

Schedule for the rest of the quarter

- 11/25 (Today): Clustering and latent variable models
- • 11/27 (Thu): No class, happy Thanksgiving!
- • 12/2 (Tue): Guest lecture by Leo on bandits [not on exam]
- • 12/4 (Thu): Foundation models
- 12/8 (Mon): Final exam!

Clustering & Latent Variable Models

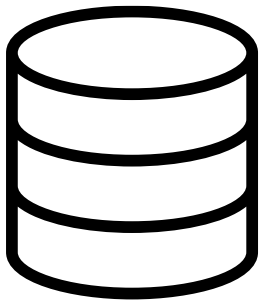
CSE 446/546

Sewoong Oh & Pang Wei Koh

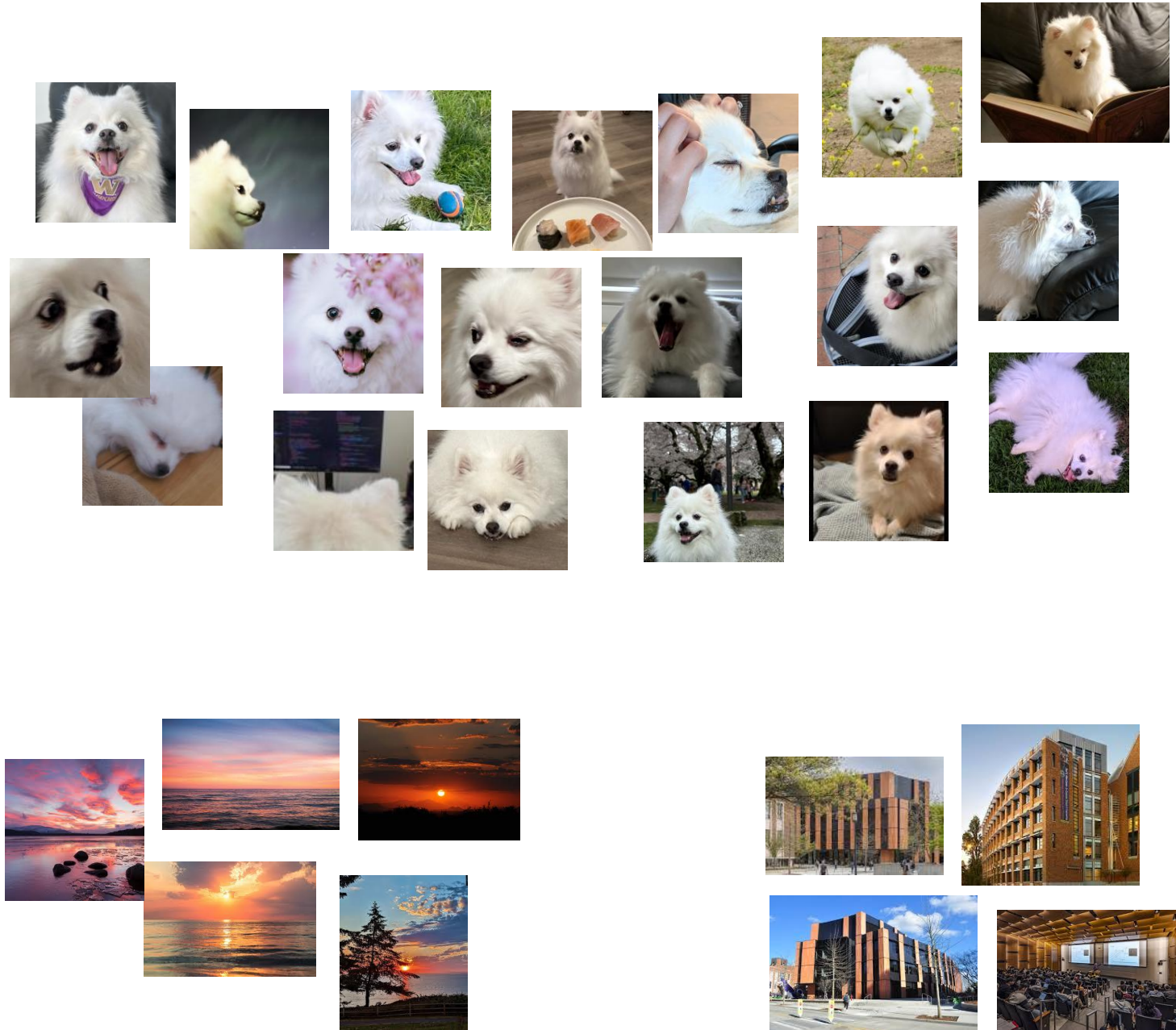
Clustering

- Fundamental problem in unsupervised ML
- Goal: Group “similar” data points into clusters

Clustering images



My photos



Clustering web search

The screenshot displays the Clustering Workbench interface. On the left, the 'Query' field contains the word 'vaccine'. The main area shows a treemap visualization of search results, with 'SARS-CoV2 (49)' and 'T Cell (27)' highlighted by red circles. The right sidebar lists search results, including 'COVID-19 VACCINES (69 docs, 14 subclusters)' and 'SARS-COV2 (49 docs, 14 subclusters)'. The top status bar indicates 250 results, 29 clusters, 84.4% clustered docs, and 66ms clustering time.

Clustering Workbench

Cluster

250 results, 29 clusters, 84.4% clustered docs, 66ms clustering time

Data source: PubMed

Clustering algorithm: Lingo3G

Query: vaccine

Max results: 250

API key

Parameters affecting the number, structure and content of clusters.

Minimum cluster size: 0

Maximum cluster size: 0.4

Clusters: list, treemap, pie-chart

Results: list

Export

Health Care (24)

Inactivation (10)

T Cell (27)

P.1 (8)

HIV-1 (4)

Strains (18)

Species (7)

Review (15)

Testing (15)

Side Effects (6)

United States (4)

Vaccine Acceptance (3)

B. 1.351 (2)

Public Institutions (2)

Main Protease (2)

Children (2)

SARS-CoV-2 (19)

Antibodies (14)

Review (17)

Patients (60)

Cases (18)

Immune Responses (13)

Hong Kong (3)

Streptococcus Pneumoniae (5)

DNA (8)

December 2019 (3)

FoamTree

COVID-19 VACCINES (69 docs, 14 subclusters)

- Infection (27)
- Review (15)
- Coronavirus Disease 2019 (15)
- Testing (15)
- mRNA Vaccines (10)
- Participants (8)
- Side Effects (6)
- Sources (6)
- United States (4)
- Vaccine Acceptance (3)
- B. 1.351 (2)
- Public Institutions (2)
- Main Protease (2)
- Children (2)

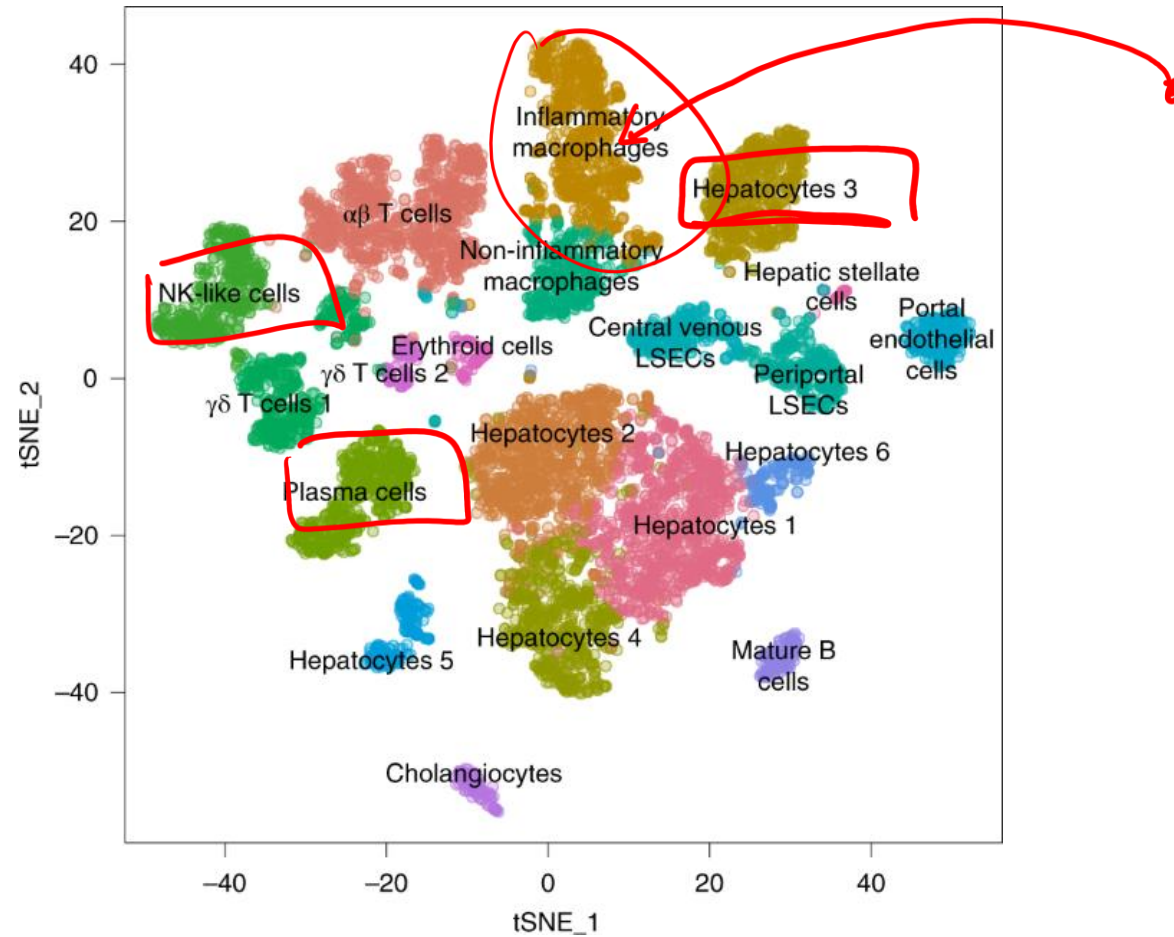
SARS-COV2 (49 docs, 14 subclusters)

