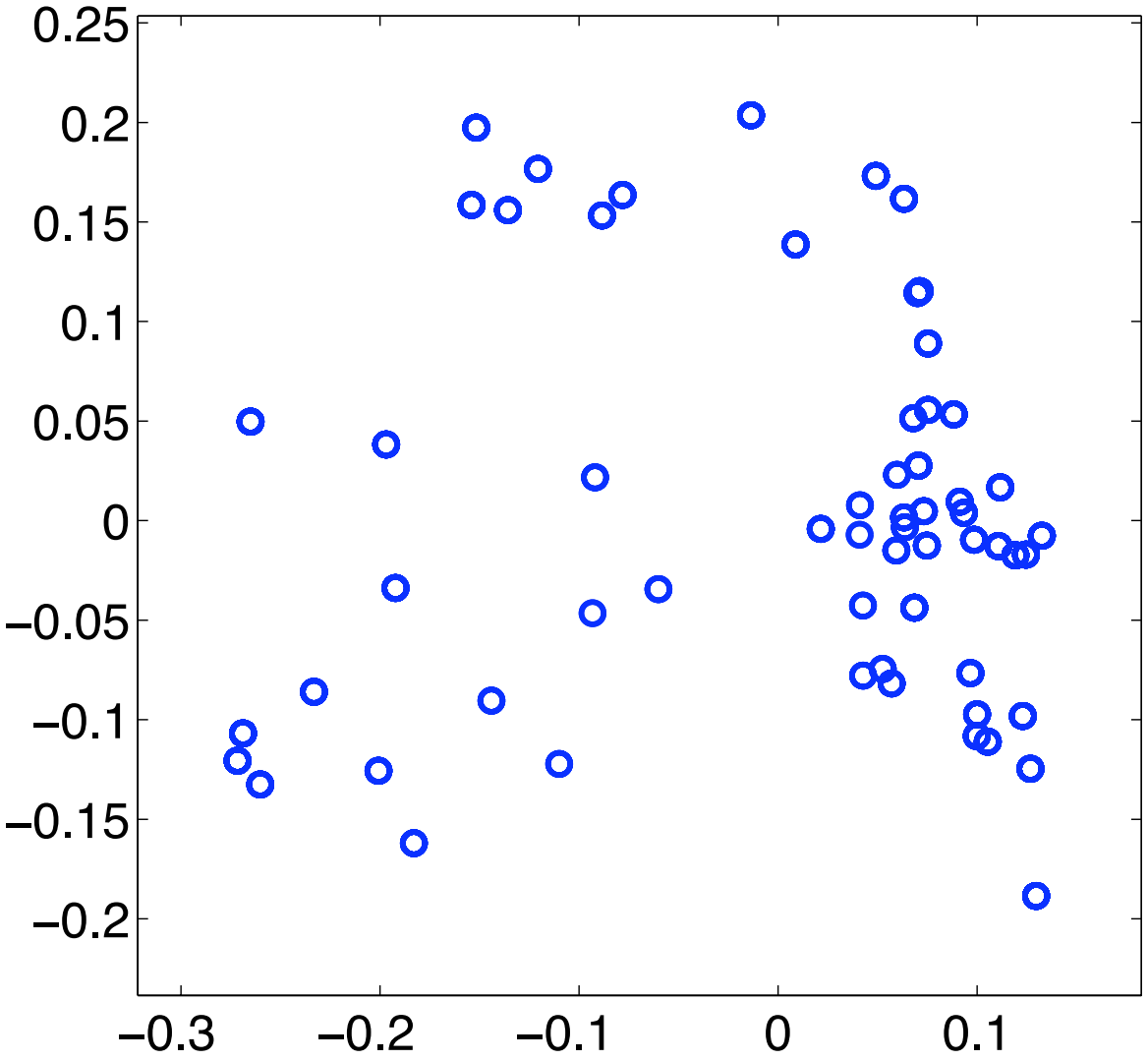
Clustering



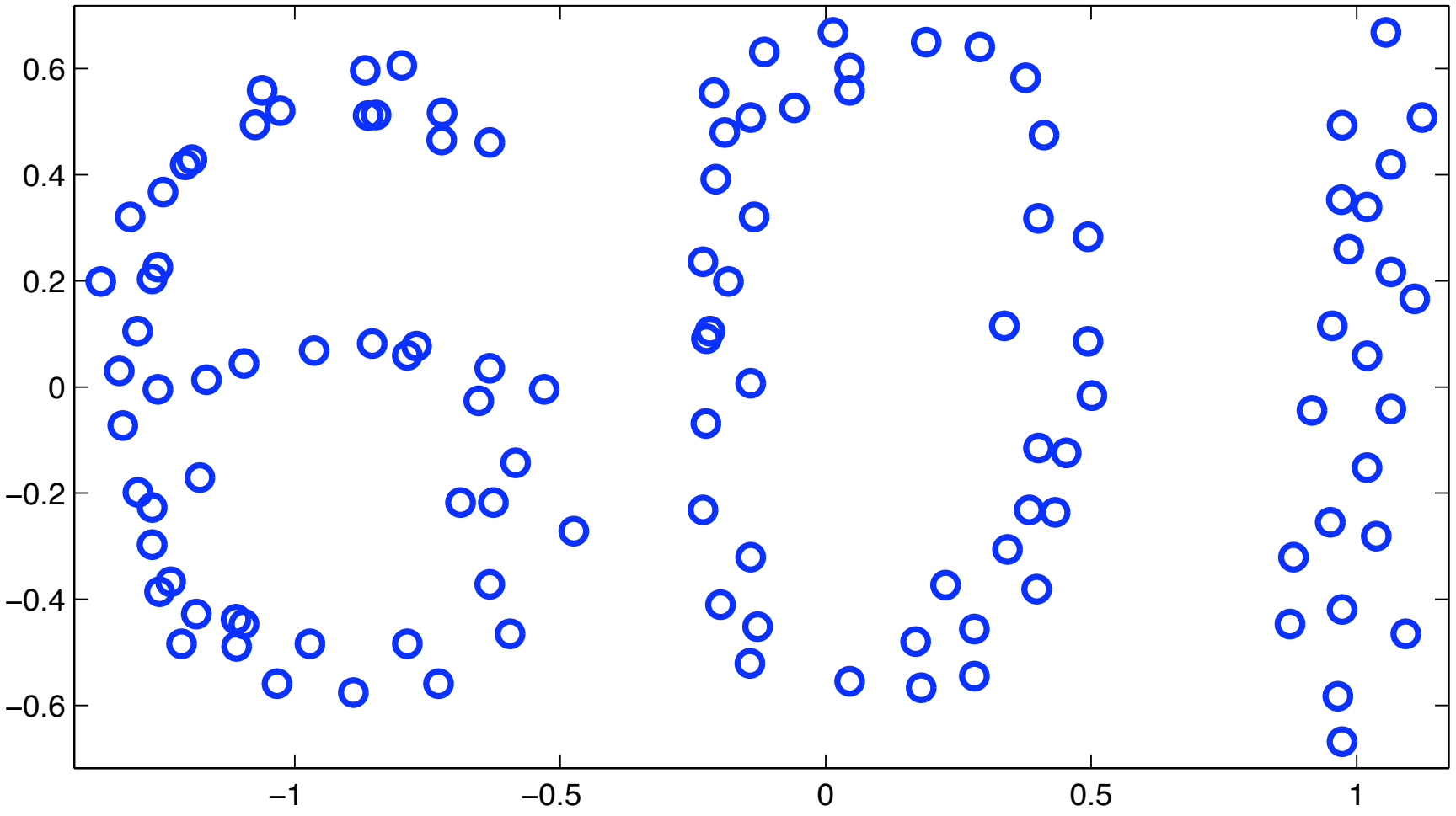
- Given a set of points, break them into groups of "similar" points
- Why?
  - Understanding
  - Inputs to other method
  - Compression

Ex: vector quantization

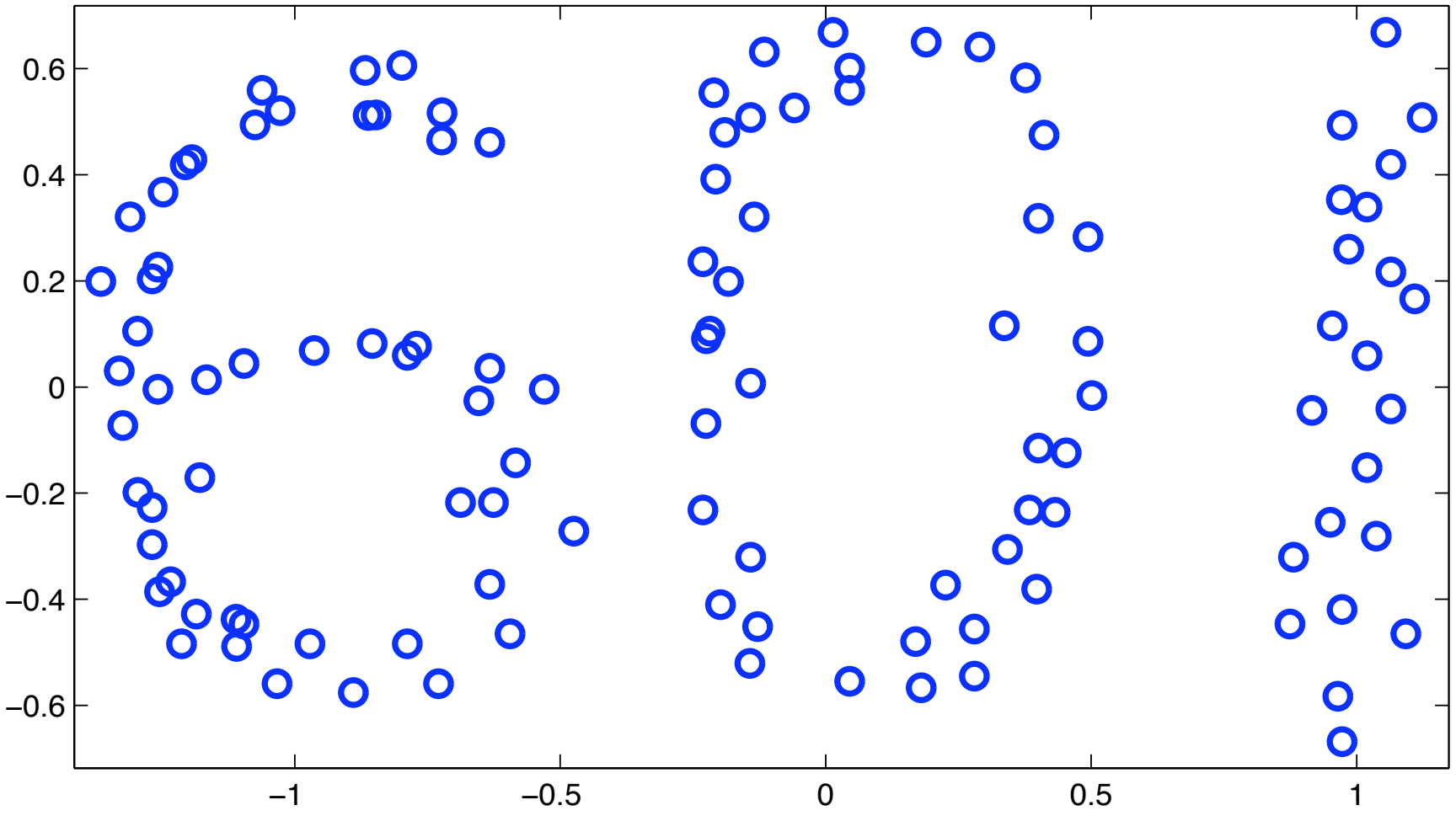
figure 9.3 from Bishop

points = RGB tuples, 1 per pixel
cluster centers = the best RGB colors if we can use only K

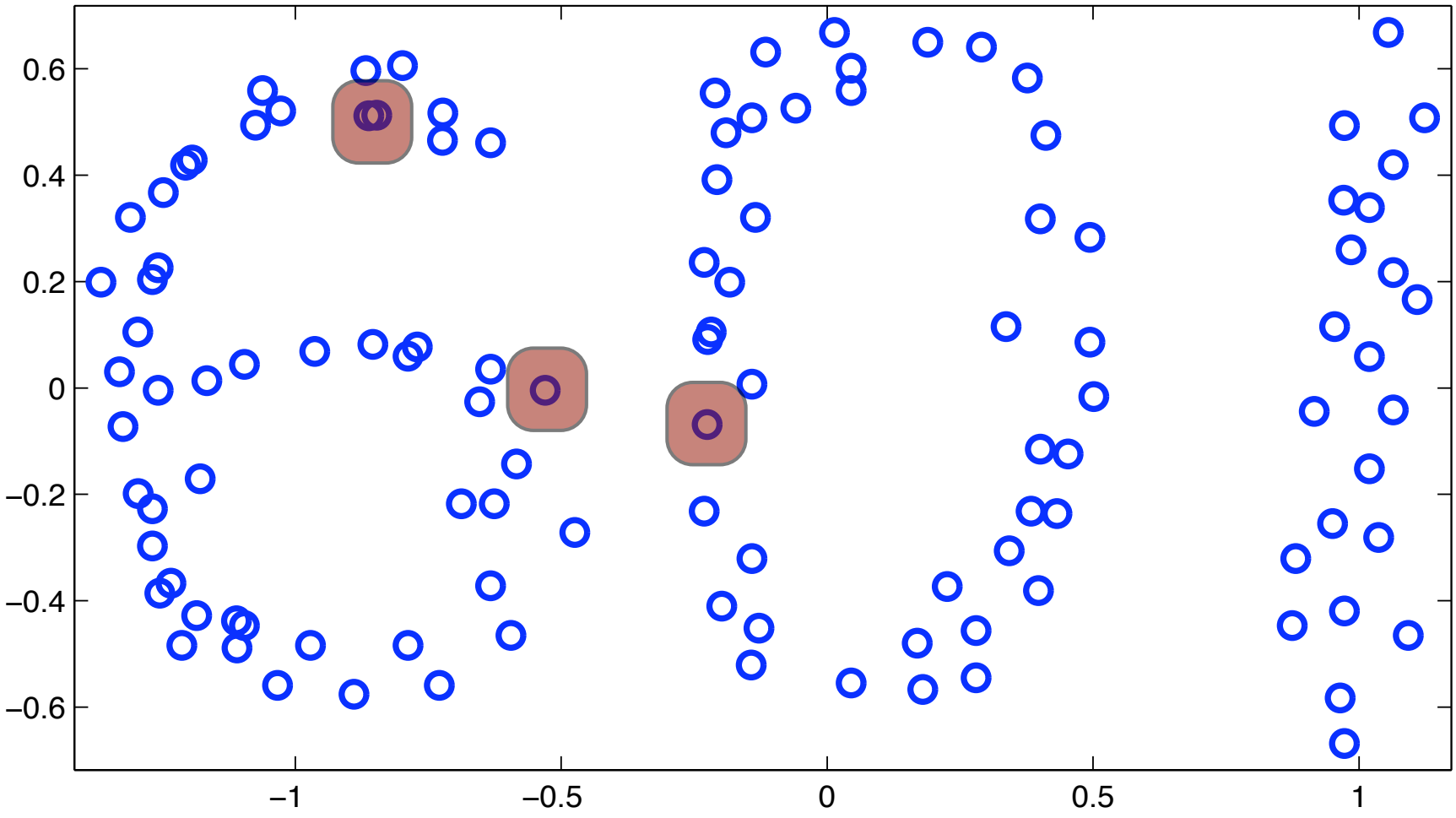
Clustering



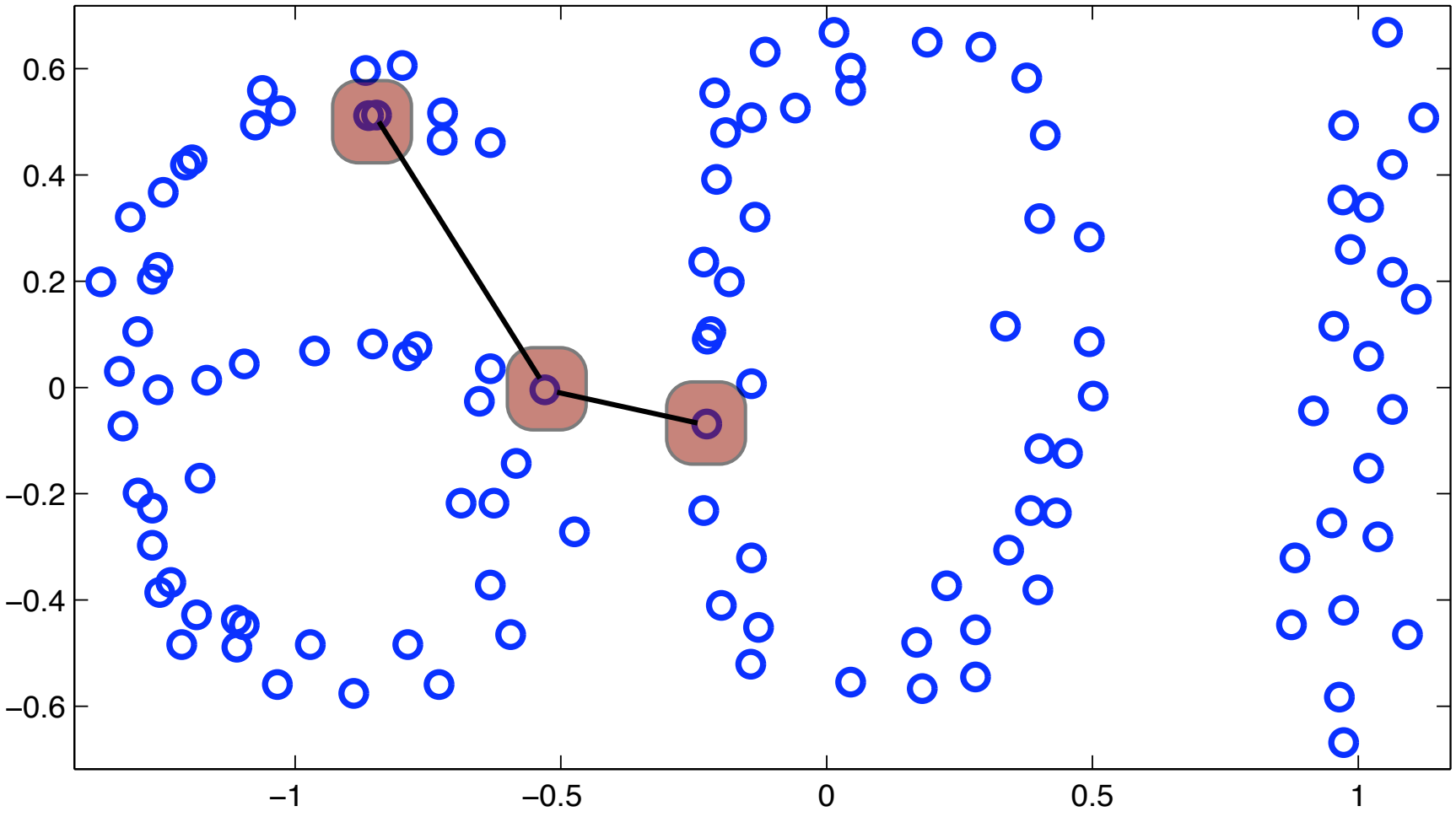
What do we mean by "similar"?



