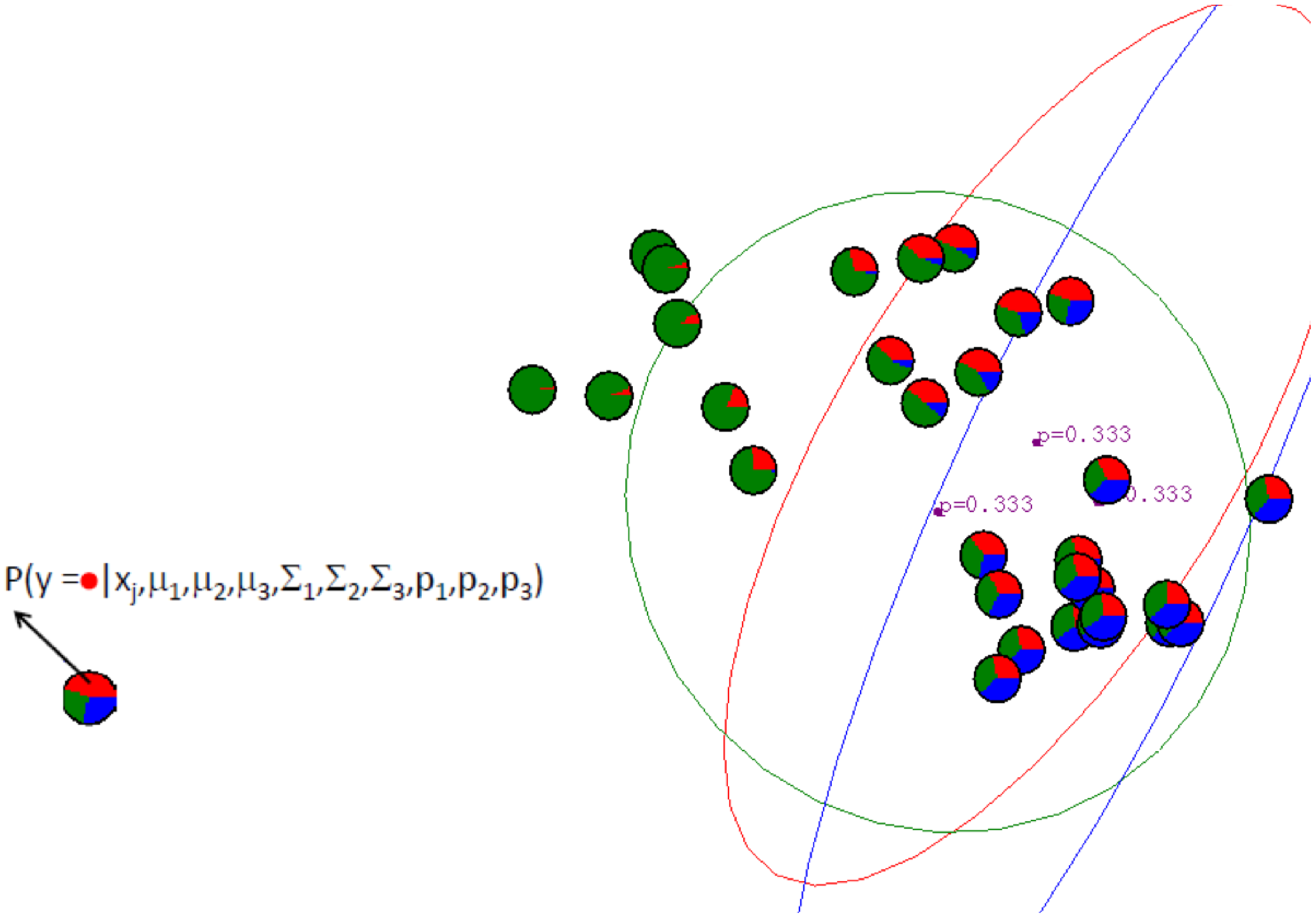
EM for Gaussian Mixture Model: Example

covariance_type="diag"

covariance_type="spherical"

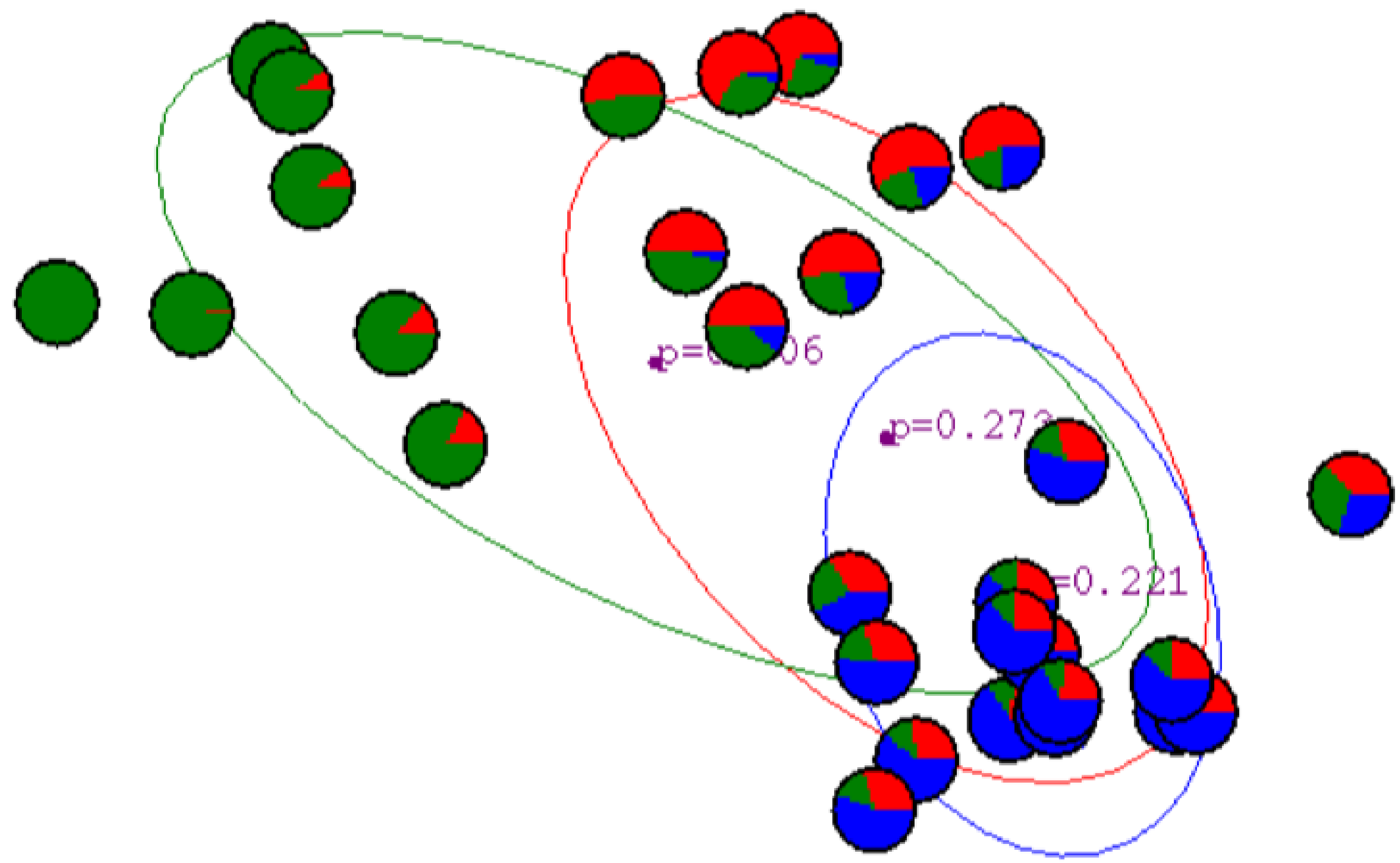
covariance_type="full"

