Clustering

S. R. M. Prasanna

Dept of Electrical Engineering
Indian Institute of Technology Dharwad

prasanna@iitdh.ac.in

November 26, 2020

Acknowledgements

- Artificial Neural Networks by Prof. B. Yegnanarayana, PHI, 1999
- Machine learning video lectures by Prof. Andrew Ng, Stanford Uty
- Jason Brownlee "Basics of Linear Algebra for Machine Learning", Online book, 2018.
- Pattern Recognition NPTEL video lectures by S. Das and Murthy, IIT Madras.
- Machine learning NPTEL video lectures by Ravindran, IIT Madras.
- My TAs Jagabandhu Mishra and Seema K.

About Clustering

- A collection of objects on the basis of similarity and dissimilarity between them.
- Task of dividing the features into a number of groups such that features in the same group are more similar or closer.
- Process to find meaningful structure, explanatory of underlying processes.
- Type of unsupervised learning.

Some Important Clustering Methods

- Partition based clustering: k-means and fuzzy c means clustering
- Model based clustering: Gaussian mixture model based clustering
- Hierarchy based clustering: Agglomerative or bottom up, Divisive or top down

Partition based: K-means clustering

- No. of means (K) for clustering is a parameter along with M feature vectors $\mathrm{X} = \{\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_\mathrm{M}\}$.
- **Initialization:** From given X feature vectors, randomly choose K feature vectors and declare them cluster centroids.
- **Cluster Assignment:** Each feature vector is assigned to one of the centroids based on its closeness.
- **Computing New Centroids:** Compute new mean vectors or centroids using the assigned feature vectors for each cluster.
- **Cost Function:** Find out the cost involved for measuring total distance wrt to old means and new means.
- Repeat this process until no change in centroid values or difference in cost function values is ($\leq$ threshold).

Partition based: K-means clustering

- Randomly initialize K centroids, $\mu_1, \mu_2, ..., \mu_K$.

- Cluster Assignment: For $i = 1, 2, \ldots, M$, assign $c^{(i)} = k$, where $k = 1, 2, \ldots, K$ using $\arg\min_k ||x^{(i)} - \mu_k||$

- Cost Function: J(C, μ) $= \dfrac{1}{M} \sum_{i=1}^{M} ||x^{(i)} - \mu_{c^{(i)}}||$), $\min J(C, \mu)$

- Compute new means:
  - Let $\delta(i, k) = 1$ if $c^{(i)} = k$, 0 otherwise and $N_k = 0, 1 \leq k \leq K$
  - Find out $N_k = N_k + \delta(i, k)$ for $1 \leq k \leq K$ and $1 \leq i \leq M$
  - $\hat{\mu_k} = \frac{1}{N_k} \sum_{i=1}^{M} x^{(i)} \delta(i, k)$, for $1 \leq k \leq K$

Partition based: K-means clustering

- Cluster assignment for new means: For $i = 1, 2, \ldots, M$, assign $c^{(i)} = k$, where, $k = 1, 2, \ldots, K$ using $\arg\min_k ||x^{(i)} - \hat{\mu_k}||$

- Cost Function: $J(\hat{C}, \hat{\mu}) = \frac{1}{M} \sum_{i=1}^{M} ||x^{(i)} - \hat{\mu}_{c^{(i)}}||)$, $\min J(C, \mu)$

- If $|J(C, \mu) - J(\hat{C}, \hat{\mu})| > \delta$, then $J(C, \mu) = J(\hat{C}, \hat{\mu})$, and $\mu_k = \hat{\mu_k}$.

- Repeat steps from compute new means till above until $|J(C, \mu) - J(\hat{C}, \hat{\mu})| > \delta$.

