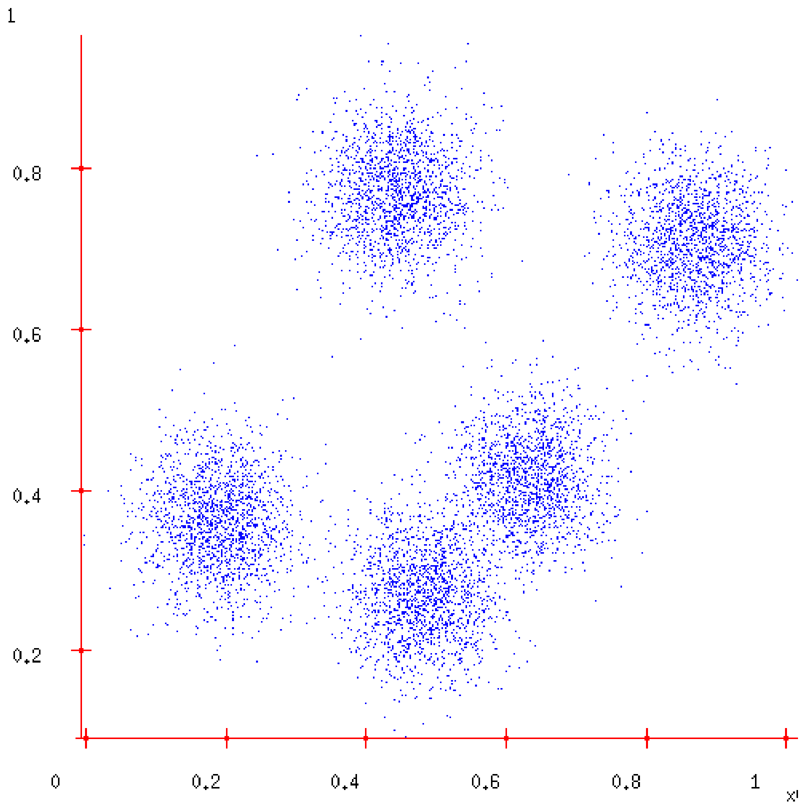
Outline

- K-Means
- Hierarchical Clustering
- Model-based Clustering (GMM and Expectation Maximization)
- Evaluation of Clustering Algorithms

Model-based Clustering: Gaussian Mixture Model

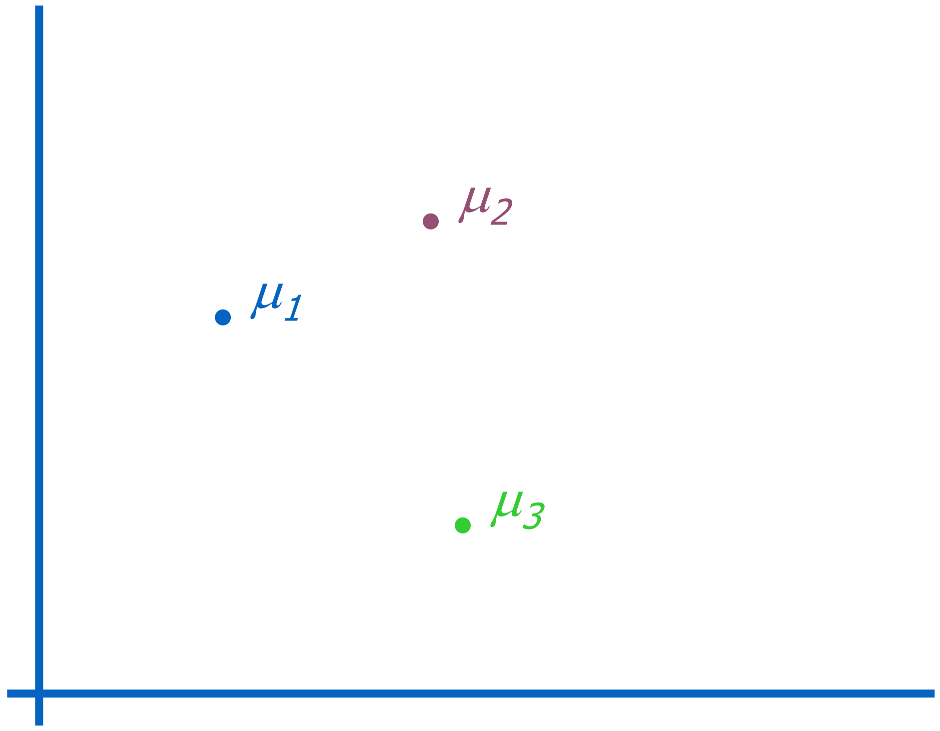
- Density estimation with multimodal/clumpy data



Slide Courtesy: Andrew Moore, CMU

Gaussian Mixture Model (GMM)

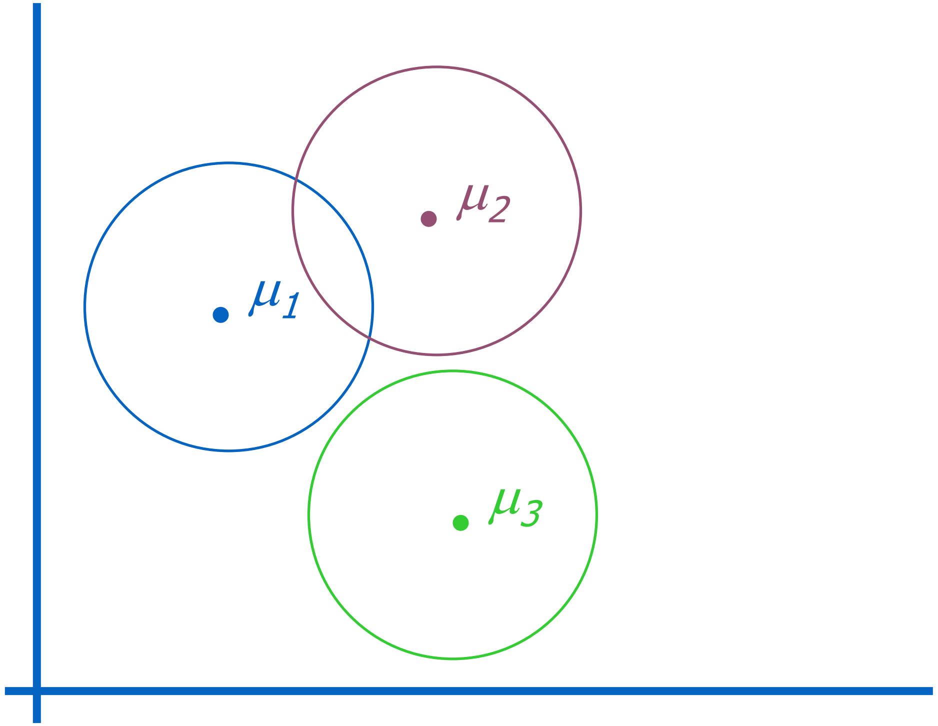
- The GMM assumption
- There are k components. The i^{th} component is called ω_i
- Component ω_i has an associated mean vector μ_i



