K-means clustering algorithm

The solutions to each sub-problem are given by

1. $\mathbf{m}_k^* = \frac{1}{n_k} \sum_{i:C(i)=k} \mathbf{x}_i$
2. $C^*(i) = \arg\min_k \|\mathbf{x}_i - \mathbf{m}_k\|_2^2$

Algorithm

Initialize $\mathbf{m}_k, k = 1, \ldots, K$

Repeat until clusters don't change

- $C(i) = \arg\min_k \|\mathbf{x}_i - \mathbf{m}_k\|_2^2$
- $\mathbf{m}_k = \frac{1}{n_k} \sum_{i:C(i)=k} \mathbf{x}_i$

Beyond K-means clustering

In K-means clustering, by measuring the within-cluster scatter as

$$W(C) = \sum_{k=1}^{K} \sum_{i:C(i)=k} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_2^2$$

we are implicitly assuming that each cluster is roughly spherical in shape

This is probably unrealistic

But if we don't even know what the clusters are we definitely don't know how they are shaped…

Clustering with Gaussian mixture models

One way to extend the basic idea behind K-means clustering to allow for more general cluster shapes is to assume

- the clusters are ***elliptical***
- each cluster can be modeled using a ***multivariate Gaussian density***
- the full data set is modeled using a ***Gaussian mixture model*** (GMM)

We can then perform clustering based on a maximum likelihood estimation of the GMM

Gaussian mixture models

Recall the multivariate Gaussian density

$$\phi(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{d}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})}$$

where $\mathbf{x} \in \mathbb{R}^d, \boldsymbol{\mu} \in \mathbb{R}^d, \boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}, \boldsymbol{\Sigma} \succeq 0$

A random variable X follows a ***Gaussian mixture model*** if its density has the form

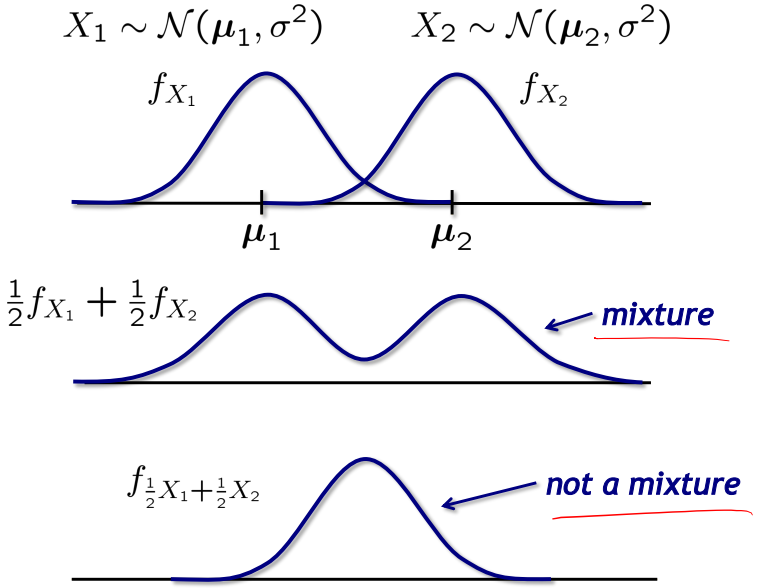
$$f(\mathbf{x}) = \sum_{k=1}^{K} w_k \phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where $0 < w_k < 1, \sum_{k=1}^{K} w_k = 1$

$\boldsymbol{\mu}_k \in \mathbb{R}^d$

$\boldsymbol{\Sigma}_k \in \mathbb{R}^{d \times d}, \boldsymbol{\Sigma}_k \succeq 0$

Example



Simulating a GMM

Suppose we know

$$\boldsymbol{\theta} = (w_1, \ldots, w_K, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$$

How can we simulate a realization of the GMM?

Basic idea

1. Choose one of the "components" at random, weighted according to w_k
   - i.e., $\mathbb{P}[\text{picking component } k] = w_k$
2. Draw a realization from $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

Why does this work?

