EM-algorithm for GMMs

This is the EM-algorithm as given by Murphy, "Machine Learning - A Probabilistic Approach", p 353.
assume data set $\mathbf{X}$ with examples $\overrightarrow{x_i}, i = 1,\ldots,N$ and K classes you want to cluster $\mathbf{X}$ into.

EM-for-GMM($\mathbf{X}, \mathbf{K}$)

1. Initialize $\theta_k^0 = (\pi_k^0, \overrightarrow{\mu_k}^0, \Sigma_k^0)$, where
   π_k is the class prior for class k (e.g., assume uniform distribution here initially)
   $\overrightarrow{\mu_k}$ are the means for the attribute values j in class k (e.g., the means over a random subset of the data)
   Σ_k is the covariance for the attribute values in class k (can be simplified to variance σ_{jk}^2 for each attribute j if a G-NBC is assumed as the model)
2. Iterate over E and M steps as follows:

   E-step:

   compute $r_{ik}^t = \dfrac{\pi_k^{t-1} P(\overrightarrow{x}_i \mid \theta_k^{t-1})}{\sum_{k'} \pi_{k'}^{t-1} P(\overrightarrow{x}_i \mid \theta_{k'}^{t-1})}$ where $P(\overrightarrow{x}_i \mid \theta_k^{t-1}) = \prod_j \dfrac{1}{\sqrt{2\pi(\sigma_{kj}^{t-1})^2}} e^{-\frac{1}{2(\sigma_{kj}^{t-1})^2}(x-\mu_{kj}^{t-1})^2}$, assuming

that

   the covariance can be substituted with σ_{jk}^2 for attribute j and class k.

   M-step:

   compute $r_k^t = \sum_i r_{ik}^t \quad and \quad \pi_k^t = \dfrac{r_k^t}{N}$, then update the means and variances:

   $$\overrightarrow{\mu_k}^t = \frac{\sum_i r_{ik}^t \overrightarrow{x_i}}{r_k^t} \quad and \quad \Sigma_k^t = \frac{\sum_i r_{ik}^t \overrightarrow{x}_i \overrightarrow{x}_i^T}{r_k^t} - \overrightarrow{\mu_k^t}\overrightarrow{\mu_k^t}^T$$ (from which the new σ_{jk}^2 can be extracted)

3. Stop, when the $\overrightarrow{\mu_k}$ and Σ_k are not changing significantly anymore.

EM-algorithm for k-Means

This is the EM-algorithm as given by Murphy, "Machine Learning - A Probabilistic Approach", p 356.
assume data set $\mathbf{X}$ with examples $\overrightarrow{x_i}, i = 1,\ldots,N$ and K classes you want to cluster $\mathbf{X}$ into.

k-Means($\mathbf{X}, \mathbf{K}$)
1. Initialize $\overrightarrow{\mu_k}^0$, assume fixed class priors π_k
2. *Iterate* over E and M steps as follows:

E-step:

Assign each data point to its closest cluster centre: $z_i = \arg\min_k || \overrightarrow{x}_i - \overrightarrow{\mu}_k ||_2^2 = L_2(\overrightarrow{x}_i - \overrightarrow{\mu}_k)^2$

M-step:

Update each cluster centre by computing the means of all points assigned to it:

$$\overrightarrow{\mu}_k = \frac{1}{N_k} \sum_{i:z_i=k} \overrightarrow{x}_i$$

Until converged