Introduction to Machine Learning Applications

Spring 2023

Exam 2 review

Minor Gordon

gordom6@rpi.edu

Rensselaer

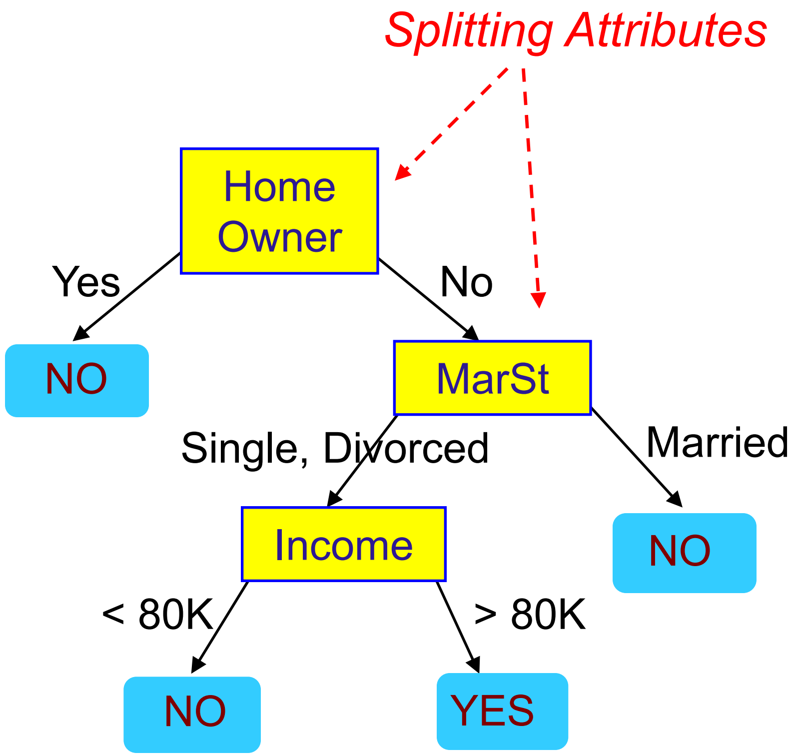
Decision trees

Example of a Decision Tree

categorical categorical continuous class

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data

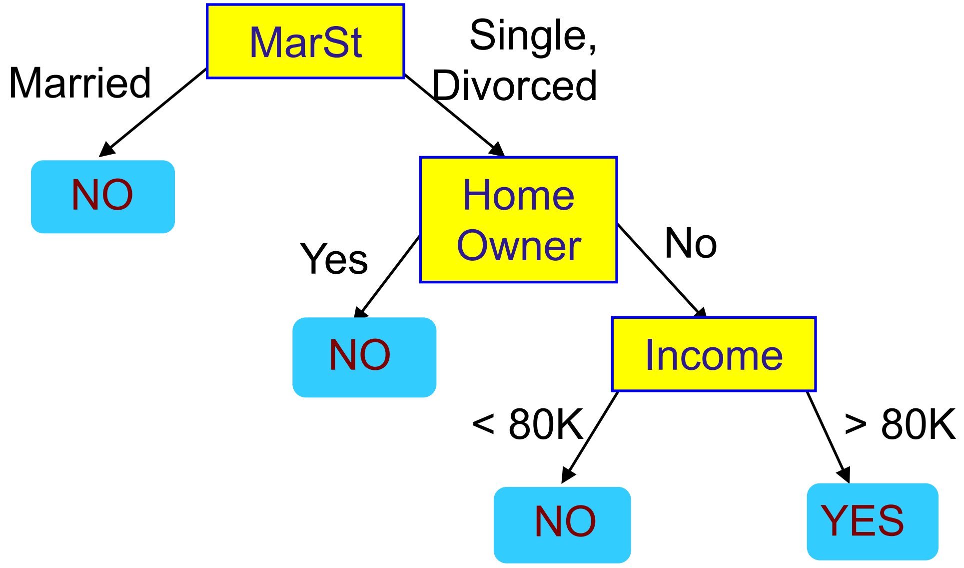


Model: Decision Tree

Another Example of Decision Tree

categorical categorical continuous class

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**



There could be more than one tree that fits the same data!

Design Issues of Decision Tree Induction

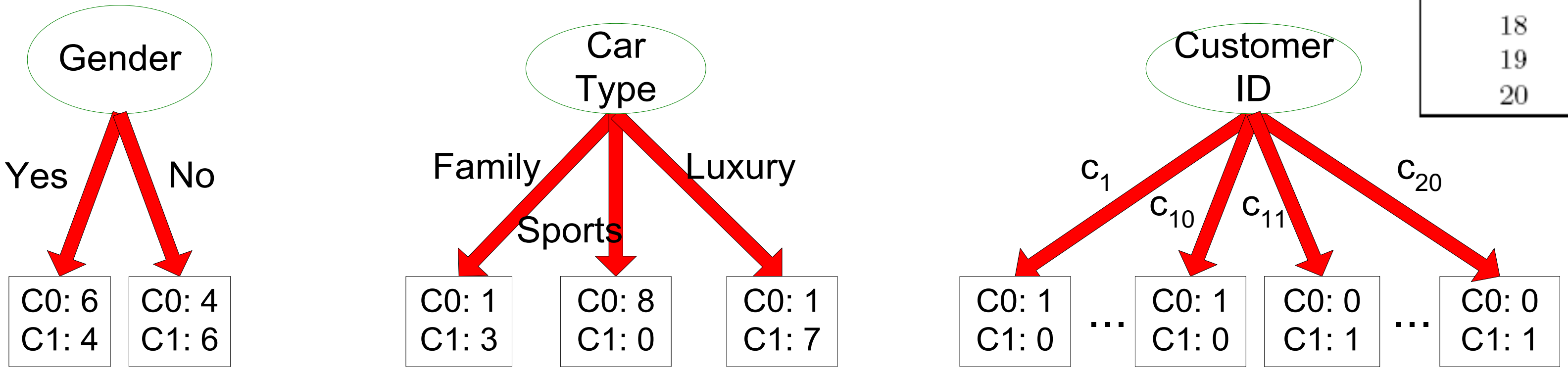
- How should training **records be split**?
  - Method for specifying test condition
    - depending on attribute types
  - Measure for evaluating the goodness of a test condition