Simulating a GMM

Let $S \in \{1, \ldots, K\}$ be a discrete random variable such that

$$\mathbb{P}[S = k] = w_k$$

Generate X as follows:

1. Generate a realization s of S
2. Generate $X \sim \mathcal{N}(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$

The density of X generated this way is

$$f(\mathbf{x}) = \sum_{k=1}^{K} f(\mathbf{x}|S = k) \cdot \mathbb{P}[S = k]$$

$$= \sum_{k=1}^{K} w_k \phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

GMMs for clustering

The variable S is called a (hidden) ***state variable***

We can imagine that every realization from a GMM is actually associated with a (hidden) realization of the state variable S

In the context of clustering, our objective is to estimate the parameters of the GMM and define clusters based on the estimated means/covariances

That is, we assume $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim f(\mathbf{x}; \boldsymbol{\theta})$ and reduce clustering to a ***parameter estimation*** problem

Of course, we have to do all of this without observing the hidden states $s_1, \ldots, s_n$ associated with $\mathbf{x}_1, \ldots, \mathbf{x}_n$

Maximum likelihood estimation

We will approach this problem from the perspective of ***maximum likelihood estimation***

- $f(\mathbf{x};\boldsymbol{\theta})$: arbitrary density parameterized by $\boldsymbol{\theta}$
- n iid realizations $\mathbf{x}_1,\ldots,\mathbf{x}_n \sim f(\mathbf{x};\boldsymbol{\theta})$ denoted by $\mathbf{X} = (\mathbf{x}_1,\ldots,\mathbf{x}_n)$

The ***likelihood function*** of $\boldsymbol{\theta}$ is

$$\mathcal{L}(\boldsymbol{\theta};\mathbf{X}) := f(\mathbf{X};\boldsymbol{\theta}) = \prod_{i=1}^{n} f(\mathbf{x}_i;\boldsymbol{\theta})$$

The ***maximum likelihood estimator (MLE)*** is

$$\widehat{\boldsymbol{\theta}} = \widehat{\boldsymbol{\theta}}(\mathbf{X}) := \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta};\mathbf{X})$$

Example

Suppose $\mathbf{x}_1,\ldots,\mathbf{x}_n \sim \mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ ➡ $\boldsymbol{\theta} = (\boldsymbol{\mu},\boldsymbol{\Sigma})$

$$\mathcal{L}(\boldsymbol{\theta};\mathbf{X}) = \prod_{i=1}^{n} \frac{e^{-\frac{1}{2}(\mathbf{x}_i-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i-\boldsymbol{\mu})}}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}|}}$$

Computationally, it is more convenient to maximize the ***log-likelihood***

$$\begin{aligned}\ell(\boldsymbol{\theta};\mathbf{X}) &= \log \mathcal{L}(\boldsymbol{\theta};\mathbf{X}) \\ &= \sum_{i=1}^{n} -\frac{1}{2}\left[d\log(2\pi) + \log|\boldsymbol{\Sigma}| \right. \\ &\qquad \left. +(\mathbf{x}_i-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i-\boldsymbol{\mu})\right]\end{aligned}$$

Example

By taking the gradient of $\ell(\boldsymbol{\theta};\mathbf{X})$ and setting this to zero, we obtain that the MLE of the parameters is given by

$$\widehat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$$

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i-\widehat{\boldsymbol{\mu}})(\mathbf{x}_i-\widehat{\boldsymbol{\mu}})^T$$

MLE for a GMM

Now consider a GMM and assume that K is known

The likelihood function is

$$\mathcal{L}(\boldsymbol{\theta};\mathbf{X}) = \prod_{i=1}^{n} f(\mathbf{x}_i;\boldsymbol{\theta}) = \prod_{i=1}^{n}\left(\sum_{k=1}^{K} w_k\phi(\mathbf{x}_i;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\right)$$

and the log likelihood is

$$\ell(\boldsymbol{\theta};\mathbf{X}) = \sum_{i=1}^{n}\log\left(\sum_{k=1}^{K} w_k\phi(\mathbf{x}_i;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\right)$$

There is (unfortunately) no known closed-form maximizer

State variables

Just for kicks, let's suppose that we actually also had access to realizations of the state variables associated with $\mathbf{X}$

Notation: $\mathrm{s} = (s_1, \ldots, s_n)$, $I_k = \{i : s_i = k\}$, $n_k = |I_k|$

The likelihood function is given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}; \mathbf{X}, \mathrm{s}) &= \prod_{i=1}^{n} f(\mathbf{x}_i|\boldsymbol{\theta}, s_i) \cdot \mathbb{P}[S_i = s_i|\boldsymbol{\theta}] \\ &= \prod_{i=1}^{n} w_{s_i} \phi(\mathbf{x}_i; \boldsymbol{\mu}_{s_i}, \boldsymbol{\Sigma}_{s_i}) \\ &= \prod_{k=1}^{K} w_k^{n_k} \prod_{i \in I_k} \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\end{aligned}$$

