Comp 135 Introduction to Machine Learning and Data Mining

Fall 2016

Professor: Roni Khardon

Computer Science
Tufts University

Soft k-Means Clustering

- Pick k cluster centers
- Repeat:
  - Associate examples with centers
    $p_{i,j}$ ~~ similarity b/w center i and ex j
  - Re-calculate means
    as weighted average of examples in cluster
- Until convergence

Mixture Models

- Motivated by soft k-means
- We develop a "generative model" for clustering:
  - Assume there are *k* clusters
  - Clusters are not required to have the same number of points
  - And not required to have the same shape

Mixture of Normals in 1D

Repeat for $i = 1, \ldots, N$

Pick cluster Id z_i from discrete distribution with parameters $p_1, p_2, \ldots, p_k$

Note: $z_i \in \{1, 2, \ldots, k\}$

Pick the example x_i from normal distribution with parameters μ_{z_i}, σ_{z_i}

Example: when $z_i = 3$ using μ_3 and σ_3

Given a dataset generated by this process the clustering task is to identify the parameters $\{p_j, \mu_j, \sigma_j\}$ $j = 1, \ldots, k$

Mixture of Normals in 1D

Repeat for $i = 1, \ldots, N$

Pick cluster Id z_i from discrete distribution with parameters $p_1, p_2, \ldots, p_k$

Note: $z_i \in \{1, 2, \ldots, k\}$

Pick the example x_i from normal distribution with parameters μ_{z_i}, σ_{z_i}

Example: when $z_i = 3$ using μ_3 and σ_3

To simplify analysis in class we assume $\forall j, p_j = 1/k$ and $\forall j, \sigma_j = \sigma,$ are known

Maximum likelihood estimation

- First analyze assuming z_i are known
- Convenient notation: represent the number z_i as a "unit vector" bit sequence
- Example: k=4 $z_i = 1 \Rightarrow 1000$
  $z_i = 2 \Rightarrow 0100$
  $z_i = 3 \Rightarrow 0010$
  $z_i = 4 \Rightarrow 0001$
- Notation: $z_{i,j}$ is j'th bit within z_i

$z_i = 2 \Rightarrow 0100 \Rightarrow z_{i,2} = 1$ $z_{i,3} = 0$

Maximum likelihood estimation

- First analyze assuming z_i are known
- The Complete Data includes all the x_i, z_i

$$Data = (x_1, z_1), (x_2, z_2), \ldots, (x_N, z_N)$$

comp135 Roni Khardon, Tufts University

Maximum likelihood estimation

- The Likelihood

$$L = \prod_i p(z_i) p(x_i | z_i, \mu_{z_i})$$

$$= \prod_i (1/k) \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x_i - \mu_{z_i})^2}$$

$$= \prod_i (1/k) \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2} \boxed{\sum_j z_{i,j} (x_i - \mu_j)^2}}$$

Notation trick: exactly one term remains from the sum!

comp135 Roni Khardon, Tufts University

Maximum likelihood estimation

$$L = \prod_i (1/k) \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2} \sum_j z_{i,j} (x_i - \mu_j)^2}$$

$$LogL = \text{const} - \frac{1}{2\sigma^2} \sum_i \sum_j z_{i,j} (x_i - \mu_j)^2$$

$$\frac{\partial LogL}{\mu_j} = \ldots = 0 \quad \Rightarrow$$

$$\mu_j = \frac{\sum_i z_{i,j} \; x_i}{\sum_i z_{i,j}}$$

This is not surprising.

Why?

comp135 Roni Khardon, Tufts University

Maximum likelihood estimation

- First analyze assuming z_i are known
- The Complete Data includes all the x_i, z_i

$$Data = (x_1, z_1), (x_2, z_2), \ldots, (x_N, z_N)$$

- The Observed Data includes all the x_i

$$Data = x_1, x_2 \ldots, x_N$$

- → Cannot use previous estimate.
- What is the likelihood in this case?

comp135 Roni Khardon, Tufts University

Maximum likelihood estimation

- The Observed Data includes all the x_i

$$Data = x_1, x_2 \ldots, x_N$$

- Maximum likelihood prescribes that we should optimize:

$$p(\text{observed}) = p(x_1, \ldots, x_N)$$

$$= \sum_{z_1} \sum_{z_2} \ldots \sum_{z_N} p(x_1, \ldots, x_N, z_1, \ldots, z_N)$$

The Equation for the likelihood needs to sum out (marginalize) over the z_i. No simple closed form.

comp135 Roni Khardon, Tufts University

The EM Algorithm

- A general algorithm for maximizing likelihood when we have hidden random variables
- The algorithm has a simple form when applied to mixture models
- We will constrain ourselves to that simple form
- And will mention the general scheme of the EM algorithm briefly

comp135 Roni Khardon, Tufts University

The EM Algorithm

- EM is an iterative algorithm
- Initialize probability model p'
- Repeat
  - use p' to calculate an improved model p''
  - Set $p'=p''$
- Until no further improvement

