So far, supervised learning

$h: X \to \mathbb{R}$ "regression"

$h: X \to \{0,1,\ldots,K\}$ "classification"

Unsupervised

Clustering
K-means

Machine Learning – CSE546
Emily Fox
University of Washington

November 4, 2013

Clustering images

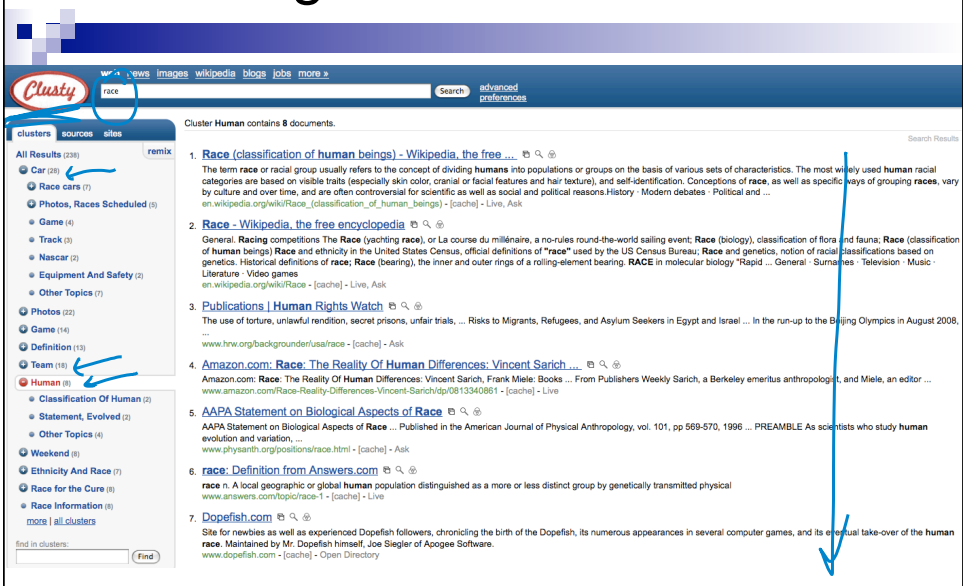
key: no labels given

Set of Images

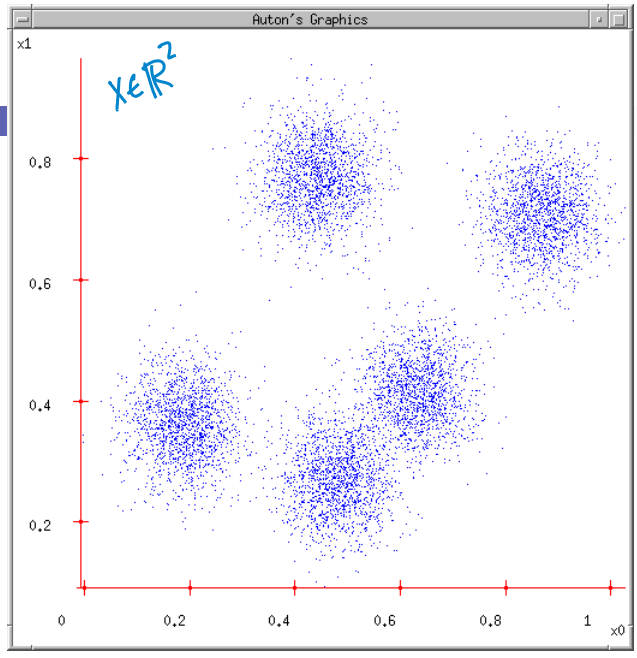
organize into coherent "themes"

[Goldberger et al.]

