



Clustering K-means

Machine Learning – CSEP546

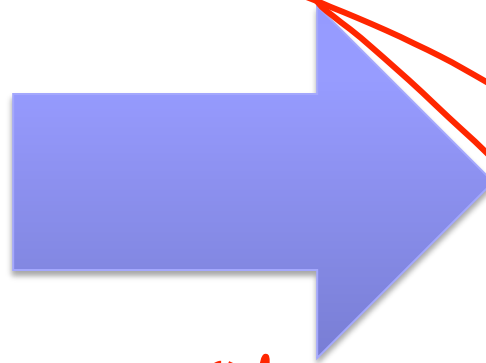
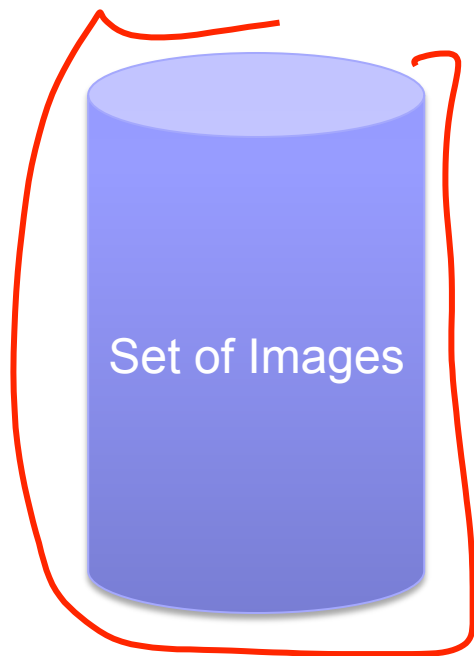
Carlos Guestrin

University of Washington

February 18, 2014

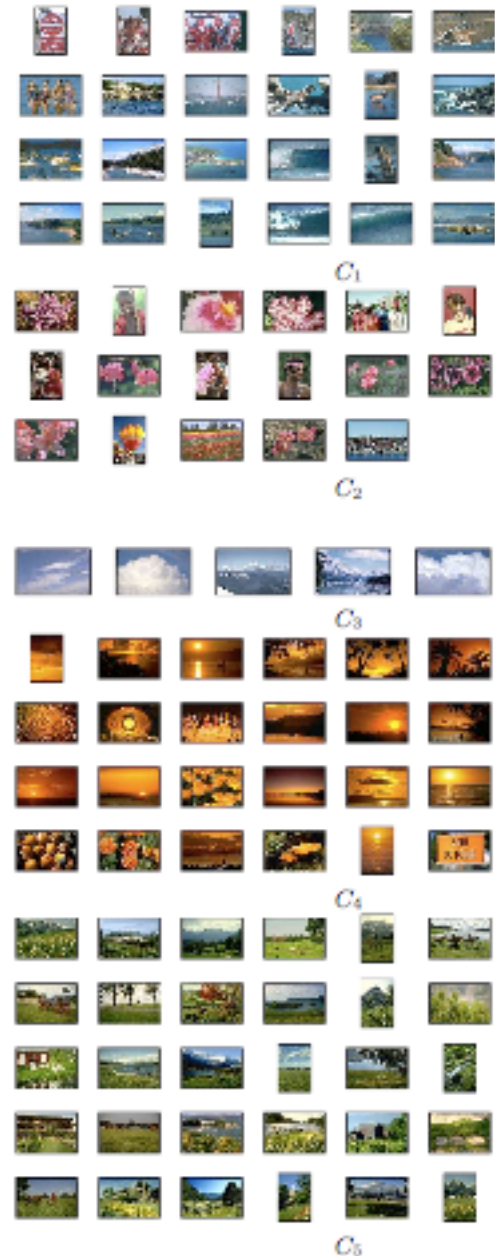
Clustering images

Unsupervised \Rightarrow no given labels



"discover"

groups of image



Clustering web search results

Clusty

web news images wikipedia blogs jobs more »

race Search advanced preferences

clusters sources sites

All Results (238) remix

- Car (28)
 - Race cars (7)
 - Photos, Races Scheduled (5)
 - Game (4)
 - Track (3)
 - Nascar (2)
 - Equipment And Safety (2)
 - Other Topics (7)
- Photos (22)
- Game (14)
- Definition (13)
- Team (18)
- Human (8)
 - Classification Of Human (2)
 - Statement, Evolved (2)
 - Other Topics (4)
- Weekend (8)
- Ethnicity And Race (7)
- Race for the Cure (8)
- Race Information (8)

find in clusters: Find

Cluster Human contains 8 documents.

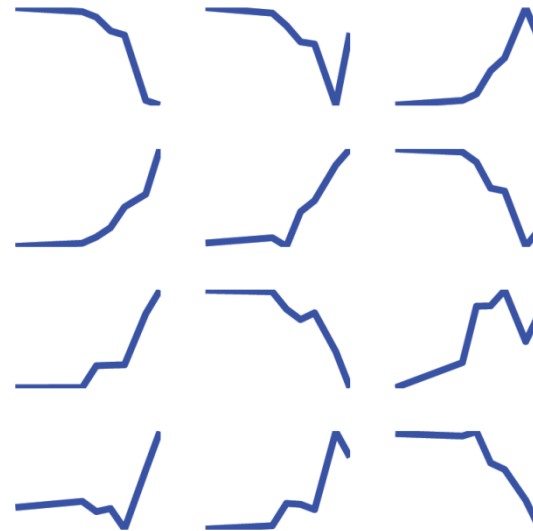
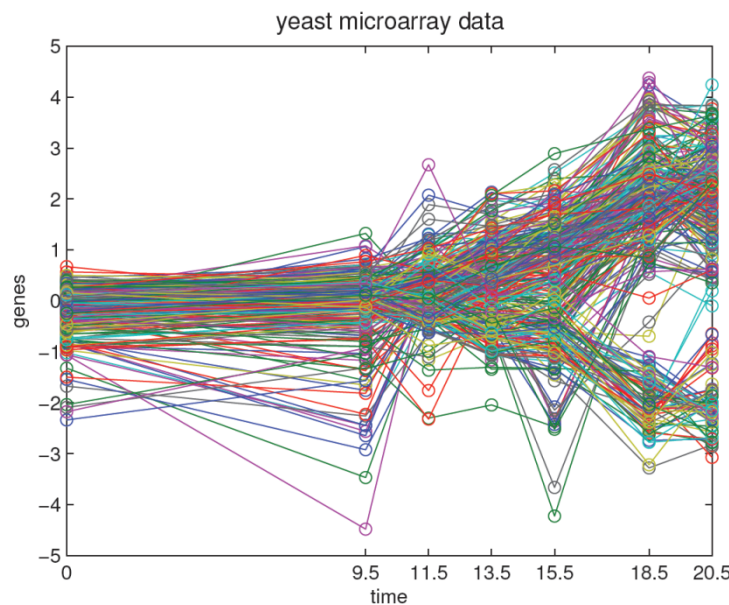
Search Results

- [1. Race \(classification of human beings\) - Wikipedia, the free ...](#)
The term **race** or racial group usually refers to the concept of dividing **humans** into populations or groups on the basis of various sets of characteristics. The most widely used **human** racial categories are based on visible traits (especially skin color, cranial or facial features and hair texture), and self-identification. Conceptions of **race**, as well as specific ways of grouping **races**, vary by culture and over time, and are often controversial for scientific as well as social and political reasons. History · Modern debates · Political and ...
[en.wikipedia.org/wiki/Race_\(classification_of_human_beings\)](#) - [cache] - Live, Ask
- [2. Race - Wikipedia, the free encyclopedia](#)
General. **Racing** competitions The **Race** (yachting **race**), or La course du millénaire, a no-rules round-the-world sailing event; **Race** (biology), classification of flora and fauna; **Race** (classification of **human** beings) **Race** and ethnicity in the United States Census, official definitions of "**race**" used by the US Census Bureau; **Race** and genetics, notion of racial classifications based on genetics. Historical definitions of **race**; **Race** (bearing), the inner and outer rings of a rolling-element bearing. **RACE** in molecular biology "Rapid ... General · Surnames · Television · Music · Literature · Video games
[en.wikipedia.org/wiki/Race](#) - [cache] - Live, Ask
- [3. Publications | Human Rights Watch](#)
The use of torture, unlawful rendition, secret prisons, unfair trials, ... Risks to Migrants, Refugees, and Asylum Seekers in Egypt and Israel ... In the run-up to the Beijing Olympics in August 2008, ...
[www.hrw.org/background/usa/race](#) - [cache] - Ask
- [4. Amazon.com: Race: The Reality Of Human Differences: Vincent Sarich ...](#)
Amazon.com: **Race: The Reality Of Human Differences: Vincent Sarich, Frank Miele: Books ...** From Publishers Weekly Sarich, a Berkeley emeritus anthropologist, and Miele, an editor ...
[www.amazon.com/Race-Reality-Differences-Vincent-Sarich/dp/0813340861](#) - [cache] - Live
- [5. AAPA Statement on Biological Aspects of Race](#)
AAPA Statement on Biological Aspects of **Race** ... Published in the American Journal of Physical Anthropology, vol. 101, pp 569-570, 1996 ... PREAMBLE As scientists who study **human** evolution and variation, ...
[www.physanth.org/positions/race.html](#) - [cache] - Ask
- [6. race: Definition from Answers.com](#)
race n. A local geographic or global **human** population distinguished as a more or less distinct group by genetically transmitted physical
[www.answers.com/topic/race-1](#) - [cache] - Live
- [7. Dopefish.com](#)
Site for newbies as well as experienced Dopefish followers, chronicling the birth of the Dopefish, its numerous appearances in several computer games, and its eventual take-over of the **human** **race**. Maintained by Mr. Dopefish himself, Joe Siegler of Apogee Software.
[www.dopefish.com](#) - [cache] - Open Directory

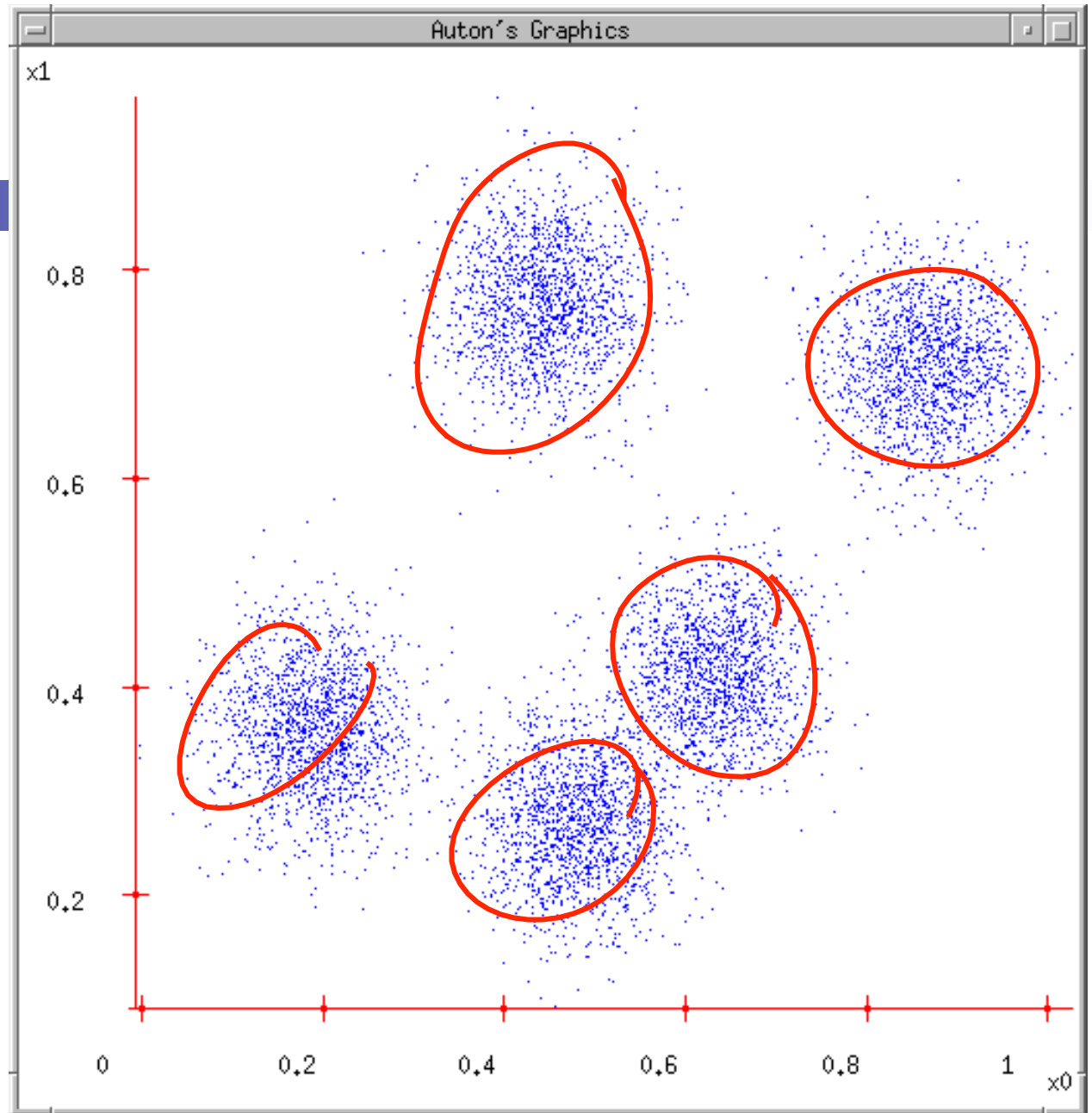
Example

(Taken from Kevin Murphy's ML textbook)

- Data: gene expression levels
- Goal: cluster genes with similar expression trajectories