Partition based: Fuzzy C-means clustering

- No. of means (K) for clustering is a parameter along with M feature vectors $\mathrm{X} = \{\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_\mathrm{M}\}$.
- **Initialization:** From given X feature vectors, randomly choose K feature vectors and declare them cluster centroids.
- **Cluster Assignment:** Each feature vector is assigned to all the centroids based on its closeness measured as degree of association.
- **Computing New Centroids:** Compute new mean vectors or centroids using all the feature vectors for each cluster weighted by their association.
- **Cost Function:** Find out the cost involved for measuring total distance wrt to old means and new means.
- Repeat this process until no change in centroid values or difference in cost function values is ($\leq$ threshold).

Partition based: Fuzzy C-means clustering

- Randomly initialize K centroids, $\mu_1, \mu_2, ..., \mu_K$.
- Cluster Assignment: For $i = 1, 2, \ldots, M$, assign $c_k^{(i)} = \frac{||x^{(i)} - \mu_k||}{\sum_{k=1}^{K} ||x^{(i)} - \mu_k||}$, where $k = 1, 2, \ldots, K$
- Cost Function: $\mathrm{J}(\mathrm{C}, \mu) = \dfrac{1}{M \times K} \sum_{i=1}^{M} \sum_{k=1}^{K} c_k^{(i)} ||x^{(i)} - \mu_{c^{(i)}k}||),$
- Compute new means: $\hat{\mu_k} = \frac{1}{M} \sum_{i=1}^{M} x^{(i)} c_k^{(i)}$ for $1 \leq k \leq K$

Partition based: Fuzzy C-means clustering

- Cluster assignment for new means: For $i = 1, 2, \ldots, M$, assign $\hat{c}_k^{(i)} = \frac{||x^{(i)} - \hat{\mu}_k||}{\sum_{k=1}^{K} ||x^{(i)} - \hat{\mu}_k||}$, where $k = 1, 2, \ldots, K$

- Cost Function: $\mathrm{J}(, \hat{\mu}) = \frac{1}{M \times K} \sum_{i=1}^{M} \sum_{k=1}^{K} \hat{c}_k^{(i)} ||x^{(i)} - \hat{\mu}_{c^{(i)}{}_k}||)$,

- If $|J(C, \mu) - J(\hat{C}, \hat{\mu})| > \delta$, then $J(C, \mu) = J(\hat{C}, \hat{\mu})$, and $\mu_k = \hat{\mu_k}$.

- Repeat steps from compute new means till above until $|J(C, \mu) - J(\hat{C}, \hat{\mu})| > \delta$.

Gaussian Mixture Model (GMM)

- k-means exploits only mean of the cluster or distribution as representation for class-specific information.
- Second order moment like variance also contains class-specific information.
- Gaussian distribution can exploit both mean and variance.
- In case of scalar it is univariate and in case of vector it is multivariate Gaussian distribution.
- The distribution of feature vectors in case of speech for each class is non-Gaussian in nature.
- The same can be modeled using a Gaussian mixture model (GMM).

Univariate vs Multivariate Gaussian Distribution

- $g(x) = \frac{1}{\sqrt{2\pi}\sigma} exp^{\frac{-(x-\mu)^2}{2\sigma^2}}$
- Let $\mathbf{x} = [x_1, x_2]^T$ be a vector with mean vector $\mu = [\mu_1, \mu_2]^T$ and variance vector $\sigma^2 = [\sigma_1^2, \sigma_2^2]^T$
- $g(x_1) = \frac{1}{\sqrt{2\pi}\sigma_1} exp^{-\frac{(x_1-\mu_1)^2}{2\sigma_1^2}}$ and $g(x_2) = \frac{1}{\sqrt{2\pi}\sigma_2} exp^{-\frac{(x_2-\mu_2)^2}{2\sigma_2^2}}$
- Let $g(x_1, x_2) = g(x_1)g(x_2) = \frac{1}{\sqrt{2\pi}\sigma_1} exp^{-\frac{(x_1-\mu_1)^2}{2\sigma_1^2}} \frac{1}{\sqrt{2\pi}\sigma_2} exp^{-\frac{(x_2-\mu_2)^2}{2\sigma_2^2}}$
- $g(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2} exp^{-\frac{1}{2}(\frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2})}$