Clustering web search results

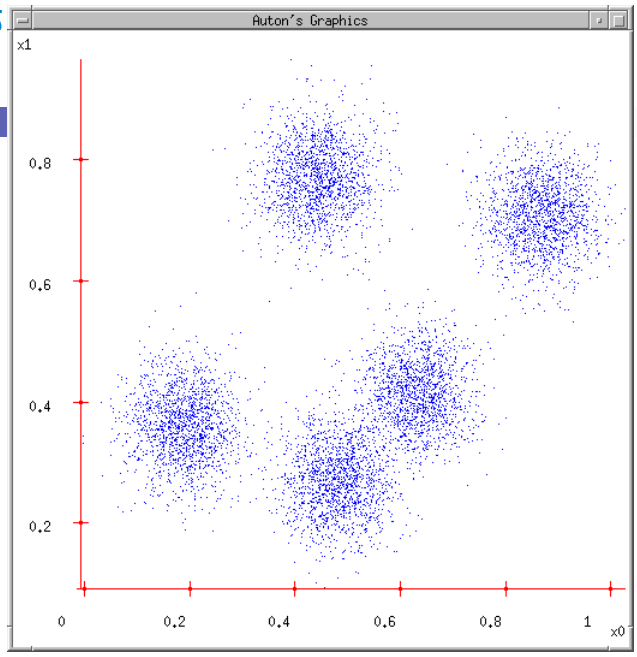


Some Data



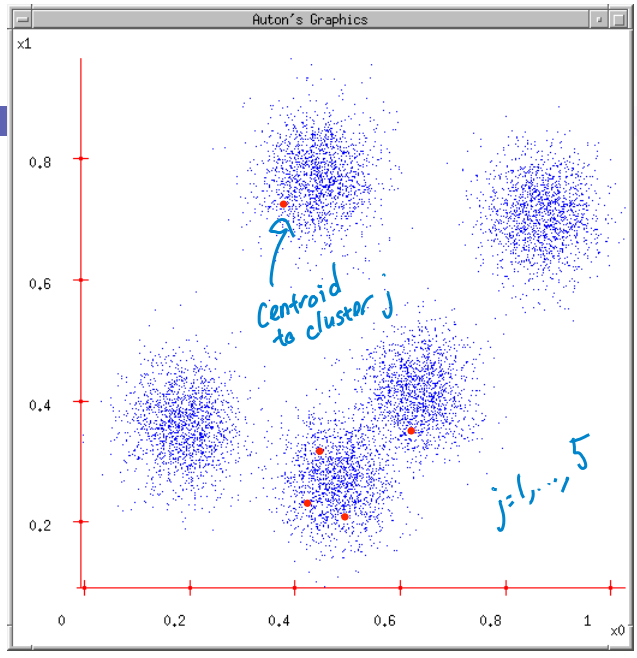
K-means

1. Ask user how many clusters they'd like. *(e.g. k=5)*



K-means

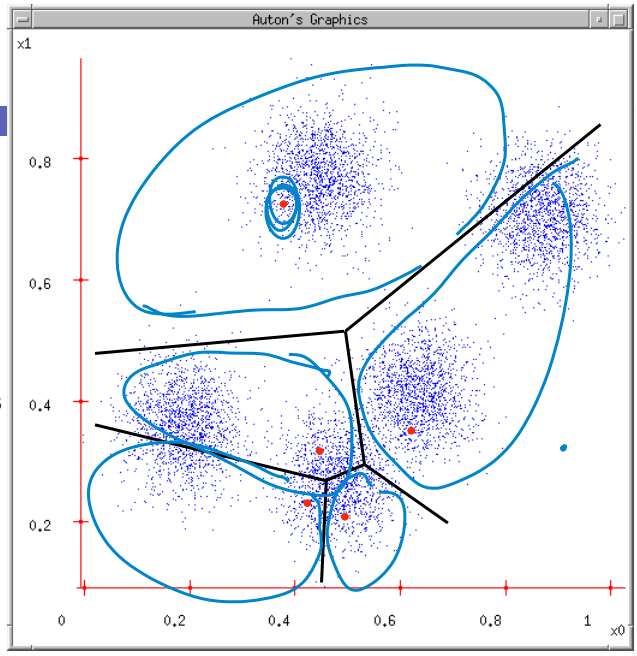
1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations



K-means

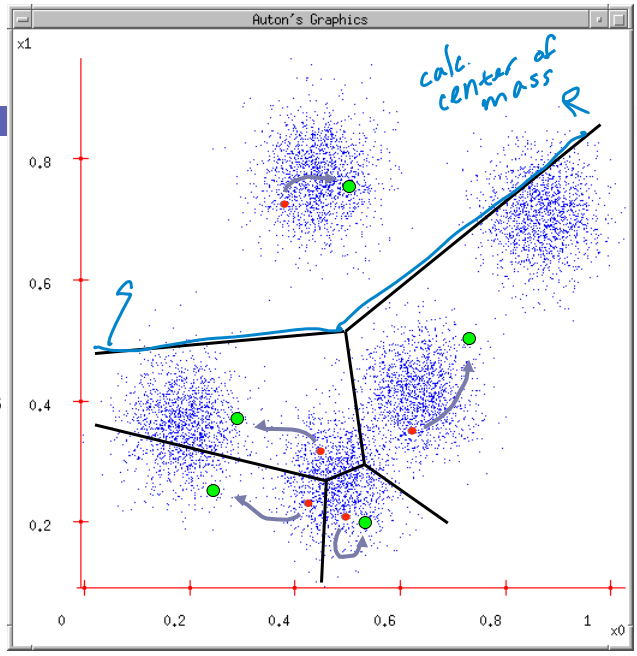
1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to. (Thus each Center "owns" a set of datapoints)

Voronoi tessellation



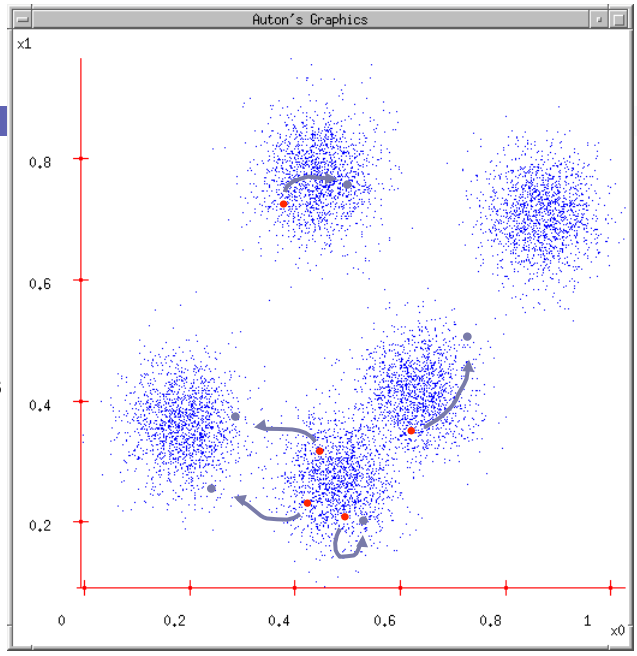
K-means

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns



K-means

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!



K-means

- Randomly initialize *k* centers (or "smart(y)")
  - $\mu^{(0)} = \mu_1^{(0)}, \ldots, \mu_k^{(0)}$ ← iteration

converged when nothing moves (no point changes its cluster)

- **Classify**: Assign each point j∈{1,…N} to nearest center: = fix μ, opt. C
  - $C^{(t)}(j) \leftarrow \arg\min_i ||\mu_i - x_j||^2$ (i: ith cluster center, j: jth data pt)

C(j)=k ⇒ jth obs. is assoc. w/ cluster k

iterate:

- **Recenter**: μ_i becomes centroid of its point: fix C, opt. μ
  - $\mu_i^{(t+1)} \leftarrow \arg\min_\mu \sum_{j:C(j)=i} ||\mu - x_j||^2$ ← $\mu_i = \frac{\sum_{j:C(j)=i} x_j}{|\{j: C(j)=i\}|}$
  - Equivalent to μ_i ← average of its points!

What is K-means optimizing?

- Potential function F(μ,C) of centers μ and point allocations C:
  - $F(\mu, C) = \sum_{j=1}^{N} ||\mu_{C(j)} - x_j||^2$

k centers, allocations

- Optimal K-means:
  - $\min_\mu \min_C F(\mu, C)$

optimization is hard, but iteratively easy

coordinate descent!

Does K-means converge??? Part 1

$f(a,b) = g(a) + h(b) \Rightarrow \min_a \min_b f(a,b) = \min_a g(a) + \min_b h(b)$

- Optimize potential function:

$$\min_\mu \min_C F(\mu, C) = \min_\mu \min_C \sum_{i=1}^{k} \sum_{j:C(j)=i} ||\mu_i - x_j||^2$$

- Fix μ, optimize C

$$\min_{C: C(1),\ldots,C(N)} \sum_{j=1}^{N} ||\mu_{C(j)} - x_j||^2 = \min_{C(1)} \min_{C(2)} \cdots \min_{C(N)} \sum_{j=1}^{N} ||\mu_{C(j)} - x_j||^2$$

$$= \sum_{j=1}^{N} \min_{C(j)} ||\mu_{C(j)} - x_j||^2$$

ind. minimizations

exactly the "classification step"