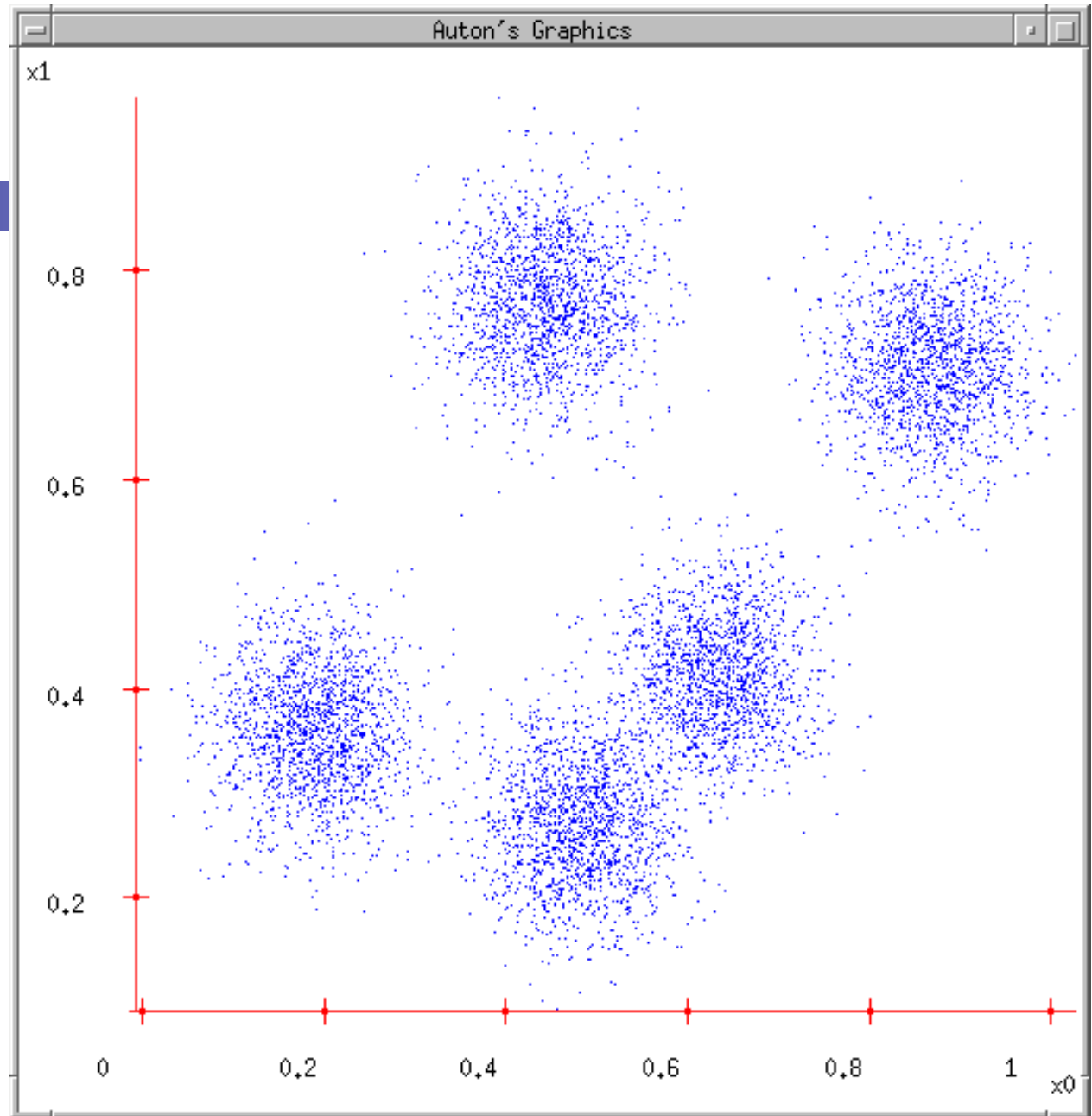


Some Data



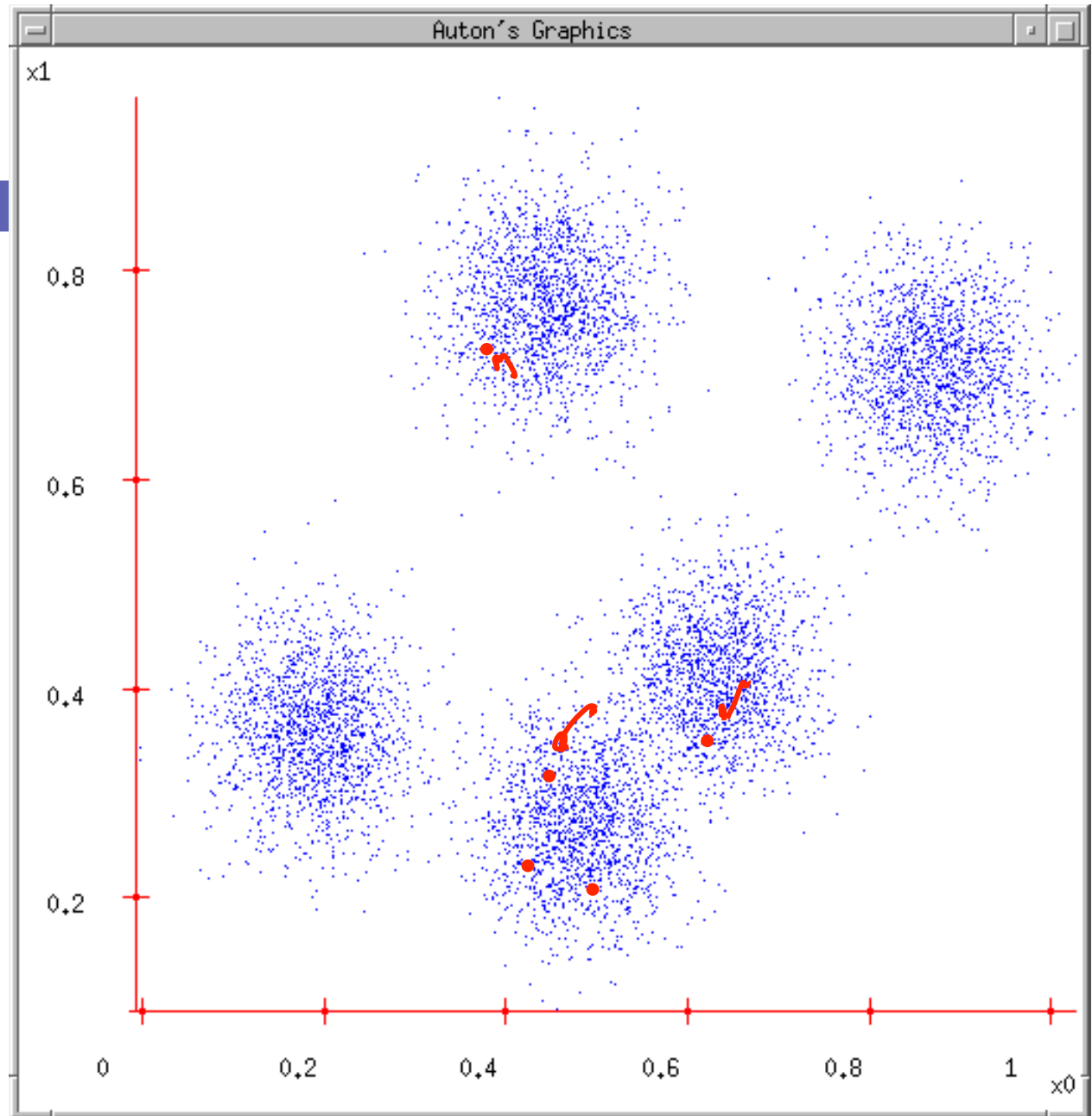
K-means

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



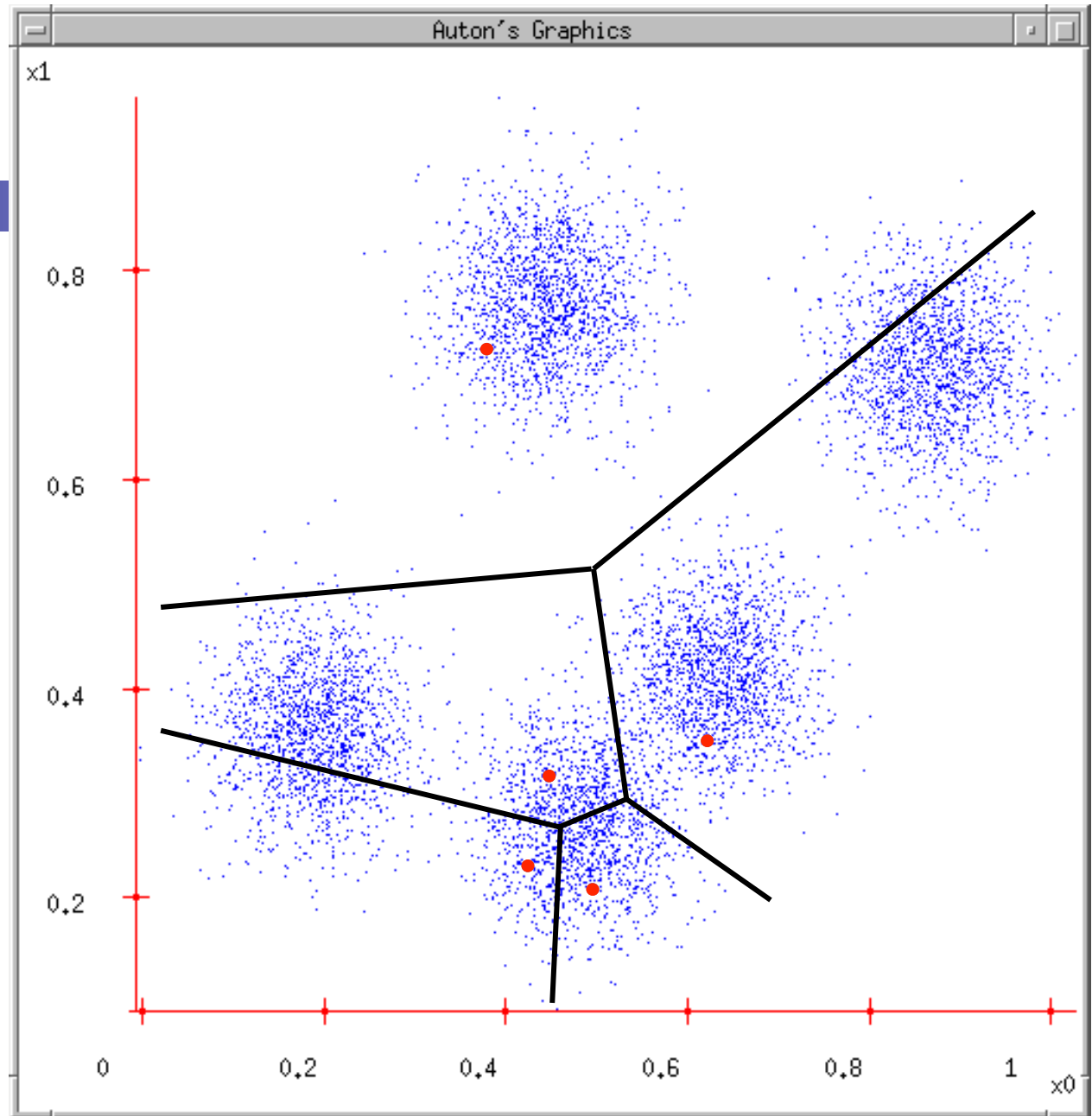
K-means

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



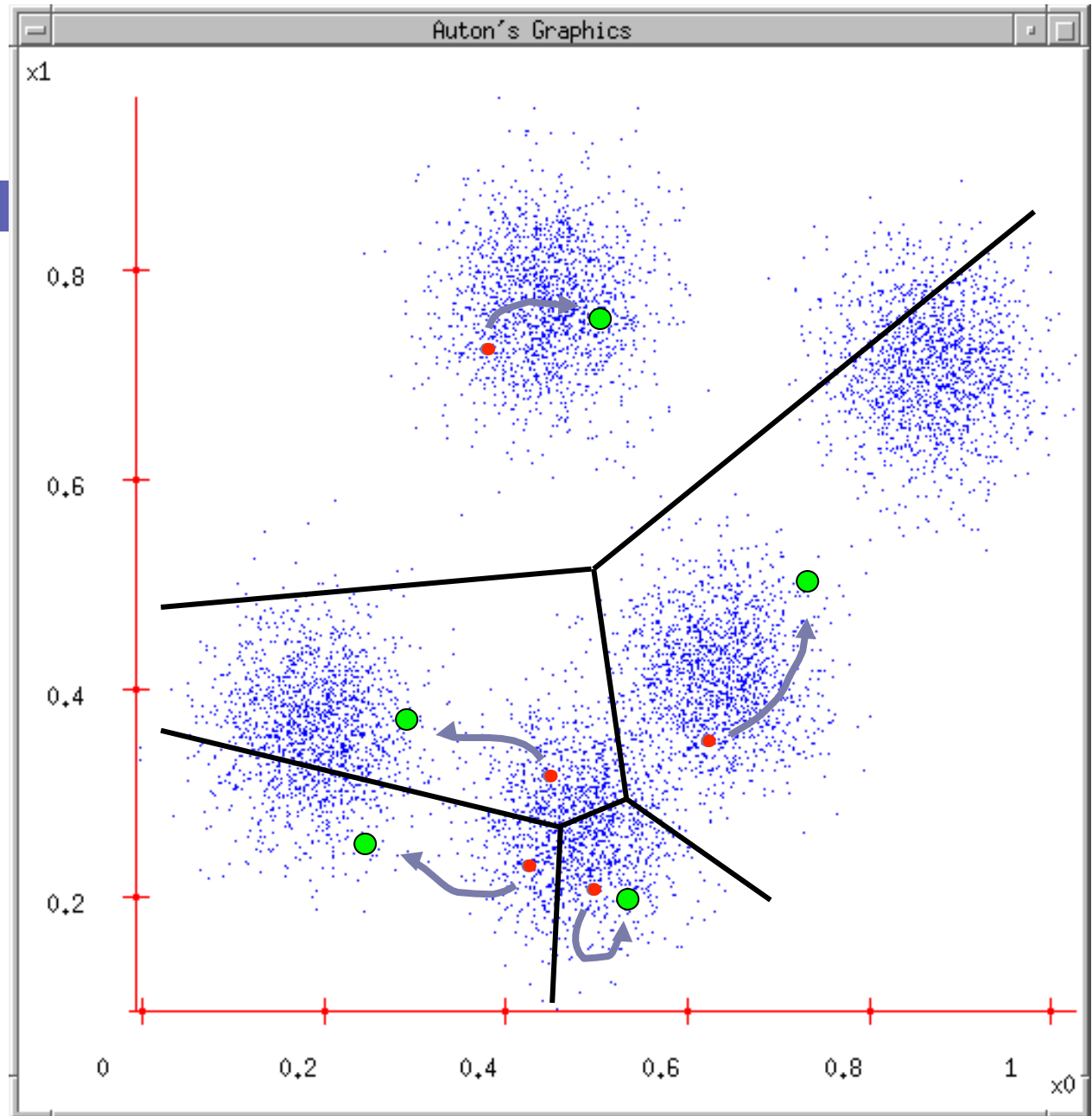
K-means

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)



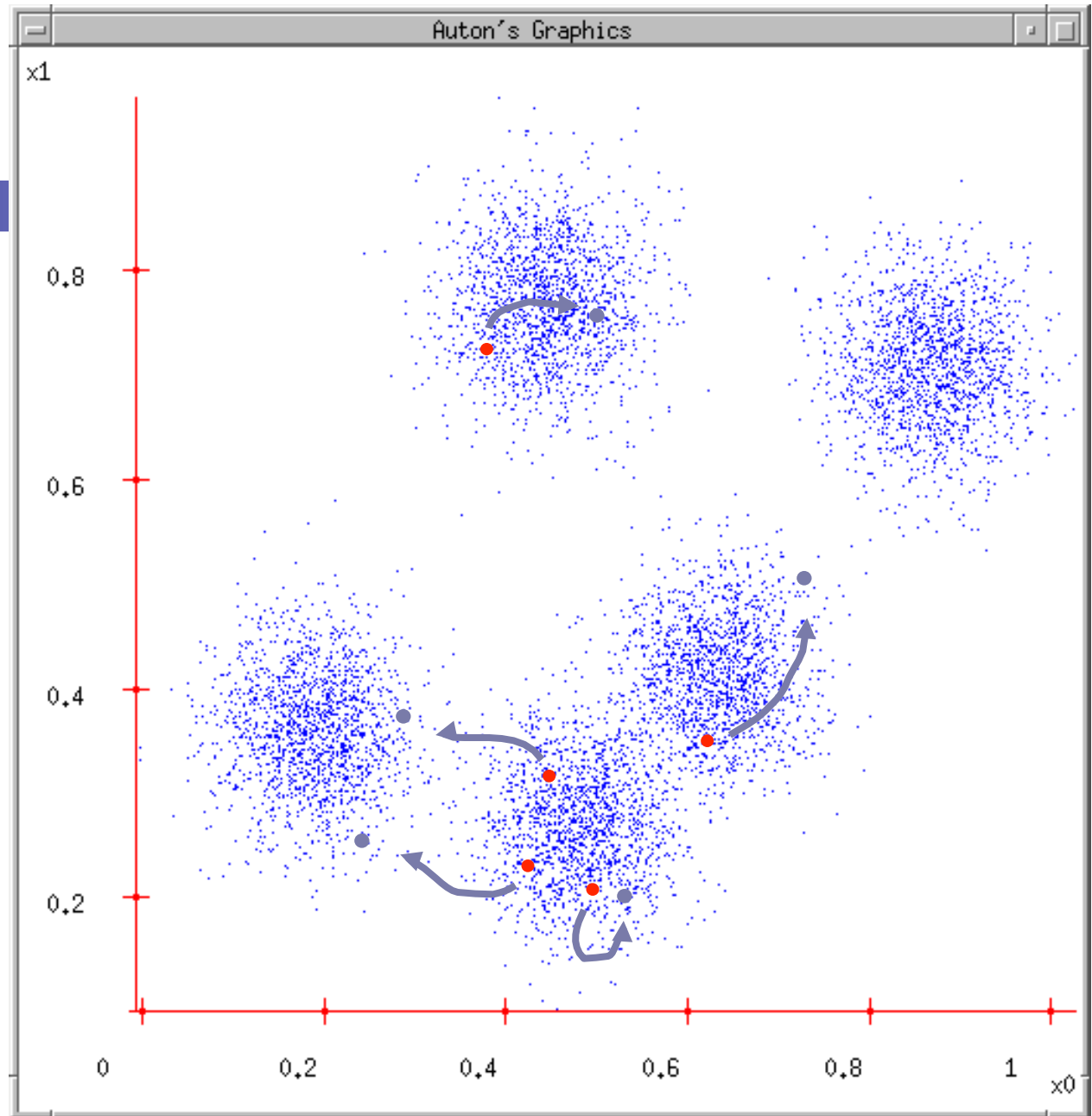
K-means

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 "smartly")
 - $\mu^{(0)} = \mu_1^{(0)}, \dots, \mu_k^{(0)}$

iteration (pointing to $\mu^{(0)}$)
center index (pointing to k)
- **Classify:** Assign each point $j \in \{1, \dots, N\}$ to nearest center:

data point (pointing to j)
nearest neighbor, choose your metric (pointing to "nearest")

 - $C^{(t)}(j) \leftarrow \arg \min_i \|\mu_i - x_j\|^2$

Fix μ , opt C (pointing to the formula)
- **Recenter:** μ_i becomes centroid of its point:

Fix C , opt μ (pointing to the formula)

 - $\mu_i^{(t+1)} \leftarrow \arg \min_{\mu} \sum_{j: C(j)=i} \|\mu - x_j\|^2$

next center (pointing to $\mu_i^{(t+1)}$)
if $\|\cdot\|$ is Euclidean $\Rightarrow \mu_i^{(t+1)} = \frac{\sum_{j: C(j)=i} x_j}{\# \text{ points in cluster } i}$ (pointing to the formula)
all points associated with cluster i (pointing to $j: C(j)=i$)
 - Equivalent to $\mu_i \leftarrow$ average of its points!

What is K-means optimizing?

or loss function

- Potential function $F(\mu, C)$ of centers μ and point allocations C :

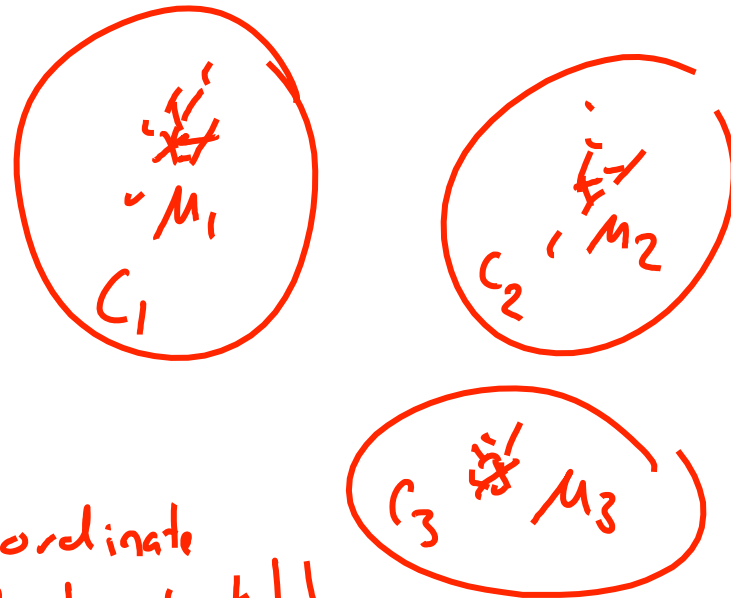
$$\square F(\mu, C) = \sum_{j=1}^N \|\mu_{C(j)} - x_j\|^2$$

the center
assignment
data point

- Optimal K-means:

$$\square \min_{\mu} \min_C F(\mu, C)$$

NP-hard, but we'll do coordinate descent & hope for the best!!