Univariate vs Multivariate Gaussian Distribution

- $g(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2} exp^{(-\frac{1}{2}[(x_1-\mu_1);(x_2-\mu_2)]^T[\frac{(x_1-\mu_1)}{\sigma_1^2};\frac{(x_2-\mu_2)}{\sigma_2^2}])}$
- $g(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2} exp^{(-\frac{1}{2}[(x_1-\mu_1);(x_2-\mu_2)]^T[\frac{1}{\sigma_1^2},0;0,\frac{1}{\sigma_2^2}][(x_1-\mu_1);(x_2-\mu_2)])}$
- Let $\Sigma = [\sigma_1^2, 0; 0, \sigma_2^2]$
- $g(\mathbf{x}) = \frac{1}{2\pi\sigma_1\sigma_2} exp^{(-\frac{1}{2}[(x-\mu)^T\Sigma^{-1}(x-\mu)])}$
- $(x - \mu) = [(x_1 - \mu_1); (x_2 - \mu_2)]$
- $|\Sigma| = |[\sigma_1^2, 0; 0, \sigma_2^2]| = |\sigma_1^2, 0; 0, \sigma_2^2| = \sigma_1^2\sigma_2^2$
- $g(\mathbf{x}) = \frac{1}{2\pi|\Sigma|^{1/2}} exp^{(-\frac{1}{2}[(x-\mu)^T\Sigma^{-1}(x-\mu)])}$
- $g(\mathbf{x}, \mu, \Sigma) = \frac{1}{2\pi|\Sigma|^{1/2}} exp^{(-\frac{1}{2}[(x-\mu)^T\Sigma^{-1}(x-\mu)])}$

Univariate vs Multivariate Gaussian Distribution

- For 2D case, $g(\mathbf{x}, \mu, \Sigma) = \frac{1}{2\pi|\Sigma|^{1/2}} exp^{(-\frac{1}{2}[(x-\mu)^T \Sigma^{-1} (x-\mu)])}$
- Generalizing to n-dimension,
  $g(\mathbf{x}, \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} exp^{(-\frac{1}{2}[(x-\mu)^T \Sigma^{-1} (x-\mu)])}$
- Above is the equation for modeling the cluster of a n-dimension feature vectors by a multivariate Gaussian distribution.
- In case of k-means, we model using only μ, where as, in Gaussian distribution, we model using both μ and Σ.

Clustering using Multivariate Gaussian Distribution

- Randomly initialize k centroids, $\mu_1, \mu_2, \ldots, \mu_K$.
- Consider with unit variance, i.e., $\sigma_1 = \sigma_2 = \ldots = \sigma_K = 1$
- For $i = 1 \ldots M$, assign $c^{(i)} = k$, where, $k = 1, 2, \ldots, K$ using $arg \min_k ||x^{(i)} - \mu_k||$
- For each assignment accumulate cost $(C = \sum_{i=1}^{M} \min_k ||x^{(i)} - \mu_k||)$.
- Compute new means, $\hat{\mu_1}, \hat{\mu_2}, \ldots, \hat{\mu_K}$
- Computer new variances, $\hat{\sigma_1}, \hat{\sigma_2}, \ldots, \hat{\sigma_K}$ using feature vectors assigned for each cluster.
- Repeat this process until means and variances do not change beyond certain value.

Gaussian Mixture Model (GMM)

- Each multivariate Gaussian distribution is modelling independently. This still has hard partitioning.
- However, features from different clusters may be related.
- Can we get to soft partitioning.
- Each feature vector is related with all clusters using a probability.
- This leads to a mixture model: $\lambda(x, w, \mu, \Sigma) = \sum_{k=1}^{K} w_k g_k(x, \mu_k, \Sigma_k)$
- $\sum_{k=1}^{K} w_k = 1$.
- How to estimate the gmm parameters, i.e., w, μ, Σ ?