- How should the **splitting procedure stop**?
  - Stop splitting if all the records belong to the same class or have identical attribute values
  - Early termination

How to determine the best split

Before Splitting: 10 records of class 0, 10 records of class 1

Customer Id	Gender	Car Type	Shirt Size	Class
1	M	Family	Small	C0
2	M	Sports	Medium	C0
3	M	Sports	Medium	C0
4	M	Sports	Large	C0
5	M	Sports	Extra Large	C0
6	M	Sports	Extra Large	C0
7	F	Sports	Small	C0
8	F	Sports	Small	C0
9	F	Sports	Medium	C0
10	F	Luxury	Large	C0
11	M	Family	Large	C1
12	M	Family	Extra Large	C1
13	M	Family	Medium	C1
14	M	Luxury	Extra Large	C1
15	F	Luxury	Small	C1
16	F	Luxury	Small	C1
17	F	Luxury	Medium	C1
18	F	Luxury	Medium	C1
19	F	Luxury	Medium	C1
20	F	Luxury	Large	C1



Which test condition is the best?

Ensemble modeling

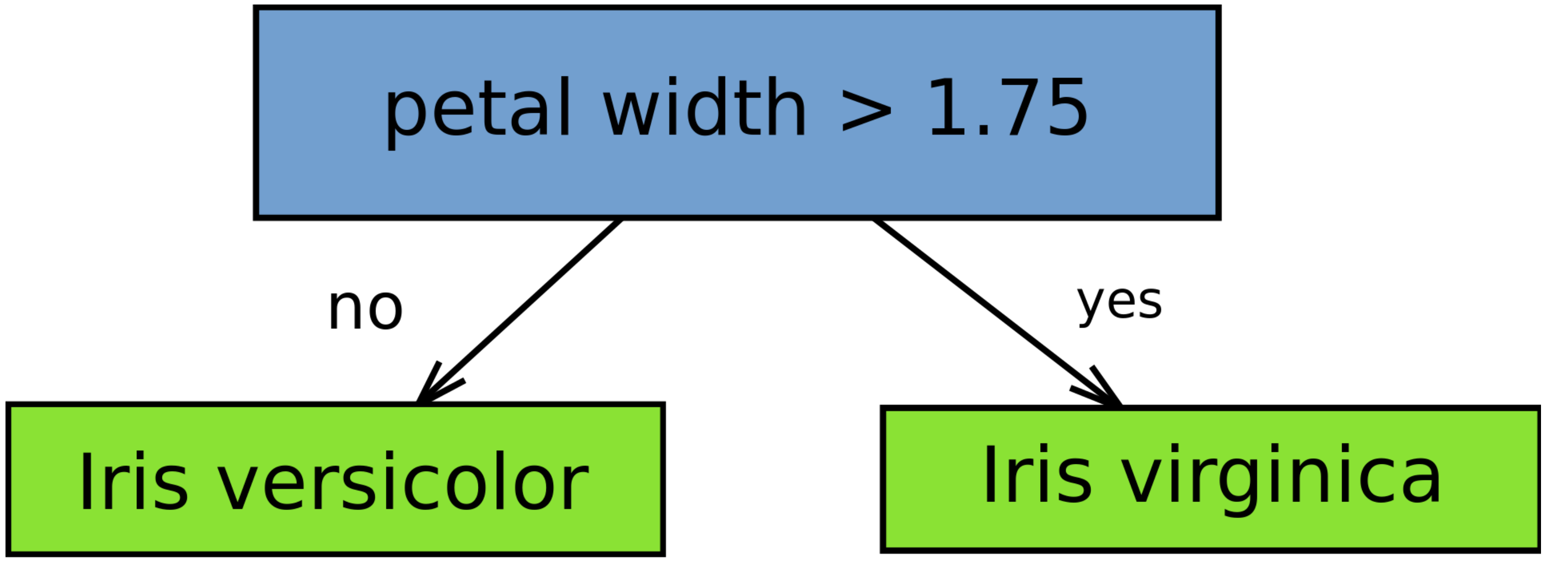
Categories of Ensemble Tree Models

- **Boosted Trees** Incrementally training each new instance to emphasize the training instances previously mis-modeled.
  - **ADABOOST (Adaptive Boosting)**
  - **XGBOOST (Gradient Boosting)**
- **Bagged Trees (Bootstrap Aggregating):** Create many different trees by repeatedly resampling with replacement.
  - **RANDOM FOREST** (one of many)