Does K-means converge??? Part 1

- 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

$$\min_{C(1)} \min_{C(2)} \dots \min_{C(N)} \sum_{j=1}^N \|\mu_{C(j)} - x_j\|^2 \leftarrow \text{indep optimisations per data point}$$

$$\Rightarrow \text{For each } j: \min_{C(j)} \|\mu_{C(j)} - x_j\|^2 \leftarrow \text{"classify" step}$$

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

$$\min_{\mu_1} \min_{\mu_2} \dots \min_{\mu_k} \sum_{i=1}^k \sum_{j: C(j)=i} \|\mu_i - x_j\|^2 \quad \leftarrow \text{indep opt per cluster center}$$

$$\Rightarrow \text{For each } i: \min_{\mu_i} \sum_{j: C(j)=i} \|\mu_i - x_j\|^2 \leftarrow \text{"recenter"}$$

\Rightarrow For Euclidean norm, just average

Coordinate descent algorithms

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

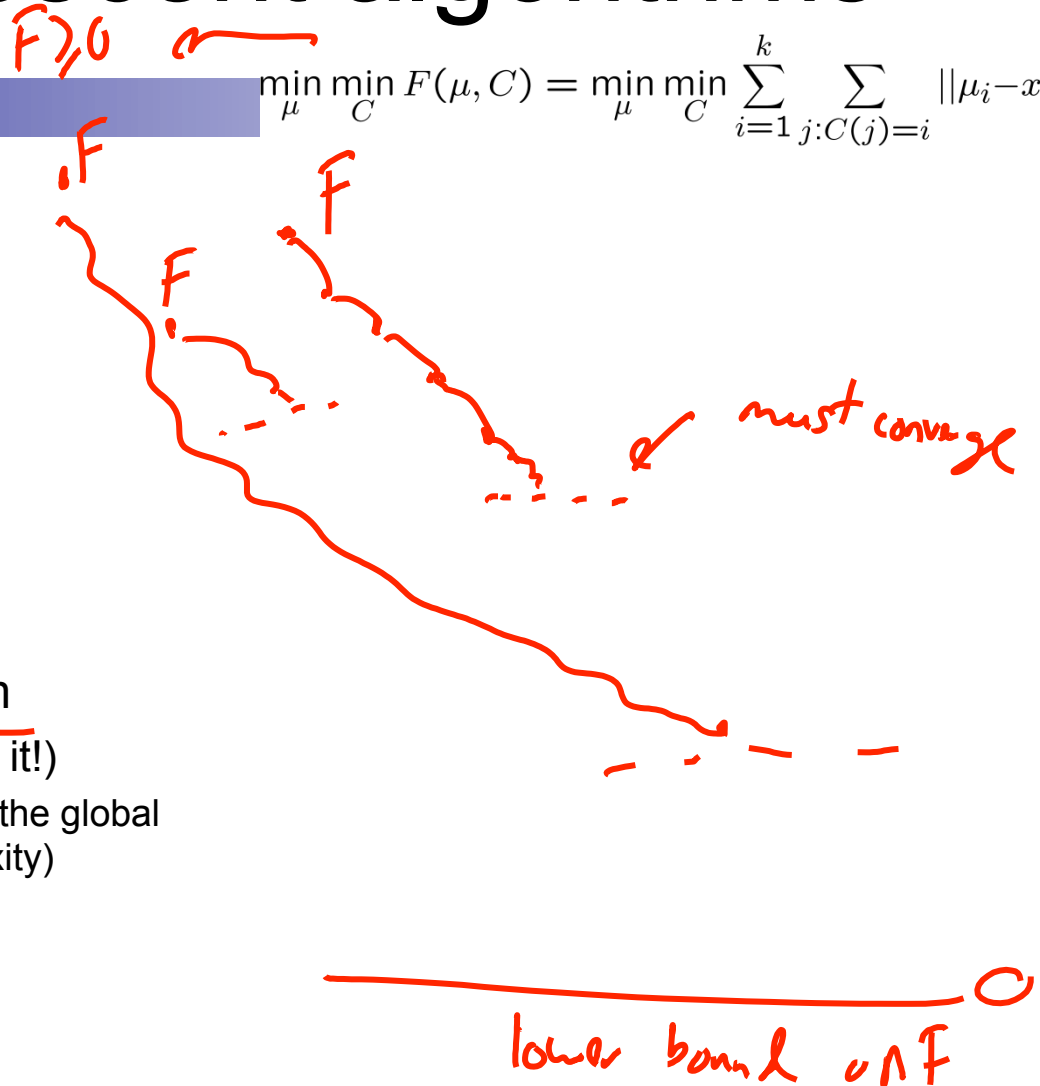
■ Want: $\min_a \min_b F(a, b)$

■ 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)



■ K-means is a coordinate descent algorithm!



Mixtures of Gaussians

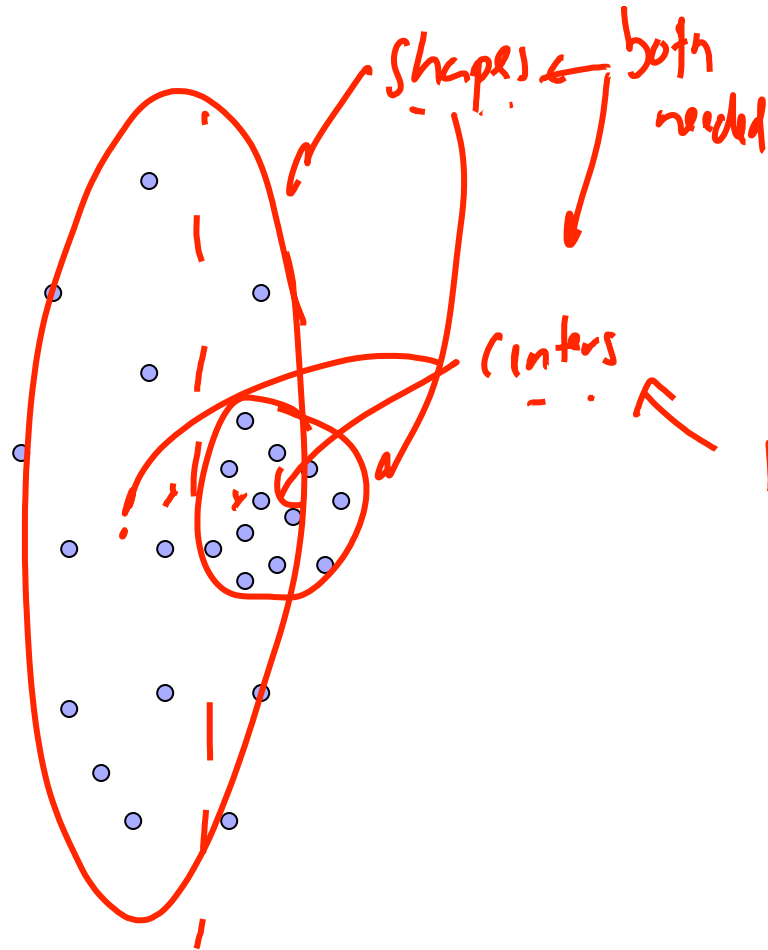
Machine Learning – CSEP546

Carlos Guestrin

University of Washington

February 18, 2014

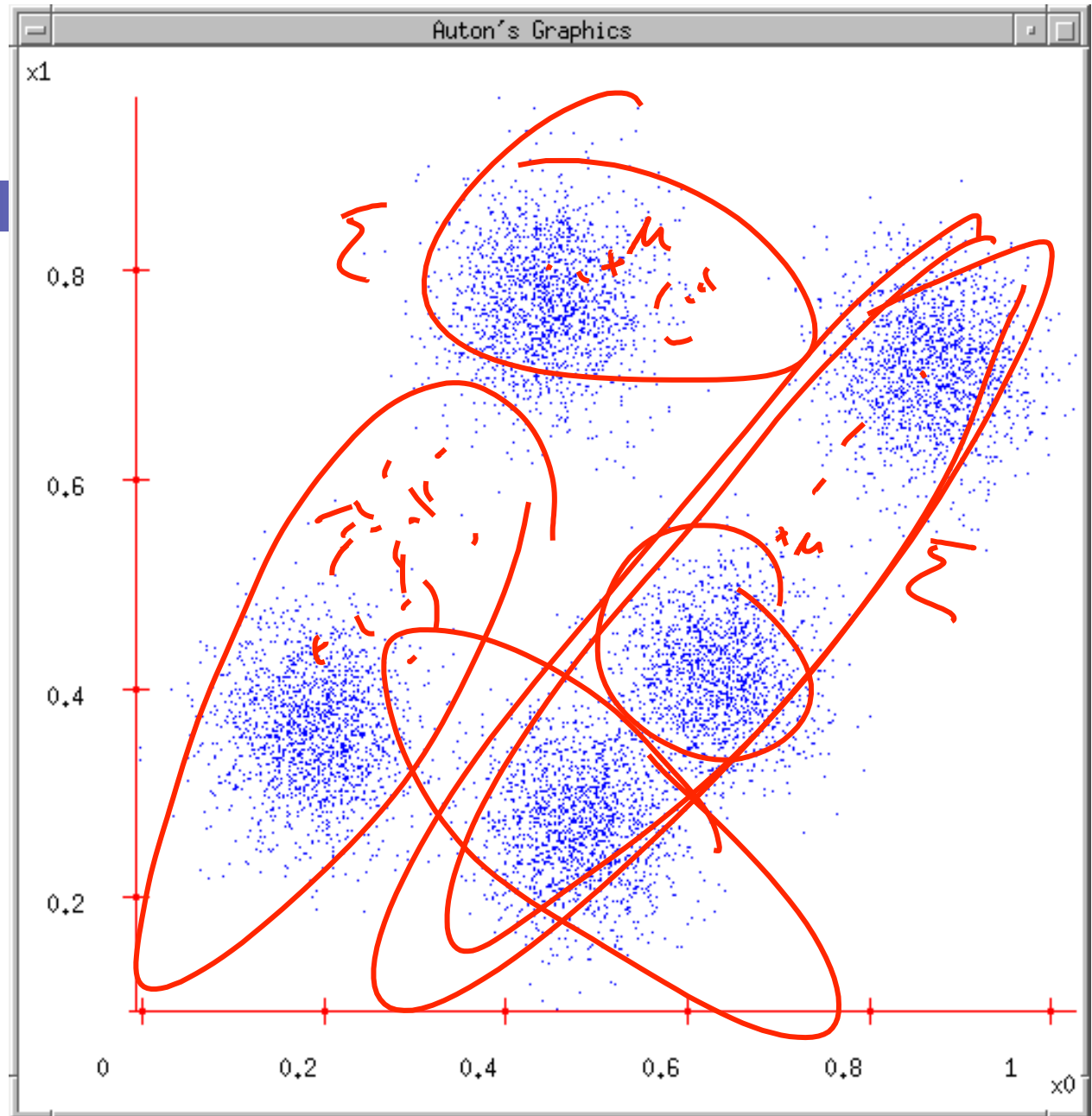
(One) bad case for k-means



- Clusters may overlap
- Some clusters may be “wider” than others

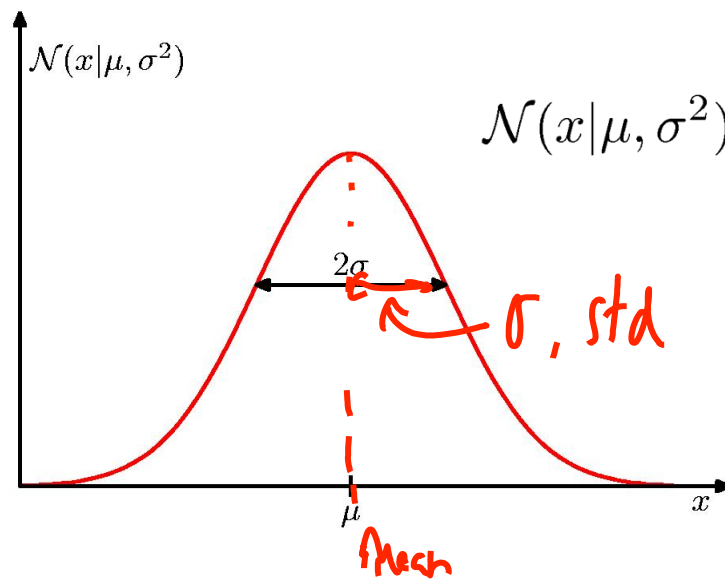
Centers are not enough

Non-spherical data

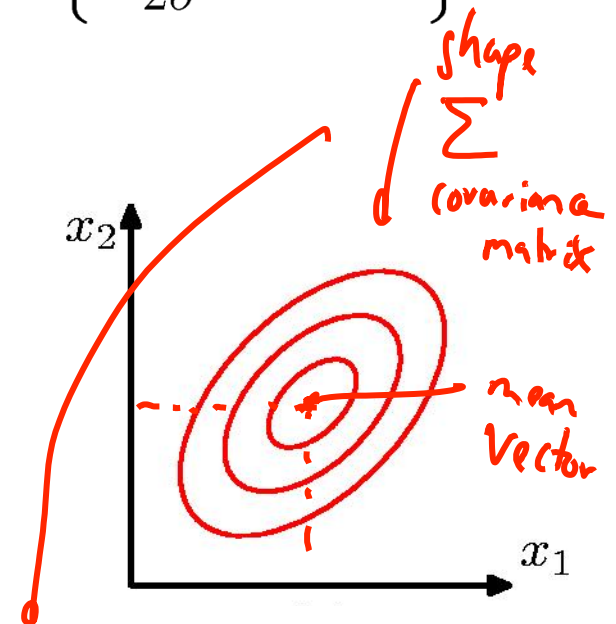


Quick Review of Gaussians

■ Univariate and multivariate Gaussians



$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp \left\{ -\frac{1}{2\sigma^2} (x - \mu)^2 \right\}$$



$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}$$

Two-Dimensional Gaussians

uncorrelated

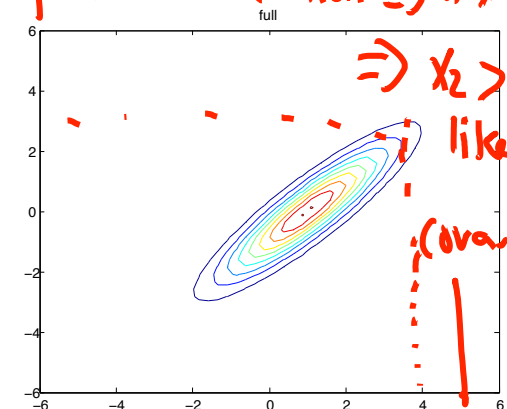
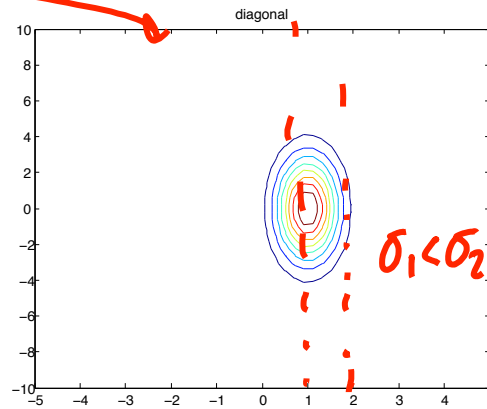
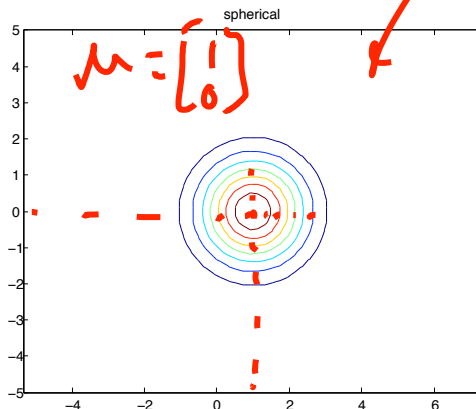
correlated variable

$\sigma_{12} > 0 \Rightarrow$ positive correlation \Rightarrow if $x_1 > \mu_1$

$\Rightarrow x_2 > \mu_2$
likely

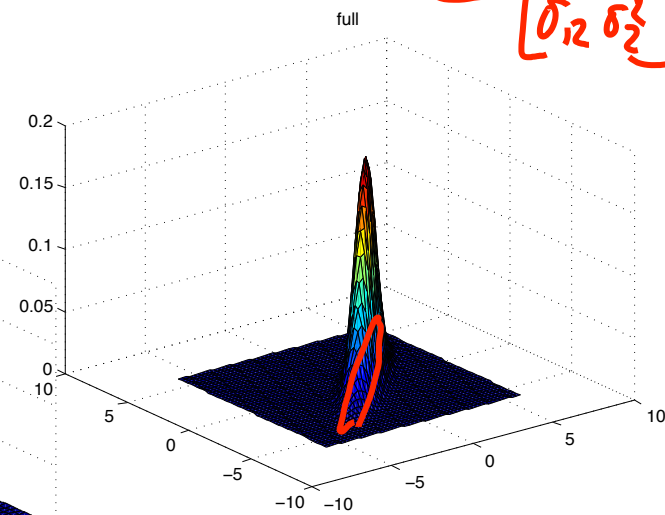
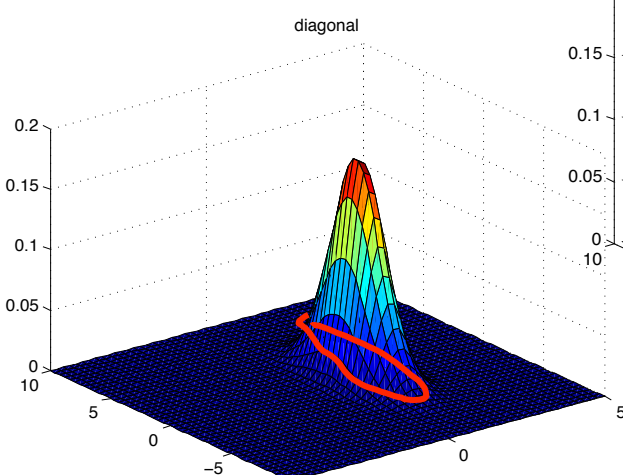
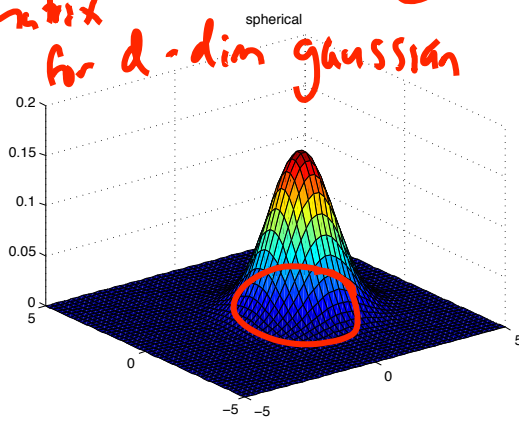
covariance

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



$d \times d : \Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$
matrix
for d-dim gaussian

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$



Gaussians in d Dimensions

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

per dim
variance

$$\mu = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_i \\ \vdots \\ \mu_d \end{bmatrix}$$

avg of
ith dim

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \dots & \sigma_{dd} \end{bmatrix}$$

Symmetric
matrix
 $\sigma_{ij} = \sigma_{ji}$

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{13} & \sigma_{14} \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_5^2 \\ & & & & \sigma_{56} \\ & & & & & \sigma_d^2 \end{bmatrix}$$

covariance
between dims 5 & 6

Learning Gaussians

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

- Given data: x^1, x^2, \dots, x^n

- MLE for mean:

$$\mu = \frac{\sum_i x^i}{n}$$

- MLE for covariance:

$$\sigma_{ij} = \frac{1}{n} \sum_{u=1}^n \sum_{v=1}^n (\mu_i - x_i^u) (\mu_j - x_j^v)$$

