- Today's Reading:
  - HMS, 9.6, 6.4, 7.3.2, 7.4, 8.4

- Today's Lecture:
  - Probabilistic Model-based Clustering
  - Descriptive Modeling

- Upcoming Due Dates:
  - H2 due Thu 3/7

Families of Clustering Algorithms

- Partition-based methods
  - e.g., K-means
- Hierarchical clustering
  - e.g., hierarchical agglomerative clustering
- Probabilistic model-based clustering
  - e.g., mixture models

Probabilistic Model-based Clusterir

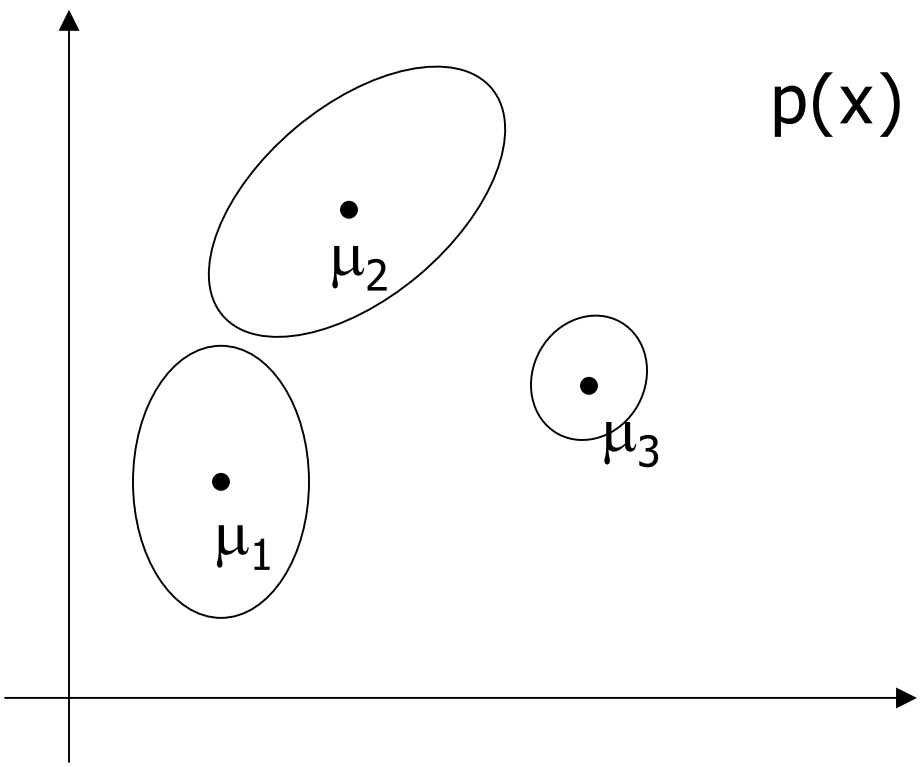
- Assume a probability model for each compone cluster
- Mixture Model:

$$f(x) = \sum_{k=1}^{K} w_k f_k(x; \theta_k)$$

- where f_k are component distributions
- components: gaussian, poisson, exponential

Gaussian Mixture Models (GMM)