For each there are many other related models

Weak classifier

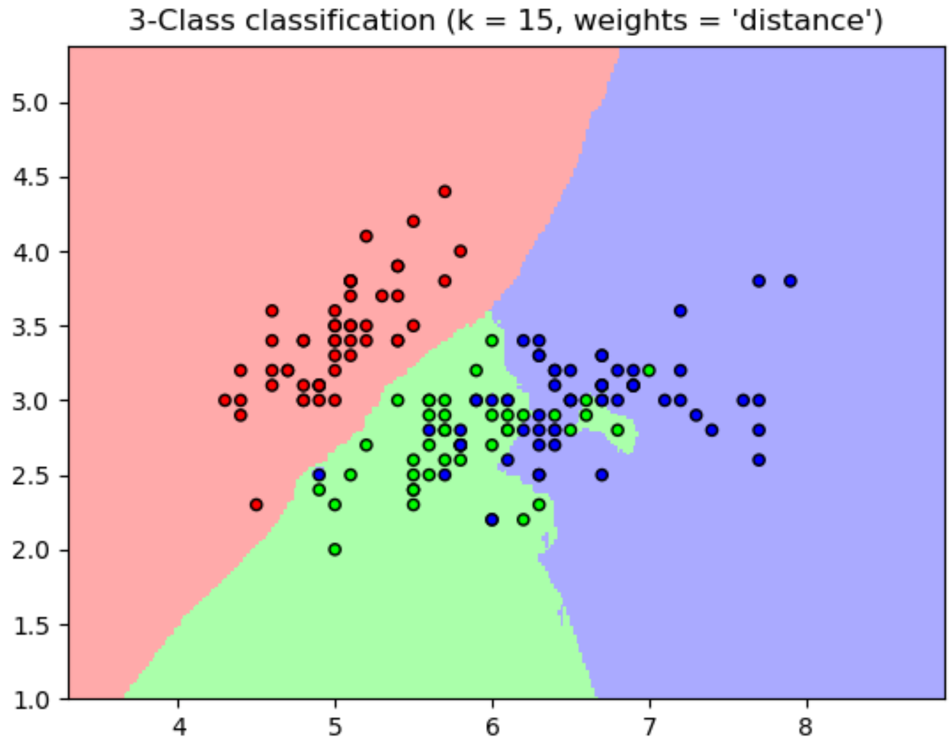
- Classifier that performs poorly, but better than random guessing
- Example: decision stump



k Nearest Neighbors (kNN)

Nearest Neighbor Classification

- Imagine data projected in a n-dimensional space, where n is the number of features
- Classification can be based on K neighbors or density



k-NN variations

- Best choice of *k* depends upon the data
  - Hyperparameter optimization
- Skewed class distribution causes issues for majority voting
  - Weight the classification, accounting for distance from the distance point to *k* nearest neighbors

