Latent Variable Models

CS6780 – Advanced Machine Learning
Spring 2019

Thorsten Joachims
Cornell University

Reading: Murphy 11.1 – 11.4.2 and 11.4.7

Clustering as Mixture Modeling

- Setup
  - Learning Task: $P(X)$
  - Training Sample: $S = (\vec{x}_1, \dots, \vec{x}_n)$
  - Hypothesis Space: $H = \{h_1, \dots, h_{|H|}\}$
    - each describes $P(X|h_i)$ where h_i are parameters
  - Goal: learn which $P(X|h_i)$ produces the data
- What to predict?
  - Predict where new points are going to fall

Mixture of Gaussians

Gaussian Mixture Model (GMM):

The data X is generated by

$$P(X = \vec{x}|h) = \sum_{j=1}^{k} P(X = \vec{x}|Y = j, h) P(Y = j)$$

where each mixture component

$$P(X = \vec{x}|Y = j, h) = N\left(X = \vec{x}\middle|\vec{\mu}_j, \Sigma_j\right)$$

and $h = (\vec{\mu}_1, \Sigma_1, \dots, \vec{\mu}_k, \Sigma_k)$.

EM Algorithm for GMM

- EM Algorithm for (simplified) GMM
  - Assume $P(Y)$ and k known and $\Sigma_i = 1$.
  - REPEAT
    - $P(Y = j|X = \vec{x}_i, \vec{\mu}_1, \dots, \vec{\mu}_k) = \frac{P(X=\vec{x}_i|Y=j,\vec{\mu}_j)P(Y=j)}{\sum_{l=1}^{k} P(X=\vec{x}_i|Y=l,\vec{\mu}_l)P(Y=l)} = \frac{e^{-0.5\left(\vec{x}_i-\vec{\mu}_j\right)^2} P(Y=j)}{\sum_{l=1}^{k} e^{-0.5\left(\vec{x}_i-\vec{\mu}_l\right)^2} P(Y=l)}$
    - $\vec{\mu}_j = \frac{\sum_{i=1}^{n} P\left(Y = j\middle|X = \vec{x}_i, \vec{\mu}_1, \dots, \vec{\mu}_k\right)\vec{x}_i}{\sum_{i=1}^{n} P\left(Y = j\middle|X = \vec{x}_i, \vec{\mu}_1, \dots, \vec{\mu}_k\right)}$

Mixture of "X"

General Mixture Model:

The data X is generated by

$$P(X = \vec{x}|h) = \sum_{j=1}^{k} P(X = \vec{x}|Y = j, h) P(Y = j)$$

where each mixture component $P(X = \vec{x}|Y = j, h)$ is

- Gaussian: $N\left(X = \vec{x}\middle|\vec{\mu}_j, \Sigma_j\right)$ [real vectors]
- Independent Bernoullis: $\mathrm{Ber}\left(X = \vec{x}\middle|\vec{\mu}_j\right)$ [bitvectors]
- Independent Poisson: $\mathrm{Po}isson\left(X = \vec{x}\middle|\vec{\mu}_j\right)$ [counts]
- Multinomial: $\mathrm{Mul}\left(X = \vec{x}\middle|\vec{\mu}_j, l\right)$ [counts]

and h collects the respective parameters.

Latent Variable Models

- Data: $(x_1, z_1), \dots, (x_n, z_n)$ where
  - x_i are observed and
  - z_i are unobserved (i.e. latent) (the y_i in mixture).
- Approach: Maximum likelihood (or MAP) by marginalizing over the z_i

$$l(h) = \sum_{i=1}^{n} \log P(x_i|h) = \sum_{i=1}^{n} \log\left[\sum_{z_i} P(x_i, z_i|h)\right]$$

General EM Algorithm

- Data: $(x_1, z_1), \dots, (x_n, z_n)$
- Auxiliary Function:

$$Q(h|q) = \sum_i E_{z_i \sim q_i} [\log P(x_i, z_i | h)] + Ent(q_i)$$

- Algorithm:
  - E-Step: Compute distribution q_i^t of each z_i based on current h^t
  - M-Step: Maximize $Q(h|q^t)$ to get h^{t+1}
- Convergence:

$$l(h^{t+1}) \geq Q(h^{t+1}|q^t) \geq Q(h^t|q^t) = l(h^t)$$

General EM for Mixture Models

- Model:
  - $P(X = x|h) = \sum_{j=1}^{k} P(X = x|Y = j, h)P(Y = j)$
  - Component distributions $P(X = \vec{x}|Y = j, h)$
- Algorithm
  - REPEAT
    - E-Step: $P(Y = j|h) = \frac{P(X=\vec{x}_i|Y=j,h)P(Y=j)}{\sum_{l=1}^{k} P(X=\vec{x}_i|Y=l,h)P(Y=l)}$
    - M-Step: Optimize Q with respect to h

Beyond Mixture Models

- Latent Variable Models for
  - Missing feature imputation (missing features)
  - Semi-supervised learning (missing labels)
  - Censored regression (mortality analysis)
  - Hidden Markov models with unobserved states (speech recognition)
  - Matrix factorization (recommender systems)