Slide Courtesy: Andrew Moore, CMU

Gaussian Mixture Model (GMM)

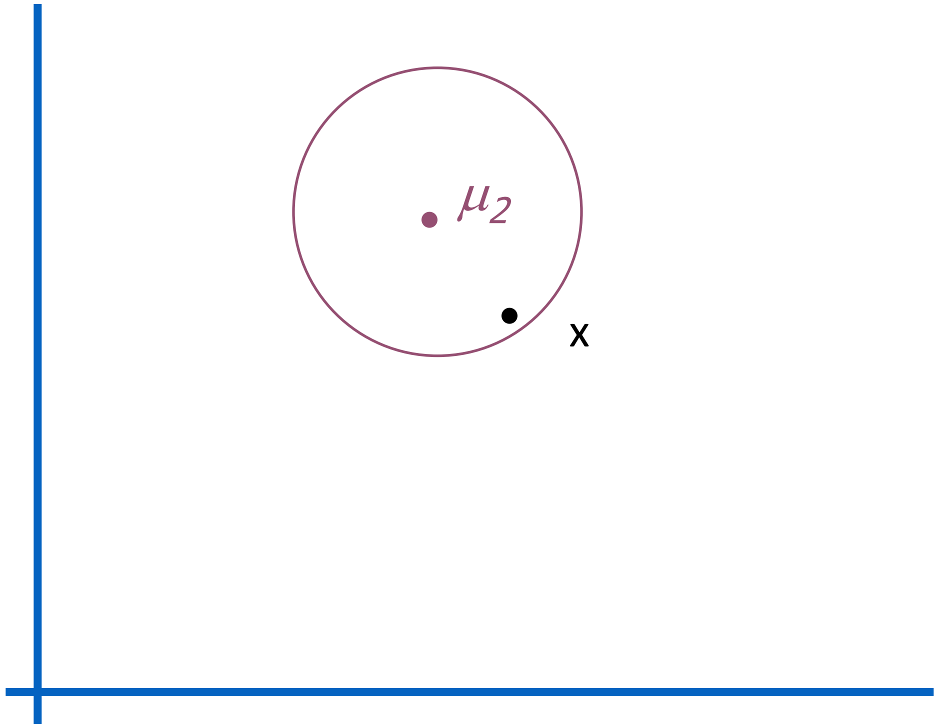
- The GMM assumption
- There are k components. The i^{th} component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2 \boldsymbol{I}$



Slide Courtesy: Andrew Moore, CMU

Gaussian Mixture Model (GMM)

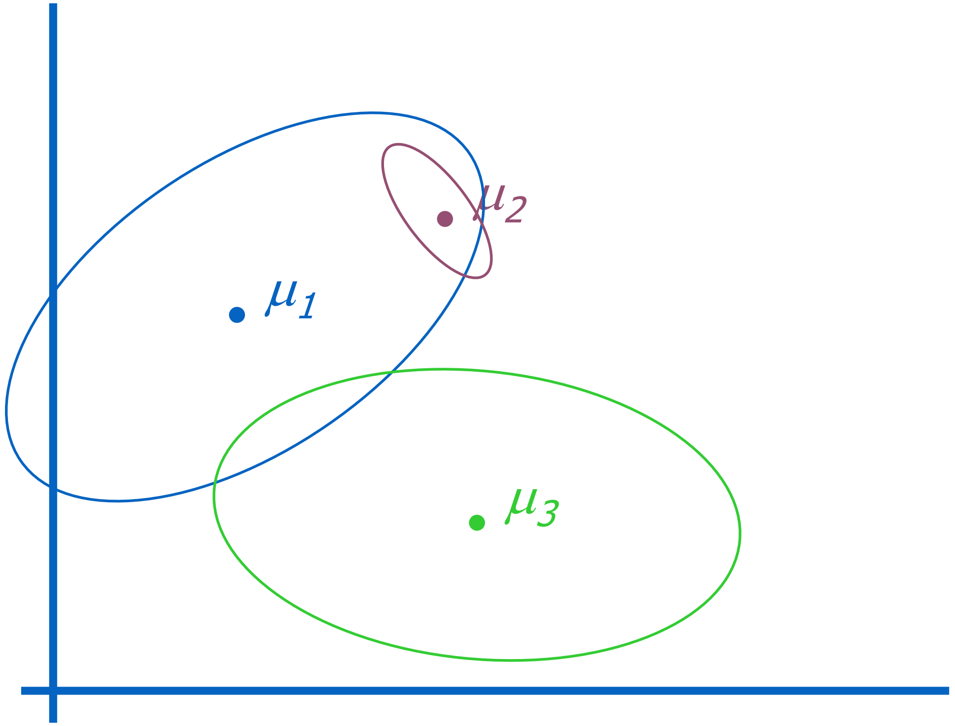
- The GMM assumption
- There are k components. The i^{th} component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2 \boldsymbol{I}$
- Assume that each datapoint is generated according to the following recipe:
    - Pick a component at random. Choose component i with probability $P(\omega_i)$.
    - Datapoint ~ $N(\mu_i, \Sigma_i)$



