Clustering

Cynthia Rudin

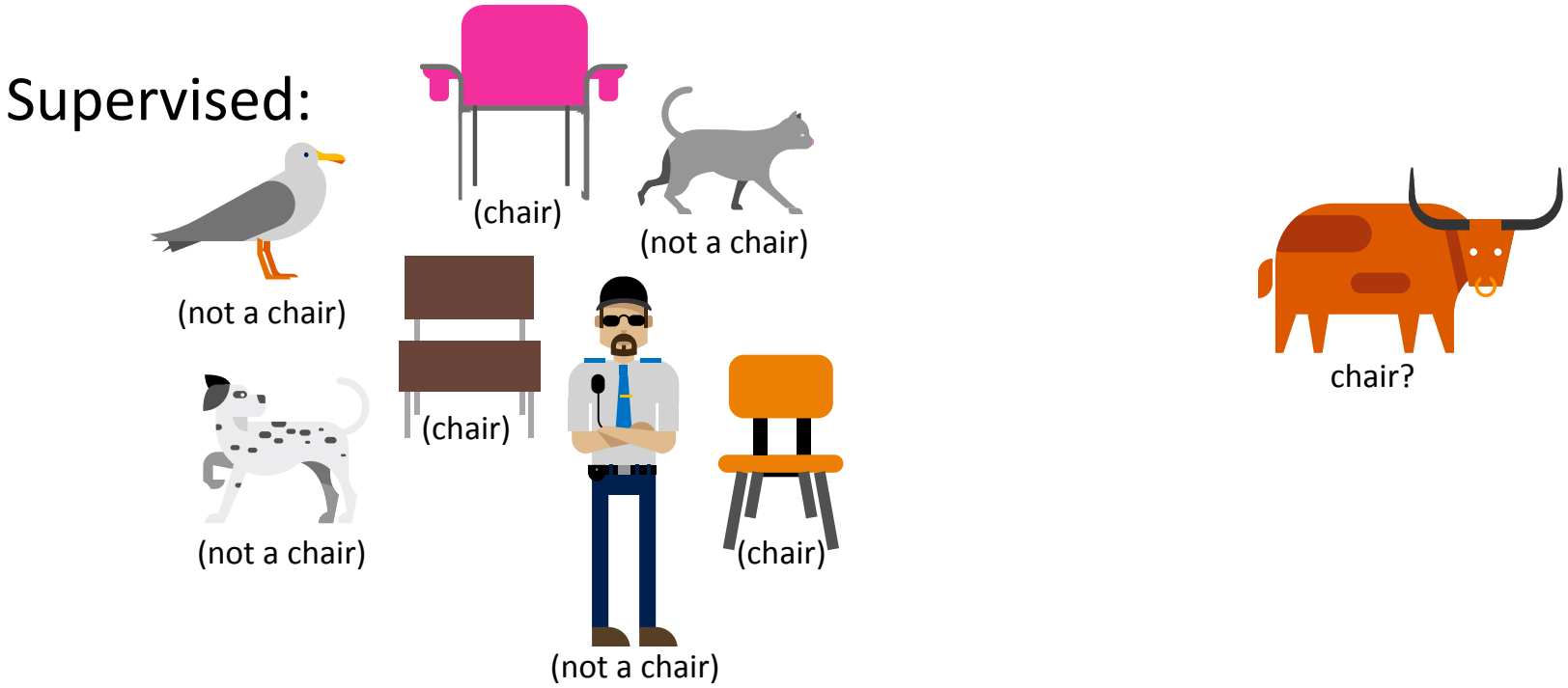
Duke Machine Learning

Clustering

- K-Means clustering algorithm and examples
- K-Means derivation
- Hierarchical agglomerative clustering algorithm and examples
- Comparison

Unsupervised Learning

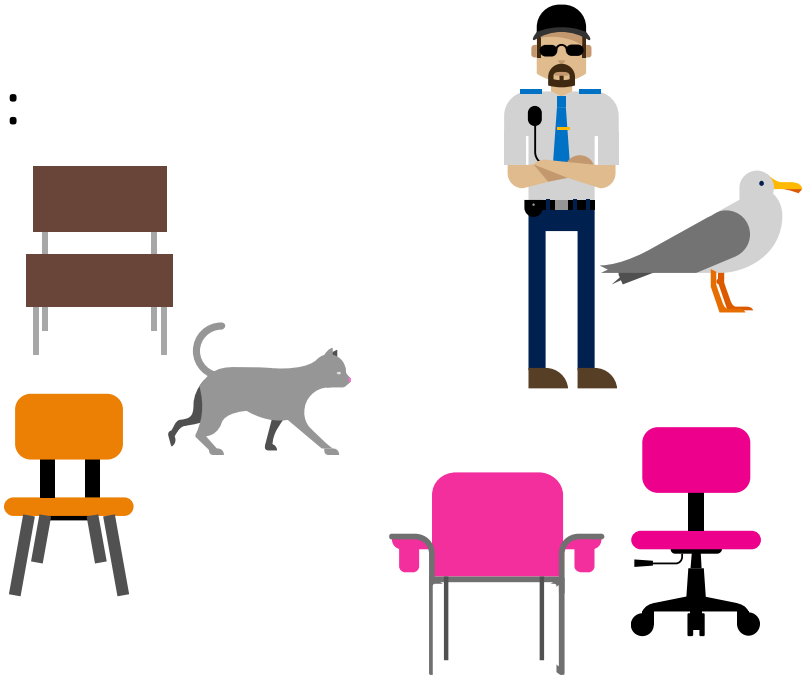
- "Unsupervised" means that the training data has no ground truth labels to learn from.



Unsupervised Learning

- “Unsupervised” means that the training data has no ground truth labels to learn from.

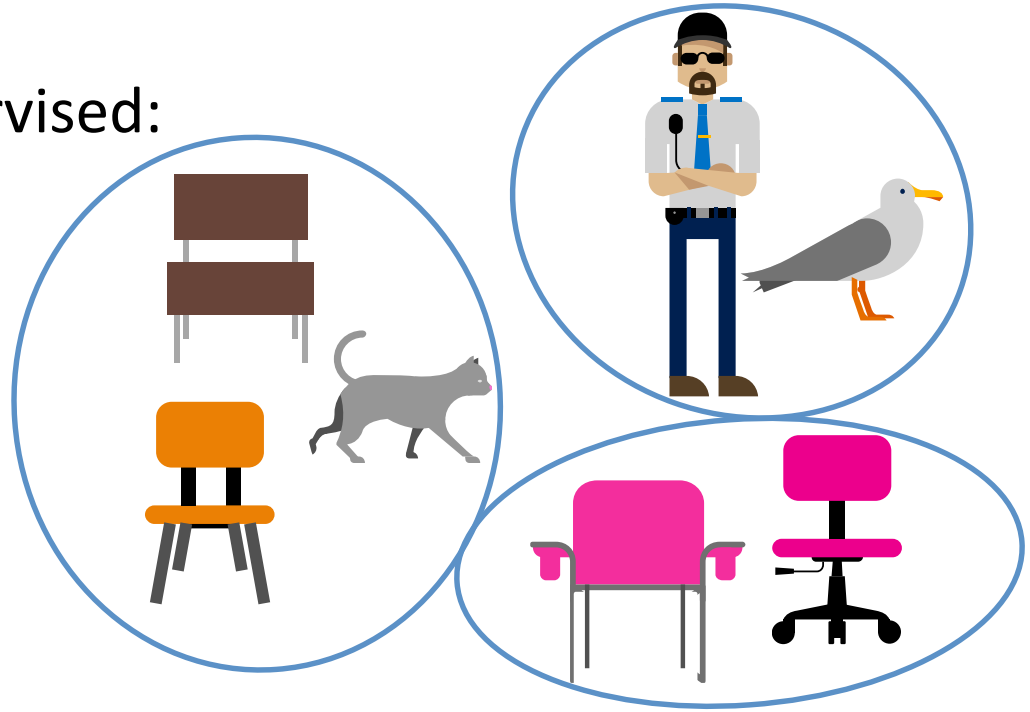
Unsupervised:



Unsupervised Learning

- “Unsupervised” means that the training data has no ground truth labels to learn from.

Unsupervised:

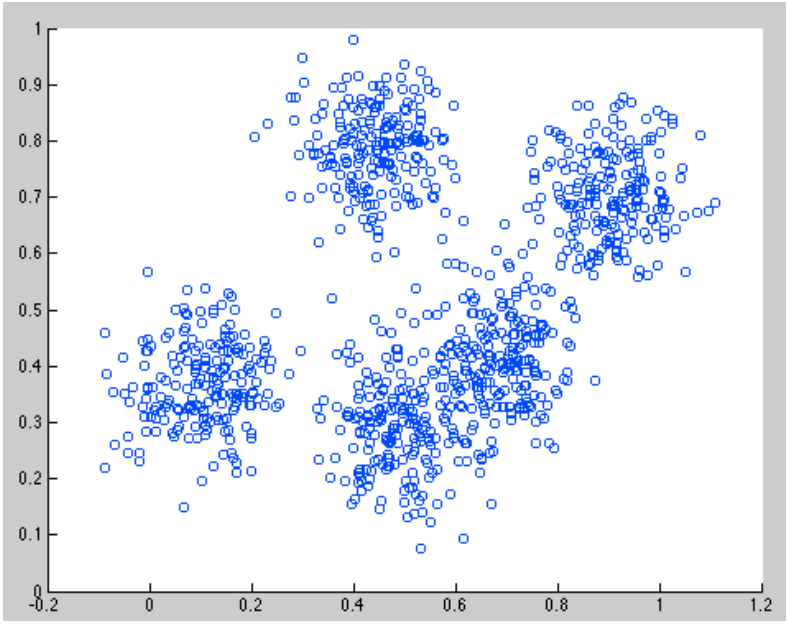


Unsupervised Learning

- “Unsupervised” means that the training data has no ground truth labels to learn from.
- Clustering is an key unsupervised problem.

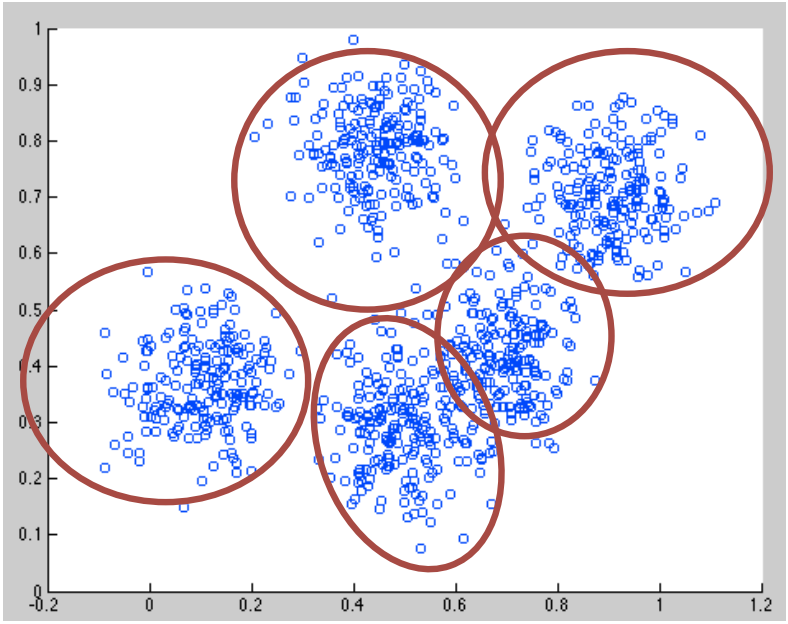
Clustering

- Data within a cluster should be similar to other members of the cluster.



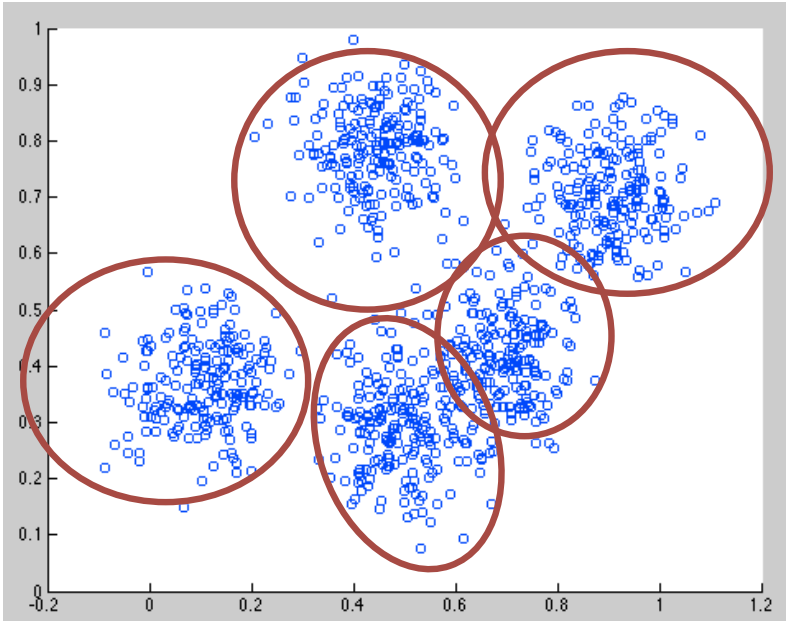
Clustering

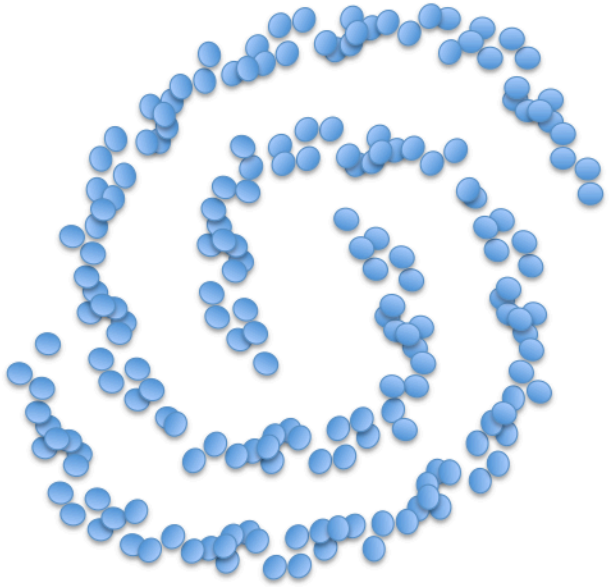
- Data within a cluster should be similar to other members of the cluster.



Clustering

- Data within a cluster should be similar to other members of the cluster.





Clustering

Applications include:

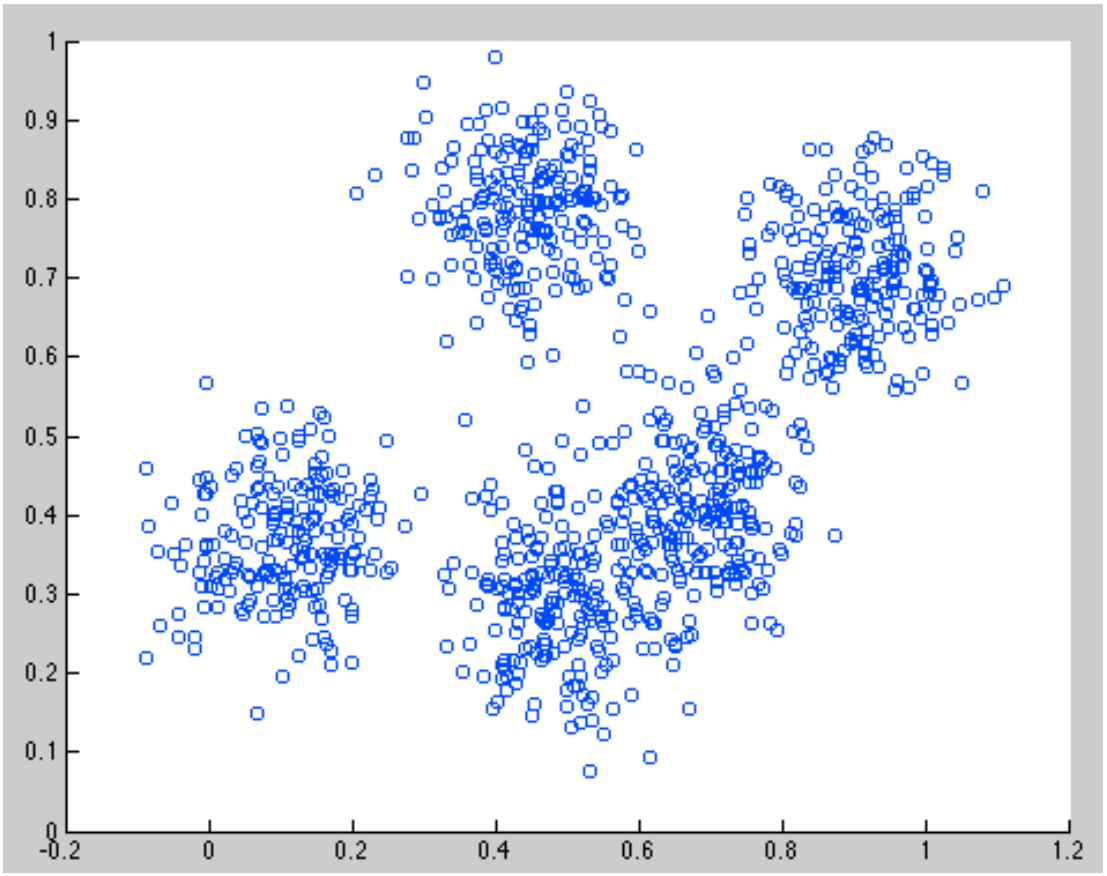
- Automatically grouping documents/webpages into topics
  - For instance, grouping news stories from today into categories
- Clustering large number of products
  - E.g., etsy products
- Clustering customers into those with similar purchase behavior

K-Means Clustering

Clustering

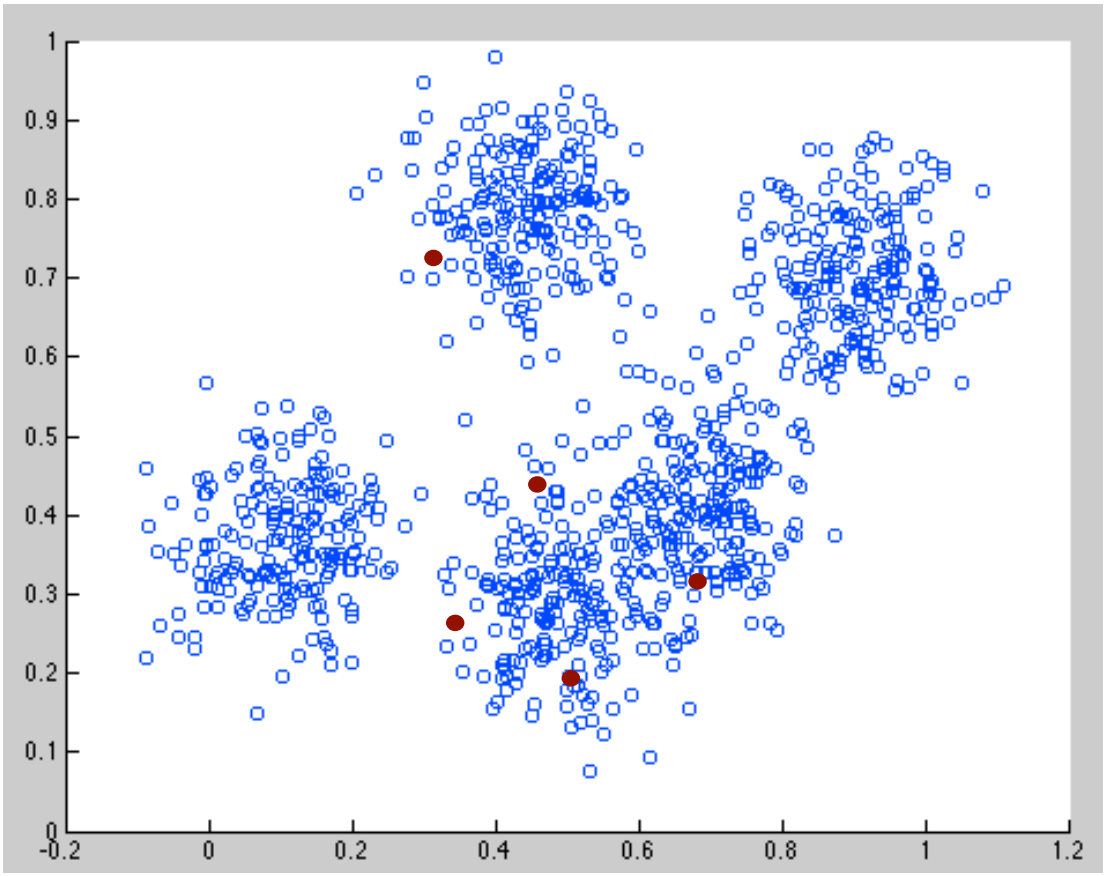
- K-means clustering
  - Input number of clusters, randomly initialize centers
  - Assign all points to the closest cluster center
  - Change cluster centers to be in the middle of its points
  - Repeat until convergence

K-Means in action



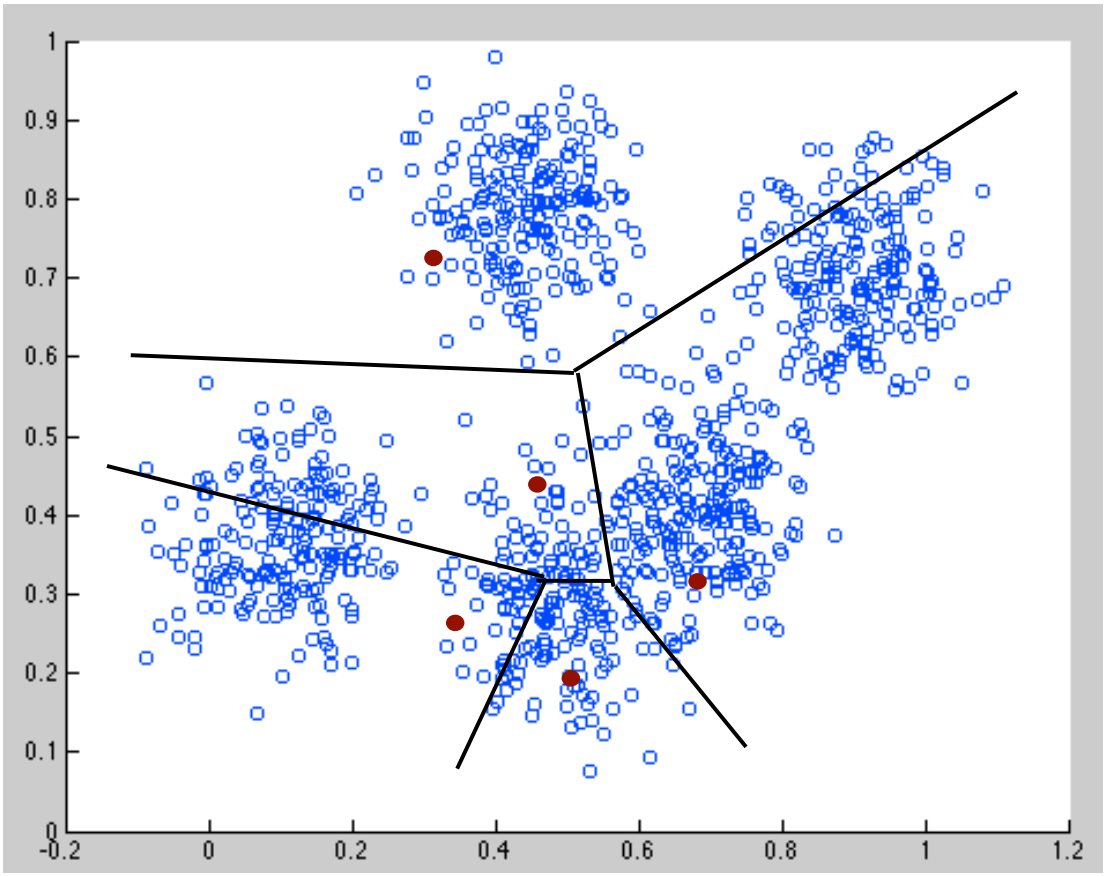
1. Input number of clusters, randomly initialize centers
2. Assign all points to the closest cluster center
3. Change cluster centers to be in the middle of its points
4. Repeat until convergence

K-Means in action



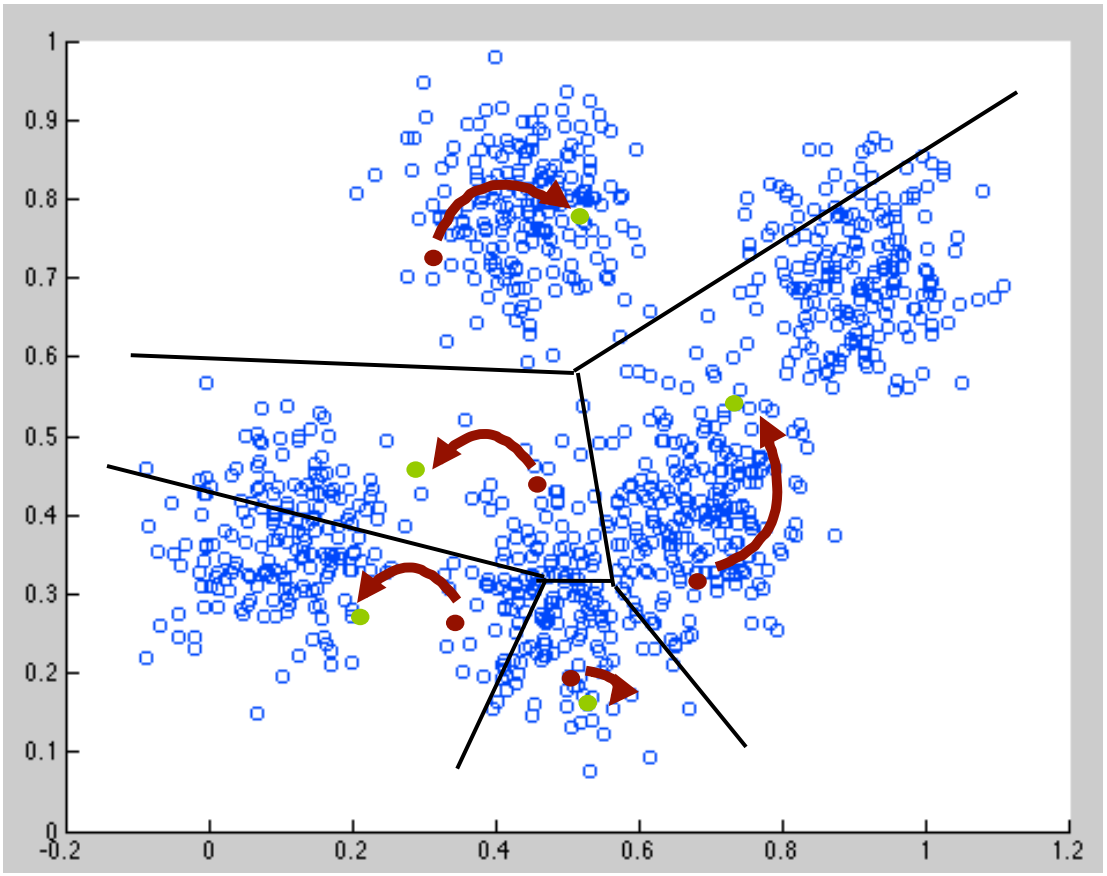
1. Input number of clusters, randomly initialize centers
2. Assign all points to the closest cluster center
3. Change cluster centers to be in the middle of its points
4. Repeat until convergence

K-Means in action



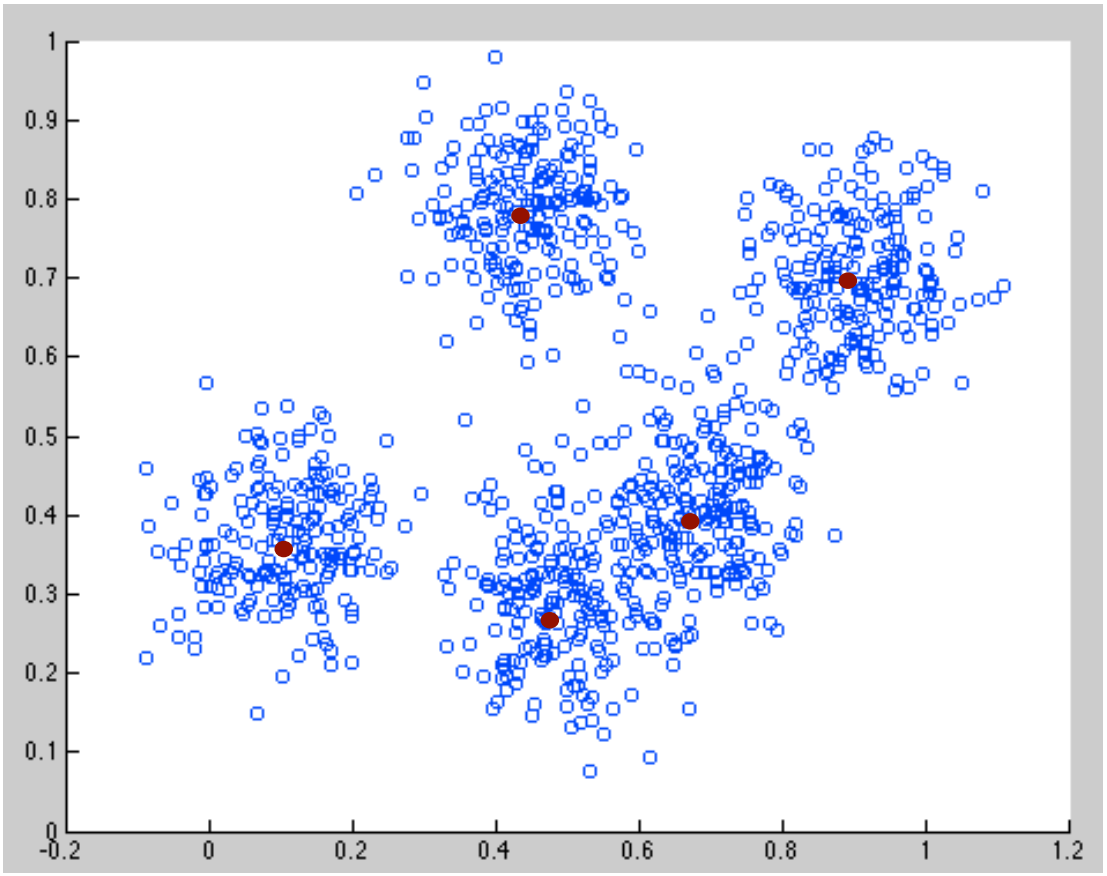
1. Input number of clusters, randomly initialize centers
2. Assign all points to the closest cluster center
3. Change cluster centers to be in the middle of its points
4. Repeat until convergence