Expectation-Maximization (EM) Algorithm

- Assume an initial model λ
- Using given training FVs x and model λ, re-estimate model parameters leading to improved model $\bar{\lambda}$
- Repeat above step until, $|p(x/\bar{\lambda}) - p(x/\lambda)| > \delta$, then $\lambda = \bar{\lambda}$
- At the end model λ represents gmm providing best possible representation for the distribution present in x.

Implementation of EM Algorithm

- Let $\mathrm{X} = \mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_\mathrm{M}$, M training feature vectors, each of N dimension.
- Need to develop K mixture GMM.
- Using k-means clustering, estimate μ_k. Also w_k and σ_k using the FVs assigned for each cluster.
- Construct Σ using σ_k
- Initial model: λ with pars (w_k, μ_k, Σ_k) is ready

Re-estimation in EM Algorithm

- A posteriori probability computation: $p(k/x_i, \lambda) = \frac{w_k g_k(x_i)}{\sum_{k=1}^{K} w_k g_k(x_i)}$, where $i = 1, \ldots, M$
- Re-estimated weight parameter: $\bar{w_k} = \frac{1}{M} \sum_{i=1}^{M} p(k/x_i, \lambda)$
- Re-estimated mean vector: $\bar{\mu_k} = \frac{\sum_{i=1}^{M} p(k/x_i,\lambda) x_i}{\sum_{i=1}^{M} p(k/x_i,\lambda)}$
- Re-estimated covariance matrix: $\bar{\Sigma} = \frac{\sum_{i=1}^{M} p(k/x_i,\lambda)(x_i - \mu_i)(x_i - \mu_i)^T}{\sum_{i=1}^{M} p(k/x_i,\lambda)}$
- Re-estimated model: $\bar{\lambda} = (\bar{w}, \bar{\mu}, \bar{\Sigma})$
- Repeat above re-estimation steps until, $|p(x/\bar{\lambda}) - p(x/\lambda)| > \delta$, then $\lambda = \bar{\lambda}$

Clustering using GMM

- Compute initial GMM model λ using k-means algorithm
- Re-estimate GMM model $\bar{\lambda}$ using EM algorithm.
- During testing, $p(x/\lambda)$ gives a vector of M dimension, where each value represents the prob of the test feature vector belonging to specific mixture component. Hence the soft partitioning.

About Hierarchical Clustering

- Arrange data points or feature vectors on some hierarchy.
- Possibility of combining feature vectors at same level of hierarchy.
- Same concept can be used for clustering feature vectors termed as **hierarchical clustering**.
- Two ways of hierarchical clustering, namely, **bottom-up, top-down**.
- **Bottom-up or agglomerative clustering** starts from individual feature vectors and clusters them till all feature vectors are combined into single cluster at some hierarchy.
- **Top-Down or divisive clustering** starts from single cluster containing all feature vectors and divides them into clusters of smaller sizes based on some hierarchy till we reach clusters having single feature vector.

Tree structure or Dendogram

- Hierarchical clustering leads to a tree like structure termed as **dendogram**.
- Required level of clustering can be achieved by cutting the tree at particular level.
- If you need k clusters, then cut the tree at which it splits into k branches.

Agglomerative Hierarchical Clustering

- Given M feature vectors for clustering.
- To begin with, consider each feature vector as a separate cluster. Thus we have M clusters to begin with.
- Find out the pairs of clusters which are closest at this level of hierarchy based on some cluster distance criterion.
- Merge these two clusters to form next higher level of clusters, where no of clusters are lesser than previous level of hierarchy.
- Repeat the above two steps till we finally reach single cluster.

Agglomerative Clustering Algorithm

- **Intialization:** Choose $R_0 = \{C_i = \{x_i, i = 1, \ldots, N\}\}$ as the initial clustering and $t = 0$.