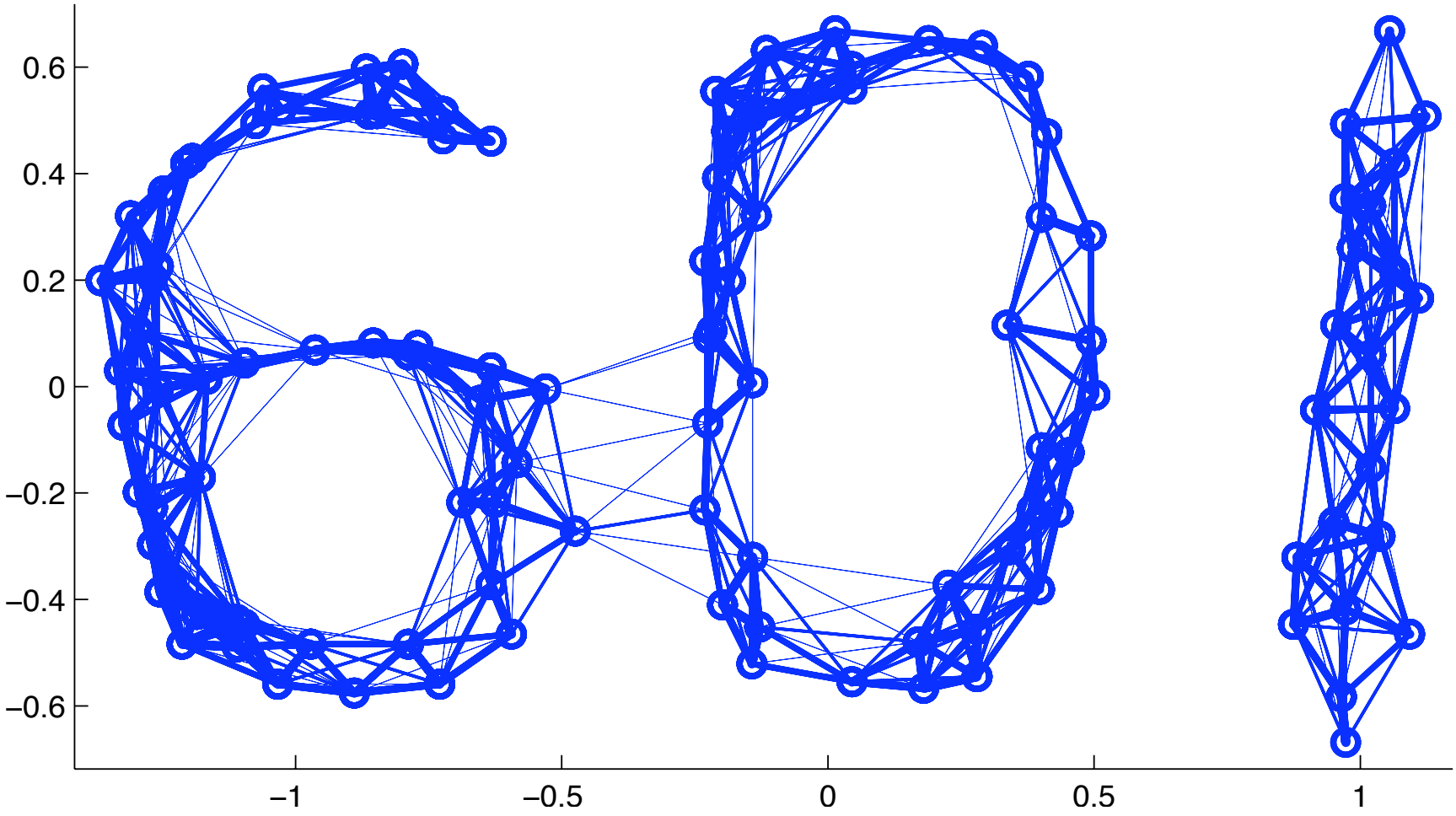
What do we mean by "similar"?



What do we mean by "similar"?



Spectral embedding: adjacency graph



Spectral embedding

$$A_{ij} = e^{-\|x_i - x_j\|^2/2\sigma^2}$$ adjacency matrix

$$d_i = \sum_j A_{ij}$$ "degrees"

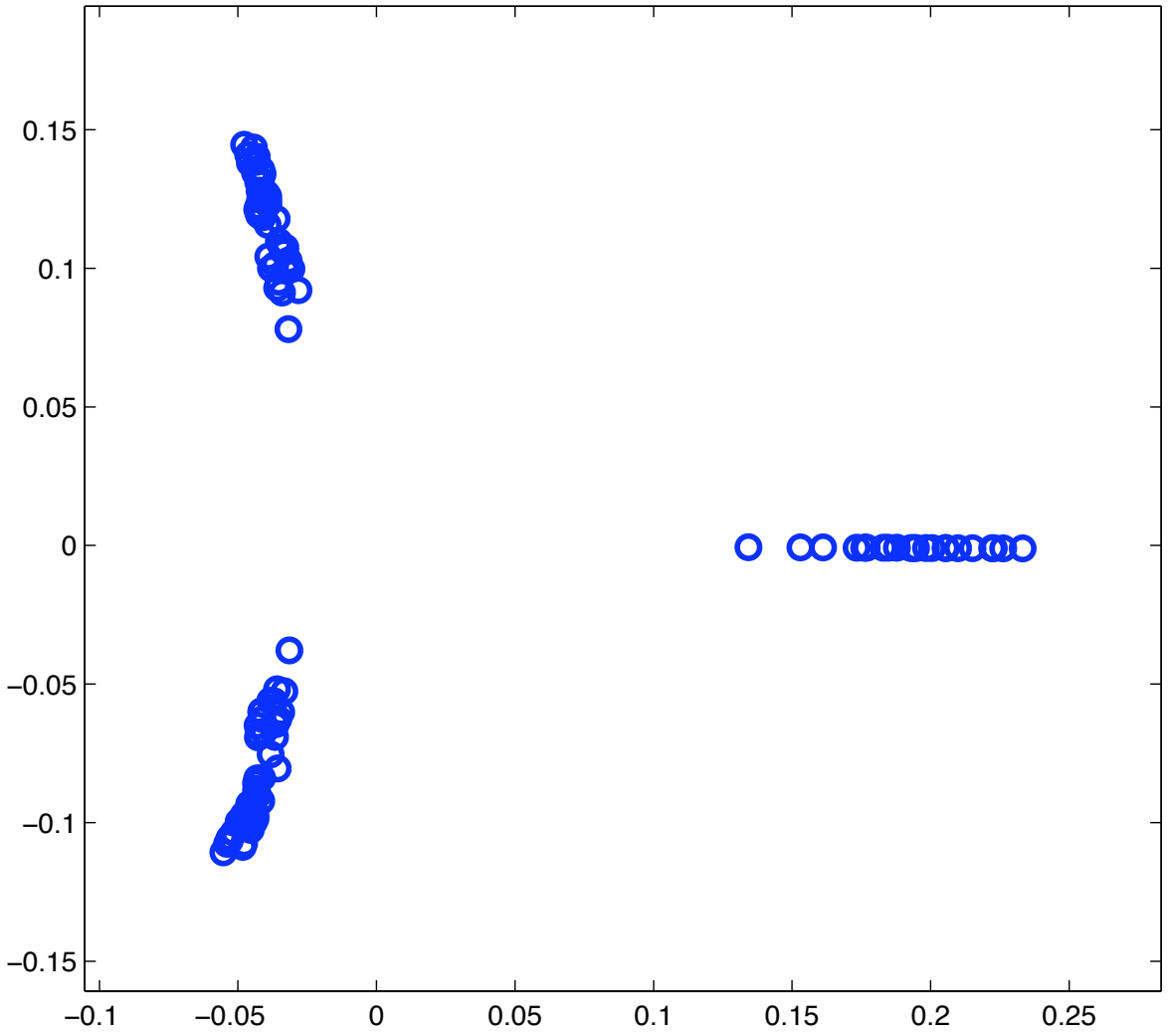
$$T_{ij} = \frac{1}{\sqrt{d_i d_j}} A_{ij}$$ normalized adjacency

$$U\Sigma V^T \approx T$$ SVD rank k+1

$$y_i = \Sigma U^T_{i,2:k+1}$$

↑ embedding

After embedding



Most common similarity metric

- …is just plain L_2 distance
  - perhaps with diagonal scaling, sqrt($x^T D x$)
  - rarely, sqrt($x^T A x$) for full, symmetric, positive definite A

Mahalanobis distance

- Many, many fancier metrics…
- But, after spectral embedding, L_2 is usually fine

A clustering objective

- Data $X_1, \ldots, X_n$ $\in \mathbb{R}^d$
- Suppose k clusters
  - centers $\mu_1, \ldots, \mu_k$ $\in \mathbb{R}^d$
- Let $z_{ij} = \begin{cases} 1 & \text{if } x_i \text{ assigned to } \mu_j \\ 0 & \text{o/w} \end{cases}$
- $L(z, \mu) = \sum_i \sum_j z_{ij} \|x_i - \mu_j\|^2$

$$Z = \left\{ z_{ij} \in \{0,1\} \;\middle|\; \sum_j z_{ij} = 1 \;\; \forall i \right\}$$

Optimization

- Finding $\min_{z,\mu} L(z, \mu)$ is an integer quadratic program
  - NP-complete
- Many common heuristics with varying degrees of success
  - in some circumstances, provable approximations

Alternating optimization for k-means

- Initialize z (or μ)
- Alternately minimize wrt. μ:

$$0 = \frac{d}{d\mu_j} L = \frac{d}{d\mu_j} \sum_i \sum_j z_{ij} \|x_i - \mu_j\|^2 = 2 \sum_i z_{ij} (x_i - \mu_j) \cdot (-1)$$

$$2 \sum_i z_{ij} (x_i - \mu_j)(-1) \qquad (-1)$$

$$\Rightarrow \sum_i z_{ij} x_i = \left(\sum_i z_{ij}\right) \mu_j \Rightarrow \mu_j = \text{mean of } x_i \text{ assigned to } j$$

$$= \sum_i z_{ij} x_i \Big/ \sum_i z_{ij}$$

- and z, subject to

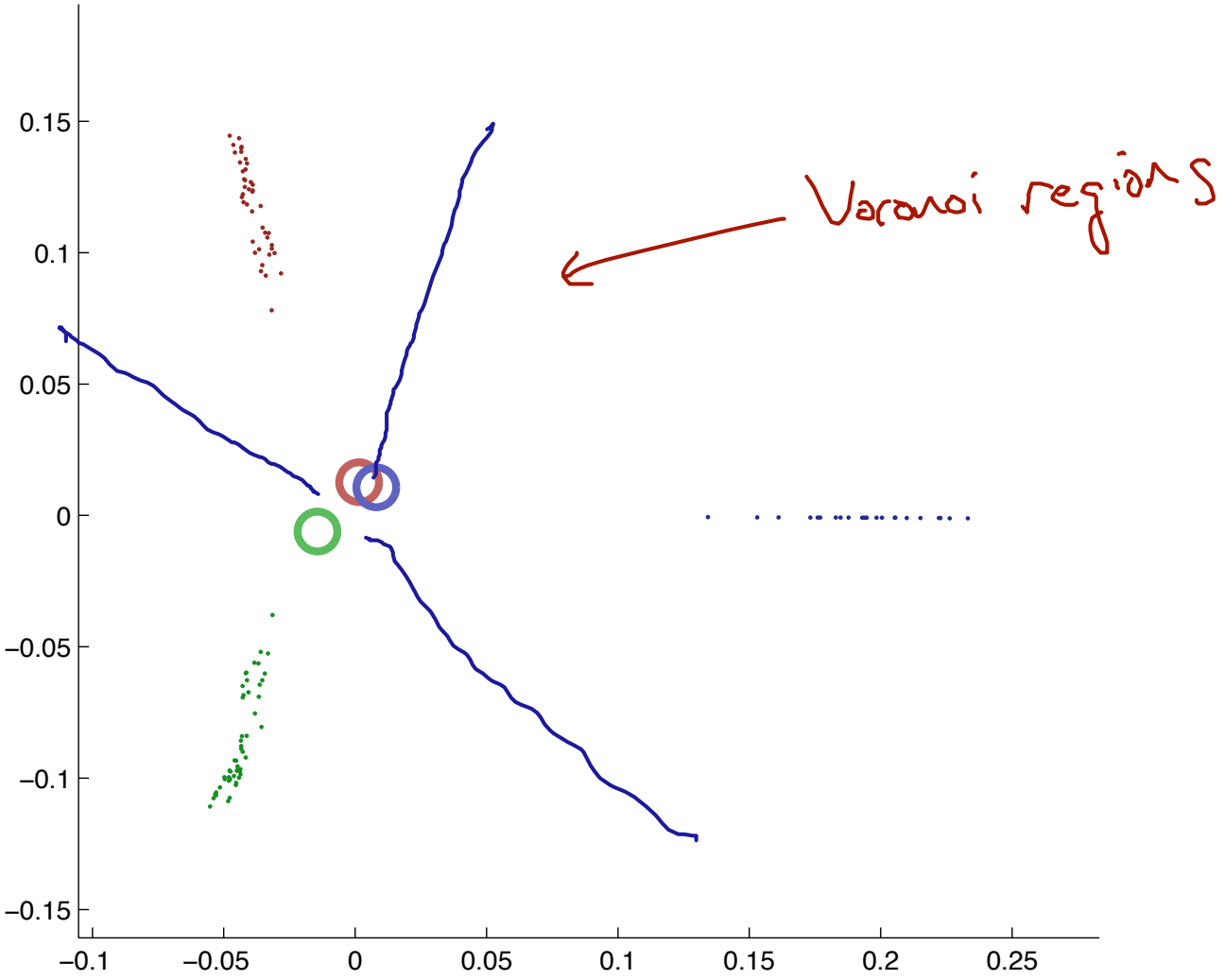
$$\forall i, \; j_i^* = \arg\min_j \|\mu_j - x_i\|^2 \qquad z_{ij_i^*} = 1$$