K-Means in action



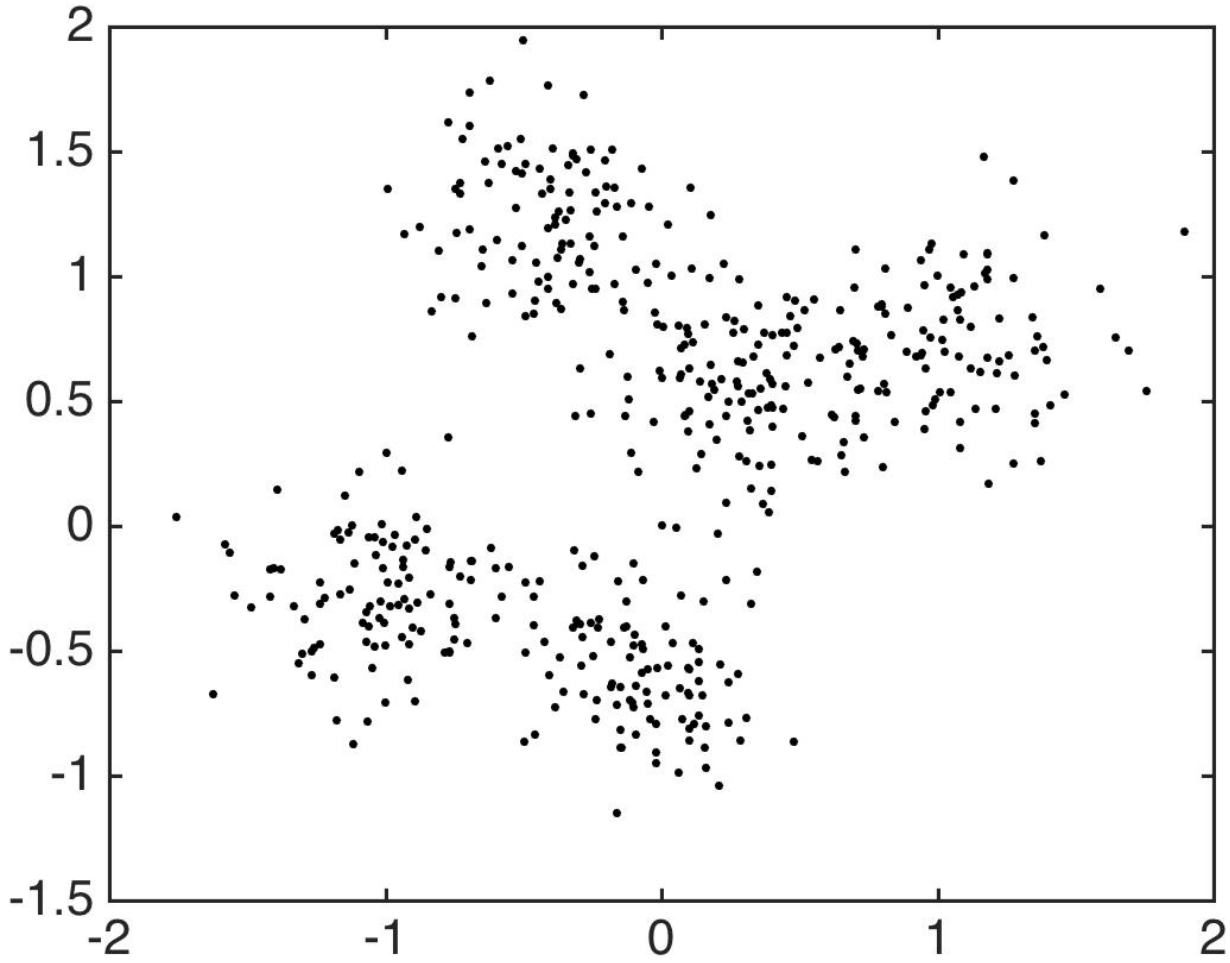
1. Input number of clusters, randomly initialize centers
2. Assign all points to the closest cluster center
3. Change cluster centers to be in the middle of its points
4. Repeat until convergence

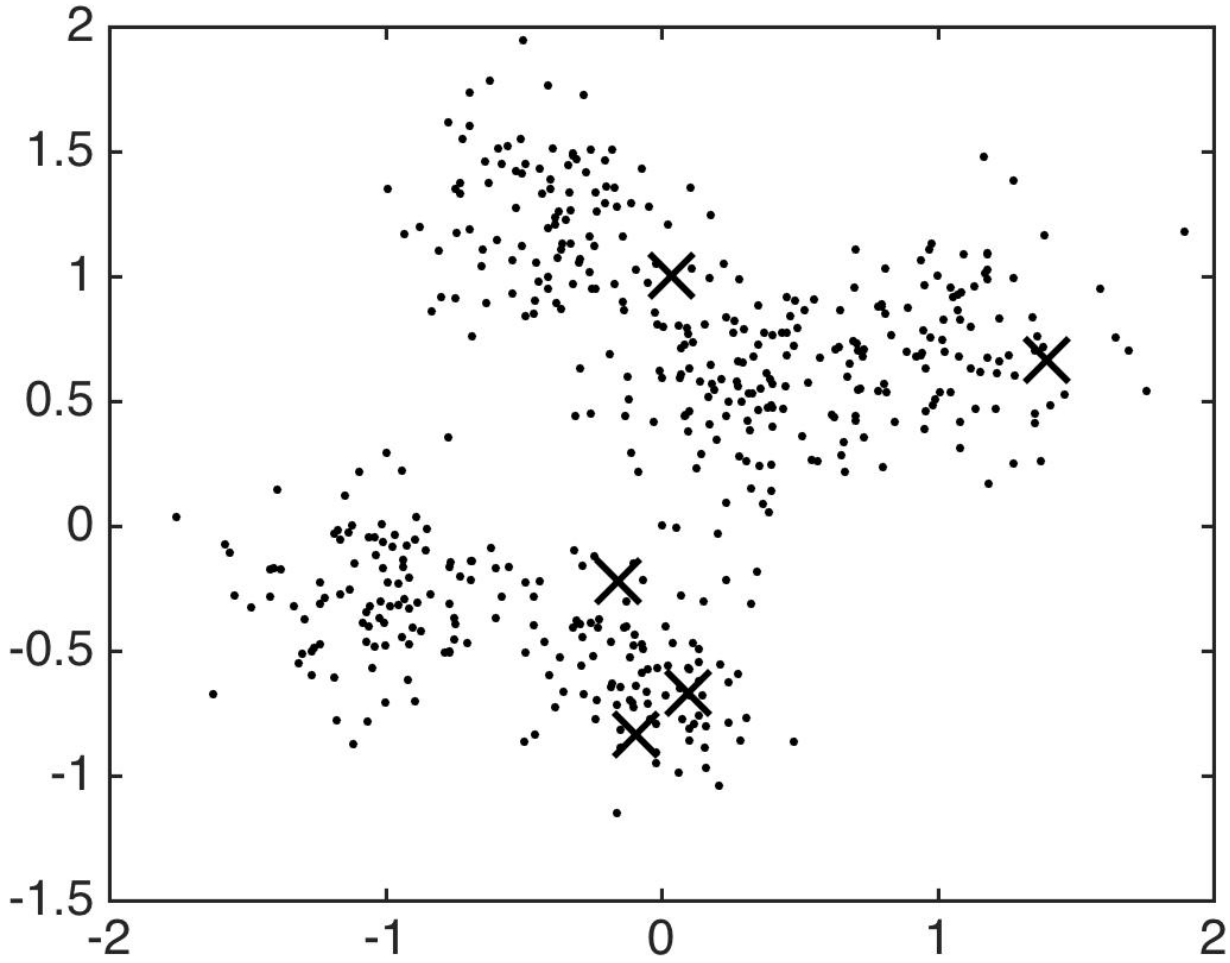
K-Means in action

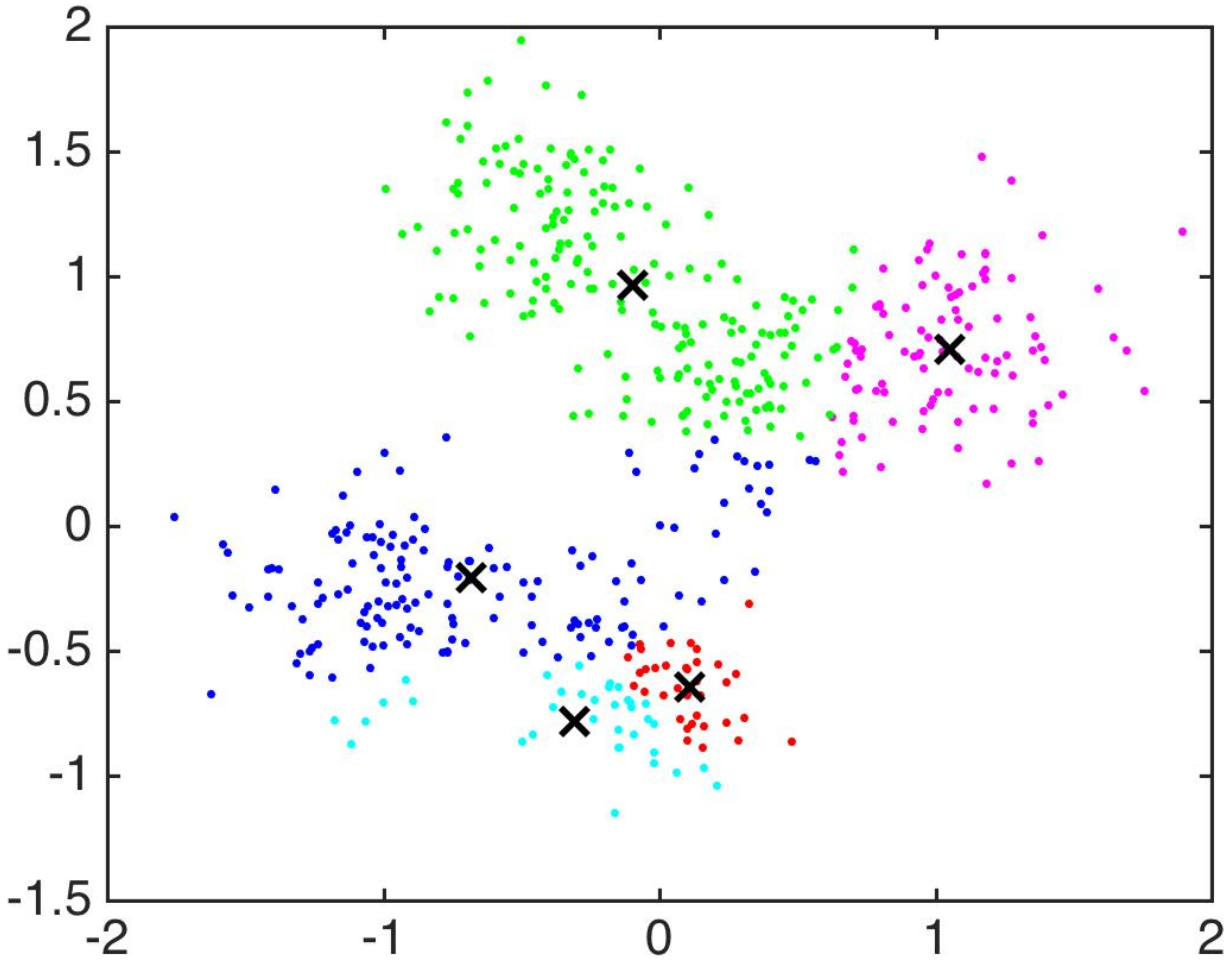


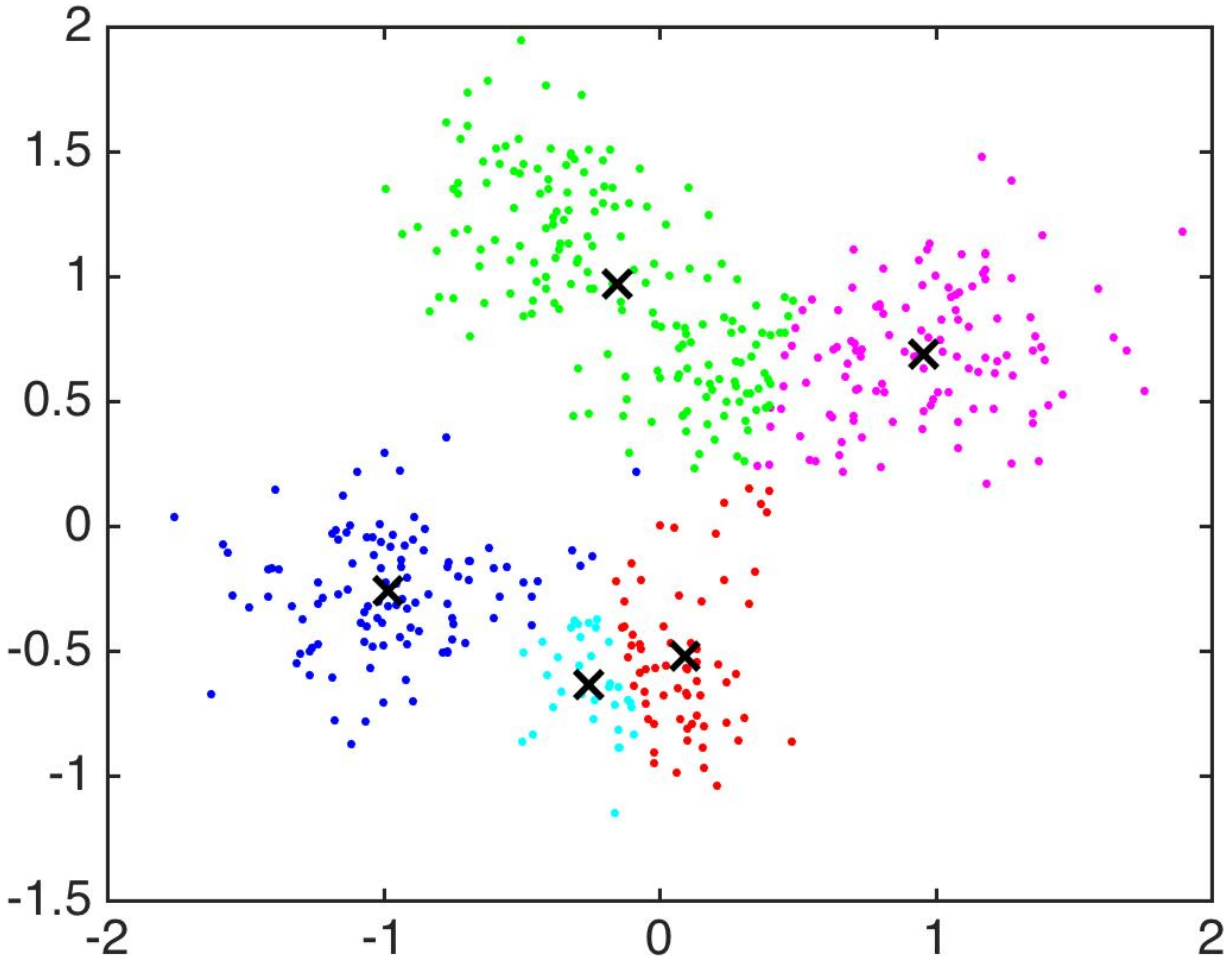
1. Input number of clusters, randomly initialize centers
2. Assign all points to the closest cluster center
3. Change cluster centers to be in the middle of its points
4. Repeat until convergence

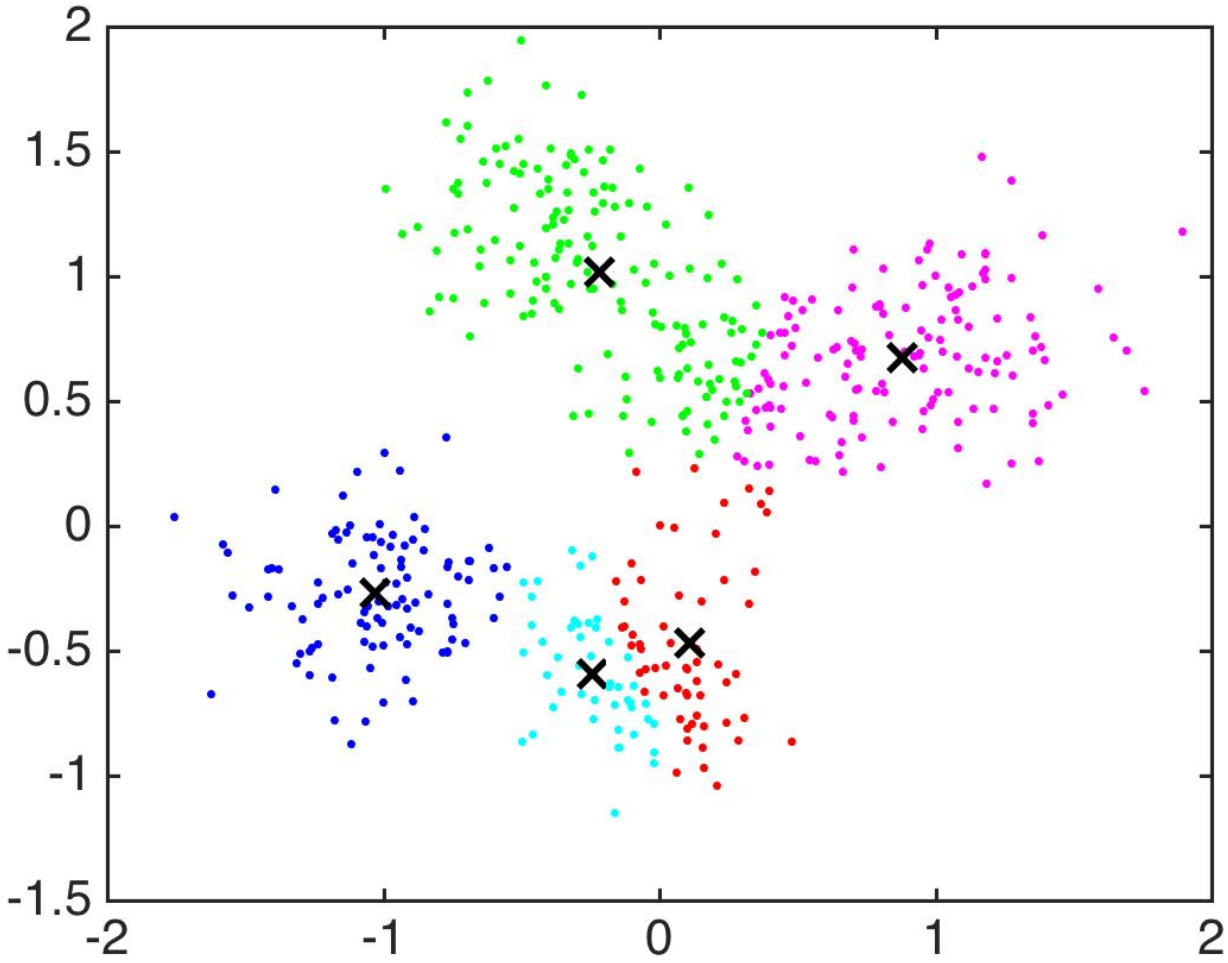
Ready to see it?