Slide Courtesy: Andrew Moore, CMU

Gaussian Mixture Model (GMM)

- The GMM assumption
- There are k components. The ith component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2 \boldsymbol{I}$
- Assume that each datapoint is generated according to the following recipe:
  - Pick a component at random. Choose component i with probability $P(\omega_i)$.
  - Datapoint ~ N(μ_i, Σ_i)



Slide Courtesy: Andrew Moore, CMU

Gaussian Mixture Model (GMM)

- Given the means and σ^2, we can compute P(data | $\boldsymbol{\mu}_1,\boldsymbol{\mu}_2..\boldsymbol{\mu}_k, \sigma^2$). How do we find the $\boldsymbol{\mu}_i$ s and σ^2 which give max likelihood?
- The normal max likelihood trick:

$$\text{Set } \frac{d}{d\mu_i} \log \text{Prob } (\ldots.) = 0$$

  and solve for μ_i's.
- Use gradient descent
  - Slow but doable
- Use a much faster and popular method: EM

Slide Courtesy: Andrew Moore, CMU

Expectation Maximization (EM)

- We'll get back to unsupervised learning/clustering/GMM soon.
- The EM algorithm was explained and given its name in a classic 1977 paper by Arthur Dempster, Nan Laird, and Donald Rubin.
- They pointed out that the method had been "proposed many times in special circumstances" by earlier authors.
- EM is typically used to compute maximum likelihood estimates given incomplete samples.
  - An excellent way of doing our unsupervised learning problem, as we'll see
  - Many, many other uses, including inference of Hidden Markov Models
- The EM algorithm estimates the parameters of a model iteratively. Starting from some initial guess, each iteration consists of
  - an E step (Expectation step)
  - an M step (Maximization step)

Slide Courtesy: Andrew Moore, CMU

EM: Trivial Example

Let events be "grades in a class"

w_1 = Gets an A $\quad P(A) = \frac{1}{2}$

w_2 = Gets a B $\quad P(B) = \mu$

w_3 = Gets a C $\quad P(C) = 2\mu$

w_4 = Gets a D $\quad P(D) = \frac{1}{2}-3\mu$

(Note $0 \leq \mu \leq 1/6$)

Assume we want to estimate μ from data. In a given class, there were

a A's
b B's
c C's
d D's

What's the maximum likelihood estimate of μ given a,b,c,d ?

Slide Courtesy: Andrew Moore, CMU

EM: Trivial Example

$P(A) = \frac{1}{2} \quad P(B) = \mu \quad P(C) = 2\mu \quad P(D) = \frac{1}{2}-3\mu$

$P(a,b,c,d \mid \mu) = (\frac{1}{2})^a (\mu)^b (2\mu)^c (\frac{1}{2}-3\mu)^d$

$\log P(a,b,c,d \mid \mu) = a\log \frac{1}{2} + b\log \mu + c\log 2\mu + d\log (\frac{1}{2}-3\mu)$

$$\text{FOR MAX LIKE } \mu, \text{ SET } \frac{\partial \text{LogP}}{\partial \mu} = 0$$

$$\frac{\partial \text{LogP}}{\partial \mu} = \frac{b}{\mu} + \frac{2c}{2\mu} - \frac{3d}{1/2 - 3\mu} = 0$$

$$\text{Gives max like } \mu = \frac{b+c}{6(b+c+d)}$$

So if class got

A	B	C	D
14	6	9	10

$$\text{Max likelihood estimate} : \mu = \frac{1}{10}$$

Slide Courtesy: Andrew Moore, CMU

EM: Same Example with Hidden Info

Someone tells us that

Number of High grades (A's + B's) = h

Number of C's = c

Number of D's = d

What is the max likelihood estimate of μ now?

REMEMBER

$P(A) = \frac{1}{2}$

$P(B) = \mu$

$P(C) = 2\mu$

$P(D) = \frac{1}{2} - 3\mu$

Slide Courtesy: Andrew Moore, CMU

EM: Same Example with Hidden Info

Someone tells us that

Number of High grades (A's + B's) = h

Number of C's = c

Number of D's = d

What is the max likelihood estimate of μ now?

We can answer this circularly as below

REMEMBER

P(A) = ½

P(B) = μ

P(C) = 2μ

P(D) = ½-3μ

EXPECTATION

If we know the value of μ we could compute the expected value of *a* and *b*

Since the ratio a:b should be the same as the ratio ½ : μ

$$a = \frac{\frac{1}{2}}{\frac{1}{2}+\mu} h \qquad b = \frac{\mu}{\frac{1}{2}+\mu} h$$

MAXIMIZATION

If we know the expected values of *a* and *b* we could compute the maximum likelihood value of μ

$$\mu = \frac{b+c}{6(b+c+d)}$$

Slide Courtesy: Andrew Moore, CMU

EM: Solution for Trivial Example

We begin with a guess for μ

We iterate between EXPECTATION and MAXIMIZATION to improve our estimates of μ and *a* and *b*.

Define μ(t) the estimate of μ on the t^{th} iteration

b(t) the estimate of *b* on t^{th} iteration

E-step

$$\mu(0) = \text{initial guess}$$

$$b(t) = \frac{\mu(t)h}{\tfrac{1}{2} + \mu(t)} = \mathrm{E}[b \mid \mu(t)]$$

M-step

$$\mu(t+1) = \frac{b(t)+c}{6(b(t)+c+d)}$$

$$= \text{max like est of } \mu \text{ given } b(t)$$

Continue iterating until converged.

Good news: Converging to local optimum is assured.

Bad news: "local" optimum.

EM: Convergence

- Convergence proof based on fact that Prob(data | μ) must increase or remain same between each iteration [NOT OBVIOUS]
- But it can never exceed 1 [OBVIOUS]

So it must therefore converge [OBVIOUS]

In our example, suppose we had

$h = 20$
$c = 10$
$d = 10$
$\mu(0) = 0$

t	μ(t)	b(t)
0	0	0
1	0.0833	2.857
2	0.0937	3.158
3	0.0947	3.185
4	0.0948	3.187
5	0.0948	3.187
6	0.0948	3.187

Convergence is generally <u>linear</u>: error decreases by a constant factor each time step.