- **Repeat:** $t = t + 1$, among all possible pairs of clusters (C_r, C_s) in $R_{(t-1)}$ find the one, say (C_i, C_j), such that
$g(C_i, C_j) = min_{r,s} g(C_r, C_s)$

- Define $C_q = C_i U C_j$ and produce the new clustering,
$R_t = (R_{(t-1)} - \{C_i, C_j\}) U C_q$

- Until all vectors lie in a single cluster.

Example for Agglomerative Clustering

- Let $X = \{x_i, i = 1, \dots, 5\}$, with $x_1 = [1, 1]'$, $x_2 = [2, 1]'$, $x_3 = [5, 4]'$, $x_4 = [6, 5]'$, and $x_5 = [6.5, 6]'$ and its corresponding dissimilarity matrix, when the Euclidean distance is in use, is

- $P(X) = [0, 1, 5, 6.4, 7.4;$
1, 0, 4.2, 5.7, 6.7;
5, 4.2, 0, 1.4, 2.5;
6.4, 5.7, 1.4, 0, 1.1;
7.4, 6.7, 2.5, 1.1, 0]

- (x_1, x_2) ;
(x_4, x_5);
(x_3, x_4, x_5);
$(x_1, x_2, x_3, x_4, x_5)$;

- Dendogram

Divisive Hierarchical Clustering

- Given M feature vectors for clustering.
- To begin with, construct one cluster containing all these feature vectors. Thus we have single cluster to begin with.
- Partition feature vectors in each cluster into two clusters at this level of hierarchy based on some cluster distance criterion.
- Split each cluster into two clusters to form next lower level of clusters, where no of clusters are more than previous level of hierarchy.
- Repeat the above two steps till we finally reach M clusters.

Divisive Clustering Algorithm

- **Intialization:** Choose $R_0 = \{X\}$ as the initial cluster. and $t = 0$.
- **Repeat:** $t = t + 1$,
- For $i = 1 : t$, among all possible pairs of clusters (C_r, C_s) in $R_{(t-1)}$ that form a partition of $(C_{(t-1),i})$ find the pairs, $(C^1_{(t-1),i}, C^2_{(t-1),i})$ one such that $g(C_i, C_j) = max_{r,s} g(C_r, C_s)$
- Next i
- From the t pairs defined in the previous step choose the one that maximizes g. Suppose this is $(C^1_{(t-1),j}, C^2_{(t-1),j})$.
- The new clustering is $R_t = (R_{(t-1)} - C_{(t-1),j}) U (C^1_{(t-1),j}, C^2_{(t-1),j})$
- Relabel the clusters of R_t
- Until each vector lies in a single distinct cluster.

One Cluster Divisive Procedure

- Let C_i be an already formed cluster. Goal is to split it further, so that the two resulting clusters, C_i^1 and C_i^2 are as "dissimilar" as possible.
- Initially, $C_i^1 = 0$ and $C_i^2 = C_i$.
- Identify the vector in C_i^2 whose avg dissimilarity score is max and move it to C_i^1.
- For each of the remaining $x \in C_i^2$, compute its avg dissimilarity with C_i^1 and also with remaining ones in C_i^2.
- For each fv x, move it to C_i^1, if its avg dissimilarity is lower wrt to C_i^1.

Example for Divisive Clustering

- Let $X = \{x_i, i = 1, \ldots, 5\}$, with $x_1 = [1, 1]'$, $x_2 = [2, 1]'$, $x_3 = [5, 4]'$, $x_4 = [6, 5]'$, and $x_5 = [6.5, 6]'$ and its corresponding dissimilarity matrix, when the Euclidean distance is in use, is
- (x_5) ; (x_4, x_5); (x_3, x_4, x_5);
  $(x_1, x_2)(x_3, x_4, x_5)$;
- $(x_1, x_2)(x_3, x_4, x_5)$;
- $(x_1, x_2)(x_3), (x_4, x_5)$;
- $(x_1, x_2)(x_3), (x_4), (x_5)$;
- $(x_1), (x_2), (x_3), (x_4), (x_5)$;
- Dendogram