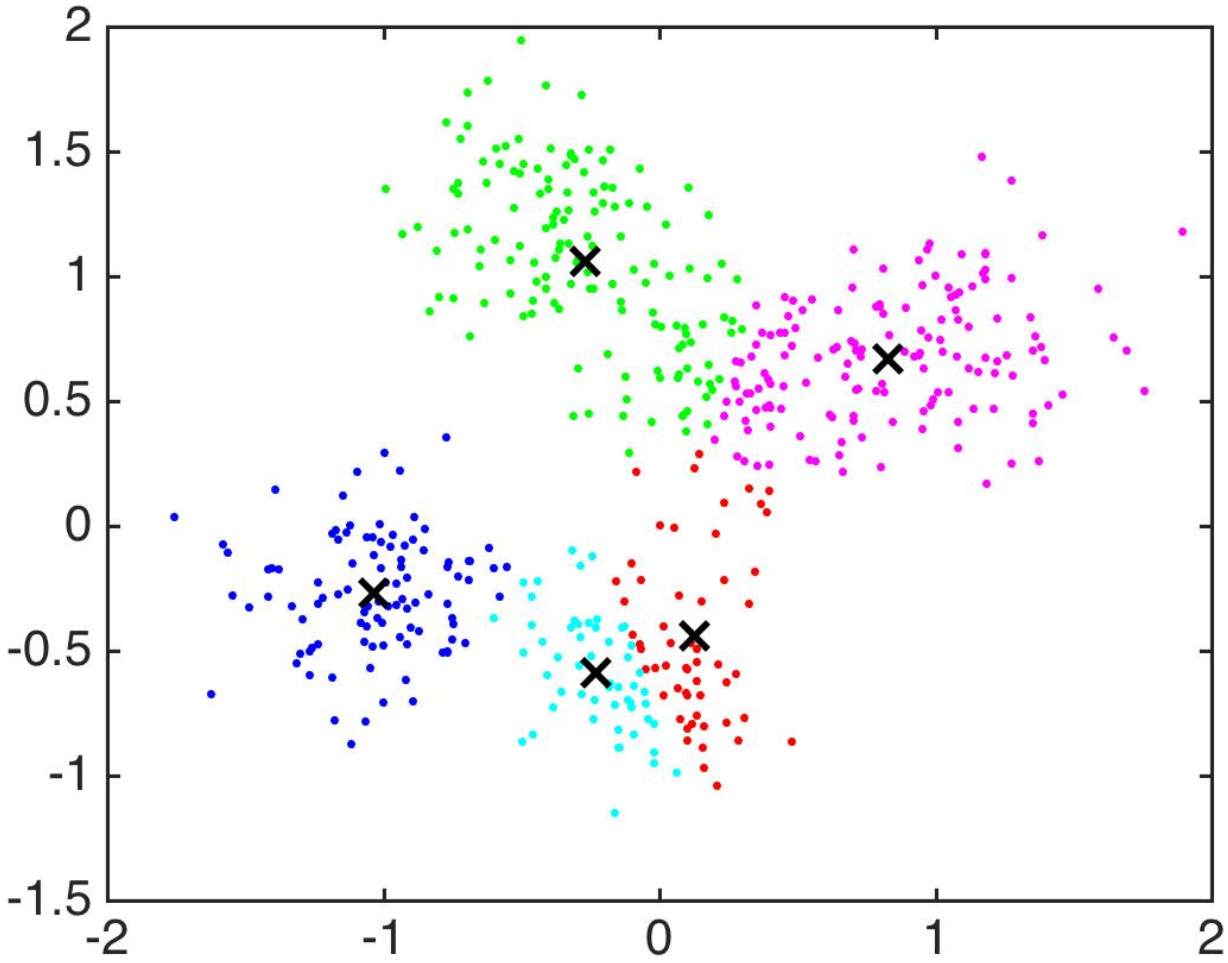

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

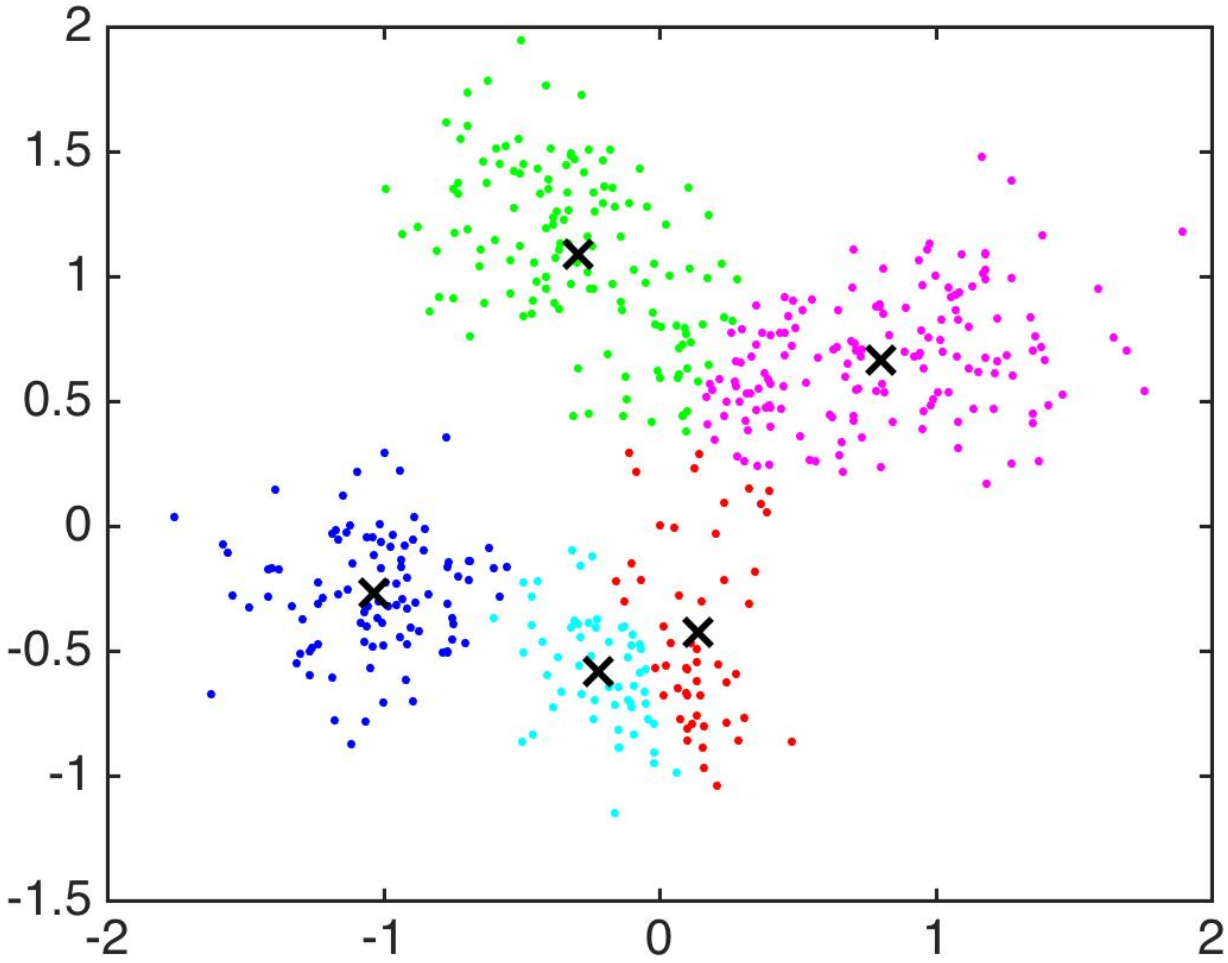

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

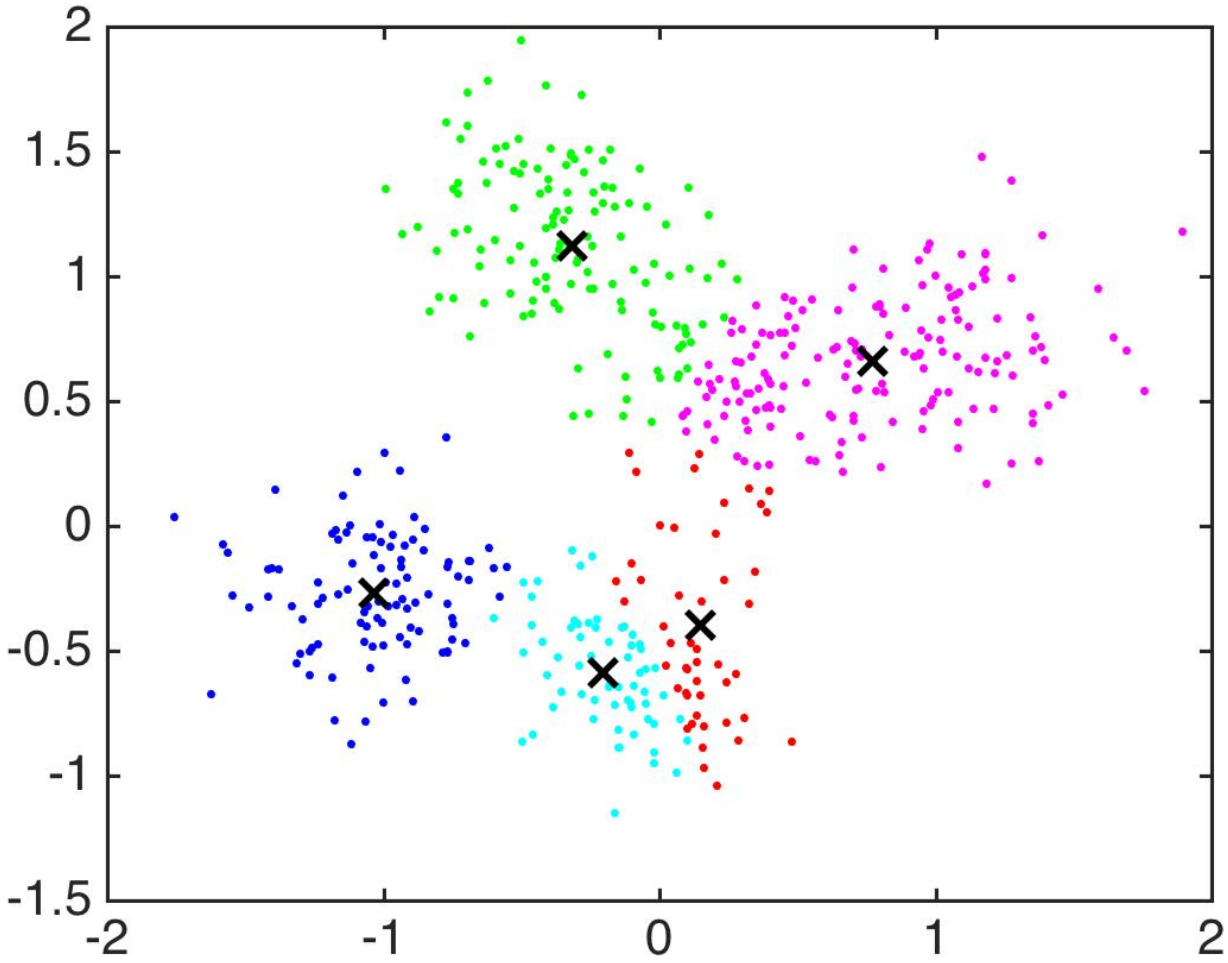

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

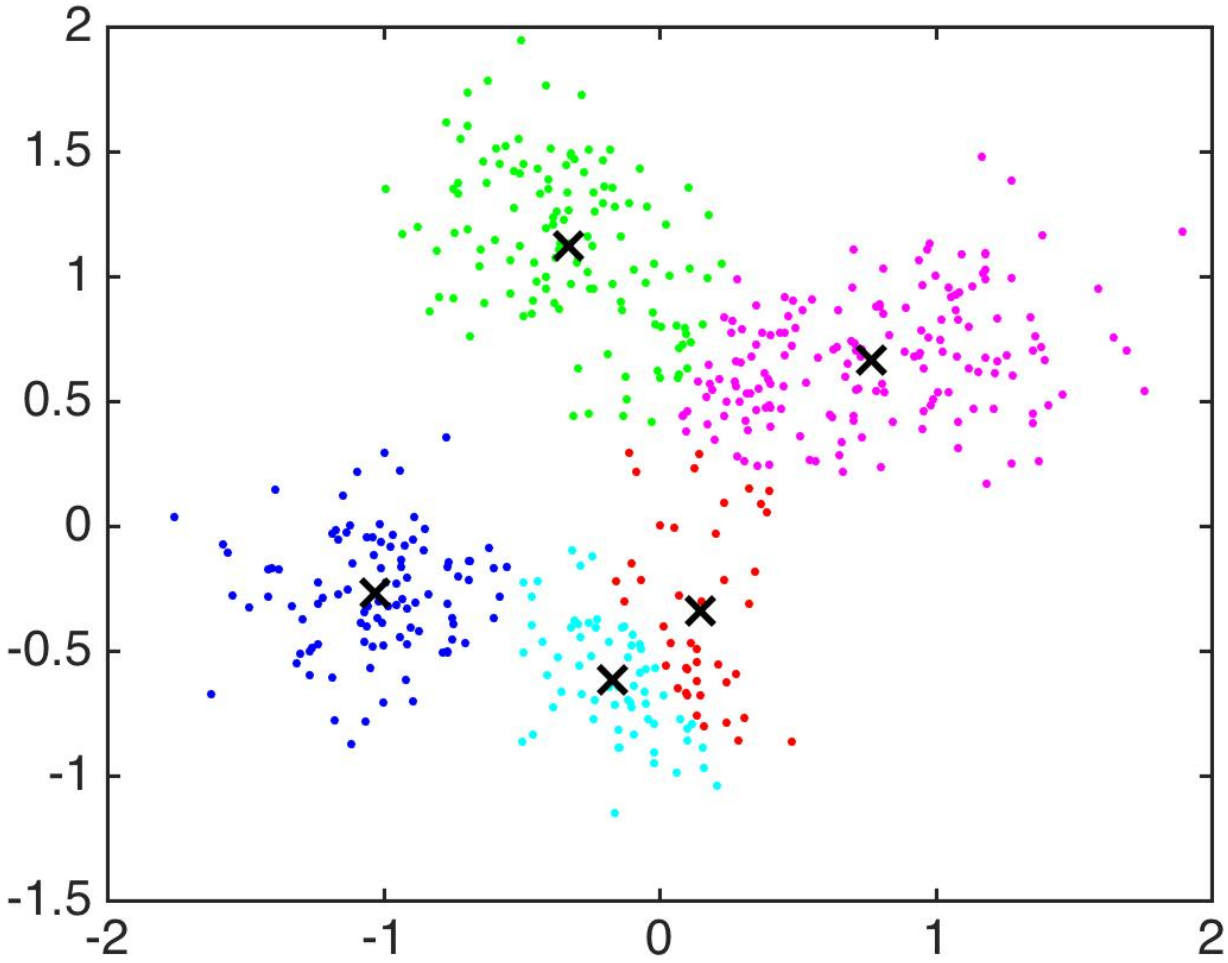

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

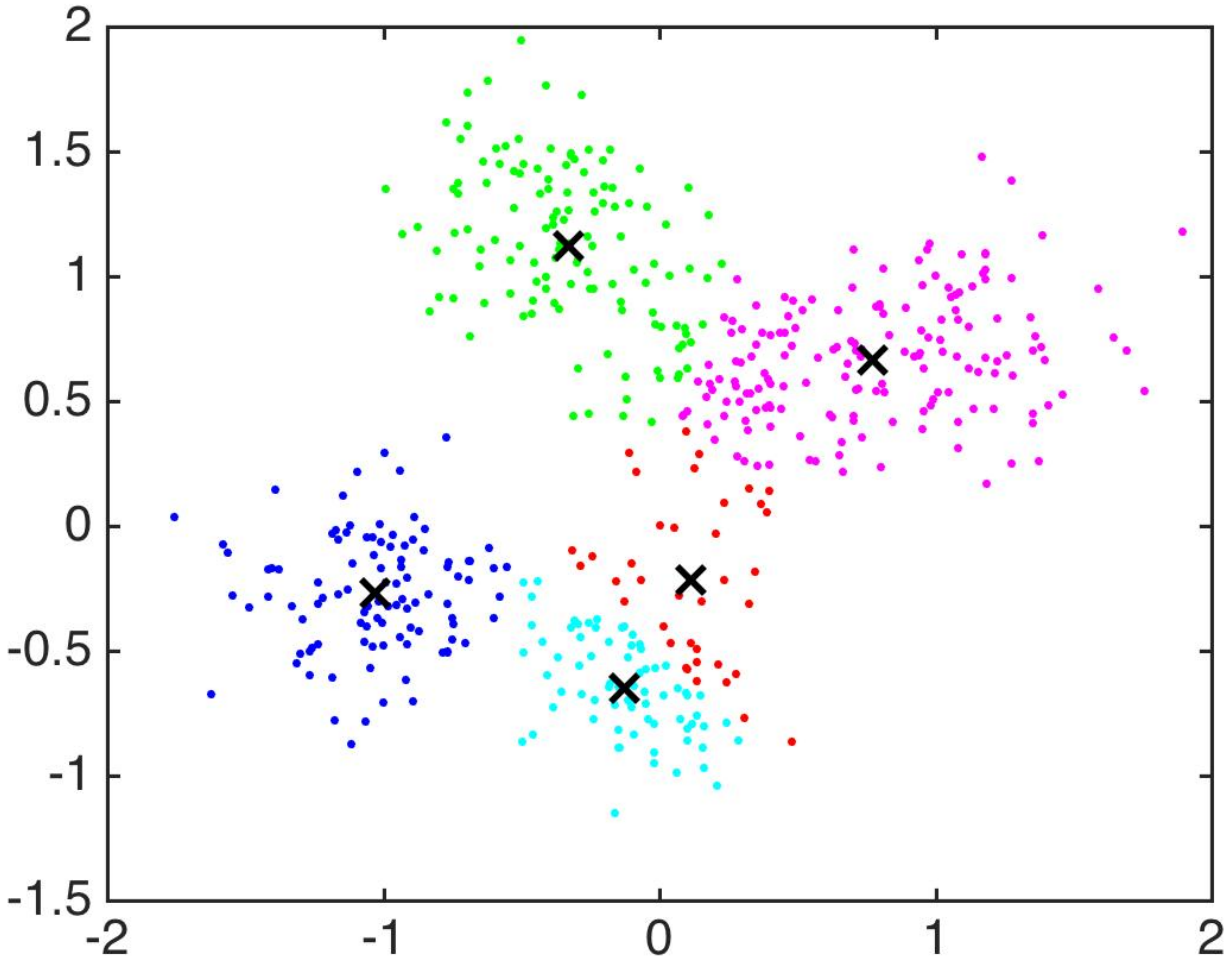

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

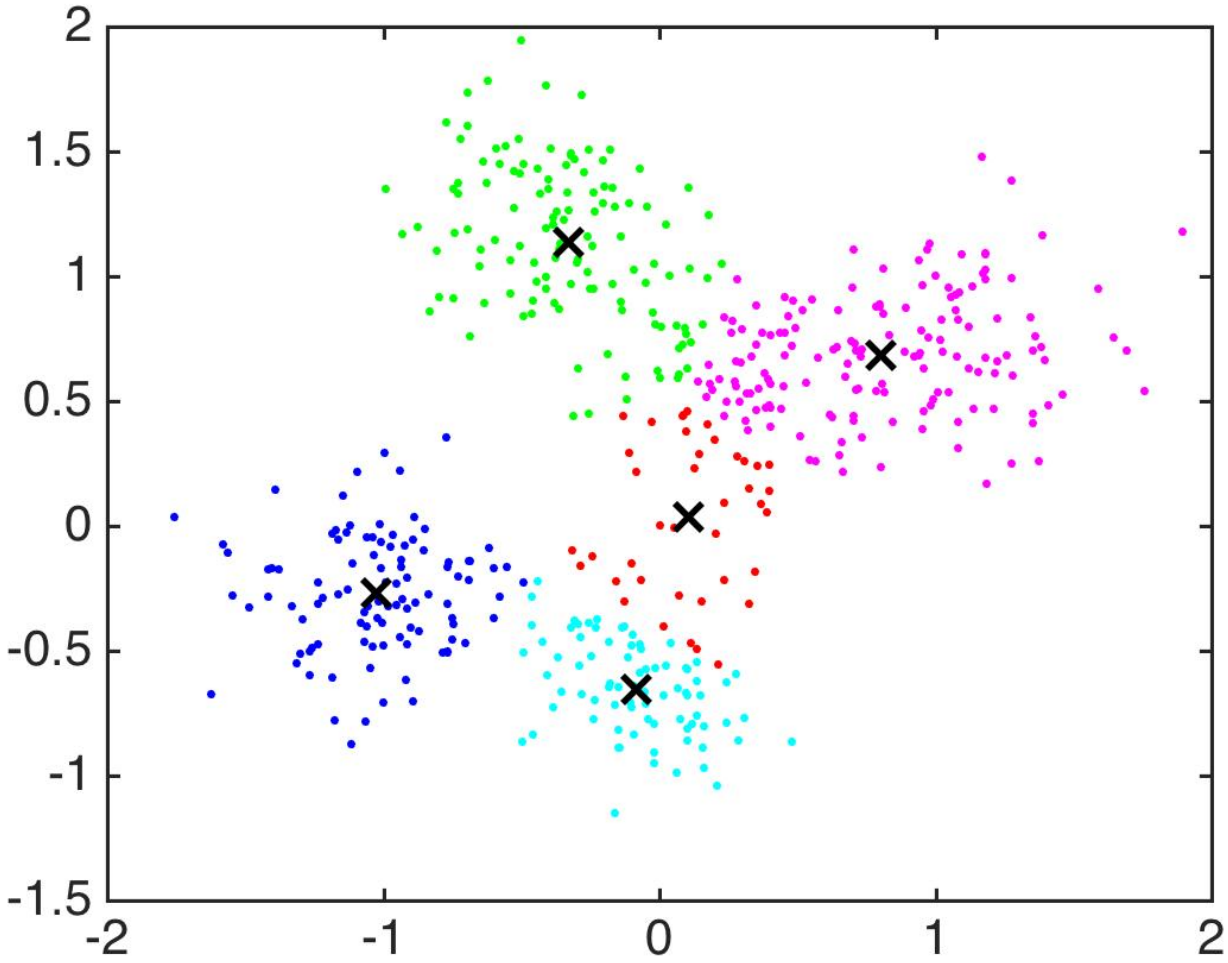

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

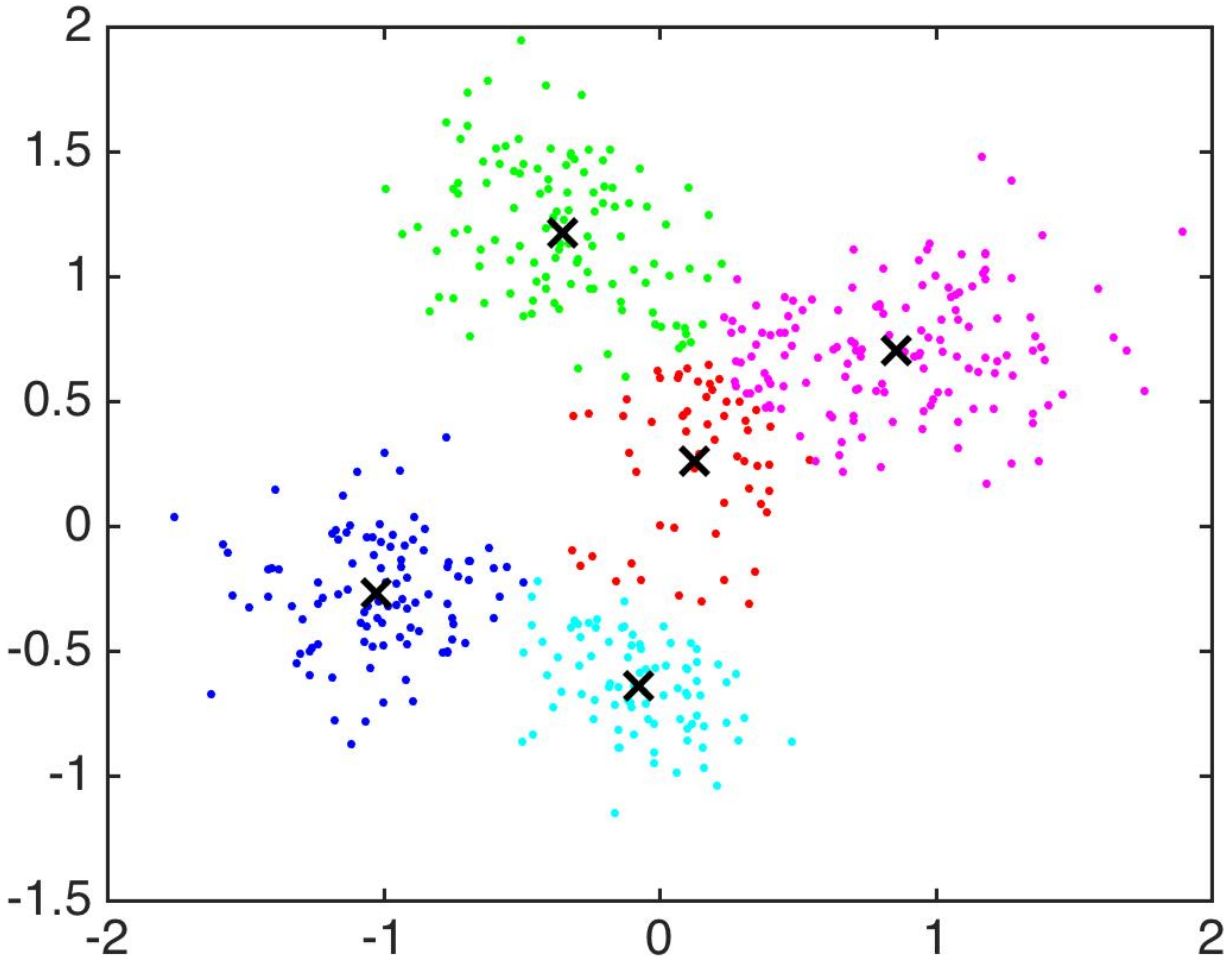

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

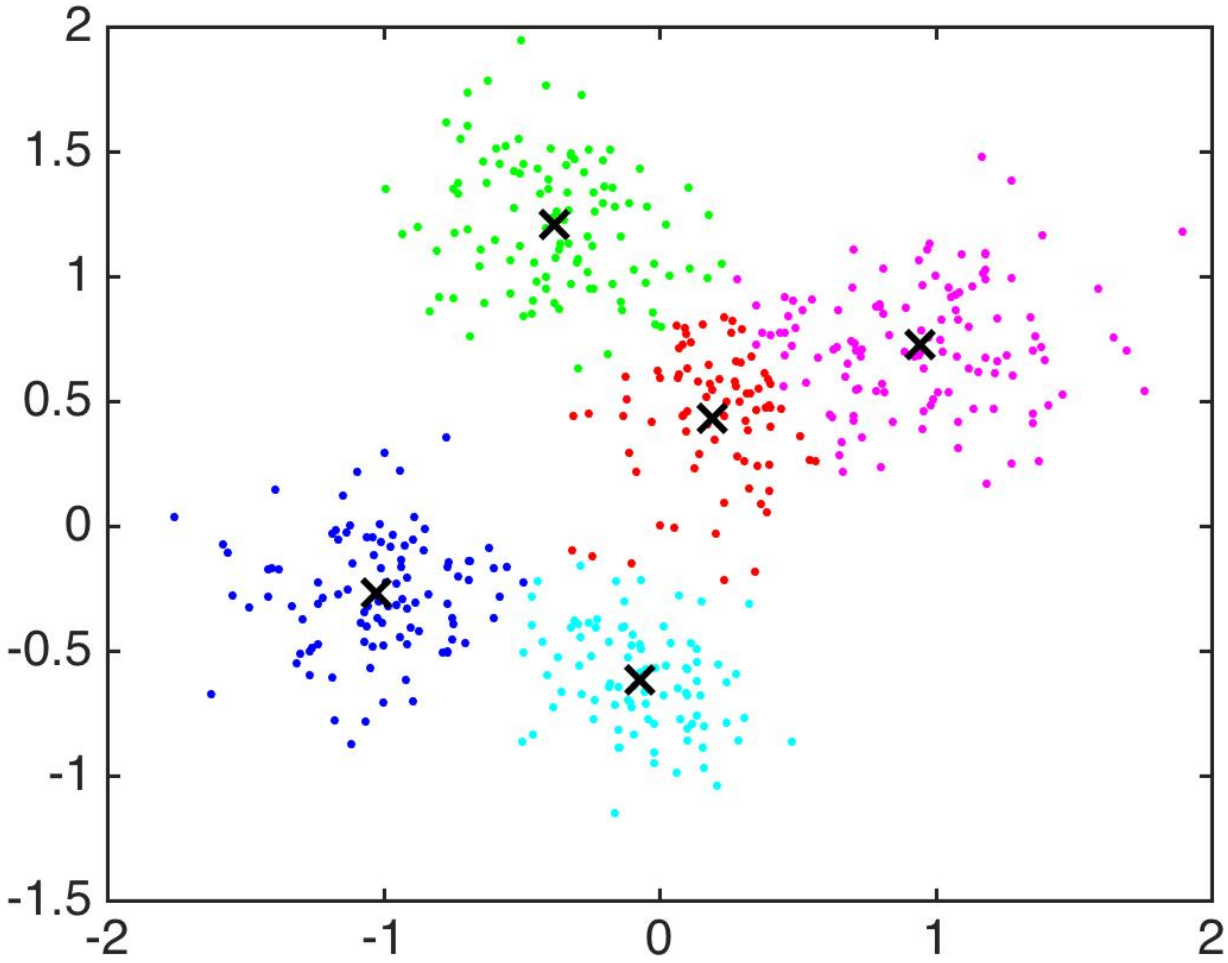

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

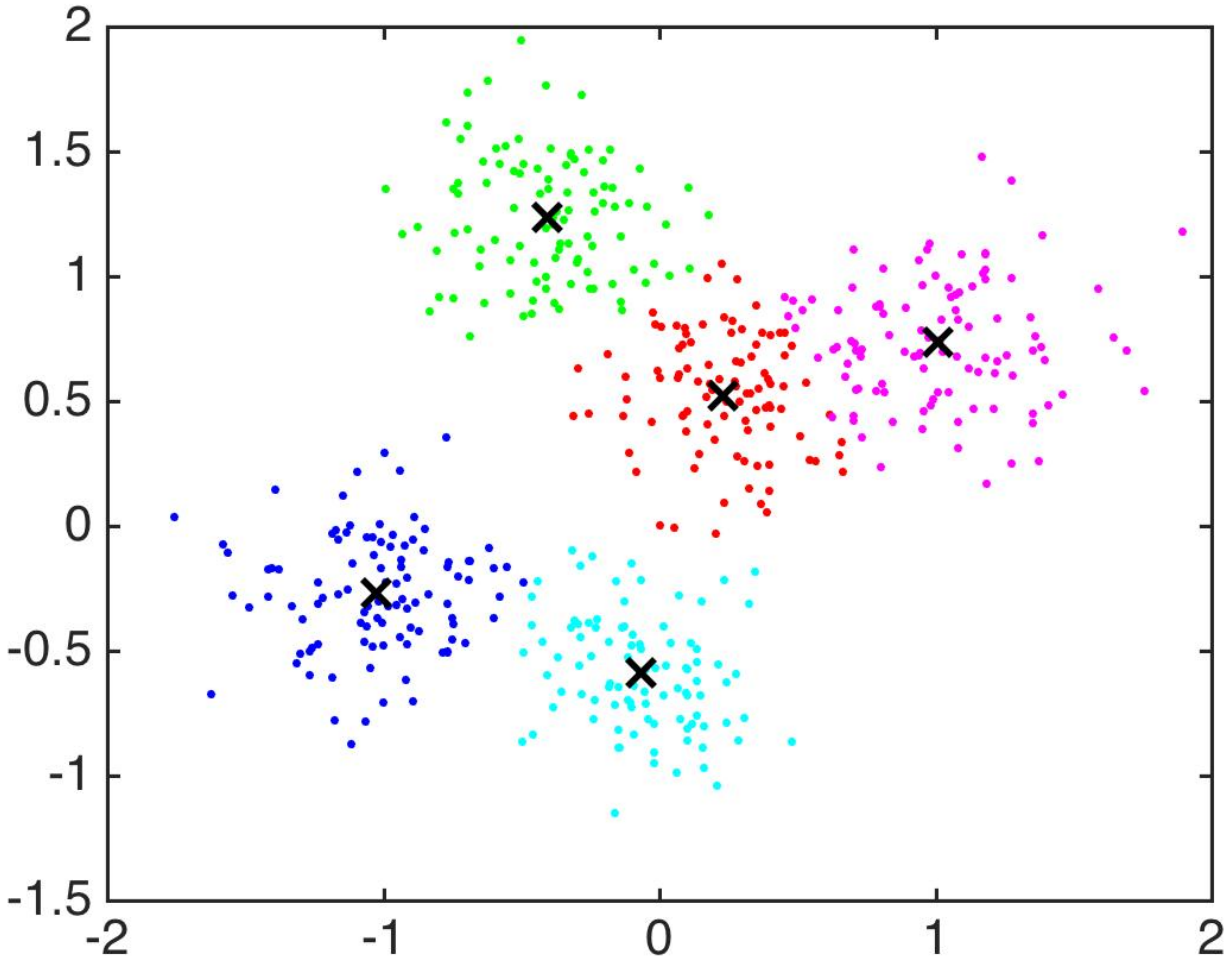

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

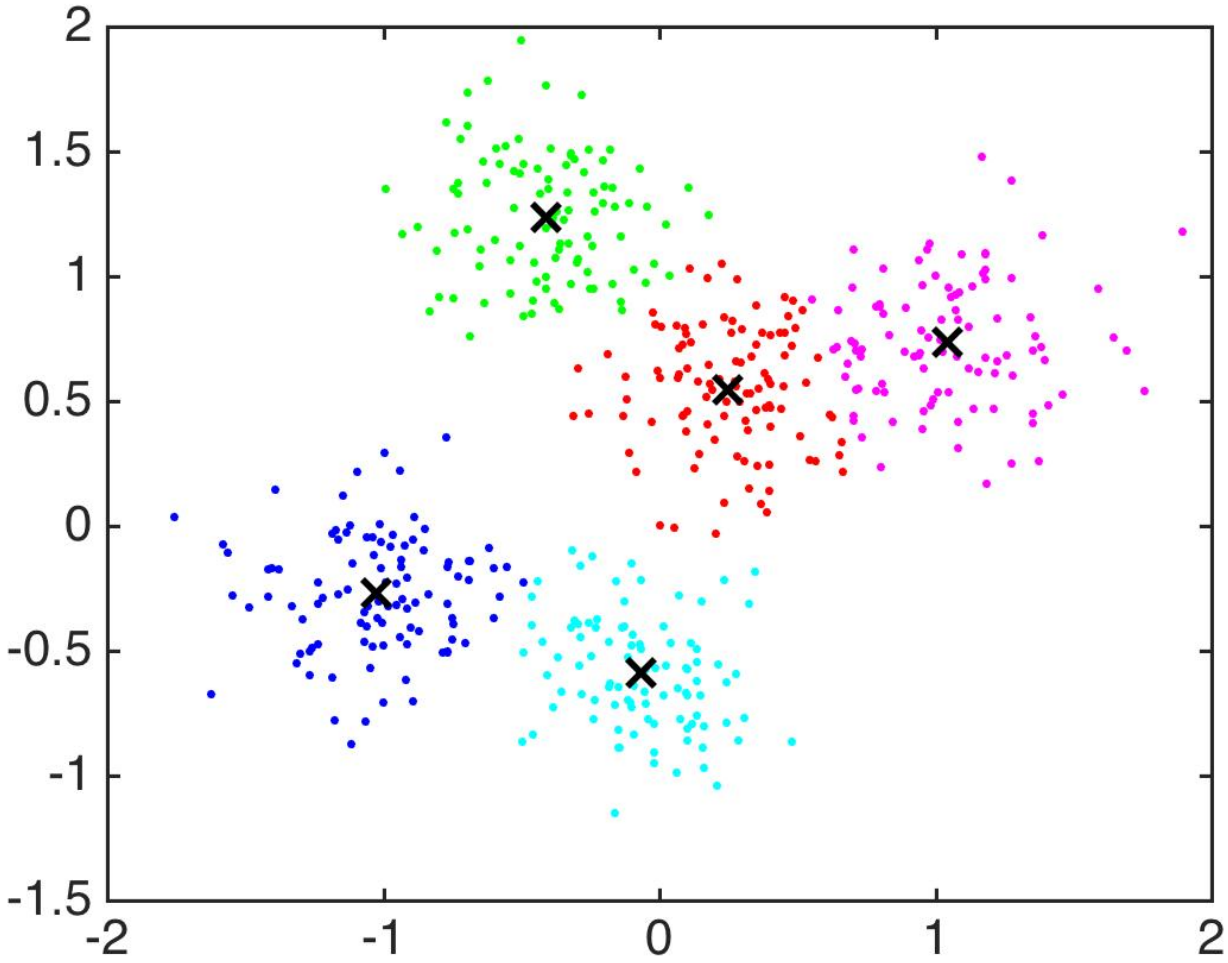

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

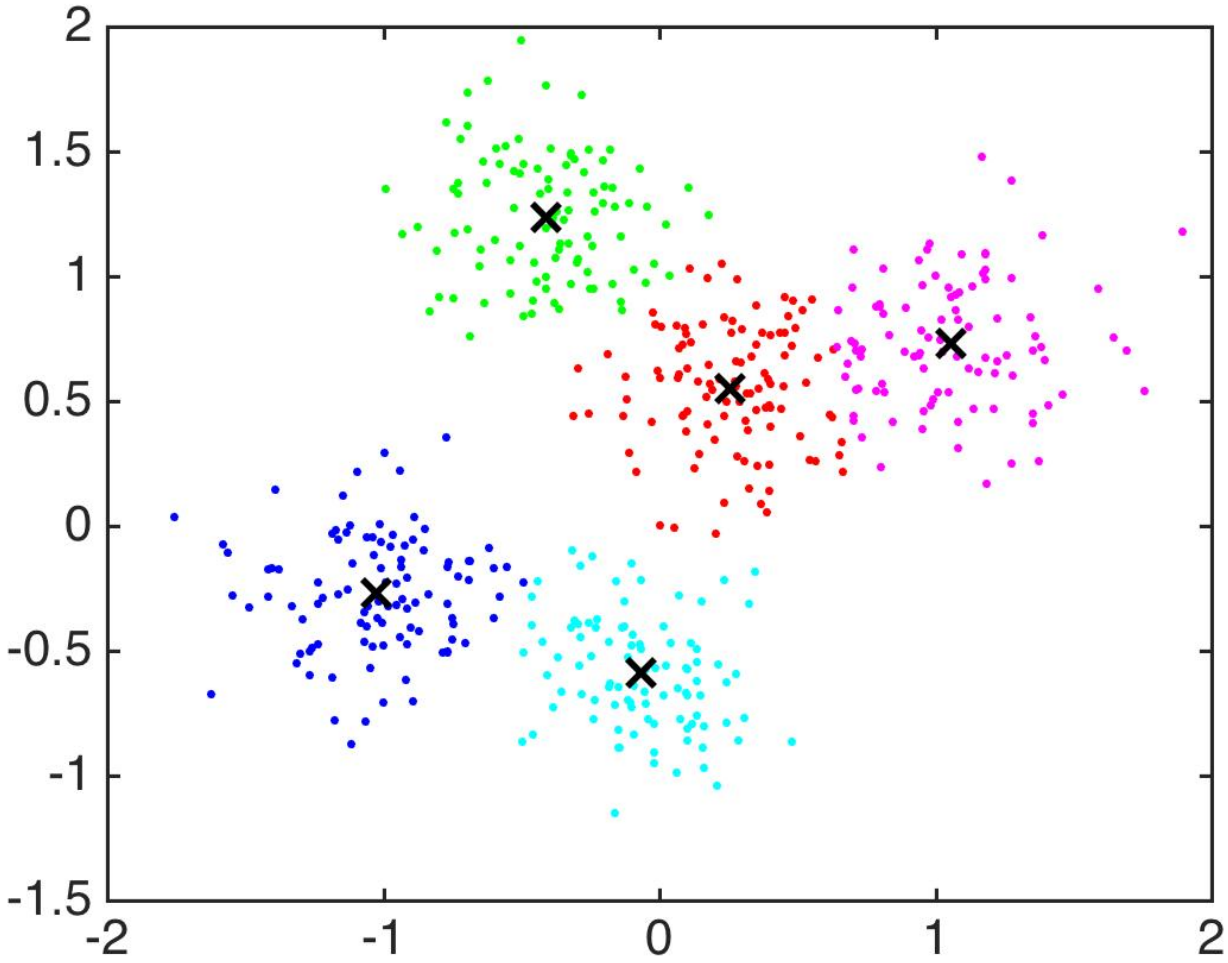

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

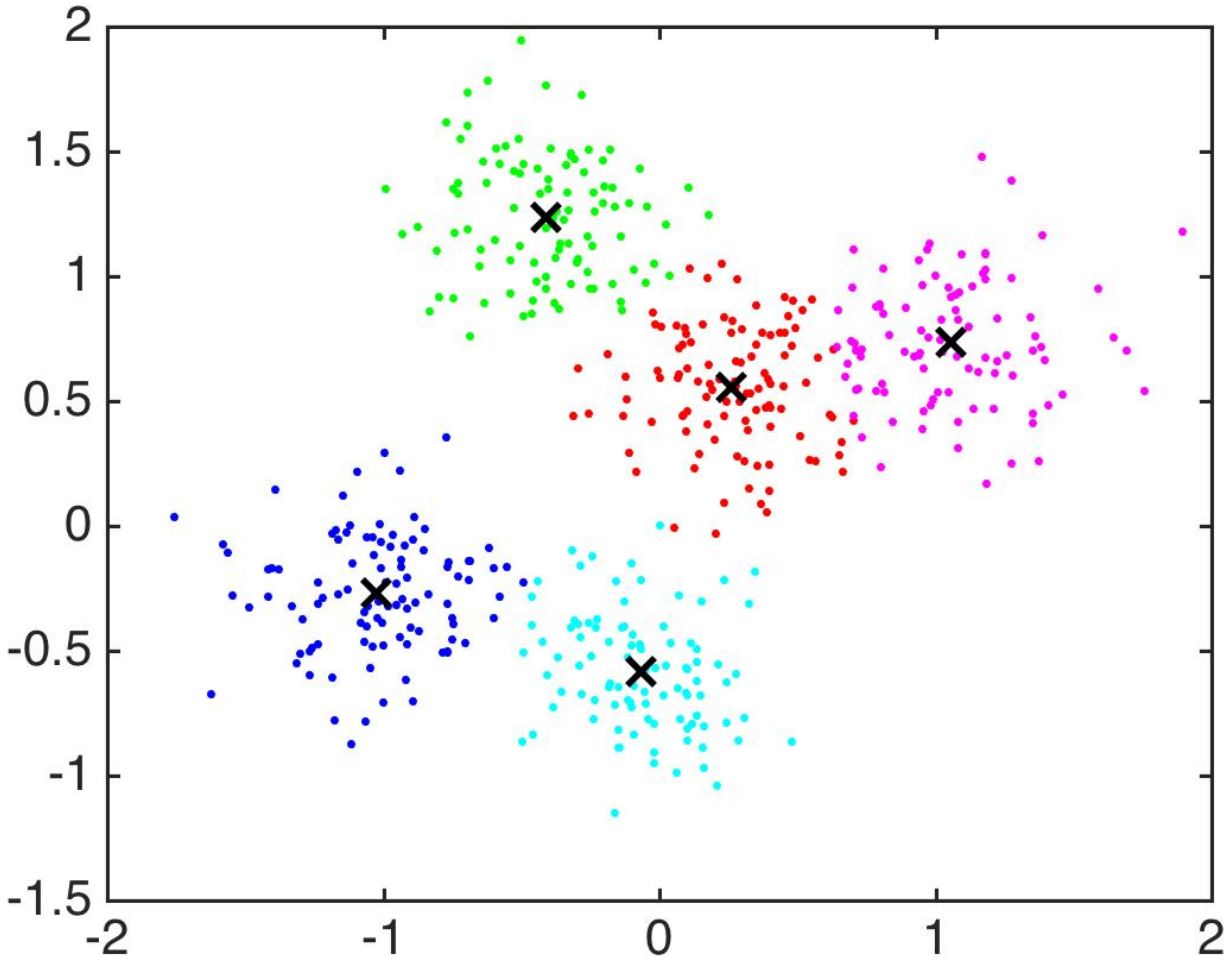

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

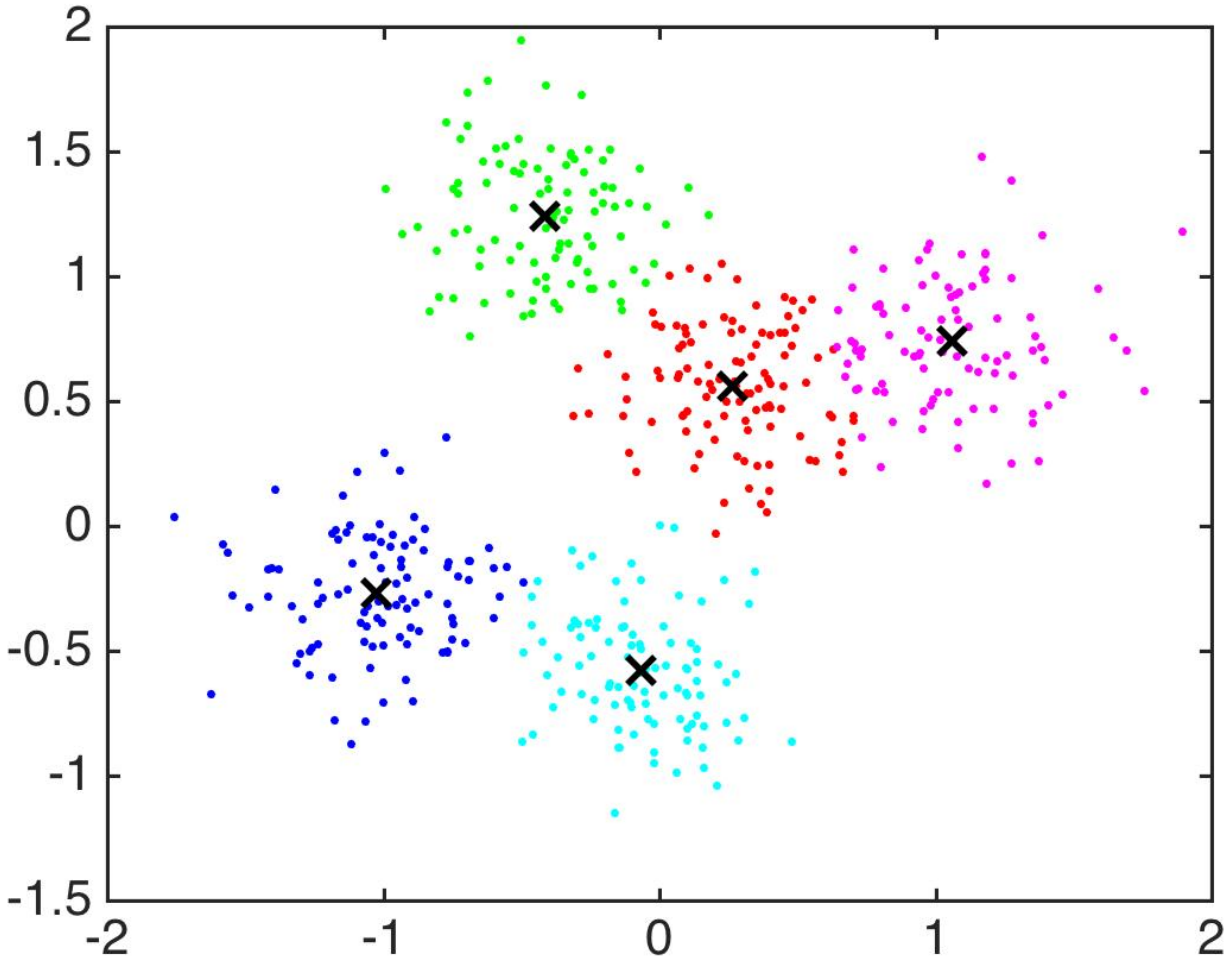

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

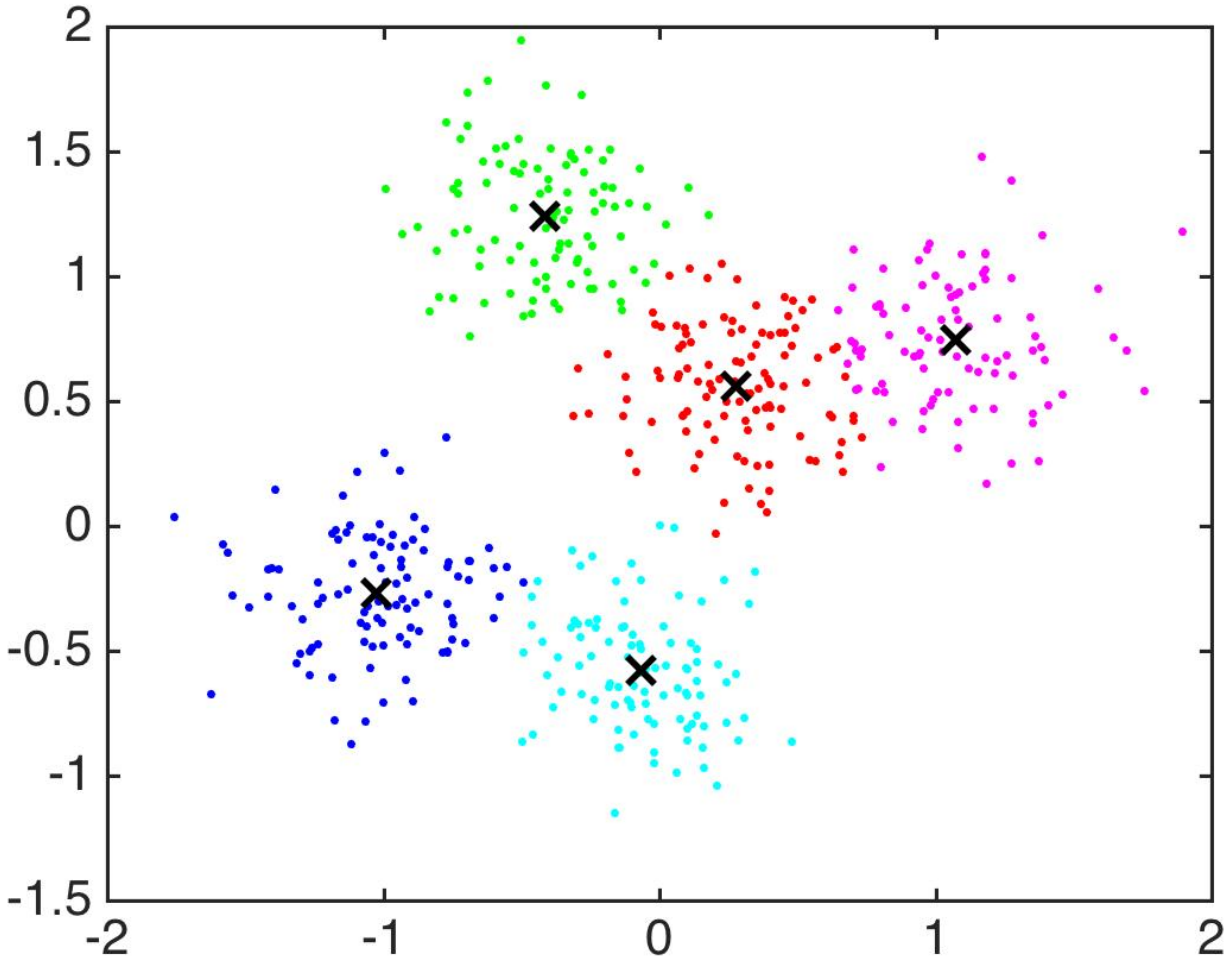

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

K-Means Actual Goal

Input: Data set $x_1,\ldots,x_n$, number of clusters K
Output: Cluster centers $c_1,..,c_K$
Goal: Minimize

$$\left(\text{dist}(x_i, c_k)\right)$$

K-Means Actual Goal

Input: Data set $x_1,\ldots,x_n$, number of clusters K
Output: Cluster centers $c_1,..,c_K$
Goal: Minimize

$$\text{cost}(c_1,...,c_K) = \sum_i \min_k \left(\text{dist}(x_i, c_k)\right)$$

Global minimization: Try all possible assignments of m points to K clusters:

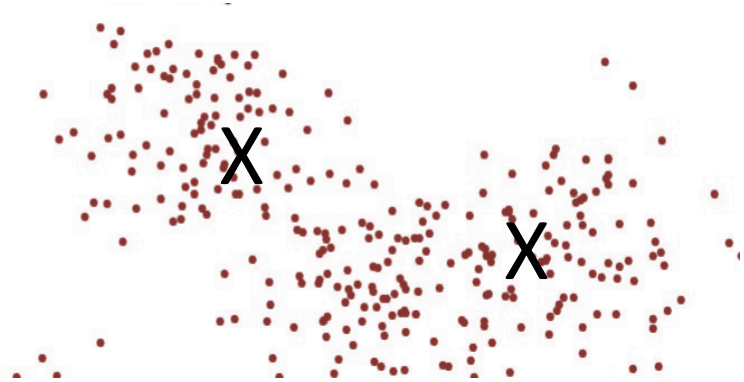
$$\text{Combos}(m,K) = \frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\begin{pmatrix} K \\ k \end{pmatrix}k^n$$

$$\text{Combos}(10,4) = 34K, \quad \text{Combos}(19,4) = 10^{10},...,\text{urg}$$

K-Means Actual Goal

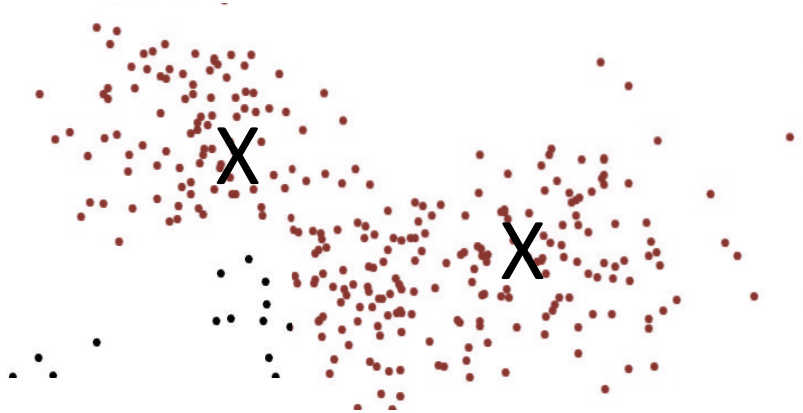
$$\text{cost}(c_1,...,c_K) = \sum_i \min_k \left(\text{dist}(x_i, c_k)\right)$$

K-Means Actual Goal



$$\text{cost}(c_1,\ldots,c_K) = \sum_i \min_k \left(\text{dist}(x_i, c_k)\right)$$

K-Means Actual Goal



$$\text{cost}(c_1,\ldots,c_K) = \sum_{k} \sum_{i:\, x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

K-Means Actual Goal