- SARS-CoV-2 Variants (12)
- Review (12)
- Symptoms (12)
- Spike Protein (10)
- Natural Infection (5)
- Allergic Reaction (4)
- Vaccine Candidates (4)
- Cancer (4)
- Healthcare Workers (3)
- HIV (2)
- Asymptomatic Infection (2)
- H1N1 (2)
- Main Protease (2)
- Exposure to SARS-CoV-2 (2)

10 results in mRNA Vaccines

- Three Doses of an mRNA Covid-19 Vaccine in Solid-Organ Transplant Recipients. The New England journal of medicine, 2021 <https://www.ncbi.nlm.nih.gov/pubmed/3416170X>
COVID-19 Vaccines mRNA Vaccines
- Primary, Recall, and Decay Kinetics of SARS-CoV-2 Vaccine Antibody Responses. ACS nano, 2021
Studies of two SARS-CoV-2 mRNA vaccines suggested that they yield ~95% protection from symptomatic infection at least short-term, but important clinical questions remain. It is unclear how vaccine-induced antibody levels quantitatively compare to the wide spectrum induced by natural SARS-CoV-2 infection. Vaccine response kinetics and magnitudes in persons with prior COVID-19 compared to virus-naï...
KEYWORDS: SARS-CoV-2, anti-RBD antibodies, humoral immunity, mRNA nanoparticle vaccine, vaccine response durability <https://www.ncbi.nlm.nih.gov/pubmed/3415978>
COVID-19 Vaccines Infection mRNA Vaccines SARS-CoV2 Natural Infection
- Correspondence: Humoral immune response to COVID-19 mRNA vaccine in patients with multiple sclerosis treated with high-efficacy disease-modifying therapies. Therapeutic advances in neurological disorders. 2021

Clustering cell types



Clustering

- This lecture, we'll study two clustering methods:
 1. K-means
 2. Mixture of Gaussians

K-means

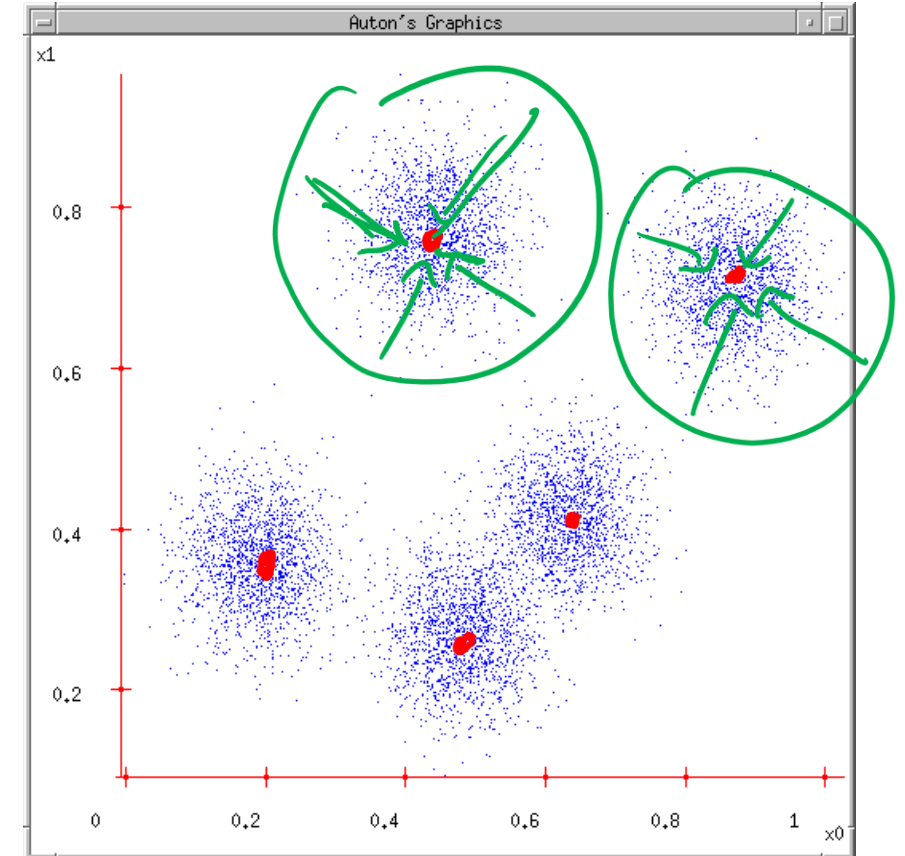
- Goal: Find cluster centers & cluster assignments that minimize the average squared distance between each point and its cluster center

for each cluster i for each data point x_j assigned to cluster i

$$\operatorname{argmin}_{\mu, C} \sum_{i=1}^k \sum_{j: C(j)=i} \|\mu_i - x_j\|_2^2$$

