# Machine Learning 10-601


Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University


October 9, 2012

Today:

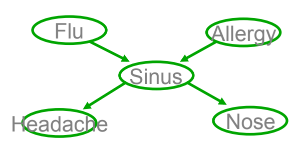
- Graphical models
- Bayes Nets:
  - Inference
  - Learning

Readings:

Required:

- Bishop chapter 9 through 9.2

## Estimate $\theta$ from partly observed data




- What if FAHN observed, but not S?
- Can't calculate MLE

$$\theta \leftarrow \arg\max_\theta \log \prod_k P(f_k, a_k, s_k, h_k, n_k | \theta)$$

- Let X be all *observed* variable values (over all examples)
- Let Z be all *unobserved* variable values
- Can't calculate MLE:

$$E[f(x)] = \sum_x P(X=x) f(x)$$

$$\theta_{MLE} \leftarrow \arg\max_\theta \log P(X, Z | \theta)$$

- EM seeks* to estimate:

$$\theta \leftarrow \arg\max_\theta E_{Z|X,\theta}[\log P(X, Z|\theta)]$$

$P(Z|X,\theta)$

* EM guaranteed to find local maximum

## EM Algorithm - Informally

EM is a general procedure for learning from partly observed data

Given observed variables X, unobserved Z (X={F,A,H,N}, Z={S})

Begin with arbitrary choice for parameters $\theta$

Iterate until convergence:

- E Step: estimate the values of unobserved Z, using $\theta$
- M Step: use observed X, plus E-step estimates for Z to derive a better $\theta$

Guaranteed to find local maximum.
Each iteration increases $E_{P(Z|X,\theta)}[\log P(X,Z|\theta')]$

## EM Algorithm

EM is a general procedure for learning from partly observed data

Given observed variables X, unobserved Z (X={F,A,H,N}, Z={S}) ✓

Define $Q(\theta'|\theta) = E_{P(Z|X,\theta)}[\log P(X,Z|\theta')]$

↑ current ↑ M step new

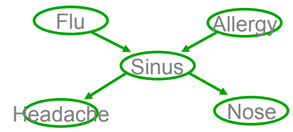
Iterate until convergence:

- E Step: Use X and current $\theta$ to calculate P(Z|X,$\theta$)
- M Step: Replace current $\theta$ by

$$\theta \leftarrow \arg\max_{\theta'} Q(\theta'|\theta)$$

Guaranteed to find local maximum.
Each iteration increases $E_{P(Z|X,\theta)}[\log P(X,Z|\theta')]$

## EM and estimating $\theta_{s|ij}$




observed X = {F,A,H,N}, unobserved Z={S}

E step: Calculate $P(Z_k|X_k; \theta)$ for each training example, k

$$P(S_k = 1|f_k a_k h_k n_k, \theta) = E[s_k] = \frac{P(S_k = 1, f_k a_k h_k n_k|\theta)}{P(S_k = 1, f_k a_k h_k n_k|\theta) + P(S_k = 0, f_k a_k h_k n_k|\theta)}$$

$P(z|x;\theta)$ $P(S=s_k|X)$

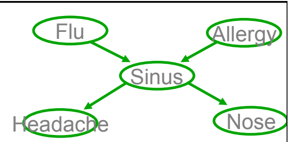
M step: update all relevant parameters. For example:

$$\theta_{s|ij} \leftarrow \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)\ E[s_k]}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

1 if args is true $E[s_k] = P(s_k=1)$ $\theta$ h|s

Recall MLE was: $\theta_{s|ij} = \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$

## EM and estimating $\theta$




More generally,
Given observed set X, unobserved set Z of boolean values

E step: Calculate for each training example, k

the expected value of each unobserved variable

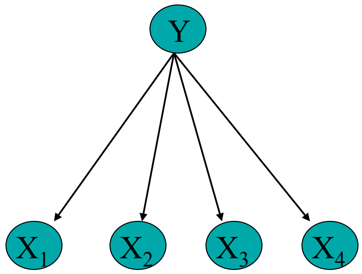
M step:

Calculate estimates similar to MLE, but
replacing each count by its <u>expected count</u>

$$\delta(Y = 1) \rightarrow E_{Z|X,\theta}[Y] \qquad \delta(Y = 0) \rightarrow (1 - E_{Z|X,\theta}[Y])$$