$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, \text{c}_1, ..., c_K) = \sum_k \sum_{i: x_i \text{ is in cluster}_k} \text{dist}(\text{x}_i, c_k)$$

$$\text{cost}(\text{c}_1, ..., c_K) = \sum_k \sum_{i: x_i \text{ is in cluster}_k} \text{dist}(\text{x}_i, c_k)$$

K-Means Actual Goal

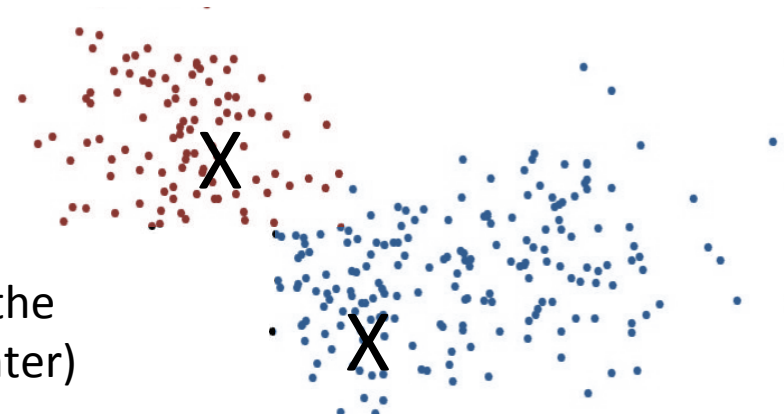
$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, \text{c}_1, ..., c_K) = \sum_k \sum_{i: x_i \text{ is in cluster}_k} \text{dist}(\text{x}_i, c_k)$$

K-Means Actual Goal

$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K) = \sum_k \sum_{i: x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

$$\min_{\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k} \text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K)$$

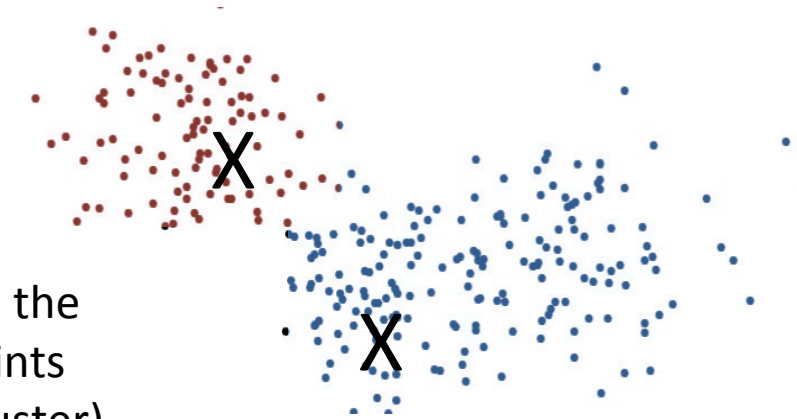
(Assign points to the nearest cluster center)



K-Means Actual Goal

$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K) = \sum_k \sum_{i:\, x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

$$\min_{c_1, c_2, .., c_k} \text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K)$$



(Assign centers to the middle of the points assigned to that cluster)

K-Means Actual Goal

$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K) = \sum_k \sum_{i:\, x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

Input: number of clusters K, randomly initialize centers c_k

Until converged:

Assign all points to the closest cluster center

$$\min_{\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k} \text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K)$$

Change cluster centers to be in the middle of its points

$$\min_{c_1, c_2, .., c_k} \text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K)$$

K-Means Actual Goal

$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K) = \sum_k \sum_{i:\, x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

Does K-Means achieve its goal?

K-Means Actual Goal

$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K) = \sum_{k} \sum_{i:\, x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

Does K-Means achieve its goal?

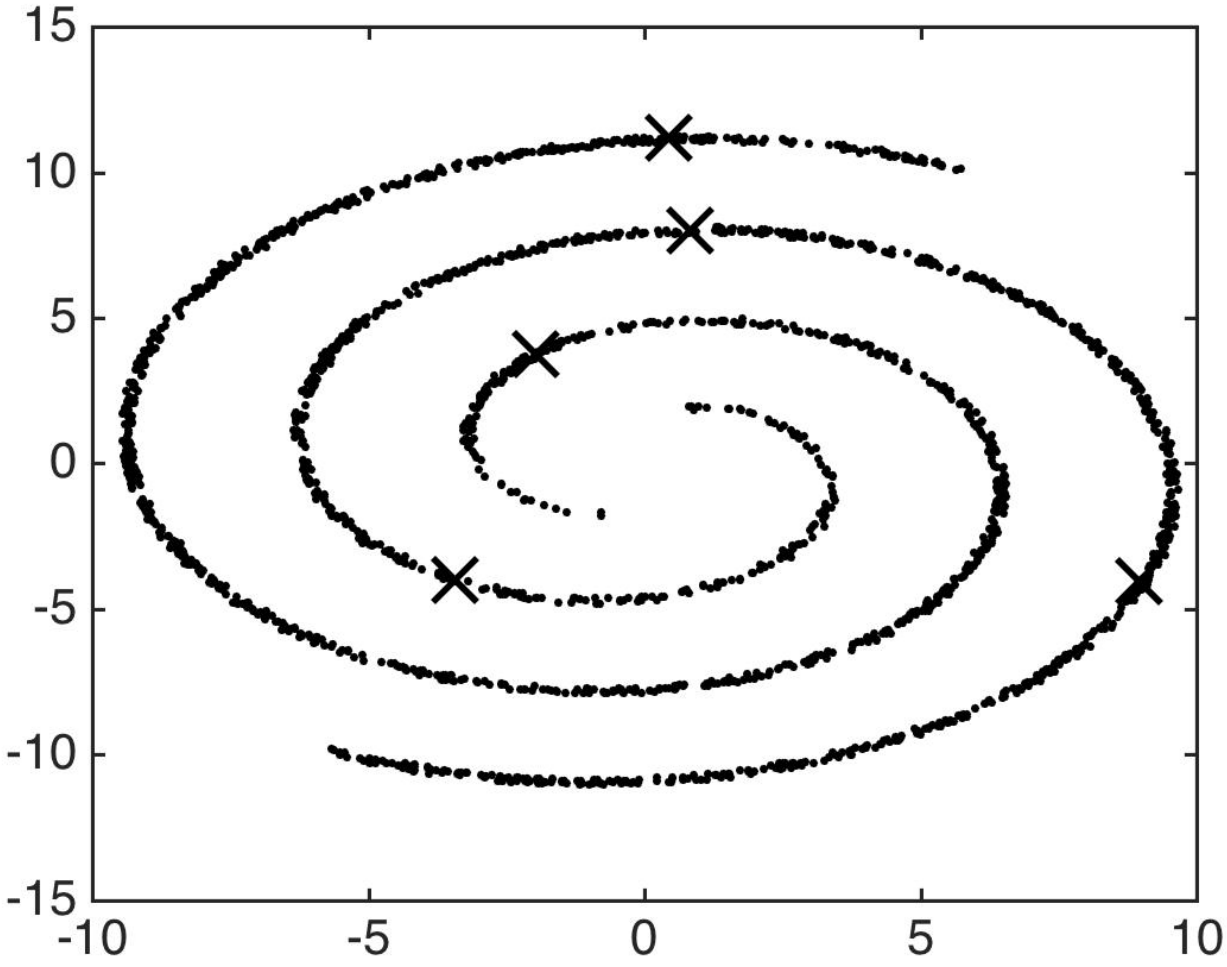
Not always. Might need multiple replicates.

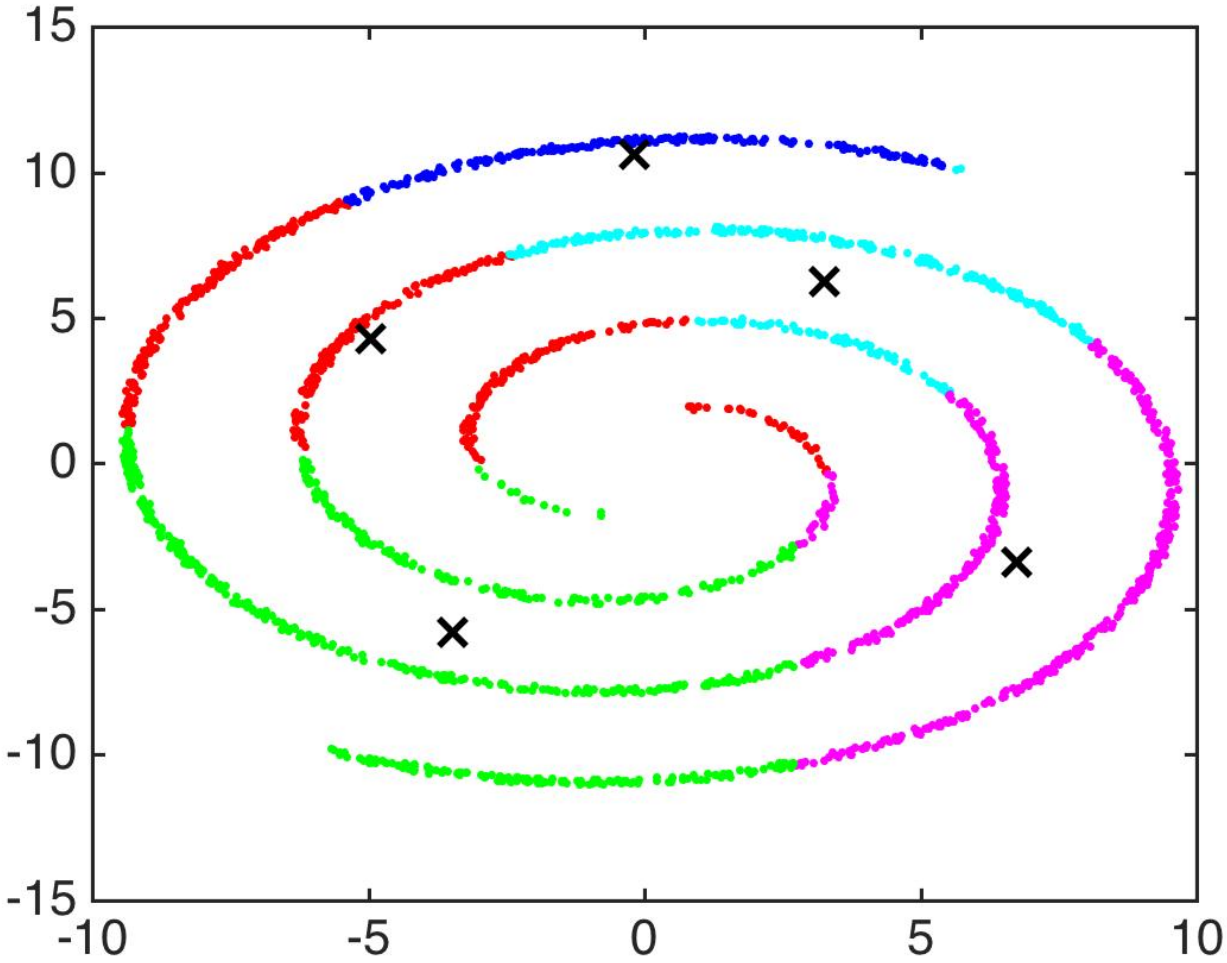
K-Means Actual Goal

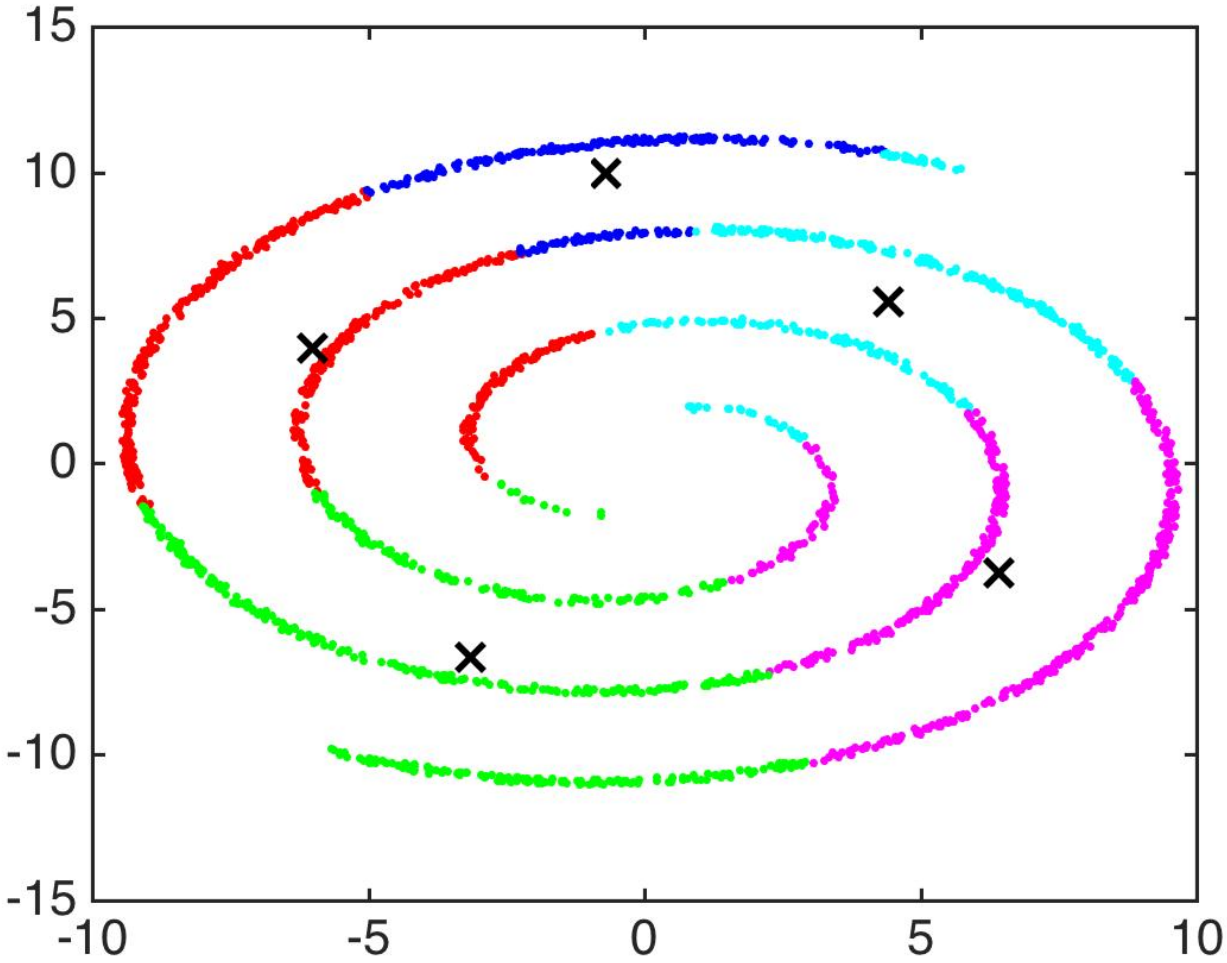
$$\text{cost}(\text{cluster}_1, \text{cluster}_2, .., \text{cluster}_k, c_1, ..., c_K) = \sum_{k} \sum_{i:\, x_i \text{ is in cluster}_k} \text{dist}(x_i, c_k)$$

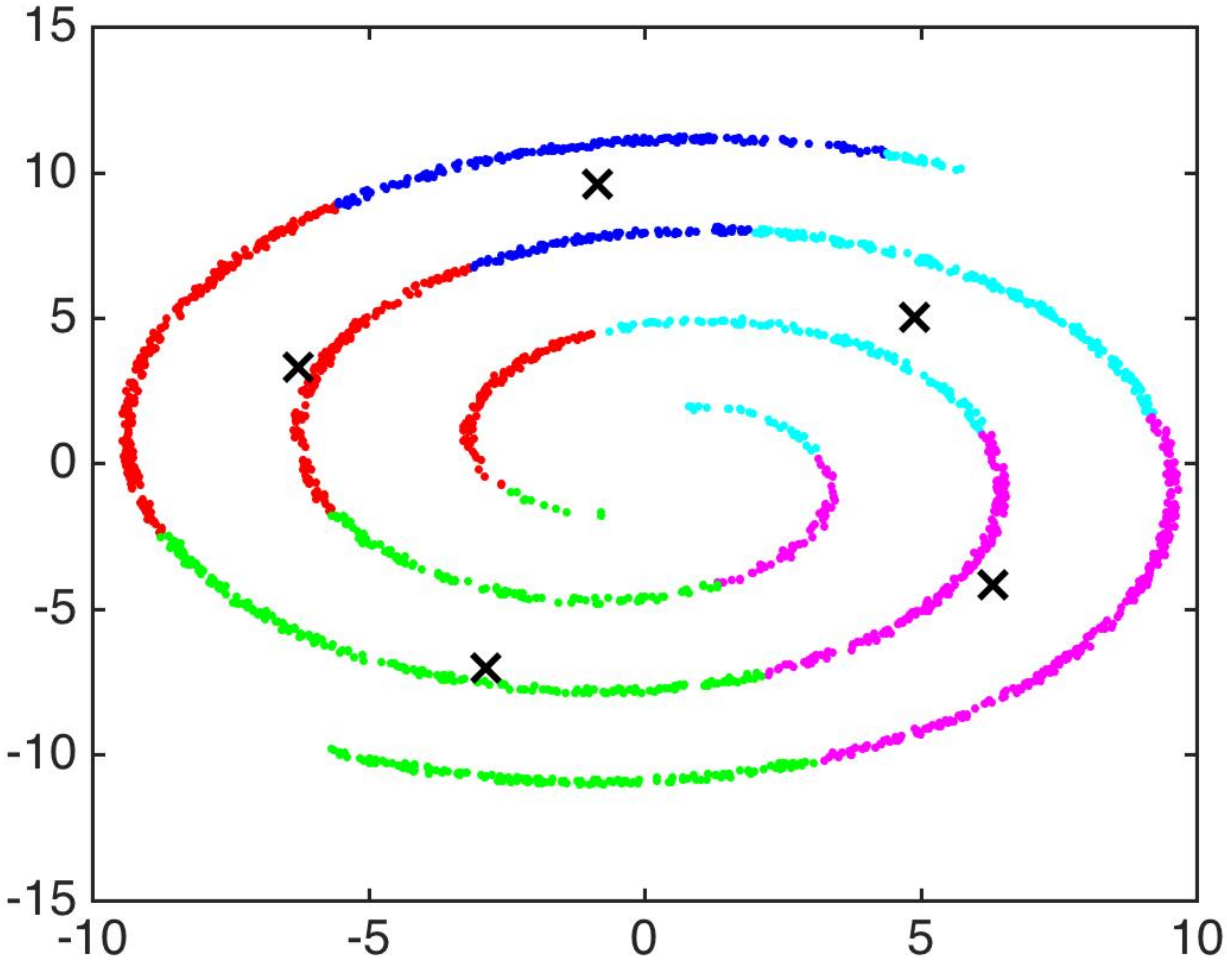
Does K-Means achieve its goal?