MLE with state variables

In this case the log-likelihood is given by

$$\ell(\boldsymbol{\theta}; \mathbf{X}, \mathrm{s}) = \sum_{k=1}^{K} n_k \log w_k + \sum_{k=1}^{K} \sum_{i \in I_k} \log \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Thus, we can maximize with respect to each $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ independently from the rest of the components

$$\widehat{\boldsymbol{\mu}}_k = \frac{1}{n_k} \sum_{i \in I_k} \mathbf{x}_i$$

$$\widehat{\boldsymbol{\Sigma}}_k = \frac{1}{n_k} \sum_{i \in I_k} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T$$

MLE with state variables

To solve

$$\max_{\mathbf{w}} \sum_{k=1}^{K} n_k \log w_k$$

$$\text{s.t.} \sum_{k=1}^{K} w_k = 1$$

we can use Lagrange multipliers

The Lagrangian is

$$L(\mathbf{w}, \lambda) = \sum_{k=1}^{K} n_k \log w_k + \lambda \left(\sum_{k=1}^{K} w_k - 1 \right)$$

MLE with state variables

$$L(\mathbf{w}, \lambda) = \sum_{k=1}^{K} n_k \log w_k + \lambda \left(\sum_{k=1}^{K} w_k - 1 \right)$$

$$\Rightarrow \frac{\partial L(\mathbf{w}, \lambda)}{\partial w_k} = \frac{n_k}{w_k} + \lambda = 0 \Rightarrow w_k = -\frac{n_k}{\lambda}$$

$$\Rightarrow \sum_{k=1}^{K} \left(-\frac{n_k}{\lambda} \right) = 1 \Rightarrow \lambda = -\sum_{k=1}^{K} n_k = -n$$

$$\Rightarrow \widehat{w}_k = \frac{n_k}{n}$$

Incomplete data

The combined data $(\mathbf{X}, \mathbf{s})$ is sometimes called the ***"complete data"***

In general, "complete data" usually refers to the combination of what we observe together with the unobserved data that we would need in order to make the MLE tractable

Unfortunately, we never have the complete data, and so we must figure out how to infer the "missing data" in order calculate the MLE

MLE with "incomplete data"

Define the "indicator" variable

$$\Delta_{i,k} = \begin{cases} 1 & \text{if } s_i = k \\ 0 & \text{if } s_i \neq k \end{cases}$$

The ***complete data log-likelihood*** can be written as

$$\begin{aligned} \ell(\boldsymbol{\theta}; \mathbf{X}, \mathbf{s}) &= \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \Delta_{i,k} \cdot w_k \cdot \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) \\ &= \sum_{i=1}^{n} \sum_{k=1}^{K} \Delta_{i,k} \left(\log w_k + \log \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) \end{aligned}$$

Expectation-Maximization (EM)

The ***expectation-maximization (EM)*** algorithm is an iterative algorithm that produces a sequence $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \ldots$ of parameter estimates

E-step

Given $\boldsymbol{\theta}^{(j)}$, compute the ***expected complete data log-likelihood***:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(j)}) = \mathbb{E}_{\mathbf{S}|\mathbf{X}} \left[\ell(\boldsymbol{\theta}; \mathbf{X}, \mathbf{S}) | \mathbf{X}; \boldsymbol{\theta}^{(j)} \right]$$

In our case

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(j)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{i,k}(\boldsymbol{\theta}^{(j)}) \left(\log w_k + \log \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right)$$

where

$$\gamma_{i,k}(\boldsymbol{\theta}^{(j)}) := \mathbb{E}_{\mathbf{S}|\mathbf{X}} \left[\Delta_{i,k} | \mathbf{X}; \boldsymbol{\theta}^{(j)} \right]$$