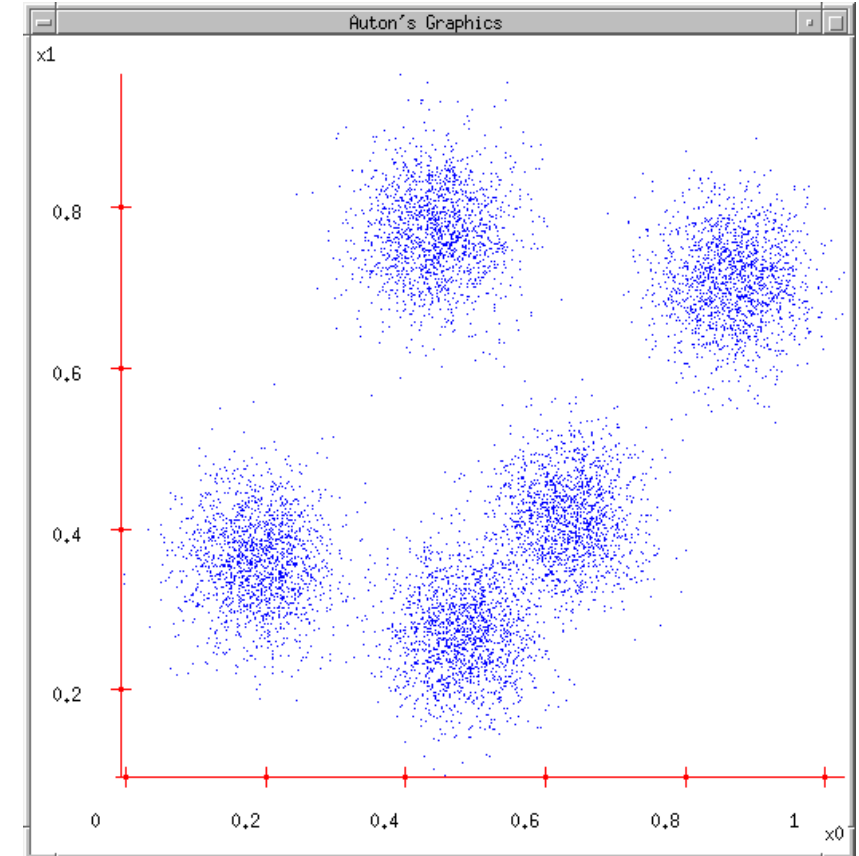
cluster assignment cluster center

$C(j) = i$ means x_j assigned to cluster i



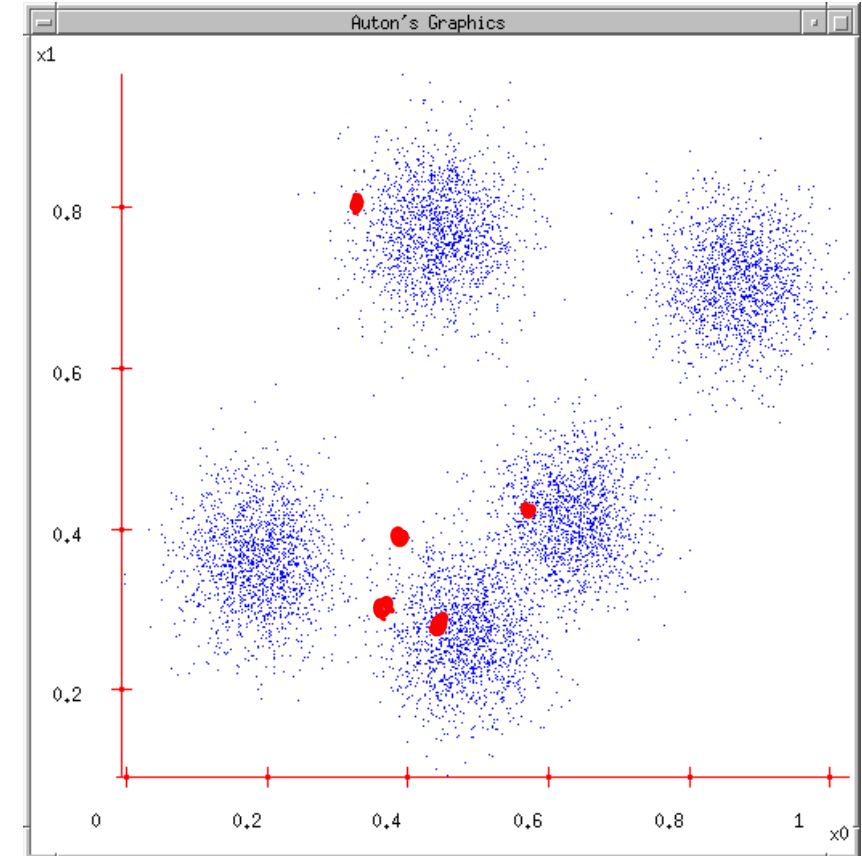
K-means

1. Ask user for # of clusters (e.g., $k=5$)



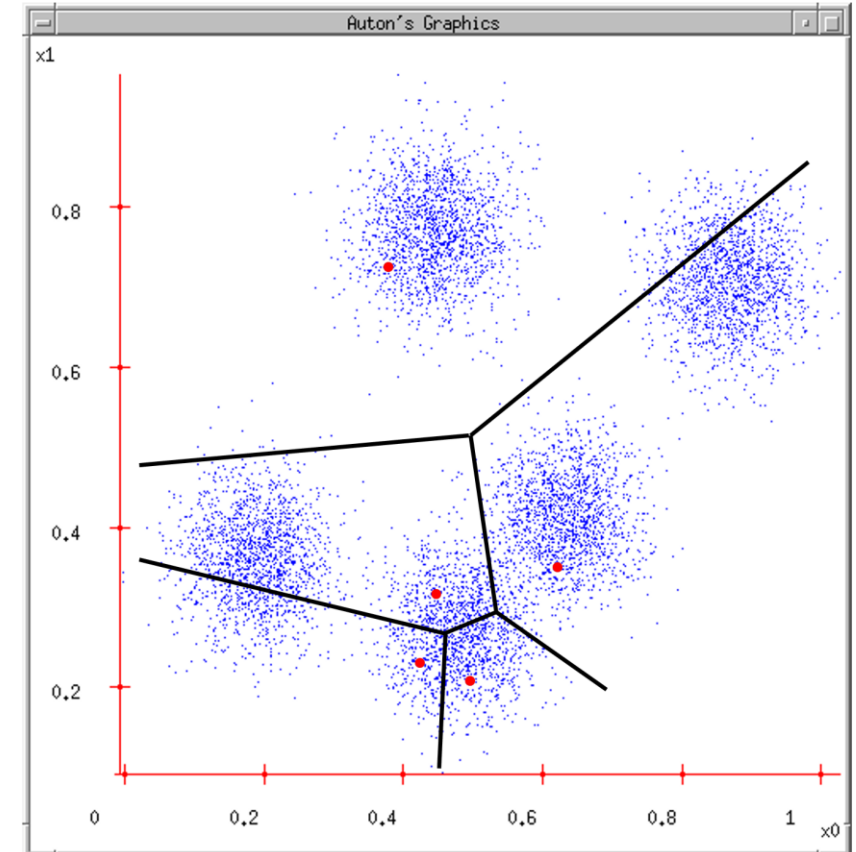
K-means

1. Ask user for # of clusters (e.g., $k=5$)
2. Randomly guess k cluster centers



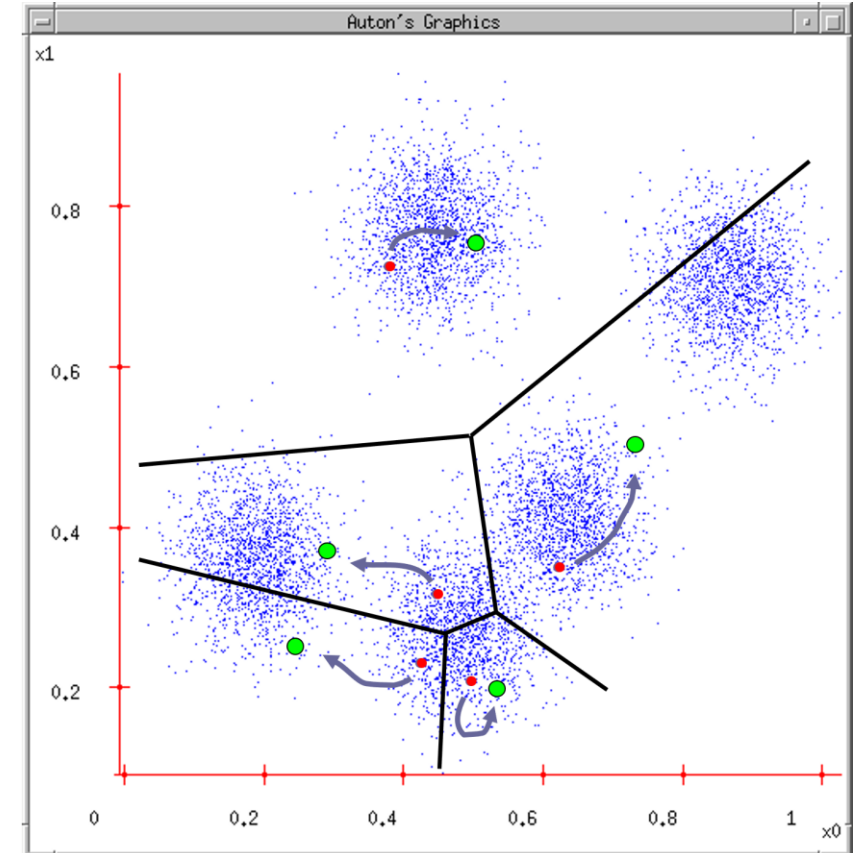
K-means

1. Ask user for # of clusters (e.g., $k=5$)
2. Randomly guess k cluster centers
3. Assign each data point to nearest cluster

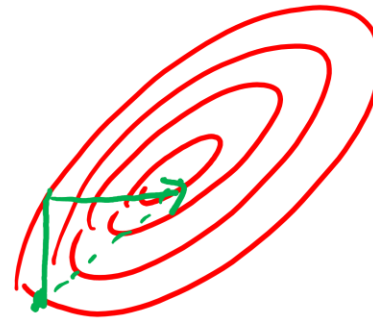


K-means

1. Ask user for # of clusters (e.g., $k=5$)
2. Randomly guess k cluster centers
3. Assign each data point to nearest cluster
4. Each center moves to centroid of points it "owns"
5. Repeat until terminated!



K-means as optimization



- Randomly initialize cluster centroids
- Classify: Assign each point x_j to nearest center

hold μ constant
optimizing C

repeat

$$C(j) \leftarrow \operatorname{argmin}_{i \in \{1, \dots, k\}} \|\mu_i^{(t)} - x_j\|_2^2$$

- Recenter: Each center becomes centroid of its points

holding C constant
optimizing μ

$$\mu_i^{(t+1)} \leftarrow \operatorname{argmin}_{\mu} \sum_{j: C(j)=i} \|\mu - x_j\|_2^2$$

average of all points x_j
assigned to i

$$\operatorname{argmin}_{\mu, C} \sum_{i=1}^k \sum_{j: C(j)=i} \|\mu_i - x_j\|_2^2$$

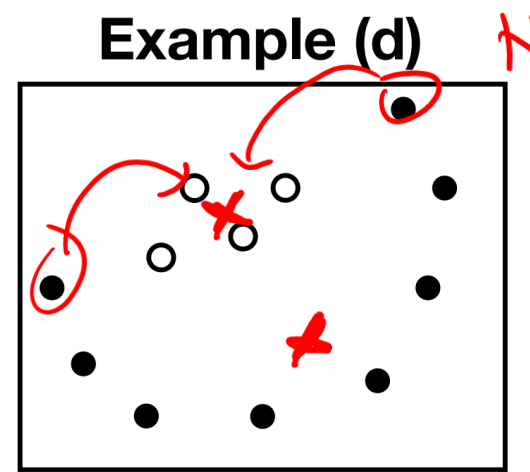
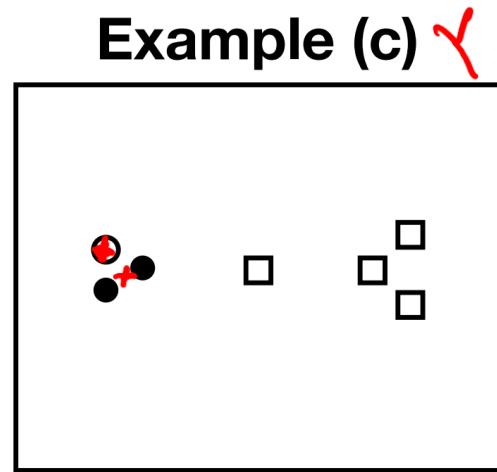
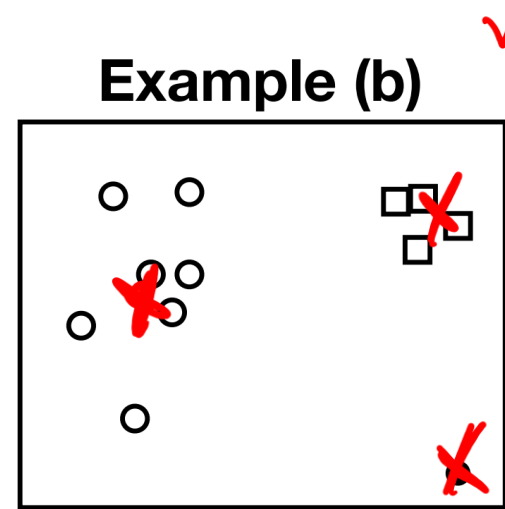
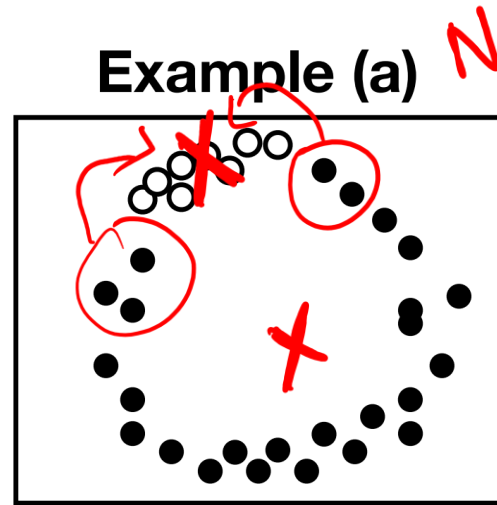
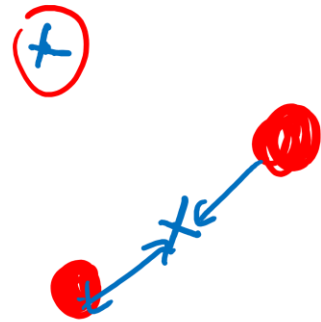
K-means is alternating minimization!

