EM Algorithm for Latent Variable Models

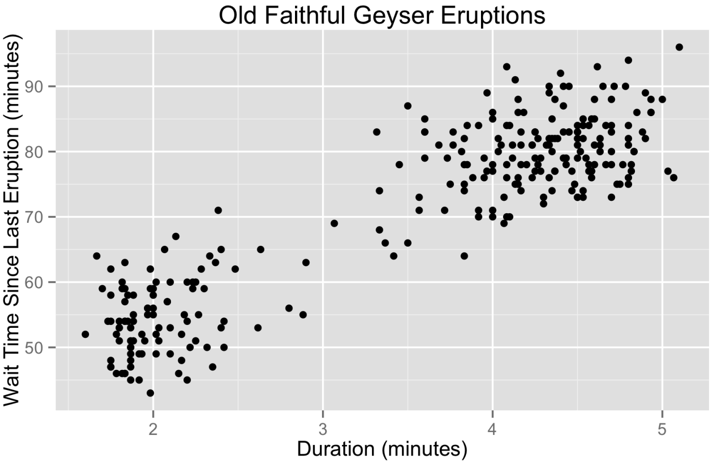
David Rosenberg

New York University

May 2, 2017

Gaussian Mixture Models (Review)

Example: Old Faithful Geyser

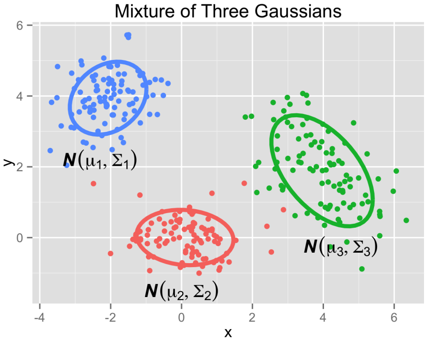


Probabilistic Model for Clustering

- Let's consider a **generative model** for the data.
- Suppose
  1. There are k clusters.
  2. We have a probability density for each cluster.
- Generate a point x as follows
  1. Choose a random cluster $z \in \{1, 2, \dots, k\}$.
  2. Choose a point x from the distribution for cluster z.

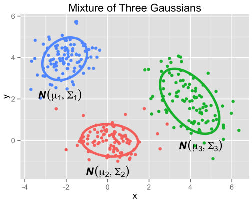
Gaussian Mixture Model ($k = 3$)

1. Choose $z \in \{1, 2, 3\}$ with $p(1) = p(2) = p(3) = \frac{1}{3}$.
2. Choose $x \mid z \sim \mathcal{N}(X \mid \mu_z, \Sigma_z)$.



Gaussian Mixture Model Parameters (k Components)

$$\begin{aligned} \text{Cluster probabilities}: &\quad \pi = (\pi_1, \ldots, \pi_k) \\ \text{Cluster means}: &\quad \mu = (\mu_1, \ldots, \mu_k) \\ \text{Cluster covariance matrices:} &\quad \Sigma = (\Sigma_1, \ldots \Sigma_k) \end{aligned}$$



For now, **suppose all these parameters are known**.
We'll discuss how to **learn** or **estimate** them later.

The GMM "Inference" Problem

- *Suppose* we know all the model parameters π, μ, Σ, and thus $p(x,z)$.
- The **inference problem**: We observe x. We want to know its cluster z.
- We can get a **soft cluster assignment** from the conditional distribution:

$$p(z \mid x) = p(x,z)/p(x)$$

- A **hard cluster assignment** is given by

$$z^* = \underset{z \in \{1,\ldots,k\}}{\arg\max}\, p(z \mid x).$$

- So if we know the model parameters, we can compute $p(z \mid x)$, and clustering is trival.