## Using Unlabeled Data to Help Train Naïve Bayes Classifier

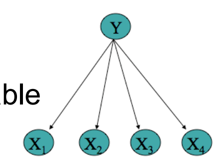
Learn P(Y|X)



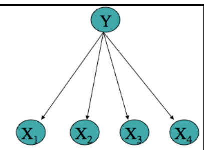

| Y | X1 | X2 | X3 | X4 |
|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| ? | 0 | 1 | 1 | 0 |
| ? | 0 | 1 | 0 | 1 |

E step: Calculate for each training example, k

the expected value of each unobserved variable



## EM and estimating $\theta$




Given observed set X, unobserved set Y of boolean values

E step: Calculate for each training example, k

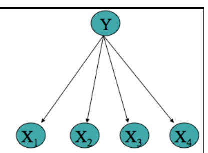
the expected value of each unobserved variable Y

$$E_{P(Y|X_1 \ldots X_N)}[y(k)] = P(y(k) = 1 | x_1(k), \ldots x_N(k); \theta) = \frac{P(y(k)=1) \prod_i P(x_i(k)|y(k)=1)}{\sum_{j=0}^{1} P(y(k)=j) \prod_i P(x_i(k)|y(k)=j)}$$

M step: Calculate estimates similar to MLE, but replacing each count by its expected count

let's use y(k) to indicate value of Y on kth example


## EM and estimating $\theta$



Given observed set X, unobserved set Y of boolean values

E step: Calculate for each training example, k

the expected value of each unobserved variable Y

$$E_{P(Y|X_1 \ldots X_N)}[y(k)] = P(y(k) = 1 | x_1(k), \ldots x_N(k); \theta) = \frac{P(y(k)=1) \prod_i P(x_i(k)|y(k)=1)}{\sum_{j=0}^{1} P(y(k)=j) \prod_i P(x_i(k)|y(k)=j)}$$

M step: Calculate estimates similar to MLE, but replacing each count by its expected count


$$\theta_{ij|m} = \hat{P}(X_i = j | Y = m) = \frac{\sum_k P(y(k) = m | x_1(k) \ldots x_N(k)) \ \delta(x_i(k) = j)}{\sum_k P(y(k) = m | x_1(k) \ldots x_N(k))}$$

MLE would be: $\hat{P}(X_i = j | Y = m) = \frac{\sum_k \delta((y(k) = m) \wedge (x_i(k) = j))}{\sum_k \delta(y(k) = m)}$

- **Inputs:** Collections $\mathcal{D}^l$ of labeled documents and $\mathcal{D}^u$ of unlabeled documents.
- Build an initial naive Bayes classifier, $\hat{\theta}$, from the labeled documents, $\mathcal{D}^l$, only. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg\max_\theta \mathrm{P}(\mathcal{D}|\theta)\mathrm{P}(\theta)$ (see Equations 5 and 6).
- Loop while classifier parameters improve, as measured by the change in $l_c(\theta|\mathcal{D};\mathbf{z})$ (the complete log probability of the labeled and unlabeled data
  - **(E-step)** Use the current classifier, $\hat{\theta}$, to estimate component membership of each unlabeled document, *i.e.*, the probability that each mixture component (and class) generated each document, $\mathrm{P}(c_j|d_i;\hat{\theta})$ (see Equation 7).
  - **(M-step)** Re-estimate the classifier, $\hat{\theta}$, given the estimated component membership of each document. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg\max_\theta \mathrm{P}(\mathcal{D}|\theta)\mathrm{P}(\theta)$ (see Equations 5 and 6).
- **Output:** A classifier, $\hat{\theta}$, that takes an unlabeled document and predicts a class label.