EM for Gaussian Mixture Model:



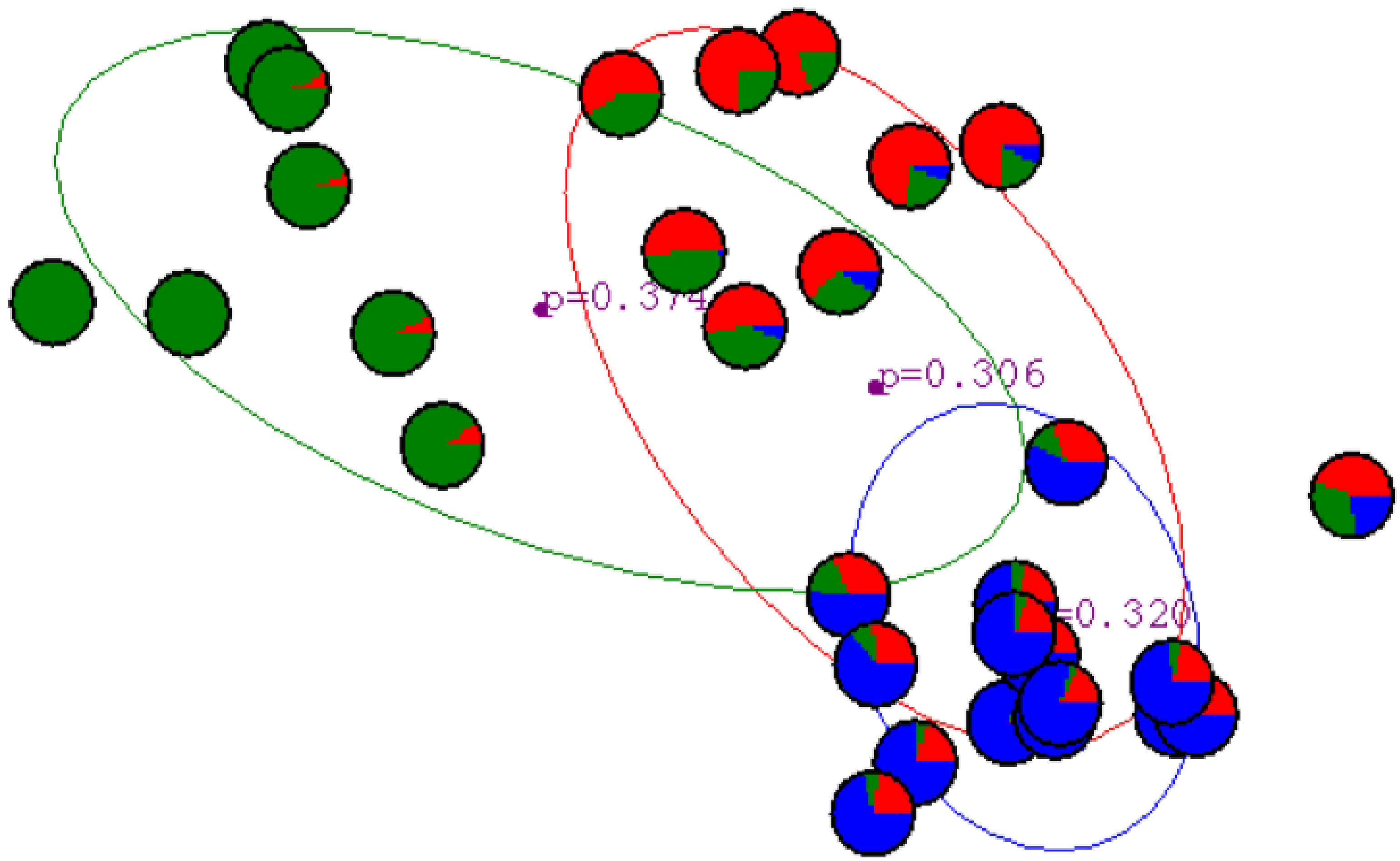
EM for Gaussian Mixture Model: Example

After 1st iteration



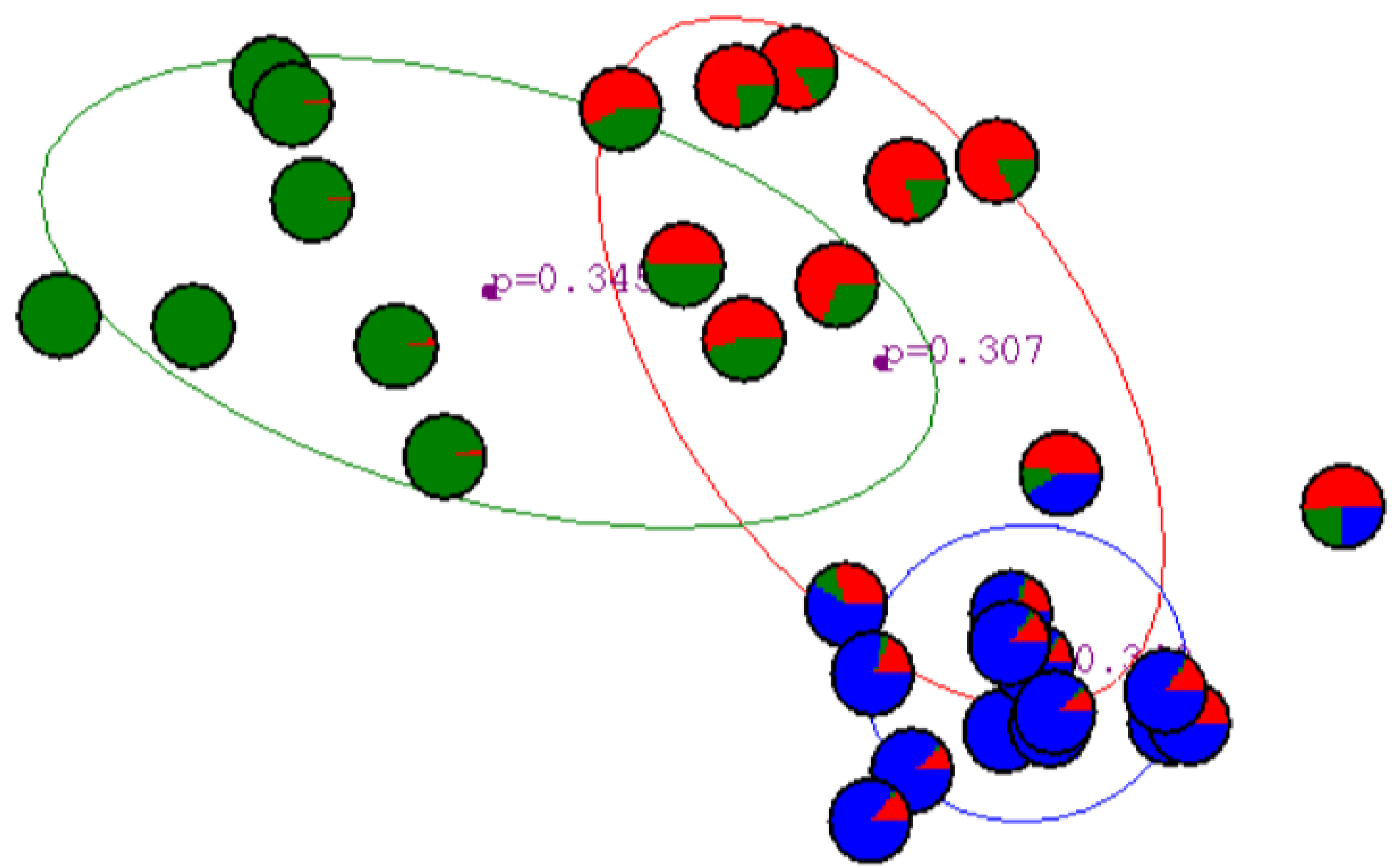
EM for Gaussian Mixture Model: Example

After 2nd iteration



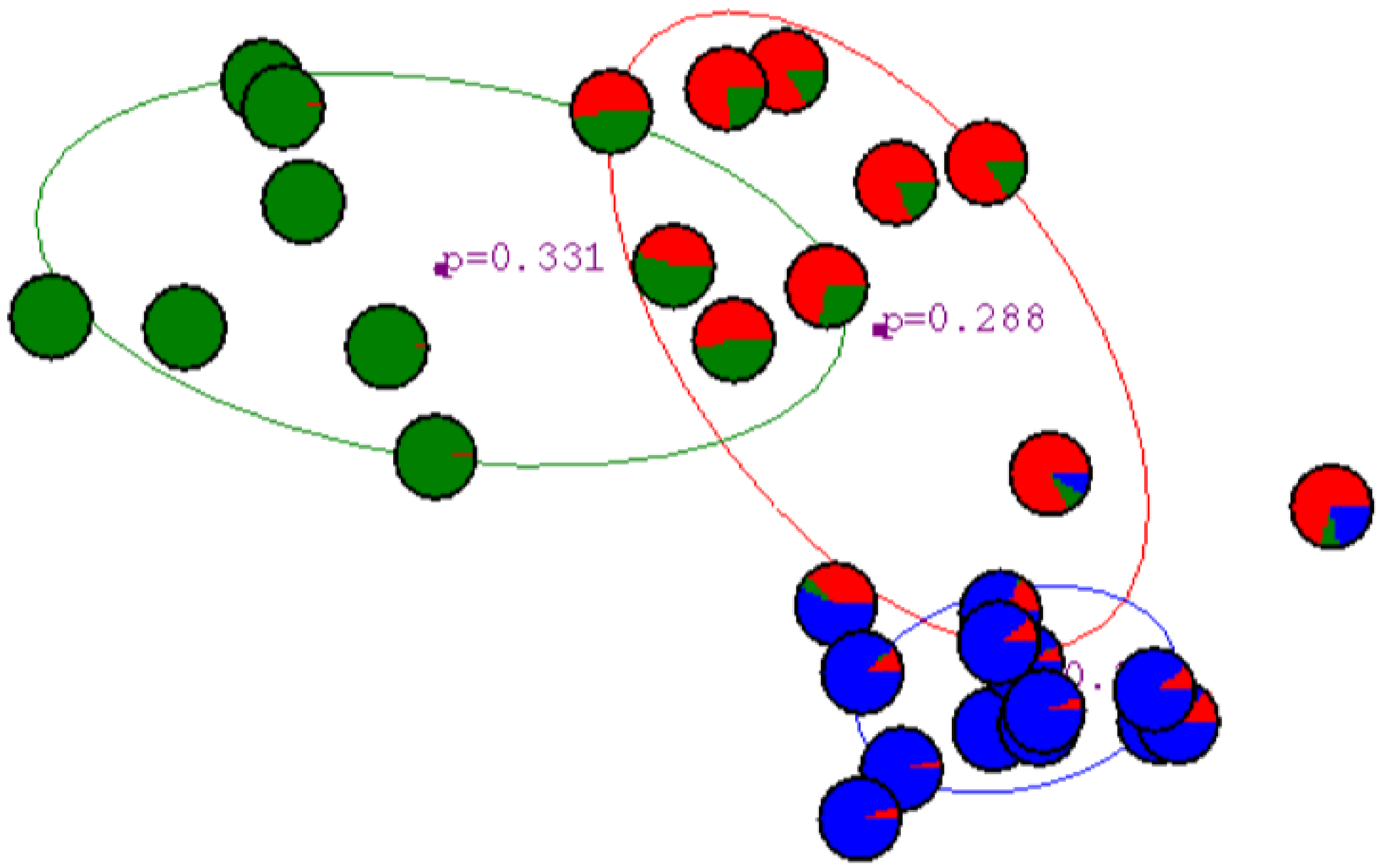
EM for Gaussian Mixture Model: Example

After 3rd iteration



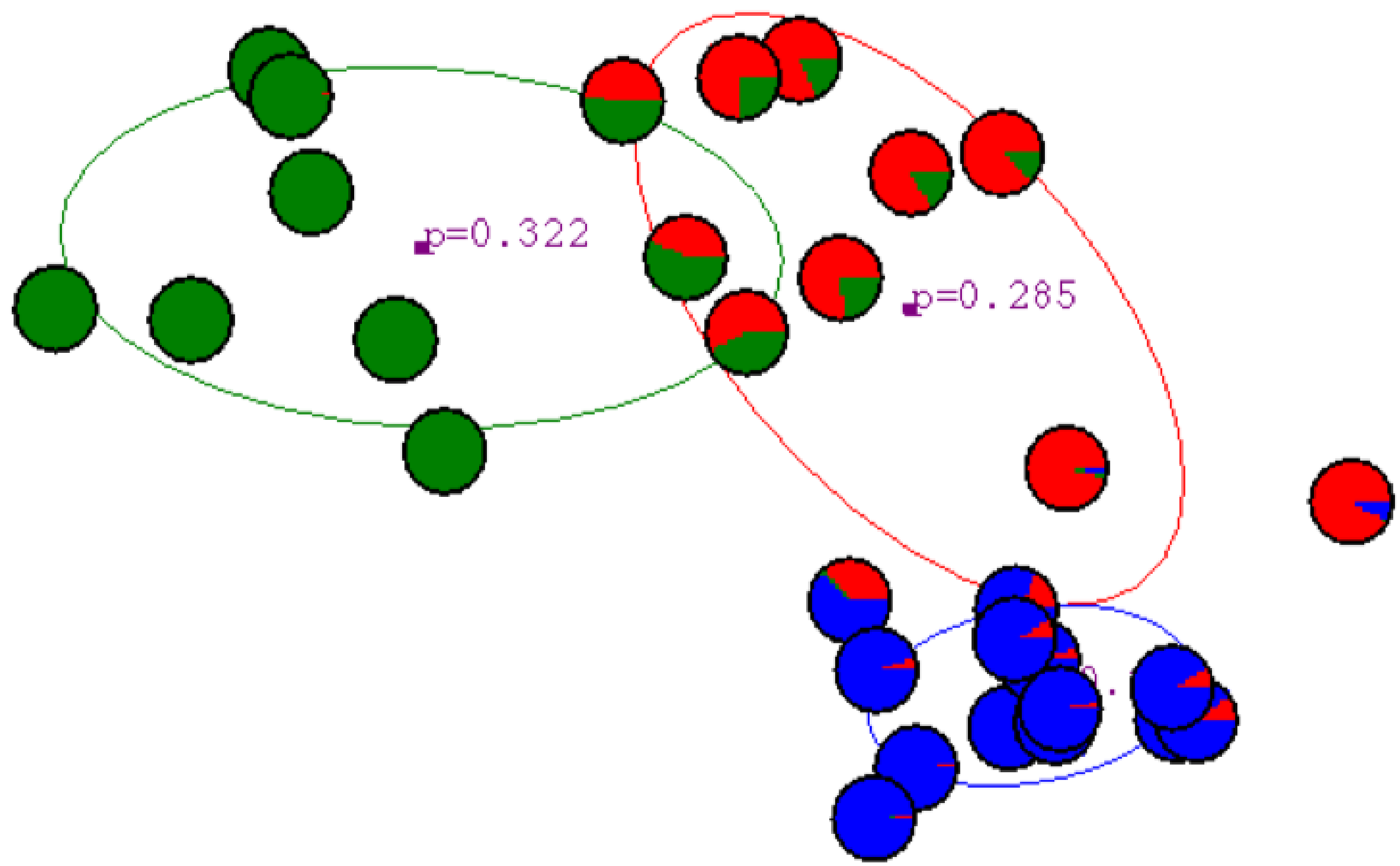
EM for Gaussian Mixture Model: Example

After 4th iteration



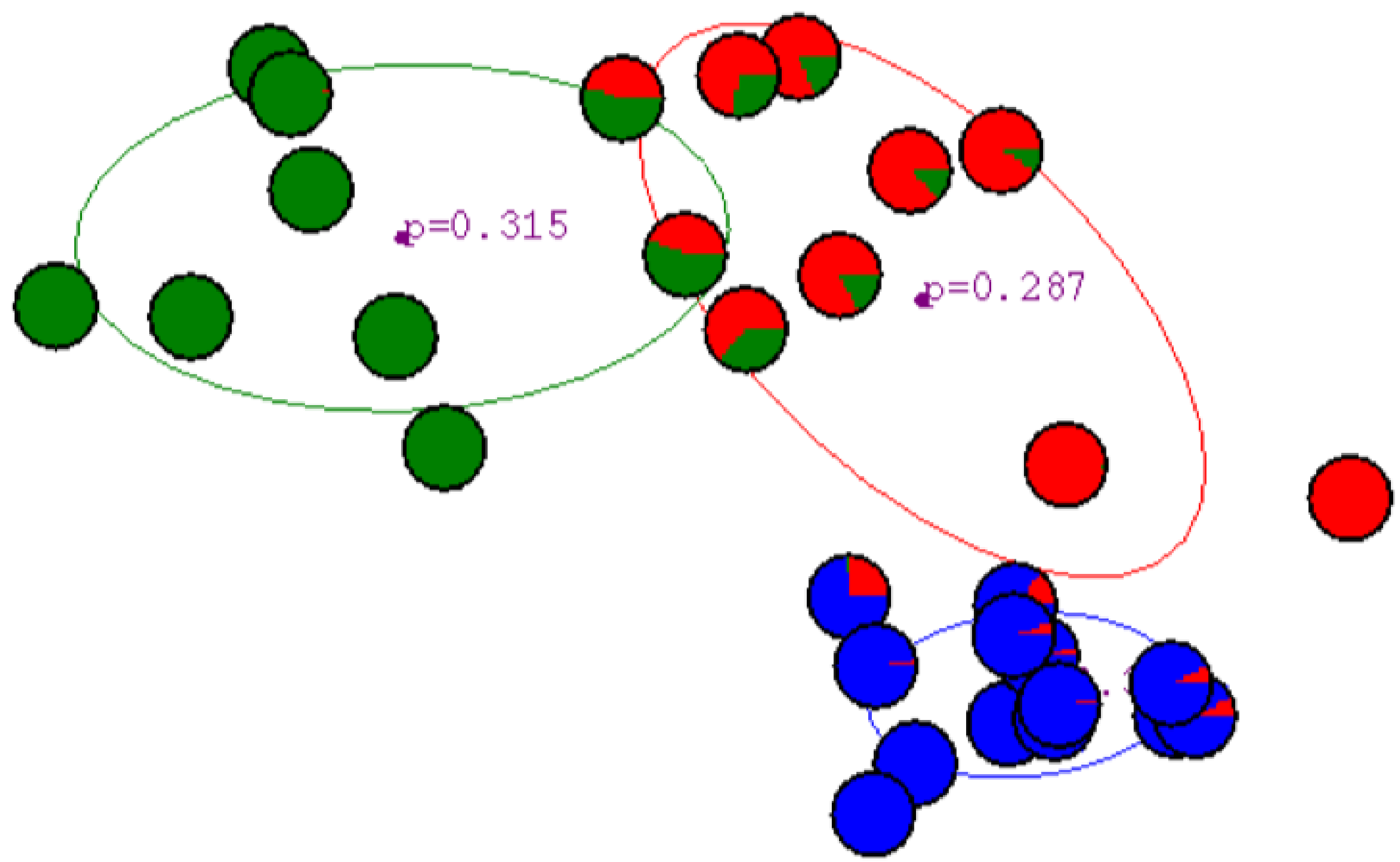
EM for Gaussian Mixture Model: Example

After 5th iteration



