Lecture Slides for

INTRODUCTION TO

Machine Learning

ETHEMAŁPAYDIN
© The MIT Press, 2004

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml
CHAPTER 7: Clustering
Semiparametric Density Estimation

- **Parametric**: Assume a single model for $p(x | C_i)$ (Chapter 4 and 5)
- **Semiparametric**: $p(x | C_i)$ is a mixture of densities
  
  Multiple possible explanations/prototypes:
  
  Different handwriting styles, accents in speech
- **Nonparametric**: No model; data speaks for itself (Chapter 8)
Mixture Densities

\[ p(x) = \sum_{i=1}^{k} p(x \mid G_i)P(G_i) \]

where \( G_i \) the components/groups/clusters, 
\( P(G_i) \) mixture proportions (priors), 
\( p(x \mid G_i) \) component densities

Gaussian mixture where \( p(x \mid G_i) \sim \mathcal{N}(\mu_i, \Sigma_i) \)
parameters \( \Phi = \{P(G_i), \mu_i, \Sigma_i\}_{i=1}^{k} \)
unlabeled sample \( \mathcal{X} = \{x_t\}_t \) (unsupervised learning)
Classes vs. Clusters

- **Supervised**: \( \mathcal{X} = \{ \mathbf{x}^t, r^t \}_t \)
- **Classes** \( C_i \ i=1,...,K \)
  \[
p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x} | C_i)P(C_i)
\]
  where \( p(\mathbf{x} | C_i) \sim \mathcal{N}(\mu_i, \Sigma_i) \)
- **\( \Phi = \{P(C_i), \mu_i, \Sigma_i\}_i^{K} \)**
- **\( \hat{P}(C_i) = \frac{\sum_{t} r_i^t}{N} \)
  \( \mathbf{m}_i = \frac{\sum_{t} r_i^t \mathbf{x}^t}{\sum_{t} r_i^t} \)
  \( S_i = \frac{\sum_{t} r_i^t (\mathbf{x}^t - \mathbf{m}_i)(\mathbf{x}^t - \mathbf{m}_i)^T}{\sum_{t} r_i^t} \)

- **Unsupervised**: \( \mathcal{X} = \{ \mathbf{x}^t \}_t \)
- **Clusters** \( G_i \ i=1,...,k \)
  \[
p(\mathbf{x}) = \sum_{i=1}^{k} p(\mathbf{x} | G_i)P(G_i)
\]
  where \( p(\mathbf{x} | G_i) \sim \mathcal{N}(\mu_i, \Sigma_i) \)
- **\( \Phi = \{P(G_i), \mu_i, \Sigma_i\}_i^{k} \)**
  Labels, \( r^t_i \) ?
**k-Means Clustering**

- Find \( k \) reference vectors (prototypes/codebook vectors/codewords) which best represent data
- Reference vectors, \( m_j, j = 1, \ldots, k \)
- Use nearest (most similar) reference:
  \[
  \| \mathbf{x}^t - m_i \| = \min_j \| \mathbf{x}^t - m_j \| 
  \]
- Reconstruction error
  \[
  E\left(\{m_i\}_{i=1}^k | \mathbf{X}\right) = \sum_t \sum_i b_i^t \| \mathbf{x}^t - m_i \| 
  \]
  \[
  b_i^t = \begin{cases} 
  1 & \text{if} \; \| \mathbf{x}^t - m_i \| = \min_j \| \mathbf{x}^t - m_j \| \\
  0 & \text{otherwise} 
  \end{cases} 
  \]
Encoding/Decoding

$$b^t_i = \begin{cases} 1 & \text{if } \|x^t - m_i\| = \min_j \|x^t - m_j\| \\ 0 & \text{otherwise} \end{cases}$$
**k-means Clustering**

Initialize $m_i, i = 1, \ldots, k$, for example, to $k$ random $x^t$

Repeat

For all $x^t \in \mathcal{X}$

$$b_i^t \left\{ \begin{array}{ll} 1 & \text{if } \|x^t - m_i\| = \min_j \|x^t - m_j\| \\ 0 & \text{otherwise} \end{array} \right.$$  

For all $m_i, i = 1, \ldots, k$

$$m_i \leftarrow \frac{\sum_t b_i^t x^t}{\sum_t b_i^t}$$

Until $m_i$ converge
Expectation-Maximization (EM)

- Log likelihood with a mixture model
  \[ \mathcal{L}(\Phi | \mathcal{X}) = \log \prod_t p(x^t | \Phi) \]
  \[= \sum_t \log \sum_{i=1}^k p(x^t | G_i) P(G_i) \]
- Assume hidden variables \( z \), which when known, make optimization much simpler
- Complete likelihood, \( \mathcal{L}_c(\Phi | \mathcal{X}, \mathcal{Z}) \), in terms of \( x \) and \( z \)
- Incomplete likelihood, \( \mathcal{L}(\Phi | \mathcal{X}) \), in terms of \( x \)
E- and M-steps