From [Nigam et al., 2000]

# 20 Newsgroups

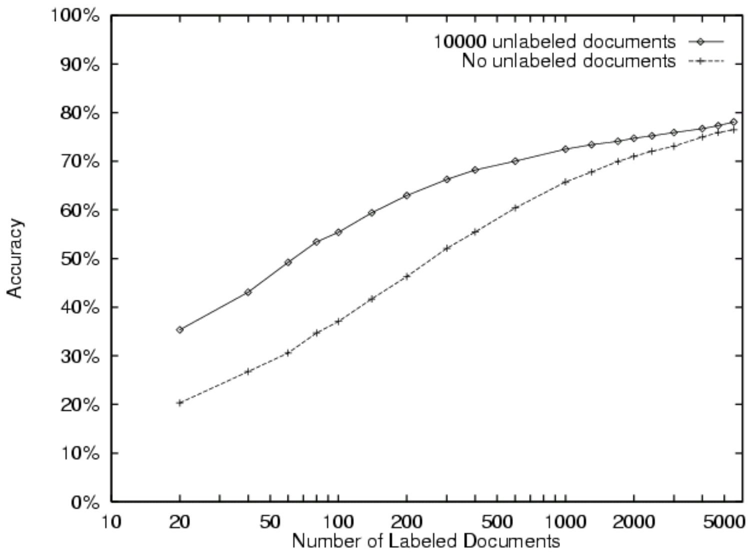


*Table 3.* Lists of the words most predictive of the course class in the WebKB data set, as they change over iterations of EM for a specific trial. By the second iteration of EM, many common course-related words appear. The symbol $D$ indicates an arbitrary digit.

| Iteration 0 | Iteration 1 | Iteration 2 |
|---|---|---|
| intelligence | $DD$ | $D$ |
| $DD$ | $D$ | $DD$ |
| artificial | lecture | lecture |
| understanding | cc | cc |
| $DD$w | $D$* | $DD$:$DD$ |
| dist | $DD$:$DD$ | due |
| identical | handout | $D$* |
| rus | due | homework |
| arrange | problem | assignment |
| games | set | handout |
| dartmouth | tay | set |
| natural | $DD$am | hw |
| cognitive | yurttas | exam |
| logic | homework | problem |
| proving | kfoury | $DD$am |
| prolog | sec | postscript |
| knowledge | postscript | solution |
| human | exam | quiz |
| representation | solution | chapter |
| field | assaf | ascii |

word w ranked by P(w|Y=course) / P(w|Y ≠ course)

Using one labeled example per class

Usupervised clustering

Just extreme case for EM with zero labeled examples…

## Clustering

- Given set of data points, group them
- Unsupervised learning
- Which patients are similar? (or which earthquakes, customers, faces, web pages, …)

## Mixture Distributions

Model joint $P(X_1 \ldots X_n)$ as mixture of multiple distributions.

Use discrete-valued random var Z to indicate which distribution is being use for each random draw

So *Mixture distribution* is of the form:

$$P(X_1 \ldots X_n) = \sum_i P(Z=i) \;\; P(X_1 \ldots X_n | Z)$$

Mixture of *Gaussians*:

- Assume each data point X=<X1, … Xn> is generated by one of several Gaussians, as follows:

1. randomly choose Gaussian i, according to P(Z=i)
2. randomly generate a data point <x1,x2 .. xn> according to $N(\mu_i, \Sigma_i)$

## EM for Mixture of Gaussian Clustering

Let's simplify to make this easier:

1. assume $X$=<$X_1$ ... $X_n$>, and the $X_i$ are conditionally independent given Z.

$$P(X|Z=j) = \prod_i N(X_i|\mu_{ji}, \sigma_{ji})$$

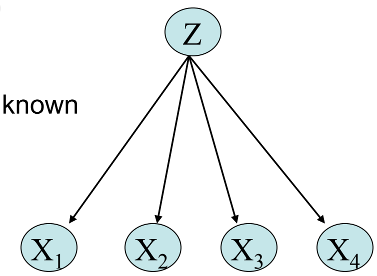
2. assume only 2 clusters (values of Z), and $\forall i, j, \sigma_{ji} = \sigma$

$$P(\mathbf{X}) = \sum_{j=1}^{2} P(Z=j|\pi) \prod_i N(x_i|\mu_{ji}, \sigma)$$

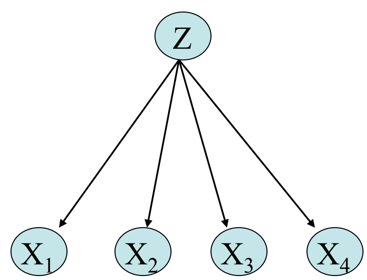
3. Assume σ known, $\pi_1 \ldots \pi_K, \mu_{1i} \ldots \mu_{Ki}$ unknown

