CS685: Data Mining
Model-Based Clustering Methods

Arnab Bhattacharya
arnabb@cse.iitk.ac.in

Computer Science and Engineering,
Indian Institute of Technology, Kanpur
http://web.cse.iitk.ac.in/~cs685/

1st semester, 2020-21

Mon 1030-1200 (online)

Probabilistic Model-Based Clustering

- Assume a model for each cluster
- Observed data points are just samples from these models

Probabilistic Model-Based Clustering

- Assume a model for each cluster
- Observed data points are just samples from these models
- Dataset D consists of k clusters, $C_1, \dots, C_k$
- Each cluster has a *prior* probability ω_j that captures its background probability
- Assuming that there are only k clusters, $\sum_{j=1}^{k} \omega_j = 1$
- This constitutes a mixture model

Probabilistic Theory of Clustering

- Probability density function in each C_j is given by f_j

Probabilistic Theory of Clustering

- Probability density function in each C_j is given by f_j
- A data point O_i is generated from a cluster C_j with probability

$$P(O_i|C_j) = f_j(O_i)$$

Probabilistic Theory of Clustering

- Probability density function in each C_j is given by f_j
- A data point O_i is generated from a cluster C_j with probability

$$P(O_i|C_j) = f_j(O_i)$$

- O_i is generated from the mixture model $C = (C_1, \ldots, C_k)$ with probability

$$P(O_i|C) = \sum_{j=1}^{k} (\omega_j . f_j(O_i))$$

Probabilistic Theory of Clustering

- Probability density function in each C_j is given by f_j
- A data point O_i is generated from a cluster C_j with probability

$$P(O_i|C_j) = f_j(O_i)$$

- O_i is generated from the mixture model $C = (C_1, \ldots, C_k)$ with probability

$$P(O_i|C) = \sum_{j=1}^{k} (\omega_j . f_j(O_i))$$

- The entire dataset D is generated from C with probability (assuming all O_i's to be independent)

$$P(D|C) = \prod_{i=1}^{n} P(O_i|C) = \prod_{i=1}^{n} \left[\sum_{j=1}^{k} (\omega_j . f_j(O_i)) \right]$$

Probabilistic Theory of Clustering

- Probability density function in each C_j is given by f_j
- A data point O_i is generated from a cluster C_j with probability

$$P(O_i|C_j) = f_j(O_i)$$

- O_i is generated from the mixture model $C = (C_1, \dots, C_k)$ with probability

$$P(O_i|C) = \sum_{j=1}^{k} (\omega_j . f_j(O_i))$$

- The entire dataset D is generated from C with probability (assuming all O_i's to be independent)

$$P(D|C) = \prod_{i=1}^{n} P(O_i|C) = \prod_{i=1}^{n} \left[\sum_{j=1}^{k} (\omega_j . f_j(O_i)) \right]$$

- Clustering: find C (along with its parameters) that *maximize* $P(D|C)$

Expectation-Maximization (EM)

- How to find the parameters of the clusters?
- May not be analytically solvable or may be computationally intractable

Expectation-Maximization (EM)

- How to find the parameters of the clusters?
- May not be analytically solvable or may be computationally intractable
- Expectation-maximization (EM) framework
- General framework to solve many parameter estimation problems
- Two important steps

Expectation-Maximization (EM)

- How to find the parameters of the clusters?
- May not be analytically solvable or may be computationally intractable
- Expectation-maximization (EM) framework
- General framework to solve many parameter estimation problems
- Two important steps
- Expectation step (E-step): Given the current estimate of the model parameters, distribute (may be probabilistically) the data to models
  - Distribute according to expectation

Expectation-Maximization (EM)

- How to find the parameters of the clusters?
- May not be analytically solvable or may be computationally intractable
- Expectation-maximization (EM) framework
- General framework to solve many parameter estimation problems
- Two important steps
- Expectation step (E-step): Given the current estimate of the model parameters, distribute (may be probabilistically) the data to models
  - Distribute according to expectation
- Maximization step (M-step): Given the current distribution of data to models, update the parameters of the models to maximize the generation of the data
  - Maximize according to data

Expectation-Maximization (EM)