Does K-means converge??? Part 2

- Optimize potential function:

$$\min_{\mu} \min_{C} F(\mu, C) = \min_{\mu} \min_{C} \sum_{i=1}^{k} \sum_{j:C(j)=i} ||\mu_i - x_j||^2$$

- Fix C, optimize μ

$$\min_{\mu: \mu_1, \ldots, \mu_k} \sum_{j=1}^{N} \|\mu_{C(j)} - x_j\|^2 = \min_{\mu: \mu_1, \ldots, \mu_k} \sum_{i=1}^{k} \sum_{j:C(j)=i} \|\mu_i - x_j\|^2$$

$$= \sum_{i=1}^{k} \min_{\mu_i} \sum_{j:C(j)=i} \|\mu_i - x_j\|^2$$

"recenter step"

μ_i = center of pts = avg = $\dfrac{\sum_{j:C(j)=i} x_j}{|\{j: C(j)=i\}|}$

Coordinate descent algorithms

$$\min_{\mu} \min_{C} F(\mu, C) = \min_{\mu} \min_{C} \sum_{i=1}^{k} \sum_{j:C(j)=i} ||\mu_i - x_j||^2 \geq 0$$

- Want: $\min_a \min_b F(a,b)$
- Coordinate descent:
  - fix a, minimize b
  - fix b, minimize a
  - repeat
- Converges!!!
  - if F is bounded
  - to a (often good) local optimum
    - as we saw in applet (play with it!)
      - (For LASSO it converged to the global optimum, because of convexity)

$(a,b) \bullet F(a,b)$

$\downarrow \min_a$

$(a',b) \bullet F(a',b) \leq F(a,b)$

$\downarrow \min_b$

$(a',b') \bullet F(a',b') \leq F(a',b) \leq F(a,b)$

non-increasing sequence

Random restarts help

- K-means is a coordinate descent algorithm!

model that can be used for clustering, density est.

Mixtures of Gaussians

Machine Learning – CSE546

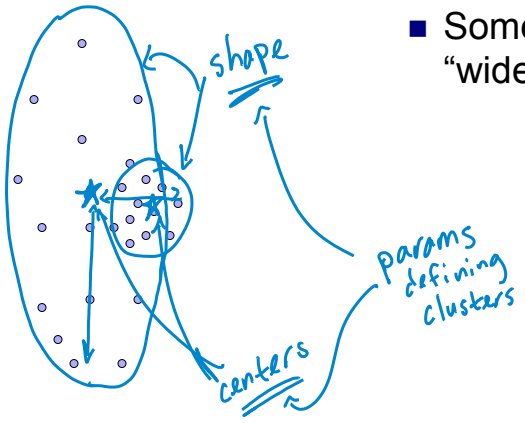
Emily Fox

University of Washington

November 4, 2013

(One) bad case for k-means

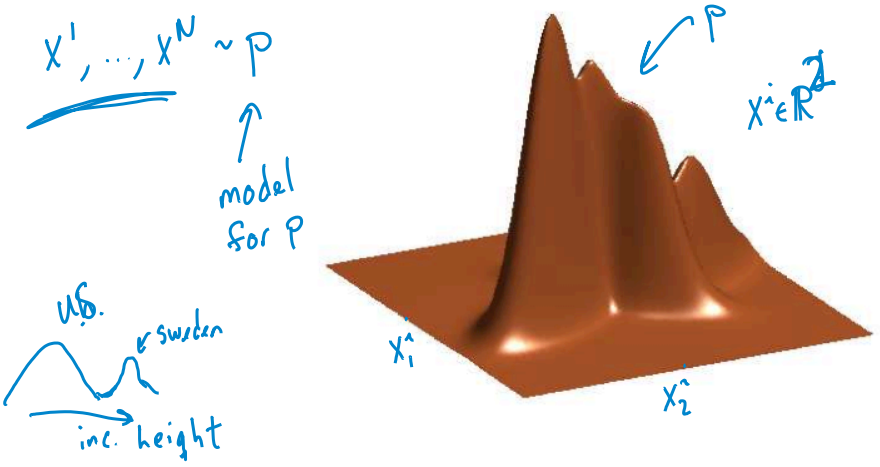
- Clusters may overlap
- Some clusters may be "wider" than others