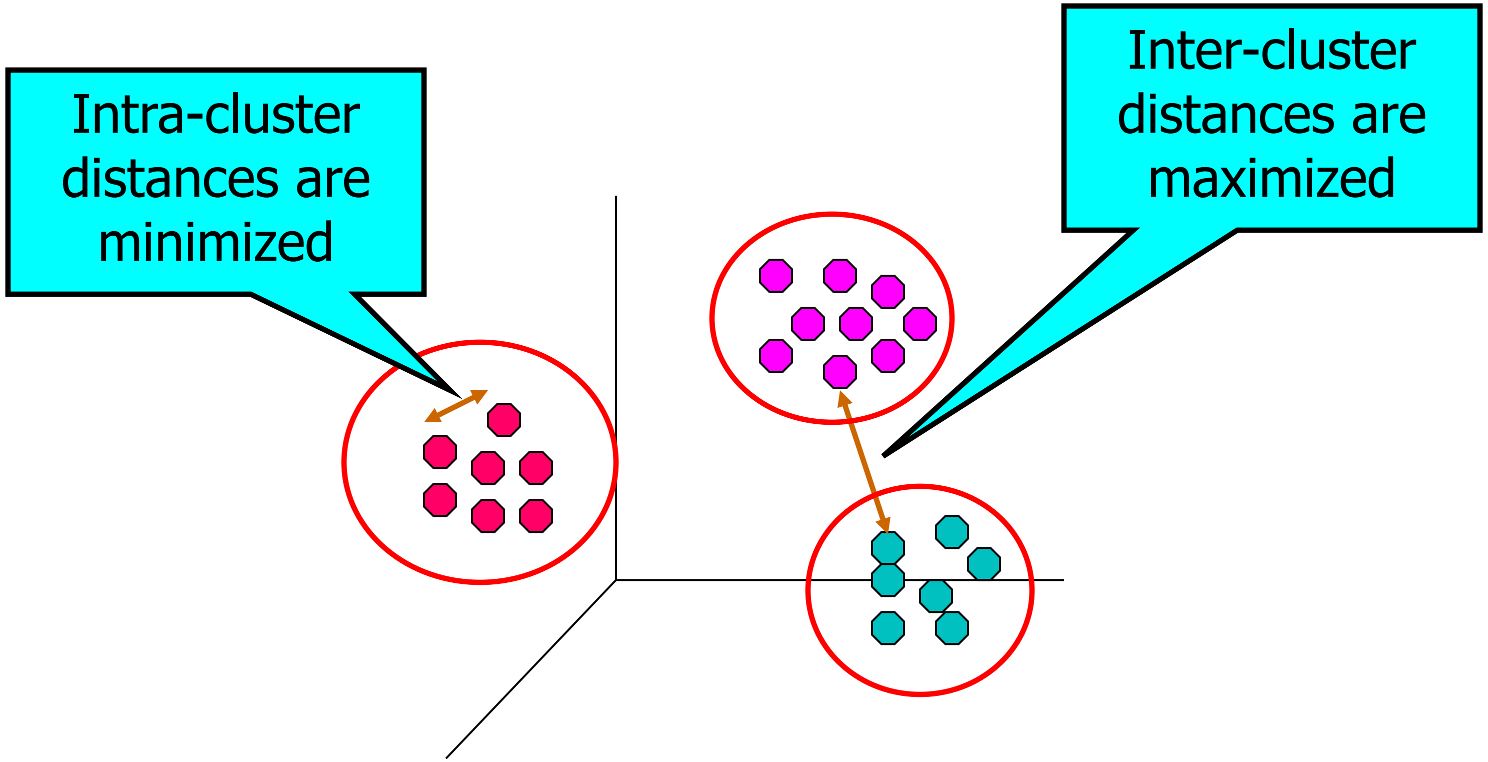
k-NN distances

- Distance metrics: Euclidean, Jaccard coefficient (binary vectors), Hamming distance, …
- May need feature engineering and dimensionality reduction to make Euclidean distance more useful
- Naïve KNN computes distances from the test example to all stored examples
    - [Nearest neighbor search algorithms]

Clustering

What is Cluster Analysis?

- Given a set of objects, place them in groups such that the objects in a group are similar (or related) to one another and different from (or unrelated to) the objects in other groups



Notion of a Cluster can be Ambiguous

How many clusters?

Six Clusters

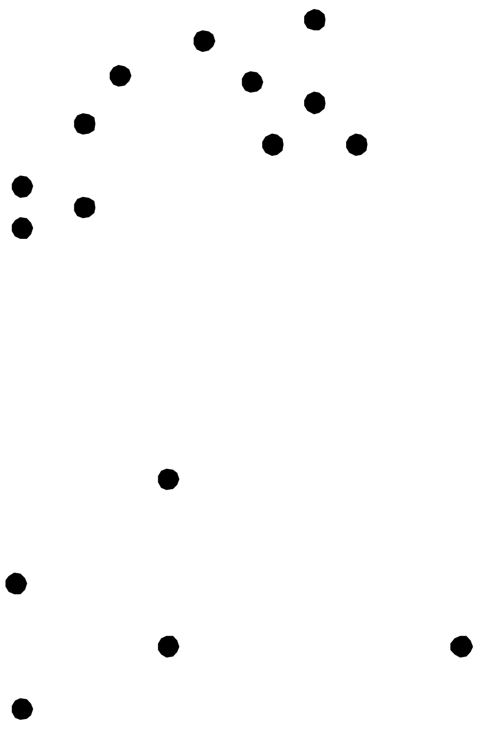
Two Clusters

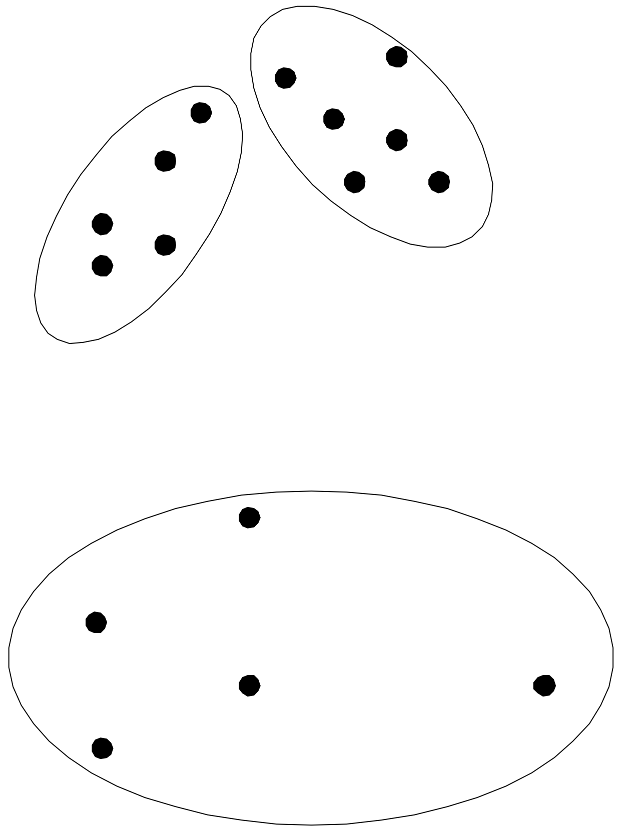
Four Clusters

Types of Clusterings

- A clustering is a set of clusters
- Important distinction between hierarchical and partitional sets of clusters
  - Partitional Clustering
    - A division of data objects into non-overlapping subsets (clusters)
  - Hierarchical clustering
    - A set of nested clusters organized as a hierarchical tree