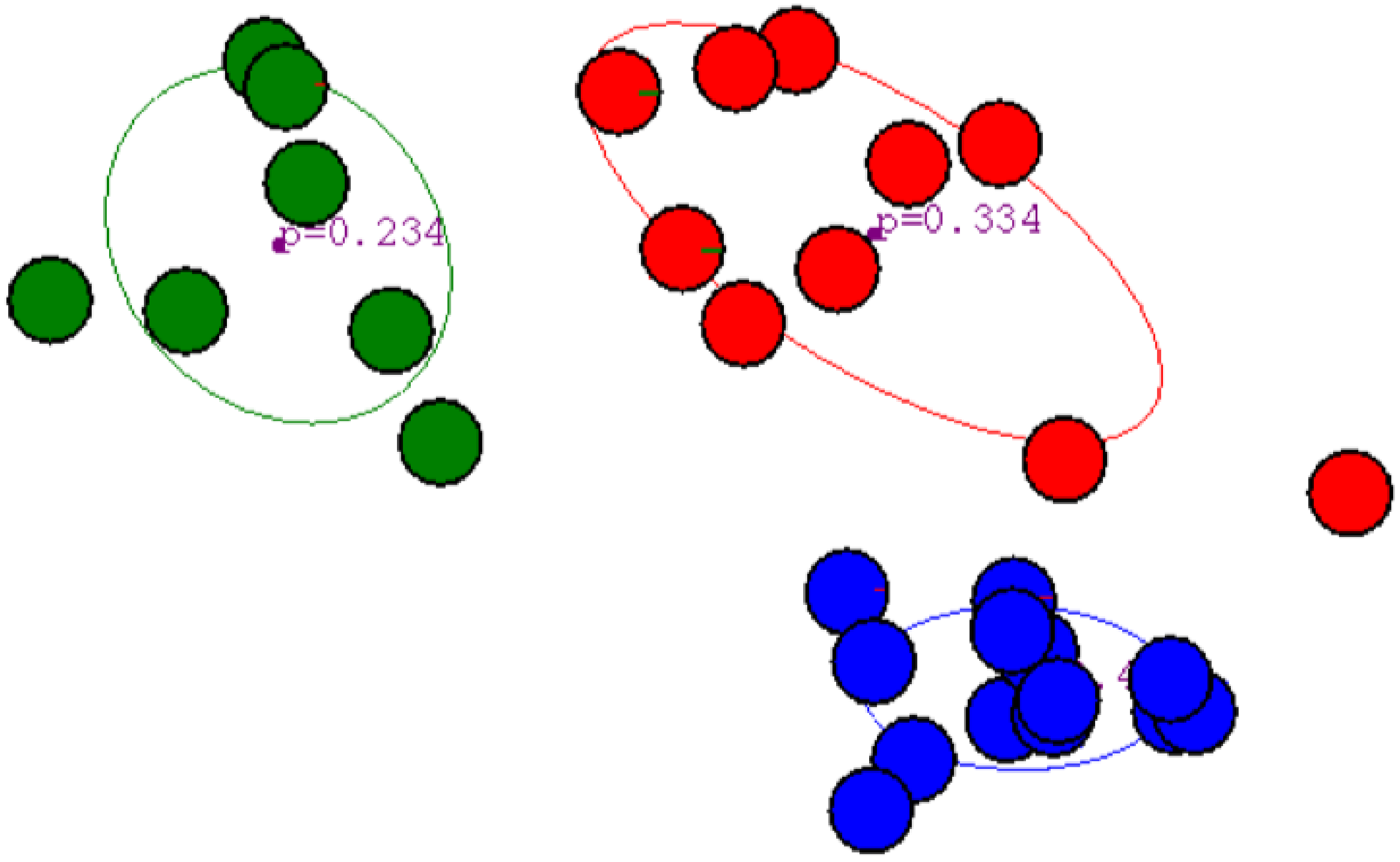
EM for Gaussian Mixture Model: Example

After 6th iteration



EM for Gaussian Mixture Model: Example

After 20th iteration



EM Algorithm for GMM (matrix form)

Given a Gaussian mixture model, the goal is to maximize the likelihood function with respect to the parameters comprising the means and covariances of the components and the mixing coefficients).

1. Initialize the means $\boldsymbol{\mu}_j$, covariances Σ_j and mixing coefficients $\boldsymbol{\pi}_j$, and evaluate the initial value of the log likelihood.
2. E step. Evaluate the responsibilities using the current parameter values

$$\gamma(z_k) = \frac{\pi_k \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_j, \Sigma_j)}$$

Book : C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006

EM Algorithm for GMM (matrix form)

3. M step. Re-estimate the parameters using the current responsibilities

$$\mu_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})\,\mathbf{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})} \qquad \Sigma_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})} \qquad \pi_k = \frac{1}{N}\sum_{n=1}^{N} \gamma(z_{nk})$$

4. Evaluate log likelihood

$$\ln p(\mathbf{X} \mid \mu, \Sigma, \pi) = \sum_{n=1}^{N} \ln\left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k) \right\}$$

If there is no convergence, return to step 2.

Book : C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006

Relationship to K-means

- K-means makes **hard** decisions.
  - Each data point gets assigned to a single cluster.
- GMM/EM makes **soft** decisions.
  - Each data point can yield a posterior $p(z|x)$
- K-means is a special case of EM.