When the world is not Gaussian

- Distribution of male heights in US

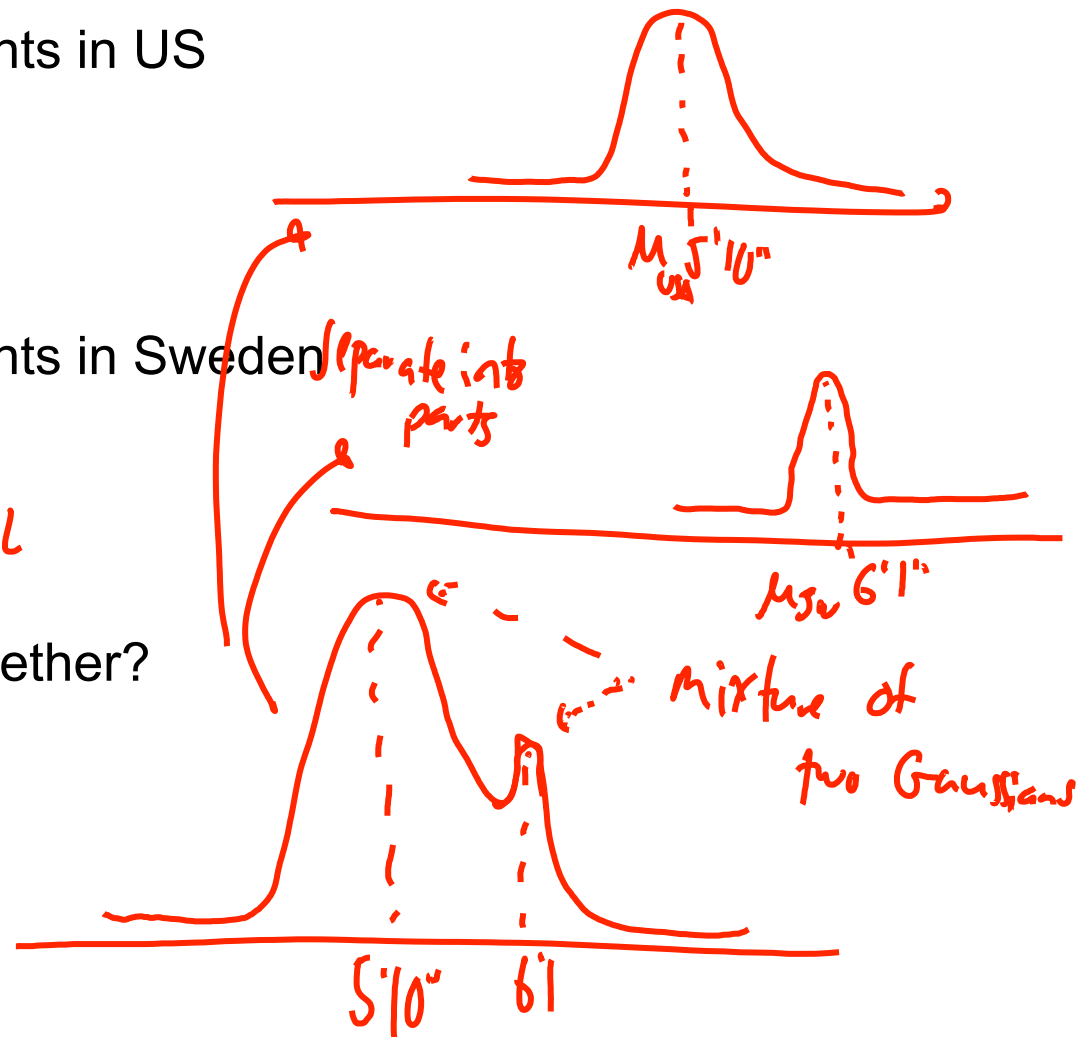
- Distribution of male heights in Sweden

- What if we mix these together?

pop USA: 366M

pop Sweden: 10M

GOAL



Gaussian Mixture Model

- Most commonly used mixture model

- Observations: x^1, \dots, x^m

K Gaussians

- Parameters:

$\mu_i, \Sigma_i \quad i = 1 \dots K \quad z^j \in \{1, 2, \dots, K\}$

- Cluster indicator:

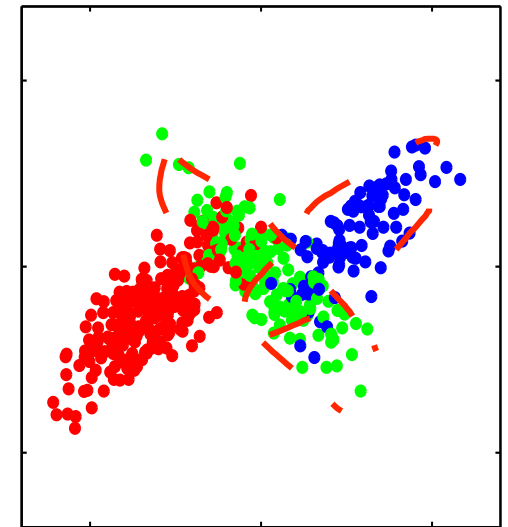
$z^j \leftarrow$ which Gaussian a point comes from

- Per-cluster likelihood:

Learn a Gaussian per cluster, if we had z^j

- Ex. z^i = country of origin, x^i = height of i^{th} person

□ k^{th} mixture component = distribution of heights in country k



Generative Model

$$\begin{aligned}\pi_{usa} &= \frac{300M}{310M} \\ \pi_{sweden} &= \frac{10M}{310M}\end{aligned}$$

- We can think of *sampling* observations from the model

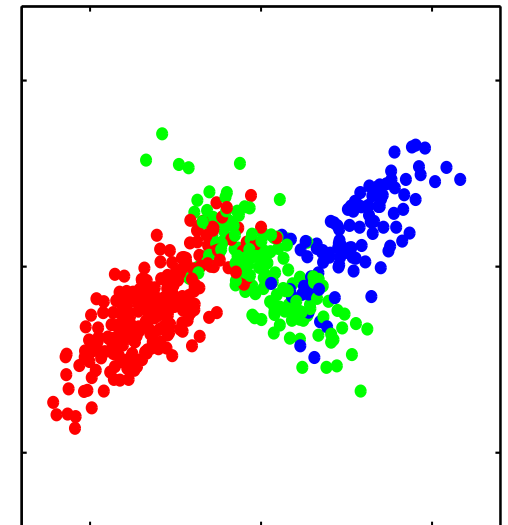
- For each observation i ,
 - Sample a cluster assignment

z^i sampled with prob $\pi_1, \pi_2, \dots, \pi_k$

- Sample the observation from the selected Gaussian

$$x^i \sim N(\mu_i, \Sigma_i)$$

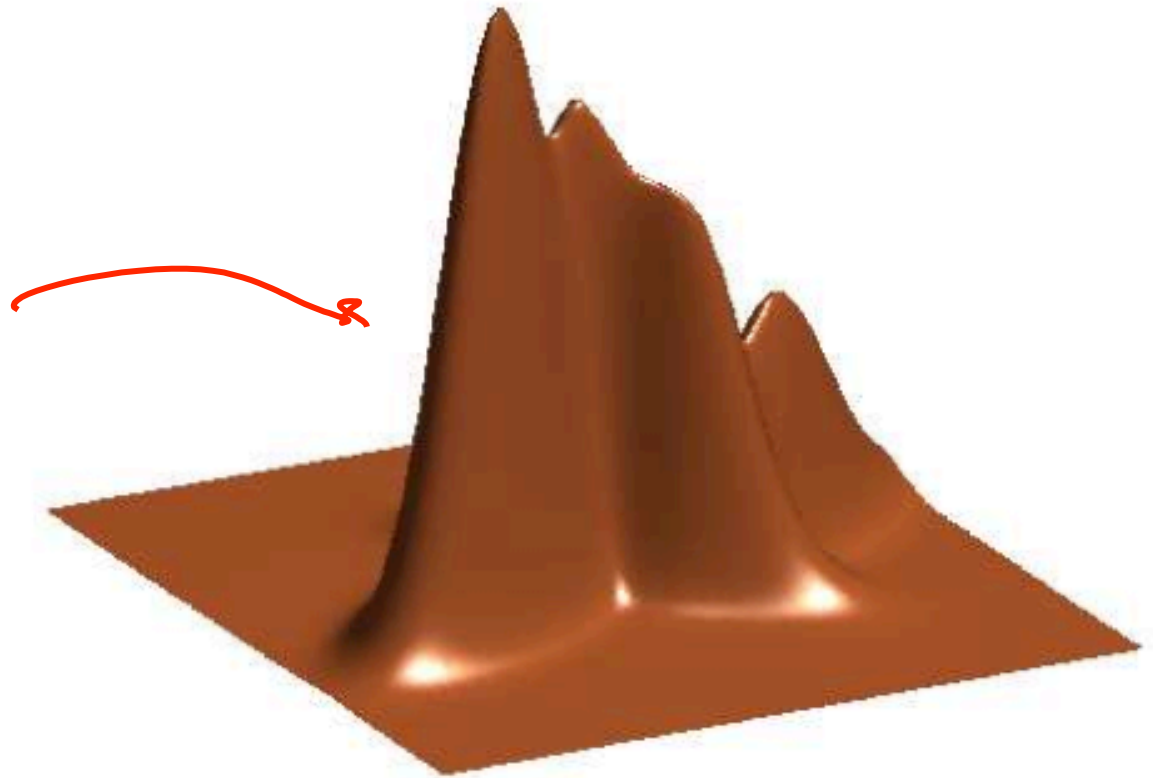
$$z^i = \text{sweden}, \quad x^i \sim N(6'1'', (3'')^2)$$