$$z_{ij} = 0 \quad j \neq j_i^*$$

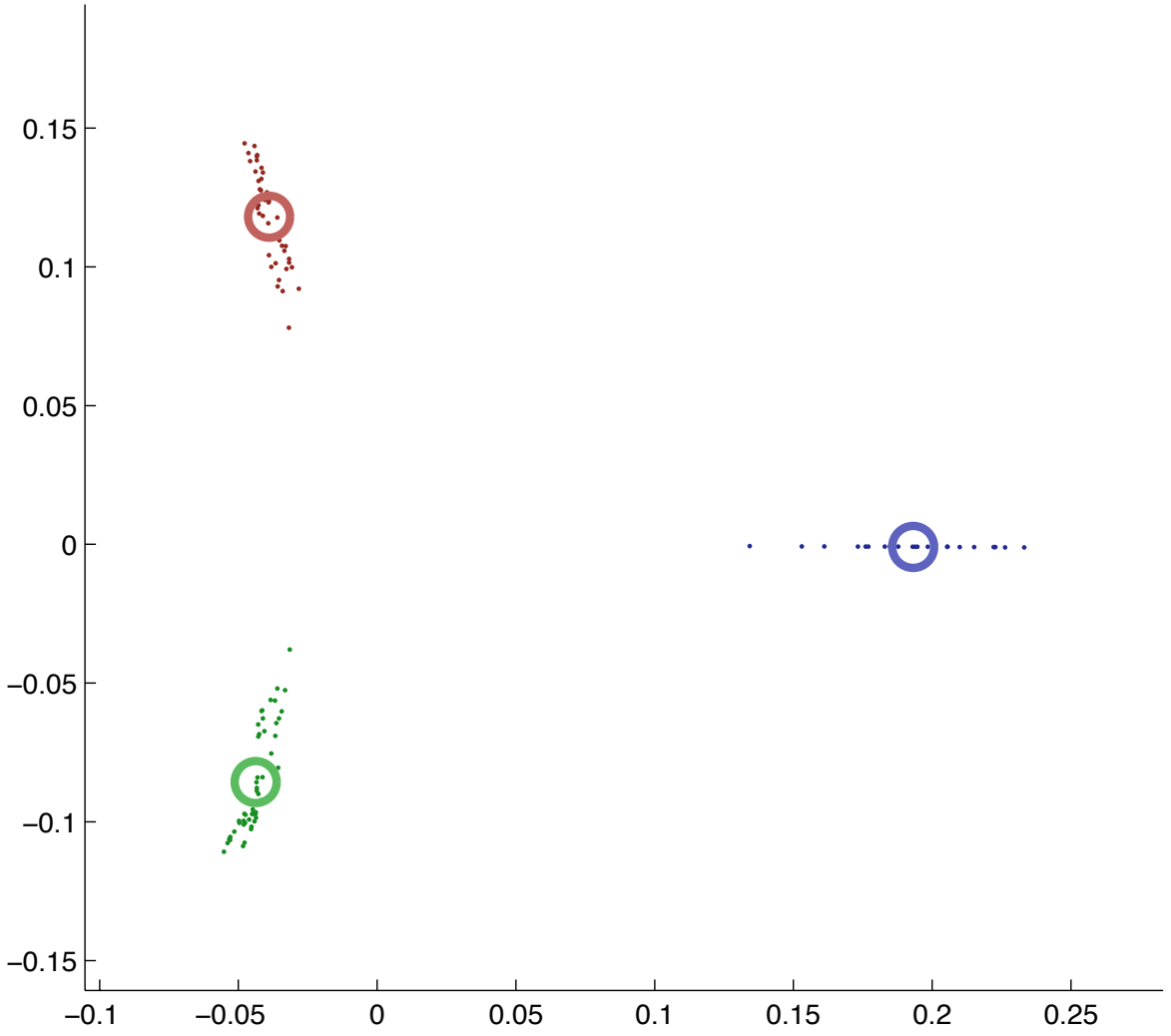
Convergence

- Min over z or μ decreases $L(z, \mu) \geq 0$
  - so, $L_1 \geq L_2 \geq L_3 \geq \ldots \geq 0$
- So, $L_t \rightarrow$ some limit λ
- There are finitely many settings for z
  - and one setting of μ that's optimal for each z
- So, for $t \geq$ some time T, both z_t and μ_t must remain constant

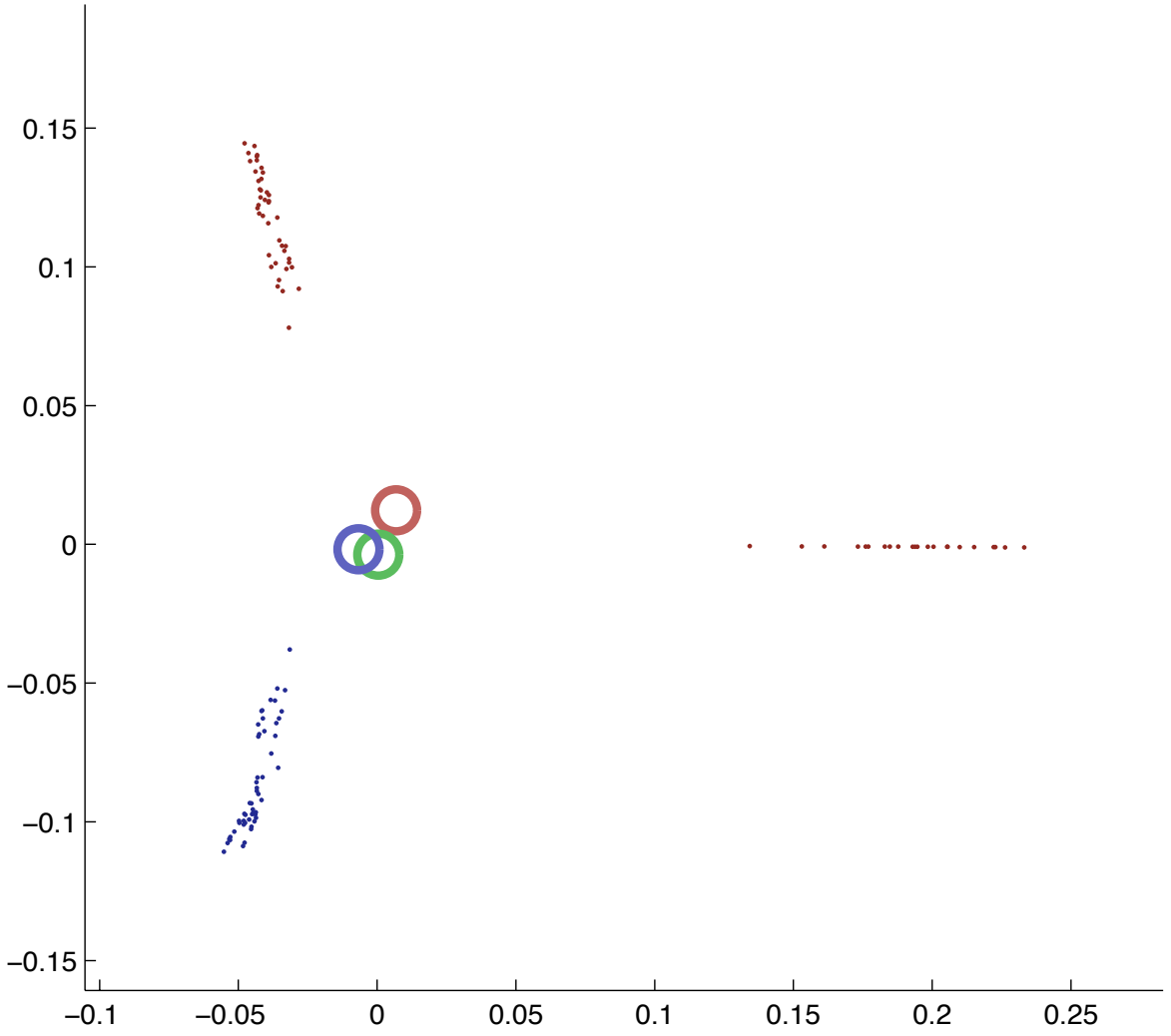
Example


Voronoi regions
0.15
0.1
0.05
0
−0.05
−0.1
−0.15
−0.1
−0.05
0
0.05
0.1
0.15
0.2
0.25