Observed: $X$=<$X_1$ ... $X_n$>

Unobserved: $Z$




## EM




Given observed variables X, unobserved Z

Define $Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')]$

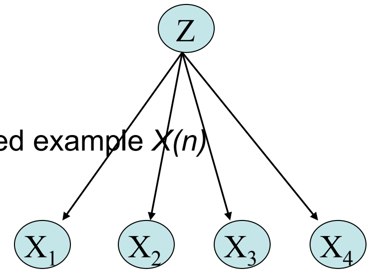
where $\theta = \langle \pi, \mu_{ji} \rangle$

Iterate until convergence:

- E Step: Calculate $P(Z(n)|X(n),\theta)$ for each example $X(n)$.
- M Step: Replace current $\theta$ by

$$\theta \leftarrow \arg\max_{\theta'} Q(\theta'|\theta)$$

## EM – E Step




Calculate *P(Z(n)|X(n),θ)* for each observed example *X(n)*

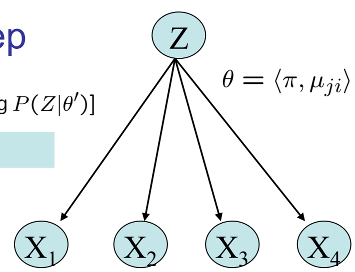
$X(n)=<x_1(n), x_2(n), \ldots x_T(n)>$.

$$P(z(n)=k|x(n),\theta) = \frac{P(x(n)|z(n)=k,\theta) \;\; P(z(n)=k|\theta)}{\sum_{j=0}^{1} p(x(n)|z(n)=j,\theta) \;\; P(z(n)=j|\theta)}$$

$$P(z(n)=k|x(n),\theta) = \frac{[\prod_i P(x_i(n)|z(n)=k,\theta)] \;\; P(z(n)=k|\theta)}{\sum_{j=0}^{1} \prod_i P(x_i(n)|z(n)=j,\theta) \;\; P(z(n)=j|\theta)}$$

$$P(z(n)=k|x(n),\theta) = \frac{[\prod_i N(x_i(n)|\mu_{k,i},\sigma)] \;\; (\pi^k(1-\pi)^{(1-k)})}{\sum_{j=0}^{1}[\prod_i N(x_i(n)|\mu_{j,i},\sigma)] \;\; (\pi^j(1-\pi)^{(1-j)})}$$

## EM – M Step




$\theta = \langle \pi, \mu_{ji} \rangle$

First consider update for π

$$Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X,Z|\theta')] = E[\log P(X|Z,\theta') + \log P(Z|\theta')]$$

π' has no influence

$$\pi \leftarrow \arg\max_{\pi'} E_{Z|X,\theta}[\log P(Z|\pi')]$$

Count z(n)=1

$$E_{Z|X,\theta}\left[\log P(Z|\pi')\right] = E_{Z|X,\theta}\left[\log\left(\pi'^{\sum_n z(n)}(1-\pi')^{\sum_n(1-z(n))}\right)\right]$$

$$= E_{Z|X,\theta}\left[\left(\sum_n z(n)\right)\log\pi' + \left(\sum_n (1-z(n))\right)\log(1-\pi')\right]$$

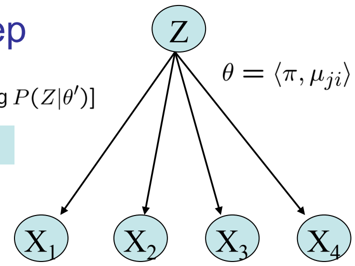
$$= \left(\sum_n E_{Z|X,\theta}[z(n)]\right)\log\pi' + \left(\sum_n E_{Z|X,\theta}[(1-z(n))]\right)\log(1-\pi')$$

$$\frac{\partial E_{Z|X,\theta}[\log P(Z|\pi')]}{\partial \pi'} = \left(\sum_n E_{Z|X,\theta}[z(n)]\right)\frac{1}{\pi'} + \left(\sum_n E_{Z|X,\theta}[(1-z(n))]\right)\frac{(-1)}{1-\pi'}$$

$$\pi \leftarrow \frac{\sum_{n=1}^{N} E[z(n)]}{\left(\sum_{n=1}^{N} E[z(n)]\right) + \left(\sum_{n=1}^{N}(1-E[z(n)])\right)} = \frac{1}{N}\sum_{n=1}^{N} E[z(n)]$$