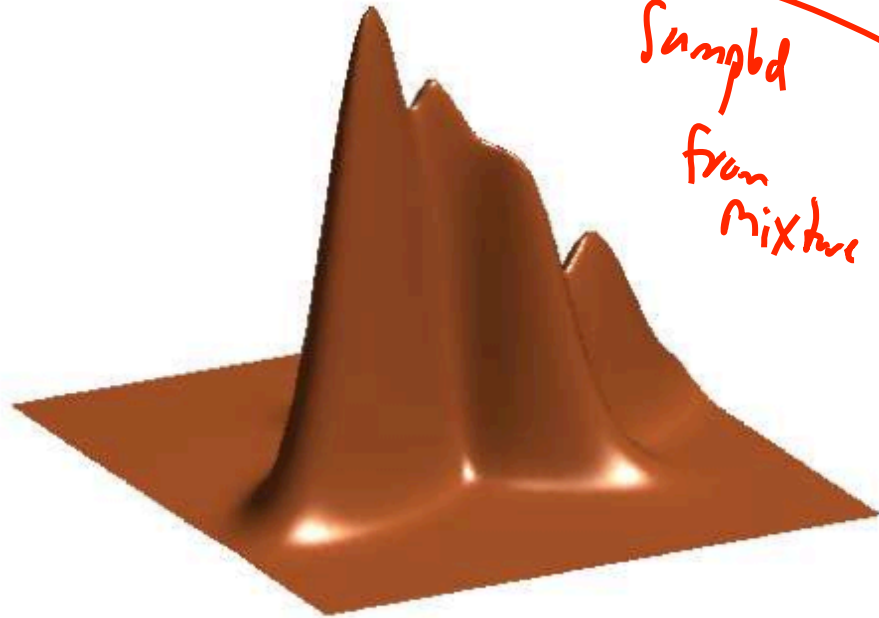
Density Estimation

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

*e.g., fit mixture of
Gaussians
from points*

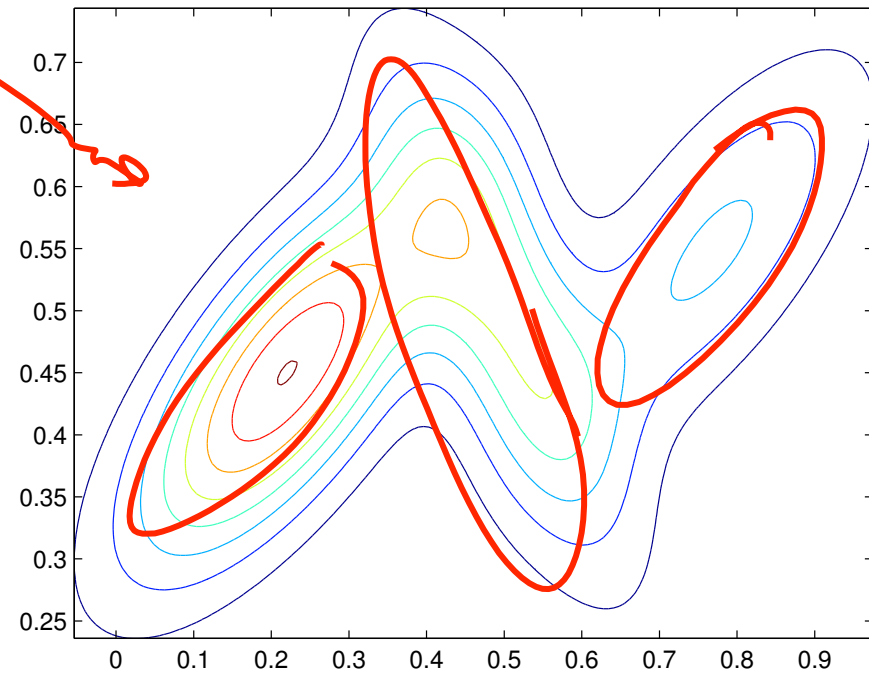


Density Estimation



x^i
sampled
from
mixture

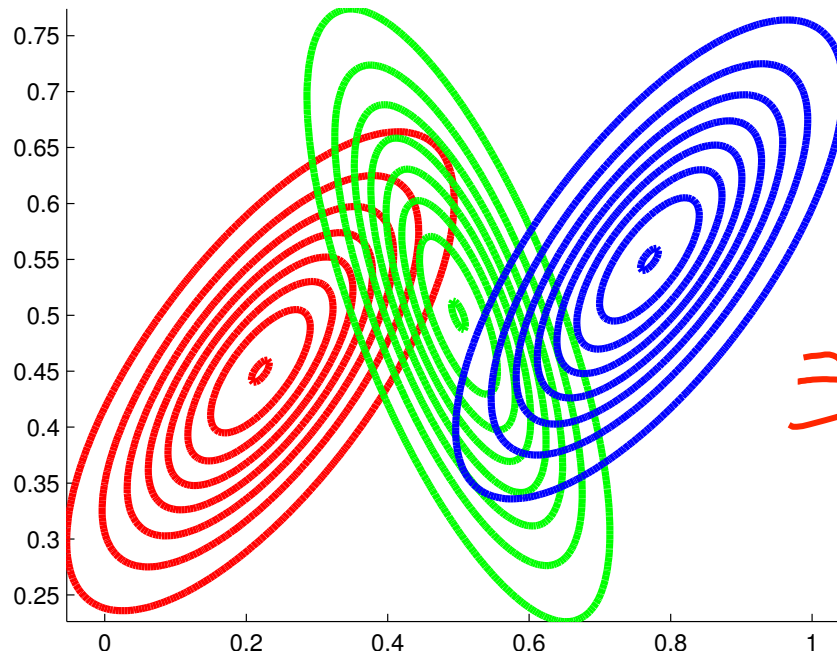
Contour Plot of Joint Density



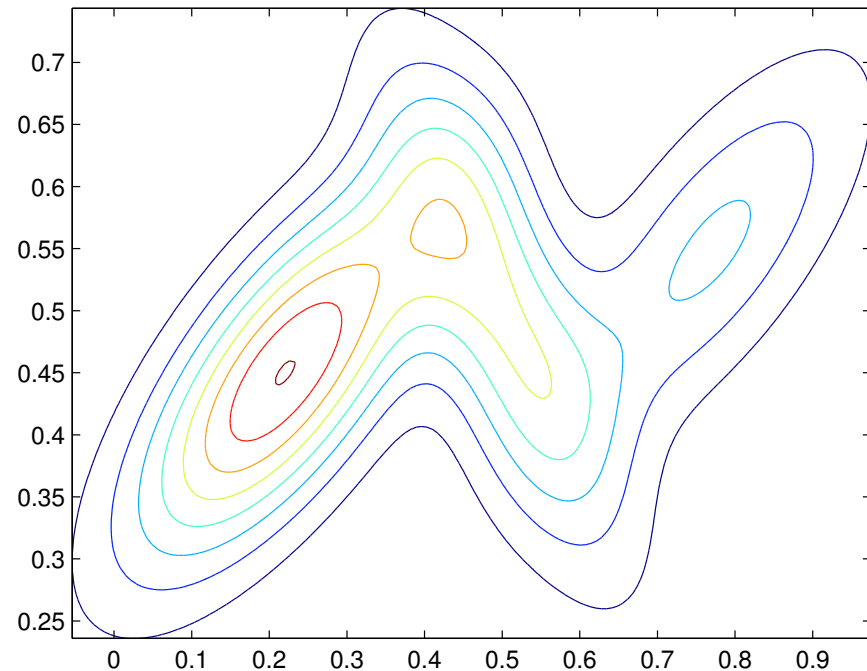
Density as Mixture of Gaussians

- Approximate density with a mixture of Gaussians

Mixture of 3 Gaussians



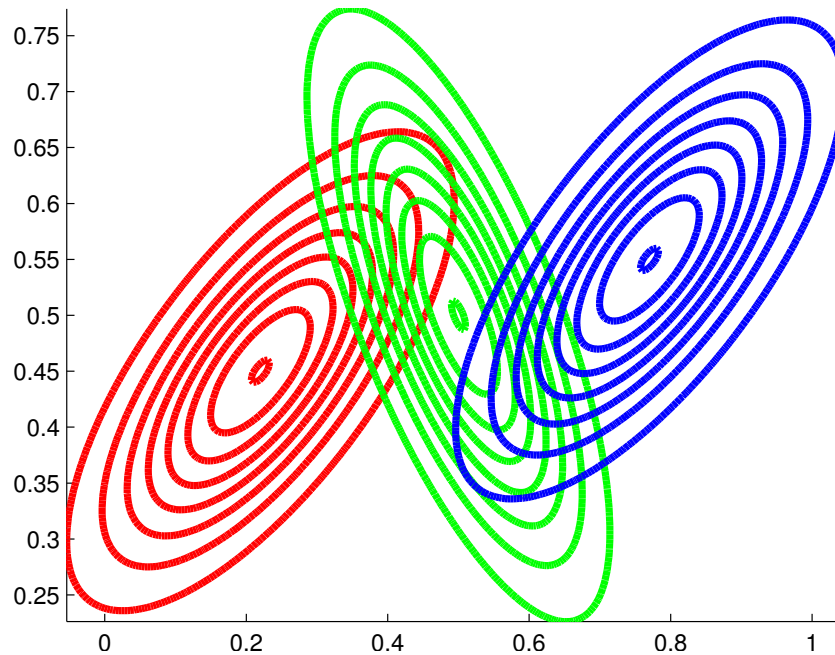
Contour Plot of Joint Density



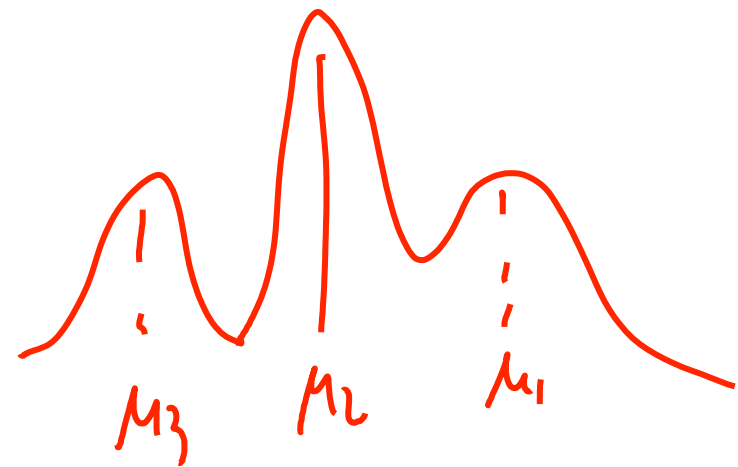
Density as Mixture of Gaussians

- Approximate density with a mixture of Gaussians

Mixture of 3 Gaussians



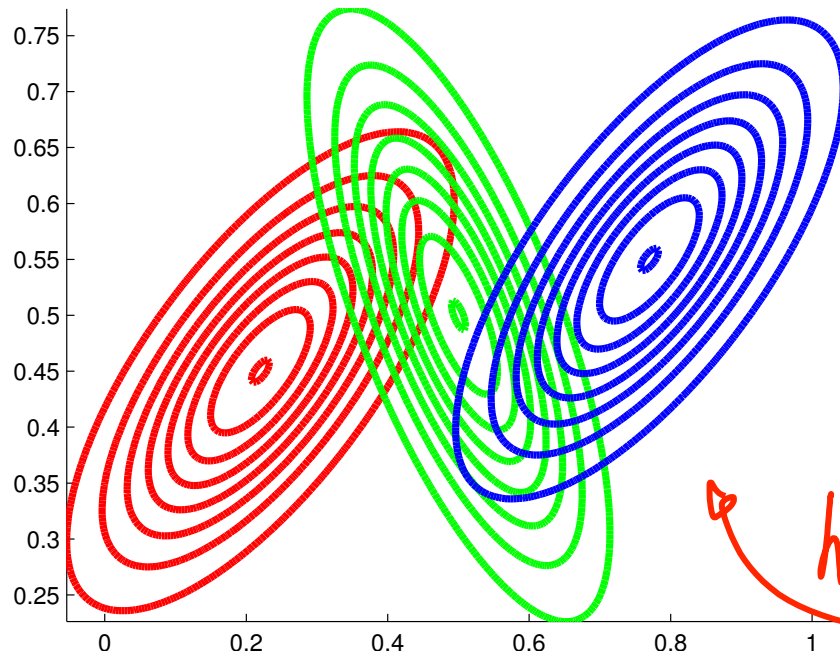
$$p(x^i | \pi, \mu, \Sigma) = \sum_{j=1}^K \pi_j N(\mu_j, \Sigma_j)$$



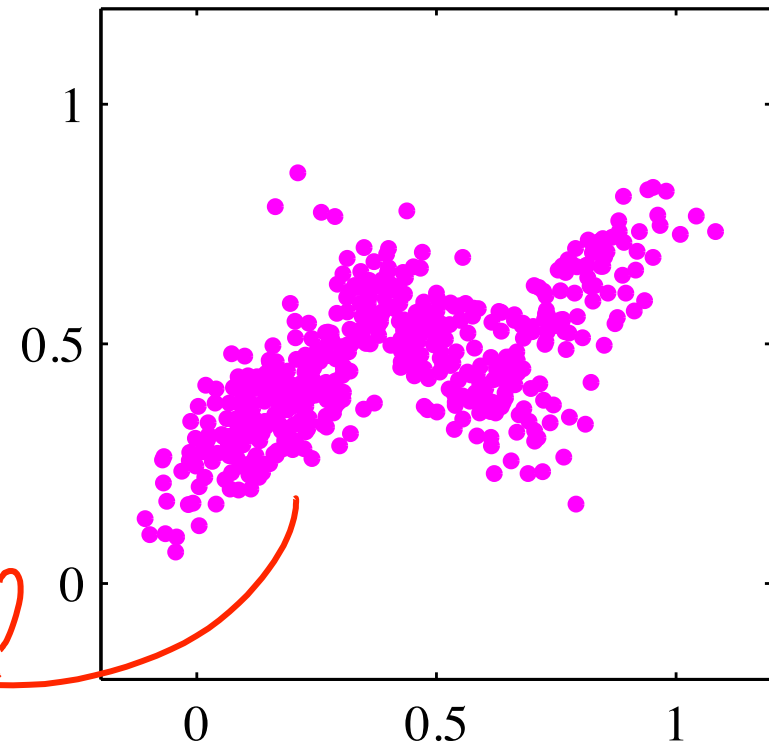
Density as Mixture of Gaussians

- Approximate with density with a mixture of Gaussians

Mixture of 3 Gaussians



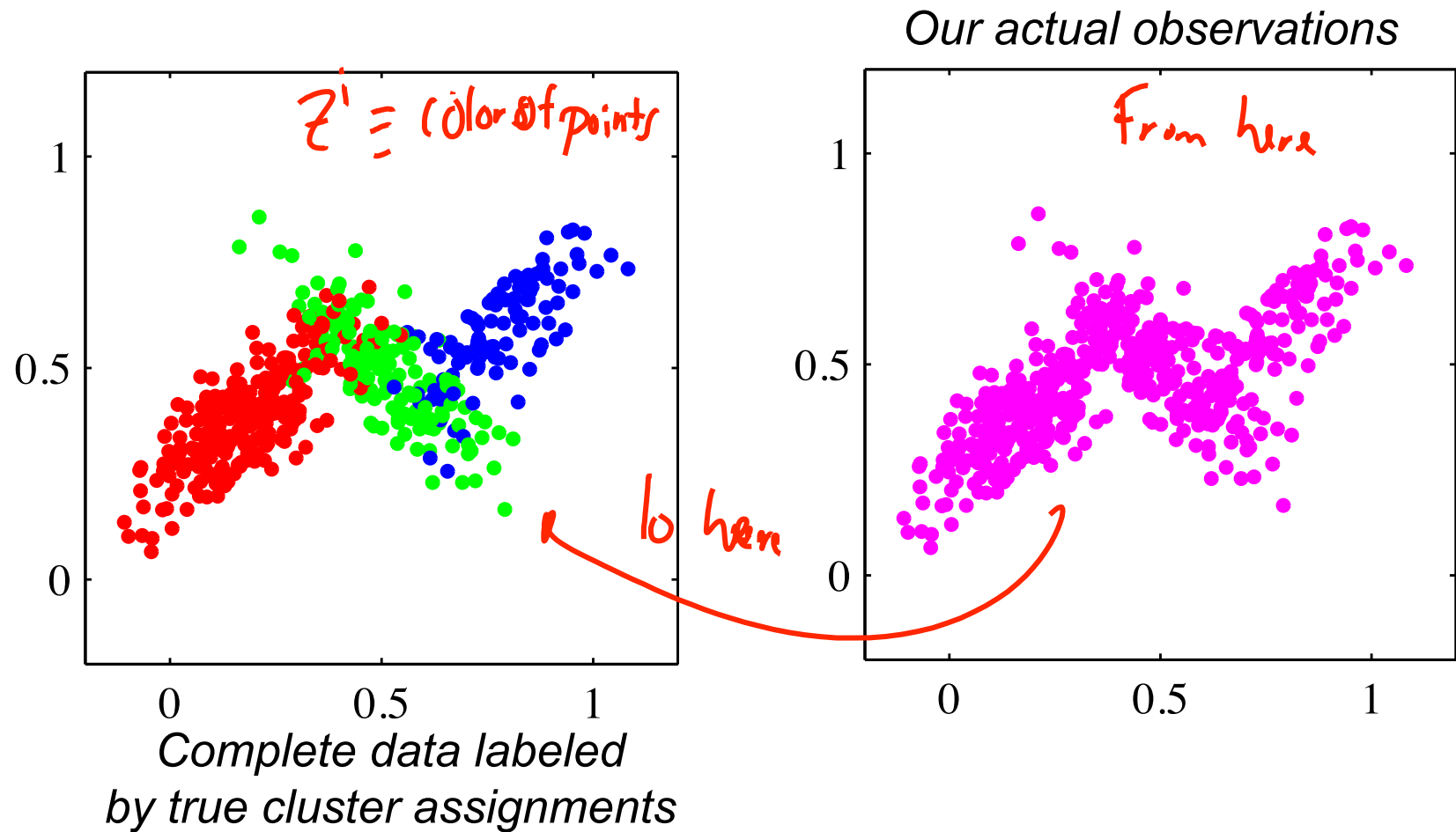
Our actual observations



how??

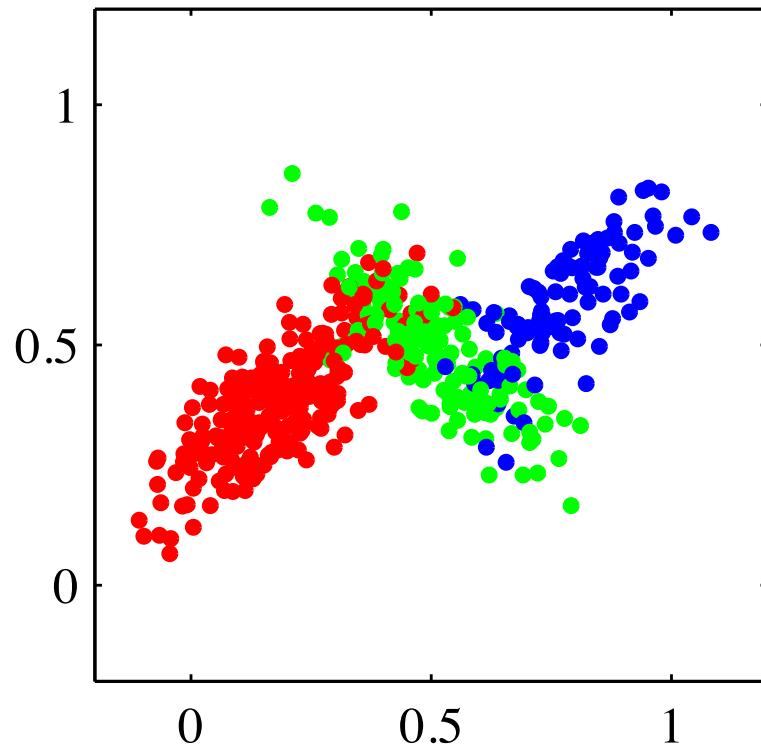
Clustering our Observations

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



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

$z^i \leftarrow \text{color of each point}$

- Then we have

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

if we had z^i , estimating

$\mu_{\text{blue}}, \Sigma_{\text{blue}}$ would be easy

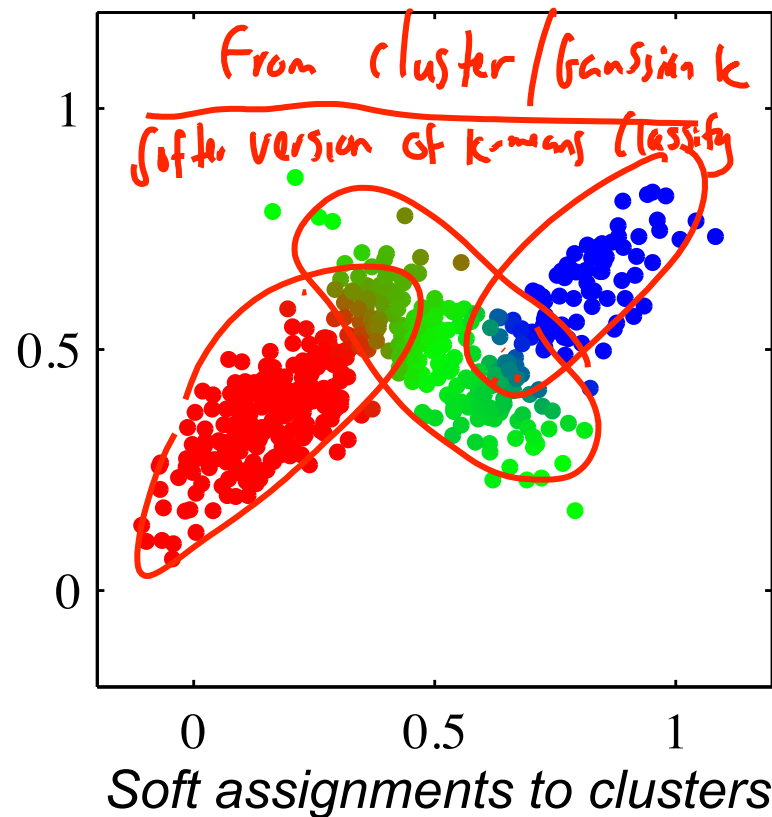
Clustering our Observations

the same dist but likely to be wider

- We must infer the cluster assignments from the observations

r_{ik} ← how much point i comes

"responsibilities"



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

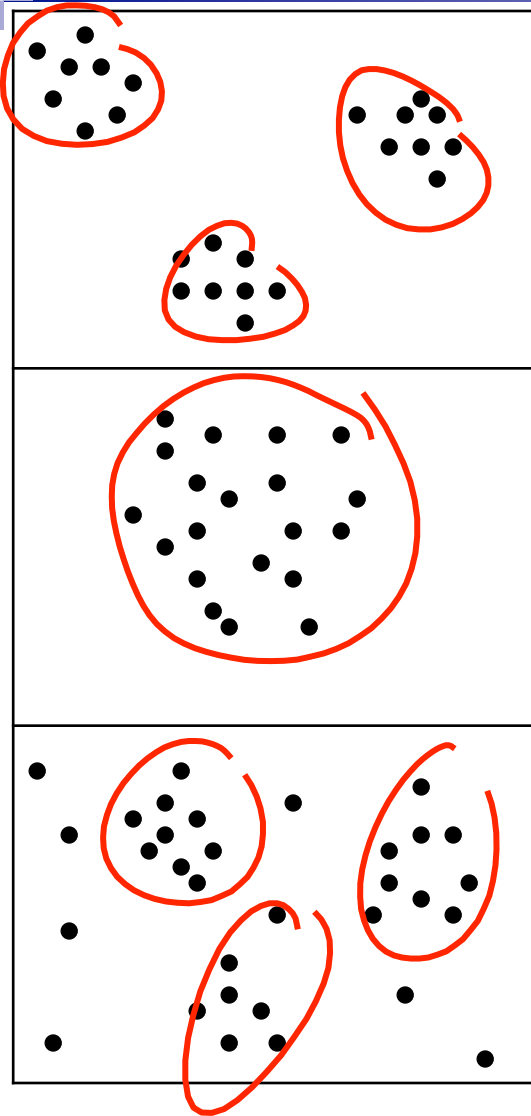
$$r_{ij} = p(z^i = j | x^i, \pi, \mu, \Sigma) =$$

$$= \frac{\pi_j N(\mu_j, \Sigma_j)}{\sum_{l=1}^K \pi_l N(\mu_l, \Sigma_l)}$$

normalize:

$$\sum_j r_{ij} = 1$$

Unsupervised Learning: not as hard as it looks



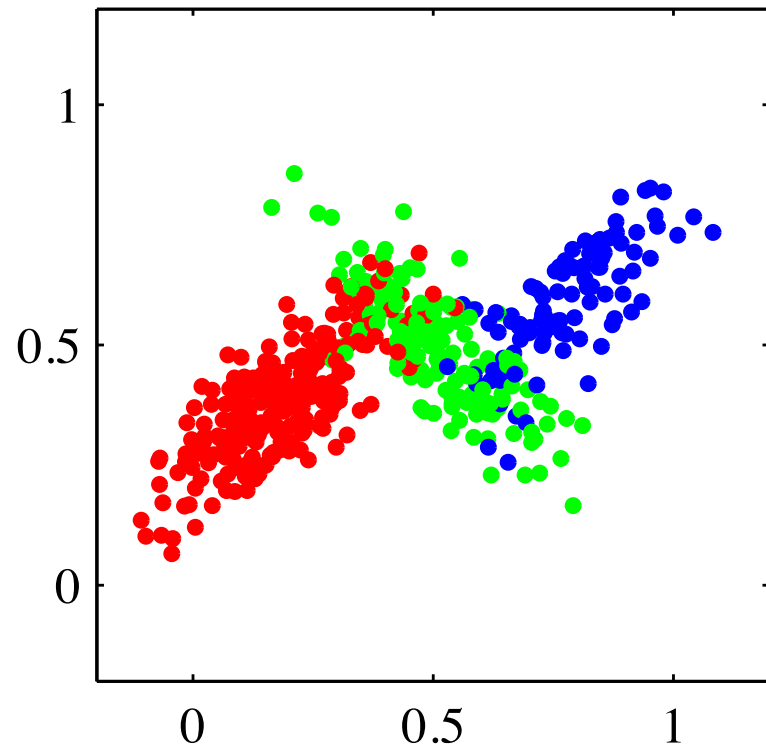
Sometimes easy

Sometimes impossible

and sometimes in between

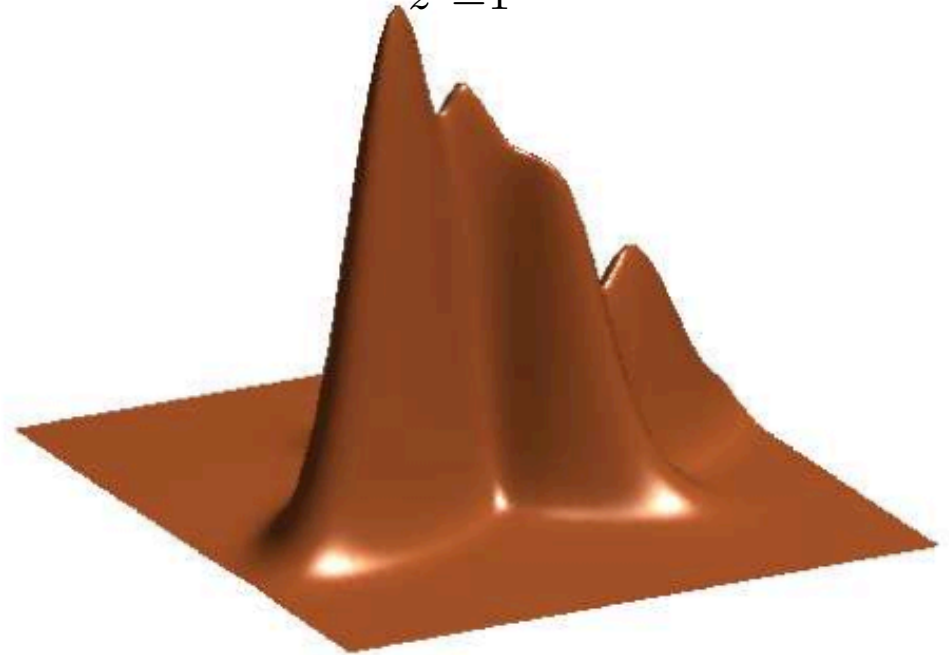
Summary of GMM Concept

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



*Complete data labeled
by true cluster assignments*

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



*Surface Plot of Joint Density,
Marginalizing Cluster Assignments*

Summary of GMM Components

- Observations

$$x^i \in \mathbb{R}^d, \quad i = 1, 2, \dots, N$$

- Hidden cluster labels $z_i \in \{1, 2, \dots, K\}, \quad i = 1, 2, \dots, N$

- Hidden mixture means $\mu_k \in \mathbb{R}^d, \quad k = 1, 2, \dots, K$

- Hidden mixture covariances $\Sigma_k \in \mathbb{R}^{d \times d}, \quad k = 1, 2, \dots, K$

- Hidden mixture probabilities $\pi_k, \quad \sum_{k=1}^K \pi_k = 1$

Gaussian mixture marginal and conditional likelihood :

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

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



Kmeans
MoG

A vertical red line runs down the left side of the slide, starting from the top of the blue header area and extending past the bottom of the text area.