The GMM "Learning" Problem

- Given data $x_1, \ldots, x_n$ drawn i.i.d. from a GMM,
- Estimate the parameters:

$$\begin{aligned} \text{Cluster probabilities}: \quad & \pi = (\pi_1, \ldots, \pi_k) \\ \text{Cluster means}: \quad & \mu = (\mu_1, \ldots, \mu_k) \\ \text{Cluster covariance matrices:} \quad & \Sigma = (\Sigma_1, \ldots \Sigma_k) \end{aligned}$$

- Traditional approach is maximum [marginal] likelihood:

$$(\pi, \mu, \Sigma) = \underset{\pi, \mu, \Sigma}{\arg\max}\, p(x_1, \ldots, x_n)$$

- Unfortunately, this is very difficult.
- Note that the fully observed problem is easy. That is:

$$(\pi, \mu, \Sigma) = \underset{\pi, \mu, \Sigma}{\arg\max}\, p(x_1, \ldots, x_n, z_1, \ldots, z_n)$$

EM Algorithm for Latent Variable Models

General Latent Variable Model

- Two sets of random variables: z and x.
- z consists of unobserved **hidden variables**.
- x consists of **observed variables**.
- Joint probability model parameterized by $\theta \in \Theta$:

$$p(x, z \mid \theta)$$

Definition

A **latent variable model** is a probability model for which certain variables are never observed.

e.g. The Gaussian mixture model is a latent variable model.

Complete and Incomplete Data

- Suppose we have a data set $\mathcal{D} = (x_1, \ldots, x_n)$.
- To simplify notation, take x to represent the entire dataset

$$x = (x_1, \ldots, x_n),$$

  and z to represent the corresponding unobserved variables

$$z = (z_1, \ldots, z_n).$$

- An observation of x is called an **incomplete data set**.
- An observation (x, z) is called a **complete data set**.

Our Objectives

- **Learning problem**: Given incomplete dataset $\mathcal{D} = x = (x_1, \ldots, x_n)$, find MLE

$$\hat{\theta} = \underset{\theta}{\arg\max}\, p(\mathcal{D} \mid \theta).$$

- **Inference problem**: Given x, find conditional distribution over z:

$$p\left(z_i \mid x_i, \theta\right).$$

- For Gaussian mixture model, learning is hard, inference is easy.
- For more complicated models, inference can also be hard. (See DSGA-1005)

Log-Likelihood and Terminology

- Note that
$$\arg\max_{\theta} p(x \mid \theta) = \arg\max_{\theta} \left[\log p(x \mid \theta)\right].$$
- Often easier to work with this "**log-likelihood**".
- We often call $p(x)$ the **marginal likelihood**,
  - because it is $p(x,z)$ with z "marginalized out":
$$p(x) = \sum_{z} p(x,z)$$
- We often call $p(x,y)$ the **joint**. (for "joint distribution")
- Similarly, $\log p(x)$ is the **marginal log-likelihood**.

The EM Algorithm **Key Idea**

- Marginal log-likelihood is hard to optimize:

$$\max_{\theta} \log p(x \mid \theta)$$

- **Typically** the complete data log-likelihood is easy to optimize:

$$\max_{\theta} \log p(x, z \mid \theta)$$

- What if we had a **distribution** $q(z)$ for the latent variables z?
- Then maximize the **expected complete data log-likelihood**:

$$\max_{\theta} \sum_{z} q(z) \log p(x, z \mid \theta)$$

- EM **assumes** this maximization is relatively easy.

Lower Bound for Marginal Log-Likelihood

- Let $q(z)$ be any PMF on $\mathcal{Z}$, the support of z:

$$
\begin{aligned}
\log p(x \mid \theta) &= \log\left[\sum_z p(x, z \mid \theta)\right] \\
&= \log\left[\sum_z q(z)\left(\frac{p(x, z \mid \theta)}{q(z)}\right)\right] \qquad \text{(log of an expectation)} \\
&\geqslant \underbrace{\sum_z q(z) \log\left(\frac{p(x, z \mid \theta)}{q(z)}\right)}_{\mathcal{L}(q,\theta)} \qquad \text{(expectation of log)}
\end{aligned}
$$

- Inequality is by Jensen's, by concavity of the log.

This inequality is the basis for "**variational methods**", of which EM is a basic example.

The ELBO

- For any PMF $q(z)$, we have a lower bound on the marginal log-likelihood

$$\log p(x \mid \theta) \geqslant \underbrace{\sum_z q(z) \log\left(\frac{p(x, z \mid \theta)}{q(z)}\right)}_{\mathcal{L}(q,\theta)}$$

- Marginal log likelihood $\log p(x \mid \theta)$ also called the **evidence**.
- $\mathcal{L}(q, \theta)$ is the **evidence lower bound**, or "**ELBO**".

In EM algorithm (and variational methods more generally), we maximize $\mathcal{L}(q, \theta)$ over q and θ.