E-step for GMMs

In the E-step we need to compute

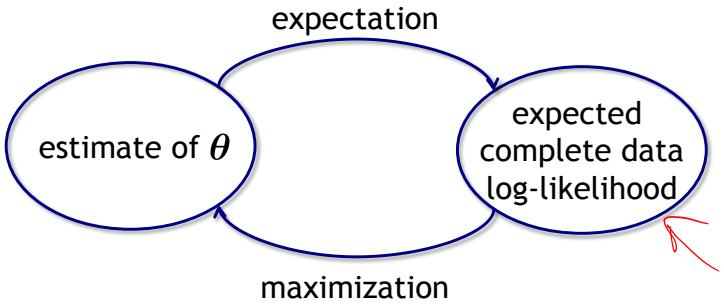
$$\begin{aligned} \gamma_{i,k}^{(j)} &= \mathbb{E}_{\mathbf{S}|\mathbf{X}} \left[\Delta_{i,k} | \mathbf{X}; \boldsymbol{\theta}^{(j)} \right] \\ &= \mathbb{P} \left[\Delta_{i,k} = 1 | \mathbf{X}; \boldsymbol{\theta}^{(j)} \right] \\ &= \mathbb{P} \left[S_i = k | \mathbf{x}_i; \boldsymbol{\theta}^{(j)} \right] \qquad \text{Bayes' rule} \\ &= \frac{f(\mathbf{x}_i | S_i = k; \boldsymbol{\theta}^{(j)}) \mathbb{P} \left[S_i = k; \boldsymbol{\theta}^{(j)} \right]}{f(\mathbf{x}_i; \boldsymbol{\theta}^{(j)})} \\ &= \frac{w_k^{(j)} \phi(\mathbf{x}_i; \boldsymbol{\mu}_k^{(j)}, \boldsymbol{\Sigma}_k^{(j)})}{\sum_{\ell=1}^{K} w_\ell^{(j)} \phi(\mathbf{x}_i; \boldsymbol{\mu}_\ell^{(j)}, \boldsymbol{\Sigma}_\ell^{(j)})} \end{aligned}$$

Expectation-Maximization (EM)

M-step

Given the expected complete data log-likelihood, compute the maximum likelihood estimate

$$\boldsymbol{\theta}^{(j+1)} = \arg\max_{\boldsymbol{\theta}} \; Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(j)})$$



M-step for GMMs

Maximizing

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(j)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{i,k}^{(j)} (\log w_k + \log \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k))$$

with respect to $\boldsymbol{\theta} = (\{w_k\}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\})$ yields

$$\widehat{w}_k = \frac{\sum_{i=1}^{n} \gamma_{i,k}^{(j)}}{n}$$

$$\widehat{\boldsymbol{\mu}}_k = \frac{\sum_{i=1}^{n} \gamma_{i,k}^{(j)} \mathbf{x}_i}{\sum_{i=1}^{n} \gamma_{i,k}^{(j)}}$$

$$\widehat{\boldsymbol{\Sigma}}_k = \frac{\sum_{i=1}^{n} \gamma_{i,k}^{(j)} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k^{(j+1)})(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k^{(j+1)})^T}{\sum_{i=1}^{n} \gamma_{i,k}^{(j)}}$$

EM algorithm for GMMs

Initialize $w_k^{(0)}, \boldsymbol{\mu}_k^{(0)}, \boldsymbol{\Sigma}_k^{(0)}$

Repeat unit termination criterion satisfied

E-Step: Compute

$$\gamma_{i,k}^{(j)} = \frac{w_k^{(j)} \phi(\mathbf{x}_i; \boldsymbol{\mu}_k^{(j)}, \boldsymbol{\Sigma}_k^{(j)})}{\sum_{\ell=1}^{K} w_\ell^{(j)} \phi(\mathbf{x}_i; \boldsymbol{\mu}_\ell^{(j)}, \boldsymbol{\Sigma}_\ell^{(j)})}$$

M-Step: Compute

$$w_k^{(j+1)} = \frac{\sum_{i=1}^{n} \gamma_{i,k}^{(j)}}{n} \qquad \boldsymbol{\mu}_k^{(j+1)} = \frac{\sum_{i=1}^{n} \gamma_{i,k}^{(j)} \mathbf{x}_i}{\sum_{i=1}^{n} \gamma_{i,k}^{(j)}}$$

$$\boldsymbol{\Sigma}_k^{(j+1)} = \frac{\sum_{i=1}^{n} \gamma_{i,k}^{(j)} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k^{(j+1)})(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k^{(j+1)})^T}{\sum_{i=1}^{n} \gamma_{i,k}^{(j)}}$$

Initialization and termination

In general, the likelihood has many local maxima, so a good initialization of the algorithm is critical