Partitional Clustering

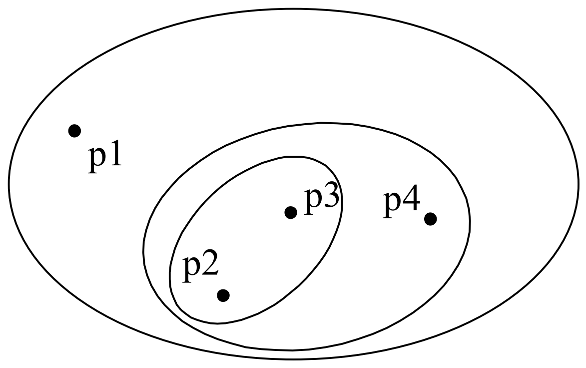


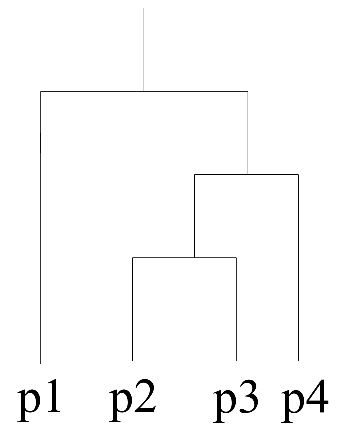


Original Points

A Partitional Clustering

Hierarchical Clustering





Characteristics of the Input Data Are Important

- Type of proximity or density measure
  - Central to clustering
  - Depends on data and application

- Data characteristics that affect proximity and/or density are
  - Dimensionality
    - Sparseness
  - Attribute type
  - Special relationships in the data
    - For example, autocorrelation
  - Distribution of the data

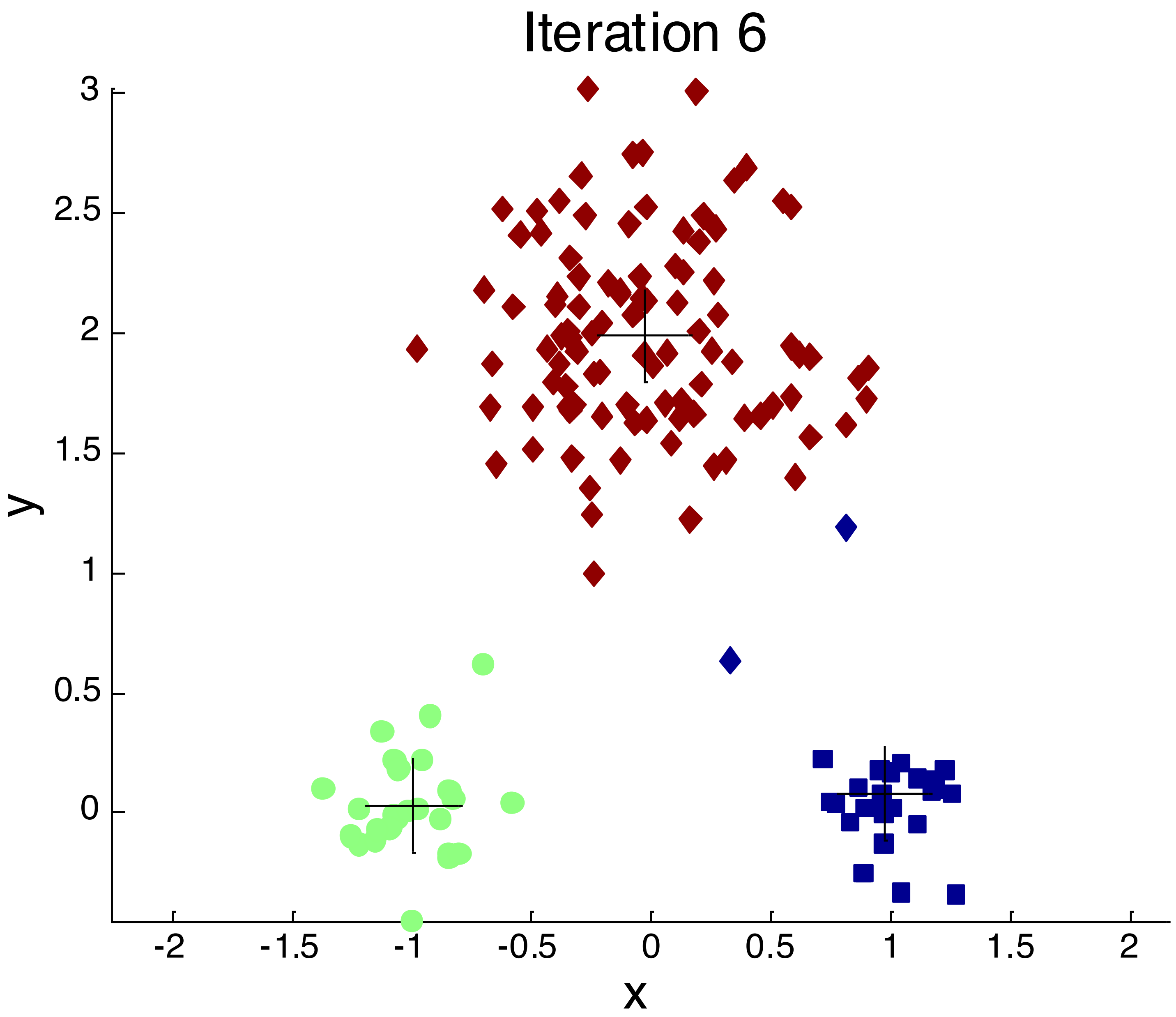
- Noise and Outliers
  - Often interfere with the operation of the clustering algorithm
- Clusters of differing sizes, densities, and shapes