General form of EM

- Given a joint distribution over observed and latent variables: $p(X, Z|\theta)$
- Want to maximize: $p(X|\theta)$

1. Initialize parameters θ^{old}

2. E Step: Evaluate: $p(Z|X, \theta^{old})$

3. M-Step: Re-estimate parameters (based on expectation of complete-data log likelihood)

$$\theta^{new} = \operatorname{argmax}_\theta \sum_Z p(z_k|X, \theta^{old}) \ln p(X, z_k|\theta) = argmax_\theta\ Exp[\log(p(x, z_k|\theta)]$$

1. Check for convergence of params or likelihood

$$\theta^{new} = \operatorname{argmax}_\theta \sum_k p(z_k|X, \theta^{old}) \ln p(X, z_k|\theta)$$

Jensen's inequality

$$\begin{aligned}
\ell(\theta|x) &= \log p(x \mid \theta) \\
&= \log \sum_k p(x, z_k| \theta) \\
&= \log \sum_k q(z_k| x) \frac{p(x, z_k \mid \theta)}{q(z_k| x)}
\end{aligned}$$

Will lead to maximize this

$$\ge \sum_k q(z_k| x) \log \frac{p(x, z_k \mid \theta)}{q(z_k| x)}$$

Maximizing this

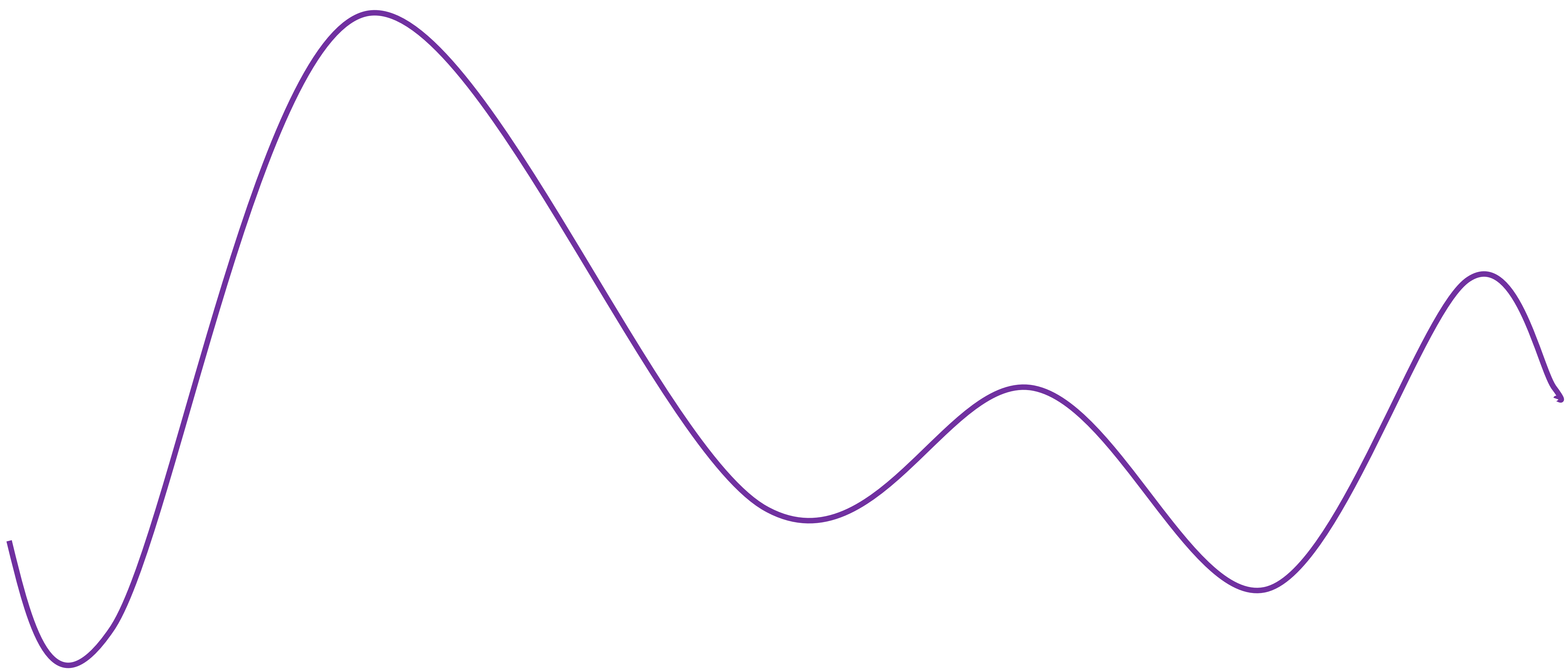
$$= \log \sum_k q(z_k|x) \frac{p(x, z_k|\theta)}{q(z_k|x)}$$

$$\geq \sum_k q(z_k|x) \log \frac{p(x, z_k|\theta)}{q(z_k|x)}$$

$$l(\theta|x) = \log p(x|\theta) = \log \sum_k p(x, z_k|\theta) \geq \sum_k q(z_k|x) \log \frac{p(x, z_k|\theta)}{q(z_k|x)}$$

$$\log\left(\sum_k p(x, z_k|\theta)\right) = \log\left(\sum_z p(x|\theta, z_k) * p(z_k|\theta)\right)$$

$$= \mathbf{log}(N(x|\mu_0, \Sigma_0) * \pi_0 + \cdots + N(x|\mu_k, \Sigma_k) * \pi_0)$$