Does k-means converge? *Yes, in finite time*

$$\operatorname{argmin}_{\mu, C} \sum_{i=1}^k \sum_{j: C(j)=i} \|\mu_i - x_j\|_2^2 \geq 0$$

- After each iteration, objective always decreases or stays the same
- Does it terminate in finite time?
 - Finite set of values for cluster assignments k^n
 - Objective function decreases or optimization terminates
 - Objective is lower bounded by 0
 - Therefore, converges in at most k^n steps

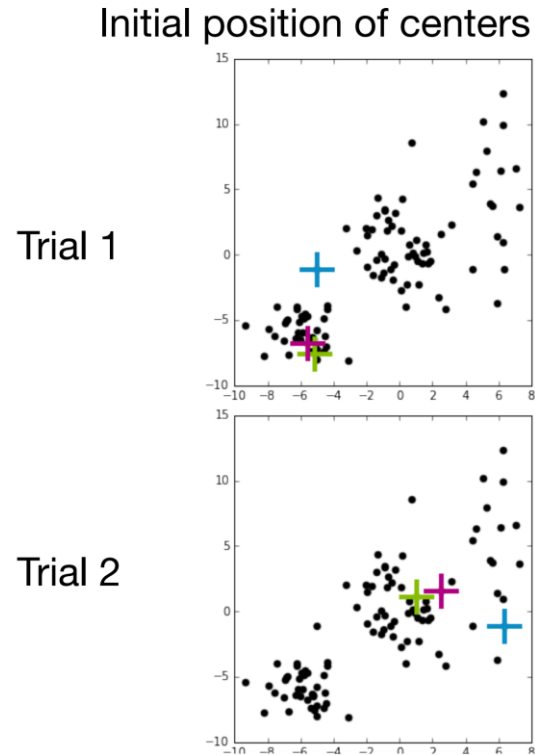
Has k-means converged?



Downsides of k-means

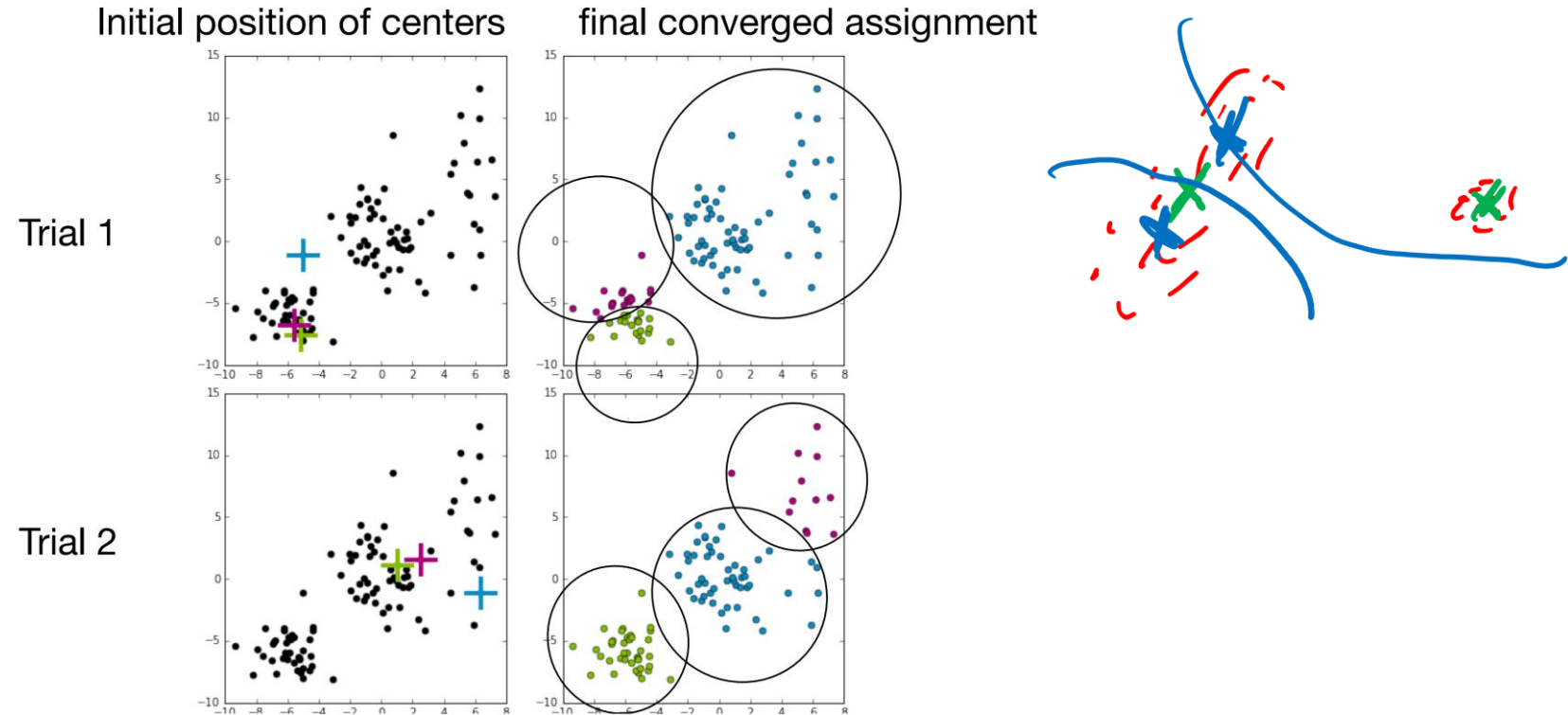
training loss?
cross-validation?

- Requires number of clusters k to be specified by us
- Final solution depends on init



Downsides of k-means

- Requires number of clusters k to be specified by us
- Final solution depends on init



k-means++: A smarter initialization



1. Choose first cluster center μ_1 uniformly at random from data points
2. For $k = 2, \dots, K$:
 - For each data point x_i , compute distance d_i to nearest cluster center
 - Choose new cluster from data points with probability of x_i chosen proportional to d_i^2

Apply standard k-means after this initialization

When does k-means fail?