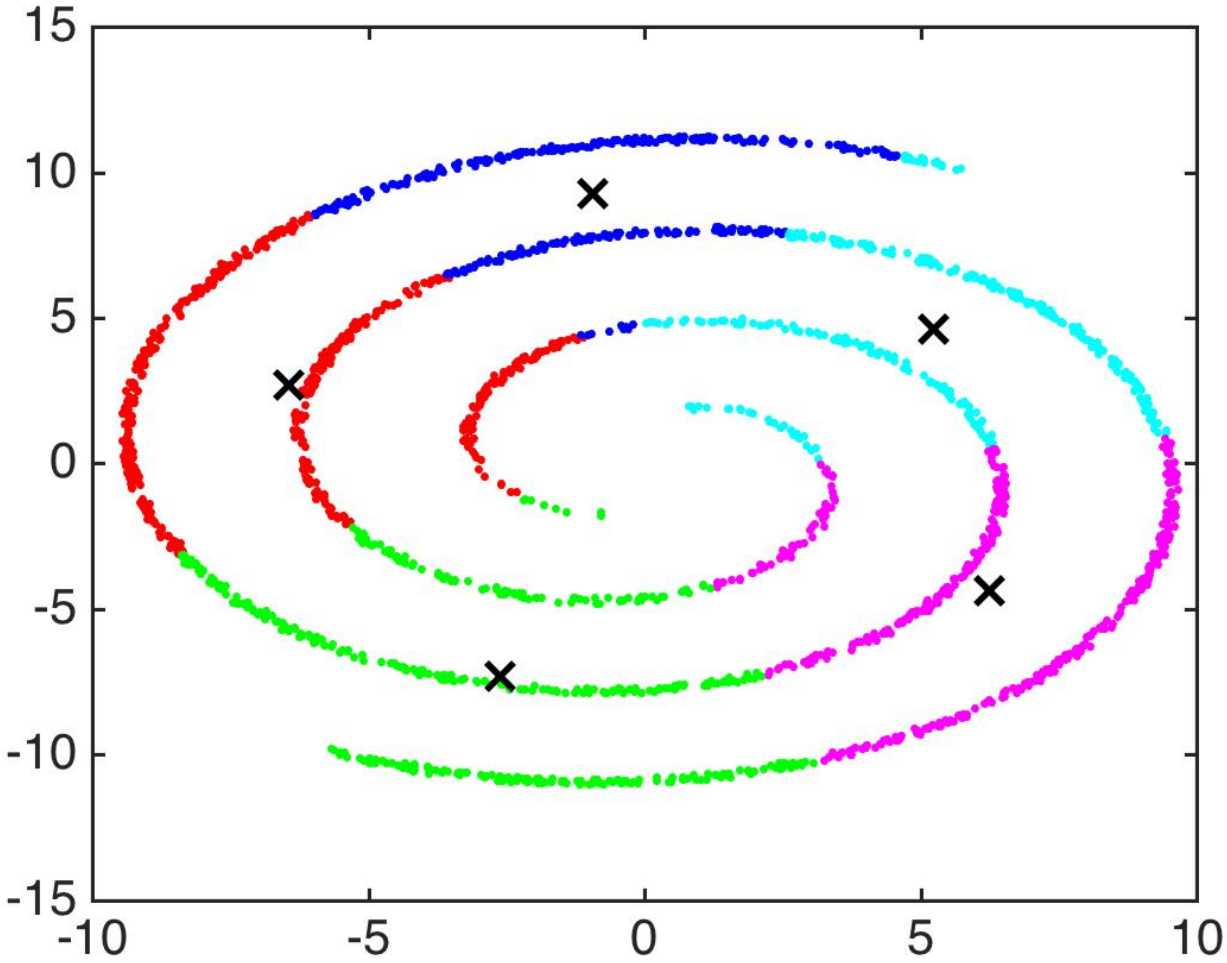
Not always. Might need multiple replicates. Even worse than that, its goal might not be the right one…