- K components
- model for each component cluster $N(\mu_k, \sigma_k)$

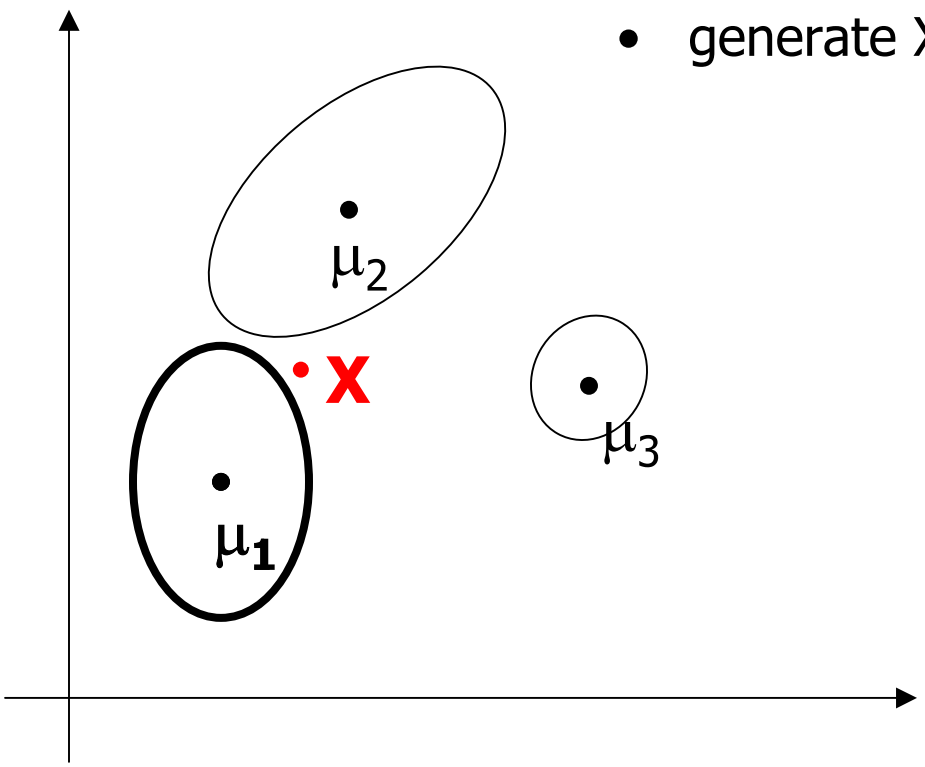


$$p(x) = \sum_{k=1}^{K} w_k f(x; \mu_k,$$

much borrowed from Andrew Moore's Clustering Gaussian Mixture

GMM cont.

- Generative Model
  - choose component with proba
  - generate X ~ N(μ_k,σ_k)



GMM cont.

- But, we need to learn a model from the data..
- D = {x(1), …, x(n)}

First, suppose we know w_1, …, w_k
$P(x \mid w_j, \mu_1, \ldots, \mu_k, \sigma_1, \ldots, \sigma_k) = ?$

Next,
$P(x \mid \mu_1, \ldots, \mu_k, \sigma_1, \ldots, \sigma_k) = ?$

And,
$P(D \mid \mu_1, \ldots, \mu_k, \sigma_1, \ldots, \sigma_k) = ?$

Look Familiar?

- P(D | μ_1 , …, μ_k , σ_1 , …, σ_k) gives us the probability for particular choice of μ_1 , …, μ_k , σ_1 , …, σ_k
- How do we choose μ_i σ_i?
- Find μ_i that maximize the likelihood
- Set

$$\frac{\partial}{\partial \mu_j} \log P(D \mid \mu_1, \ldots, \mu_k, \sigma_1, \ldots, \sigma_k) = 0$$

  and solve for μ_i

- Problem: we have a bunch on non-linear non-analytical solvable equations
- One solution: gradient descent…. slow
- instead….

Expectation Maximization (EM)

- Dempster, Laird, and Rubin, 1977
- **extremely** popular recently
- applicable in a wide range of problems
- many uses: hidden markov models,
- basic idea is quite simple…

EM setup

- Let D = {x(1), …, x(n)} be n observed data vectors
- Let H = {z(1),…, z(n)} be n values of hidden variable Z (might be the cluster labels)
- Then the log-likelihood of the observed data is

$$l(\theta) = \log p(D \mid \theta) = \log \sum_H p(D, H \mid \theta)$$

- *both* θ and H are unknown
- Let Q(H) be any probability distribution for H.

$$\begin{aligned} l(\theta) &= \log \sum_H p(D, H \mid \theta) \\ &= \log \sum_H Q(H) \frac{p(D, H \mid \theta)}{Q(H)} \\ &\geq \sum_H Q(H) \log \frac{p(D, H \mid \theta)}{Q(H)} \\ &= \sum_H Q(H) \log p(D, H \mid \theta) + \sum_H Q(H) \log \frac{1}{Q} \\ &= F(Q, \theta) \end{aligned}$$

lower bound on $l(\theta)$

Jensen's Inequality

- http://www.engineering.usu.edu/classes/ece/7
cture2/node5.html

EM Algorithm

- EM algorithm alternates between

Intuition:
•In the E-step, we estimate the distribution on the hidden
conditioned on a particular setting of the parameter vecto
•In the M-step, we choose new set of parameters θ^{k+1} to m
the expected log-likelihood of observed data

- Maximum for E step:

$$Q^{k+1} = p(H \mid D, \theta^k)$$

- Maximum for M step:

$$\theta^{k+1} = \arg \max_{\theta} \sum_{H} p(H \mid D, \theta^k) \log p(D, H \mid \theta^k)$$

Notes

- Often both the E and M step can be solved in
  form
- Neither the E step nor the M step can decreas
  log-likelihood
- Under relatively general conditions the algorith
  guaranteed to converge to a local maximum o
  likelihood
- We must specify a starting point for the algori
  for example a random choice of θ or Q
- We must specify stopping criteria, or converge
  detection
- Computational complexity: number of iteratio
  time to compute E and M steps

Back to our example…

- We want to fit a normal mixture distribution:

$$f(x) = \sum_{k=1}^{K} w_k f_k(x; \mu_k, \sigma_k)$$

w_k is prior probability of x belonging to component

$\theta = (w_1, \ldots. w_k, \mu_1, \ldots, \mu_k, \sigma_1, \ldots, \sigma_k)$ is parameter vector

then probability of data point x coming from kth class is:

E-step: $$\hat{P}(k \mid x) = \frac{w_k f_k(x; \mu_k, \sigma_k)}{f(x)}$$

M-step: $$\hat{w}_k = \frac{1}{n} \sum_{k=1}^{n} \hat{P}(k \mid x(i)) \qquad \hat{\mu}_k = \frac{1}{n w_k} \sum_{k=1}^{n} \hat{P}(k \mid x($$

$$\hat{\sigma}_k = \frac{1}{n w_k} \sum_{k=1}^{n} \hat{P}(k \mid x(i))(x(i) - \mu_k)^2$$

EM Demo

EM demo

http://diwww.epfl.ch/mantra/tutorial/english/gaussi

EM Comments

- complexity of EM for multivariate gaussian mix
  with K components: dominated by calculation
  covariance matrices.
  - With p dimensions, $O(Kp^2)$ covariance parameters t
    estimated
  - Each requires summing over n data points and clust
    weights, leading to $O(Kp^2n)$ per step
- Often times there are large increases in likelih
  over first few iteration and then can slowly co
  likelihood as function of iterations not necessa
  concave

Likelihood as function of EM Iterati

figure 9.2 in text

Probabilistic Model-based clusterin

- model provides full distributional description fo component; we may be able to interpret differ in the distributions
- Given the model, each data point has a K-com vector of probabilities that it belongs to each g (why this is called soft-clustering)
- EM algorithm can be extended for MAP and Ba estimation
- Method can be extended to data that are not i dimensional vector form. mixtures of sequenc models, e.g. markov models, others
- key cost: assumption of parametric model

and finally...

how do we choose K?

How to choose K

- Choose K that maximizes likelihood?
- NOT.
- As K is increased, the value of the likelihood a
  maximum cannot decrease
- Problem of scoring models with different comp

Model Selection

- First, as we mentioned before, there are two p goals in constructing a model
- Goal #1: summarization
  - we would like to describe the data as precisely as p
  - given that our model is comprehensible (subjective) compact (objective)
  - general approach based on data compression and information-theoretic arguments results in a score f

score(θ,M) = # of bits to describe the data given the m number of bits to describe the model

$$\text{score}(\theta, M) = \underbrace{-\log p(D \mid \theta, M)}_{\text{\# of bits to transmit the data that is not accounted for by the model}} \underbrace{-\log p(\theta, M)}_{\text{\# of bits to transm the model}}$$

Model Selection, cont.

- Goal #2: generalize from the available data to
  data
  - goodness of fit is part of the objective
  - but, since the data is not the entire population, we
    learn a model that will generalize to other new data
    instances

- In both cases, we want a score function that s
  compromise between how well the model fits
  data and the simplicity of the model, although
  theoretical motivations for this are quite differ

Bias-Variance

- Model too flexible $\Rightarrow$ overfit the data $\Rightarrow$ high v
- Model too restrictive $\Rightarrow$ can't fit the data $\Rightarrow$ h
- Bias-variance tradeoff: compromise

Penalize Complexity

- Score(M) = error(M) + penalty-function(M)
- Penalty-function:
  - M_1, …, M_K K models
  - θ_k best parameters
  - complexity of model M depends number of paramet
- AIC - Akaike information criterion:

$$S_{AIC}(M_k) = 2S_L(\hat{\theta}_k; M_k) + 2d_k$$

- BIC – Bayesian information criterion

$$S_{BIC}(M_k) = 2S_L(\hat{\theta}_k; M_k) + d_k \log n$$

- others –Minimum Description Length (MDL), Structural Risk Minimization (SRM)

Bayesian Approach

- compute posterior probability of each model d
  and select the one with highest posterior:

$$p(M_k \mid D) \propto p(D \mid M_k)p(M_k)$$
$$= \int p(D, \theta_k \mid M_k)p(M_k)d\theta_k$$
$$= \int p(D \mid, \theta_k \mid M_k)p(M_k)d\theta_k$$

implicitly penalizes complexity, since higher dimen
parameter spaces (more complex models) will mea
probability mass in $p(\theta_k \mid M_K)$ is spread more thinl

often computationally intractable; use MCMC to es
use approximation.

External Validation

- General Idea:
  - Split data randomly into two disjoint sets
  - the training set
  - the validation set
  - construct a model based on the training set
  - compute an estimate for the score based on the val set
  - this estimate for the score will be an unbiased estim

External Validation, cont.

- popular method: Cross-validation
  - repeatedly split the data into train and validation se
    average results
  - n-fold cross-validation
  - leave one out cross-validation

Summary

- Families of Clustering Algorithms
  - Partition-based methods
  - Hierarchical clustering
  - Probabilistic model-based clustering
- EM Algorithm
- Methods for scoring models with different complexities

Next Time

- Density Estimation
- Reading: ch 9

References

- *Principles of Data Mining*, Hand, Mannila, Smy
  MIT Press, 2001.
- Andrew Moore's excellent lecture notes, http:/
  2.cs.cmu.edu/~awm/781/