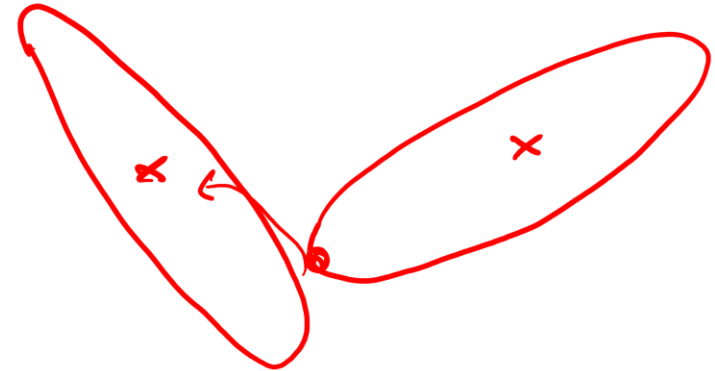


*k-means assumes
equally sized,
spherical clusters*

Disparate cluster sizes

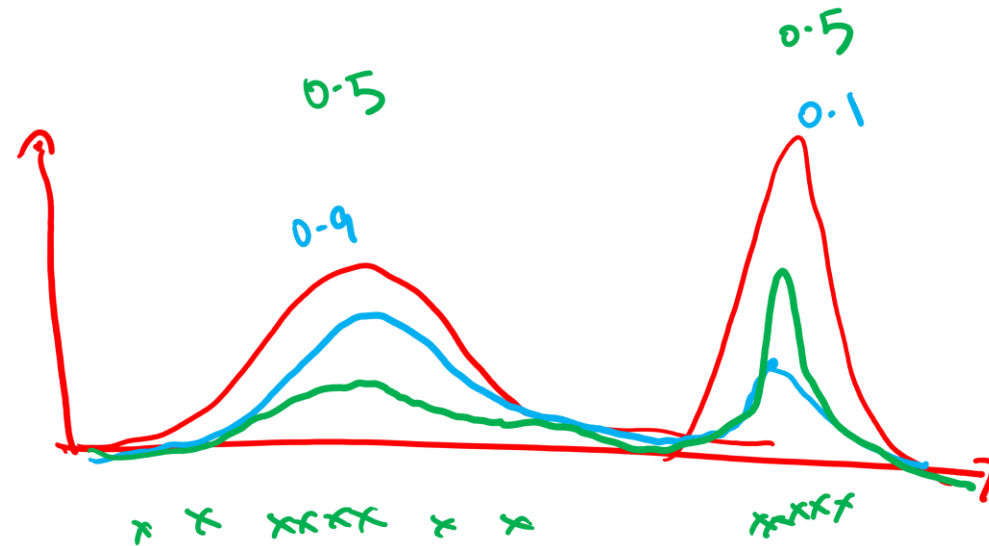
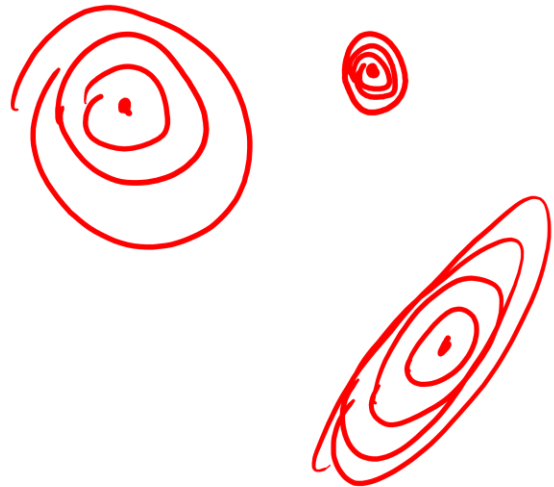


Differently shaped clusters



Gaussian mixture models (GMMs)

- Key idea: model each cluster as a Gaussian distribution with its own mean and covariance
- Define a distribution over data points that is a mixture of Gaussians



Gaussian mixture models (GMMs)

- Parameters:
 - Mixing weights π_j , for $j = 1, \dots, K$ *add up to 1*
 - Means μ_j , for $j = 1, \dots, K$
 - Covariances Σ_j , for $j = 1, \dots, K$
- Under the GMM, sample x_i is drawn as follows:
 - First sample a cluster $z_i \in \{1, \dots, K\}$ w.p. π
 - Sample $x_i \sim N(\mu_{z_i}, \Sigma_{z_i})$
- Given parameters of the GMM, how do we recover clusters?

$$p(z_j | x) \quad \text{for all } j = 1, \dots, K$$

How do we fit GMMs?

- The same way we always do: Maximum likelihood estimation!
 - Define the likelihood $P_{\theta}(x)$
 - Find θ that maximizes the likelihood of observing the data x_1, \dots, x_n
- For simplicity, assume $x \in \mathbb{R}$
- Likelihood:

$$P(x | \pi, \mu, \sigma) = \sum_{j=1}^k \frac{P(x, z=j | \pi, \mu, \sigma)}{P(z=j) P(x | z=j)}$$
$$= \sum_{j=1}^k \pi_j \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(\mu_j - x)^2}{\sigma_j^2}}$$

$\pi_j \leftarrow \frac{P(z=j) P(x | z=j)}{P(x | \mu_j, \sigma_j^2)}$

Recap lecture 1: Fitting a single Gaussian

$$P(x | \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\frac{\partial}{\partial \mu} \log P(\mathcal{D} | \mu, \sigma) = \frac{\partial}{\partial \mu} \left[-n \log(\sigma\sqrt{2\pi}) - \sum_{i=1}^n \frac{(x_i - \mu)^2}{2\sigma^2} \right] \xrightarrow{\text{set to 0}} \hat{\mu}_{MLE} = \frac{1}{n} \sum_{i=1}^n x_i$$

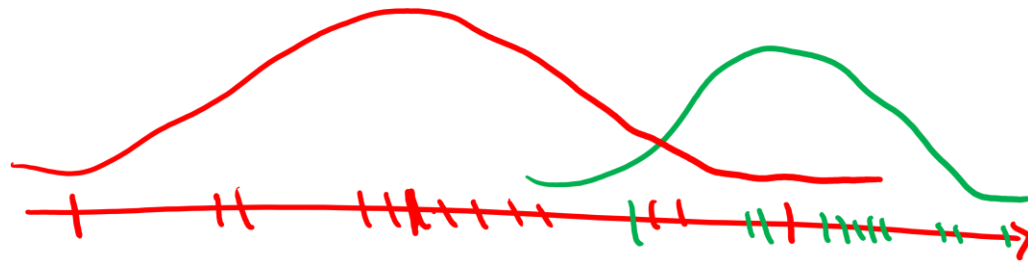
$$\frac{\partial}{\partial \sigma} \log P(\mathcal{D} | \mu, \sigma) = \frac{\partial}{\partial \sigma} \left[-n \log(\sigma\sqrt{2\pi}) - \sum_{i=1}^n \frac{(x_i - \mu)^2}{2\sigma^2} \right] \xrightarrow{\text{set to 0}} \hat{\sigma}_{MLE}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu}_{MLE})^2$$

Fitting multiple Gaussians

$$\log P(\mathcal{D}|\pi, \mu, \sigma) = \sum_{i=1}^n \log \sum_{k=1}^K \pi_k P(x_i|\mu_k, \sigma_k)$$

likelihood

- Can differentiate, but... no closed form solution (try this at home!)
- What if we knew which of the k Gaussians each x_i came from?



Expectation Maximization (EM) algorithm for GMMs

Repeat until convergence:

- E-step: For each x_i , guess which Gaussian it came from

$$r_{ij} = P(z_i = j | x_i) = \frac{P(x_i, z_i = j)}{\sum_k P(x_i, z_i = k)} = \frac{\pi_j N(x_i | \mu_j, \sigma_j)}{\sum_k \pi_k N(x_i | \mu_k, \sigma_k)}$$

- M-step: Fit Gaussian parameters accordingly

weighted MLE

$$n_j = \sum_{i=1}^n r_{ij}$$

$$\pi_j = \frac{n_j}{n}$$

$$\mu_j = \frac{1}{n_j} \sum_{i=1}^n r_{ij} x_i$$

$$\sigma_j^2 = \frac{1}{n_j} \sum_{i=1}^n r_{ij} (x_i - \mu_j)^2$$

Why does EM work?

- How do we justify this procedure?
- Does it converge?
- What does it converge to?

- Key idea:
 - E-step forms a tight lower bound to $P(\mathcal{D}|\pi, \mu, \sigma)$
 - M-step optimizes this lower bound

Latent variable models

- Previously, we've studied models where all variables are observed

$$p(x) = N(x | \mu, \sigma^2)$$
$$p(y|x) = \sigma(yw^T x)$$
$$\vdots$$

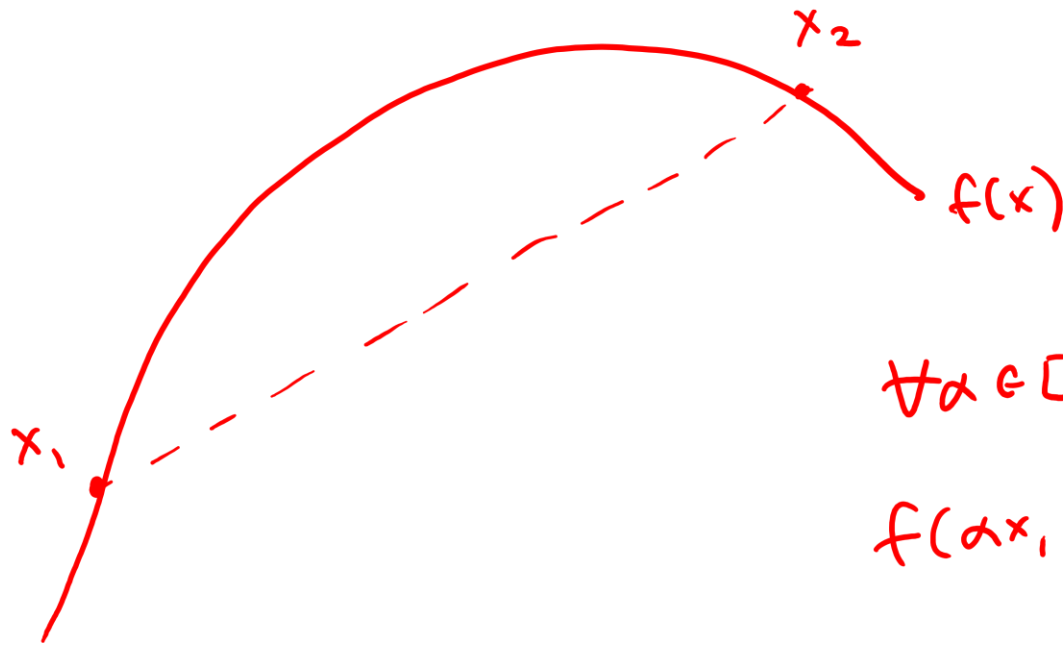
- Now, we have a latent (=unobserved) variable z

$$p(x) = \sum_z p(x, z)$$
$$\log p(x) = \log \sum_z p(x, z)$$

- EM is a general method for optimizing latent variable models

Detour: Jensen's inequality

*Named after Danish mathematician Johan Jensen, no relation to NVIDIA



sign flip if f convex

$\forall \alpha \in [0, 1]$, concave f ,

$$f(\alpha x_1 + (1-\alpha)x_2) \geq \alpha f(x_1) + (1-\alpha)f(x_2)$$

$$\underline{f(\mathbb{E}[x])} \geq \mathbb{E}[f(x)]$$

The Evidence Lower Bound (ELBO)

$$\begin{aligned} \log p(x; \theta) &= \log \sum_z p(x, z; \theta) \\ &= \log \sum_z Q(z) \frac{p(x, z; \theta)}{Q(z)} \\ &\geq \sum_z Q(z) \log \frac{p(x, z; \theta)}{Q(z)} \end{aligned}$$

(By Jensen's inequality, ELBO(x; Q, θ)
holds for all dists Q(z))

$$\mathbb{E}_{z \sim Q} \left(\frac{p(x, z; \theta)}{Q(z)} \right)$$

What should Q be to make this tight at a given Q?