A good initialization for EM in the case of GMM is

$$w_k^{(0)} = \tfrac{1}{K}$$

$$\boldsymbol{\mu}_k^{(0)} = \text{ a randomly selected } \mathbf{x}_i \quad \text{(sampled without replacement)}$$

$$\boldsymbol{\Sigma}_k^{(0)} = \text{ the sample covariance}$$

Possible termination criteria are to stop iterating when

$$|\ell(\boldsymbol{\theta}^{(j+1)}; \mathbf{X}) - \ell(\boldsymbol{\theta}^{(j)}; \mathbf{X})| \leq \epsilon$$

or

$$|Q(\boldsymbol{\theta}^{(j+1)}, \boldsymbol{\theta}^{(j)}) - Q(\boldsymbol{\theta}^{(j)}, \boldsymbol{\theta}^{(j)})| \leq \epsilon$$

Defining clusters

Recall that

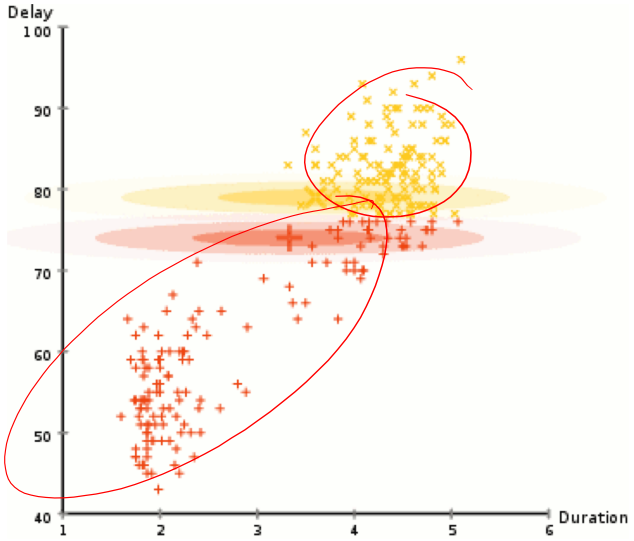
$$\gamma_{i,k}(\boldsymbol{\theta}^{(j)}) = \mathbb{P}\left[S_i = k | \mathbf{X}; \boldsymbol{\theta}^{(j)}\right]$$

A reasonable "hard" assignment of points to clusters is given by

$$C(i) = \arg\max_{k=1,\ldots,K} \gamma_{i,k}(\widehat{\boldsymbol{\theta}})$$

Alternatively, one may simply take $\gamma_{i,k}(\widehat{\boldsymbol{\theta}})$ as a "soft" assignment that expresses the "affinity" of $\mathbf{x}_i$ for cluster k

Example: Eruptions of "Old Faithful"



EM in general

Nothing in the EM algorithm as we have stated it is specific to GMMs

EM is actually an extremely general algorithm for computing ML/MAP estimators

Applies whenever having knowledge of certain "hidden variables" renders an ML/MAP estimator tractable

See *Statistical Analysis with Missing Data* by Little and Rubin (2002) for an in-depth discussion of other applications

Convergence of EM

Theorem

For each $j = 1, 2, \ldots$

$$\ell(\boldsymbol{\theta}^{(j+1)}; \mathbf{X}) \geq \ell(\boldsymbol{\theta}^{(j)}; \mathbf{X})$$

A proof based on Jensen's inequality is available in Hastie, Tibshirani, and Friedman

The convergence rate is also of interest

It is typically linear, but with a rate that depends on the proportion of observed data

Connection to K-means

Consider a GMM where each $\boldsymbol{\Sigma}_k = \sigma^2 \mathbf{I}$ for some fixed σ^2

The EM algorithm for computing the MLE of $\{w_k\}_{k=1}^K$ and $\{\boldsymbol{\mu}_k\}_{k=1}^K$ is to iterate

$$\gamma_{i,k} = \frac{w_k \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \sigma^2 \mathbf{I})}{\sum_{\ell=1}^K w_\ell \phi(\mathbf{x}_i; \boldsymbol{\mu}_\ell, \sigma^2 \mathbf{I})}$$

$$w_k = \tfrac{1}{n} \textstyle\sum_{i=1}^n \gamma_{i,k}$$

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^n \gamma_{i,k} \mathbf{x}_i}{\sum_{i=1}^n \gamma_{i,k}}$$

When $\sigma^2 \to 0$, $\gamma_{i,k} \Longrightarrow \begin{cases} 1 & \text{if } k = \arg\min_\ell \|\mathbf{x}_i - \boldsymbol{\mu}_\ell\|_2 \\ 0 & \text{otherwise} \end{cases}$

so the algorithm reduces to K-means