Slide Courtesy: Andrew Moore, CMU

Back to GMM

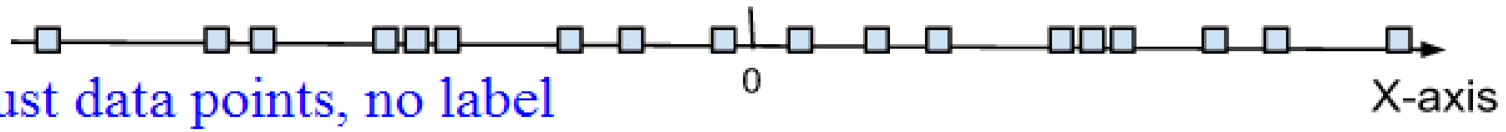
Given a training data set: X={x(1),x(2),…,x(n)}
Z={z(1),z(2),…,z(n)}
z(i) is the calss/group label of sample x(i).
As we are in Clustering setting,
X is Given and Z is unknown

Now, we model the data by specifying a joint distribution p(x(i), z(i))=p(x(i)|z(i))p(z(i))

$$z(i) \sim \text{Multinomial}(\phi)$$
$$\phi_j \geq 0,\ \sum_{j=1}^{k} \phi_j = 1$$
$$k = \#\ \text{of}\ z(i)\text{'s values}$$
$$\phi_j = p(z(i) = j)$$
$$x(i)|z(i) = j \sim \mathcal{N}(\mu_j, \Sigma_j)$$

⇨ each $x(i)$ was generated by randomly choosing $z(i)$ from $\{1, \ldots, k\}$, and then $x(i)$ was drawn from one of k Gaussians .

The parameters of our model are thus ϕ, μ and Σ.



Just data points, no label

Slide Courtesy: Andrew Moore, CMU

EM for GMM

Incomplete Data

X={x(1),x(2),…,x(n)} Given
Z={z(1),z(2),…,z(n)} unknown

unknown

The parameters of our model ϕ, μ, Σ

What is the value of z(i)?

We can answer this question circularly:

EXPECTATION

If we know the values of ϕ, μ, Σ we could compute the expected values of Z

MAXIMIZATION

If we know the expected values of Z we could compute the maximum likelihood value of ϕ, μ, Σ

We begin with a guess for ϕ, μ, Σ, and then iterate between EXPECTATION and MAXIMALIZATION to improve our estimates of ϕ, μ, Σ and Z

Continue iterating until converged.

Slide Courtesy: Andrew Moore, CMU

EM for GMM

$$\ell(\phi, \mu, \Sigma) = \sum_{i=1}^{m} \log p(x^{(i)}|z^{(i)}; \mu, \Sigma) + \log p(z^{(i)}; \phi)$$

Maximizing this with respect to ϕ, μ and Σ gives the parameters:

$$\begin{aligned}
\phi_j &= \frac{1}{m}\sum_{i=1}^{m} 1\{z^{(i)} = j\}, \\
\mu_j &= \frac{\sum_{i=1}^{m} 1\{z^{(i)} = j\}x^{(i)}}{\sum_{i=1}^{m} 1\{z^{(i)} = j\}}, \\
\Sigma_j &= \frac{\sum_{i=1}^{m} 1\{z^{(i)} = j\}(x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^{m} 1\{z^{(i)} = j\}}.
\end{aligned}$$

Slide Courtesy: Andrew Moore, CMU

EM for GMM

Repeat until convergence: {

(E-step) For each i, j, set

$$w_j^{(i)} := p(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma)$$

(M-step) Update the parameters:

$$\phi_j := \frac{1}{m} \sum_{i=1}^{m} w_j^{(i)},$$

$$\mu_j := \frac{\sum_{i=1}^{m} w_j^{(i)} x^{(i)}}{\sum_{i=1}^{m} w_j^{(i)}},$$

$$\Sigma_j := \frac{\sum_{i=1}^{m} w_j^{(i)} (x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^{m} w_j^{(i)}}$$

}

Slide Courtesy: Andrew Moore, CMU

GMM vs k-Means

Given a training data set: X={x(1),x(2),...,x(n)}
Z={z(1),z(2),...,z(n)}
z(i) is the calss/group label of sample x(i).
As we are in Clustering setting,
X is Given and Z is unknown

Model of EM

EM model the data by specifying a joint distribution p(x(i), z(i))=p(x(i)|z(i))p(z(i))

$z(i) \sim \text{Multinomial}(\phi)$
$\phi_j \geq 0, \sum_{j=1}^{k} \phi_j = 1$
$k = \#$ of $z(i)$'s values
$\phi_j = p(z(i) = j)$
$x(i)|z(i) = j \sim \mathcal{N}(\mu_j, \Sigma_j)$

$\Rightarrow$ each $x(i)$ was generated by randomly choosing $z(i)$ from $\{1, \ldots, k\}$, and then $x(i)$ was drawn from one of k Gaussians .