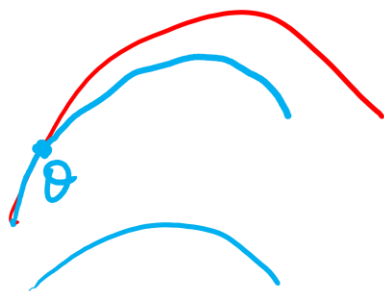
$$\frac{p(x, z; \theta)}{Q(z)} \text{ should be indep of } z$$

$$\Rightarrow Q(z) = p(z|x; \theta)$$

$$\text{so } \frac{p(x, z; \theta)}{Q(z)} = p(x; \theta)$$

pick Q to maximize

$$\sum_z Q(z) \log \frac{p(x, z; \theta)}{Q(z)}$$



EM is alternating maximization on the ELBO

$$\log P(x; \theta) \geq \text{ELBO}(x; Q, \theta)$$

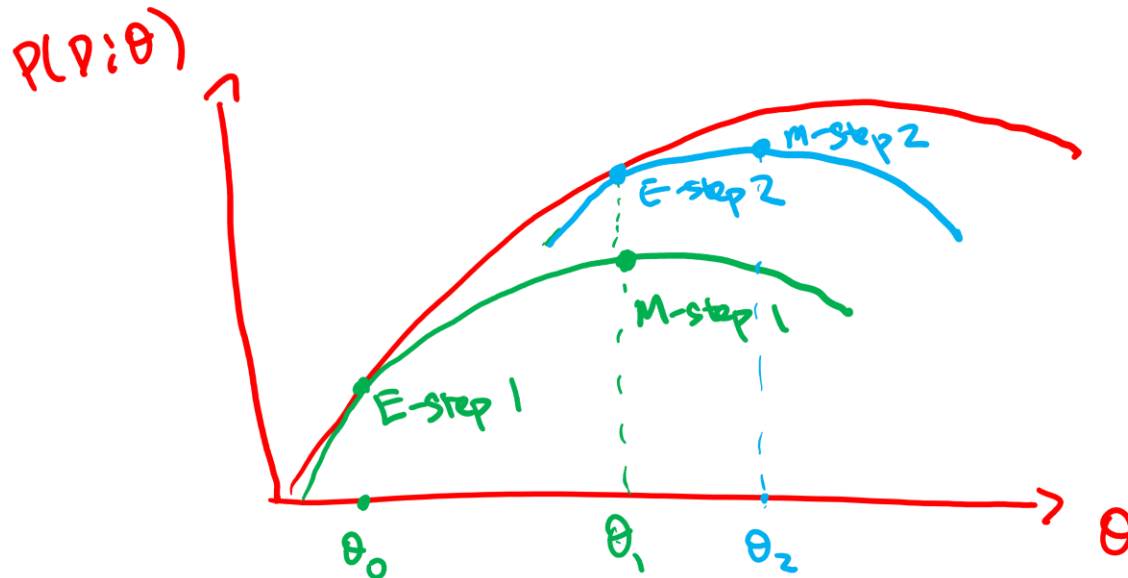
$$Q(z) = P(z | x; \theta_x)$$

E-step: Optimize Q to create tight lower bound $\text{ELBO}(x; Q, \theta_t)$

M-step: Optimize θ_{t+1} to maximize $\text{ELBO}(x; Q, \theta_t)$

$$\theta_{t+1} \leftarrow \underset{\theta}{\operatorname{argmax}} \sum_z Q(z) \log \frac{P(x, z | \theta_t)}{Q(z)}$$

for GMMs, this is convex



Convergence

- Likelihood never decreases
 - Lower bound is always tight
 - We always increase (or stay the same) on the lower bound
- Therefore, EM converges
- But it converges to a local maximum

- Do we need to use EM?

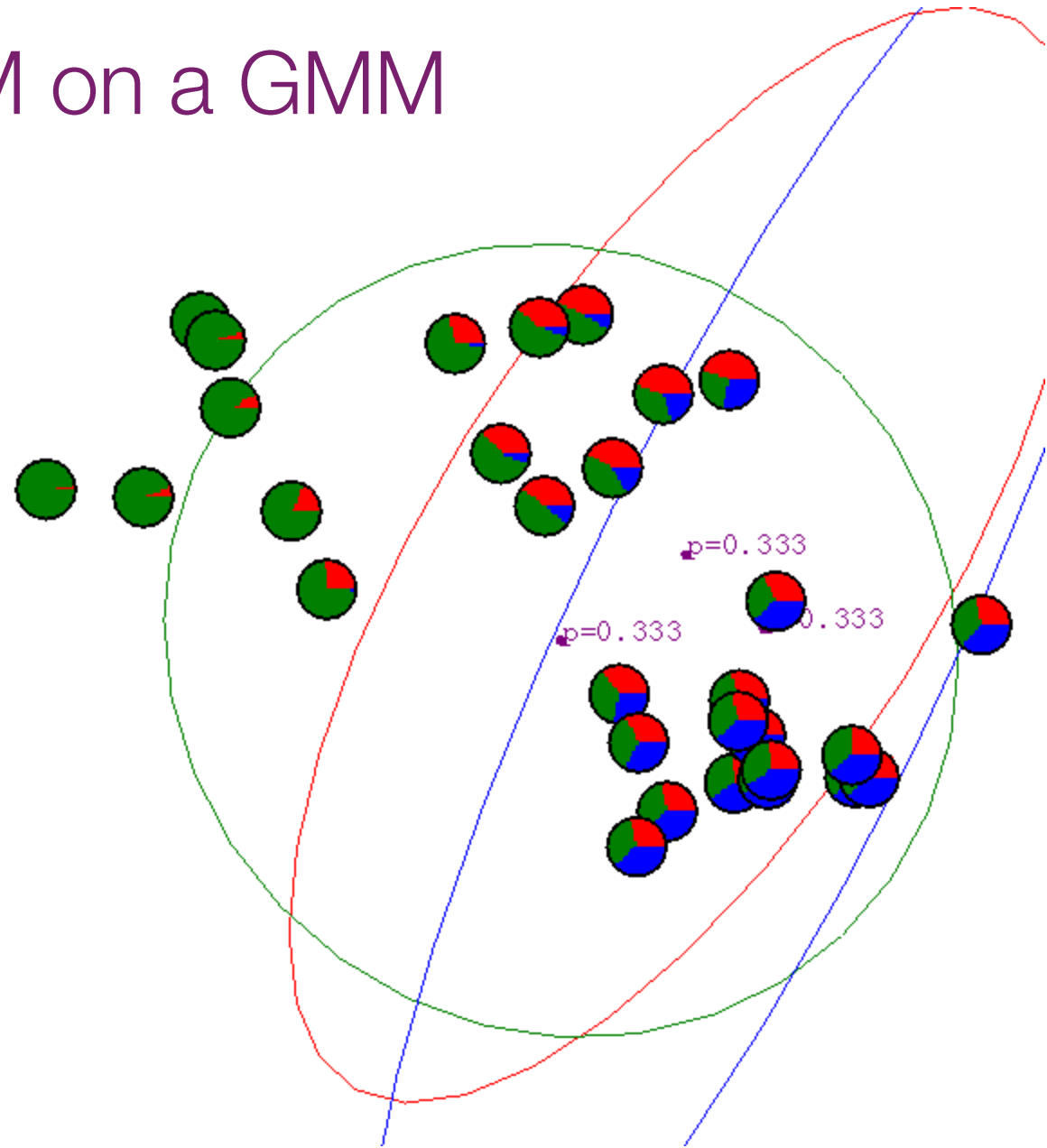
Multivariate Gaussians

Given $x \in \mathbb{R}^d$, $P(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d \det \Sigma}} \exp\left(-\frac{1}{2}(x - \mu)^\top \Sigma^{-1}(x - \mu)\right)$

