EM algorithm for GMMs: `gmm_estimate.m`

Synopsis of the algorithm :

- Start from M initial Gaussian models $\mathcal{N}(\mu_k, \Sigma_k),\ k = 1 \cdots M$, with equal priors set to $P(q_k|\Theta) = 1/M$.
- **Do** :
  1. **Estimation step** : compute the probability $P(q_k|x_n, \Theta)$ for each data point x_n to belong to the mixture q_k :

$$\begin{aligned} P(q_k|x_n,\Theta) &= \frac{P(q_k|\Theta) \cdot p(x_n|q_k,\Theta)}{p(x_n|\Theta)} \\ &= \frac{P(q_k|\Theta) \cdot p(x_n|\mu_k,\Sigma_k)}{\sum_j P(q_j|\Theta) \cdot p(x_n|\mu_j,\Sigma_j)} \end{aligned} \tag{1}$$

     In the algorithm:
     `c(k)`$= P(q_k|\Theta)$,
     `lBM(n,k)`$= \log p(x_n|q_k,\Theta)$,
     `lB(k)`$= \log p(x_n|\Theta)$,
     `gam_m(n,k)`$= P(q_k|x_n,\Theta)$.

  2. **Maximization step** :
     - update the means :

$$\mu_k^{(new)} = \frac{\sum_{n=1}^{T} x_n P(q_k|x_n,\Theta)}{\sum_{n=1}^{T} P(q_k|x_n,\Theta)} \tag{2}$$

     - update the variances :

$$\Sigma_k^{(new)} = \frac{\sum_{n=1}^{T} P(q_k|x_n,\Theta)(x_n - \mu_k^{(new)})(x_n - \mu_k^{(new)})^{\mathtt{T}}}{\sum_{n=1}^{T} P(q_k|x_n,\Theta)} \tag{3}$$

     - update the weigths :

$$P(q_k^{(new)}|\Theta^{(new)}) = \frac{1}{T}\sum_{n=1}^{T} P(q_k|x_n,\Theta) \tag{4}$$

     In the algorithm:
     `new_mu(:,k)`$= \mu_k^{(new)}$,
     `new_sigm(:,k)`$= \Sigma_k^{(new)}$,
     `new_c(k)`$= P(q_k^{(new)}|\Theta^{(new)})$.

  3. **Go to 1.(*)**

* **Until** : the total likelihood increase for the training data falls under some desired threshold.