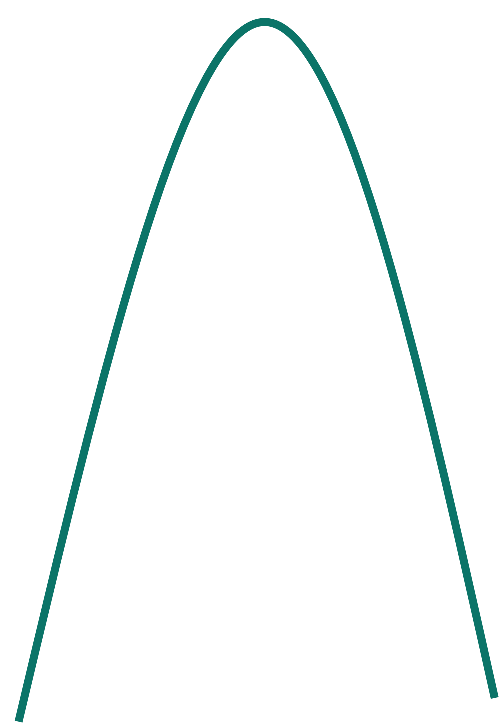


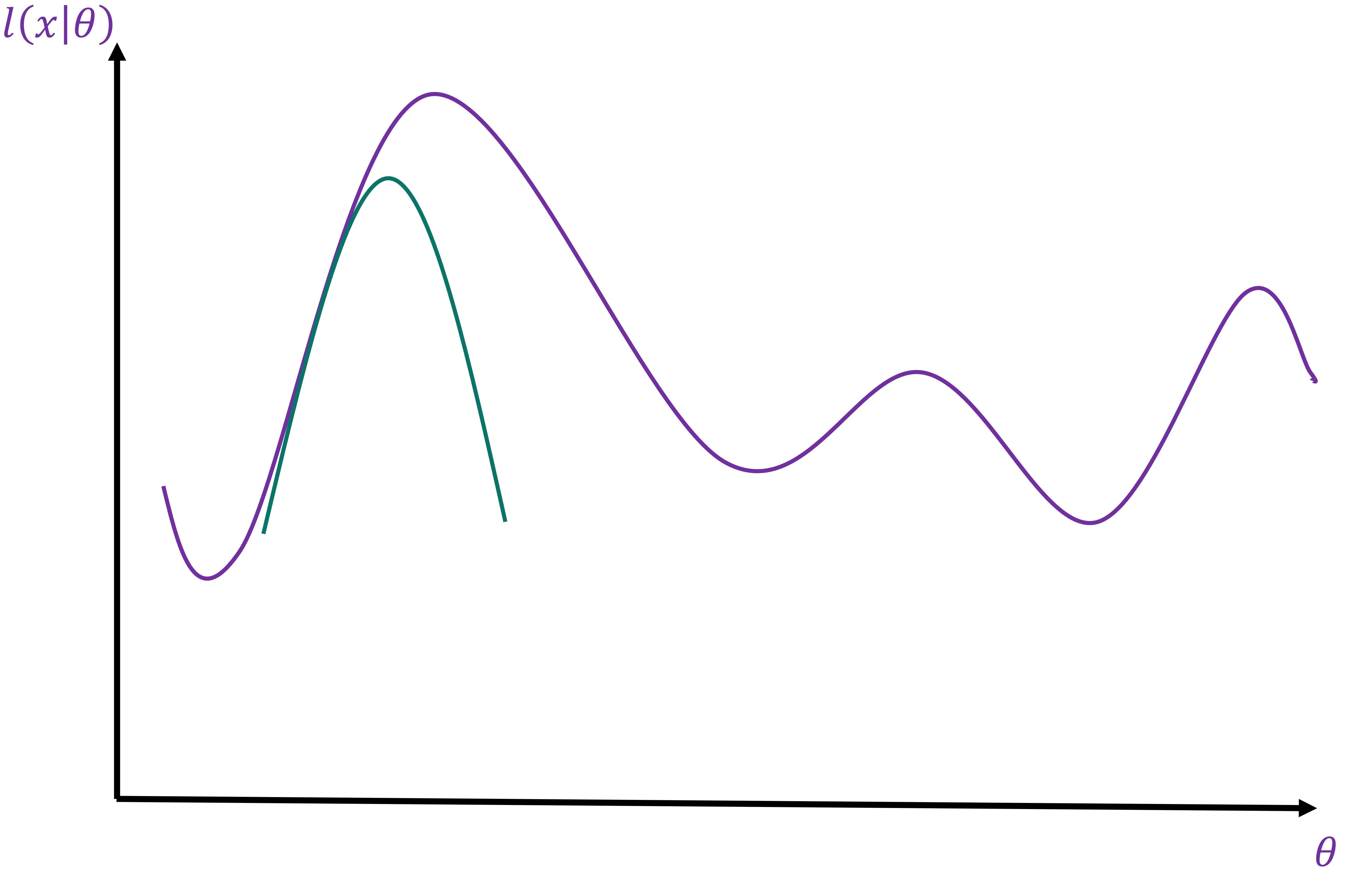
$$l(\theta|x) = \log p(x|\theta) = \log \sum_k p(x, z_k|\theta) \geq \sum_k q(z_k|x) \log \frac{p(x, z_k|\theta)}{q(z_k|x)}$$

$q(z_k|x) = C_k \Rightarrow$ It is given to us

$$\sum_k q(z_k|x) \log \frac{p(x, z_k|\theta)}{q(z_k|x)} =$$

$$C_0 \log\left(\frac{1}{C_0} * N(x|\mu_0, \Sigma_0) * \pi_0\right) + \cdots + C_k \log\left(\frac{1}{C_k} * N(x|\mu_k, \Sigma_k) * \pi_k\right)$$





$l(x|\theta)$
θ

$$l(\theta|x) = \log p(x|\theta) = \log \sum_{z} p(x, z|\theta) \geq \sum_{z} q(z|x) \log \frac{p(x, z|\theta)}{q(z|x)}$$

$$\ell(\theta|x) = \sum^{N} \sum_{k\in\{0,1\}}^{Z} p(z_k|x_n, \theta_{old}) ln\, [p(x_n, z_k|\theta)]$$

$$\log \sum_{k} P(x, z_k|\theta) \geq \sum_{k} q(z_k|x) \log \frac{P(x, z_k|\theta)}{q(z_k|x)}$$

If $\Rightarrow$ $q(z_k|x) = P(z_k|x, \theta^{old})$

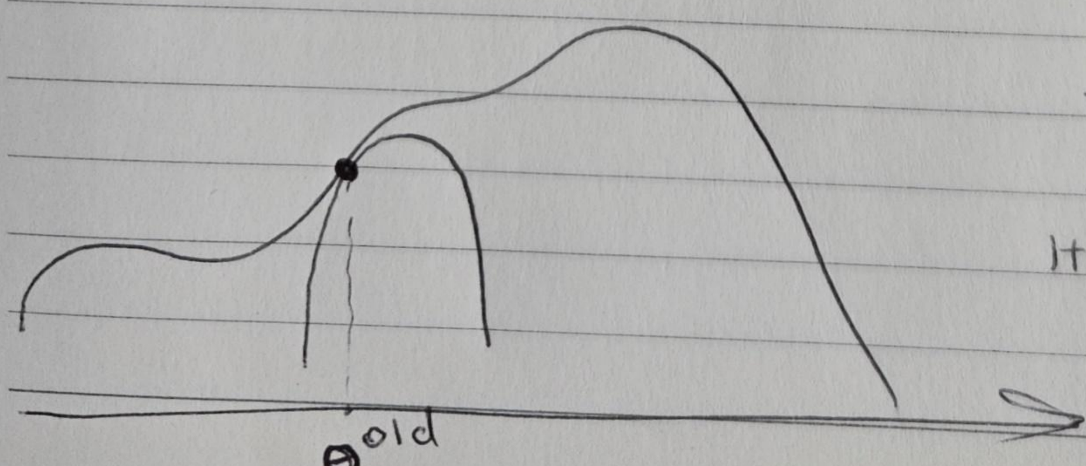
lower bound $= \sum_{k} P(z_k|x, \theta^{old}) \log \frac{P(x, z_k|\theta)}{P(z_k|x, \theta^{old})}$

lower bound $= \sum_{k} P(z_k|x, \theta^{old}) [\log P(x, z_k|\theta) - \log P(z_k|x, \theta^{old})]$

lower bound $= \sum_{k} P(z_k|x, \theta^{old}) \log P(x, z_k|\theta) + \sum_{k} P(z_k|x, \theta^{old}) * \log P(z_k|x, \theta^{old})$