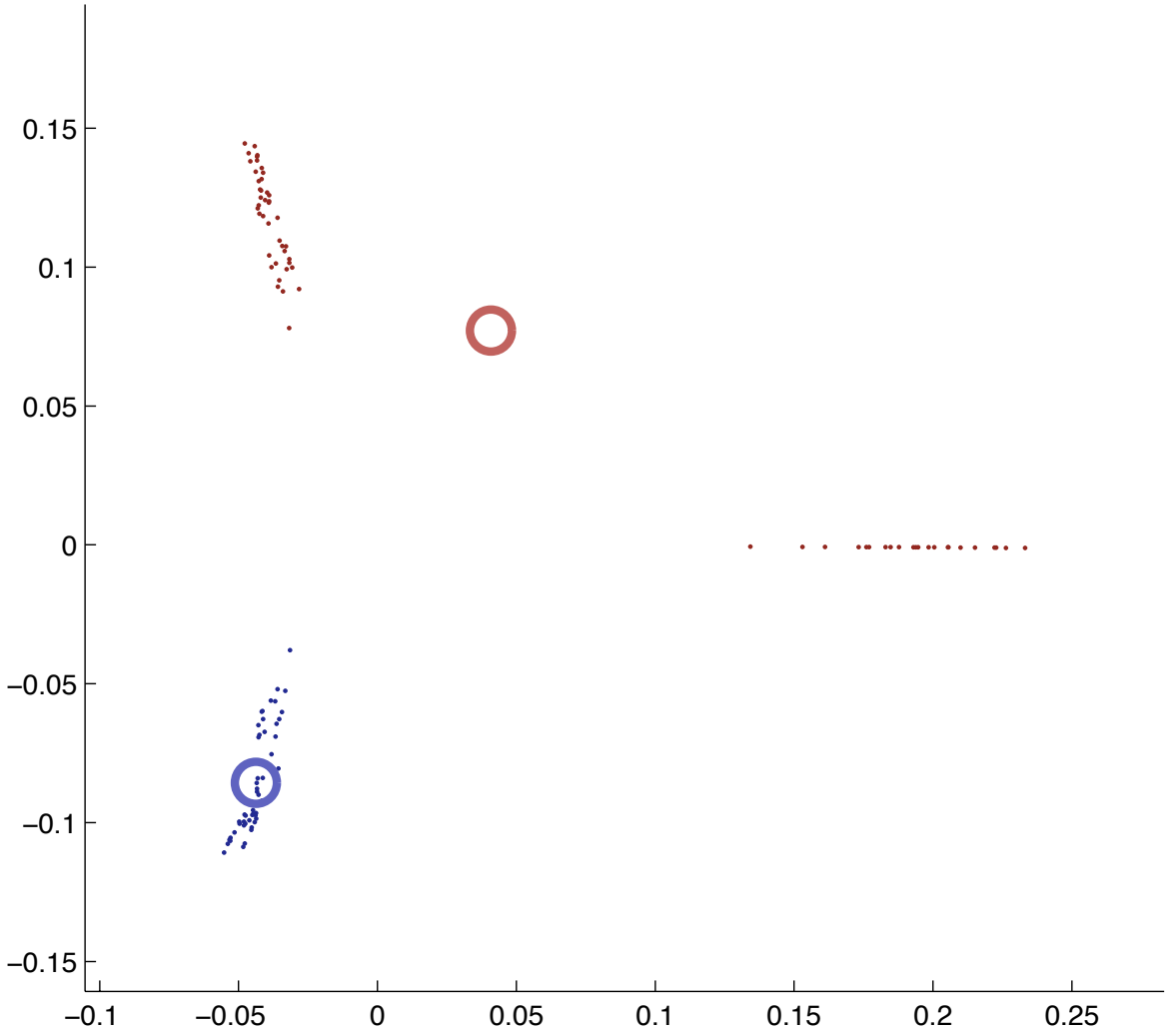
Example



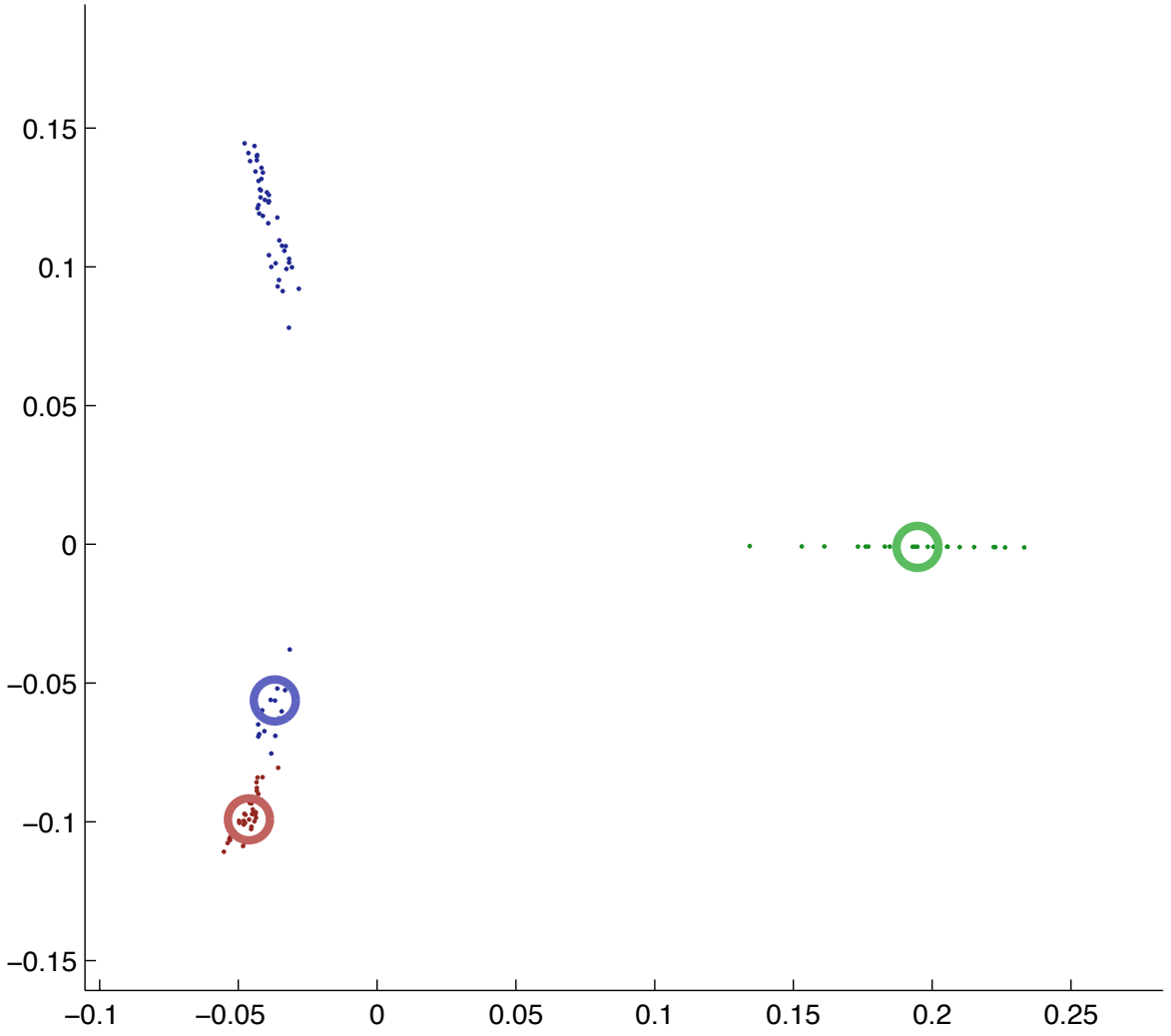
Example 2



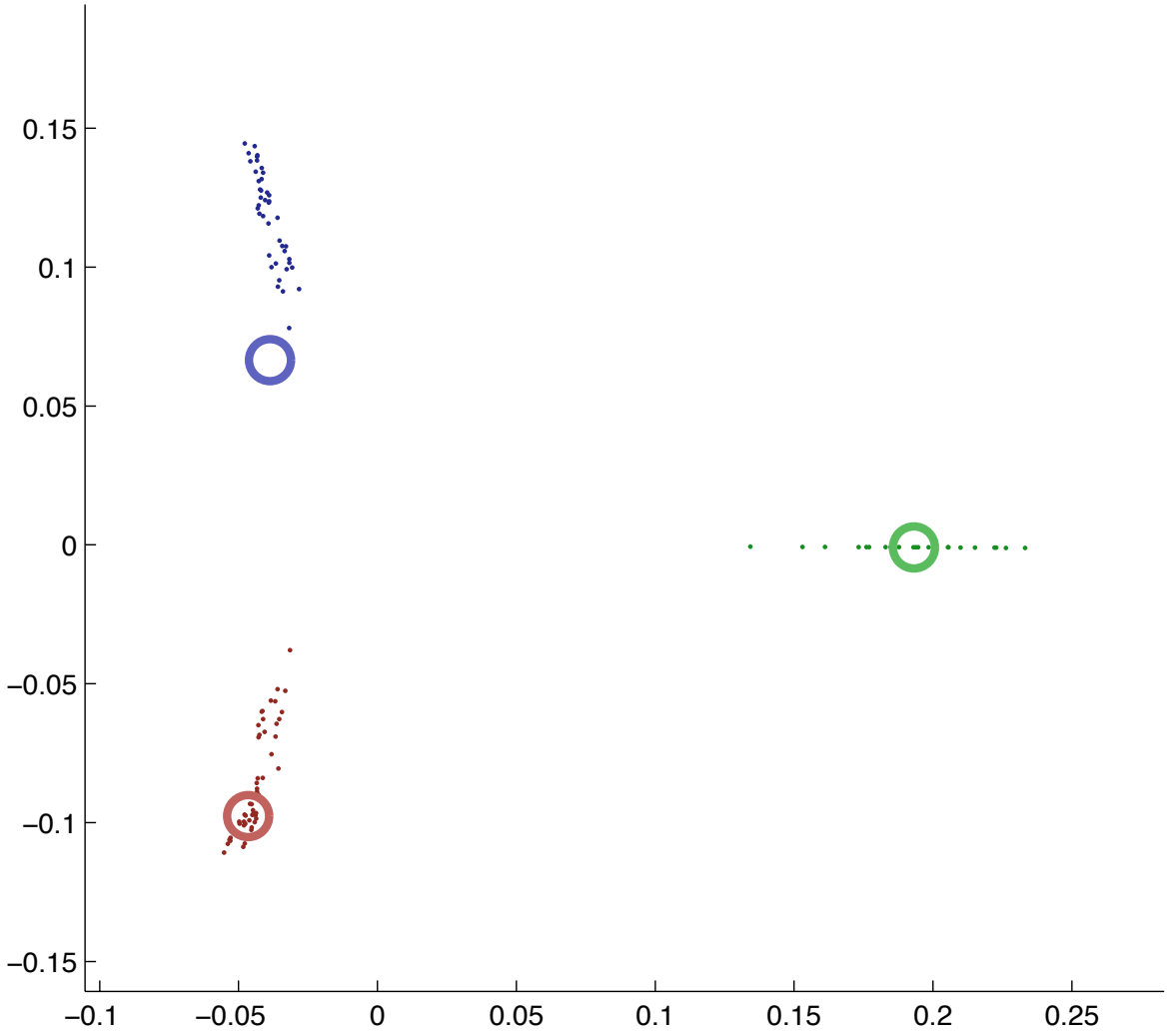
Example 2



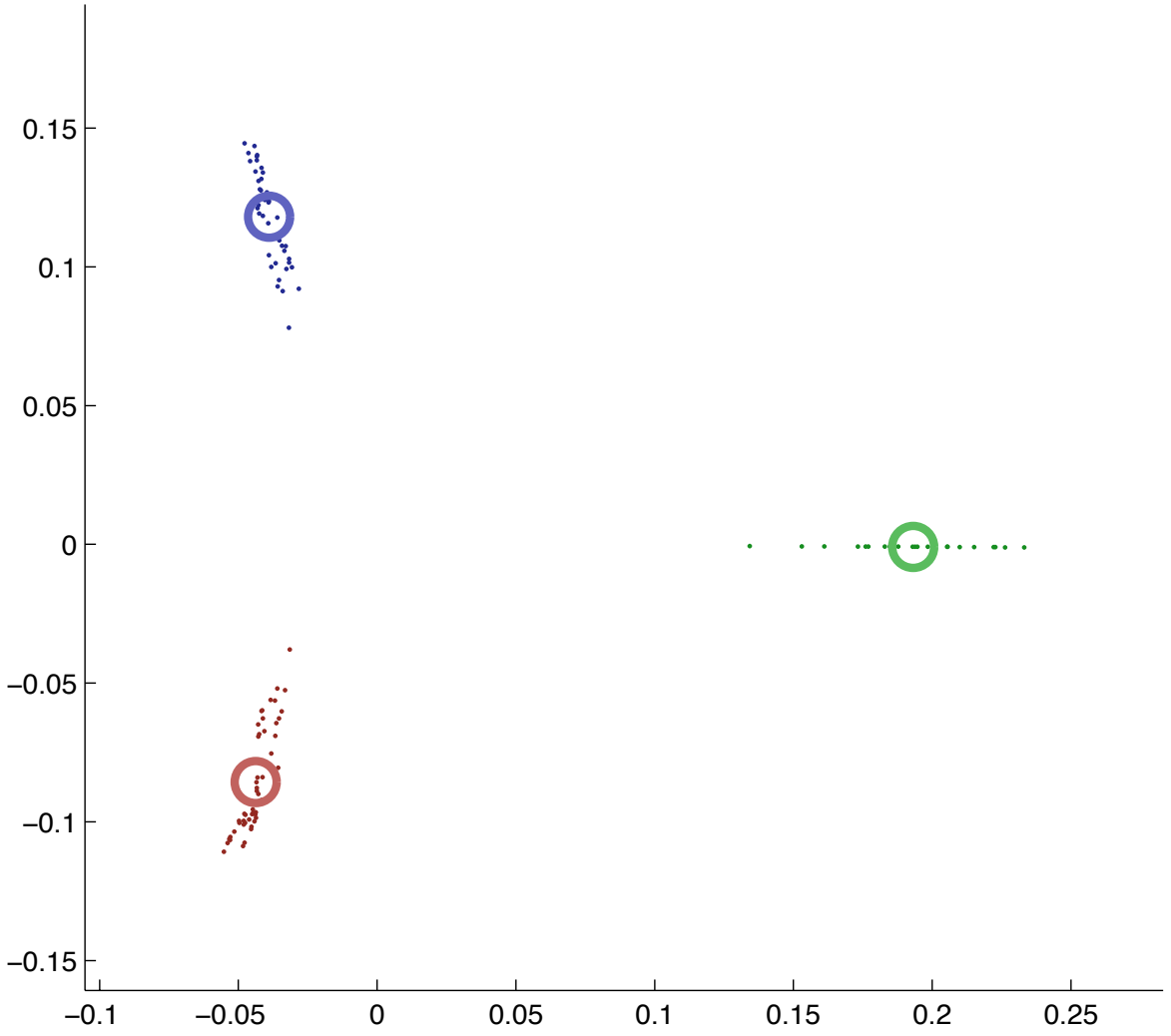
Example 3

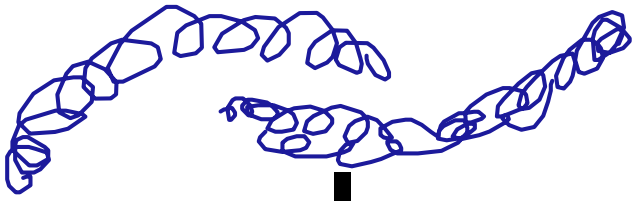


Example 3



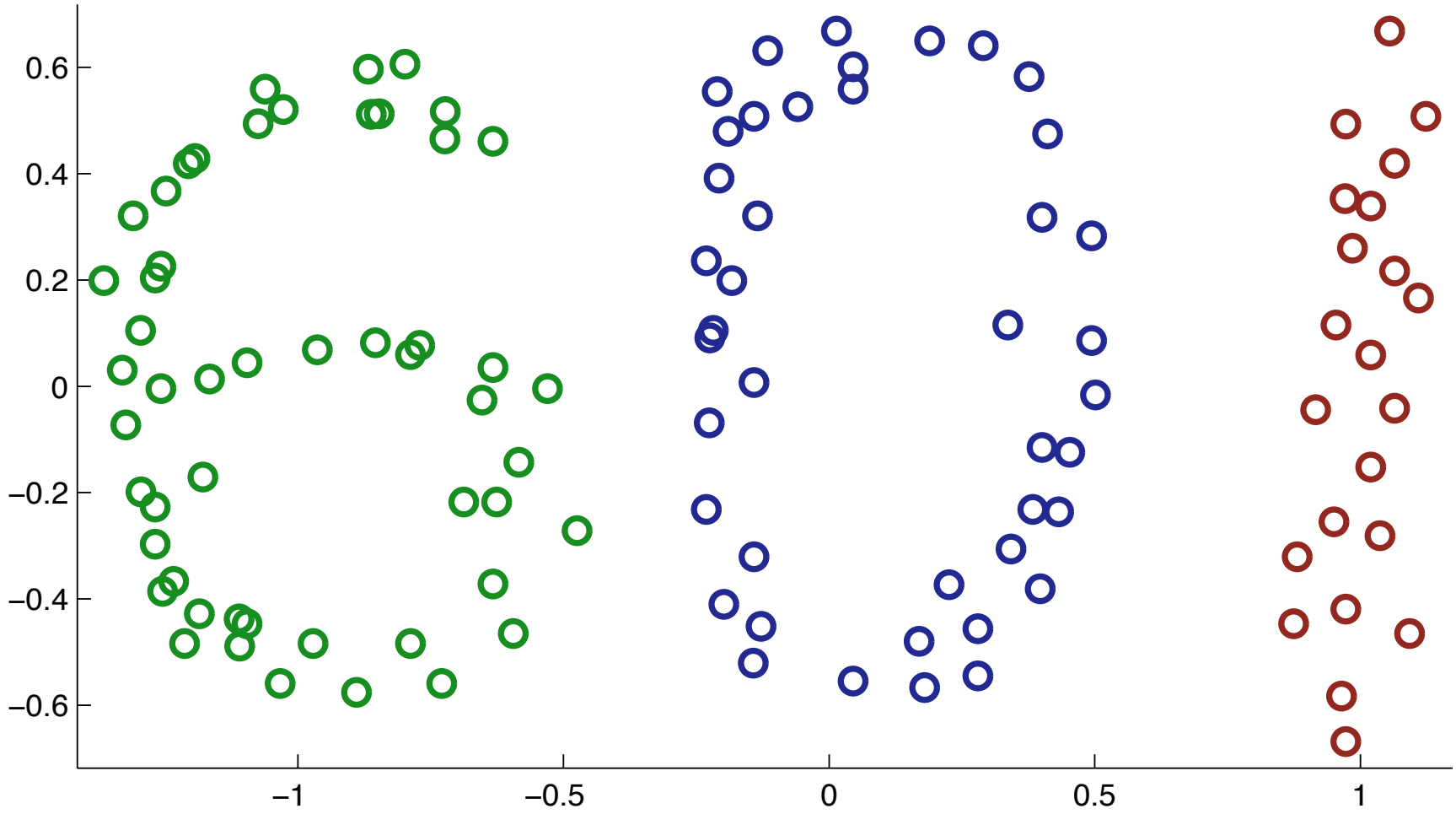
Example 3



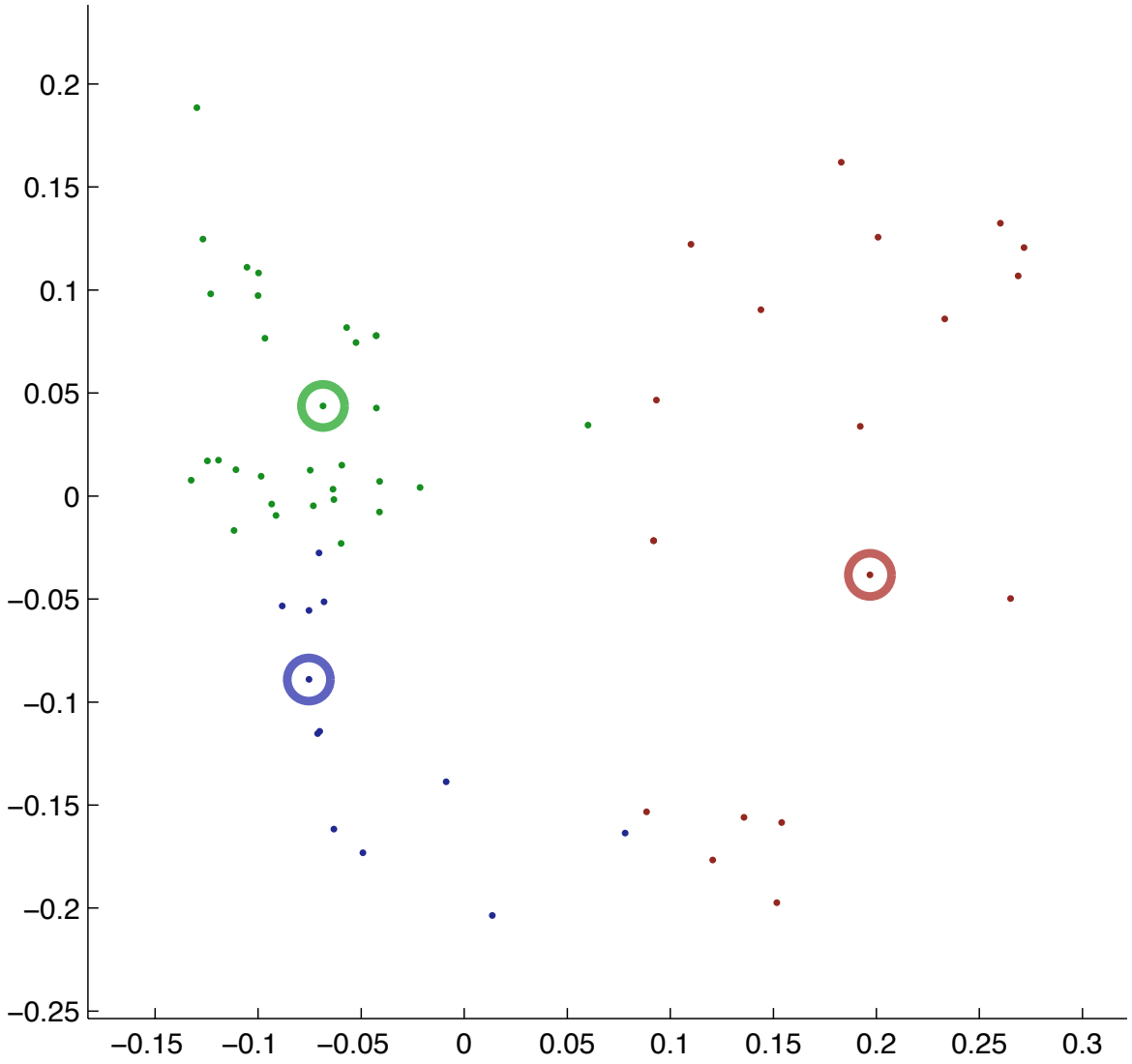


← example that's hard for L_2 dist., but easy after spectral embedding

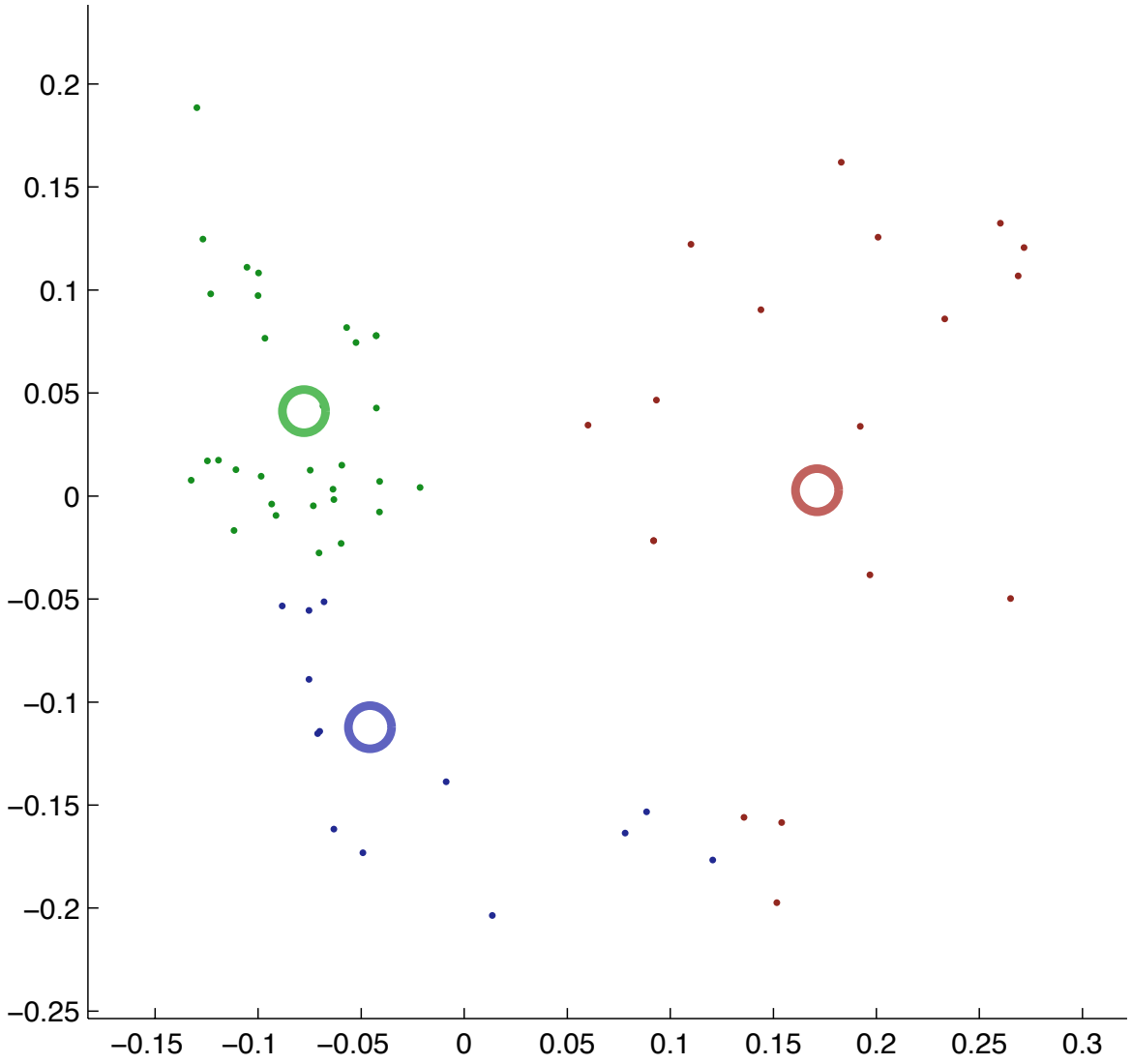
In original space



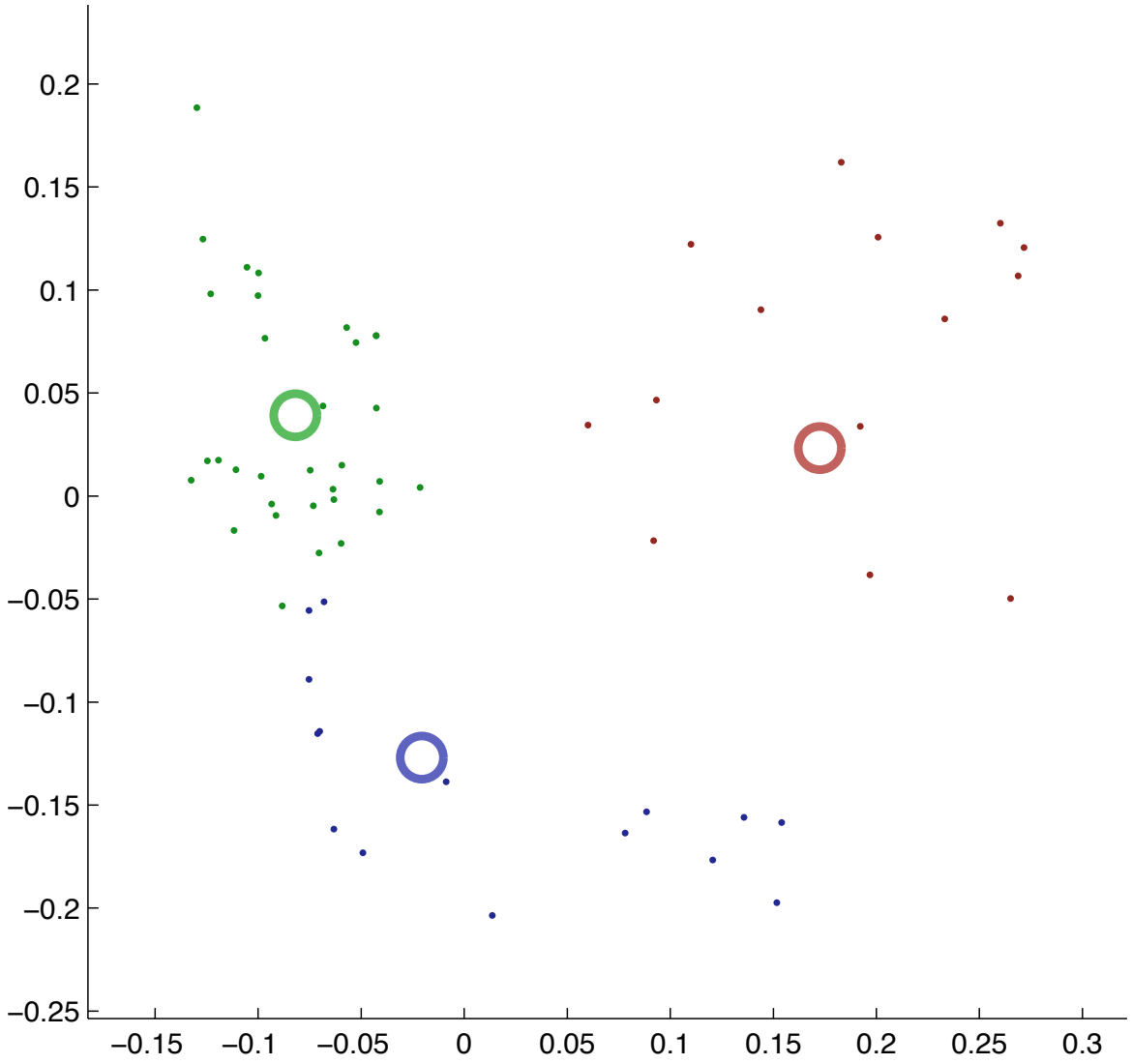
Example 4



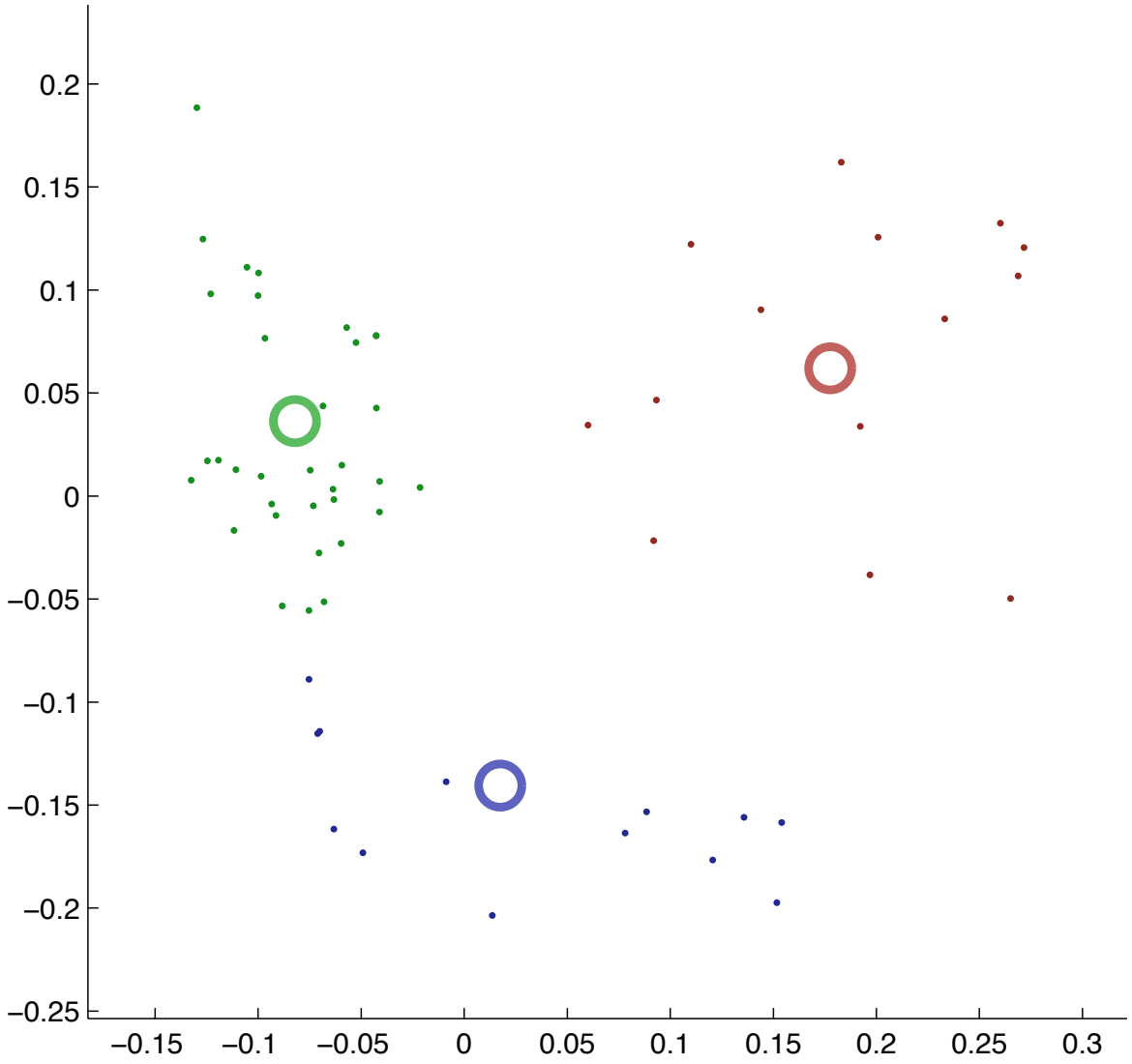
Example 4



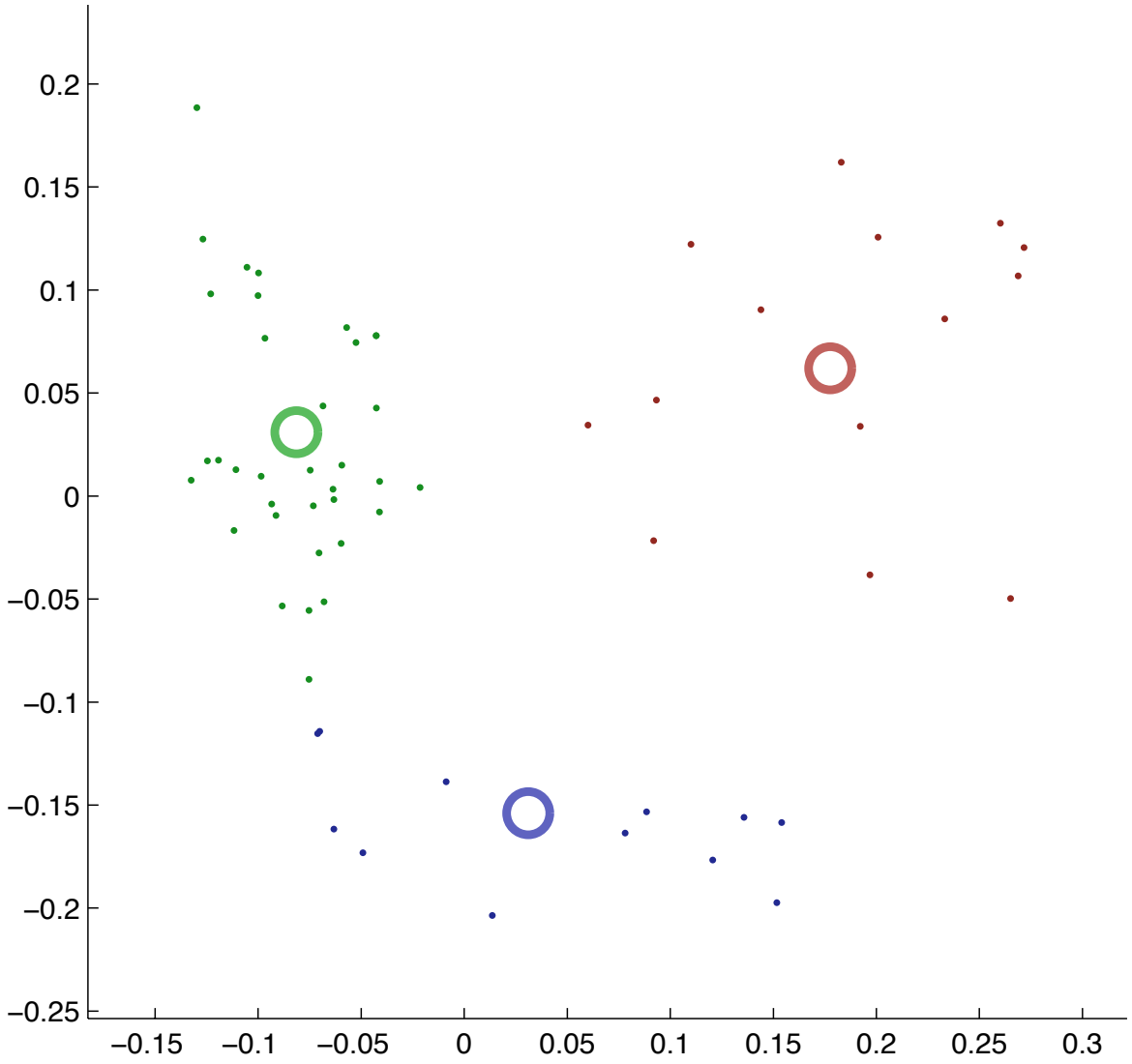
Example 4



Example 4



Example 4