## EM – M Step

Now consider update for $\mu_{ji}$




$\theta = \langle \pi, \mu_{ji} \rangle$

$$Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')] = E[\log P(X|Z,\theta') + \log P(Z|\theta')]$$

$\mu_{ji}$' has no influence

$$\mu_{ji} \leftarrow \arg\max_{\mu'_{ji}} E_{Z|X,\theta}[\log P(X|Z,\theta')]$$

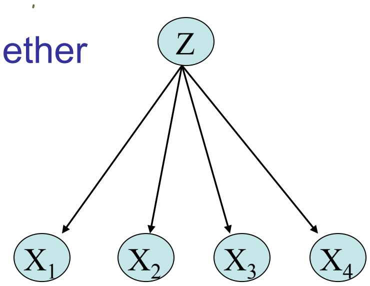
…
…
…

$$\mu_{ji} \leftarrow \frac{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta) \quad x_i(n)}{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta)}$$

Compare above to MLE if Z were observable:

$$\mu_{ji} \leftarrow \frac{\sum_{n=1}^{N} \delta(z(n) = j) \quad x_i(n)}{\sum_{n=1}^{N} \delta(z(n) = j)}$$

## EM – putting it together




Given observed variables X, unobserved Z

Define $Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')]$

where $\theta = \langle \pi, \mu_{ji} \rangle$

Iterate until convergence:

- E Step: For each observed example X(n), calculate $P(Z(n)|X(n), \theta)$

$$P(z(n) = k \mid x(n), \theta) = \frac{[\prod_i N(x_i(n)|\mu_{k,i}, \sigma)] \quad (\pi^k (1-\pi)^{(1-k)})}{\sum_{j=0}^{1} [\prod_i N(x_i(n)|\mu_{j,i}, \sigma)] \quad (\pi^j (1-\pi)^{(1-j)})}$$

- M Step: Update $\theta \leftarrow \arg\max_{\theta'} Q(\theta'|\theta)$

$$\pi \leftarrow \frac{1}{N} \sum_{n=1}^{N} E[z(n)] \qquad \mu_{ji} \leftarrow \frac{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta) \quad x_i(n)}{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta)}$$

## Mixture of Gaussians applet

Go to: http://www.socr.ucla.edu/htmls/SOCR_Charts.html
then go to Go to “Line Charts” → SOCR EM Mixture Chart

- try it with 2 Gaussian mixture components (“kernels”)
- try it with 4

## What you should know about EM

- For learning from partly unobserved data
- MLE of θ = $\arg\max_\theta \log P(data|\theta)$
- EM estimate: θ = $\arg\max_\theta E_{Z|X,\theta}[\log P(X,Z|\theta)]$
  Where X is observed part of data, Z is unobserved

- EM for training Bayes networks
- Can also develop MAP version of EM
- Can also derive your own EM algorithm for your own problem
  - write out expression for $E_{Z|X,\theta}[\log P(X,Z|\theta)]$
  - E step: for each training example $X^k$, calculate $P(Z^k | X^k, \theta)$
  - M step: chose new θ to maximize $E_{Z|X,\theta}[\log P(X,Z|\theta)]$

## K-Means Clustering
## (cheap approximation to mixture of Gaussians)

**Algorithm**

Input – Desired number of clusters, *k*

Initialize – the *k* cluster centers (randomly if necessary)

Iterate –

1. Assign the objects to the nearest cluster centers
2. Re-estimate the *k* cluster centers (aka the centroid or mean) based on current assignment

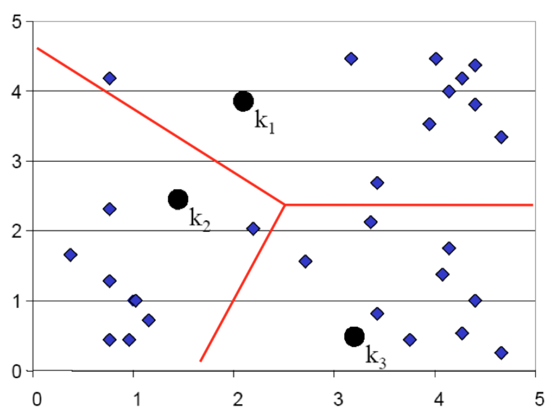
$$\vec{\mu}_k = \frac{1}{\mathcal{C}_k} \sum_{i \in \mathcal{C}_k} \vec{x}_i$$

Termination –

If none of the assignments changed in the last iteration, exit. Otherwise go to 1.