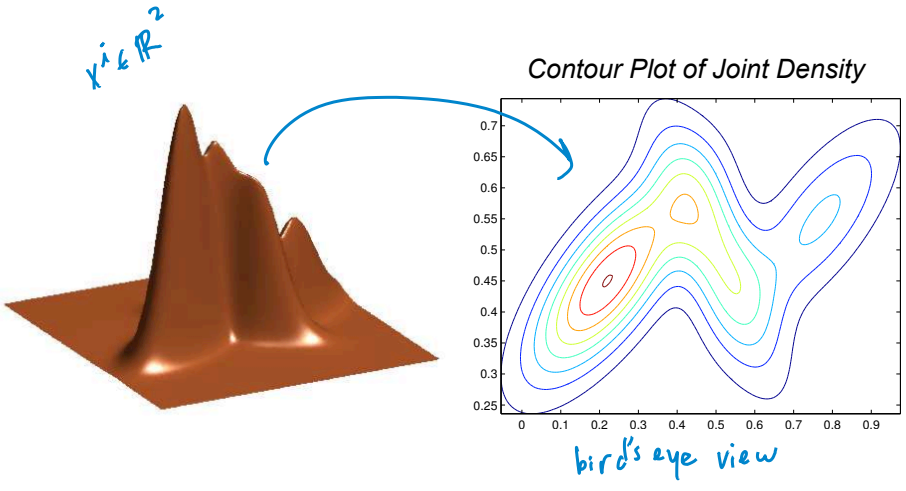
so centers alone don't tell the whole story

Density Estimation

- Estimate a density based on $x^1,\ldots,x^N$



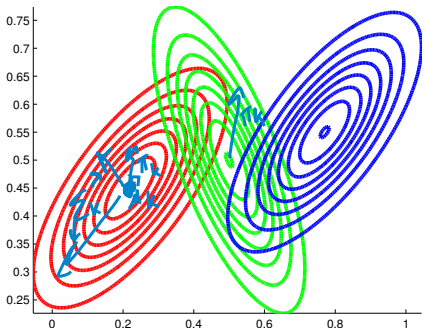
Density Estimation



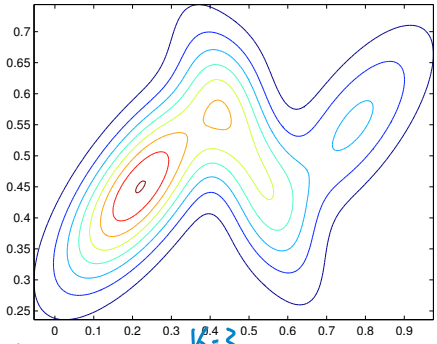
Density as Mixture of Gaussians

- Approximate density with a mixture of Gaussians

Mixture of 3 Gaussians



Contour Plot of Joint Density



Each Gaussian has weight π_k and shape params μ_k, Σ_k w/ $\sum_{k=1}^{K=3} \pi_k = 1$

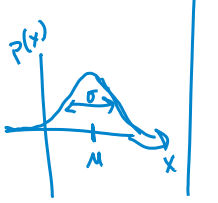
Gaussians in *d* Dimensions

$$P(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} \|\Sigma\|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)\right]$$