MLE, EM, and the ELBO

- For any PMF $q(z)$, we have a lower bound on the marginal log-likelihood

$$\log p(x \mid \theta) \geqslant \mathcal{L}(q, \theta).$$

- The MLE is defined as a maximum over θ:

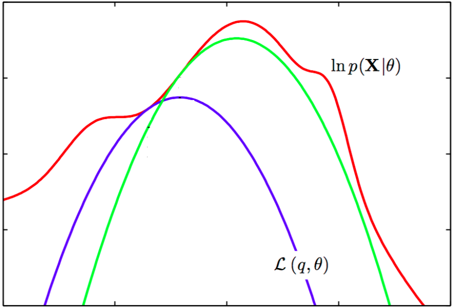
$$\hat{\theta}_{\text{MLE}} = \underset{\theta}{\arg\max} \log p(x \mid \theta).$$

- In EM algorithm, we maximize the lower bound (ELBO) over θ and q:

$$\hat{\theta}_{\text{EM}} = \underset{\theta}{\arg\max} \left[\max_{q} \mathcal{L}(q, \theta) \right]$$

A Family of Lower Bounds

- For each q, we get a lower bound function: $\log p(x \mid \theta) \geqslant \mathcal{L}(q, \theta)\ \forall \theta$.
- Two lower bounds (blue and green curves), **as functions of** θ:



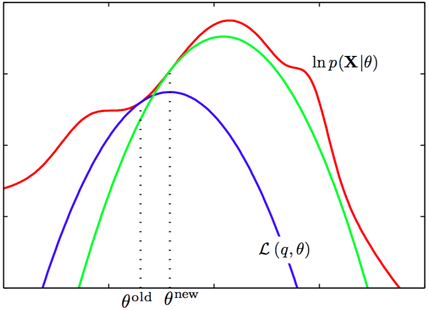
- Ideally, we'd find the maximum of the red curve. Maximum of green is close.

From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

EM: Coordinate Ascent on Lower Bound

- Choose sequence of q's and θ's by "**coordinate ascent**".
- EM Algorithm (high level):
  1. Choose initial θ^{old}.
  2. Let $q^* = \arg\max_q \mathcal{L}(q, \theta^{\text{old}})$
  3. Let $\theta^{\text{new}} = \arg\max_\theta \mathcal{L}(q^*, \theta^{\text{old}})$.
  4. Go to step 2, until converged.
- Will show: $p(x \mid \theta^{\textbf{new}}) \geqslant p(x \mid \theta^{\textbf{old}})$
- **Get sequence of θ's with monotonically increasing likelihood.**

EM: Coordinate Ascent on Lower Bound



1. Start at θ^{old}.
2. Find q giving best lower bound at $\theta^{\text{old}} \implies \mathcal{L}(q, \theta)$.
3. $\theta^{\text{new}} = \arg\max_\theta \mathcal{L}(q, \theta)$.

From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

EM: Next Steps

- We now give 2 different re-expressions of $\mathcal{L}(q,\theta)$ that make it easy to compute
  - $\arg\max_q \mathcal{L}(q,\theta)$, for a given θ, and
  - $\arg\max_\theta \mathcal{L}(q,\theta)$, for a given q.

ELBO in Terms of KL Divergence and Entropy

- Let's investigate the lower bound:

$$\begin{aligned}
\mathcal{L}(q,\theta) &= \sum_z q(z)\log\left(\frac{p(x,z\mid\theta)}{q(z)}\right) \\
&= \sum_z q(z)\log\left(\frac{p(z\mid x,\theta)p(x\mid\theta)}{q(z)}\right) \\
&= \sum_z q(z)\log\left(\frac{p(z\mid x,\theta)}{q(z)}\right) + \sum_z q(z)\log p(x\mid\theta) \\
&= -\mathrm{KL}[q(z), p(z\mid x,\theta)] + \log p(x\mid\theta)
\end{aligned}$$

- Amazing! We get back an equality for the marginal likelihood:

$$\log p(x\mid\theta) = \mathcal{L}(q,\theta) + \mathrm{KL}[q(z), p(z\mid x,\theta)]$$

Maxizing over q for fixed $\theta = \theta^{\text{old}}$.