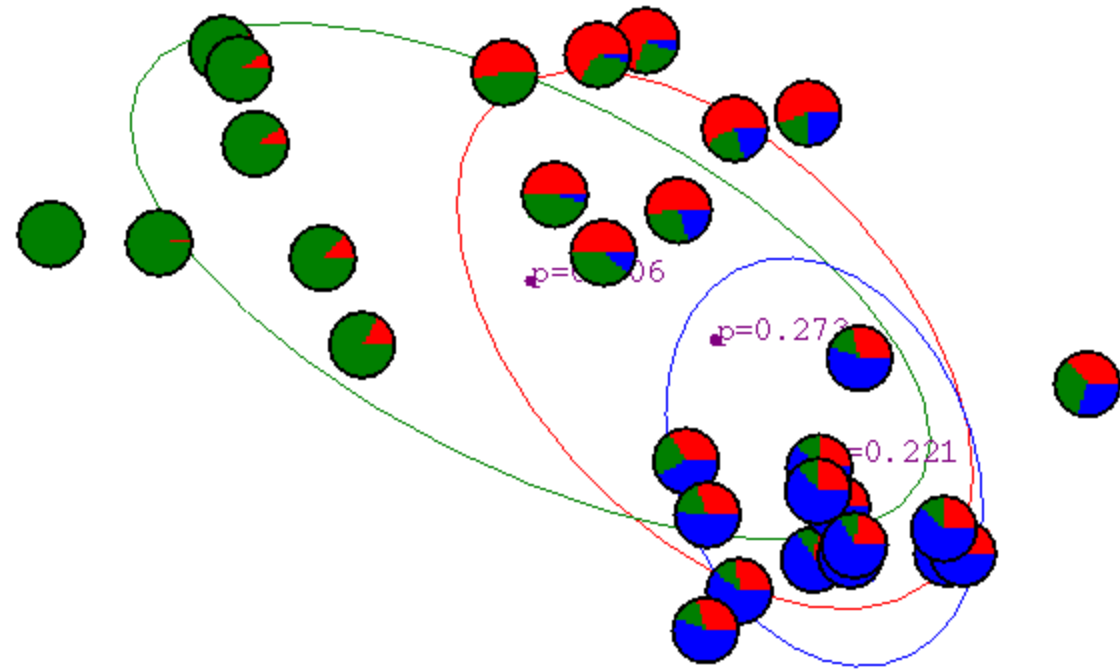
$$\hat{\mu}_{\text{MLE}} = \frac{1}{n} \sum_{i=1}^n x_i$$

$$\hat{\Sigma}_{\text{MLE}} = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu}_{\text{MLE}})(x_i - \hat{\mu}_{\text{MLE}})^\top$$