EM Algorithm for Mixture Models

- Repeat
  - [E] Calculate using p'

$$f_{i,j} = E_{p(Z|X,\{\mu'_\ell\})}[z_{i,j}] = p(z_i = j | \{\mu'_\ell\}, Data)$$

  - [M] Estimate p'' parameters using max likelihood solution of the complete data by replacing the unknown $z_{i,j}$ by $f_{i,j}$

EM for Mixtures in 1D

- [E] Calculate

$$f_{i,j} = E_{p(Z|X,\{\mu'_\ell\})}[z_{i,j}] = p(z_i = j | \{\mu'_\ell\}, Data)$$

$$f_{i,j} = \frac{p((z_i = j) \text{ and } x_i)}{p(x_i)}$$

$$= \frac{p((z_i = j) \text{ and } x_i)}{\sum_\ell p((z_i = \ell) \text{ and } x_i)}$$

$$= \frac{(1/k)\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x_i - \mu'_j)^2}}{\sum_\ell (1/k)\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x_i - \mu'_\ell)^2}}$$

First part holds for any mixture model.

EM for Mixtures in 1D

- [M] Estimate parameters using max likelihood replacing the unknown $z_{i,j}$ by $f_{i,j}$

$$\mu_j = \frac{\sum_i z_{i,j}\, x_i}{\sum_i z_{i,j}} \quad \Rightarrow$$

$$\mu''_j = \frac{\sum_i f_{i,j}\, x_i}{\sum_i f_{i,j}}$$

EM for Mixtures in 1D

- [E] Calculate for all i,j

$$f_{i,j} = \frac{(1/k)\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x_i - \mu'_j)^2}}{\sum_\ell (1/k)\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x_i - \mu'_\ell)^2}}$$

- [M] Calculate for all j

$$\mu''_j = \frac{\sum_i f_{i,j}\, x_i}{\sum_i f_{i,j}}$$

- Assign for all j: $\mu'_j = \mu''_j$

General form of EM

- Define an auxiliary function *Q(p',p")*
- Relative to observed variables *O* and hidden variables *H*

$$Q(p', p'') = E_{p'(H|O)}[\log p''(H, O)]$$

The EM Algorithm

- EM is an iterative algorithm
- Initialize probability model *p′*
- Repeat
  - use *p′* to calculate an improved model *p″*
  - Set *p′*=*p″*
- Until no further improvement

The EM Algorithm

- EM is an iterative algorithm
- Initialize probability model *p′*
- Repeat
  - Pick *p″* so as to maximize *Q(p′,p″)*
  - Set *p′*=*p″*
- Until no further improvement

EM Algorithm for Mixture Models

- Repeat
  - [E] Calculate using *p′*

    $$f_{i,j} = E_{p(Z|X,\{\mu'_\ell\})}[z_{i,j}] = p(z_i = j|\{\mu'_\ell\}, Data)$$

  - [M] Estimate *p″* parameters using max likelihood replacing the unknown $z_{i,j}$ by $f_{i,j}$

Using the same methodology on any mixture model (not just Gaussian) yields the same template.

Semi-Supervised Naïve Bayes Model

- Naïve Bayes: Probabilistic model with strong simplifying assumptions
- Illustrating application: text categorization where we have data for

  (document$_i$,label$_i$)

- What if we have many documents but labels for only a few of them?
- Can the unlabeled documents help?

Semi-Supervised Naïve Bayes Model

- What if we have many documents but labels for only a few of them?
- Can the unlabeled documents help?
- Before exploring this question we will develop the EM algorithm for this model where the labels are not known

Recall: Naïve Bayes Model

- Each class induces a distribution over features.
- Features are conditionally independent given the class
- In these slides I use the model with binary features

Recall: Naïve Bayes Model

$$p(z_i = j) = p_j$$

$$p(x_{i,\ell} = 1 | \text{class } j) = q_{j,\ell}$$

$$p(x_i | \text{class } j) = \prod_\ell q_{j,\ell}^{x_{i,\ell}} (1 - q_{j,\ell})^{(1-x_{i,\ell})}$$

$$p(z_i = j \text{ and } x_i) = p_j \prod_\ell q_{j,\ell}^{x_{i,\ell}} (1 - q_{j,\ell})^{(1-x_{i,\ell})}$$

$$p(z_i \text{ and } x_i) = \prod_j \left[p_j \prod_\ell q_{j,\ell}^{x_{i,\ell}} (1 - q_{j,\ell})^{(1-x_{i,\ell})} \right]^{z_{i,j}}$$

comp135 Roni Khardon, Tufts University

Recall: Maximum Likelihood

$$p_j = p(z_i = j) = \frac{\text{number of examples with class } j}{\text{number of examples}}$$

$$q_{j,\ell} = p(x_{i,\ell} = 1 | z_i = j) = \frac{\text{num of ex with class } j \text{ and } x_{\cdot,\ell} = 1}{\text{number of examples with class } j}$$

comp135 Roni Khardon, Tufts University

Naïve Bayes as Mixture Model

Repeat for $i = 1, \ldots, N$

Pick cluster Id z_i from discrete distribution with parameters $p_1, p_2, \ldots, p_k$