K-means is a simplified EM, it assumes that

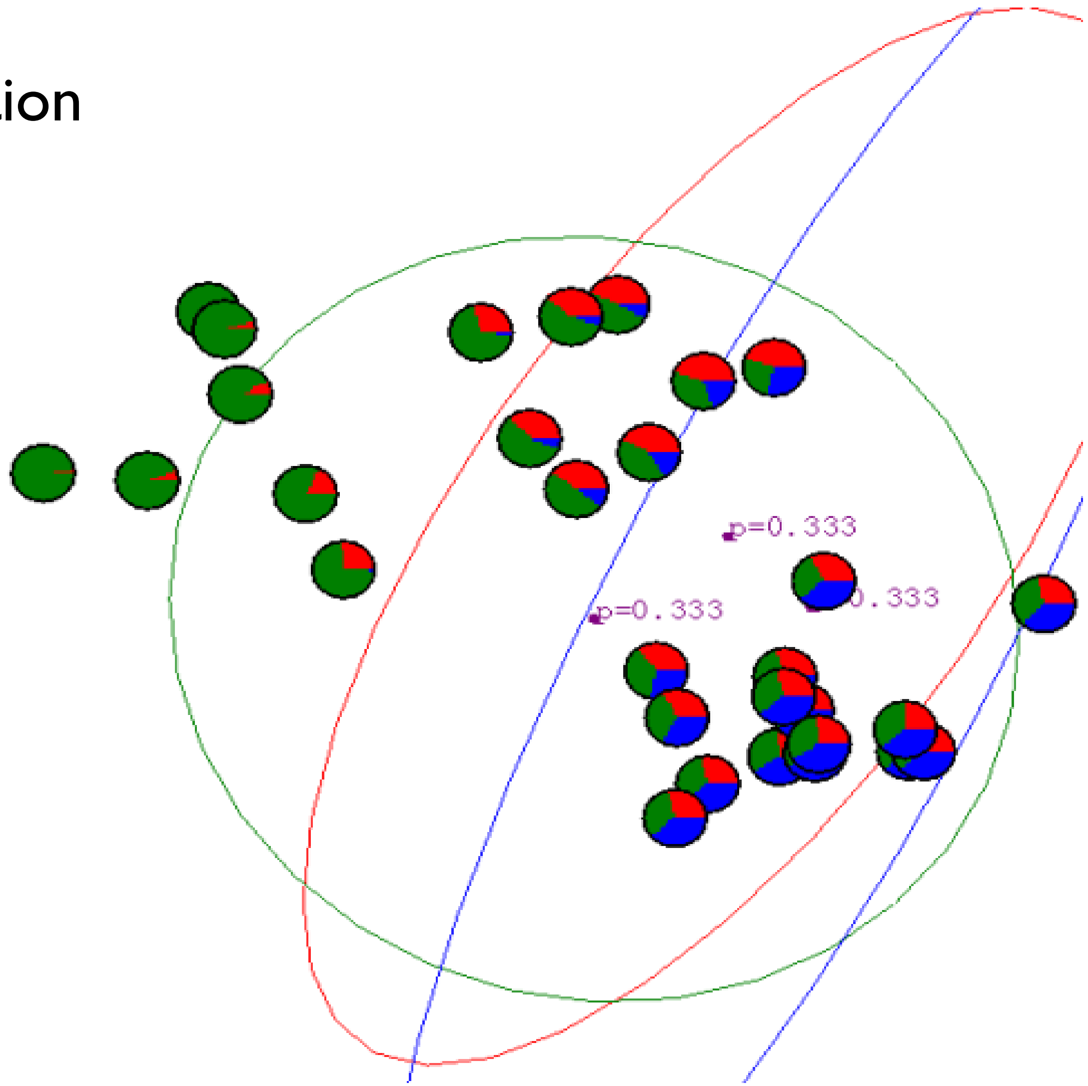
$\phi_j = \phi_i = 1/k$, and $\Sigma_j = \Sigma_i$ for i, j=1,2,...k
k is given by user

$\Rightarrow$ $\mu_1, \mu_2 \cdots \mu_k$ are the only unknown parameters of the model (the means of clusters)

Slide Courtesy: Andrew Moore, CMU

GMM: Example

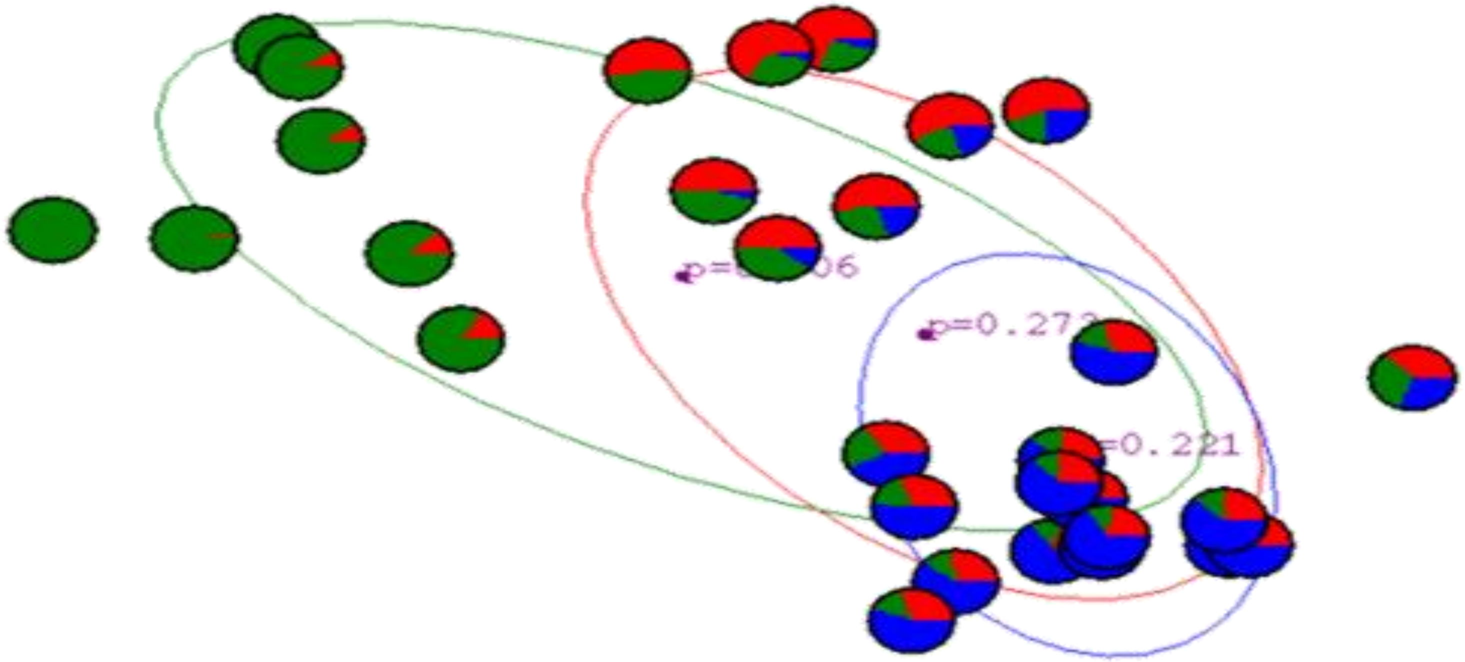
Start: 0th iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