- How to find the parameters of the clusters?
- May not be analytically solvable or may be computationally intractable
- Expectation-maximization (EM) framework
- General framework to solve many parameter estimation problems
- Two important steps
- Expectation step (E-step): Given the current estimate of the model parameters, distribute (may be probabilistically) the data to models
  - Distribute according to expectation
- Maximization step (M-step): Given the current distribution of data to models, update the parameters of the models to maximize the generation of the data
  - Maximize according to data
- Keep iterating

Expectation-Maximization (EM)

- How to find the parameters of the clusters?
- May not be analytically solvable or may be computationally intractable
- Expectation-maximization (EM) framework
- General framework to solve many parameter estimation problems
- Two important steps
- Expectation step (E-step): Given the current estimate of the model parameters, distribute (may be probabilistically) the data to models
  - Distribute according to expectation
- Maximization step (M-step): Given the current distribution of data to models, update the parameters of the models to maximize the generation of the data
  - Maximize according to data
- Keep iterating
- Mostly gets stuck in local maxima

EM for Clustering

- *Expectation step (E-step)*: Given the current cluster parameters, assign each data point to the cluster that is most likely to generate it

EM for Clustering

- *Expectation step (E-step)*: Given the current cluster parameters, assign each data point to the cluster that is most likely to generate it
- *Maximization step (M-step)*: Given the current assignment of data points to a cluster, update cluster parameters such that likelihood of points generated from it is maximum

EM for Clustering

- *Expectation step (E-step)*: Given the current cluster parameters, assign each data point to the cluster that is most likely to generate it
- *Maximization step (M-step)*: Given the current assignment of data points to a cluster, update cluster parameters such that likelihood of points generated from it is maximum
- K-means can be thought of as a type of EM algorithm
  - E-step: Assign a data point to the nearest cluster centre
  - M-step: Update cluster centre to minimize SSE of its points

EM for Mixture Models

- Assume univariate Gaussian mixture models
- Set of parameters $\theta = \{\theta_1, \dots, \theta_k\}$, each $\theta_j = (\mu_j, \sigma_j)$ for Gaussian
- Dataset $D = \{O_1, \dots, O_n\}$

EM for Mixture Models

- Assume univariate Gaussian mixture models
- Set of parameters $\theta = \{\theta_1, \dots, \theta_k\}$, each $\theta_j = (\mu_j, \sigma_j)$ for Gaussian
- Dataset $D = \{O_1, \dots, O_n\}$
- Start with random θ_j's

EM for Mixture Models

- Assume univariate Gaussian mixture models
- Set of parameters $\theta = \{\theta_1, \dots, \theta_k\}$, each $\theta_j = (\mu_j, \sigma_j)$ for Gaussian
- Dataset $D = \{O_1, \dots, O_n\}$
- Start with random θ_j's
- *E-step:* For each point O_i, probability that O_i is assigned to cluster C_j is

$$P(\theta_j | O_i, \theta) = \frac{P(O_i|\theta_j)}{\sum_{l=1}^{k} P(O_i|\theta_l)}$$

EM for Mixture Models

- Assume univariate Gaussian mixture models
- Set of parameters $\theta = \{\theta_1, \dots, \theta_k\}$, each $\theta_j = (\mu_j, \sigma_j)$ for Gaussian
- Dataset $D = \{O_1, \dots, O_n\}$
- Start with random θ_j's
- *E-step:* For each point O_i, probability that O_i is assigned to cluster C_j is

$$P(\theta_j | O_i, \theta) = \frac{P(O_i | \theta_j)}{\sum_{l=1}^{k} P(O_i | \theta_l)}$$

- *M-step:* For each cluster C_j, adjust θ_j such that expected likelihood of points, i.e., $P(O|\theta)$ is maximized

$$\mu_j = \frac{1}{k} \sum_{i=1}^{n} O_i \frac{P(\theta_j | O_i, \theta)}{\sum_{l=1}^{n} P(\theta_j | O_l, \theta)} = \frac{1}{k} \frac{\sum_{i=1}^{n} O_i . P(\theta_j | O_i, \theta)}{\sum_{i=1}^{n} P(\theta_j | O_i, \theta)}$$

$$\sigma_j = \sqrt{\frac{\sum_{i=1}^{n} (O_i - \mu_j)^2 . P(\theta_j | O_i, \theta)}{\sum_{i=1}^{n} P(\theta_j | O_i, \theta)}}$$