Application to Document Modeling

Machine Learning – CSEP546

Carlos Guestrin

University of Washington

February 18, 2014

Cluster Documents

- Cluster documents based on topic



Document Representation

- Bag of words model



ignore
order of words

word Count

$$X = \begin{bmatrix} wc_1 \\ wc_2 \\ \vdots \\ wc_d \end{bmatrix}$$

$d = |V|$ size of vocab

Issues with Document Representation

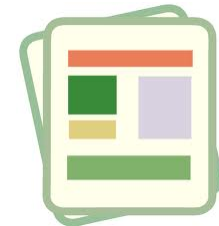
- Words counts are **bad** for standard similarity metrics



"Obama"
went to the
market

more important word in similarity
metric

rare words are more
important.



market went to
hell.

- Term Frequency – Inverse Document Frequency (tf-idf)
 - Increase importance of rare words

TF-IDF

← penalize frequent words

$$x^d = \begin{bmatrix} tfidf_1 \\ tfidf_2 \\ \vdots \\ tfidf_v \end{bmatrix}$$

vocab.size

■ Term frequency:

term doc

$$tf(t, d) = \# \text{ of occurrences of term } t \text{ in doc } d$$

occurs/not

decreases impact of repeated words

- Could also use $\{0, 1\}$, $1 + \log(tf(t, d) + 1)$.

■ Inverse document frequency:

set of documents

20 for popular words

$$idf(t, D) = \log \frac{|D|}{1 + |\{d \in D : t \in d\}|}$$

docs containing t

Normalize for long docs

→ divide by norm

→ $\frac{tf(t, d)}{\max_t \{tf(t, d)\}}$

■ tf-idf:

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D)$$

- High for document d with high frequency of term t (high “term frequency”) and few documents containing term t in the corpus (high “inverse doc frequency”)

A Generative Model

- Documents: d_1, \dots, d_n
- Associated topics: $\pi_1, \dots, \pi_k \leftarrow \text{sample topic}$
- Parameters simple mixture of Gaussians: $\downarrow j$

$$x^{d_i} = \begin{bmatrix} t_{fidf} \end{bmatrix} \sim N(\mu_j, \bar{\Sigma}_j)$$

$$t_{fidf} \geq 0$$

$$\text{Gaussians} \in \mathbb{R}$$

more typically use
dist. positive
countable

What you get from mixture model for documents

- Words give topic: $P(\text{word} | \text{topic})$ \leftarrow topic model
topic = 5 (Sports)
 $\rightarrow P(\text{ball} | \text{topic} = 5)$ is high ; $P(\text{Comp. Scientist} | \text{topic} = 5)$ low
- Topic proportions:
 $\pi_1, \pi_2 \dots \pi_k \leftarrow$ likely doc \rightarrow topic 1
?
:
k
- Topic distribution of each document:

Results from Wikipedia data $k=15$ topics

using similar model (LDA) $P(\text{words} / \text{topic})$



MLE for mixture models
for unsupervised learning

Expectation Maximization

Machine Learning – CSEP546

Carlos Guestrin

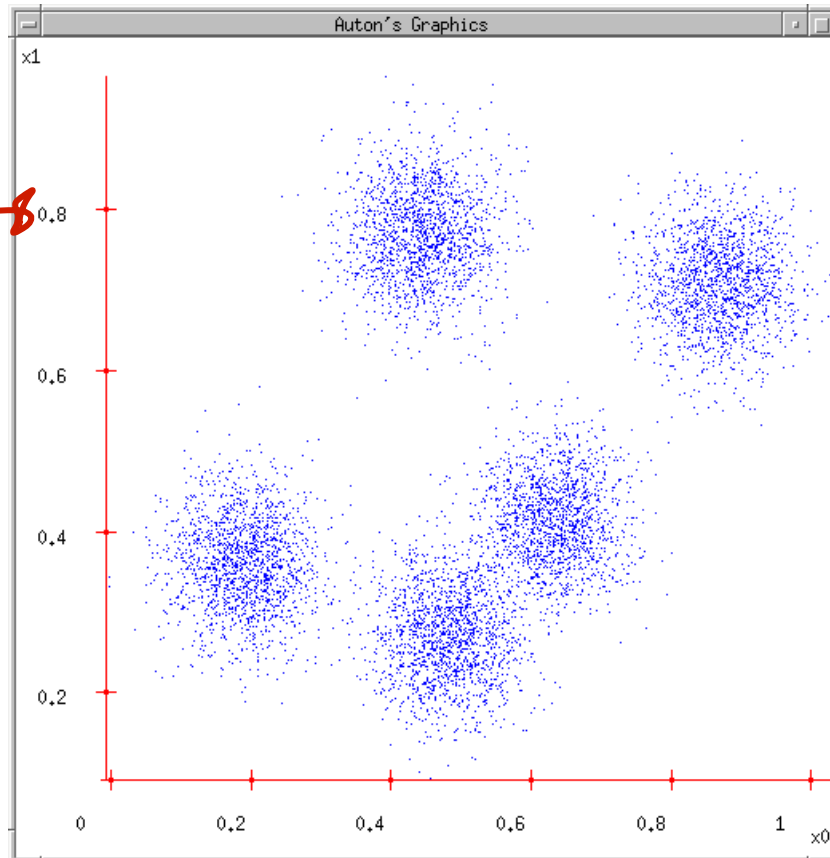
University of Washington

February 18, 2014

Next... back to Density Estimation

What if we want to do density estimation with multimodal or clumpy data?

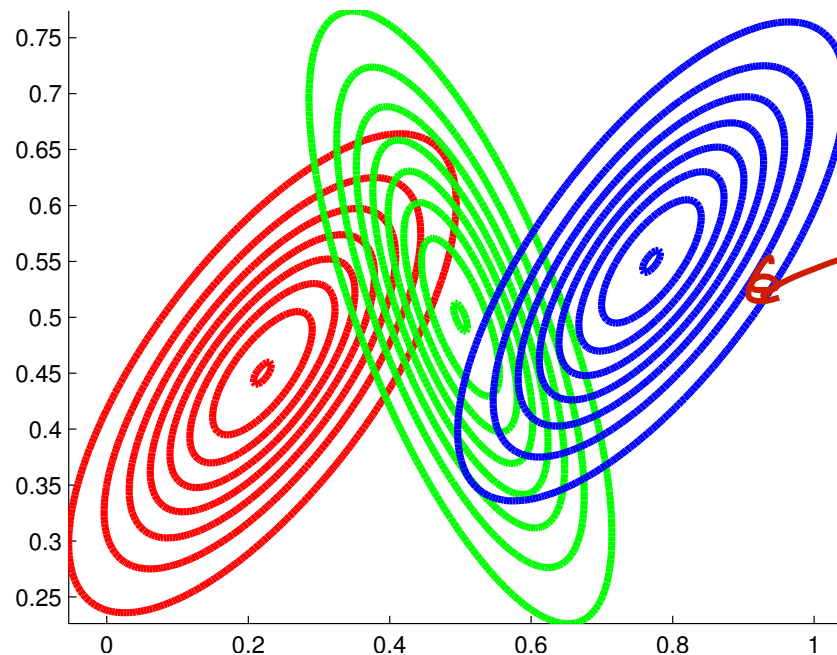
Goal: fit MoG model



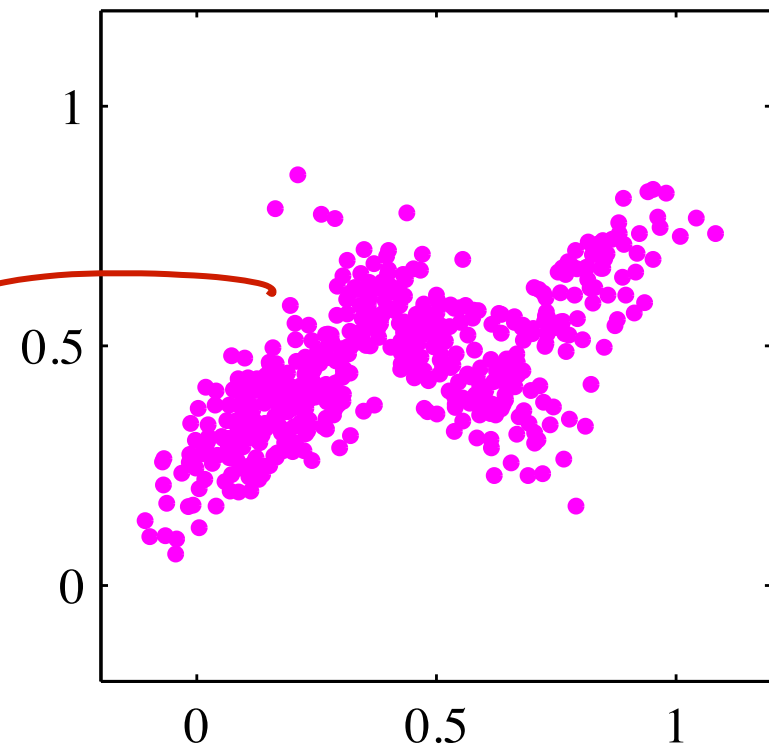
Learning Model Parameters

- Want to learn model parameters

Mixture of 3 Gaussians



Our actual observations



ML Estimate of Mixture Model Params

Don't observe $z^i \leftarrow$ cluster assignment

- Log likelihood

$$\max_{\theta=\{\mu, \Sigma\}} L_x(\theta) \triangleq \log p(\{x^i\} \mid \theta) = \sum_{i=1}^n \log \sum_{z^i} p(x^i, z^i \mid \theta)$$

$z^i \leftarrow$ arg unknown values

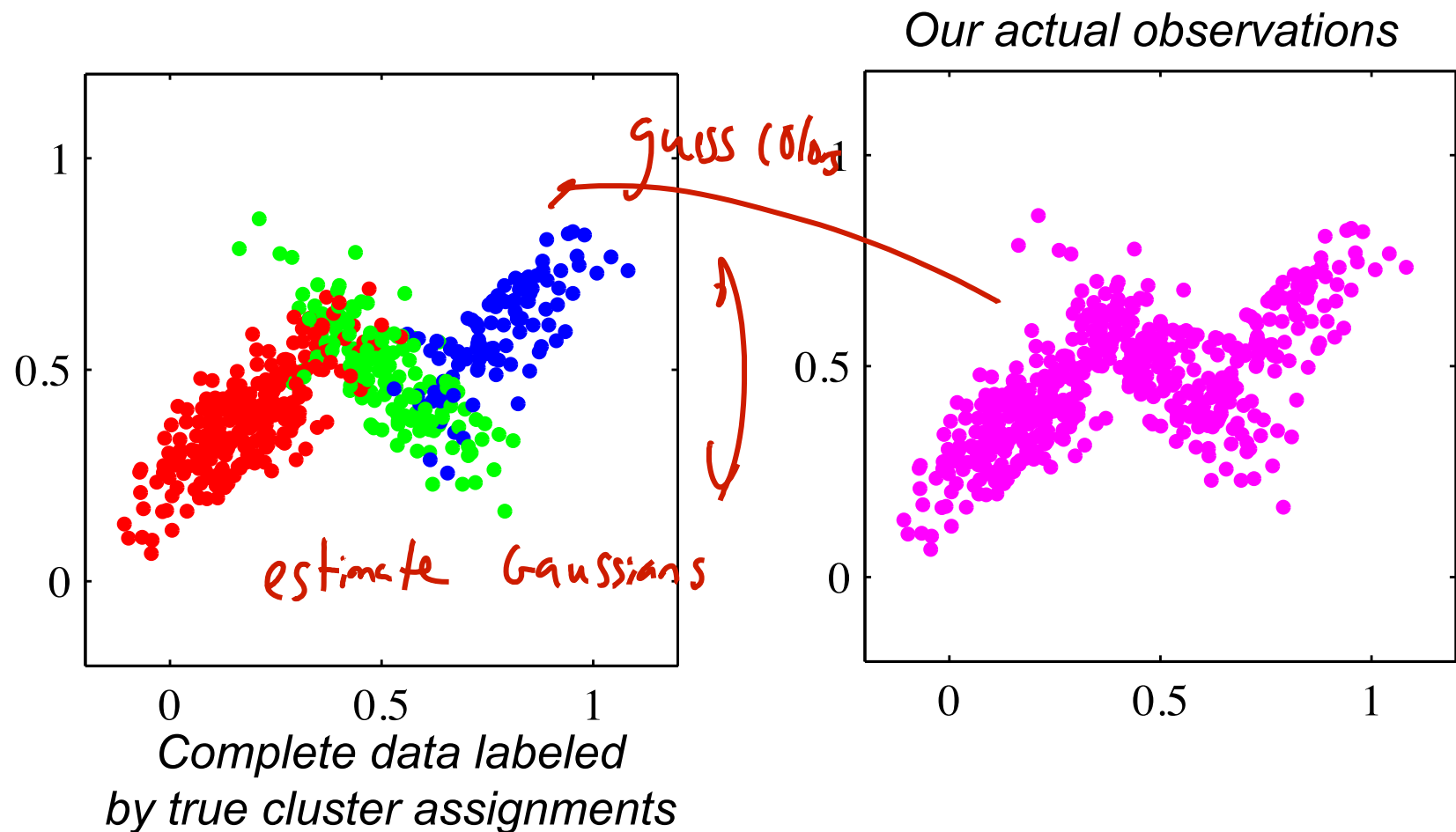
- Want ML estimate

$$\hat{\theta}^{ML} = \underset{\theta=\{\mu, \Sigma\}}{\operatorname{argmax}} L_x(\theta)$$

- Neither convex nor concave and local optima

Complete Data

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



If “complete” data were observed...

- Assume class labels z^i were observed in addition to x^i

$$L_{x,z}(\theta) = \sum_i \log p(x^i, z^i | \theta)$$

Don't know z^i , estimate
using responsibilities r_{ij}

- Compute ML estimates
 - Separates over clusters k !

$$\pi_j = \frac{\text{count}(z^i = j)}{N}$$

- Example: mixture of Gaussians (MoG) $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$

Take points in cluster j ($z^i = j$)

$\mu_j = \text{avg of points}$

Σ_j is covariance

Cluster Responsibilities

$$N(x^i; \mu_j, \Sigma_j)$$

$$p(x^i | \mu_j, \Sigma_j) = \dots e^{-\frac{1}{2}(x^i - \mu_j)^T \Sigma_j^{-1} (x^i - \mu_j)}$$

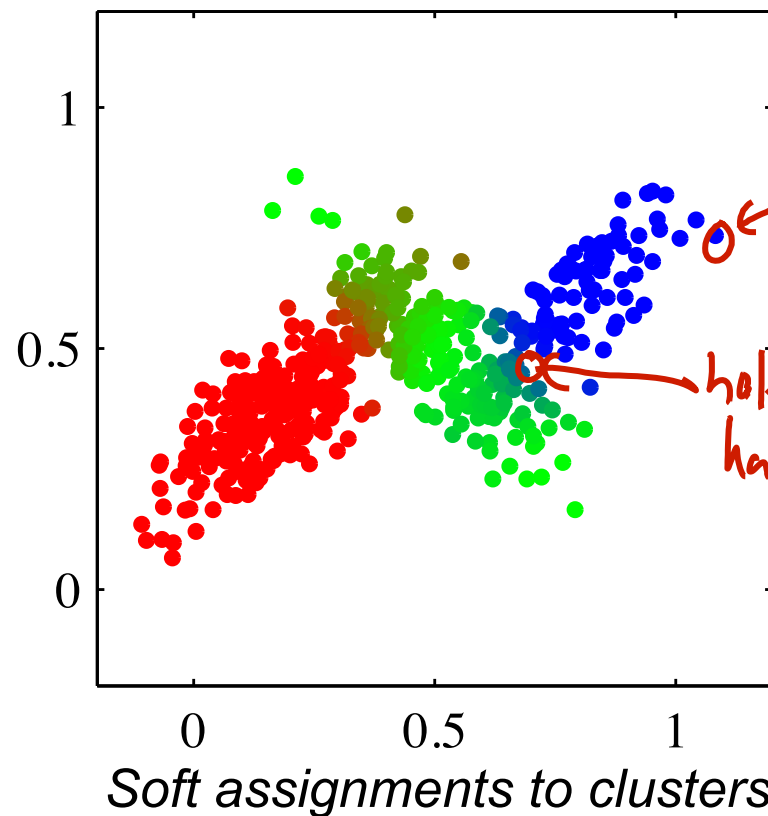
- We must infer the cluster assignments from the observations

Given Π, μ, Σ

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

$$r_{ij} = p(z^i = j | x^i, \pi, \phi) =$$

$$\frac{\pi_j N(x^i; \mu_j, \Sigma_j)}{\sum_k \pi_k N(x^i; \mu_k, \Sigma_k)}$$



Iterative Algorithm

in a soft way using r_{ij}

- Motivates a coordinate ascent-like algorithm:

1. Infer missing values z^i given estimate of parameters $\hat{\theta} = \{\pi, \Sigma, \mu\}$
2. Optimize parameters to produce new $\hat{\theta}$ given "filled in" data z^i
3. Repeat