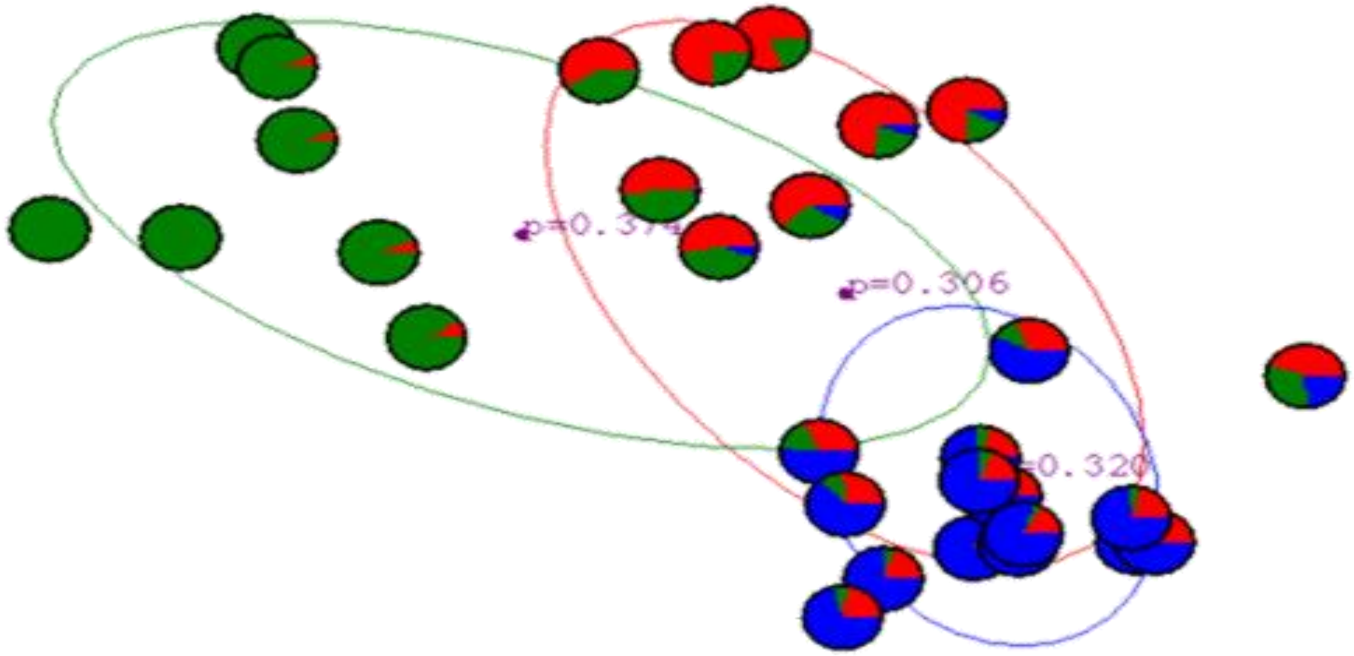
After 1st iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

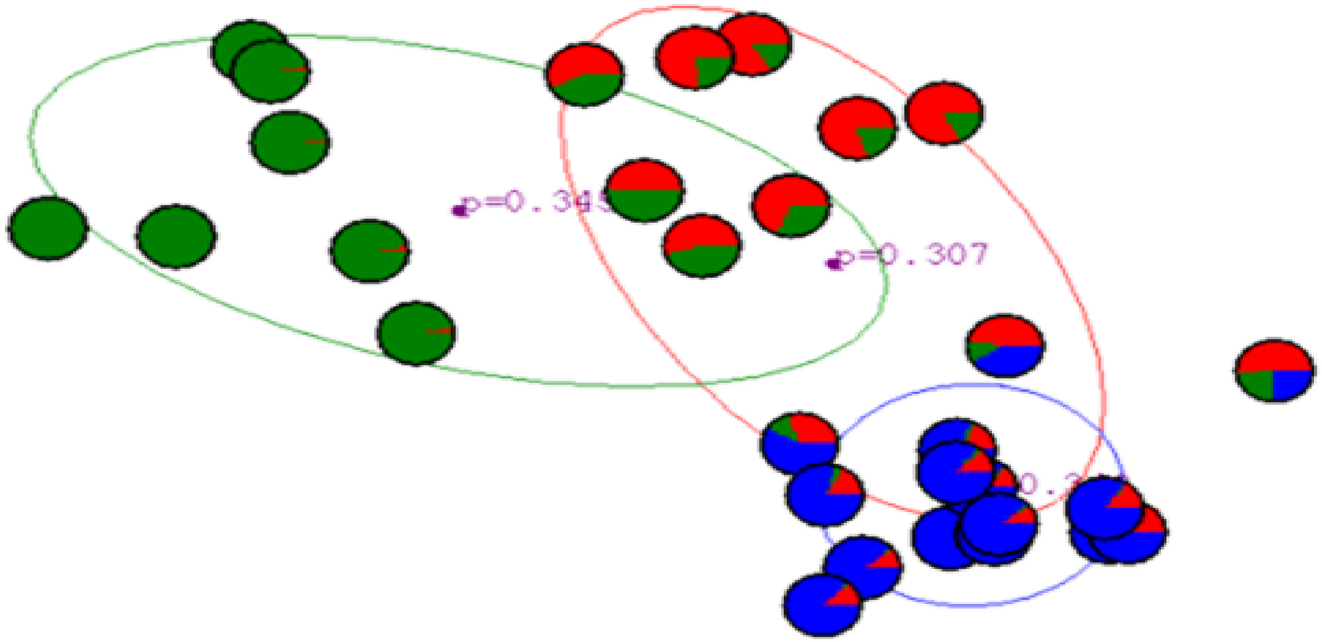
After 2nd iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