Optimization strategies

- Initialization:
  - want to get a "diverse" set of starting centers
  - simplest: select K at random
  - smarter: select too many, then remove centers that are close to others
  - or: select one at a time by maximizing minimum distance to previous centers
- Optimization:
  - want to avoid local optima
  - split / merge
  - multiple restarts

Beyond k-means

- Suppose:
  - $P(X_i \mid Z_{ij} = 1, \theta) = \text{Gaussian}(\mu_j, \sum_j)$
  - $P(Z_{ij} = 1 \mid \theta) = p_j$
- Mixture of Gaussians
- Leads to an algorithm called "soft" k-means or "fuzzy" k-means
  - vs. "hard" k-means

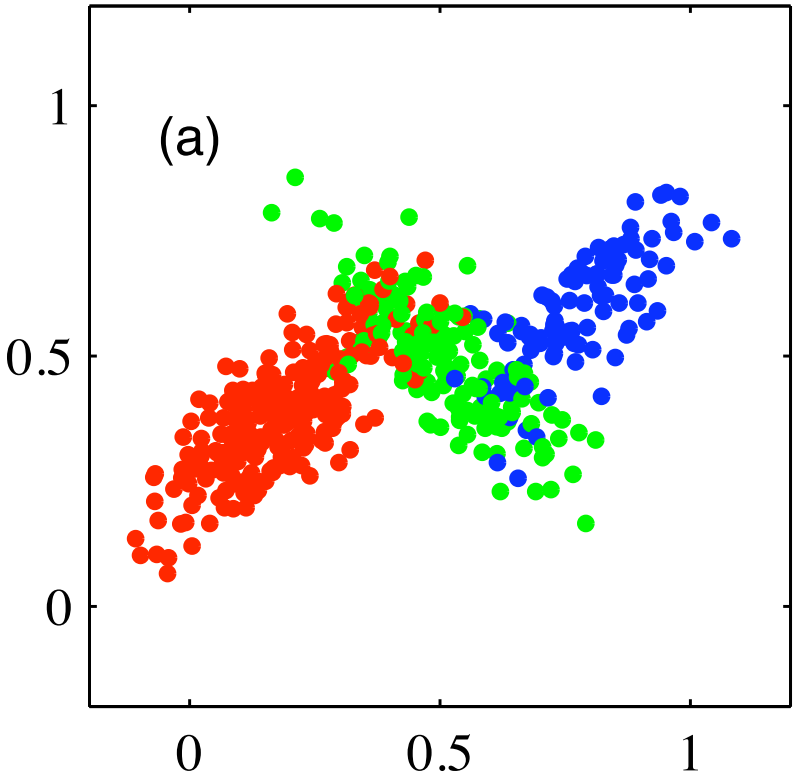
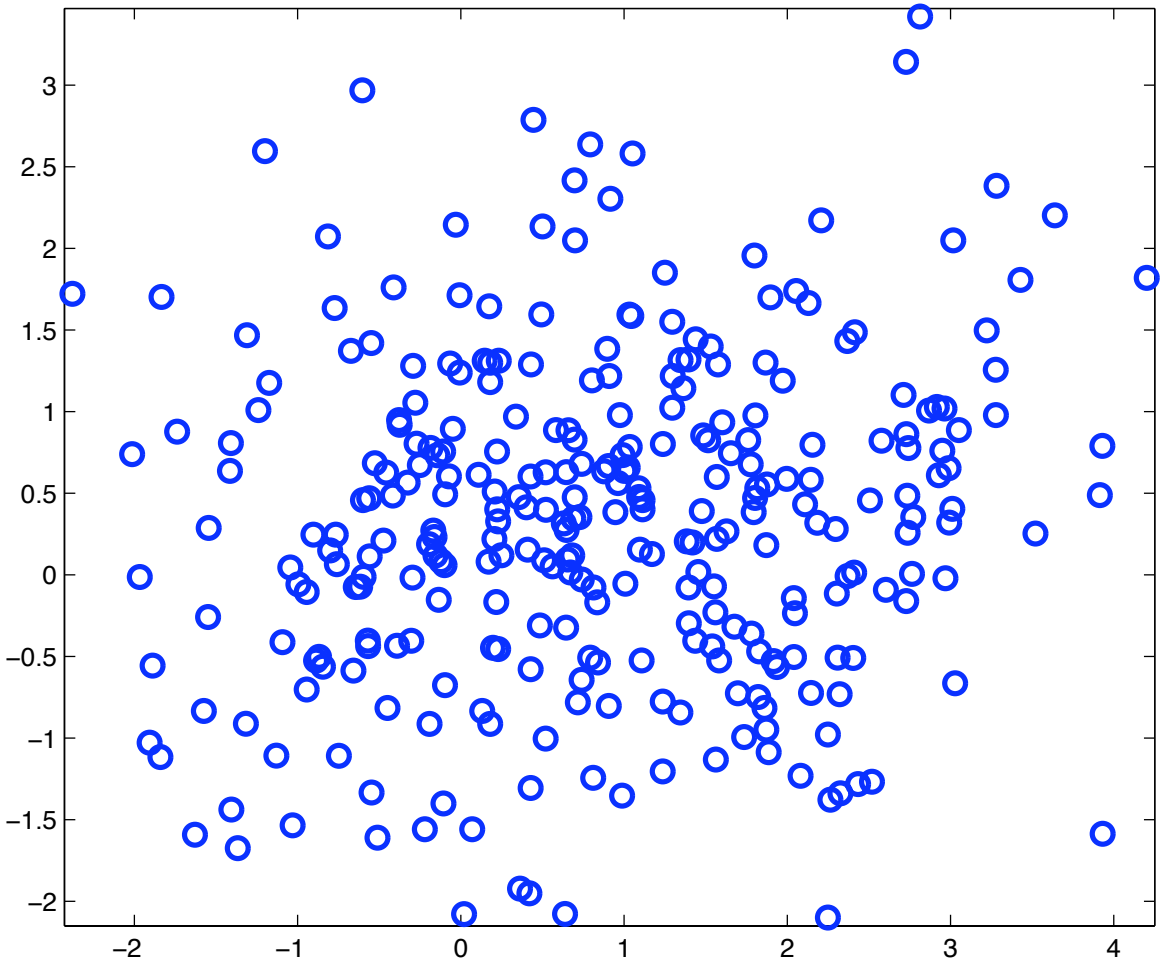