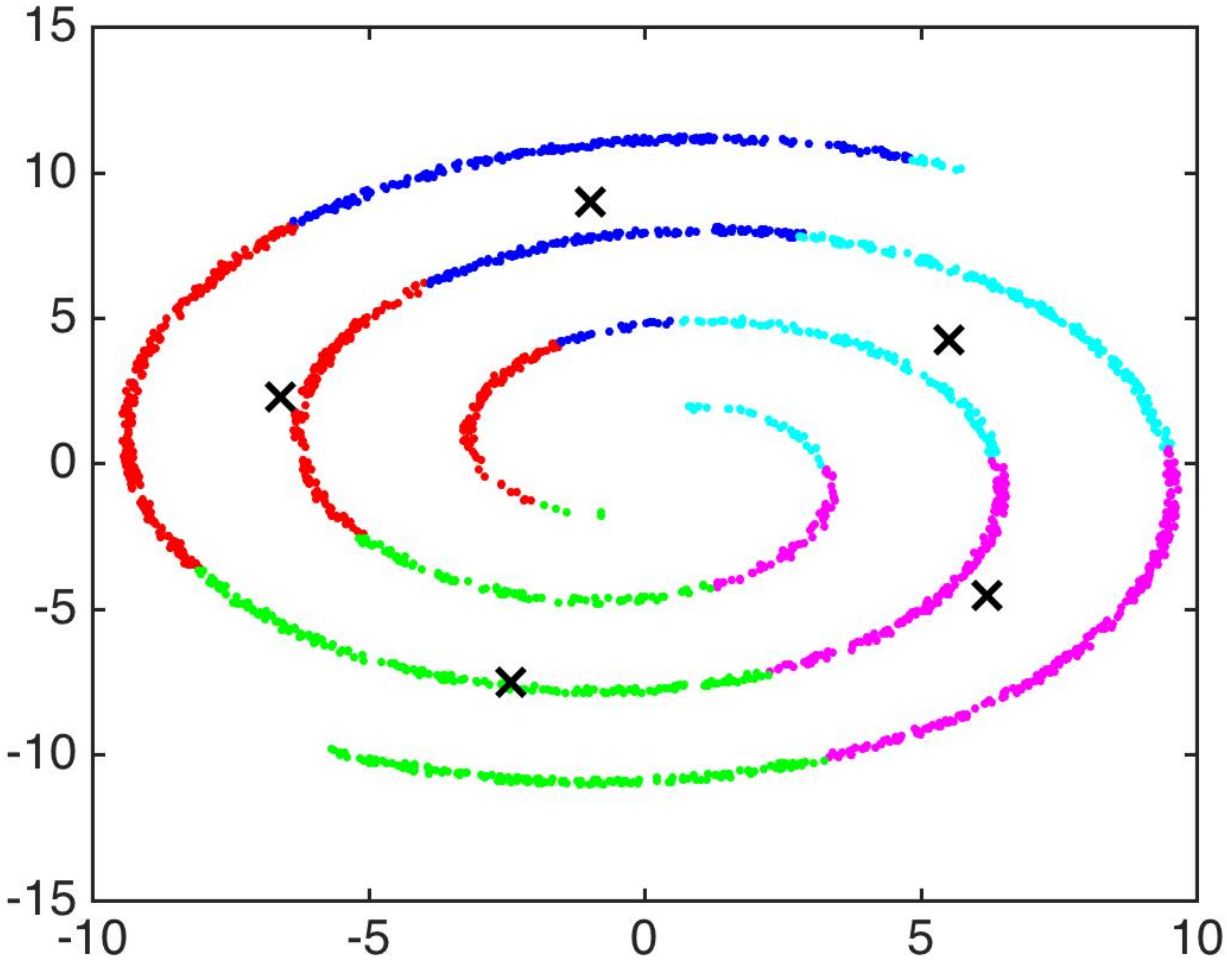

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

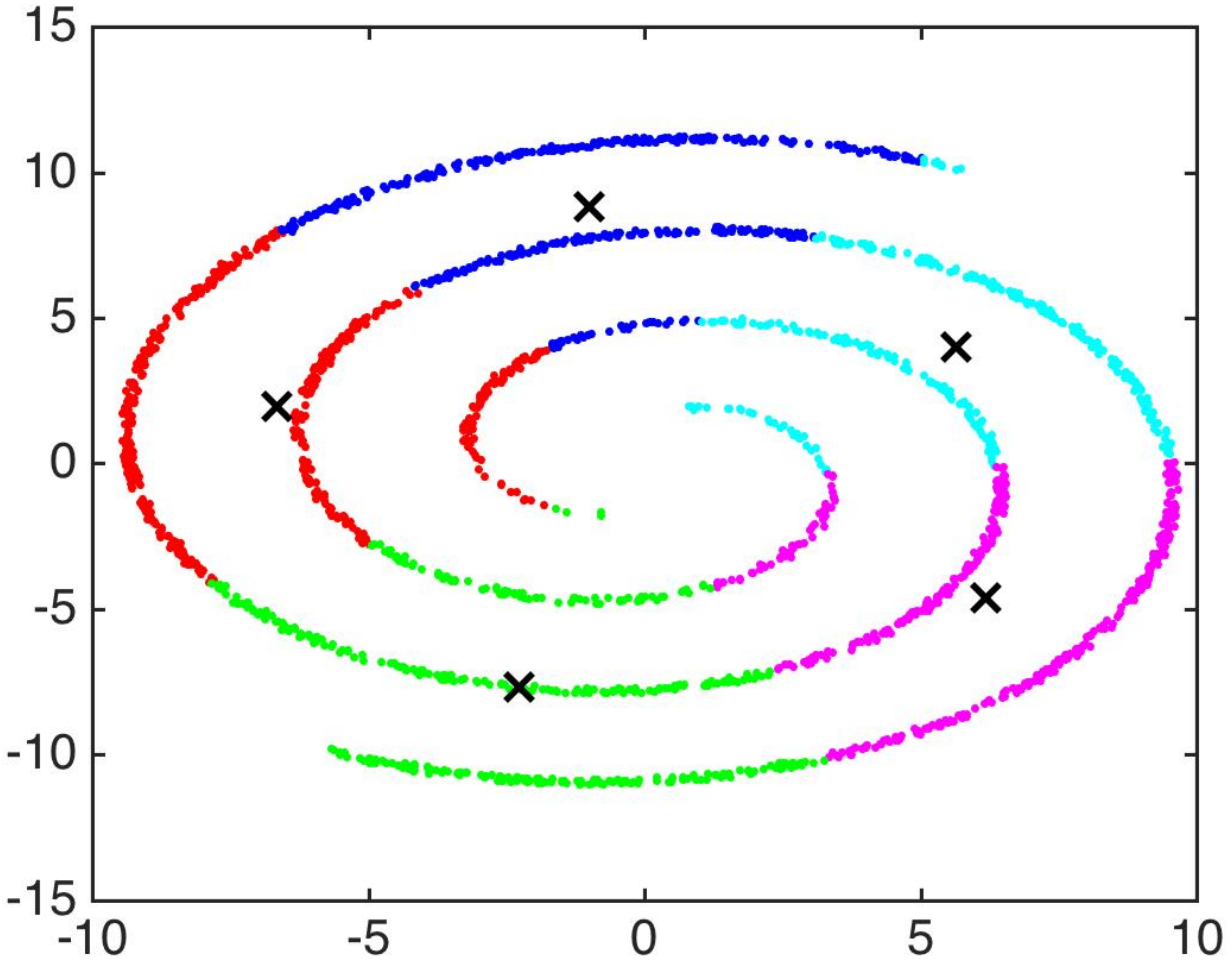

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

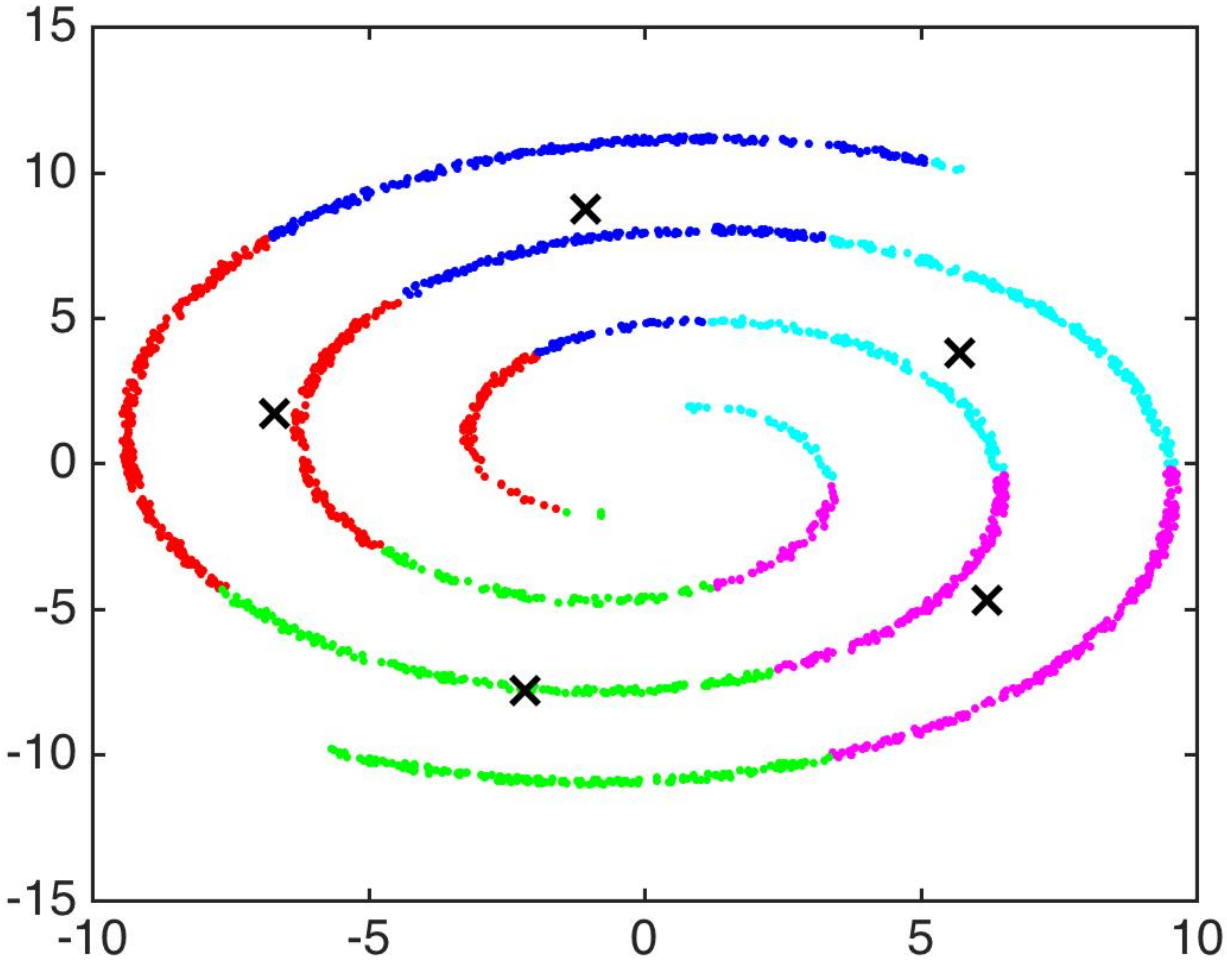

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

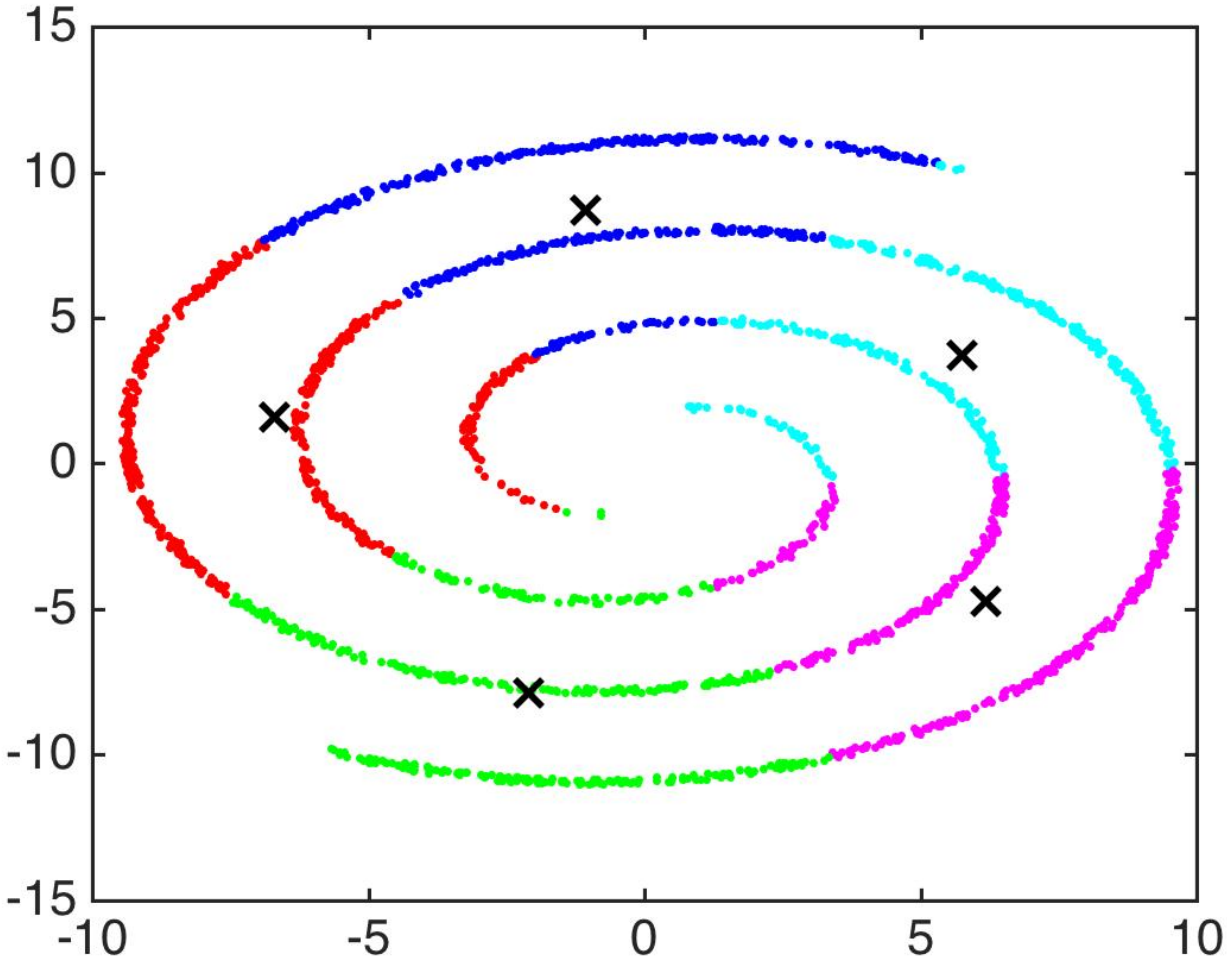

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

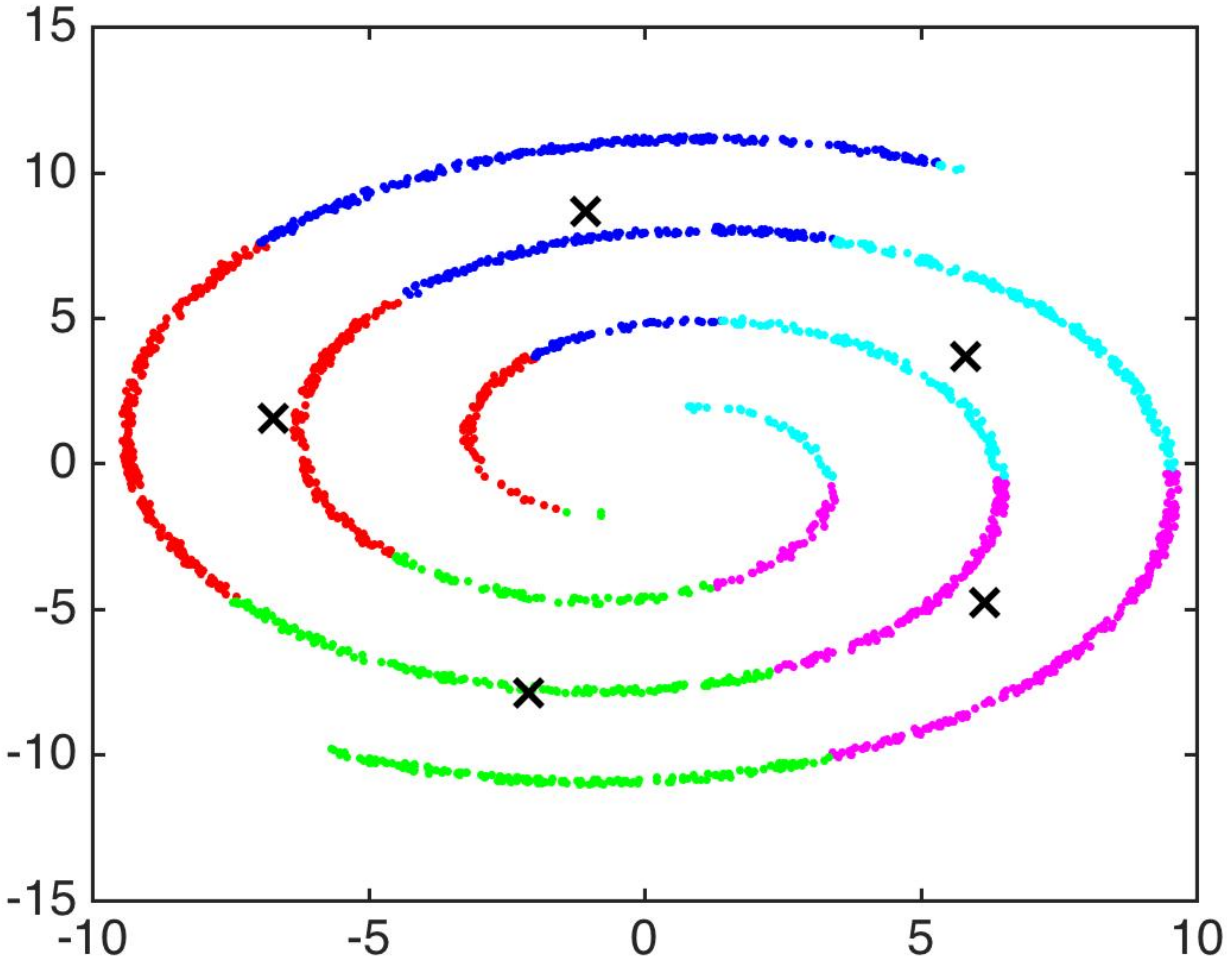

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

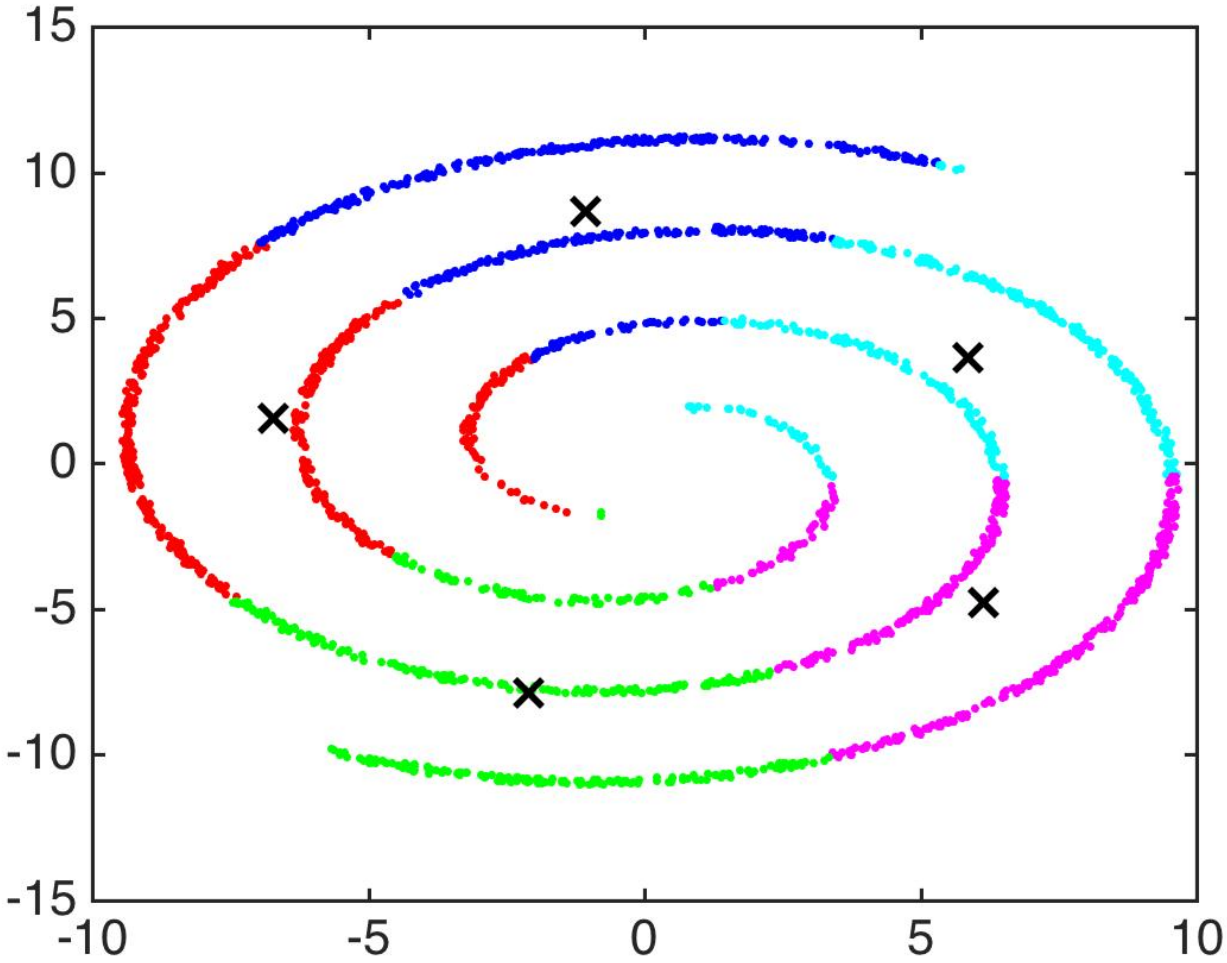

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

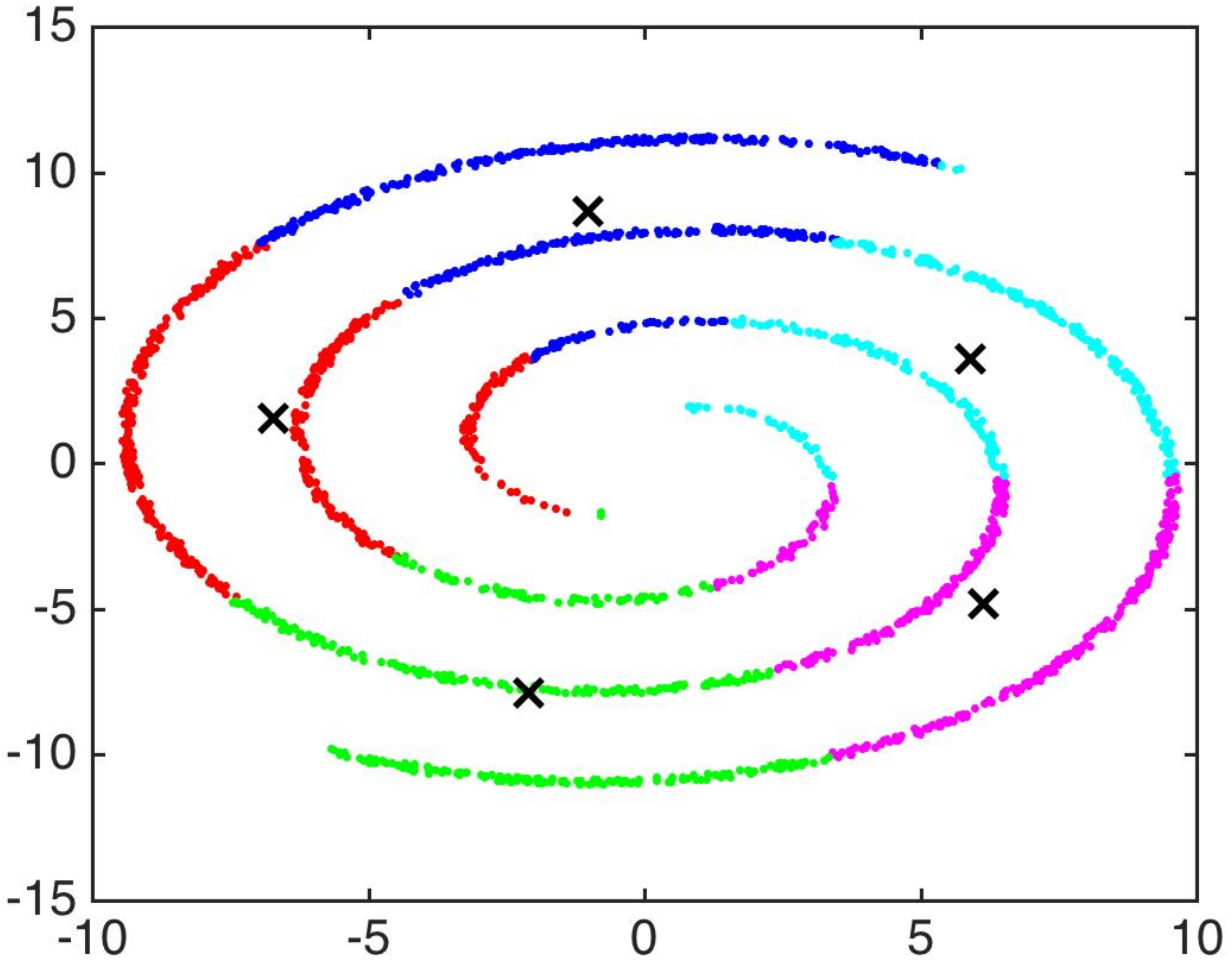

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

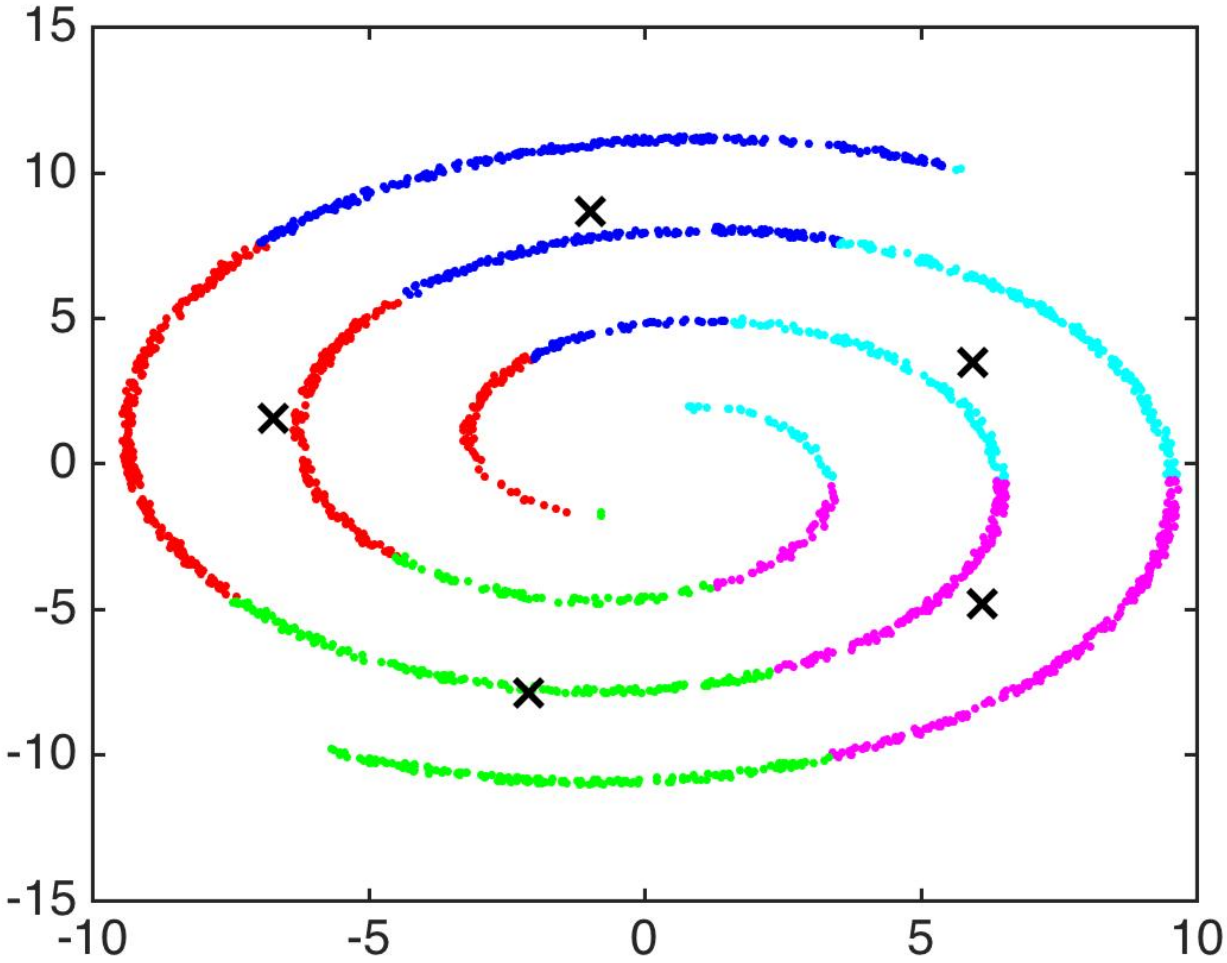

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

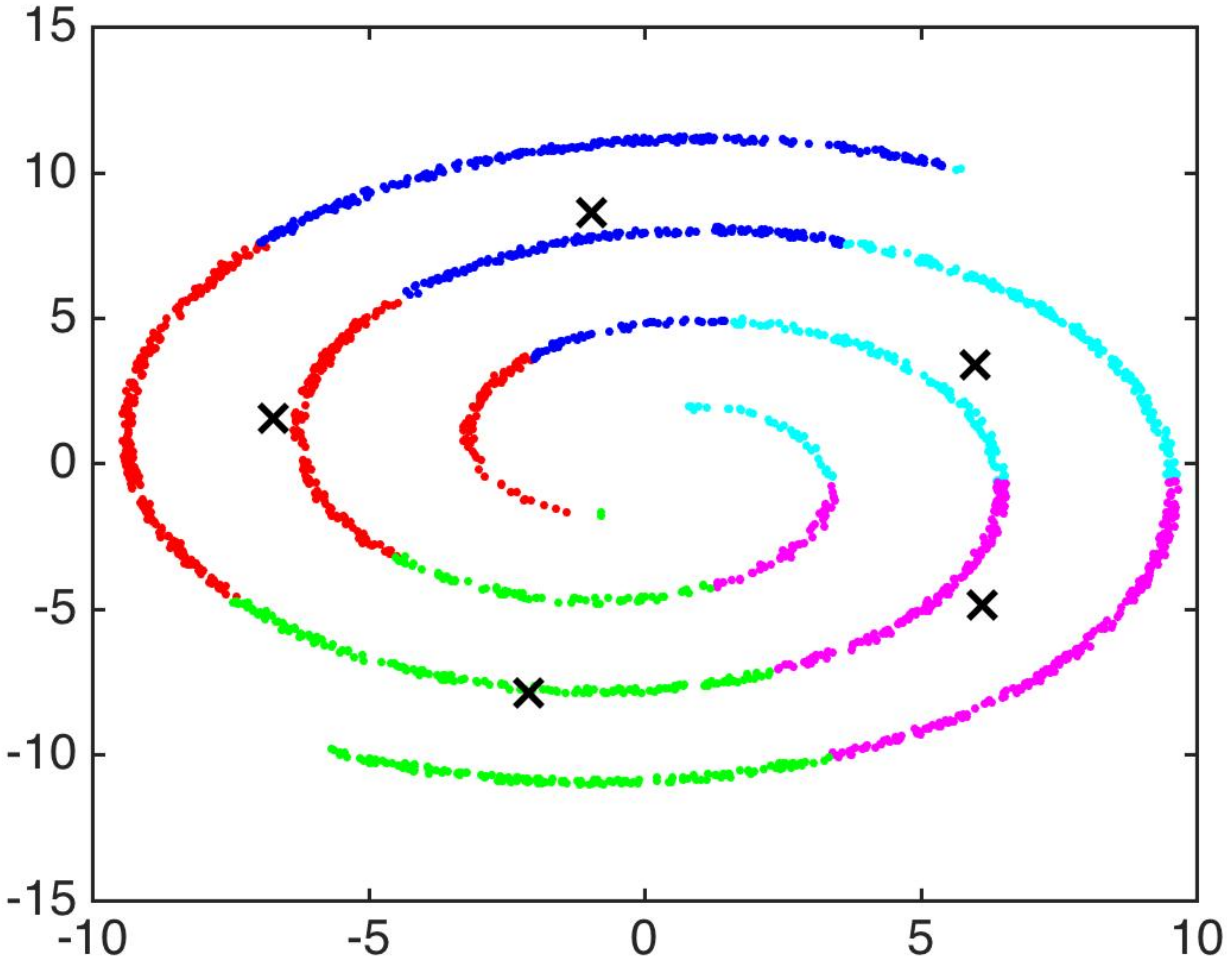

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

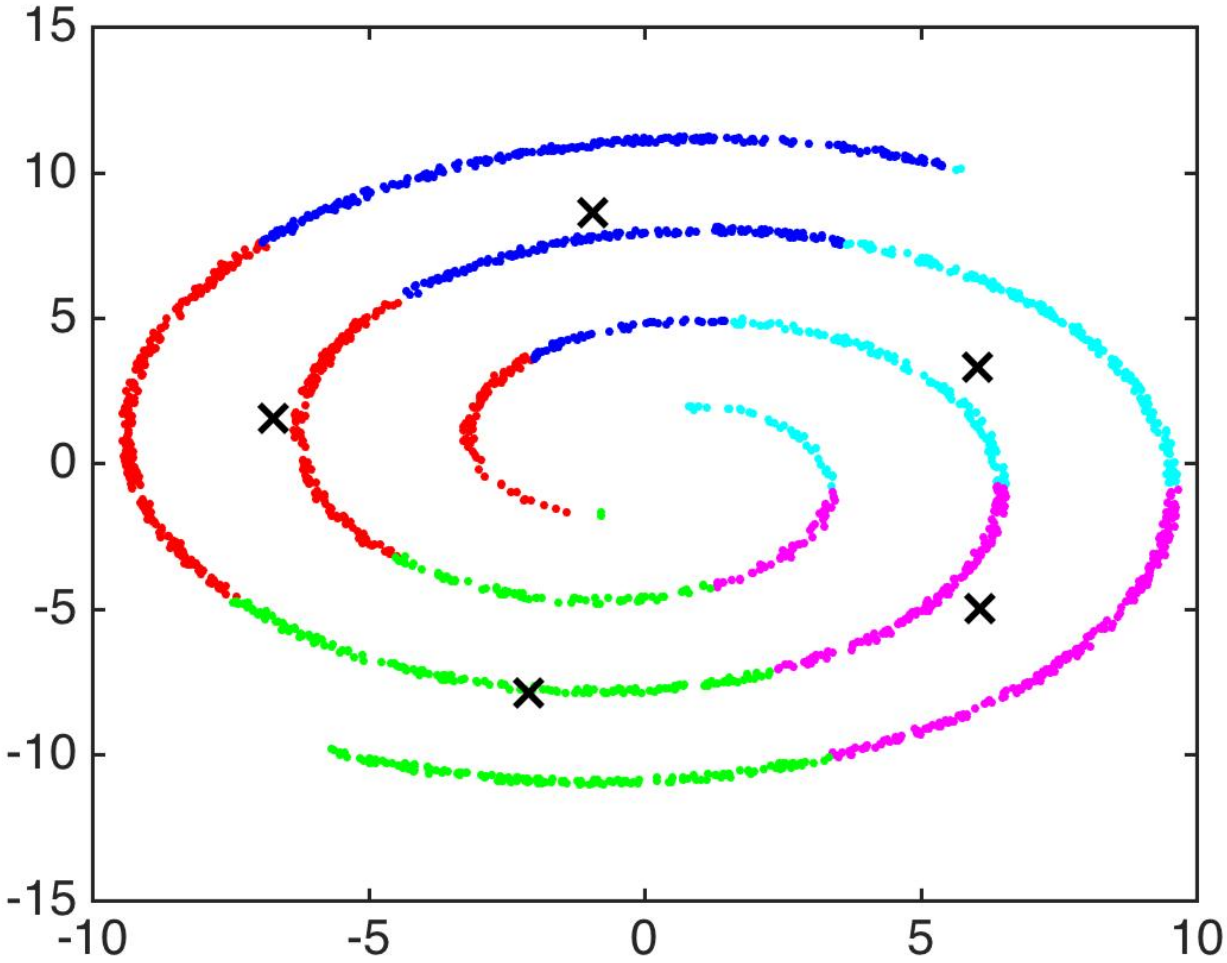

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

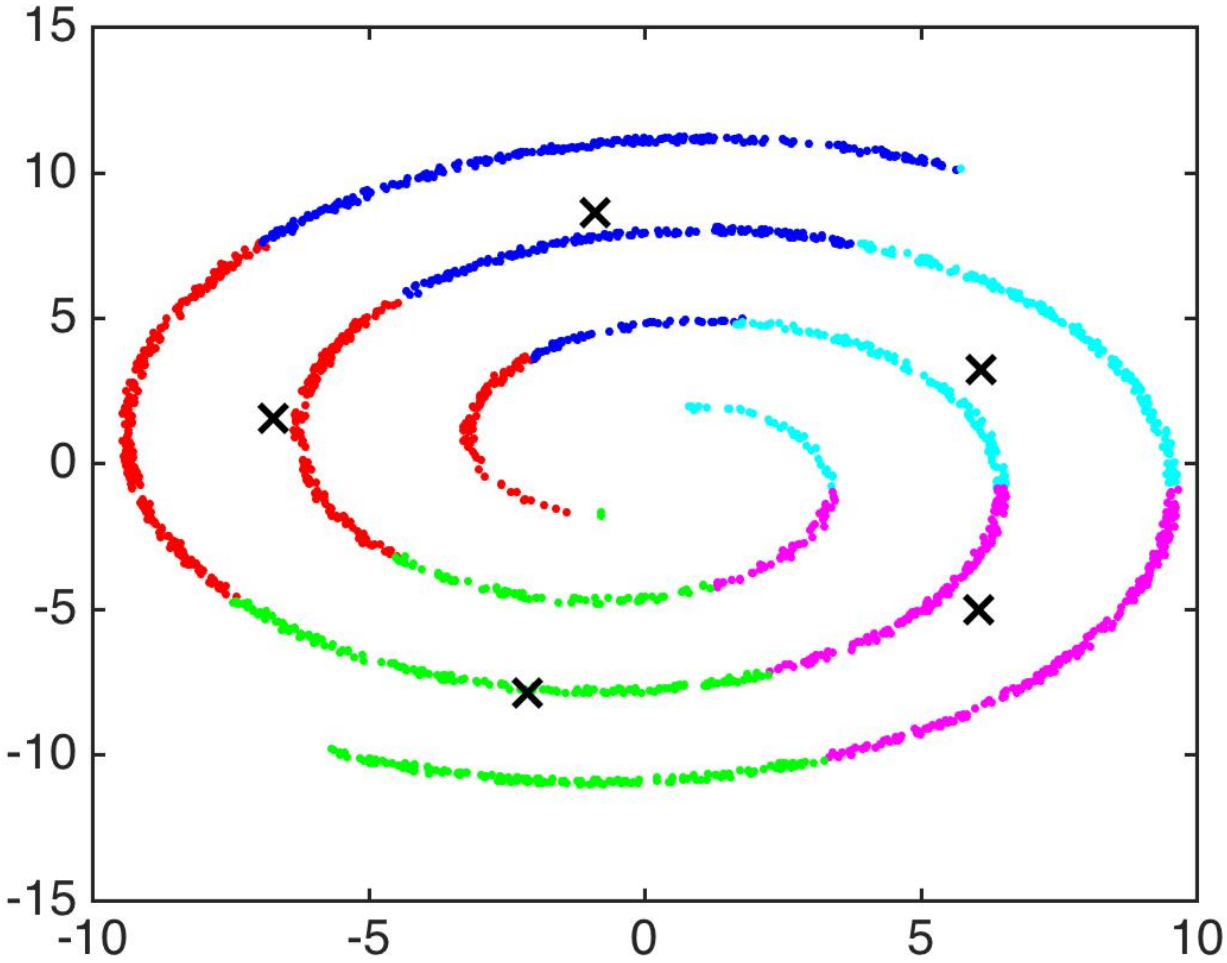

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

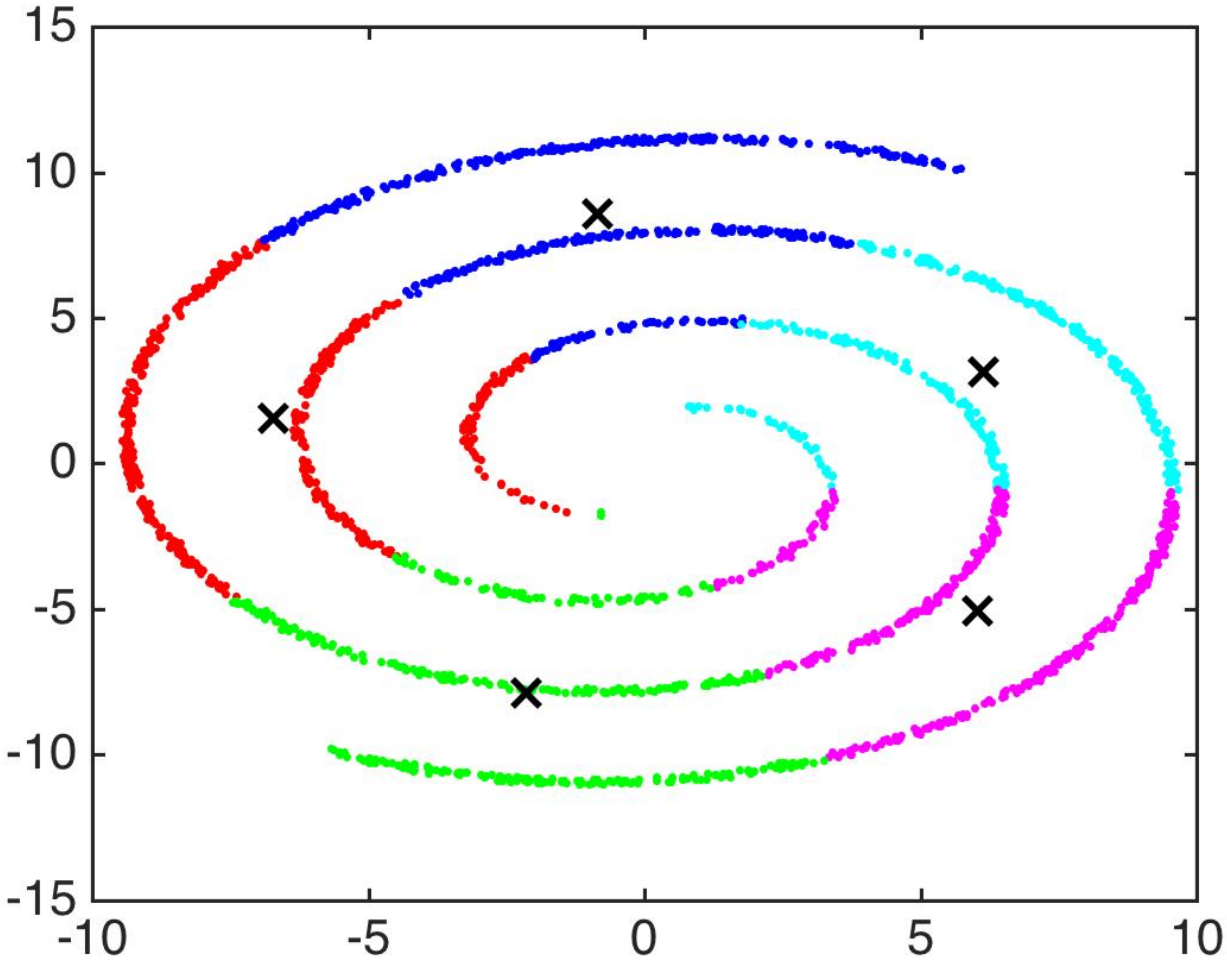

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

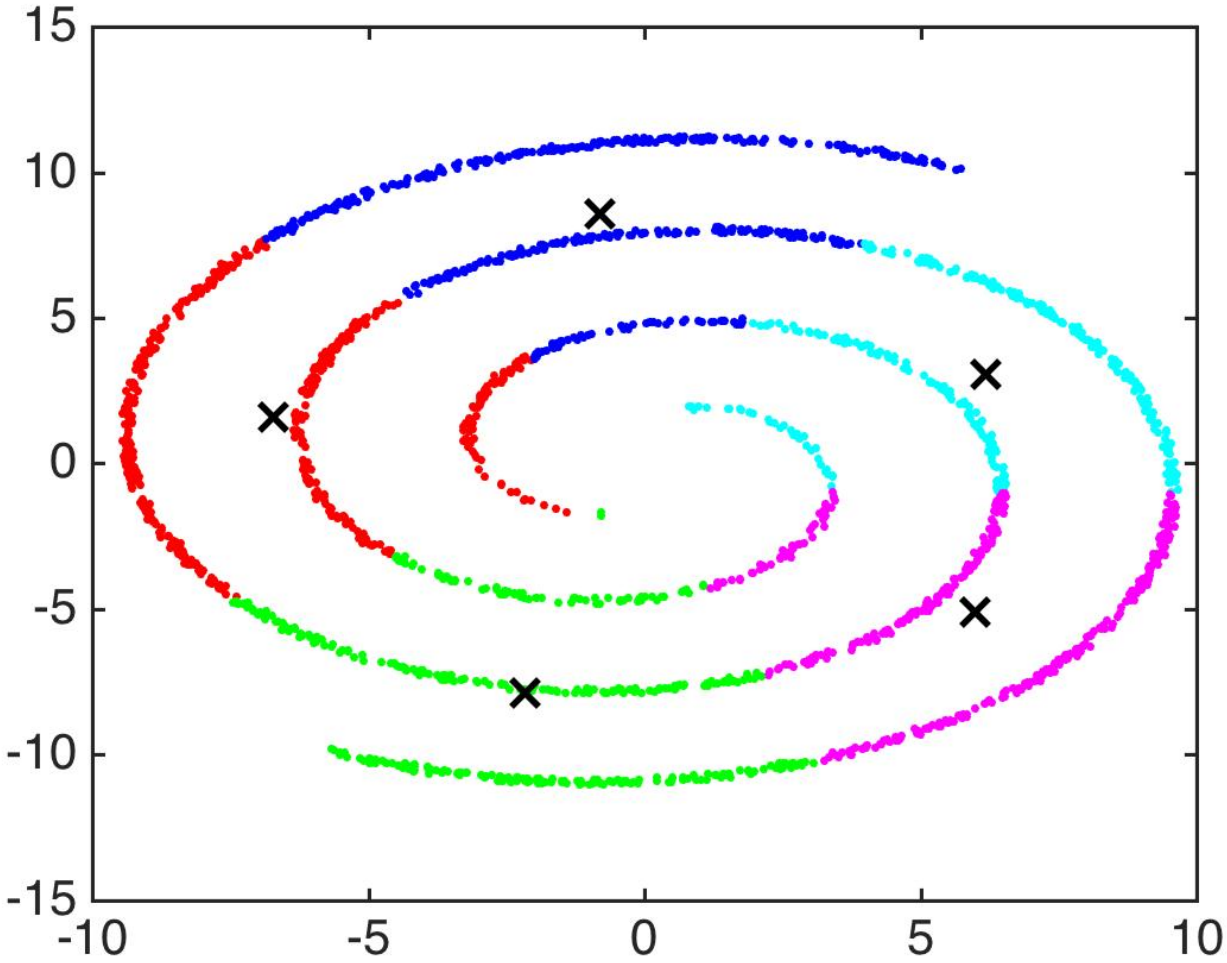

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

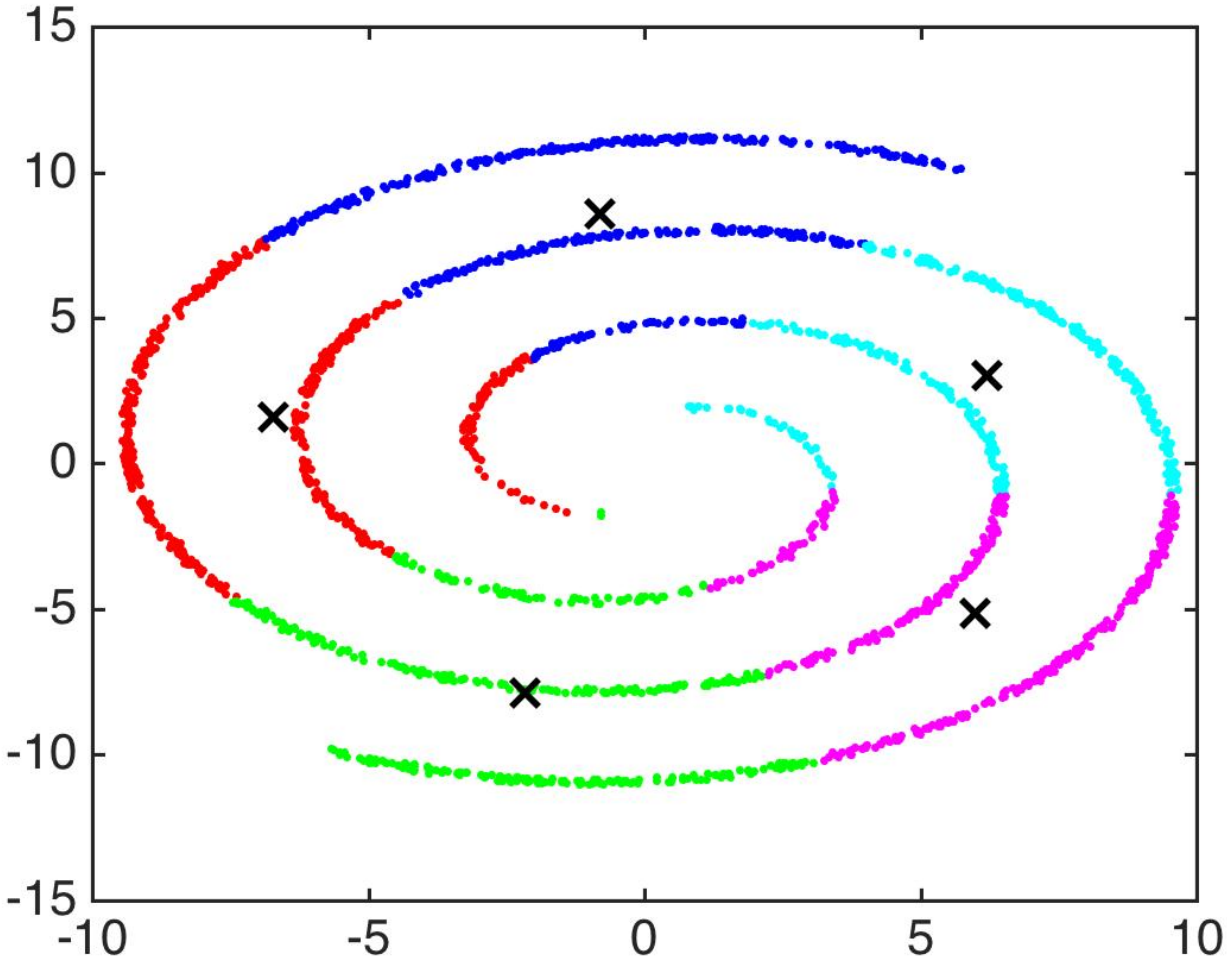

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

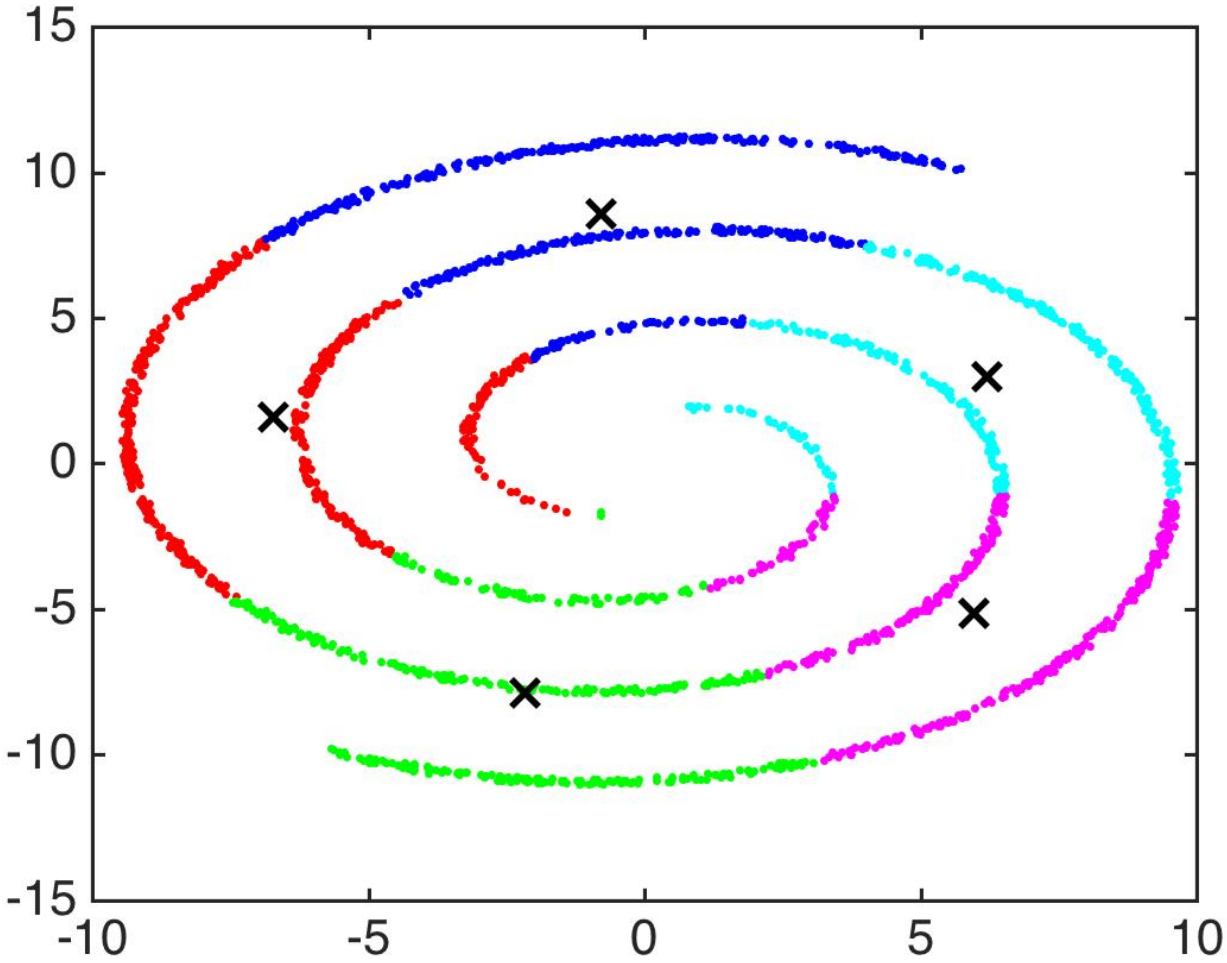