- Find q maximizing

$$\mathcal{L}(q, \theta^{\text{old}}) = -\mathrm{KL}[q(z), p(z \mid x, \theta^{\text{old}})] + \underbrace{\log p(x \mid \theta^{\text{old}})}_{\text{no } q \text{ here}}$$

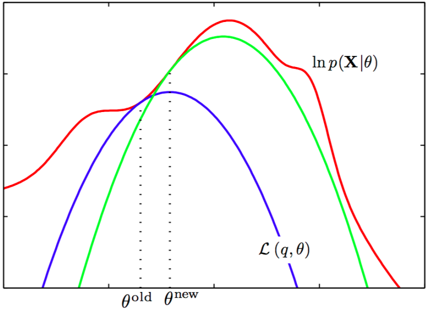
- Recall $\mathrm{KL}(p \| q) \geqslant 0$, and $\mathrm{KL}(p \| p) = 0$.
- Best q is $q^*(z) = p(z \mid x, \theta^{\text{old}})$ and

$$\mathcal{L}(q^*, \theta^{\text{old}}) = -\underbrace{\mathrm{KL}[p(z \mid x, \theta^{\text{old}}), p(z \mid x, \theta^{\text{old}})]}_{=0} + \log p(x \mid \theta^{\text{old}})$$

- Summary:

$$\begin{aligned} \log p(x \mid \theta^{\text{old}}) &= \mathcal{L}(q^*, \theta^{\text{old}}) \quad (\text{tangent at } \theta^{\text{old}}). \\ \log p(x \mid \theta) &\geqslant \mathcal{L}(q^*, \theta) \quad \forall \theta \end{aligned}$$

Tight lower bound for any chosen θ



For θ^{old}, take $q(z) = p(z \mid x, \theta^{\text{old}})$. Then

1. $\log p(x \mid \theta) \geqslant \mathcal{L}(q, \theta)\ \forall \theta$. [Global lower bound].
2. $\log p(x \mid \theta^{\text{old}}) = \mathcal{L}(q, \theta^{\text{old}})$. [Lower bound is **tight** at $\theta^{\textbf{old}}$.]

From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

Maximizing over θ for fixed q

- Consider maximizing the lower bound $\mathcal{L}(q,\theta)$:

$$
\begin{aligned}
\mathcal{L}(q,\theta) &= \sum_z q(z)\log\left(\frac{p(x,z\mid\theta)}{q(z)}\right) \\
&= \underbrace{\sum_z q(z)\log p(x,z\mid\theta)}_{\mathbb{E}[\text{complete data log-likelihood}]} \underbrace{-\sum_z q(z)\log q(z)}_{\text{no }\theta\text{ here}}
\end{aligned}
$$

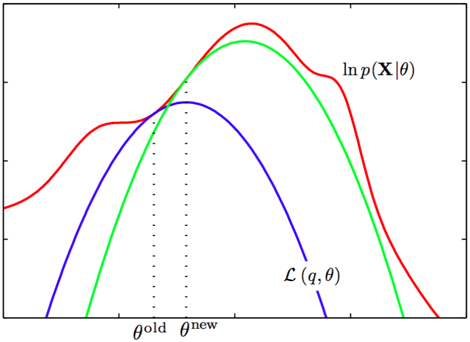
- Maximizing $\mathcal{L}(q,\theta)$ equivalent to maximizing $\mathbb{E}$ [complete data log-likelihood] (for fixed q).

General EM Algorithm

1. Choose initial θ^{old}.
2. **Expectation Step**
   - Let $q^*(z) = p(z \mid x, \theta^{\text{old}})$. [$q^*$ gives best lower bound at θ^{old}]
   - Let
   $$J(\theta) := \mathcal{L}(q^*, \theta) = \underbrace{\sum_z q^*(z) \log\left(\frac{p(x, z \mid \theta)}{q^*(z)}\right)}_{\textbf{expectation} \text{ w.r.t. } z \sim q^*(z)}$$
3. **Maximization Step**
   $$\theta^{\text{new}} = \underset{\theta}{\arg\max}\, J(\theta).$$
   [Equivalent to maximizing expected complete log-likelihood.]
4. Go to step 2, until converged.

Does EM Work?