dxd covariance matrix

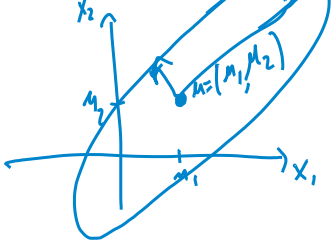
d-dim mean vector

1D: $p(x) \propto e^{\frac{-(x-\mu)^2}{2\sigma^2}}$



$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix}$$

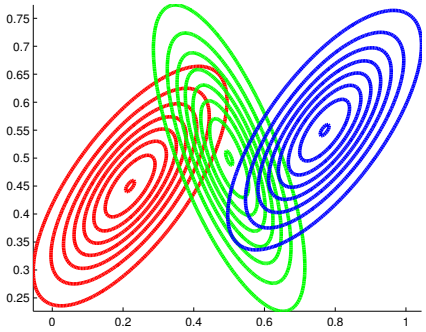
$\sigma_{12} = cov(X_1, X_2)$



Density as Mixture of Gaussians

- Approximate density with a mixture of Gaussians

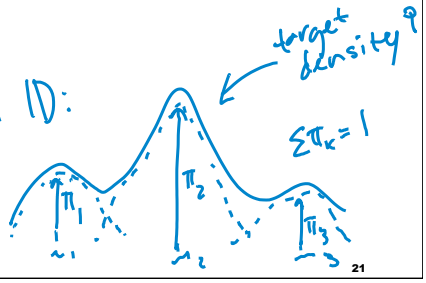
Mixture of 3 Gaussians



$$p = p(x^i|\pi,\mu,\Sigma) = \sum_{k=1}^{K} \pi_k N(x^i|\mu_k,\Sigma_k)$$

$\{\pi_1,\dots,\pi_K\}$ $\{\mu_k,\Sigma_k\}$

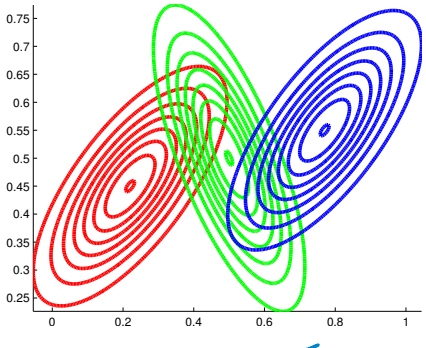
In 1D:



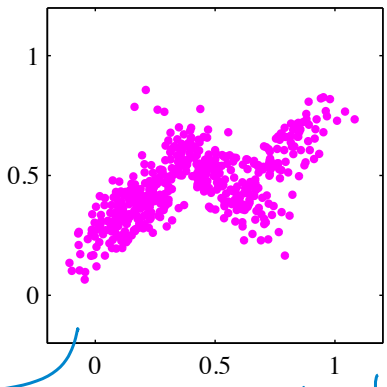
Density as Mixture of Gaussians

- Approximate with density with a mixture of Gaussians

Mixture of 3 Gaussians



Our actual observations



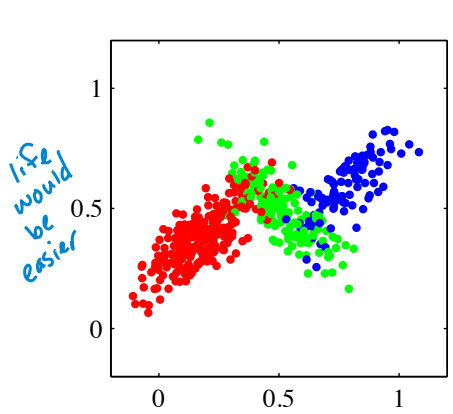
How???

from obs., est. model params

C. Bishop, Pattern Recognition & Machine Learning

Clustering our Observations

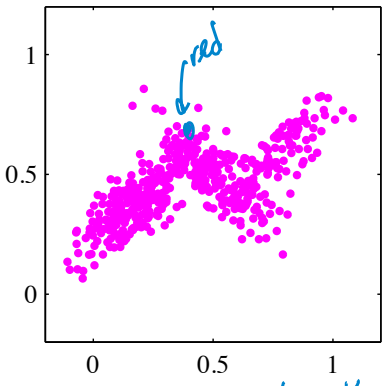
- Imagine we have an assignment of each x^i to a Gaussian



life would be easier

Complete data labeled by true cluster assignments

Our actual observations

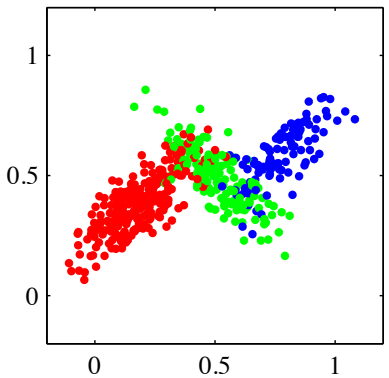


"Incomplete data"

C. Bishop, Pattern Recognition & Machine Learning

Clustering our Observations

- Imagine we have an assignment of each x^i to a Gaussian



Complete data labeled by true cluster assignments

- Introduce latent cluster indicator variable z^i

$z^i \in \{1, \dots, K\}$ $\quad c(i) \to z^i$

$\Pr(z^i = k) = \pi_k$

- Then we have

$$p(x^i | z^i = k, \pi, \mu, \Sigma) = \mathcal{N}(x^i | \mu_k, \Sigma_k)$$

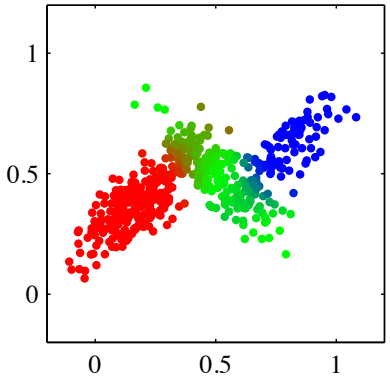
param est. is easy if we have $\{z_i\}$

$\Rightarrow$ decouples into K Gauss. est.