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

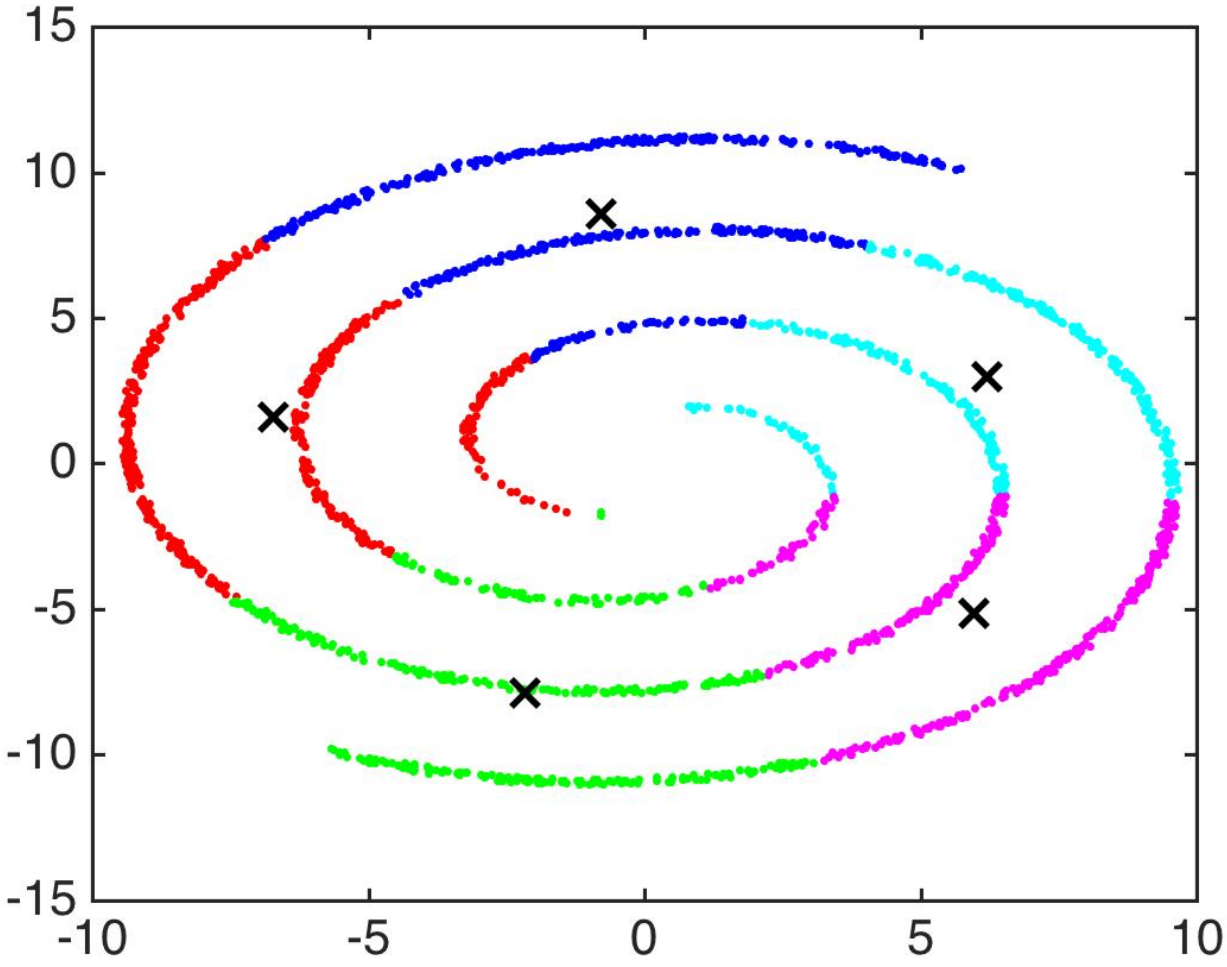

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

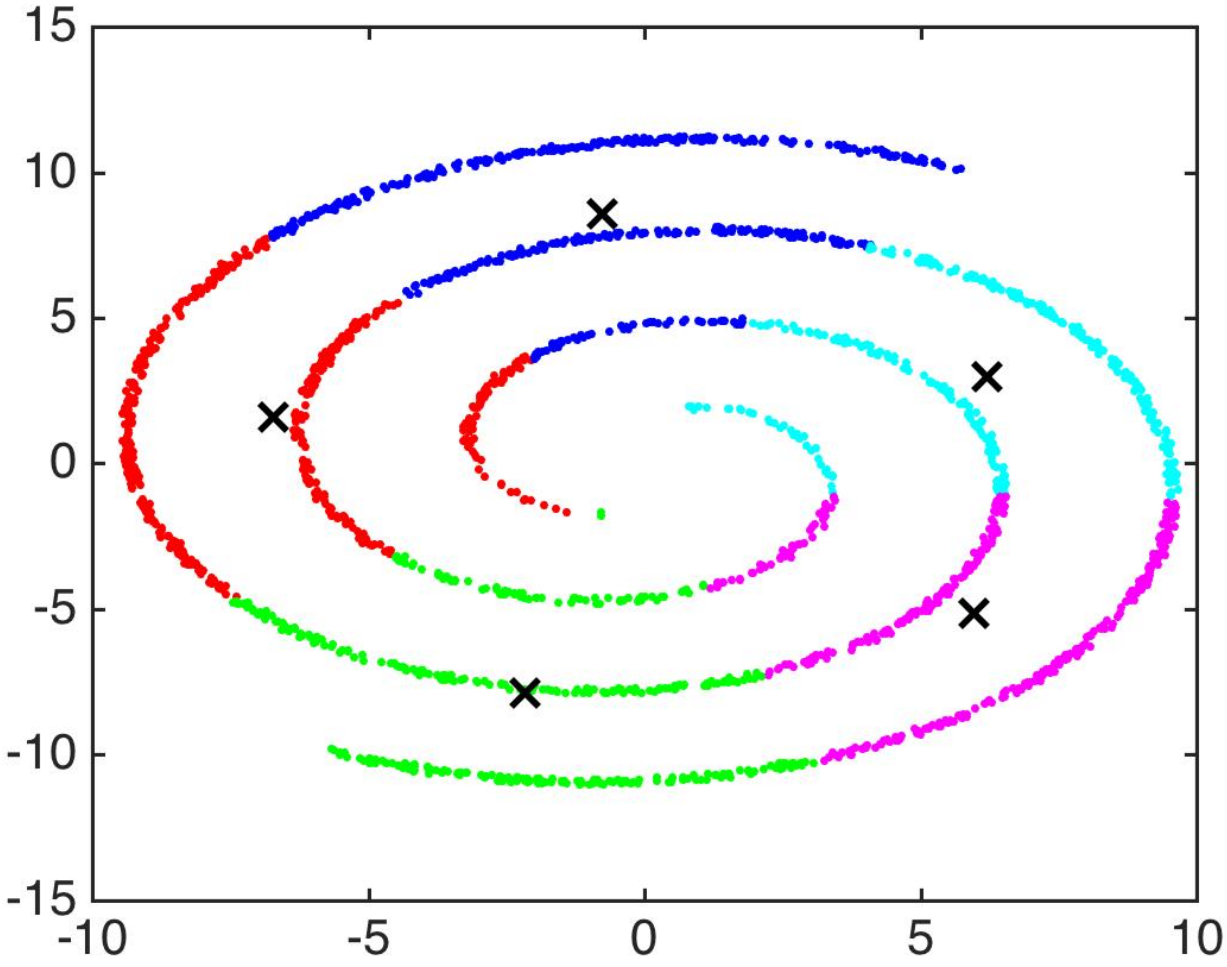

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

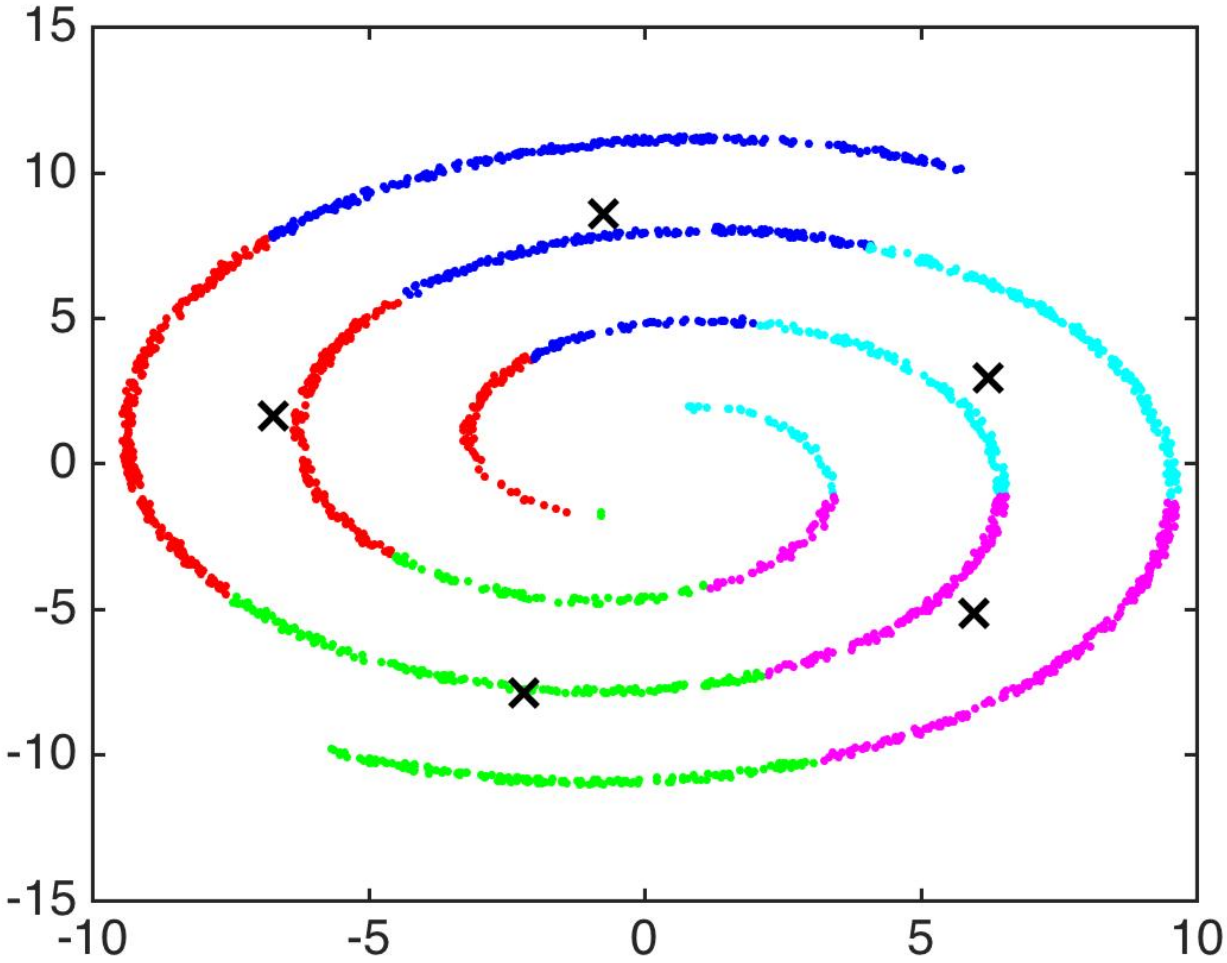

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

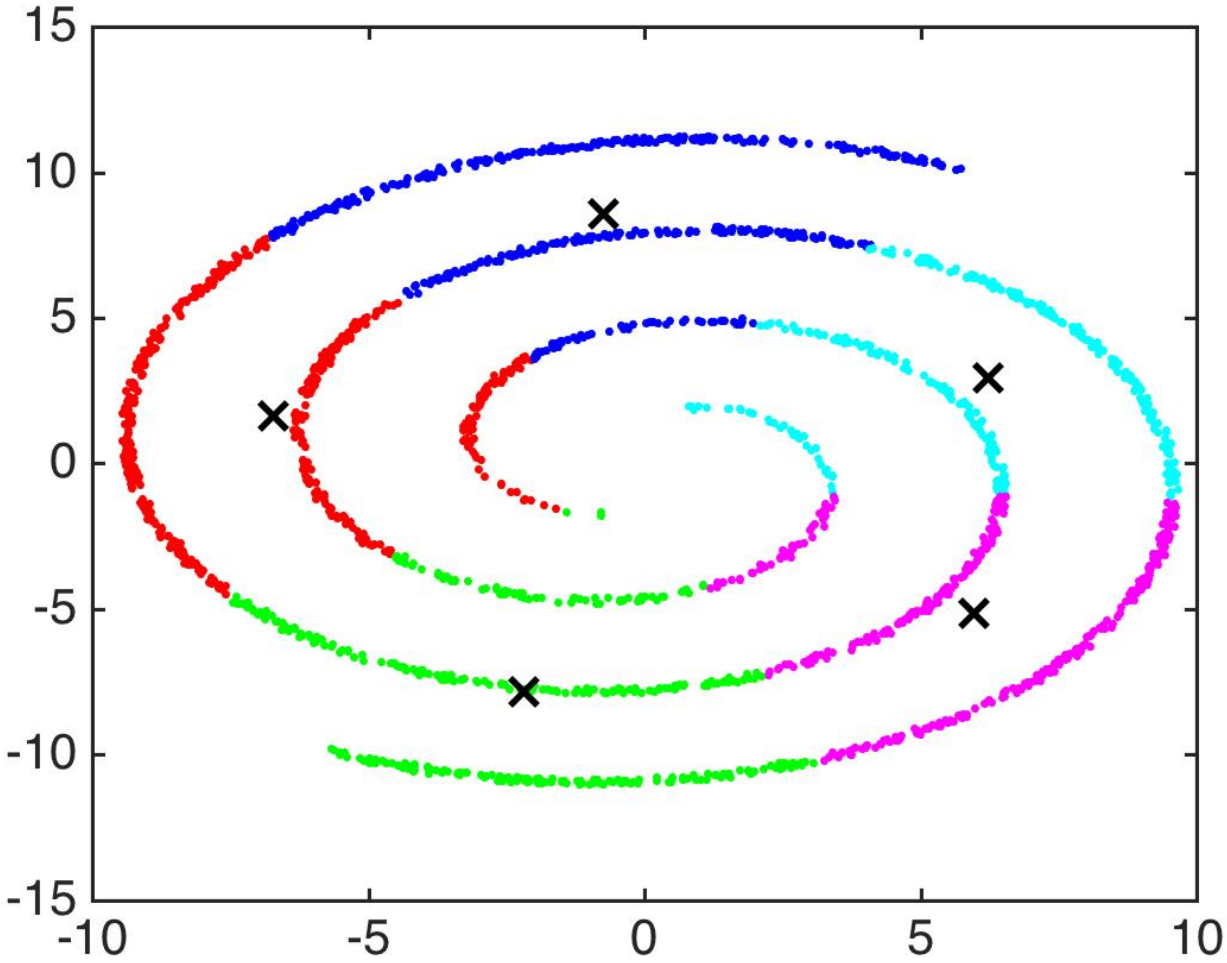

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

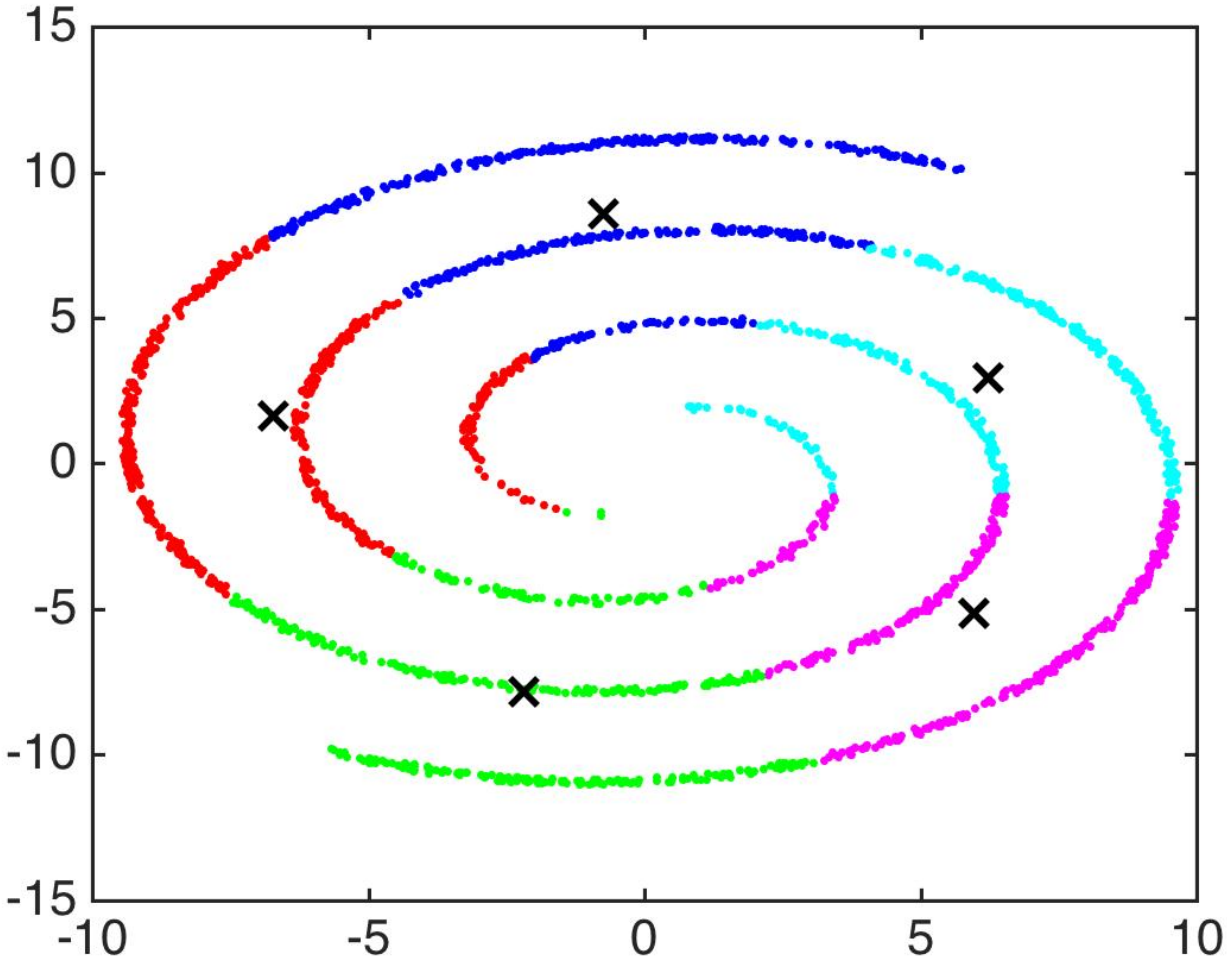

15
10
5
0
-5
-10
-15
-10
-5
0
5
10

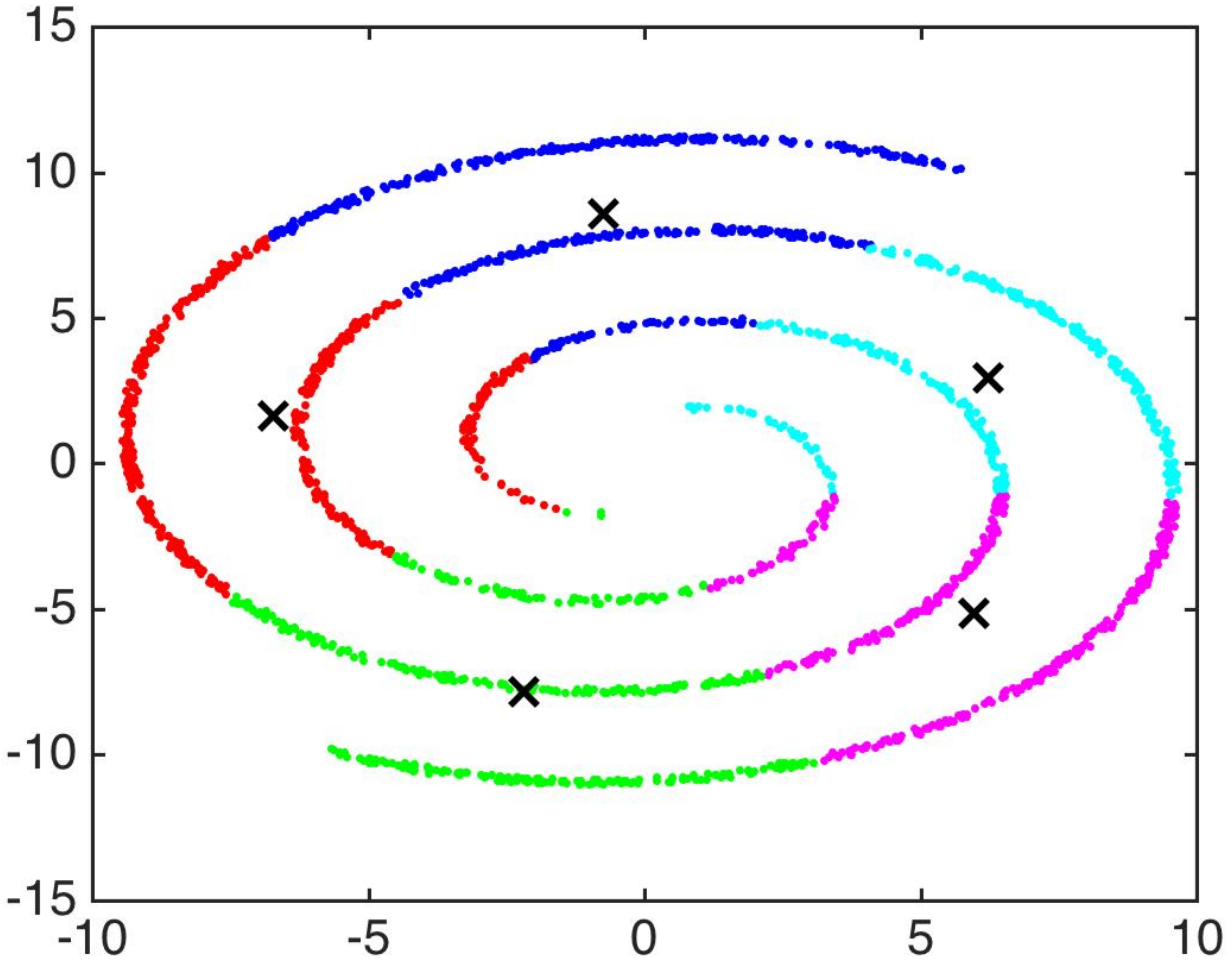

15
10
5
0
-5
-10
-15
-10
-5
0
5
10


15
10
5
0
-5
-10
-15
-10
-5
0
5
10


15
10
5
0
-5
-10
-15
-10
-5
0
5
10

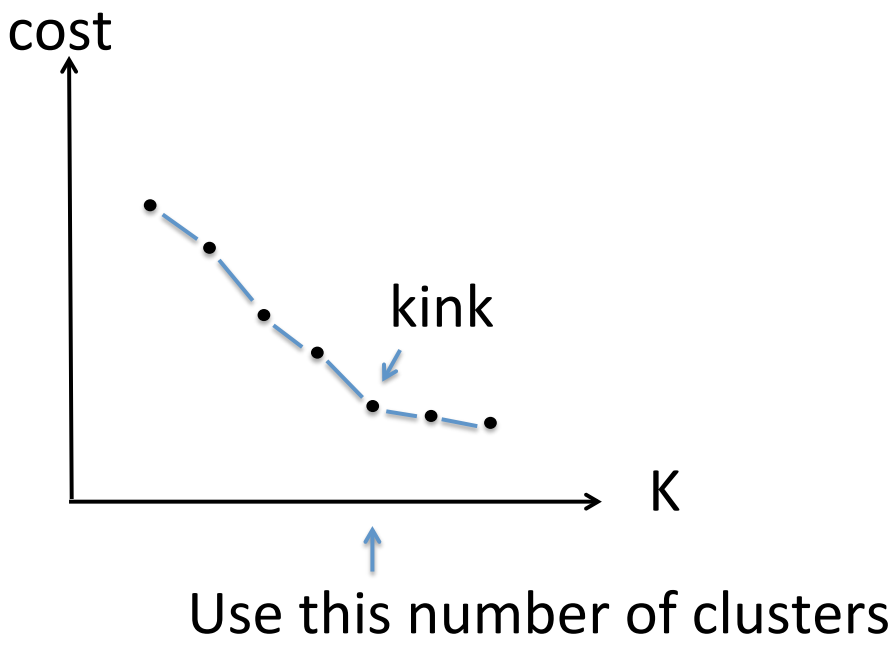

15
10
5
0
-5
-10
-15
-10
-5
0
5
10


15
10
5
0
-5
-10
-15
-10
-5
0
5
10


15
10
5
0
-5
-10
-15
-10
-5
0
5
10

How to Choose K for K-Means

How to Choose K for K-Means



- Look for the point of diminishing returns.

K-Means

- Popular clustering algorithm, computationally efficient
- Performs alternating minimization on a cost function
- Does not always fully minimize that cost function (multiple replicates might be needed for a good solution)
- Can use the cost function to evaluate whether one replicate is better than another
- Can use cost function to help choose the number of clusters
- Doesn’t work well for highly non-spherical clusters

Note: I used Euclidean distance, but can use other distances.

