Discussion 8 3/29/19

E-M with Gaussian mixture models

Expectation Maximization Algorithm for GMM

1. Initialize π = mixing weights (n_c/total # of points) for each Gaussian

   μ = mean for each Gaussian (random)

   Σ = covariance matrix for each cluster/Gaussian (Identity)

   n_c = number of points per cluster. This is usually set to be total # of points/k (# of clusters) initially

   k = # of clusters (number of Gaussians to estimate)

2. E-Step.

   For all $i = 1$ to n, $c = 1$ to k, compute:

$$\gamma(z_{ic}) = \frac{\pi_c \left[\frac{1}{2\pi^{d/2}|\Sigma_c|^{1/2}} \exp\left(-\frac{1}{2}(x_i-\mu_c)^T \Sigma_c^{-1}(x_i-\mu_c)\right)\right]}{\sum_{j=1}^{k} \pi_j \left[\frac{1}{2\pi^{d/2}|\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x_i-\mu_j)^T \Sigma_j^{-1}(x_i-\mu_j)\right)\right]}$$

Probability that the i^{th} point belongs to the c^{th} cluster.

End result: an $n \times k$ matrix

	c=1	2	→ k
x_1	0.1	0.4	....
x_2	0.2	0.5	....
x_3			
↓			
x_n			

Sum of each row = 1, since probabilities that each point lies in each cluster will add to 1 (100%)

In the above equation for $\gamma(z_{ic})$, d represents the dimensionality of the multivariate Gaussian (where $d \neq k$, the number of clusters, necessarily). If we have a GMM defined by 3 Gaussians, i.e.

$$X = \{x_1, x_2, x_3\}, \text{ where } x_i \sim N(\mu, \Sigma)$$

k = 3 = number of clusters/Gaussians, and

d = number of dimensions in the data (how many features it has.)

3. M-Step. (Update parameters)

$n_c = \sum_{i=1}^{n} \gamma(z_{ic})$ Re-estimate the number of points per cluster by summing up $\gamma(z_{ic})$ for each cluster (sum columns of E-Step matrix)

$\pi_c = n_c/n$

$\mu_c = \frac{1}{n_c} \sum_{i=1}^{n} \gamma(z_{ic}) x_i$

$\Sigma_c = \frac{1}{n_c} \sum_{i=1}^{n} \gamma(z_{ic}) (x_i - \mu_c)(x_i - \mu_c)^T$

4. Repeat until convergence.

We are trying to maximize the log-likelihood estimate of our parameters. To check for convergence, we compute the log-likelihood and see if it is close enough to the previous estimate (less than some convergence threshold).

$$MLE = \sum_{i=1}^{n} \log \sum_{j=1}^{k} \pi_j \left[\frac{1}{2\pi^{d/2} |\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2} \underbrace{(x_i-\mu_j)^T \Sigma_j^{-1} (x_i-\mu_j)}\right) \right]$$

This expression is the Mahalanobis distance. In short, it measures the distance of a point to a distribution (in this case, the cluster/Gaussian). Minimizing this distance means the points are closer to the distribution (this is ideal, since we want our μ and Σ to be correct).

Due to the negative exponent, minimizing the Mahalanobis distance will maximize the MLE, which is what we want!