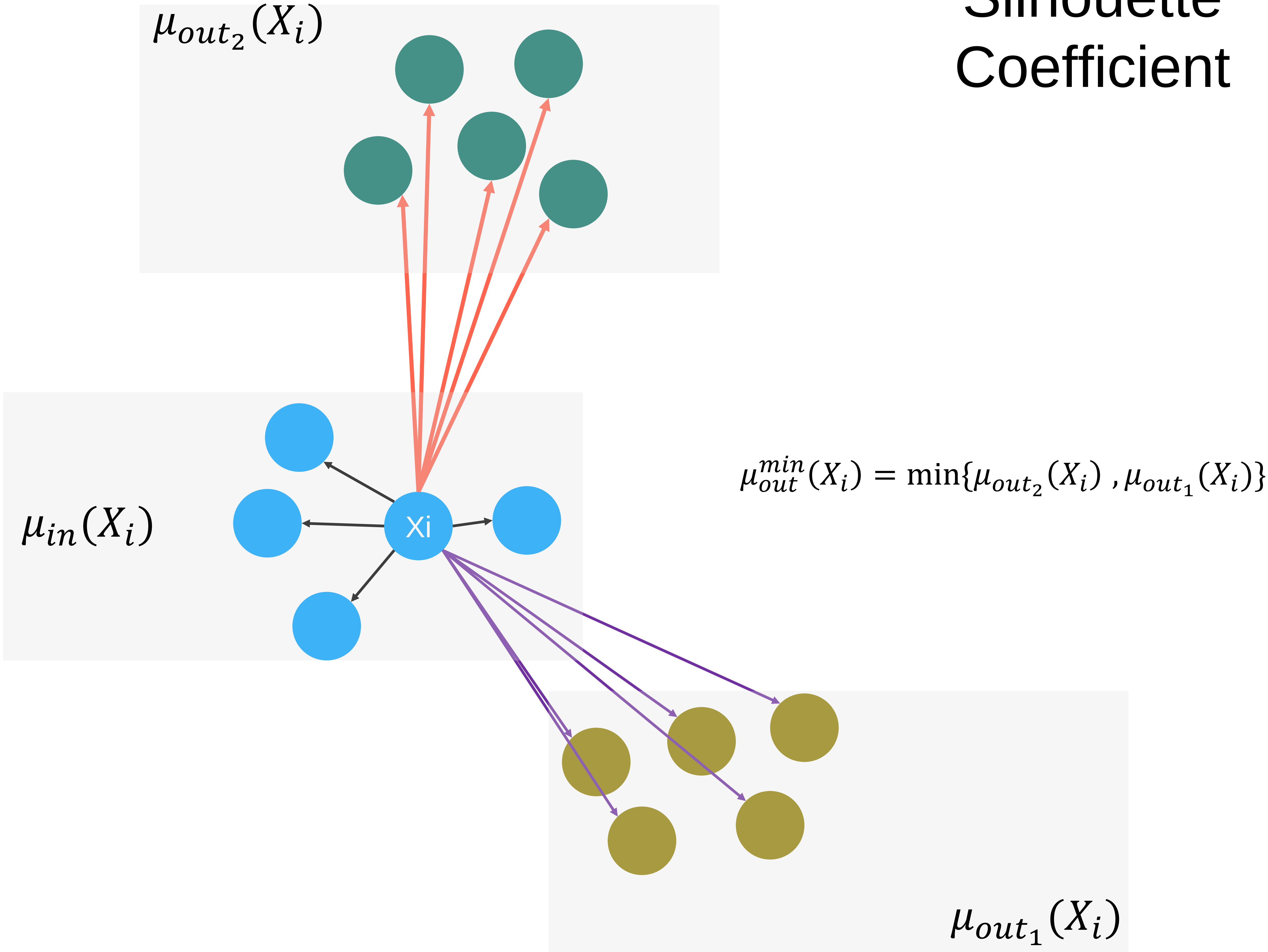
It is given to us in responsibility

It is a constant



https://www.dropbox.com/scl/fi/j0nmxf654bbluf3zowp78/EM-maximization-and-equality.mp4?rlkey=wlua7l5r88kdtydjoru7qc6f6&st=5h7rbts6&dl=0

Silhouette Coefficient



Silhouette Coefficient

Define the silhoutte coefficient of a point $\mathbf{x}_i$ as

$$s_i = \frac{\mu_{out}^{\min}(\mathbf{x}_i) - \mu_{in}(\mathbf{x}_i)}{\max\left\{\mu_{out}^{\min}(\mathbf{x}_i), \mu_{in}(\mathbf{x}_i)\right\}}$$

where $\mu_{in}(\mathbf{x}_i)$ is the mean distance from $\mathbf{x}_i$ to points in its own cluster $\hat{y}_i$:

$$\mu_{in}(\mathbf{x}_i) = \frac{\sum_{\mathbf{x}_j \in C_{\hat{y}_i}, j \neq i} \delta(\mathbf{x}_i, \mathbf{x}_j)}{n_{\hat{y}_i} - 1}$$

and $\mu_{out}^{\min}(\mathbf{x}_i)$ is the mean of the distances from $\mathbf{x}_i$ to points in the closest cluster:

$$\mu_{out}^{\min}(\mathbf{x}_i) = \min_{j \neq \hat{y}_i} \left\{ \frac{\sum_{\mathbf{y} \in C_j} \delta(\mathbf{x}_i, \mathbf{y})}{n_j} \right\}$$

The Silhouette Coefficient for clustering C: $SC = \frac{1}{n} \sum_{i=1}^{n} s_i.$

SC close to 1 implies a good clustering (Points are close to their own clusters but far from other clusters)

Take-Home Messages

- The generative process of Gaussian Mixture Model
- Inferring cluster membership based on a learned GMM
- The general idea of Expectation-Maximization
- Expectation-Maximization for GMM
- Silhouette Coefficient