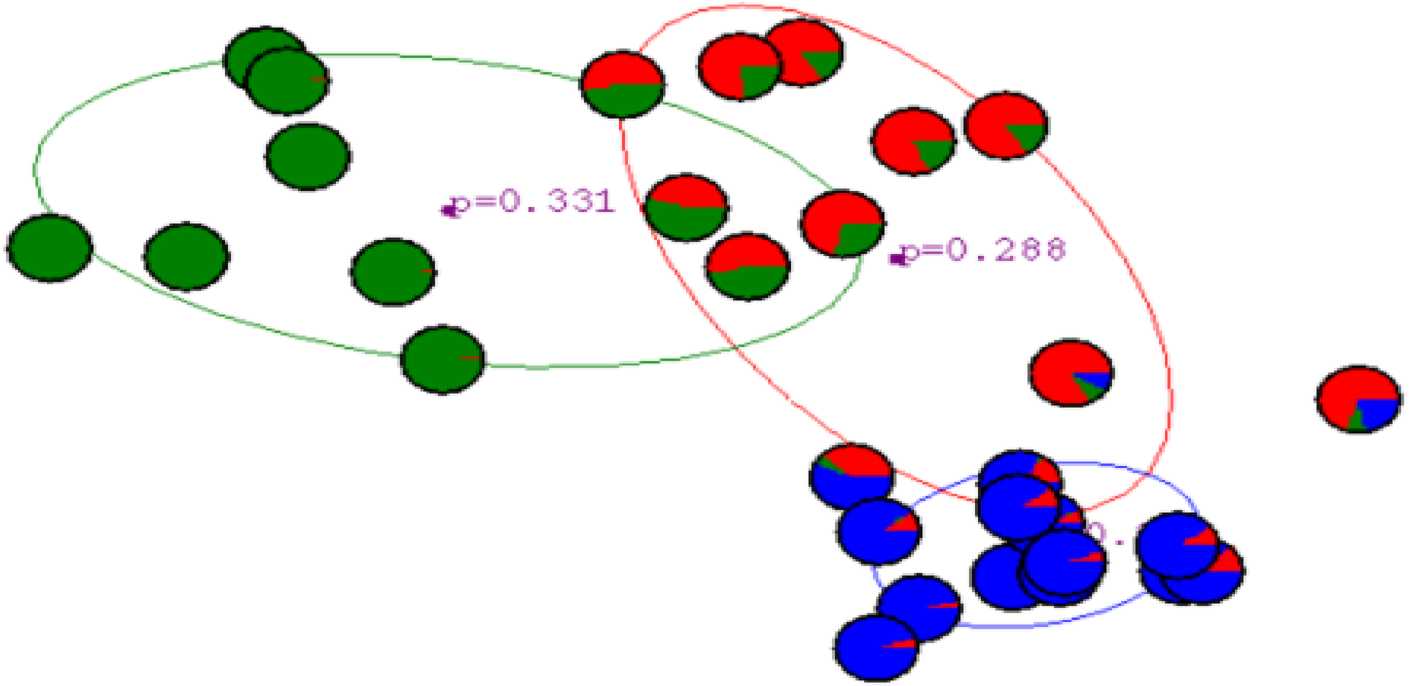
After 3rd iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

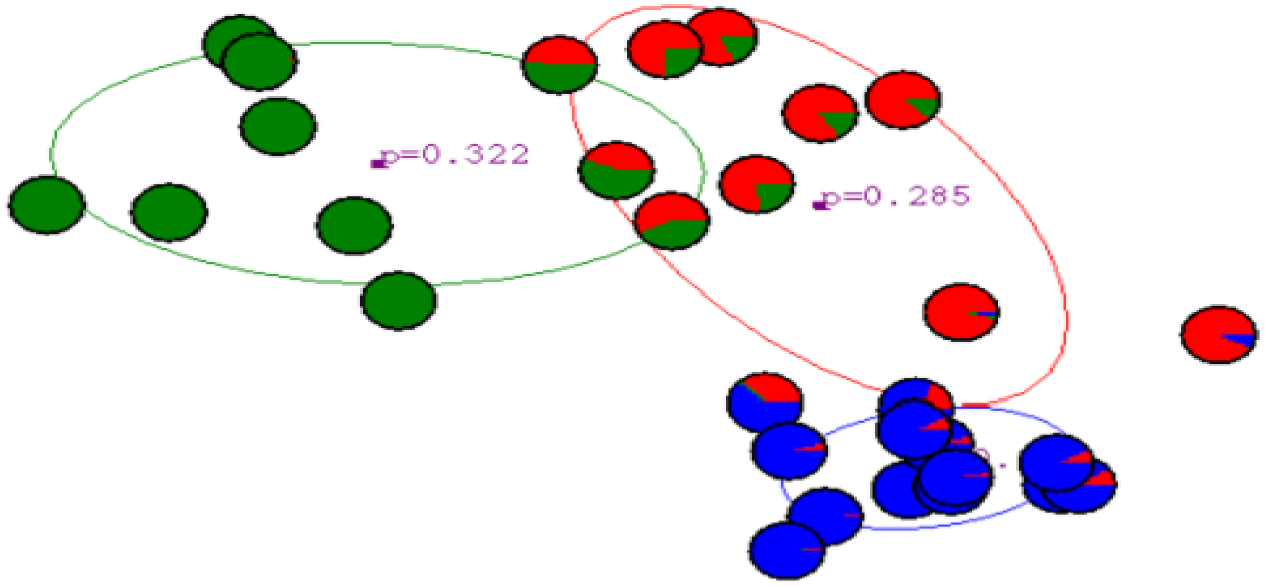
After 4th iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