EM Gives Monotonically Increasing Likelihood: By Picture



From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

EM Gives Monotonically Increasing Likelihood: By Math

1. Start at θ^{old}.
2. Choose $q^*(z) = \arg\max_q \mathcal{L}(q, \theta^{\text{old}})$. We've shown

$$\log p(x \mid \theta^{\text{old}}) = \mathcal{L}(q^*, \theta^{\text{old}})$$

3. Choose $\theta^{\text{new}} = \arg\max_\theta \mathcal{L}(q^*, \theta^{\text{old}})$. So

$$\mathcal{L}(q^*, \theta^{\text{new}}) \geqslant \mathcal{L}(q^*, \theta^{\text{old}}).$$

Putting it together, we get

$$\begin{aligned} \log p(x \mid \theta^{\text{new}}) &\geqslant \mathcal{L}(q^*, \theta^{\text{new}}) && \mathcal{L} \text{ is a lower bound} \\ &\geqslant \mathcal{L}(q^*, \theta^{\text{old}}) && \text{By definition of } \theta^{\text{new}} \\ &= \log p(x \mid \theta^{\text{old}}) && \text{Bound is tight at } \theta^{\text{old}}. \end{aligned}$$

Suppose We Maximize the ELBO...

- Suppose we have found a **global maximum** of $\mathcal{L}(q,\theta)$:

$$L(q^*,\theta^*) \geqslant L(q,\theta)\ \forall q,\theta,$$

where of course

$$q^*(z) = p(z \mid x,\theta^*).$$

- Claim: θ^* is a global maximum of $\log p(x \mid \theta^*)$.
- Proof: For any θ', we showed that for $q'(z) = p(z \mid x,\theta')$ we have

$$\begin{aligned}
\log p(x \mid \theta') &= \mathcal{L}(q',\theta') + \mathrm{KL}[q', p(z \mid x,\theta')] \\
&= \mathcal{L}(q',\theta') \\
&\leqslant \mathcal{L}(q^*,\theta^*) \\
&= \log p(x \mid \theta^*)
\end{aligned}$$

Convergence of EM

- Let θ_n be value of EM algorithm after n steps.
- Define "transition function" $M(\cdot)$ such that $\theta_{n+1} = M(\theta_n)$.
- Suppose log-likelihood function $\ell(\theta) = \log p(x \mid \theta)$ is differentiable.
- Let S be the set of stationary points of $\ell(\theta)$. (i.e. $\nabla_\theta \ell(\theta) = 0$)

Theorem

Under mild regularity conditions[a]*, for any starting point* θ_0,

- $\lim_{n\to\infty} \theta_n = \theta^*$ *for some stationary point* $\theta^* \in S$ *and*
- θ^* *is a fixed point of the EM algorithm, i.e.* $M(\theta^*) = \theta^*$. *Moreover,*
- $\ell(\theta_n)$ *strictly increases to* $\ell(\theta^*)$ *as* $n \to \infty$*, unless* $\theta_n \equiv \theta^*$.

[a] For details, see "Parameter Convergence for EM and MM Algorithms" by Florin Vaida in *Statistica Sinica* (2005). `http://www3.stat.sinica.edu.tw/statistica/oldpdf/a15n316.pdf`

Variations on EM

EM Gives Us Two New Problems

- The "E" Step: Computing

$$J(\theta) := \mathcal{L}(q^*, \theta) = \sum_z q^*(z) \log\left(\frac{p(x, z \mid \theta)}{q^*(z)}\right)$$

- The "M" Step: Computing

$$\theta^{\text{new}} = \underset{\theta}{\arg\max}\, J(\theta).$$

- Either of these can be too hard to do in practice.

Generalized EM (GEM)

- Addresses the problem of a difficult "M" step.
- Rather than finding
$$\theta^{\text{new}} = \operatorname*{arg\,max}_{\theta} J(\theta),$$
find **any** θ^{new} for which
$$J(\theta^{\text{new}}) > J(\theta^{\text{old}}).$$
- Can use a standard nonlinear optimization strategy
  - e.g. take a gradient step on J.
- We still get monotonically increasing likelihood.