fig 9.5a from Bishop

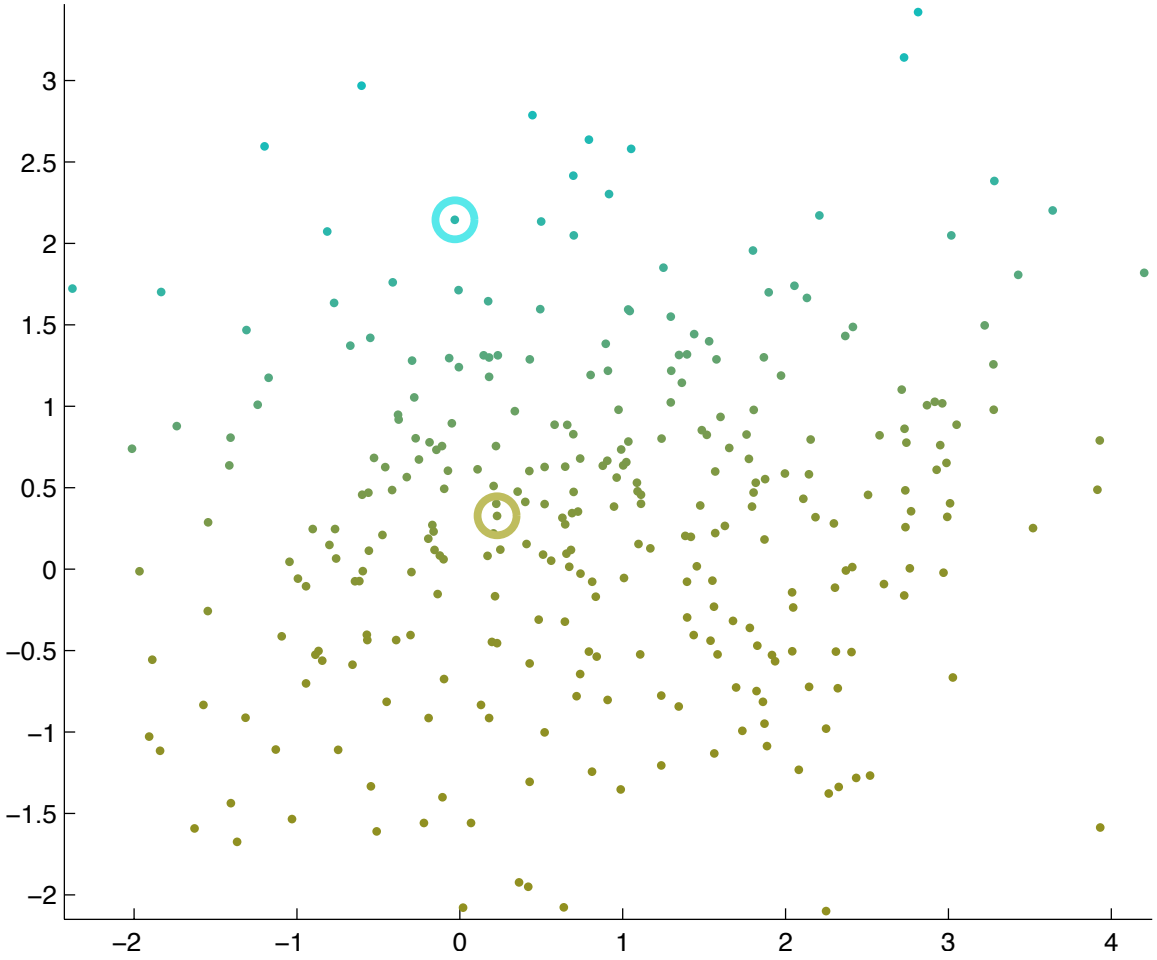
"Soft" k-means

- Find soft assignments: "E step"
  - $q_{ij} = E(z_{ij} | X, \theta) = P(z_{ij}=1 | X, \theta) \propto P(x_i | z_{ij}=1, \theta)$
    $= N(x_i | \mu_j, \Sigma_j)$
- Update means: "M step"
  - $\mu_j = \sum_i q_{ij} x_i \Big/ \sum_i q_{ij}$
- Possibly: update covariances
  - $\Sigma = \sum_i \sum_j q_{ij} (x_i - \mu_j)(x_i - \mu_j)^T / N$
- Repeat