C. Bishop, Pattern Recognition & Machine Learning

Clustering our Observations

- We must infer the cluster assignments from the observations



Soft assignments to clusters

- Posterior probabilities of assignments to each cluster *given* model parameters:

"responsibilities"

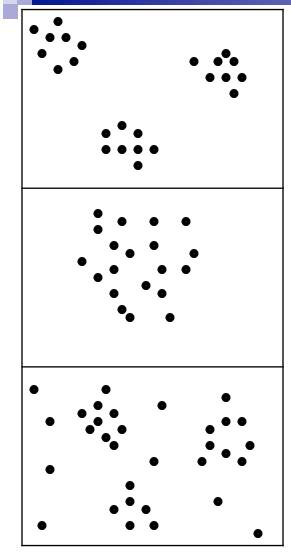
$$r_{ik} = p(z^i = k | x^i, \pi, \mu, \Sigma) =$$

$$= \frac{\pi_k N(x^i | \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x^i | \mu_j, \Sigma_j)}$$

motivates an iterative alg.

C. Bishop, Pattern Recognition & Machine Learning

Unsupervised Learning: not as hard as it looks



Sometimes easy

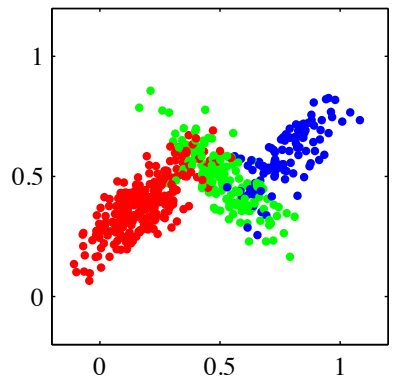
Sometimes impossible

and sometimes in between

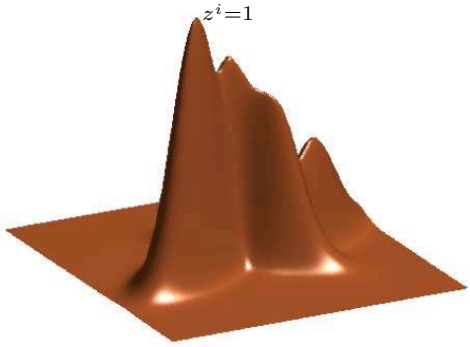
Summary of GMM Concept

- Estimate a density based on $x^1,\ldots,x^N$

$$p(x^i|\pi,\mu,\Sigma) = \sum_{z^i=1}^{K} \pi_{z^i}\mathcal{N}(x^i|\mu_{z^i},\Sigma_{z^i})$$



Complete data labeled by true cluster assignments



Surface Plot of Joint Density, Marginalizing Cluster Assignments

Summary of GMM Components

- Observations $x^i \in \mathbb{R}^d, \quad i = 1,2,\ldots,N$
- Hidden cluster labels $z_i \in \{1,2,\ldots,K\}, \quad i = 1,2,\ldots,N$
- Hidden mixture means $\mu_k \in \mathbb{R}^d, \quad k = 1,2,\ldots,K$
- Hidden mixture covariances $\Sigma_k \in \mathbb{R}^{d\times d}, \quad k = 1,2,\ldots,K$
- Hidden mixture probabilities $\pi_k, \quad \sum_{k=1}^{K} \pi_k = 1$

Gaussian mixture marginal and conditional likelihood **:**

$$p(x^i|\pi,\mu,\Sigma) = \sum_{z^i=1}^{K} \pi_{z^i}\ p(x^i|z^i,\mu,\Sigma)$$

$$p(x^i|z^i,\mu,\Sigma) = \mathcal{N}(x^i|\mu_{z^i},\Sigma_{z^i})$$