K-means Clustering

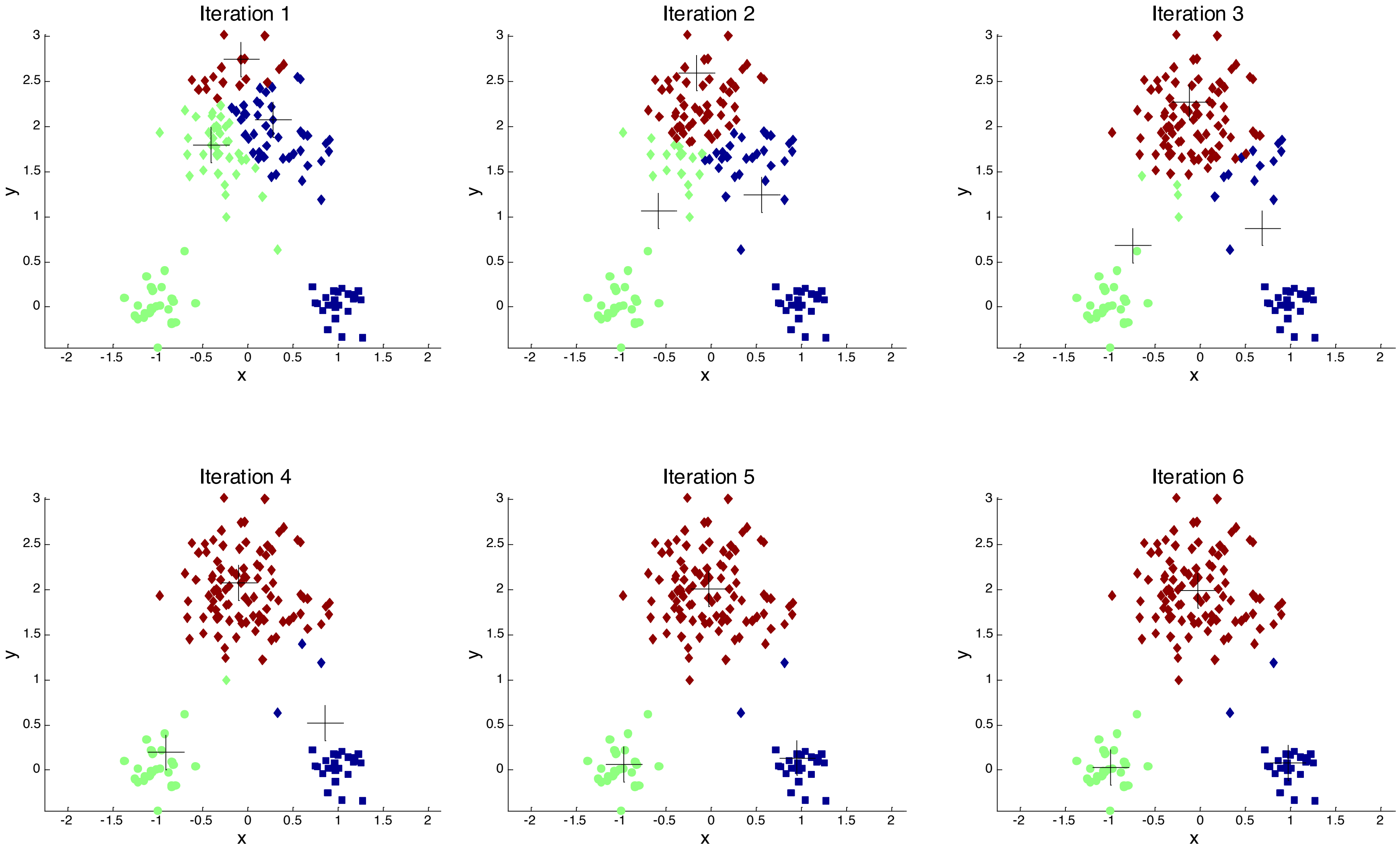
- Partitional clustering approach
- Number of clusters, K, must be specified
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- The basic algorithm is very simple

```
Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```

Example of K-means Clustering



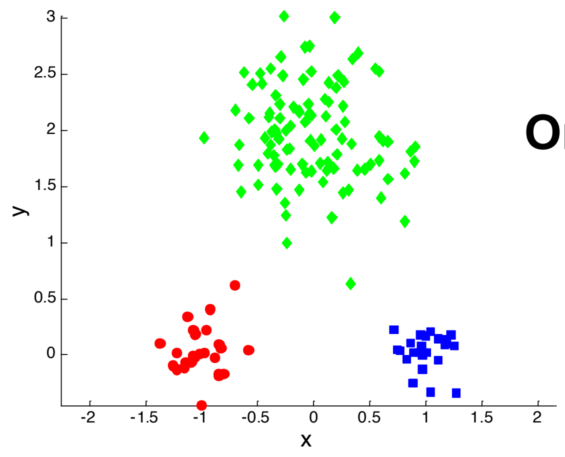
Example of K-means Clustering



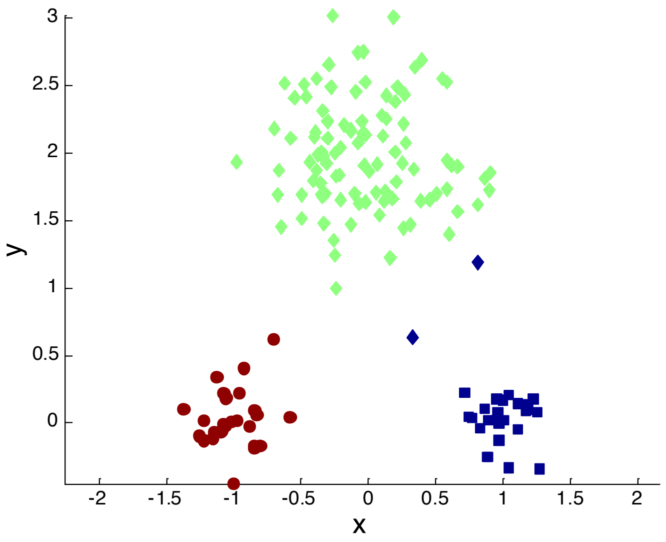