- Example: MoG

1. Infer "responsibilities"

$$r_{ij}^{(t)} = p(z^i = j \mid x^i, \hat{\theta}^{(t-1)}) =$$

$$\frac{\pi_j^{(t-1)} N(x^i : \mu_j^{(t-1)}, \Sigma_j^{(t-1)})}{\sum_l \pi_l^{(t-1)} N(x^i : \mu_l^{(t-1)}, \Sigma_l^{(t-1)})}$$

2. Optimize parameters

max w.r.t. π_j :

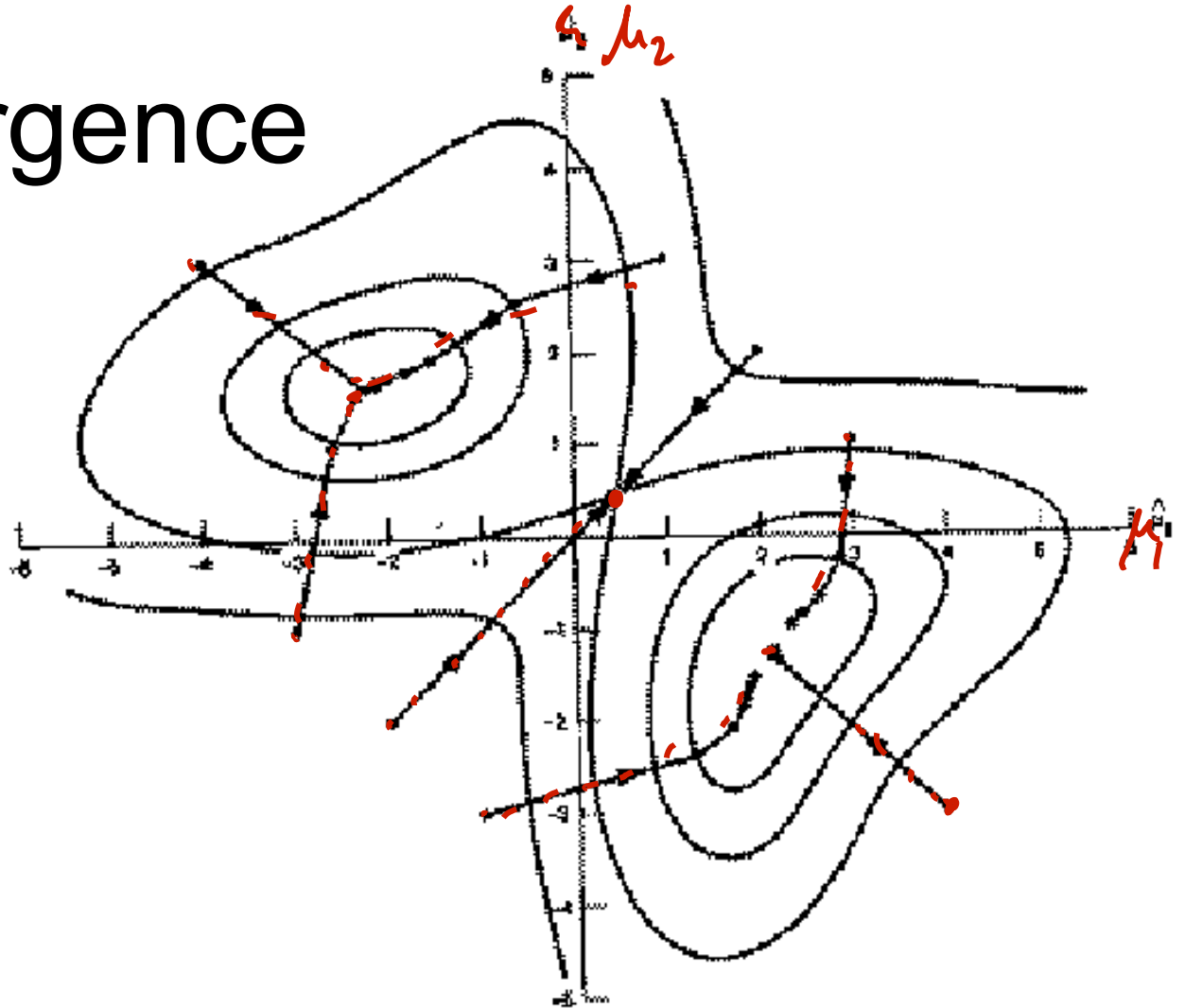
$$\pi_j^{(t)} = \frac{\sum_{i=1}^N r_{ij}^{(t)}}{N}$$

max w.r.t. μ_j, Σ_j :

$$\mu_j^{(t)} = \frac{\sum_{i=1}^N r_{ij}^{(t)} x^i}{\sum_{i=1}^N r_{ij}^{(t)}}$$

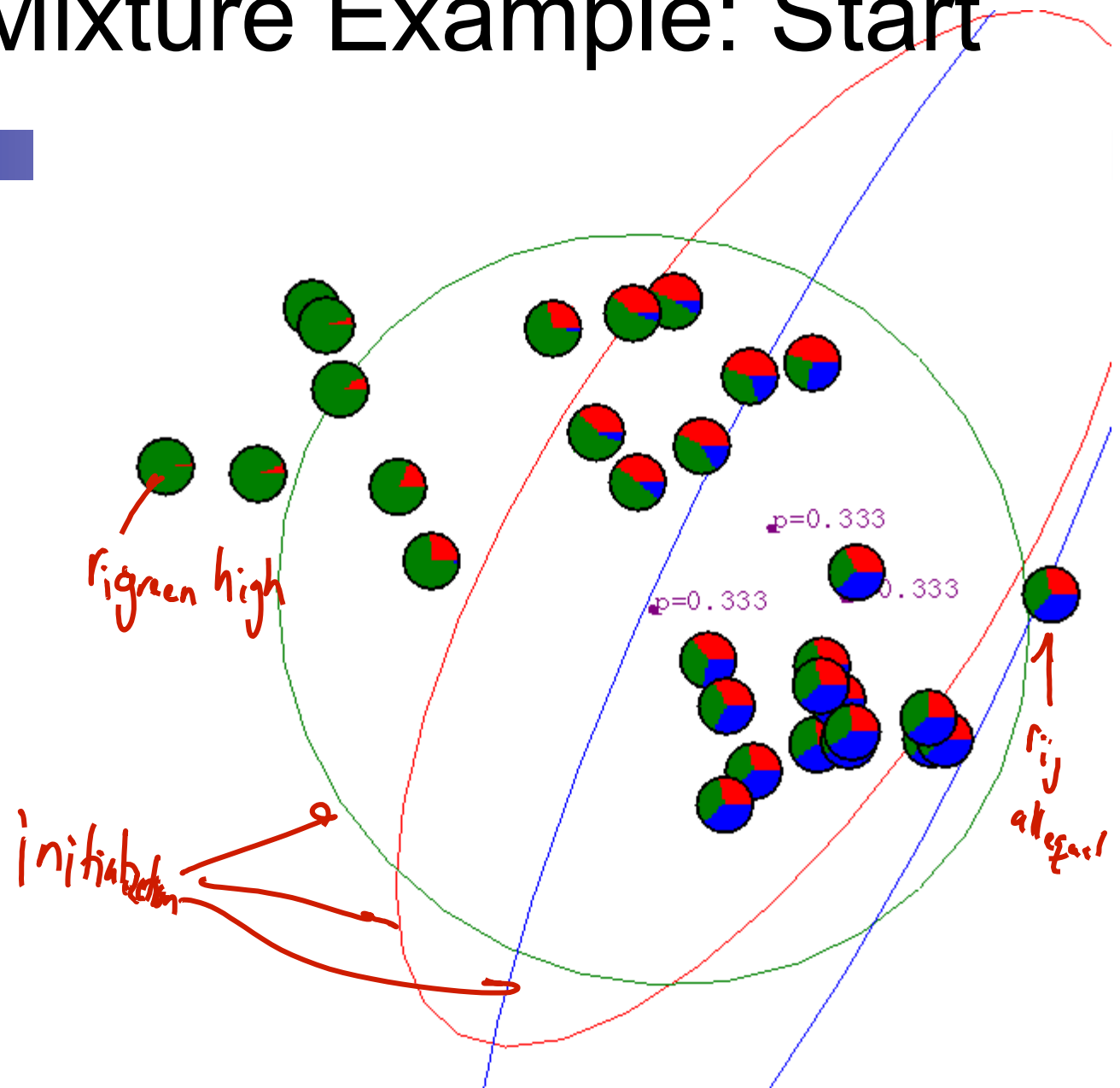
E.M. Convergence

- EM is coordinate ascent on an interesting potential function
- Coord. ascent for bounded pot. func. → convergence to a local optimum guaranteed

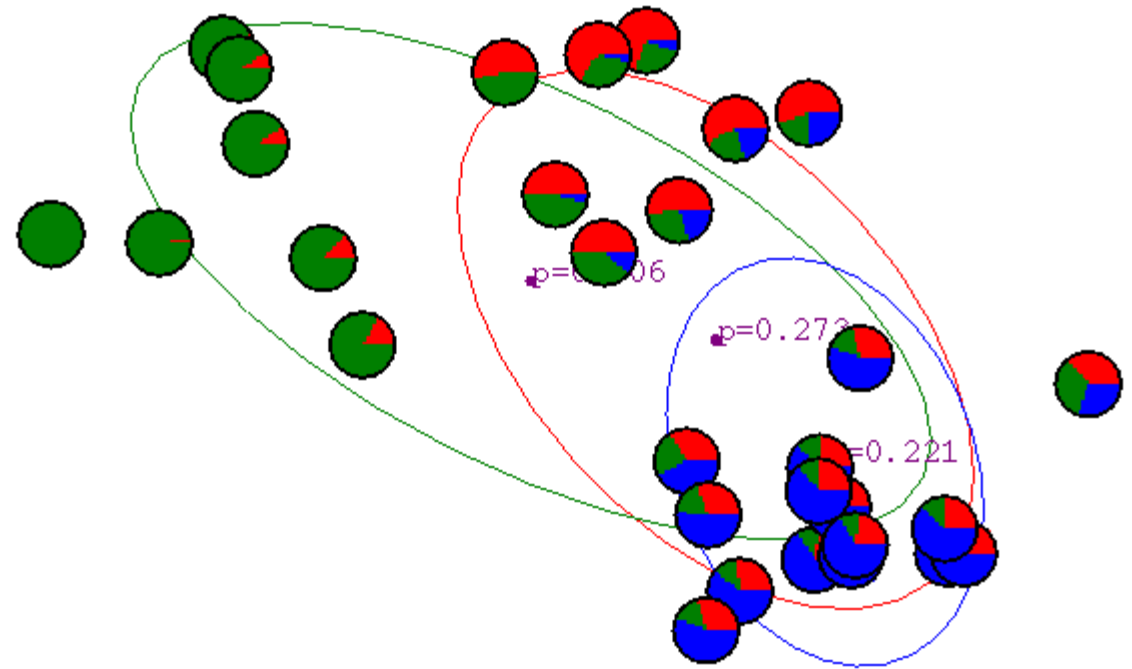


- This algorithm is REALLY USED. And in high dimensional state spaces, too. E.G. Vector Quantization for Speech Data