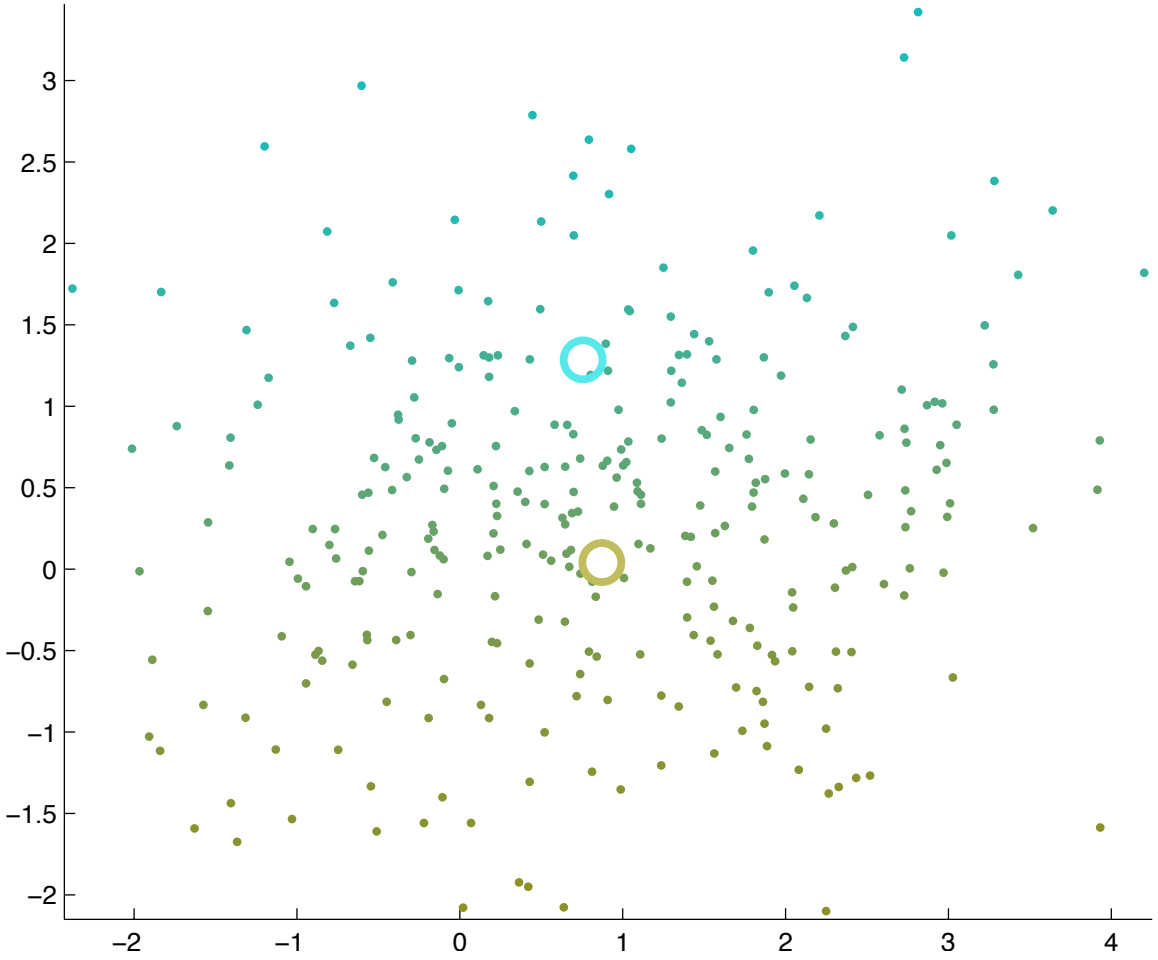
Example



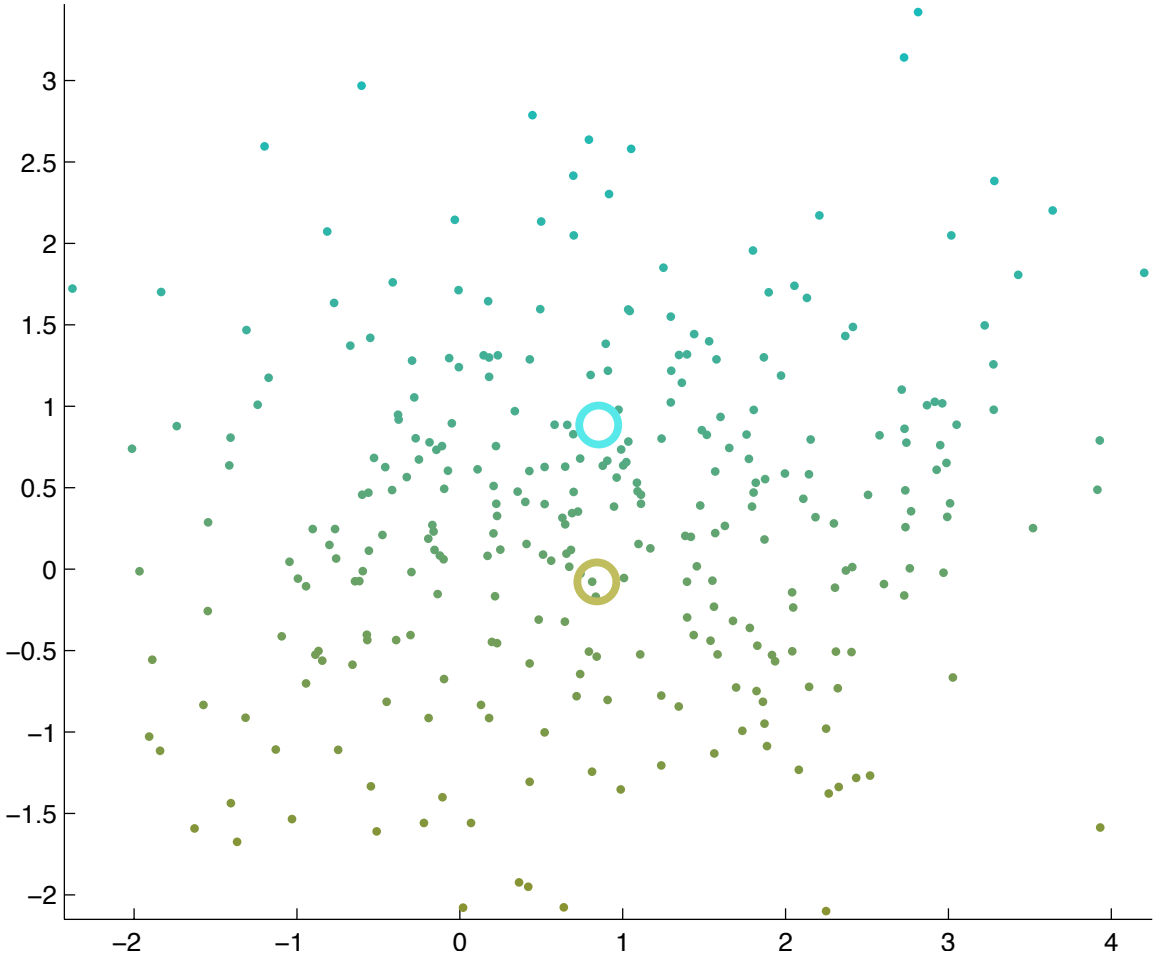
Example



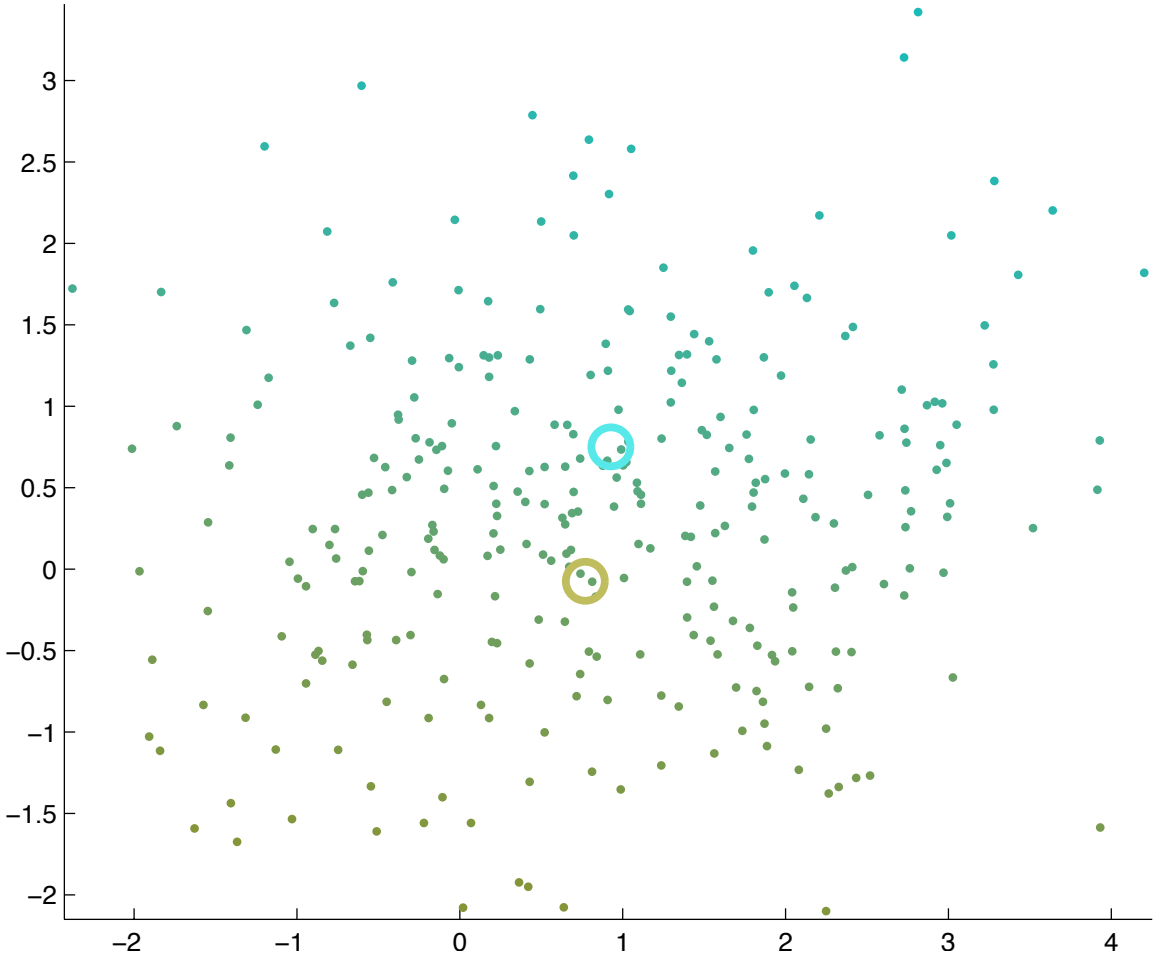
Example



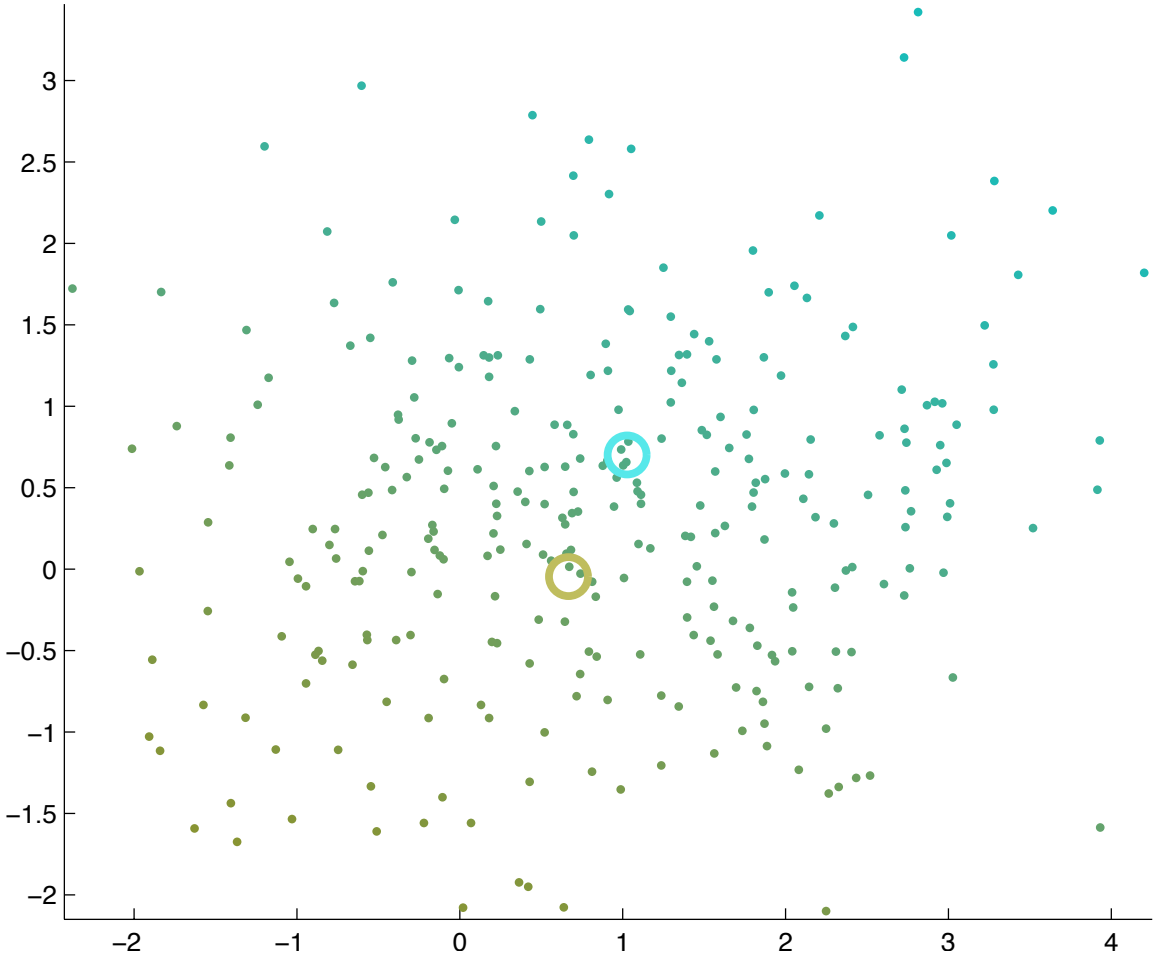
Example



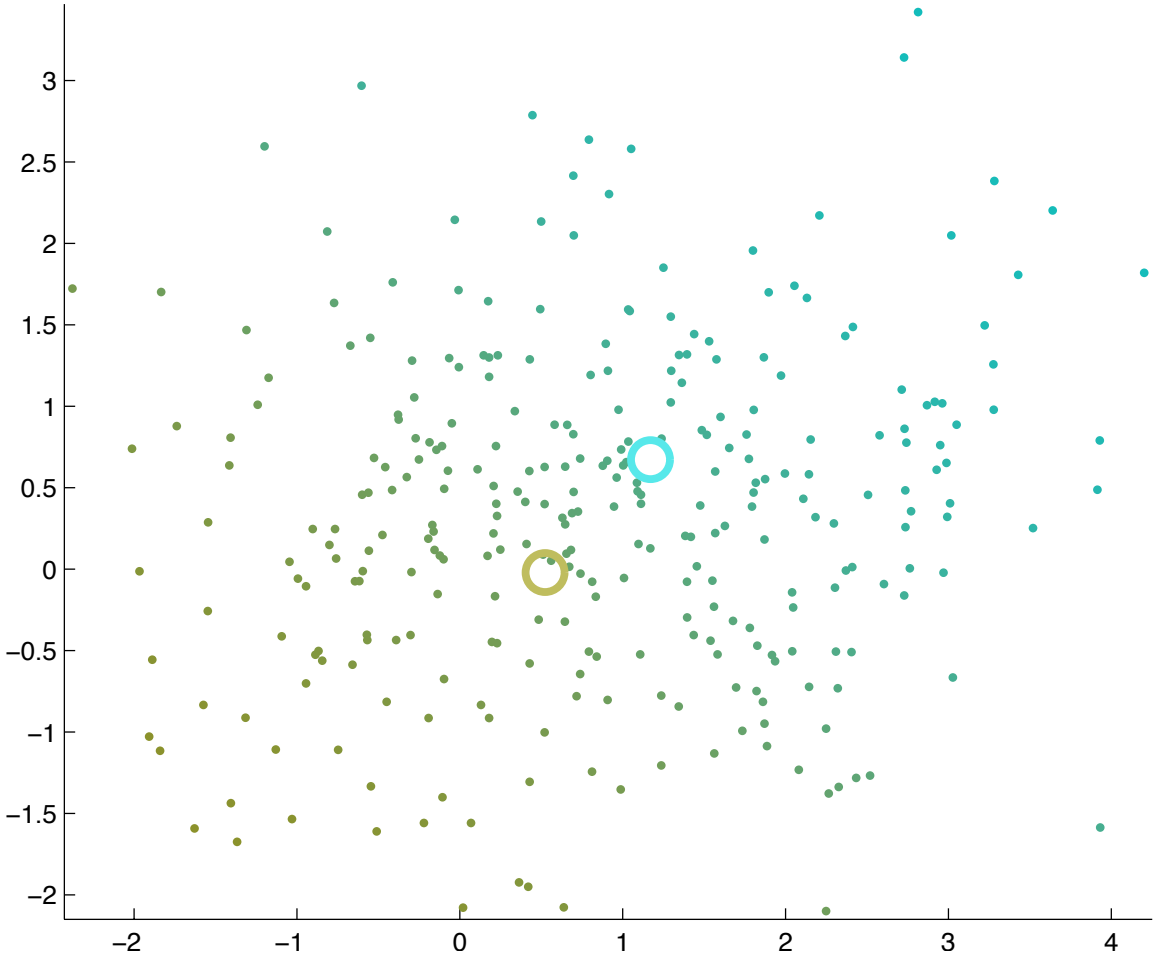
Example



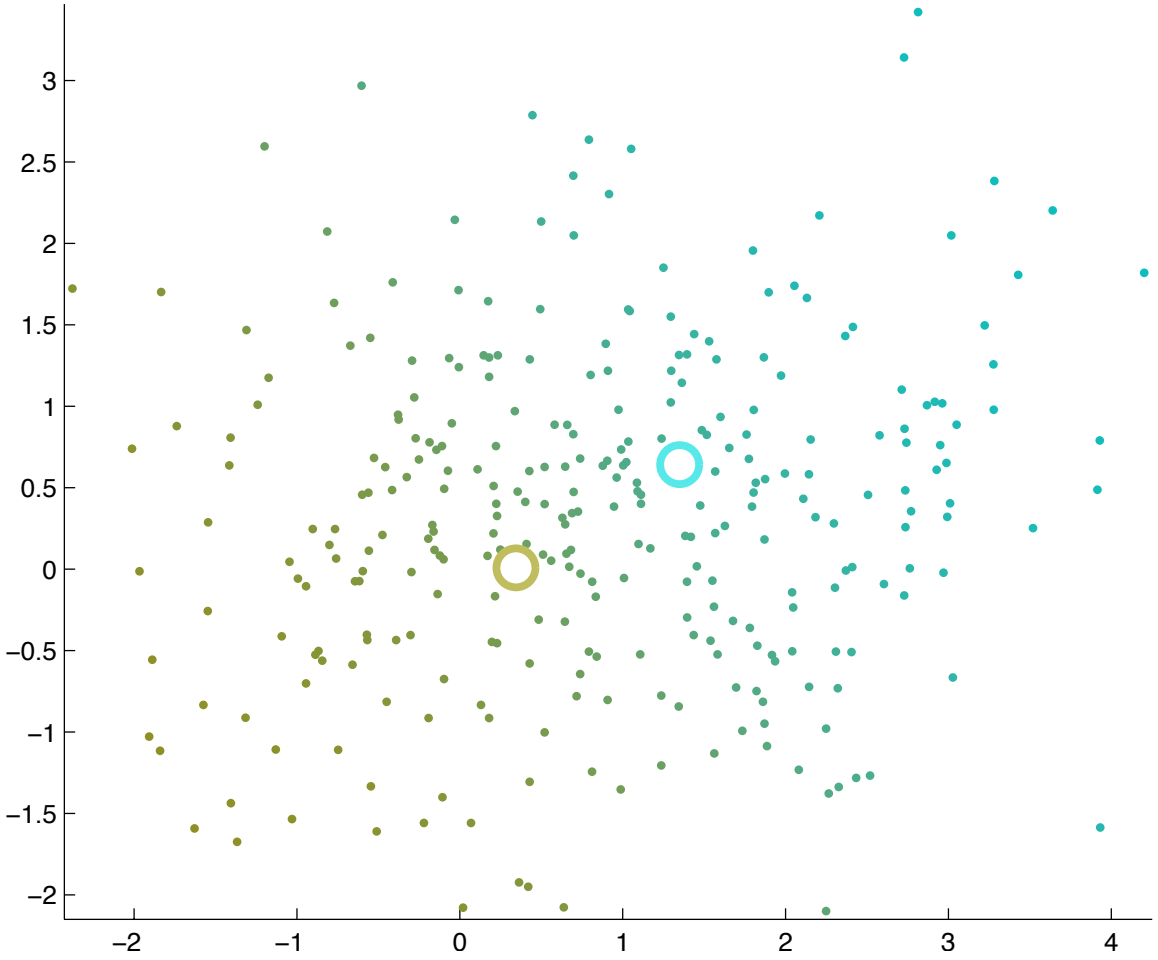
Example



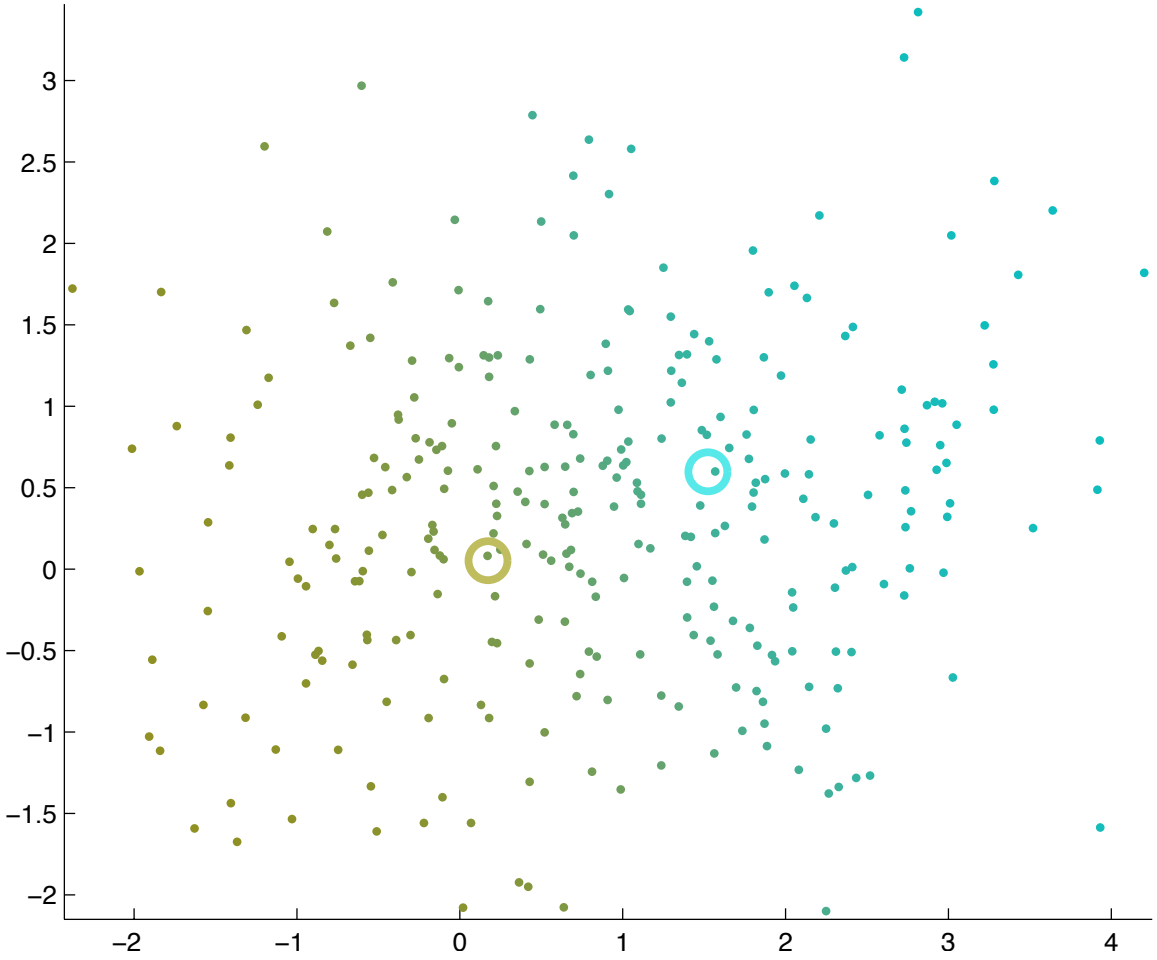
Example



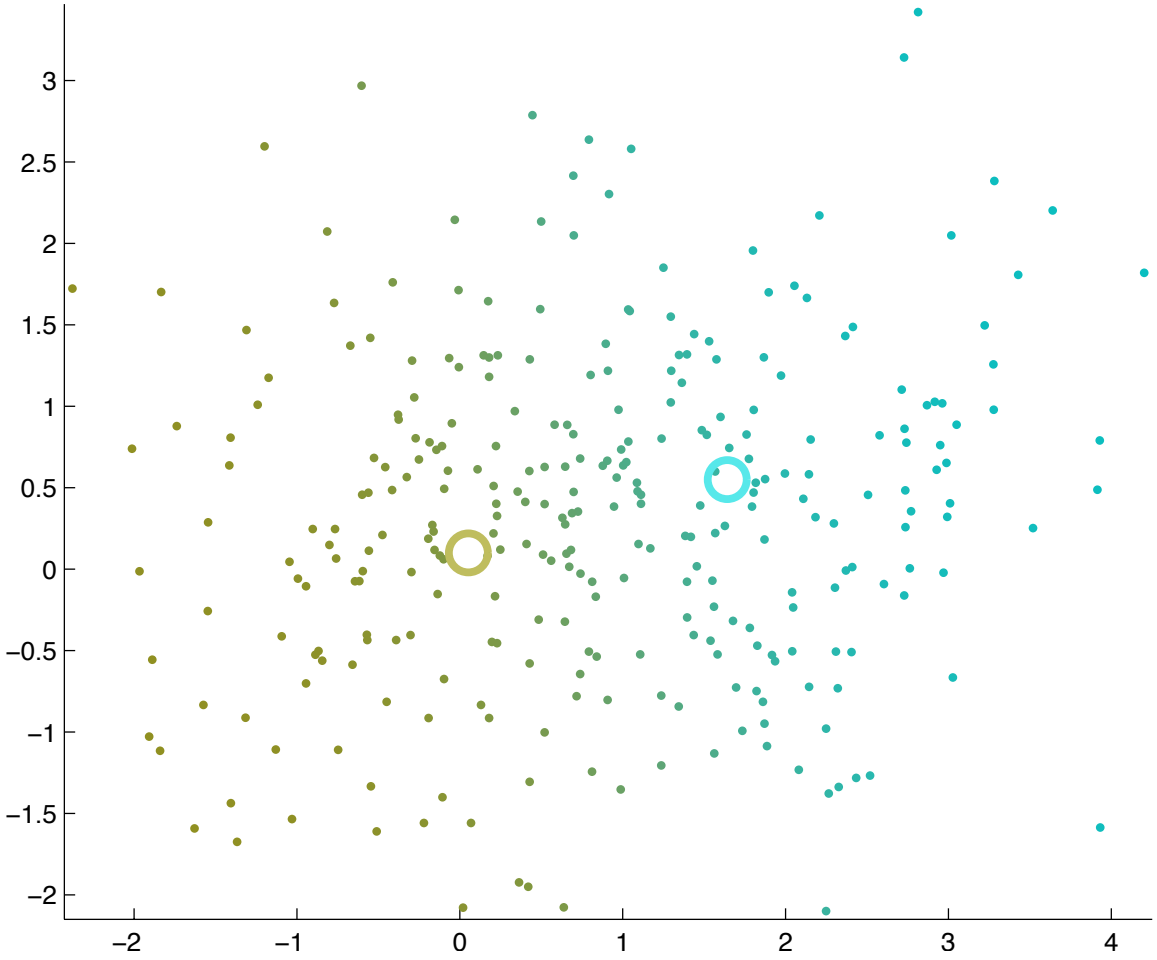
Example



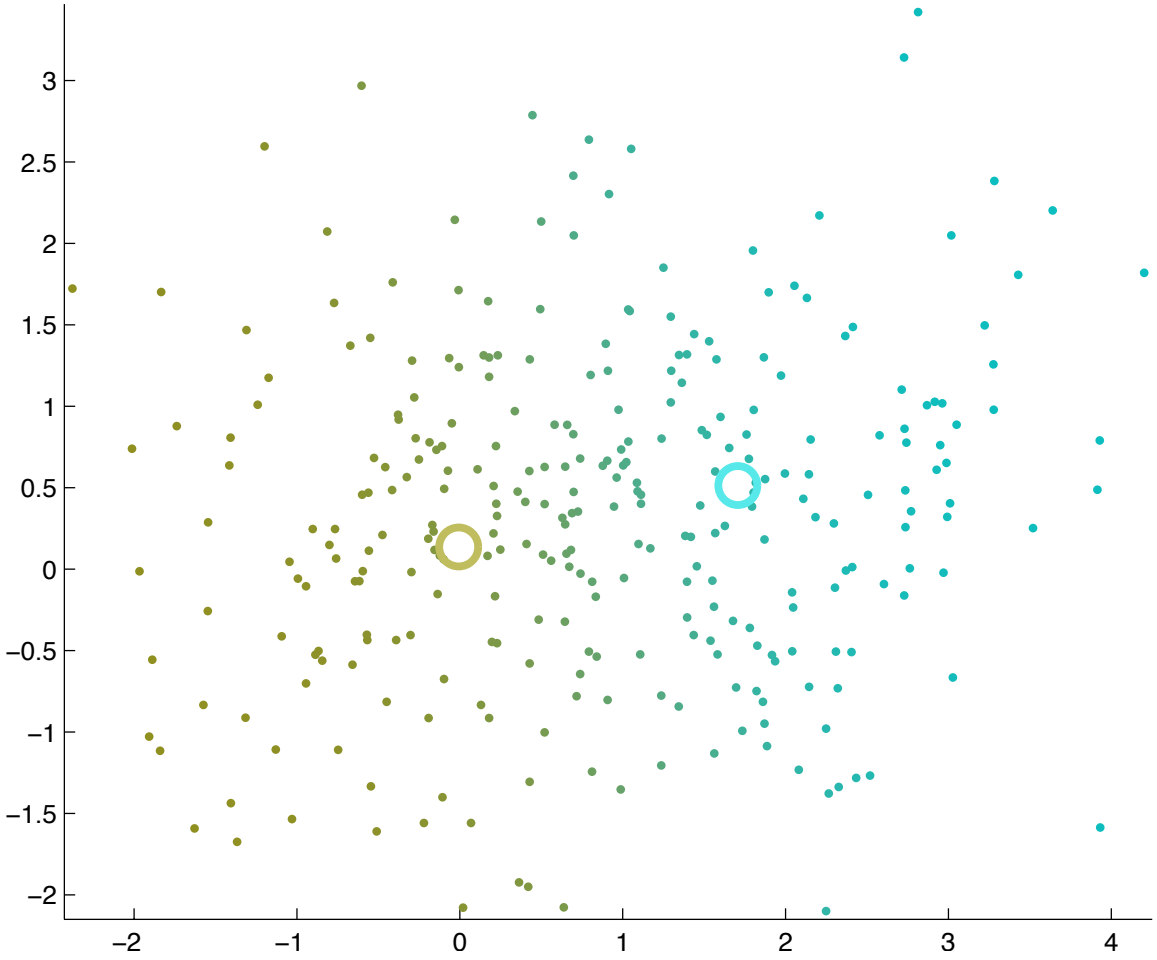
Example



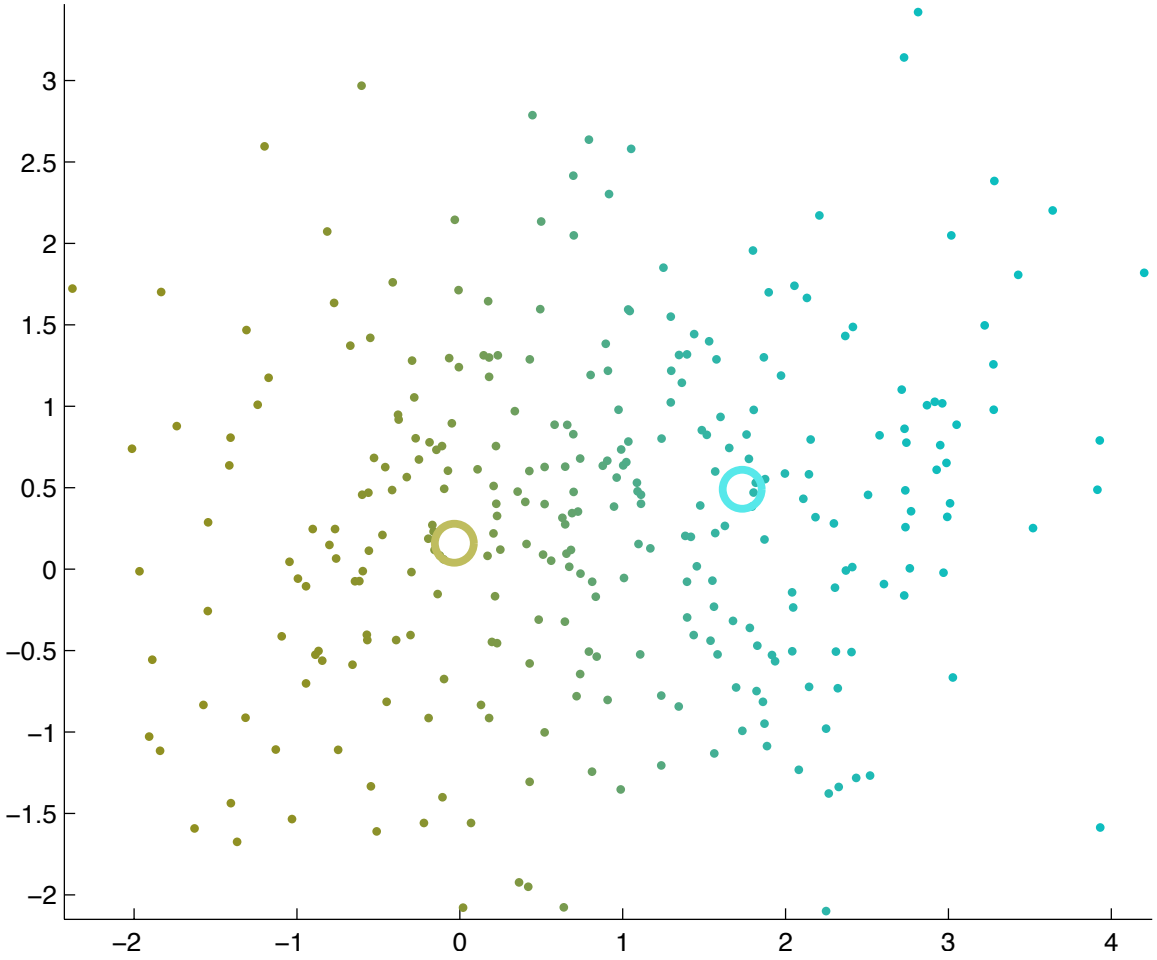
Example



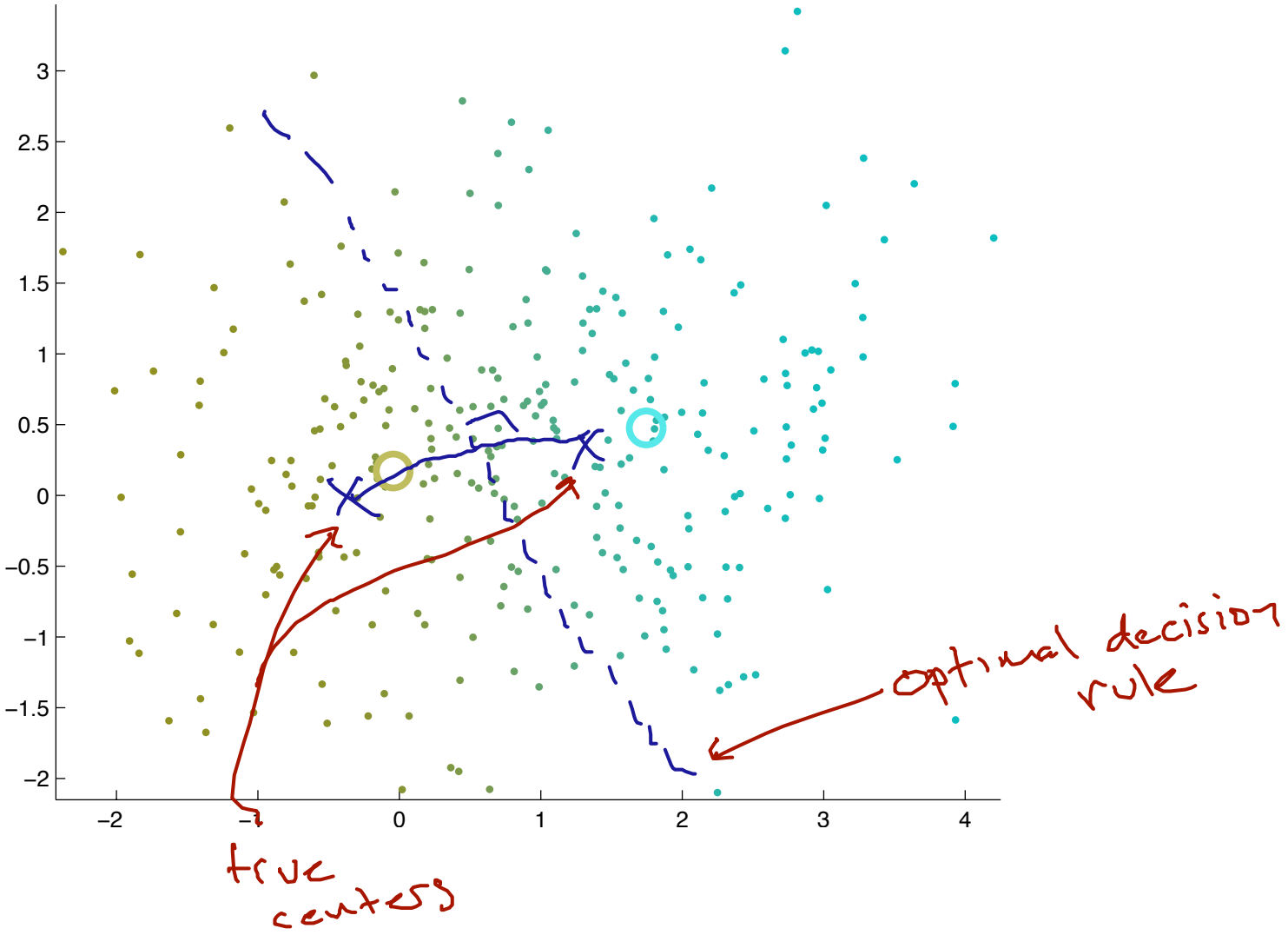
Example



Example

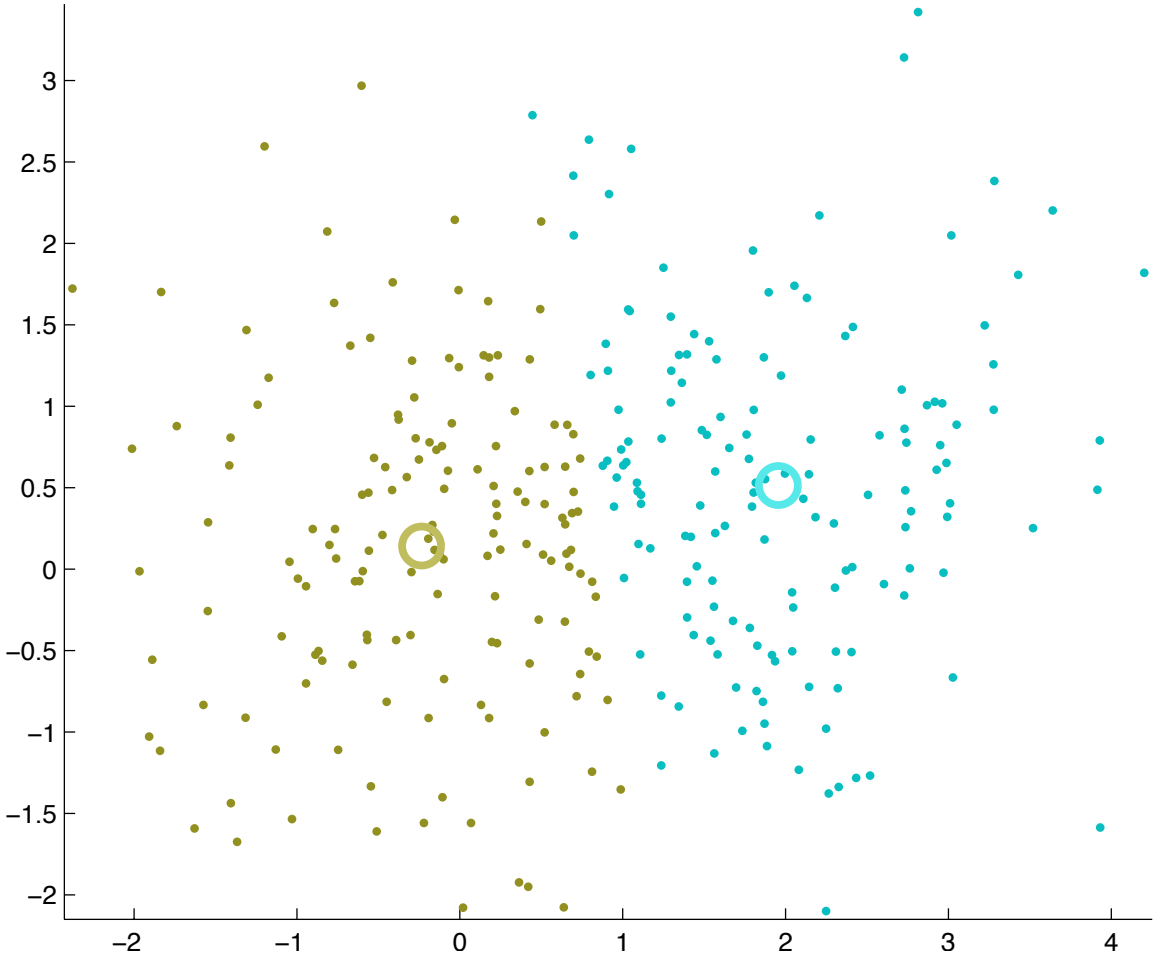


Example


3
2.5
2
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−2
−1
0
1
2
3
4
optimal decision rule
true centers

Comparison: "hard" k-means

Note bias



Comparison

- Soft k-means:
  - converges slower than hard k-means
  - more expensive per iteration
  - but asymptotically unbiased (unlike hard k-means), ***if*** we restart often enough
- Often, can initialize soft w/ hard

Hard k-means is a special case

