IAML: Mixture models and EM

Victor Lavrenko and Charles Sutton
School of Informatics

Semester 1

Mixture models

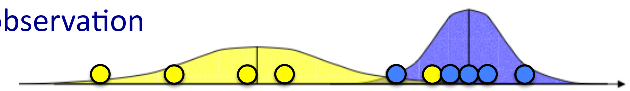
- Recall types of clustering methods
  - hard clustering: clusters do not overlap
    - element either belongs to cluster or it does not
  - soft clustering: clusters may overlap
    - stength of association between clusters and instances
- Mixture models
  - probabilistically-grounded way of doing soft clustering
  - each source: a generative model (Gaussian or multinomial)
  - parameters (e.g. mean/covariance are unknown)
- Expectation Maximization (EM) algorithm
  - automatically discover all parameters for the K "sources"

Mixture models in 1-d

- Observations $x_1 \ldots x_n$
  - K=2 Gaussians with unknown μ, σ^2
  - estimation trivial if we know the source of each observation

$$\mu_b = \frac{x_1 + x_2 + \ldots + x_{n_b}}{n_b}$$

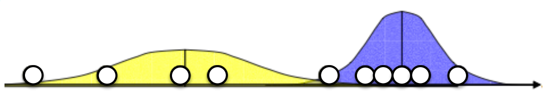
$$\sigma_b^2 = \frac{(x_1 - \mu_1)^2 + \ldots + (x_n - \mu_n)^2}{n_b}$$



- What if we don't know the source?
- If we knew parameters of the Gaussians (μ, σ^2)
  - can guess whether point is more likely to be a or b

$$P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

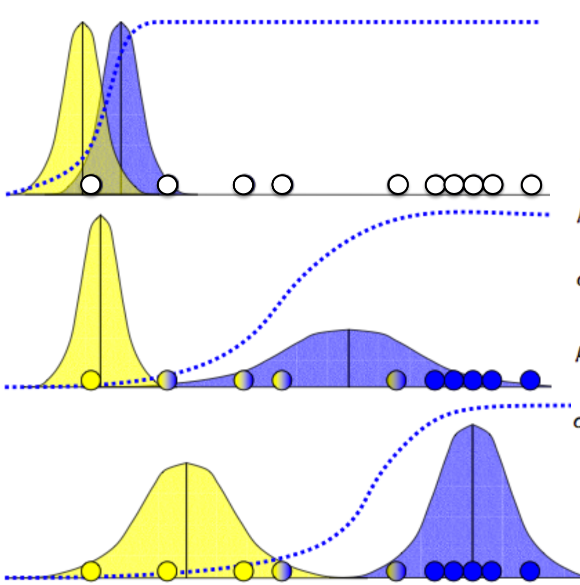
$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left(-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right)$$



Expectation Maximization (EM)

- Chicken and egg problem
  - need (μ_a, σ_a^2) and (μ_b, σ_b^2) to guess source of points
  - need to know source to estimate (μ_a, σ_a^2) and (μ_b, σ_b^2)
- EM algorithm
  - start with two randomly placed Gaussians (μ_a, σ_a^2), (μ_b, σ_b^2)
  - E-step: for each point: $P(b|x_i)$ = does it look like it came from b?
  - M-step: adjust (μ_a, σ_a^2) and (μ_b, σ_b^2) to fit points assigned to them
  - iterate until convergence

EM: 1-d example



$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left(-\frac{(x_i-\mu_b)^2}{2\sigma_b^2}\right)$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \ldots + b_n x_{n_b}}{b_1 + b_2 + \ldots + b_n}$$

$$\sigma_b^2 = \frac{b_1(x_1-\mu_1)^2 + \ldots + b_n(x_n-\mu_n)^2}{b_1 + b_2 + \ldots + b_n}$$

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \ldots + a_n x_{n_b}}{a_1 + a_2 + \ldots + a_n}$$

$$\sigma_a^2 = \frac{a_1(x_1-\mu_1)^2 + \ldots + a_n(x_n-\mu_n)^2}{a_1 + a_2 + \ldots + a_n}$$

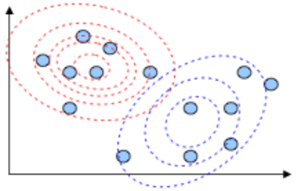
could also estimate priors:

$P(b) = (b_1 + b_2 + \ldots b_n) / n$

$P(a) = 1 - P(b)$

Gaussian mixture models: d>1

- Data with d attributes, from k sources
- Each source c is a Gaussian
- Iteratively estimate parameters:
  - prior: what % of instances came from source c? $P(c) = \frac{1}{n}\sum_{i=1}^{n} P(c \mid \vec{x}_i)$
  - mean: expected value of attribute j from source c $\mu_{c,j} = \sum_{i=1}^{n}\left(\frac{P(c \mid \vec{x}_i)}{nP(c)}\right) x_{i,j}$
  - covariance: how correlated are attributes j and k in source c? $(\Sigma_c)_{j,k} = \sum_{i=1}^{n}\left(\frac{P(c \mid \vec{x}_i)}{nP(c)}\right)(x_{i,j}-\mu_{c,j})(x_{i,k}-\mu_{c,k})$
  - based on: our guess of the source for each instance



$$P(c \mid \vec{x}_i) = \frac{P(\vec{x}_i \mid c)P(c)}{\sum_{c'=1}^{k} P(\vec{x}_i \mid c')P(c')}$$

$$P(\vec{x}_i \mid c) = \frac{1}{\sqrt{2\pi|\Sigma_c|}} \exp\left(-\frac{1}{2}\overbrace{\underbrace{(\vec{x}_i-\vec{\mu}_c)^T \Sigma_c^{-1} (\vec{x}_i-\vec{\mu}_c)}}^{}\right)$$

$$\sum_{j=1}^{d}\sum_{k=1}^{d}(x_{i,j}-\mu_{c,j})(\Sigma_c^{-1})_{j,k}(x_{i,k}-\mu_{c,k})$$

How to pick K?

- Probabilistic model $L = \log P(x_1 \ldots x_n) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} P(x_i \mid k)P(k)$
  - tries to "fit" the data (maximize likelihood)
- Pick K that makes L as large as possible?
  - $K = n$: each data point has its own "source"
  - may not work well for new data points
- Split points into training set T and validation set V
  - for each K: fit parameters of T, measure likelihood of V
  - sometimes still best when $K = n$
- Occam's razor: pick "simplest" of all models that fit
  - Bayes Inf. Criterion (BIC): $\max_p \{ L - ½\, p \log n \}$
  - Akaike Inf. Criterion (AIC): $\min_p \{ 2p - L \}$

L … **likelihood**, how well our model fits the data

p … **number of parameters** how "simple" is the model

Summary

- Walked through 1-d version
  - works for higher dimensions
    - d-dimensional Gaussians, can be non-spherical
  - works for discrete data (text)
    - d-dimensional multinomial distributions (pLSI)
- Maximizes likelihood of the data: $P(x_1 \ldots x_n) = \prod_{i=1}^{n}\sum_{k=1}^{K} P(x_i \mid k)P(k)$
- Very similar to K-means
  - sensitive to starting point, converges to a local maximum
  - convergence: when change in $P(x_1 \ldots x_n)$ is sufficiently small
  - cannot discover K (likelihood keeps growing with K)