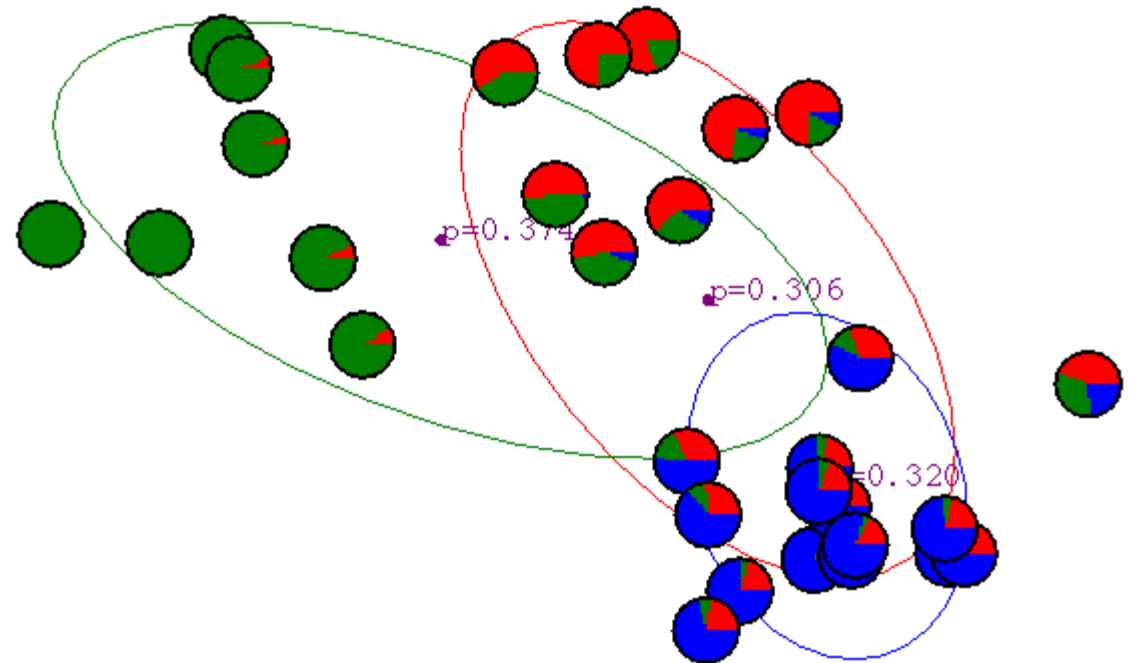
Gaussian Mixture Example: Start



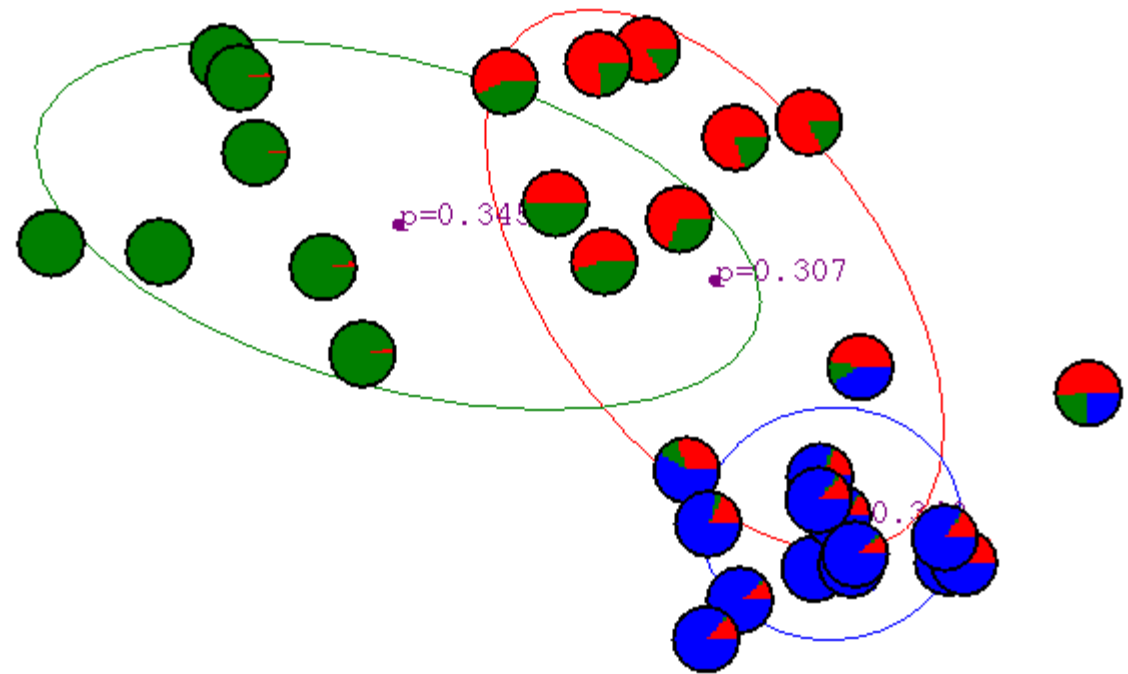
After first iteration



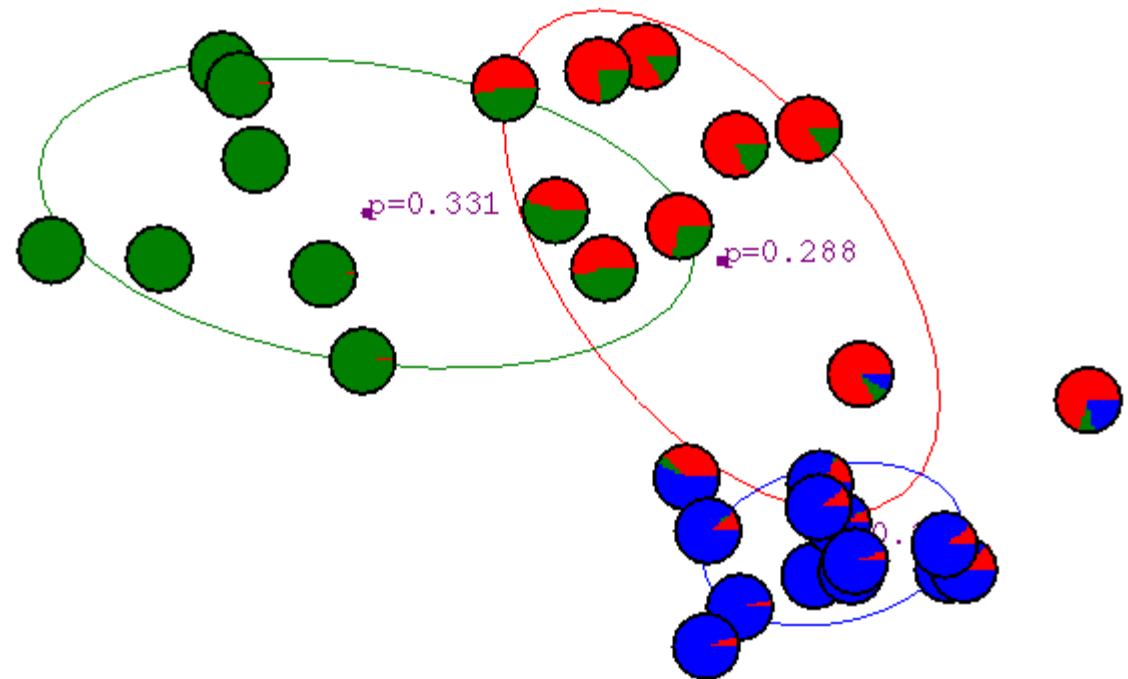
After 2nd iteration



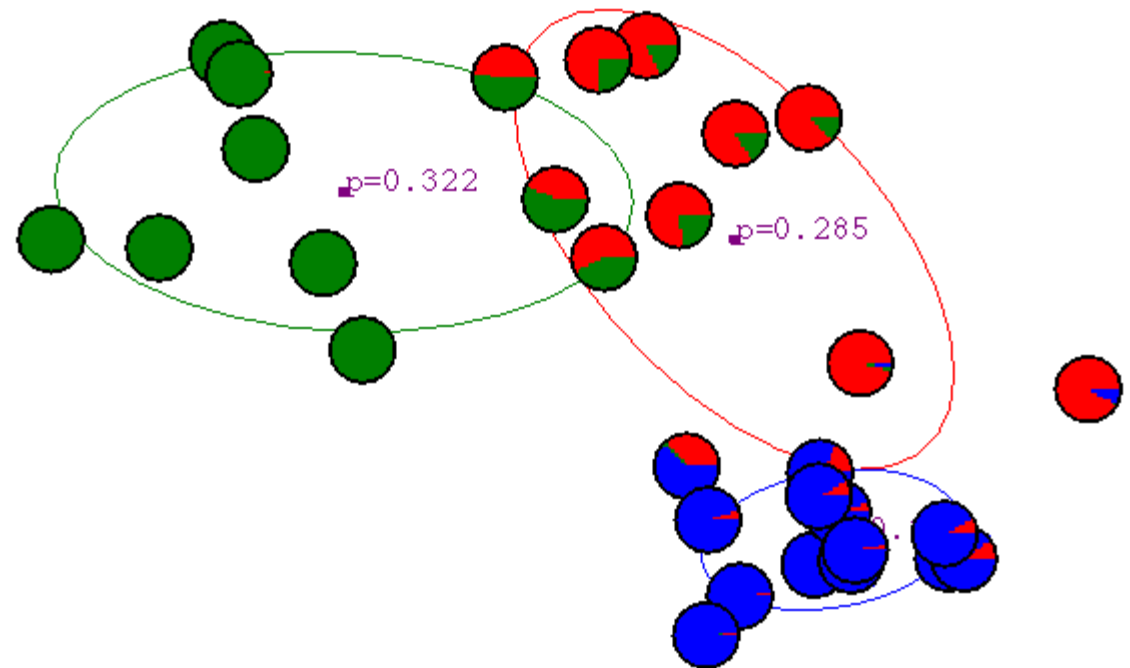
After 3rd iteration



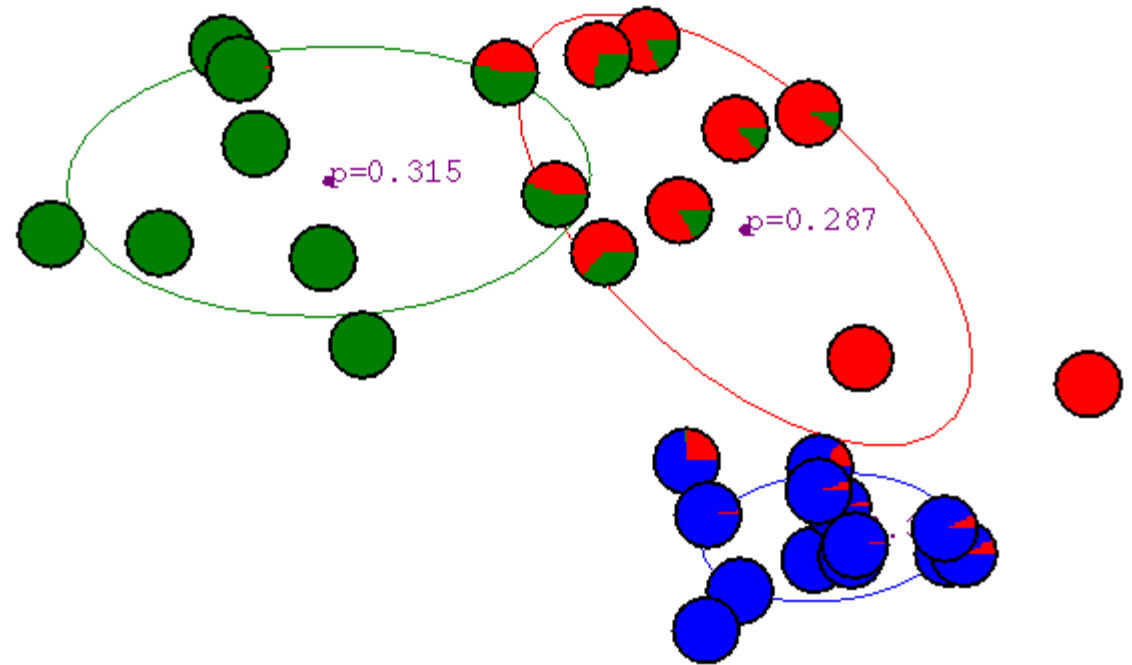
After 4th iteration



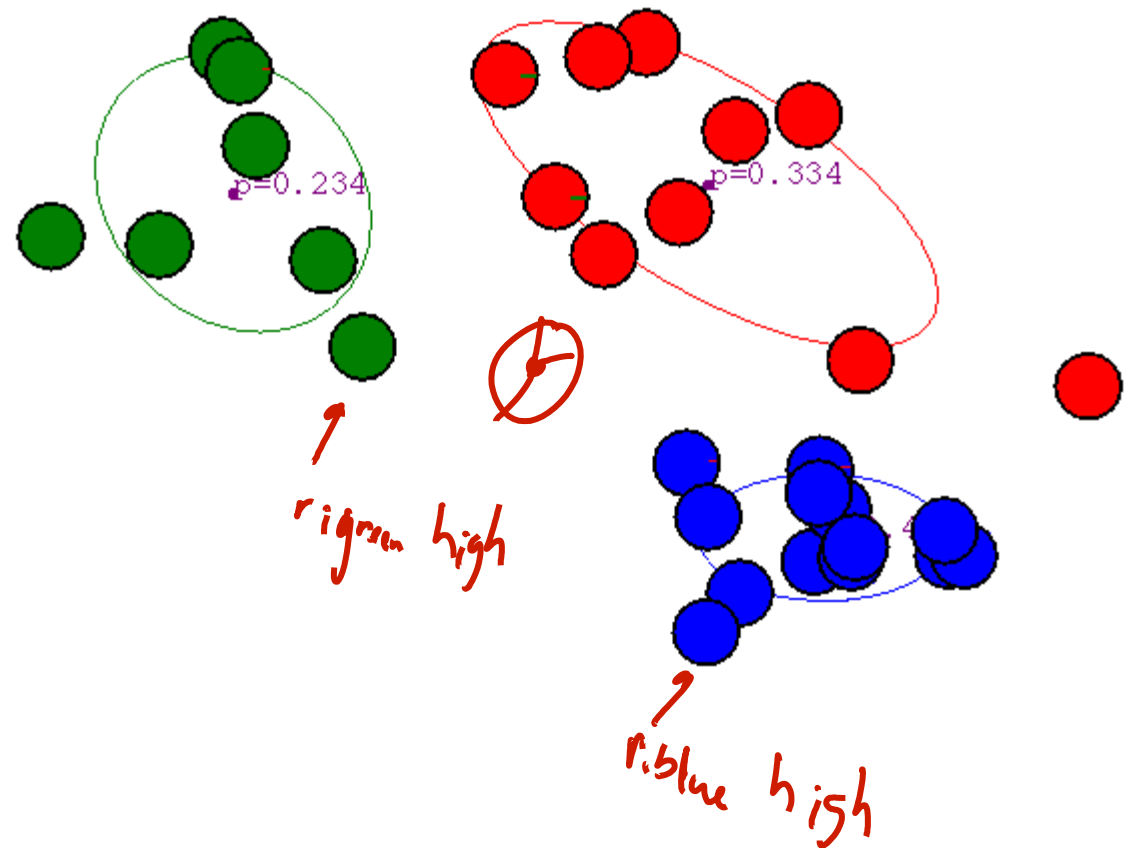
After 5th iteration



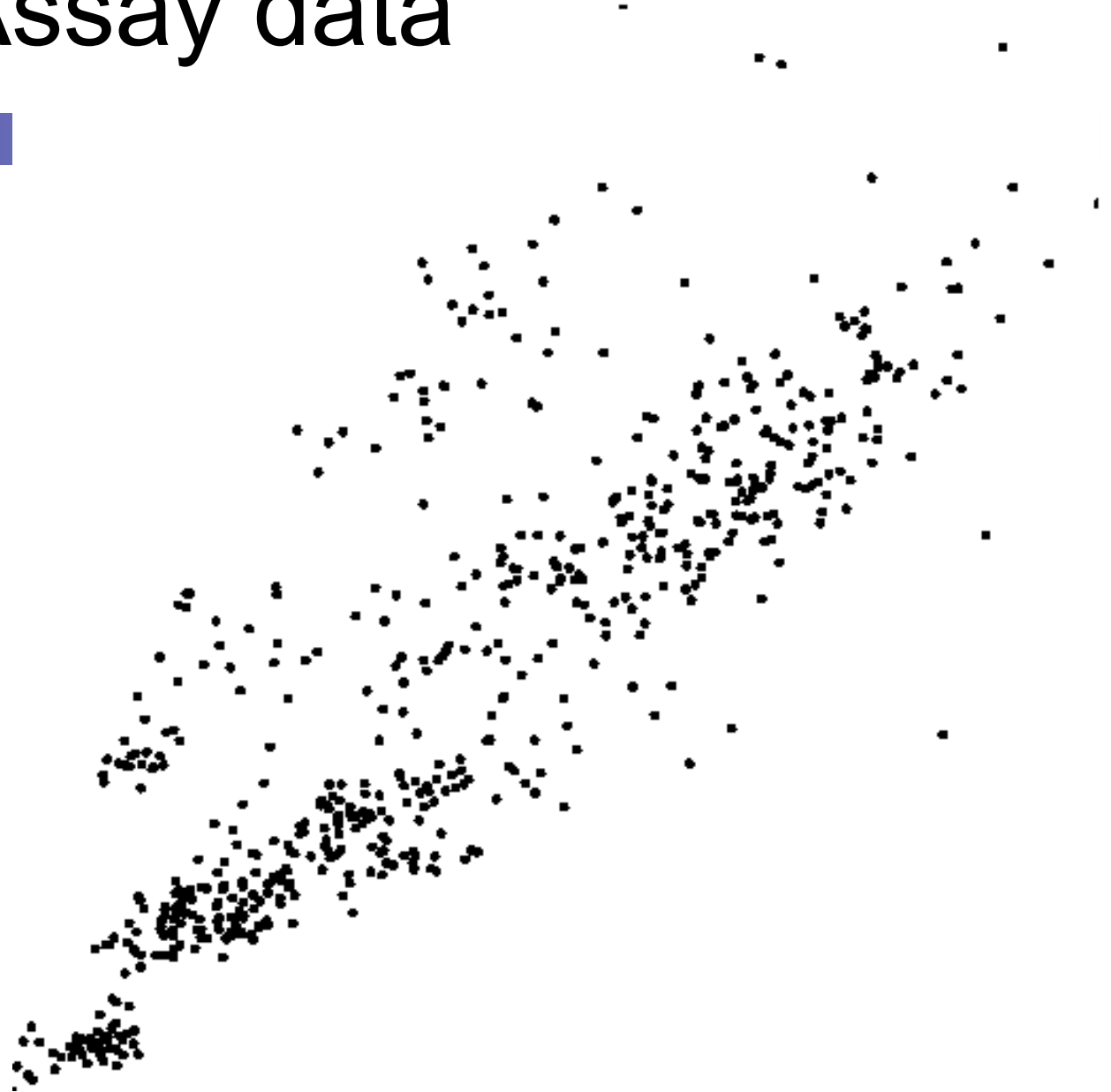
After 6th iteration



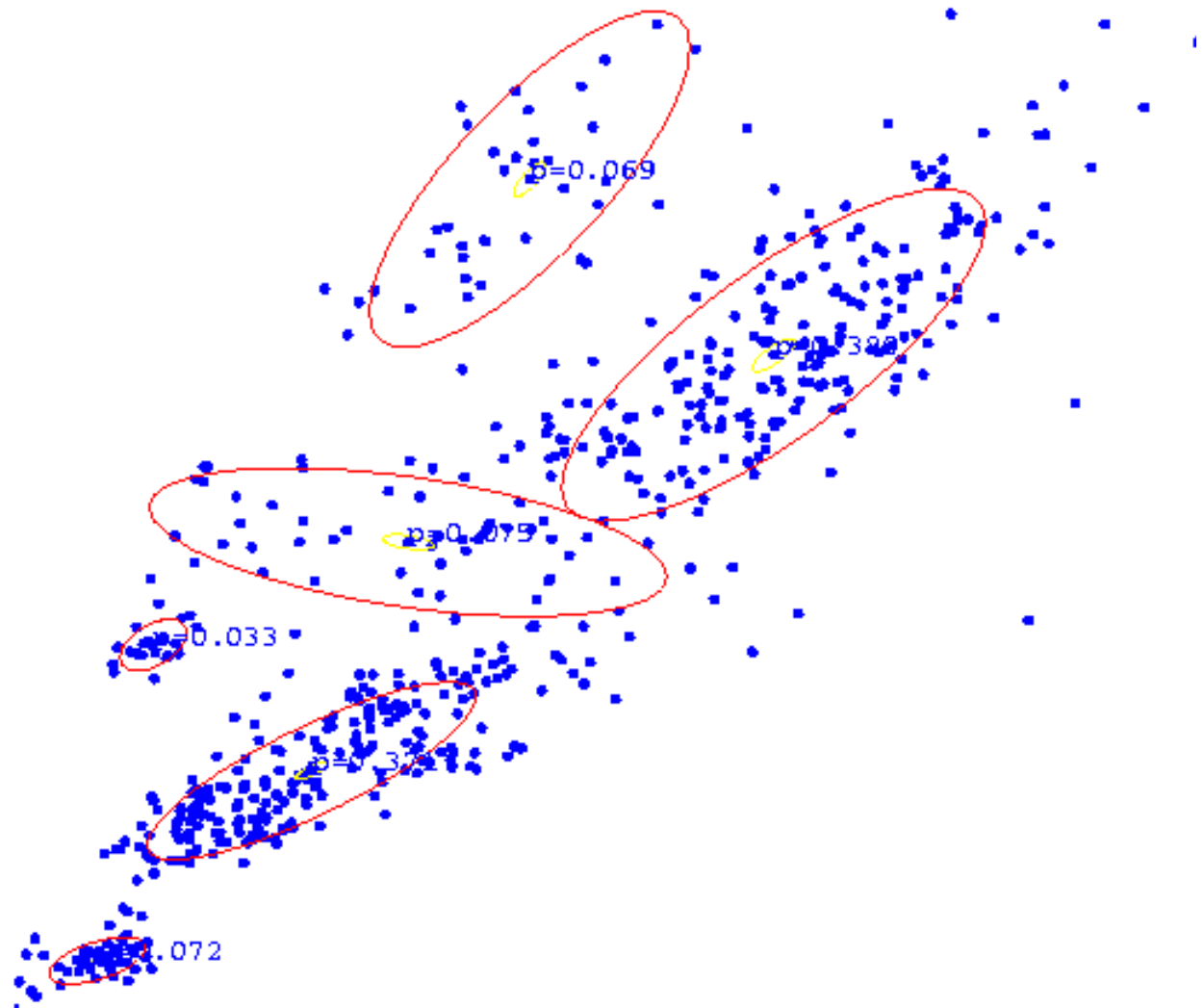
After 20th iteration



Some Bio Assay data



GMM clustering of the assay data





Resulting Density Estimator

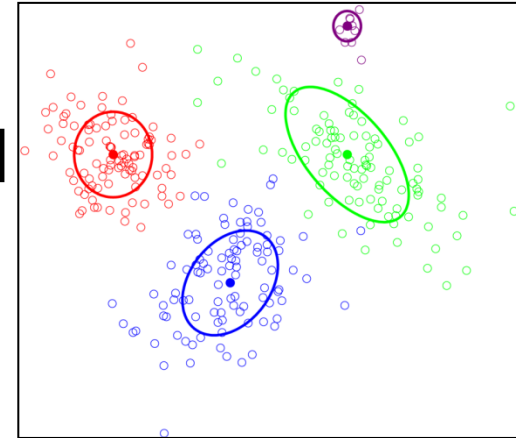


Initialization

- In mixture model case where $y^i = \{z^i, x^i\}$ there are many ways to initialize the EM algorithm
- Examples:
 - Choose K observations at random to define each cluster. Assign other observations to the nearest “centroid” to form initial parameter estimates
 - Pick the centers sequentially to provide good coverage of data
 - Grow mixture model by splitting (and sometimes removing) clusters until K clusters are formed
- Can be quite important to convergence rates in practice

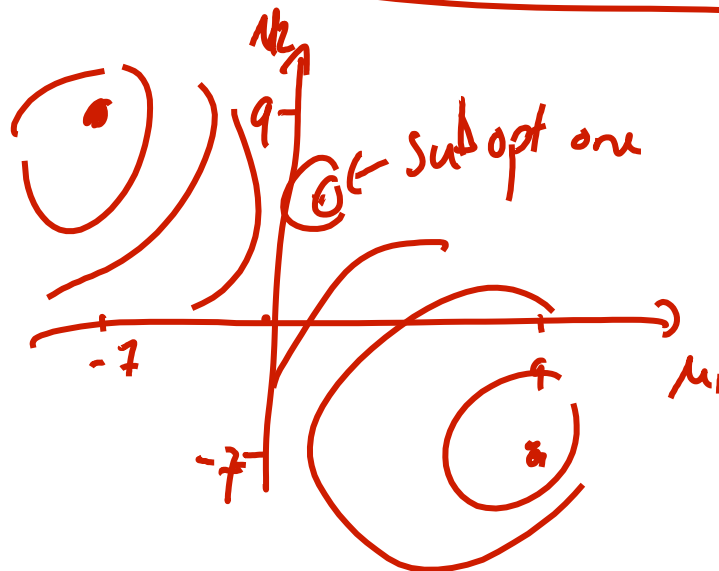
Label switching

- Color = label does not matter
- Can switch labels and likelihood is unchanged



$$\begin{array}{ll} \mu_1 = -7 & \mu_1 = 9 \\ \mu_2 = 9 & \mu_2 = -7 \end{array}$$

$$L_x(\mu_1, \mu_2)$$



What you should know



- K-means for clustering:
 - algorithm
 - converges because it's coordinate ascent
- EM for mixture of Gaussians:
 - How to “learn” maximum likelihood parameters (locally max. like.) in the case of unlabeled data
- Remember, E.M. can get stuck in local minima, and empirically it DOES
- EM is coordinate ascent