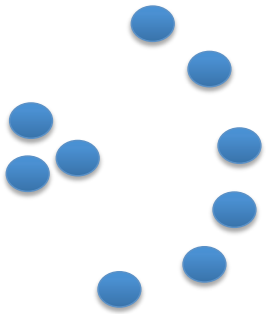
Hierarchical Agglomerative Clustering

Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points

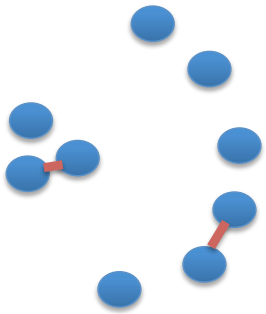
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



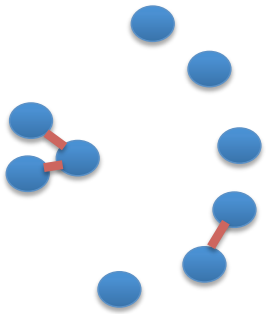
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



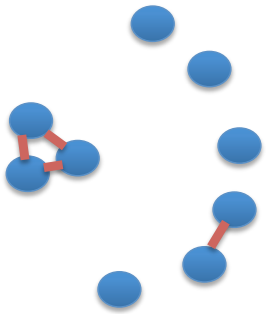
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



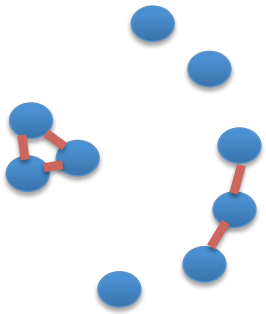
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



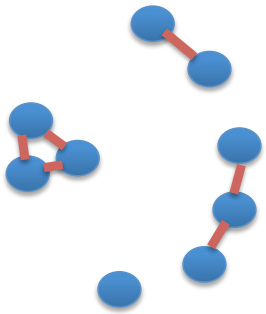
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



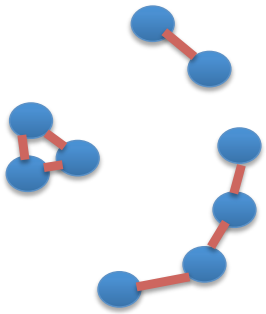
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



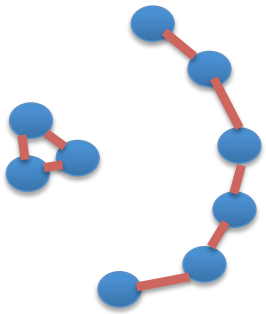
Hierarchical Agglomerative Clustering

- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points



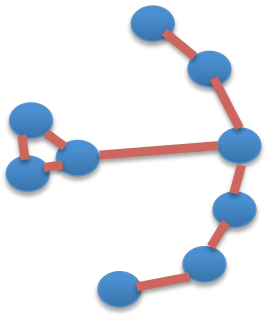
Hierarchical Agglomerative Clustering

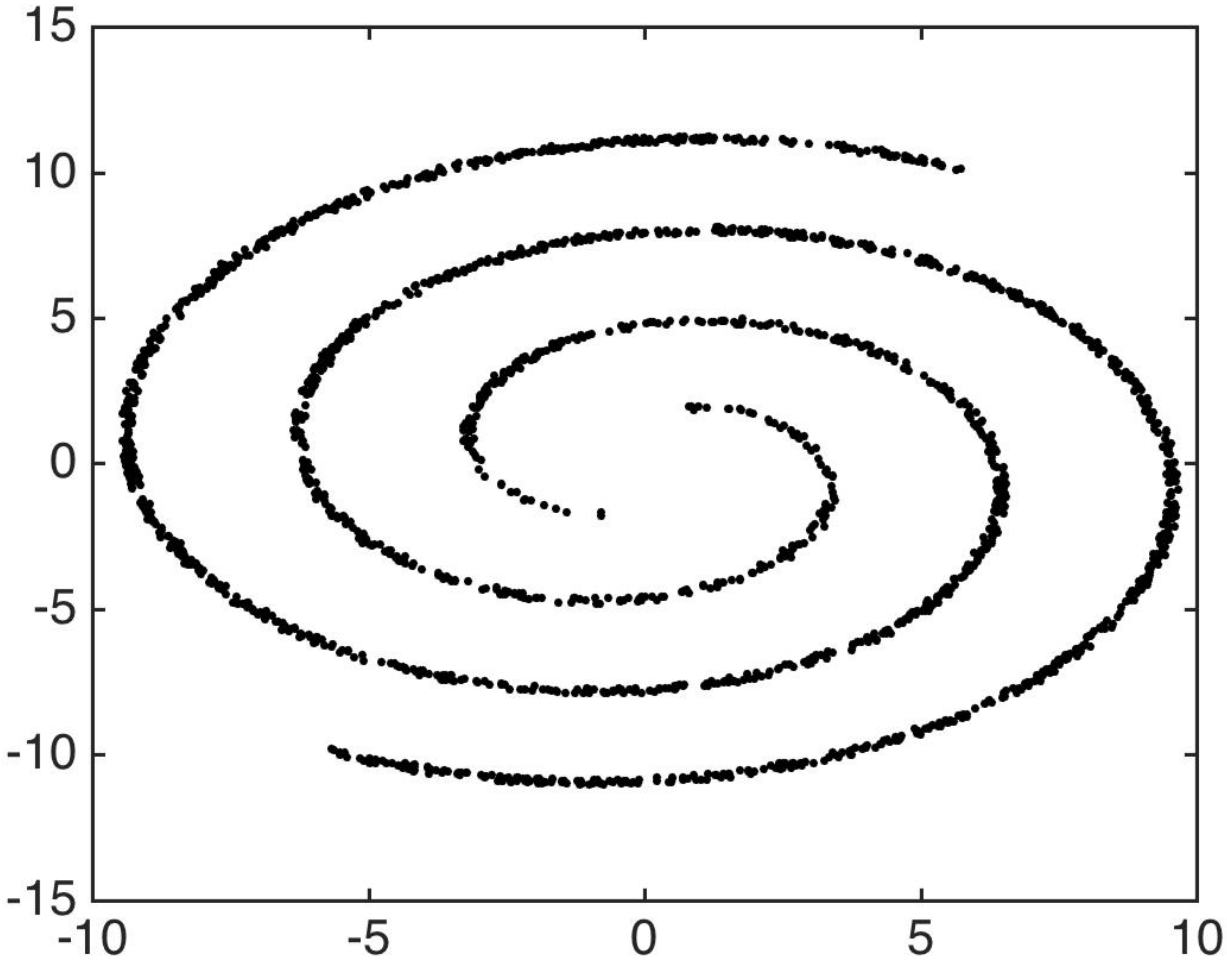
- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points

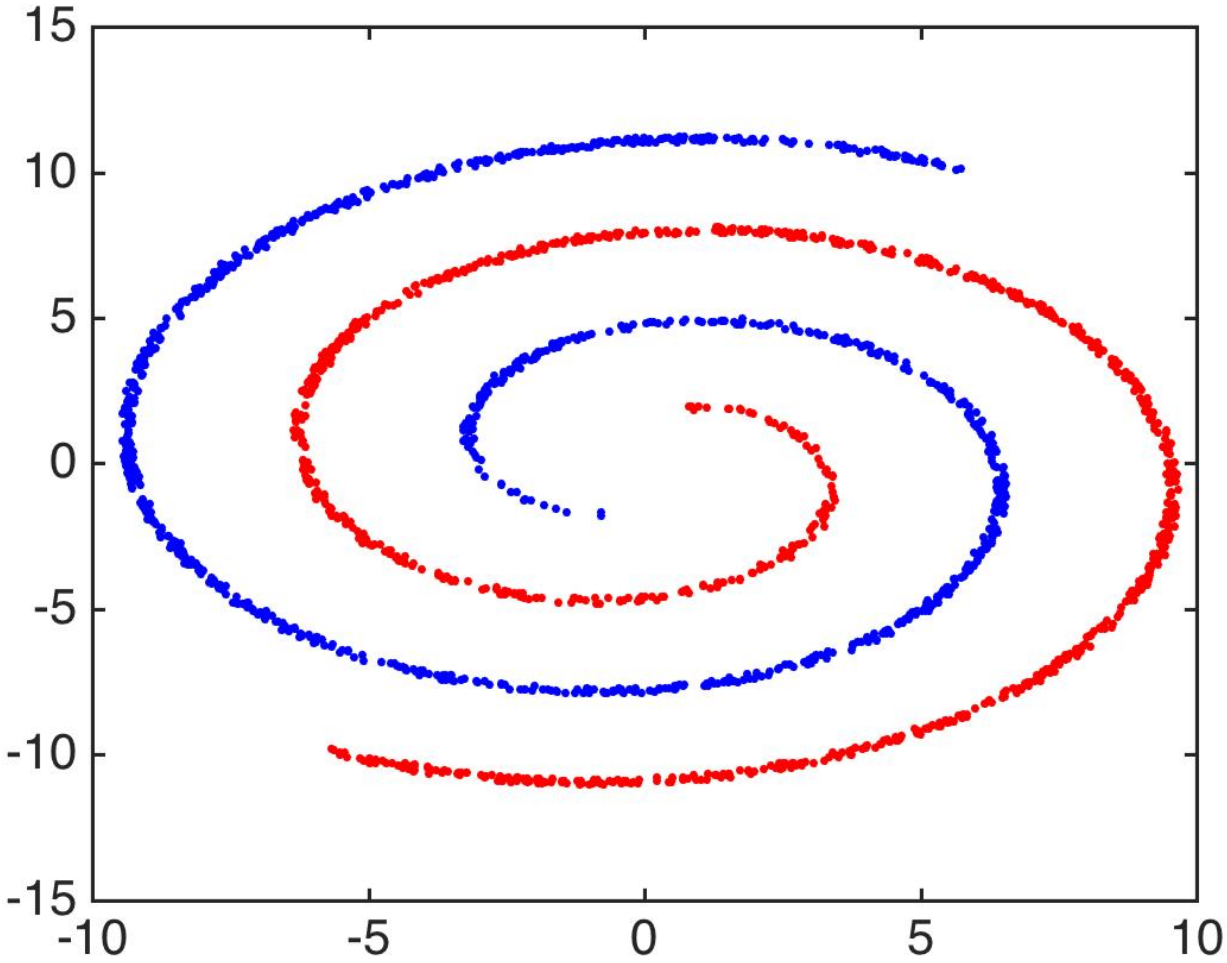


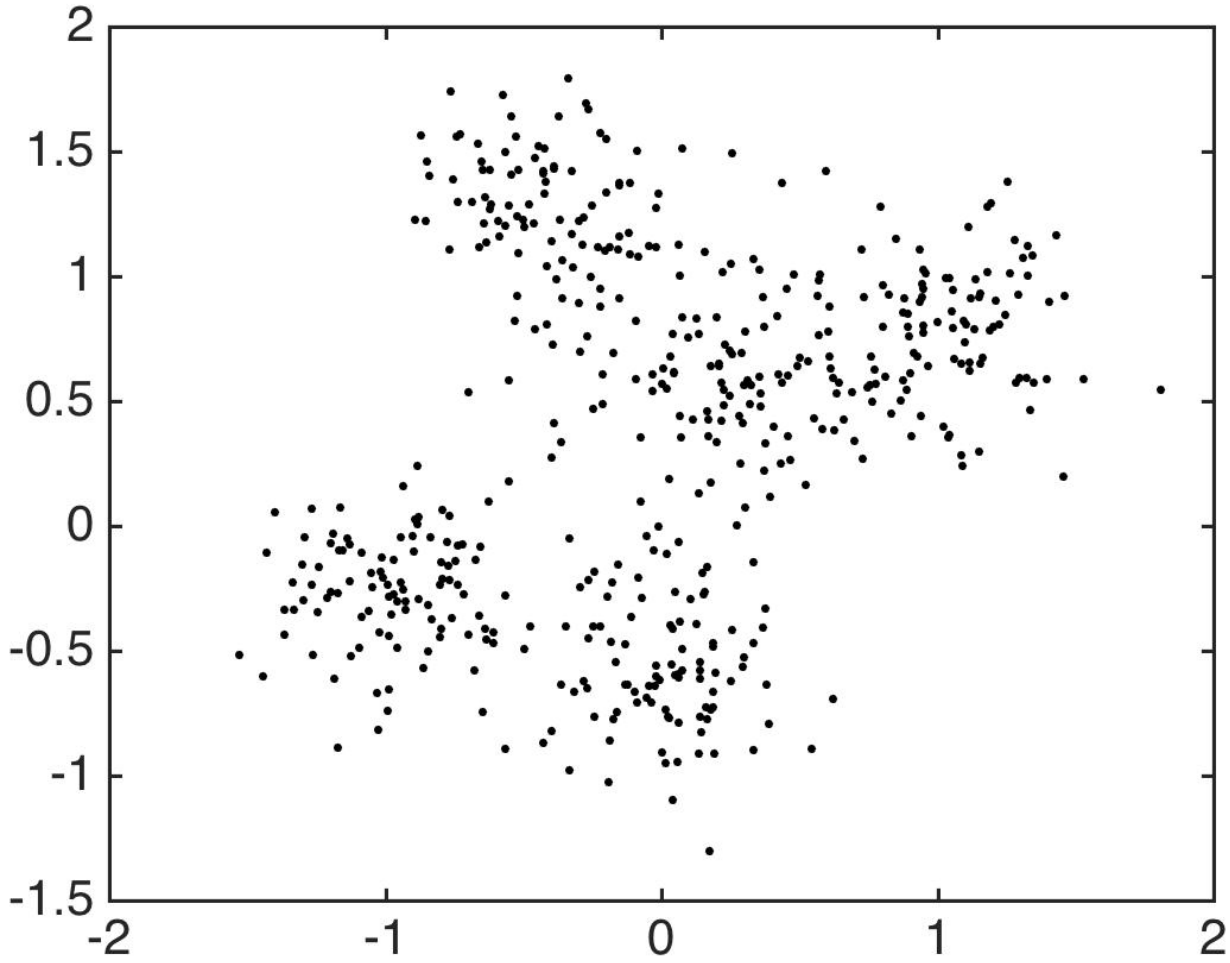
Hierarchical Agglomerative Clustering

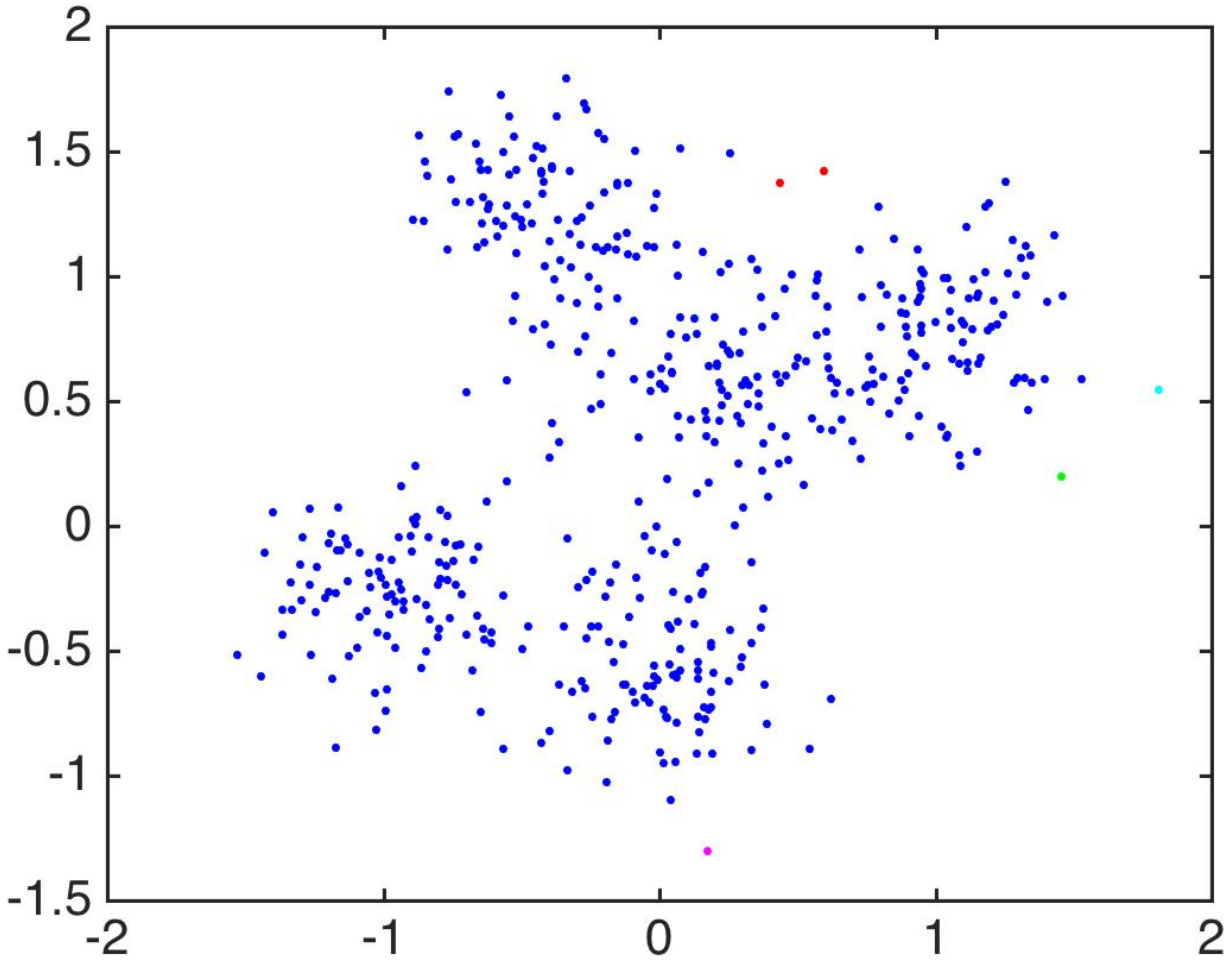
- Start with each point in its own cluster
- Repeatedly merge the clusters of the closest two points

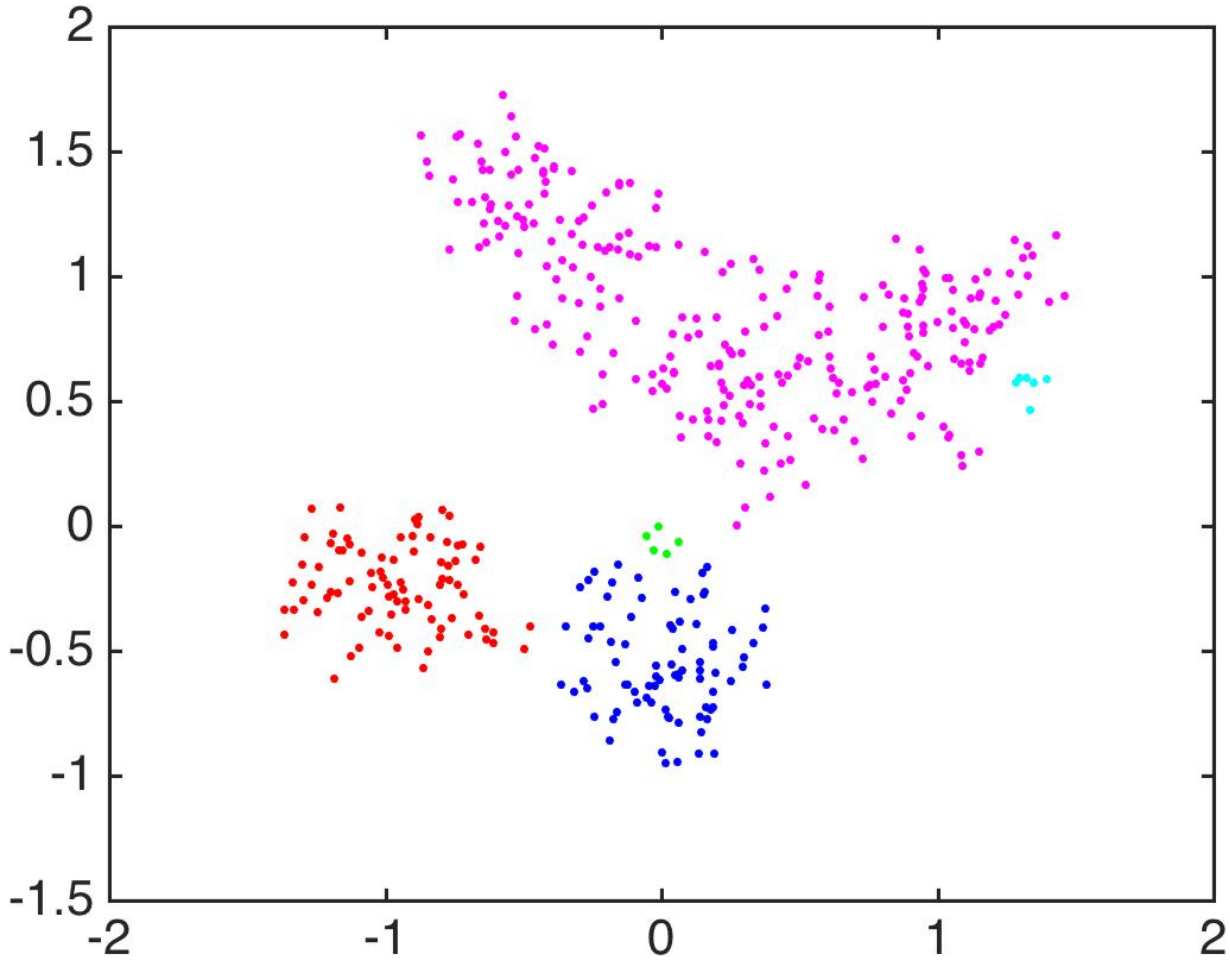



15
10
5
0
-5
-10
-15
-10
-5
0
5
10

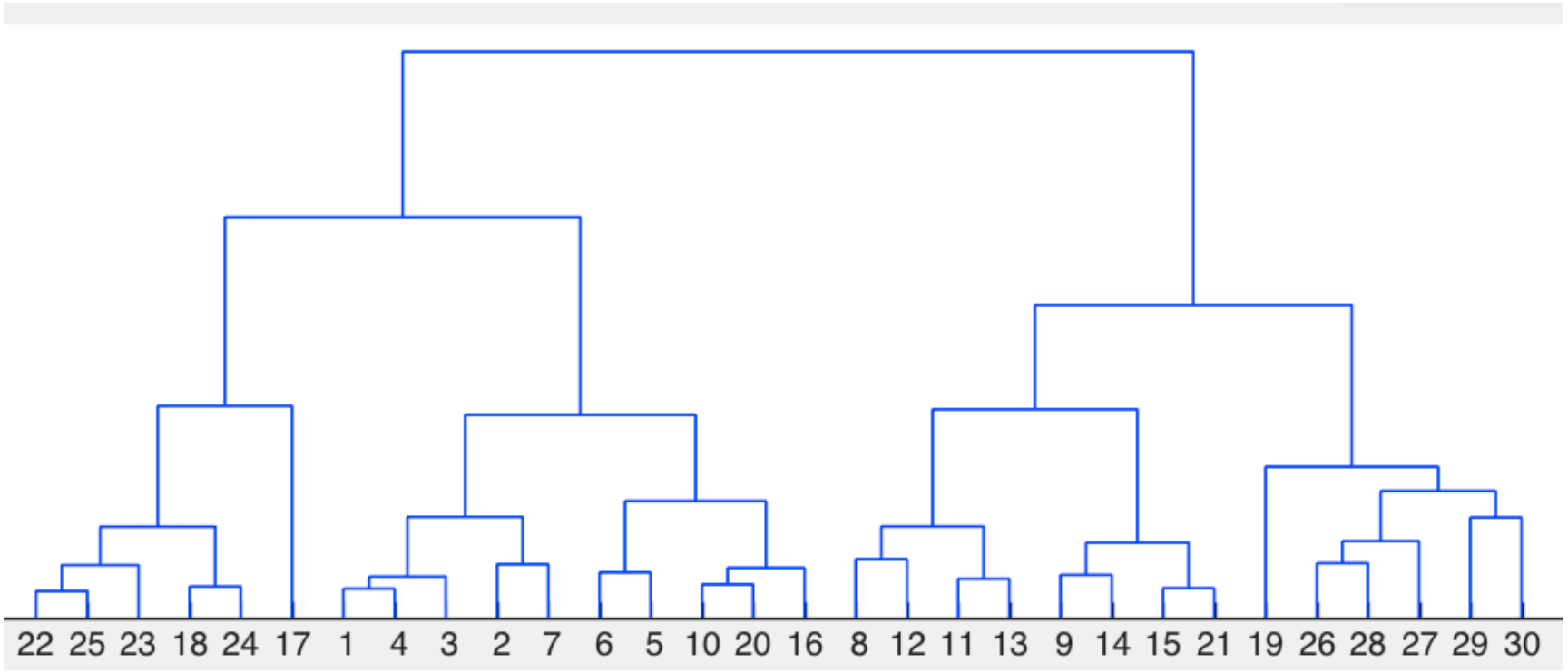

15
10
5
0
-5
-10
-15
-10
-5
0
5
10


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

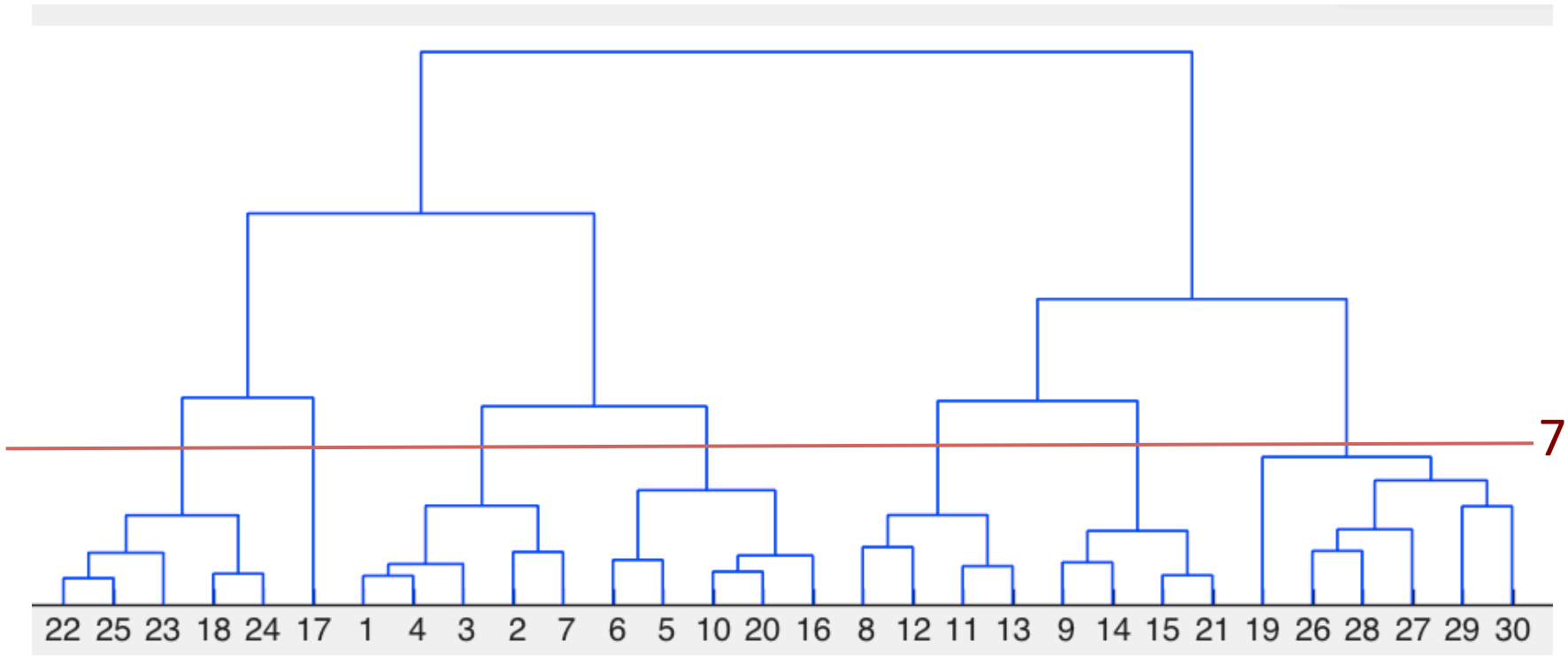

2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1
0
1
2

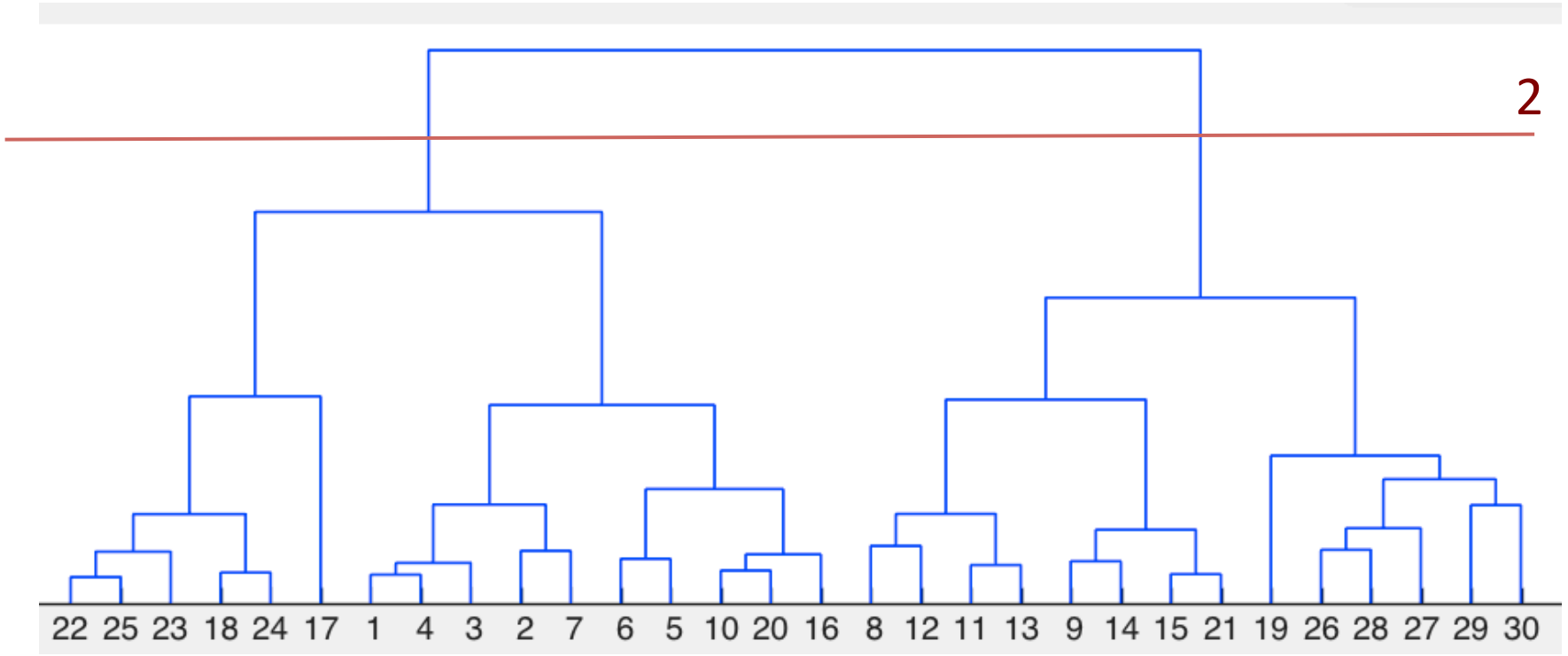
Dendrogram for Hierarchical Agglomerative Clustering



Dendrogram for Hierarchical Agglomerative Clustering



Dendrogram for Hierarchical Agglomerative Clustering



K-Means vs Hierarchical Agglomerative Clustering

- Both are useful for different sorts of problems. K-Means works well for spherical data.
- Hierarchical Agglomerative Clustering is useful when clusters are well-separated. (This means data that are close together should be in the same cluster.)
- For K-means, one needs to choose the number of clusters (try a few different ones). For hierarchical clustering, one chooses when to stop merging clusters.
- The distance metric is important. Can have a large impact on the solution.