Clustering and EM Algorithm

Yuh-Jye Lee

Dept. Applied Mathematics at NCTU

May 23, 2017

Unsupervised Learning: Clustering

- Given a dataset $S = \{\mathbf{x}^i | \mathbf{x}^i \in \mathcal{R}^n, i = 1, 2, \ldots, \ell\}$
- Note that: we don't have the *label*, y_i now.
- It is considered as a *unsupervised learning* problem
- We would like to find the *structure* within the dataset S.
  - *Similar* to one another within the *same* cluster
  - *Dissimilar* to the objects in other clusters
- There are many different type of clustering algorithms such as:
  - Bottom-up: Hierarchical Agglomerative Clustering
  - Top-Down: k-means, soft k-means, SOM and MDS

k-means Algorithm

- Try to group data into *k* *clusters* and attempt to group data points to *minimize* the sum of *squares distance* to their *central mean*.
- Here *smaller distance* implies *larger similarity*
- *Similar* to one another within the *same* cluster
- Algorithm works by iterating between two stages until the data points converge.

k-means Clustering Problem Formulation

- Given a dataset $S = \{\mathbf{x}^i | \mathbf{x}^i \in \mathcal{R}^n, i = 1, 2, \ldots, \ell\}$ and a positive integer k.
- Introduce a set of k *prototype vectors*, $\mu_j, j = 1, 2, \ldots, k$ and μ_j corresponds to the *centroid* of the j^{th} cluster.
- Goal is to find a grouping of data points and prototype vectors that minimizes the sum of squares distance of each data point.
- You have to find k *prototype vectors*, $\mu_j, j = 1, 2, \ldots, k$ and μ_j and the *membership* for *each* data point

k-means Clustering Problem Formulation

- Let r_{ij} be a *binary variable* that indicates the membership of data point $\mathbf{x^i}$ is in the cluster j or not.
- We would to find k *prototype vectors*, $\mu_j, j = 1, 2, \ldots, k$ and μ_j and the *membership* for *each* data point
- Our objective function becomes:

$$\min_{r_{ij},\mu_j} \sum_{i=1}^{\ell} \sum_{j=1}^{k} r_{ij} \|\mathbf{x^i} - \mu_j\|_2^2$$

How to solve it?

- Algorithm initializes the k *centroids* to k distinct *random data points*.
- Cycles between two stages until convergence is reached.
- Convergence: since there are only a finite set of possible assignments.

Given a Set of *Centroids*, How to Update the Membership?

Update Rule for Membership

- For each data point, determine r_{ij} where:

$$r_{ij} = \begin{cases} 1 & : \quad \textit{if} \quad j \in \arg\min \|\mathbf{x}^i - \mu_j\|_2^2 \\ 0 & : \quad \textit{otherwise} \end{cases}$$

How to Update the *Centroids* According to New Membership?

Update Rule for Centroids

- $$\mu_j = \frac{\sum_{i=1}^{\ell} r_{ij}\mathbf{x^i}}{\sum_{i=1}^{\ell} r_{ij}}, \quad j = 1, 2, \ldots, k$$

How to Select Initial Seeds? Can We Do Better than Random?

k-means++

1. Choose one center *uniformly at random* from among the data points.
2. For each data point $\mathbf{x^i}$, compute $D(x)$, the distance between $\mathbf{x^i}$ and the nearest center that has already been chosen.
3. Choose one new data point at random as a new center, using a weighted probability distribution where a point x is chosen with probability proportional to $D(x)^2$.
4. Repeat Steps 2 and 3 until k centers have been chosen.
5. Now that the initial centers have been chosen, proceed using standard *k*-means.

Examples of k-means

- Cluster black and white intensities: Intensities: 1, 3, 8, 11
  Centers $c1 = 7, \ c2 = 10$
- Consider points 0, 20, 32.

Soft k-means

Partial Membership

- Clustering typically assumes that each instance is given a "*hard*" assignment to exactly one cluster.
- Does not allow uncertainty in class membership or for an instance to belong to more than one cluster.
- *Soft clustering* gives probabilities that an instance belongs to each of a set of clusters.
- Each instance is assigned a *probability distribution* across a set of discovered categories (probabilities of all categories must sum to 1).

The Expectation Maximization Algorithm

EM-Algorithm

- The EM algorithm is an efficient iterative procedure to compute the Maximum Likelihood (ML) estimate in the presence of *missing or hidden* data.
  - In the soft k-means, we *DON'T* know the *proportion* of each instance belong to each cluster.
- In Maximum Likelihood estimation, we wish to estimate the model parameter(s) for which the observed data are the *most likely*.
- Each iteration of the EM algorithm consists of two processes:
  - E-step: the missing data are estimated given the observed data and current estimate of the model parameters.
  - M-step: the likelihood function is maximized under the assumption that the missing data are known.