Pick the example x_i from Naive Bayes distribution with parameters q_{z_i}

comp135 Roni Khardon, Tufts University

EM Algorithm

- Complete Data Likelihood

$$L = \prod_i \prod_j \left[p_j \prod_\ell q_{j,\ell}^{x_{i,\ell}} (1 - q_{j,\ell})^{(1-x_{i,\ell})} \right]^{z_{i,j}}$$

- Log Likelihood

$$LogL = \sum_i \sum_j z_{i,j} (\log p_j + \sum_\ell x_{i,\ell} \log q_{j,\ell} + (1 - x_{i,\ell}) \log(1 - q_{j,\ell}))$$

comp135 Roni Khardon, Tufts University

EM Algorithm

- Maximum Likelihood for complete data

$$LogL = \sum_i \sum_j z_{i,j} (\log p_j + \sum_\ell x_{i,\ell} \log q_{j,\ell} + (1 - x_{i,\ell}) \log(1 - q_{j,\ell}))$$

[we already solved this a few lectures ago]

$$p_j = \frac{\sum_i z_{i,j}}{N}$$

$$q_{j,\ell} = \frac{\sum_i z_{i,j} x_{i,\ell}}{\sum_i z_{i,j}}$$

comp135 Roni Khardon, Tufts University

EM Algorithm

- E Step: Calculating $f_{i,j}$

$$f_{i,j} = E_{p'(Z|X)}[z_{i,j}] = \frac{p'(z_i = j \text{ and } x_i)}{\sum_c p'(z_i = c \text{ and } x_i)}$$

$$= \frac{p'_j \prod_\ell {q'}_{j,\ell}^{x_{i,\ell}} (1 - {q'}_{j,\ell})^{(1-x_{i,\ell})}}{\sum_c p'_c \prod_\ell {q'}_{c,\ell}^{x_{i,\ell}} (1 - {q'}_{c,\ell})^{(1-x_{i,\ell})}}$$

comp135 Roni Khardon, Tufts University

EM Algorithm for Naïve Bayes

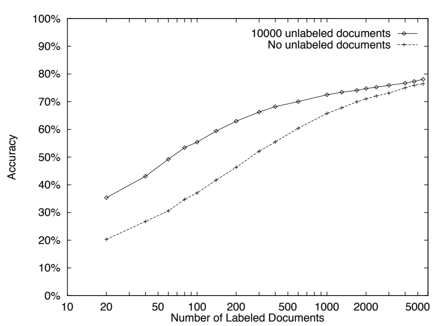
- Repeat
  - Calculate:
$$f_{i,j} = \frac{p'_j \prod_\ell q'^{x_{i,\ell}}_{j,\ell}(1-q'_{j,\ell})^{(1-x_{i,\ell})}}{\sum_c p'_c \prod_\ell q'^{x_{i,\ell}}_{c,\ell}(1-q'_{c,\ell})^{(1-x_{i,\ell})}}$$
  - Calculate:
$$p''_j = \frac{\sum_i f_{i,j}}{N}$$
$$q''_{j,\ell} = \frac{\sum_i f_{i,j} x_{i,\ell}}{\sum_i f_{i,j}}$$
  - Assign: p'←p'' and q'← q''

Semi-Supervised Naïve Bayes Model

- Naïve Bayes for text categorization
- What if we have many documents but labels for only a few of them?
- Can the unlabeled documents help?

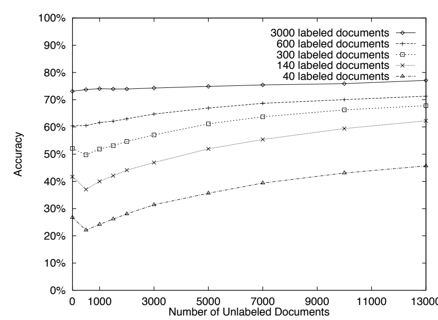
- Use EM: for examples where z_i is known use $f_{i,j}=z_{i,j}$ instead of estimating it
- Nothing else changes in the algorithm!

20 newgroups data



[From Nigam et all MLJ 1999.]

20 newgroups data



[From Nigam et all MLJ 1999.]

Summary

- EM is a general algorithmic framework for inference with hidden random variables
- It takes a simple form for mixture models alternating between estimating "fractional memberships" and using these in maximum likelihood calculations.
- General derivation through the Q(p',p'') function is applicable in more complex models.
- Mixture model easily generalizes to capture semi-supervised learning