[slide from Aarti Singh]

## K-means Clustering: Step 1

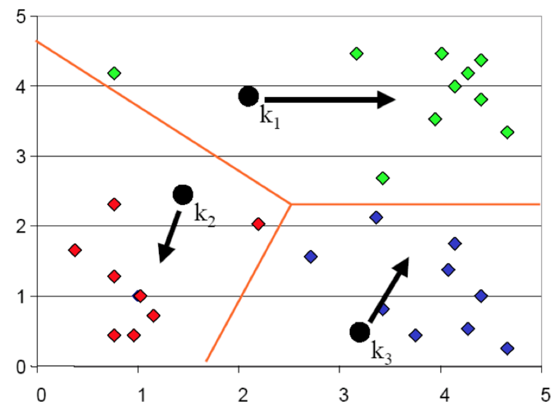



Voronoi diagram

30

[slide from Aarti Singh]

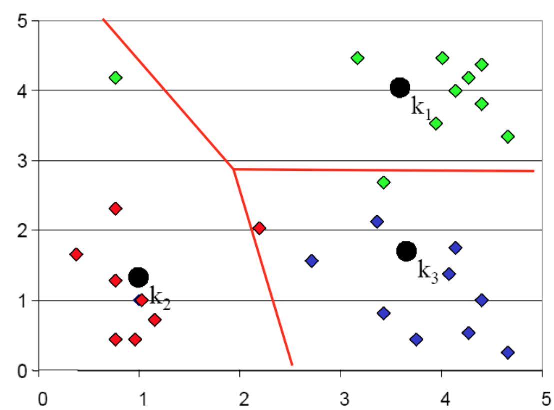
## K-means Clustering: Step 2




31

[slide from Aarti Singh]

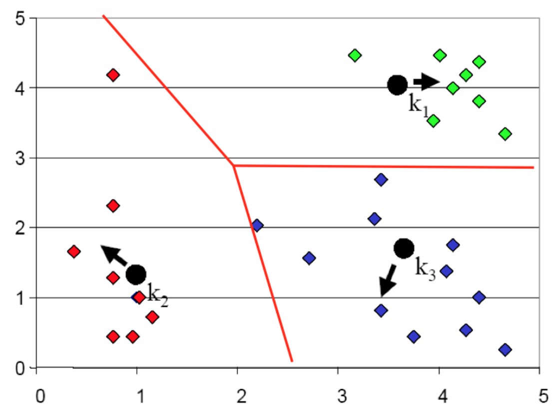
## K-means Clustering: Step 3




32

[slide from Aarti Singh]

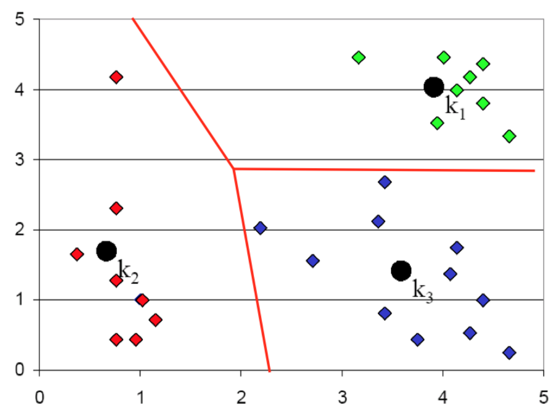
# K-means Clustering: Step 4




33

[slide from Aarti Singh]

# K-means Clustering: Step 5




34

[slide from Aarti Singh]

# EM & Mixture of Gaussians
vs.
K-Means

- Same intuition: iteratively re-estimate
  - assignments of points to clusters
  - definitions of clusters
- Difference:
  - K-Means uses “hard assignments” of points to clusters
  - Mixture-of-Gaussians uses probabilistic assignments
- Similar local optimum problems