EM and More General Variational Methods

- Suppose "E" step is difficult:
  - Hard to take expectation w.r.t. $q^*(z) = p(z \mid x, \theta^{\text{old}})$.
- Solution: Restrict to distributions $\mathcal{Q}$ that are easy to work with.
- Lower bound now looser:

$$q^* = \underset{q \in \mathcal{Q}}{\arg\min}\, \mathrm{KL}[q(z), p(z \mid x, \theta^{\text{old}})]$$

EM in Bayesian Setting

- Suppose we have a prior $p(\theta)$.
- Want to find MAP estimate: $\hat{\theta}_{\text{MAP}} = \arg\max_\theta p(\theta \mid x)$:

$$\begin{aligned} p(\theta \mid x) &= p(x \mid \theta)p(\theta)/p(x) \\ \log p(\theta \mid x) &= \log p(x \mid \theta) + \log p(\theta) - \log p(x) \end{aligned}$$

.

- Still can use our lower bound on $\log p(x, \theta)$.

$$J(\theta) := \mathcal{L}(q^*, \theta) = \sum_z q^*(z) \log\left(\frac{p(x, z \mid \theta)}{q^*(z)}\right)$$

- Maximization step becomes

$$\theta^{\text{new}} = \arg\max_\theta \left[J(\theta) + \log p(\theta)\right]$$

- Homework: Convince yourself our lower bound is still tight at θ.

Summer Homework: Gaussian Mixture Model (Hints)

Homework: Derive EM for GMM from General EM Algorithm

- Subsequent slides may help set things up.
- Key skills:
  - MLE for multivariate Gaussian distributions.
  - Lagrange multipliers

Gaussian Mixture Model (k Components)

- GMM Parameters

$$
\begin{aligned}
\text{Cluster probabilities}: \quad & \pi = (\pi_1, \ldots, \pi_k) \\
\text{Cluster means}: \quad & \mu = (\mu_1, \ldots, \mu_k) \\
\text{Cluster covariance matrices:} \quad & \Sigma = (\Sigma_1, \ldots \Sigma_k)
\end{aligned}
$$

- Let $\theta = (\pi, \mu, \Sigma)$.
- Marginal log-likelihood

$$
\log p(x \mid \theta) = \log \left\{ \sum_{z=1}^{k} \pi_z \mathcal{N}(x \mid \mu_z, \Sigma_z) \right\}
$$

$q^*(z)$ are "Soft Assignments"

- Suppose we observe n points: $X = (x_1, \ldots, x_n) \in \mathbf{R}^{n \times d}$.
- Let $z_1, \ldots, z_n \in \{1, \ldots, k\}$ be corresponding hidden variables.
- Optimal distribution q^* is:

$$q^*(z) \quad = \quad p(z \mid x, \theta).$$

- Convenient to define the conditional distribution for z_i given x_i as

$$\begin{aligned} \gamma_i^j &:= p(z = j \mid x_i) \\ &= \frac{\pi_j \mathcal{N}(x_i \mid \mu_j, \Sigma_j)}{\sum_{c=1}^k \pi_c \mathcal{N}(x_i \mid \mu_c, \Sigma_c)} \end{aligned}$$

Expectation Step

- The complete log-likelihood is

$$\begin{aligned}\log p(x, z \mid \theta) &= \sum_{i=1}^{n} \log \left[\pi_z \mathcal{N}(x_i \mid \mu_z, \Sigma_z)\right] \\ &= \sum_{i=1}^{n} \left(\log \pi_z + \underbrace{\log \mathcal{N}(x_i \mid \mu_z, \Sigma_z)}_{\text{simplifies nicely}} \right)\end{aligned}$$

- Take the expected complete log-likelihood w.r.t. q^*:

$$\begin{aligned}J(\theta) &= \sum_{z} q^*(z) \log p(x, z \mid \theta) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{k} \gamma_i^j \left[\log \pi_j + \log \mathcal{N}(x_i \mid \mu_j, \Sigma_j)\right]\end{aligned}$$

Maximization Step

- Find θ^* maximizing $J(\theta)$:

$$
\begin{aligned}
\mu_c^{\text{new}} &= \frac{1}{n_c}\sum_{i=1}^{n}\gamma_i^c x_i \\
\Sigma_c^{\text{new}} &= \frac{1}{n_c}\sum_{i=1}^{n}\gamma_i^c \left(x_i-\mu_{\text{MLE}}\right)\left(x_i-\mu_{\text{MLE}}\right)^T \\
\pi_c^{\text{new}} &= \frac{n_c}{n},
\end{aligned}
$$

for each $c = 1,\ldots,k$.