- Iterate the two steps
  1. E-step: Estimate $z$ given $\mathcal{X}$ and current $\Phi$
  2. M-step: Find new $\Phi'$ given $z$, $\mathcal{X}$, and old $\Phi$.

\[
E\text{- step}: Q(\Phi | \Phi^l) = E[\mathcal{L}_c(\Phi | \mathcal{X}, \mathcal{Z}) | \mathcal{X}, \Phi^l]
\]

\[
M\text{- step}: \Phi^{l+1} = \arg\max_{\Phi} Q(\Phi | \Phi^l)
\]

An increase in $Q$ increases incomplete likelihood

\[
\mathcal{L}(\Phi^{l+1} | \mathcal{X}) \geq \mathcal{L}(\Phi^l | \mathcal{X})
\]
EM in Gaussian Mixtures

- \( z^t_i = 1 \) if \( x^t \) belongs to \( G_i \), 0 otherwise (labels \( r^t_i \) of supervised learning); assume \( p(x|G_i) \sim \mathcal{N}(\mu_i, \Sigma_i) \)

- E-step:
  \[
  E[z^t_i | X, \Phi^t] = \frac{p(x^t | G_i, \Phi^t)P(G_i)}{\sum_j p(x^t | G_j, \Phi^t)P(G_j)} = P(G_i | x^t, \Phi^t) \equiv h_i^t
  \]

- M-step:
  \[
  P(G_i) = \frac{\sum_t h_i^t}{N} \quad m_i^{l+1} = \frac{\sum_t h_i^t x^t}{\sum_t h_i^t} \quad S_i^{l+1} = \frac{\sum_t h_i^t (x^t - m_i^{l+1})(x^t - m_i^{l+1})^T}{\sum_t h_i^t} \]

Use estimated labels in place of unknown labels
$\Pr(G_1|x) = h_1 = 0.5$
Mixtures of Latent Variable Models

- Regularize clusters
  1. Assume shared/diagonal covariance matrices
  2. Use PCA/FA to decrease dimensionality: Mixtures of PCA/FA

\[ p(x_t \mid G_i) = \mathcal{N}(m_i, V_i V_i^T + \psi) \]

Can use EM to learn \( V_i \) (Ghahramani and Hinton, 1997; Tipping and Bishop, 1999)
After Clustering

- Dimensionality reduction methods find correlations between features and group features
- Clustering methods find similarities between instances and group instances
- Allows knowledge extraction through number of clusters, prior probabilities, cluster parameters, i.e., center, range of features.
  Example: CRM, customer segmentation
Clustering as Preprocessing

Estimated group labels $h_j$ (soft) or $b_j$ (hard) may be seen as the dimensions of a new $k$ dimensional space, where we can then learn our discriminant or regressor.

- **Local representation** (only one $b_j$ is 1, all others are 0; only few $h_j$ are nonzero) vs
- **Distributed representation** (After PCA; all $z_j$ are nonzero)
Mixture of Mixtures

- In classification, the input comes from a mixture of classes (supervised).
- If each class is also a mixture, e.g., of Gaussians, (unsupervised), we have a mixture of mixtures:

\[
p(x \mid C_i) = \sum_{j=1}^{k_i} p(x \mid G_{ij}) P(G_{ij})
\]

\[
p(x) = \sum_{i=1}^{K} p(x \mid C_i) P(C_i)
\]
Hierarchical Clustering

- Cluster based on similarities/distances
- Distance measure between instances \( x^r \) and \( x^s \)
  
  Minkowski \((L_p)\) (Euclidean for \( p = 2 \))

\[
d_m(x^r, x^s) = \left[ \sum_{j=1}^{d} (x^r_j - x^s_j)^p \right]^{1/p}
\]

City-block distance

\[
d_{cb}(x^r, x^s) = \sum_{j=1}^{d} |x^r_j - x^s_j|
\]
**Agglomerative Clustering**

- Start with $N$ groups each with one instance and merge two closest groups at each iteration.
- Distance between two groups $G_i$ and $G_j$:
  - Single-link:
    $$d(G_i, G_j) = \min_{x^r \in G_i, x^s \in G_j} d(x^r, x^s)$$
  - Complete-link:
    $$d(G_i, G_j) = \max_{x^r \in G_i, x^s \in G_j} d(x^r, x^s)$$
  - Average-link, centroid
Example: Single-Link Clustering
Choosing $k$

- Defined by the application, e.g., image quantization
- Plot data (after PCA) and check for clusters
- Incremental (leader-cluster) algorithm: Add one at a time until “elbow” (reconstruction error/log likelihood/intergroup distances)
- Manual check for meaning