- Let: $\Sigma_j = \sigma^2 I \quad \sigma \to 0$
- Then: $q_{ij} \to 0$ or 1
- E-step: assign x_i to closest μ_j
- M-step: move μ_j to mean of assigned x_i

Deriving soft k-means

- Soft k-means is an example of an ***EM algorithm***
  - general strategy for MLE or MAP with ***hidden variables*** (in our case, Z_{ij})
- We'll derive soft k-means first, then generalize

Deriving soft k-means

- $P(X_i \mid Z_{ij} = 1, \theta) = \text{Gaussian}(\mu_j, \Sigma_j)$ → p_j, μ_j, Σ_j
- $P(Z_{ij} = 1 \mid \theta) = p_j$
- $P(X_i, Z_{i\cdot} \mid \theta) = \prod_j p_j^{z_{ij}} N(x_i \mid \mu_j, \Sigma_j)^{z_{ij}}$
- $L = \ln P(X \mid \theta) = \ln \sum_{z \in Z} P(X, z \mid \theta)$

$$= \ln \sum_z \exp(\ln P(X, z \mid \theta))$$

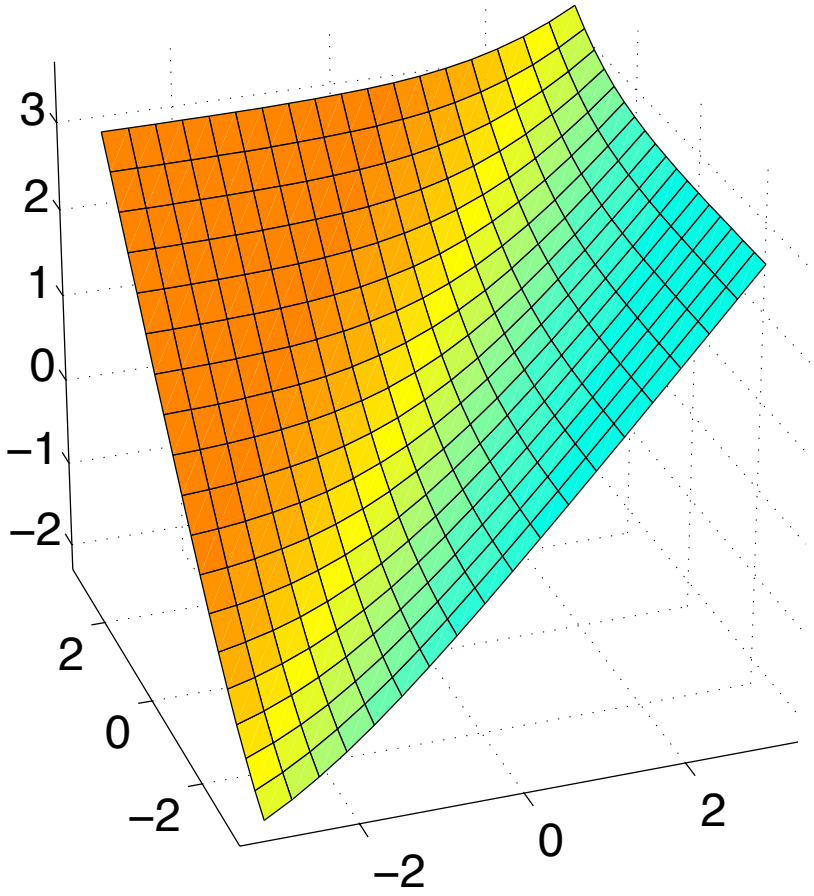
$$= \ln \sum_z \exp\left(\sum_i \sum_j z_{ij} [\ln p_j + \ln N(x_i \mid \mu_j, \Sigma_j)] \right)$$

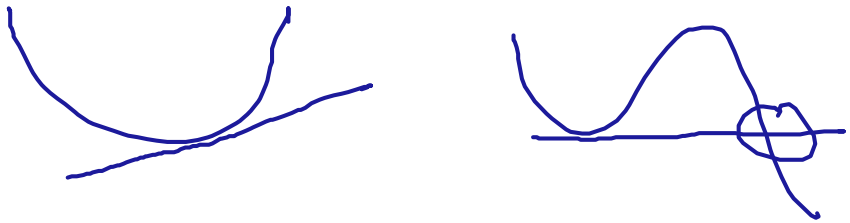
soft max

$\approx \max(x,y)$

$f(x,y) = \ln(e^x + e^y)$

← convex





Convex fns are lower-bounded by tangents