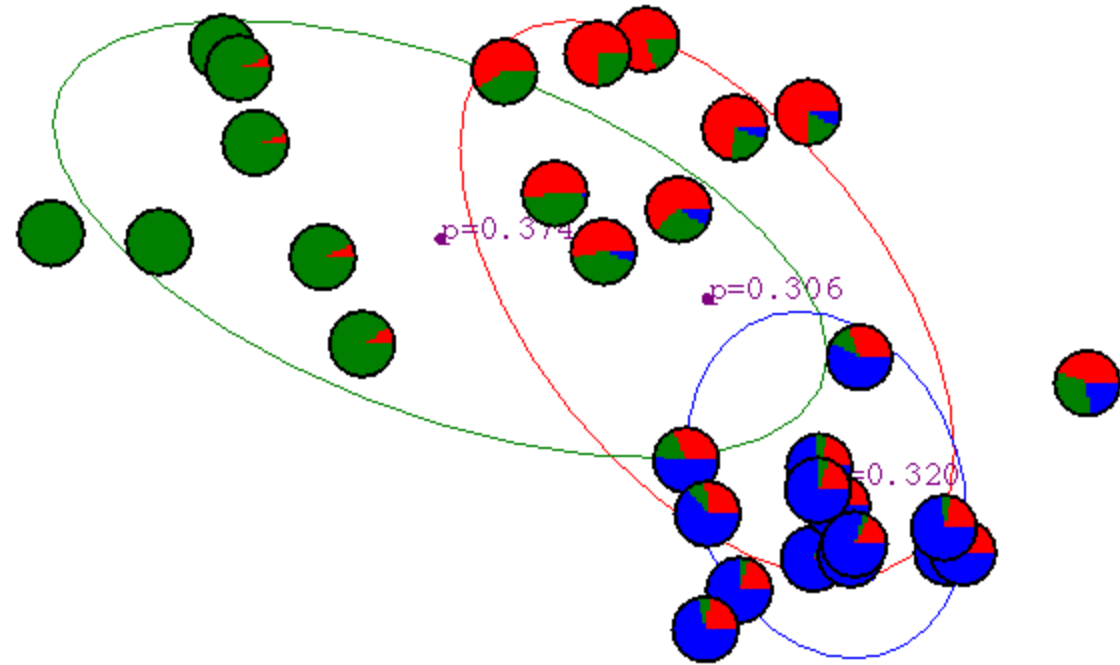
Example: Running EM on a GMM



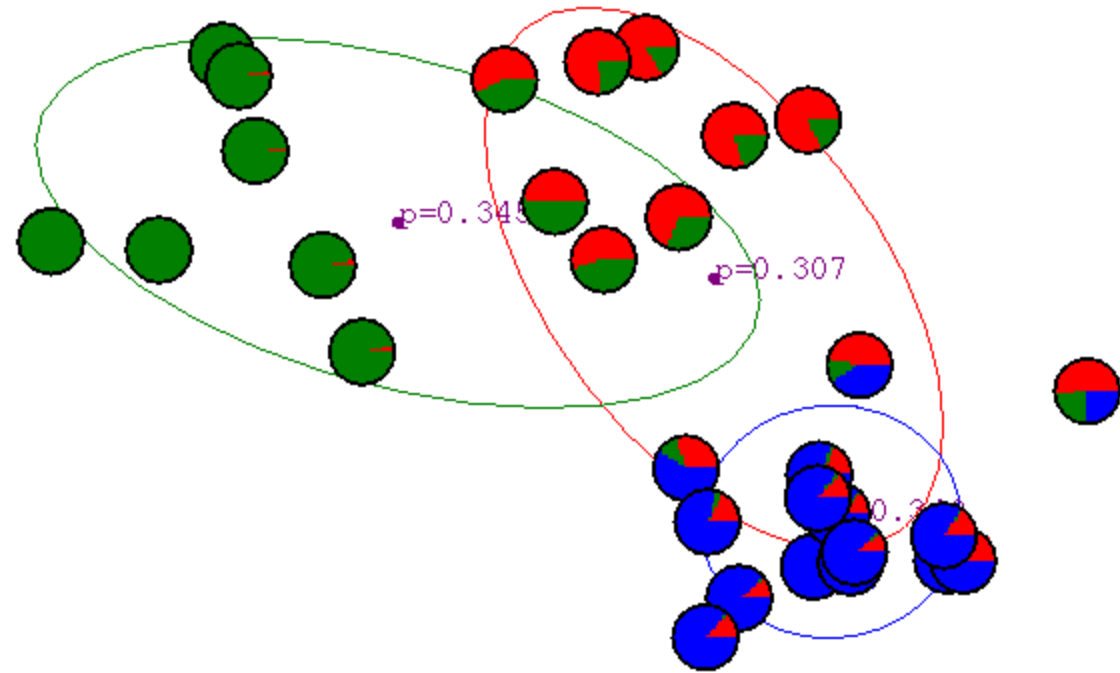
Example: Running EM on a GMM



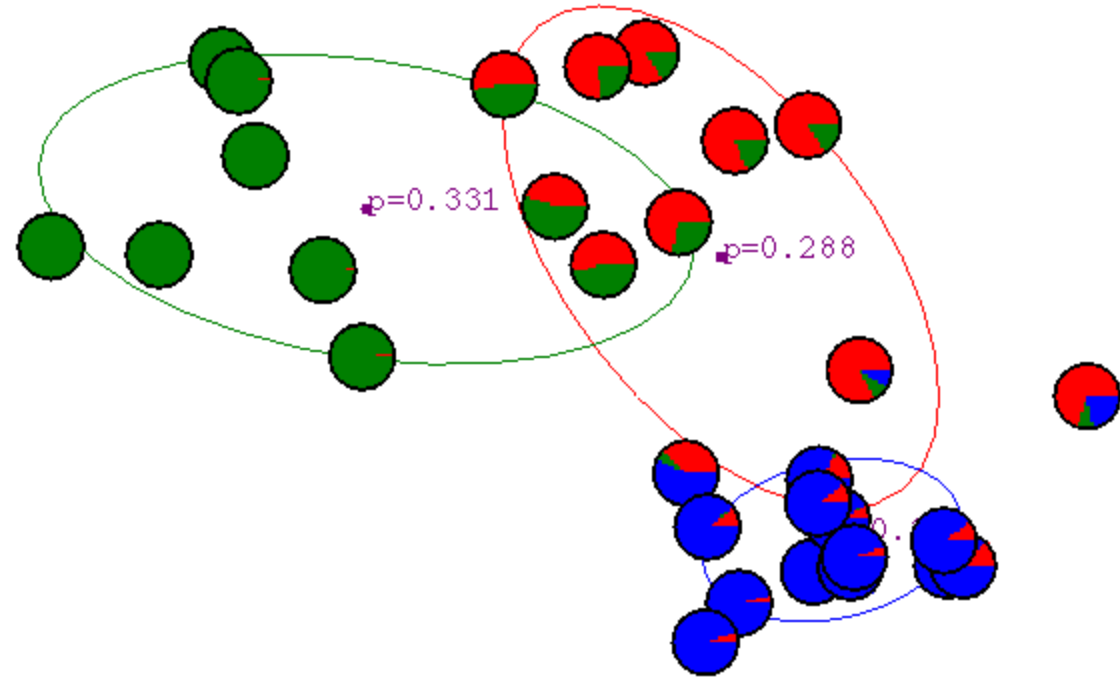
Example: Running EM on a GMM



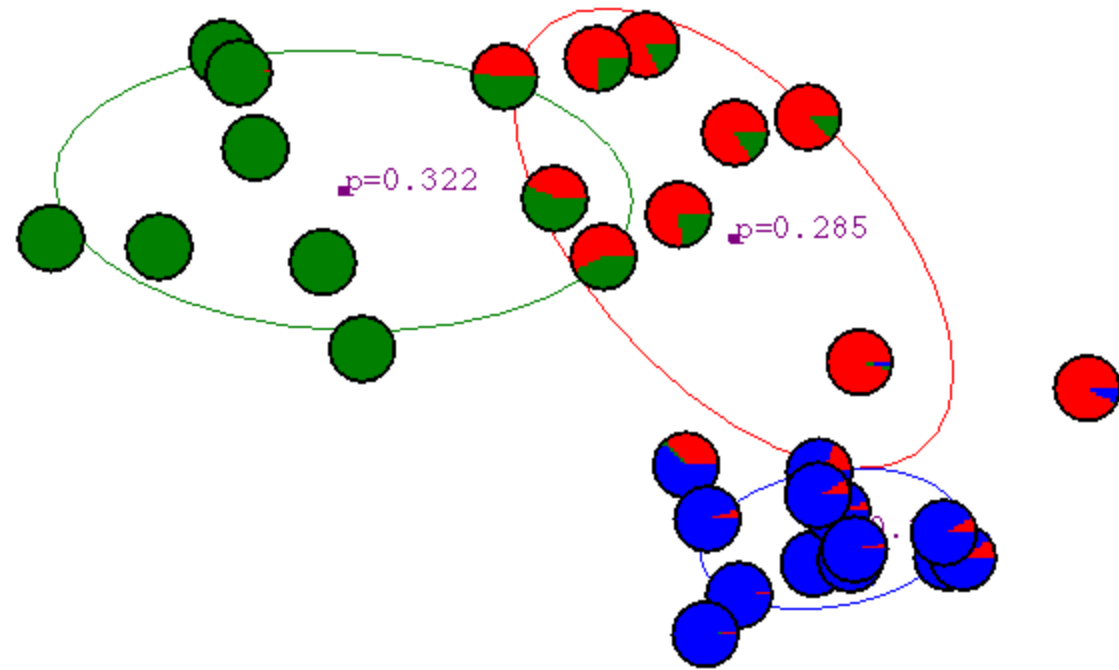
Example: Running EM on a GMM



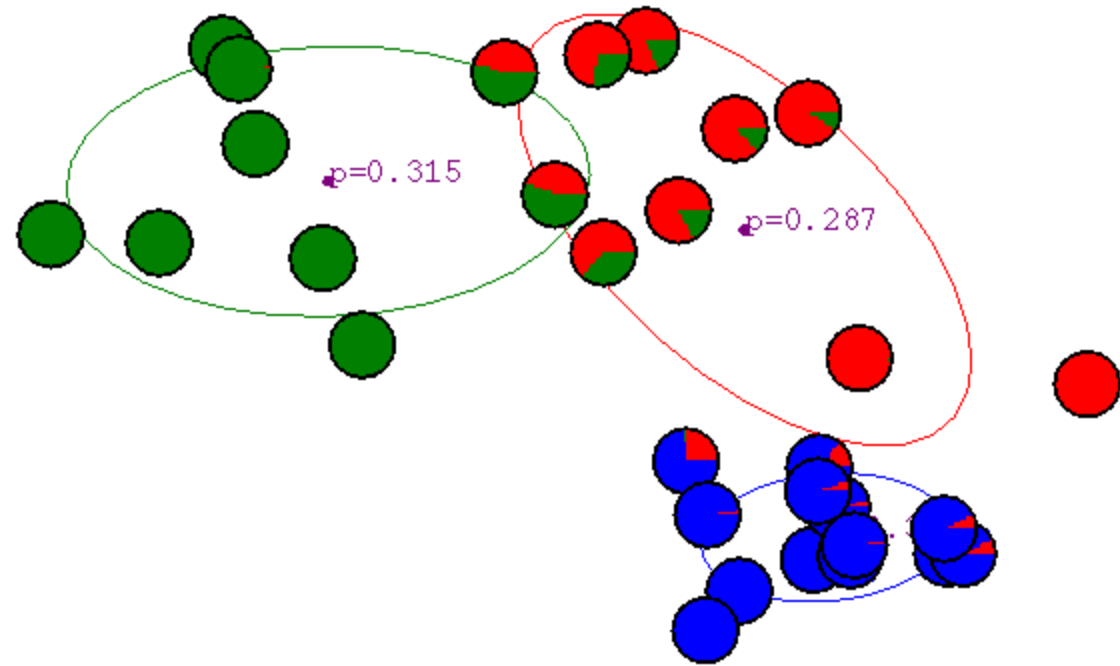
Example: Running EM on a GMM



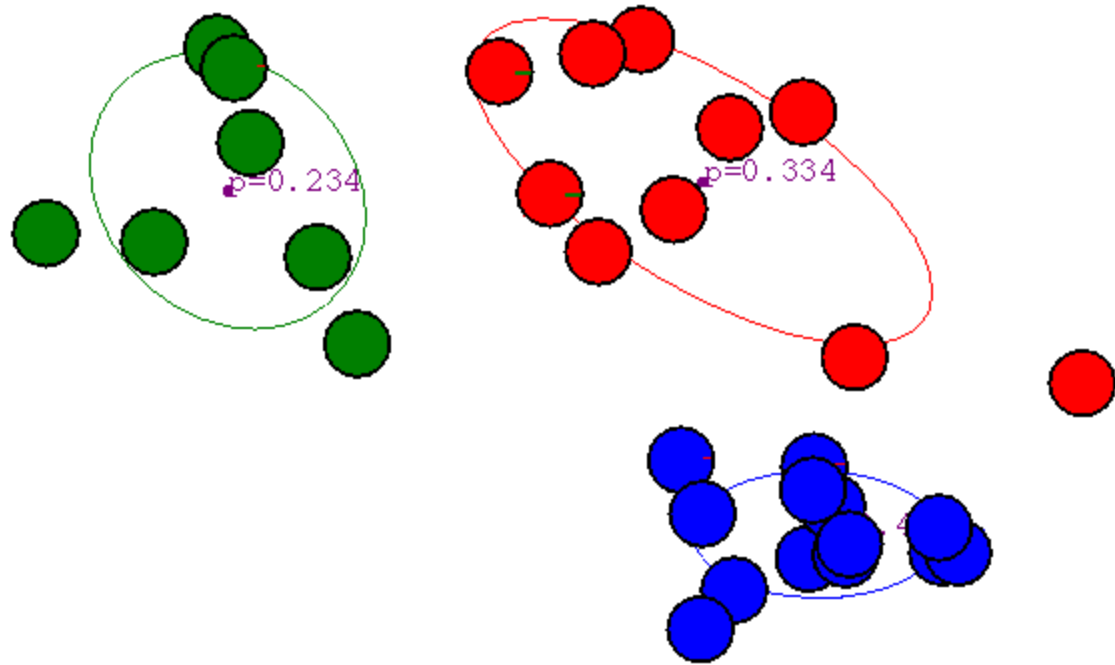
Example: Running EM on a GMM



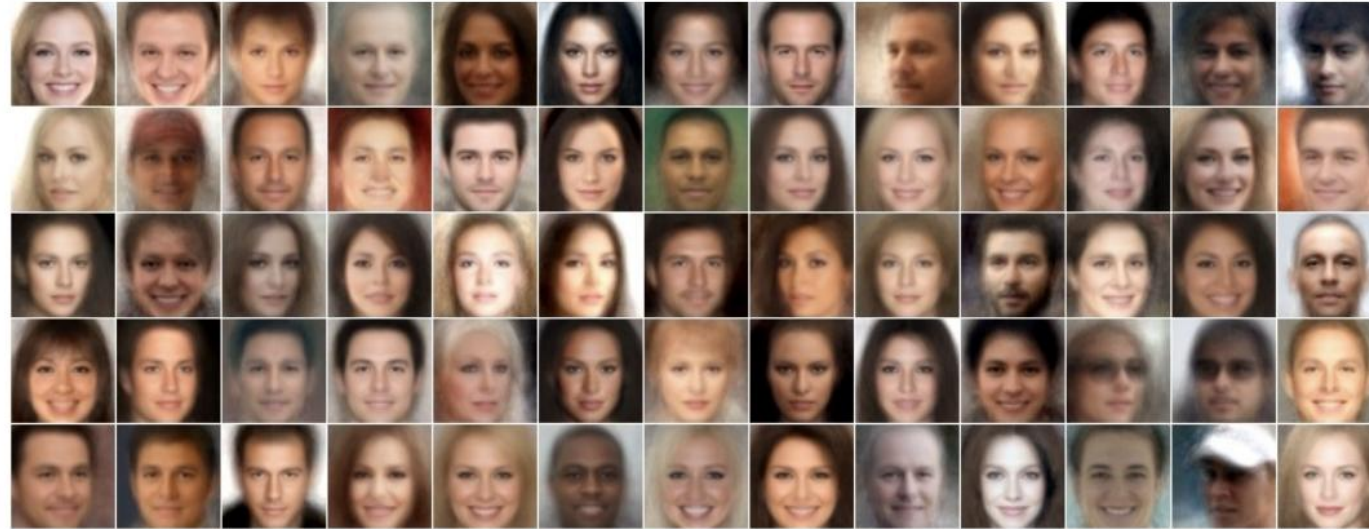
Example: Running EM on a GMM



Example: Running EM on a GMM



Example: GMMs on faces



- Generated samples from GMM ($K=1,000$) trained on CelebA
- Images: $64 \times 64 \times 3$
- Covariance: Restricted to rank-10 matrices

Example: GMMs on faces

