2022

Lecture 8: Clustering

Week 8

Lecturer: Tianyu Wang

1 Clustering

In a clustering task, one needs to partition the input space (or the data points) so that each region in the partition corresponds to one cluster.

1.1 K-Means

Given an initial set of k means (centroids) $m_1^{(1)}, \cdots, m_k^{(1)}$, the algorithm proceeds by alternating between two steps for $t = 1, 2, \cdots$

- **Assignment step**: Assign each observation to the cluster with the nearest mean: that with the least squared Euclidean distance. The set of point for cluster i is

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - m_i^{(t)} \right\|^2 \leq \left\| x_p - m_j^{(t)} \right\|^2 \ \forall j, 1 \leq j \leq k \right\},$$

  where each x_p is assigned to exactly one $S^{(t)}$, with ties broken randomly.

- **Update step**: Recalculate means (centroids) for observations assigned to each cluster.

$$m_i^{(t+1)} = \frac{1}{\left| S_i^{(t)} \right|} \sum_{x_j \in S_i^{(t)}} x_j.$$

2 Gaussian Mixture Model

In this model, we assume the data is generated from a density that is a mixture of K Gaussian densities. Consider the d-dimensional Euclidean space. Let's say the K Gaussian densities are parametrized by $(\mu_1, \Sigma_1), (\mu_2, \Sigma_2), \cdots, (\mu_K, \Sigma_K)$, where the pair (μ_i, Σ_i) parametrizes the mean and covariance of the i-th Gaussian density. Let $w_1, w_2, \cdots, w_K$ ($\sum_{i=1}^K w_i = 1$) be the weight of each Gaussian, then the joint density of the mixture of Gaussian is

$$p(x|w, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}}} \sum_{i=1}^{K} w_i \exp\left(-\frac{1}{2} (x - \mu_i)^\top \Sigma_i^{-1} (x - \mu_i) \right).$$

Suppose our data $\{x_i\}_{i=1}^n$ is generated from the above model. One can generate this dataset by first choosing a Gaussian (from the K Gaussians) and sample a point from the chosen Gaussian. We can describe this two-step procedure as: For each $i = 1, 2, \cdots, n$,

$$\begin{aligned} z_i &\sim Category(w) \\ x_i | z_i &\sim \mathcal{N}\left(\mu_{z_i}, \Sigma_{z_i}\right), \end{aligned}$$

where $Category(w)$ is the categorical distribution with parameter $w_1, w_2, \cdots, w_K$. The $Category(w)$ distribution is supported on $\{1, 2, \cdots, K\}$. The density function for $z_i \sim Category(w)$ is

$$f(z_i = k) = w_k,$$

or equivalently

$$f(z_i) = \prod_{i=1}^{K} w_i^{\mathbb{I}_{[z_i=k]}}.$$

As an analog of the K-mean algorithm, we can perform a "soft" version of **assignment step** and the **update step**. Given $\{x_i\}_{i=1}^n$ and w, μ, Σ randomly initialized, we repeat the following two steps until convergence.

- **(Assignment Step, Expectation Step, E-step)** Compute

$$\begin{aligned} \tau_{ij} &= \mathbb{E}\left[\mathbb{I}_{[z_i=j]} | x_i, w, \mu, \Sigma\right] \\ &= \mathbb{P}\left(z_i = j | x_i, w, \mu, \Sigma\right) \\ &= \frac{w_j \exp\left(-\frac{1}{2}\left(x_i - \mu_j\right)^\top \Sigma_j^{-1}\left(x_i - \mu_j\right)\right)}{\sum_{k=1}^K w_k \exp\left(-\frac{1}{2}\left(x_i - \mu_k\right)^\top \Sigma_k^{-1}\left(x_i - \mu_k\right)\right)}, \end{aligned}$$

  which is the probability of point i assigned to cluster j.

- **(Update Step, Maximization Step, M-step)** Update parameters for the mixture model:

$$\begin{aligned} w_j &= \frac{1}{n}\sum_{i=1}^n \tau_{ij} \\ \mu_j &= \frac{\sum_{i=1}^n \tau_{ij} x_i}{\sum_{i=1}^n \tau_{ij}} \\ \Sigma_j &= \frac{\sum_{i=1}^n \tau_{ij}(x_i - \mu_j)(x_i - \mu_j)^\top}{\sum_{i=1}^n \tau_{ij}}. \end{aligned}$$

The expectation step takes an expectation, and the maximization step takes a maximum. We will see an example below. We assume that the covariance matrix always stays positive definite.

Note that

$$\mathbb{P}\left(x_i, z_i | w, \mu, \Sigma)\right) = \mathbb{P}\left(x_i | z_i, w, \mu, \Sigma)\right) \mathbb{P}(z_i | w, \mu, \Sigma). \tag{1}$$

The likelihood is

$$\mathbb{P}(\{x_i\}_{i=1}^n, \{z_i\}_{i=1}^n | w, \mu, \Sigma) \propto \prod_{i=1}^n \left(\sum_{j=1}^K w_j \exp\left(-\frac{1}{2} (x_i - \mu_j)^\top \Sigma_j^{-1} (x_i - \mu_j) \right) \right).$$

And the log-likelihood is (up to constant)

$$\begin{aligned} & \log \mathbb{P}(\{x_i\}_{i=1}^n, \{z_i\}_{i=1}^n | w, \mu, \Sigma) \\ = & \sum_{i=1}^n \log \left(\sum_{j=1}^K w_j \exp\left(-\frac{1}{2} (x_i - \mu_j)^\top \Sigma_j^{-1} (x_i - \mu_j) \right) \right). \end{aligned}$$

Taking derivative of the log-likelihood gives

$$\begin{aligned} & \frac{\partial}{\partial \mu_j} \sum_{i=1}^n \log \left(\sum_{k=1}^K w_k \exp\left(-\frac{1}{2} (x_i - \mu_k)^\top \Sigma_k^{-1} (x_i - \mu_k) \right) \right) \\ = & \sum_{i=1}^n \frac{w_j \exp\left(-\frac{1}{2} (x_i - \mu_j)^\top \Sigma_j^{-1} (x_i - \mu_j) \right)}{\sum_{k=1}^K w_k \exp\left(-\frac{1}{2} (x_i - \mu_k)^\top \Sigma_k^{-1} (x_i - \mu_k) \right)} \left(-\Sigma_j^{-1} (x_i - \mu_j) \right) \\ = & -\sum_{i=1}^n \tau_{ij} \Sigma_j^{-1} (x_i - \mu_j). \end{aligned}$$

Since we assume that the covariance matrices Σ_j are positive definite, setting the above derivative to zero gives

$$\mu_j = \frac{\sum_{i=1}^n \tau_{ij} x_i}{\sum_{i=1}^n \tau_{ij}}.$$

The Expectation-Maximization algorithm is a general algorithm and applies to more problems other than GMM inference.

Acknowledgement

Reference: Machine Learning: A Probabilistic Perspective by Kevin Murphy. A thank you to wikipedia contributors.