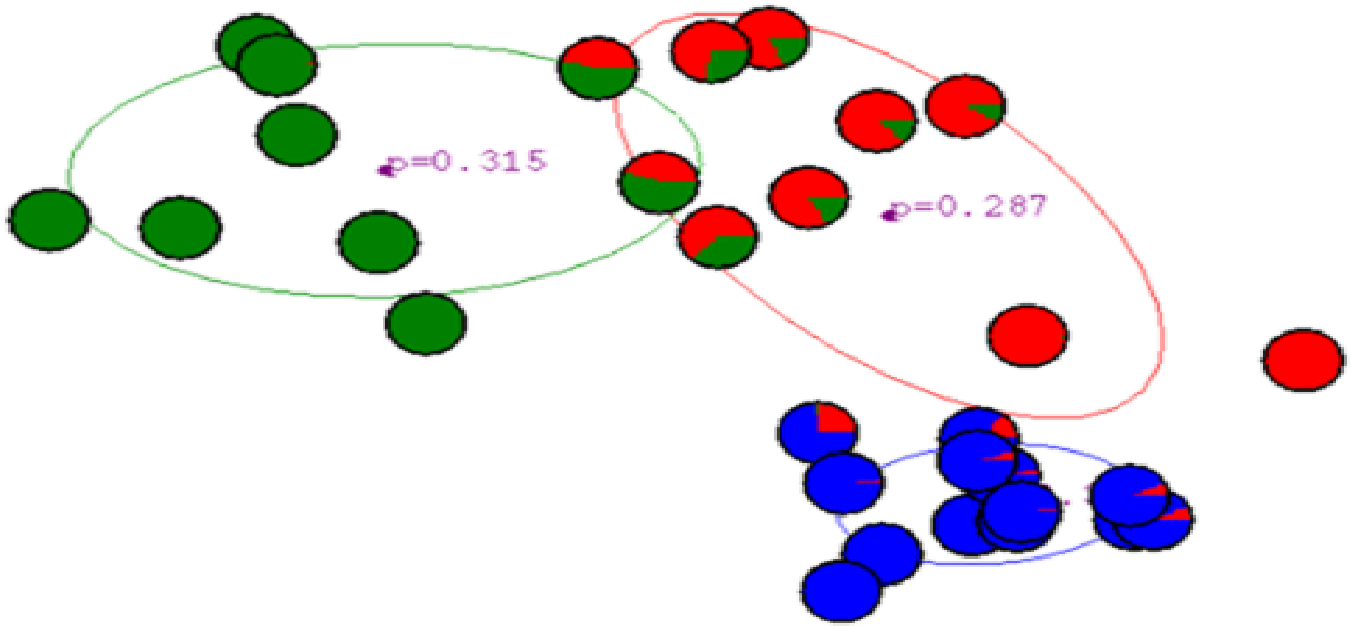
After 5th iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

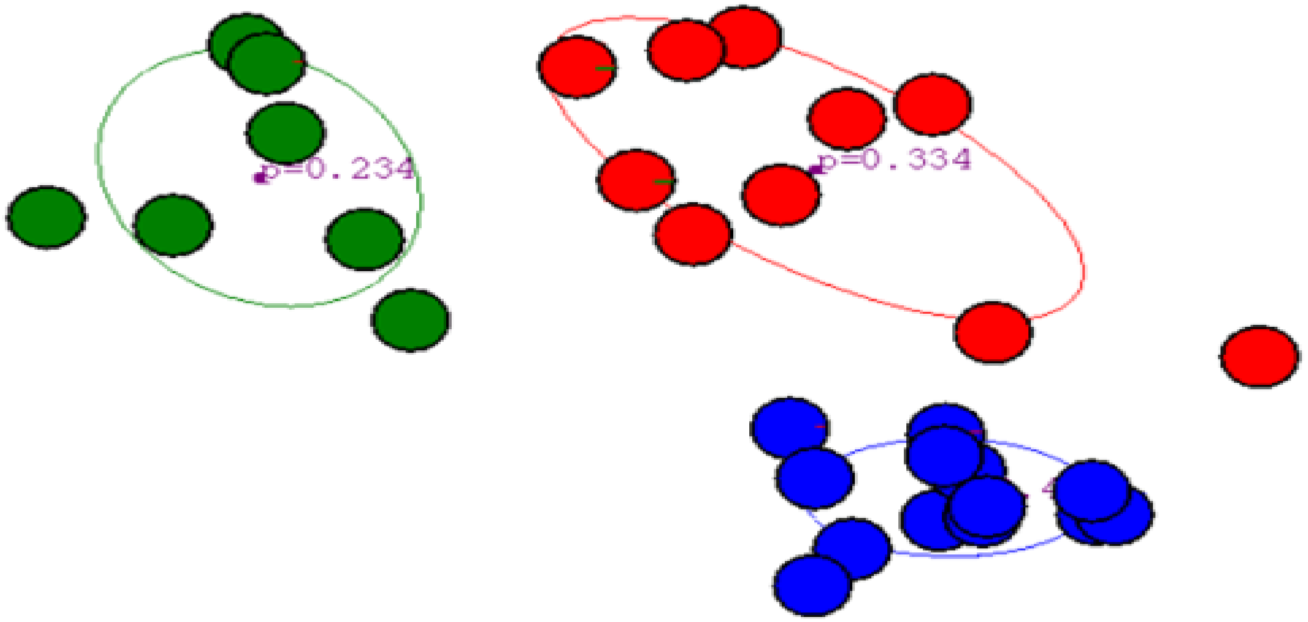
After 6th iteration



Slide Courtesy: Andrew Moore, CMU

GMM: Example

After 20th iteration



Slide Courtesy: Andrew Moore, CMU

More on EM Algorithm

- What are the EM algorithm initialization methods?
  - Random guess.
  - Initialized by k-means. After a few iterations of k-means, using the parameters to initialize EM
- What are the main advantages of parametric methods?
  - You can easily change the model to adapt to different distribution of data sets.
  - Knowledge representation is very compact. Once the model is selected, the model is represented by a specific number of parameters.
  - The number of parameters does not increase with the increasing of training data .