K-means Clustering – Details

- Simple iterative algorithm.
  - Choose initial centroids;
  - repeat {assign each point to a nearest centroid; re-compute cluster centroids}
  - until centroids stop changing.
- Initial centroids are often chosen randomly.
  - Clusters produced can vary from one run to another
- The centroid is (typically) the mean of the points in the cluster, but other definitions are possible.
- K-means will converge for common proximity measures with appropriately defined centroid
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to 'Until relatively few points change clusters'

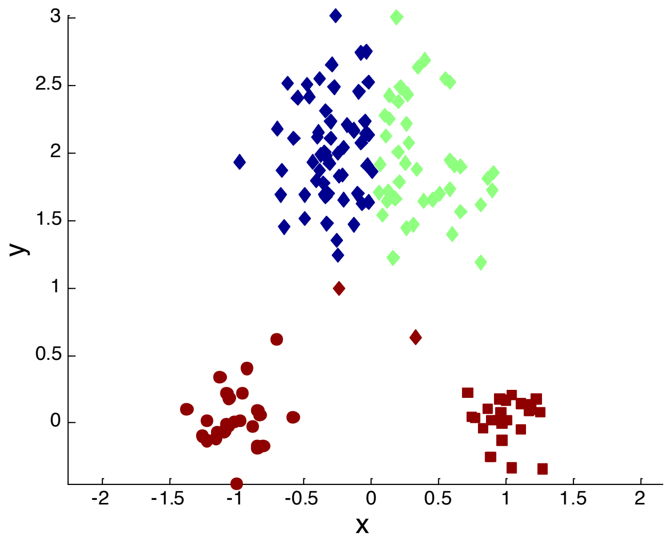
Two different K-means Clusterings



Original Points

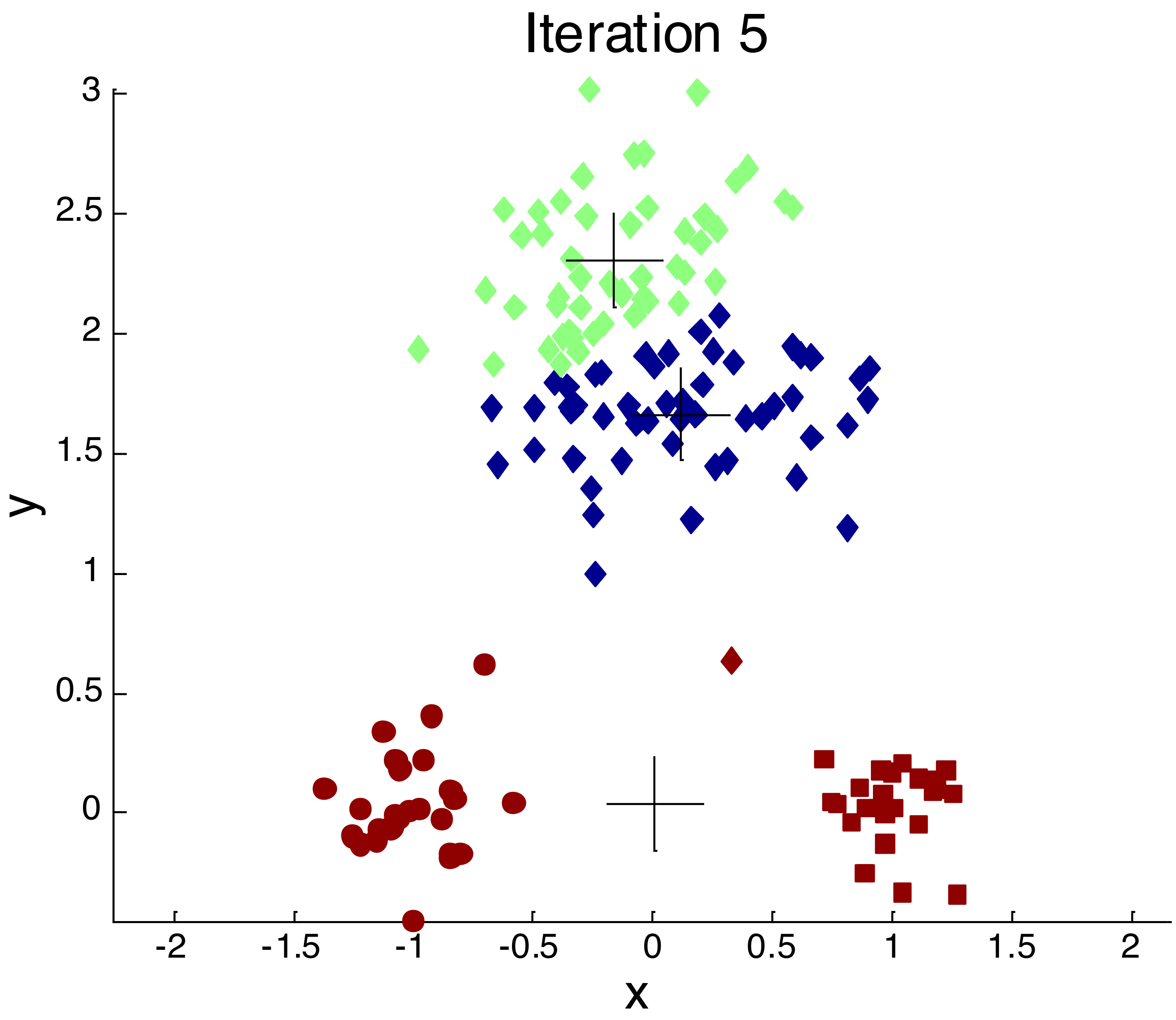


Optimal Clustering

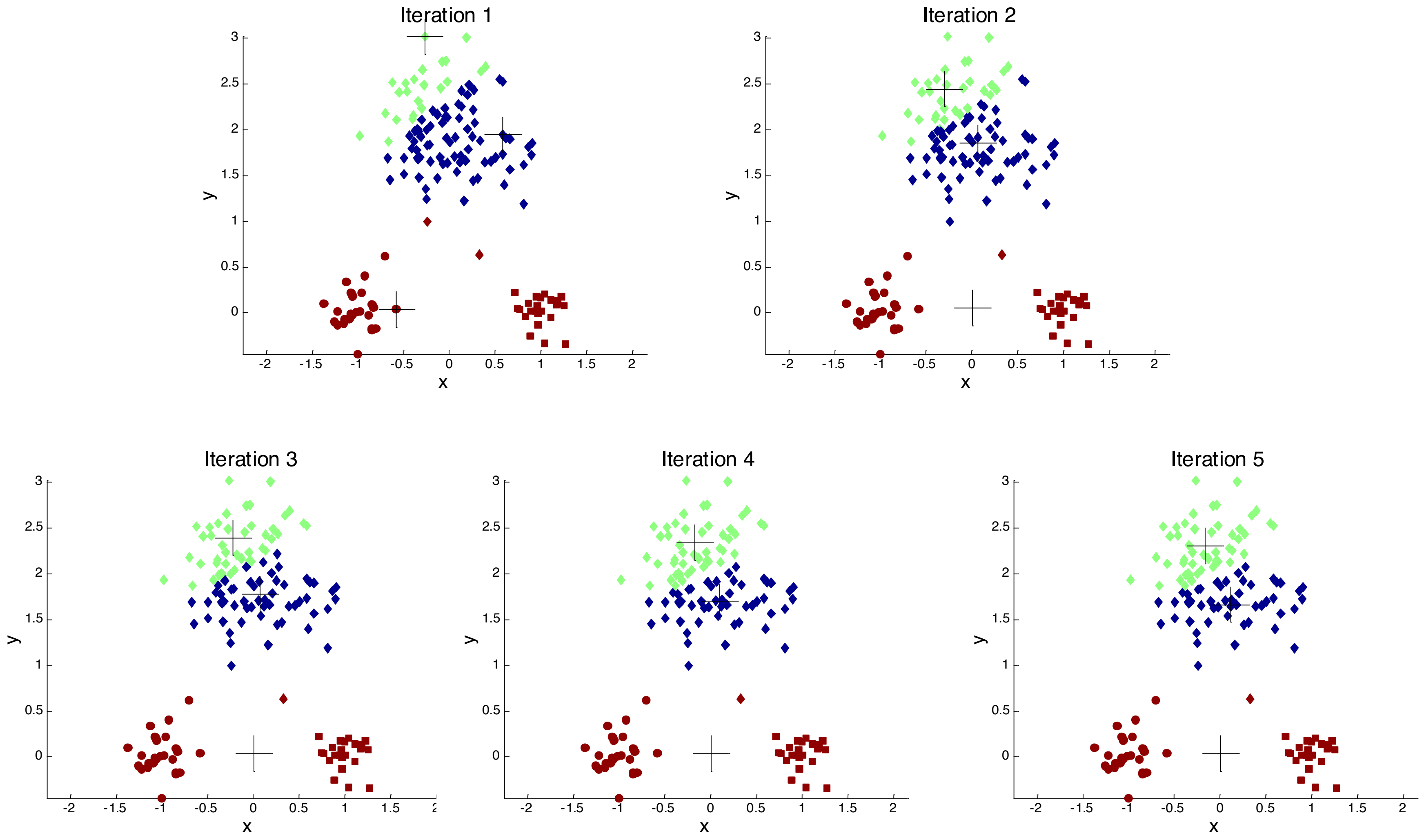


Sub-optimal Clustering

Importance of Choosing Initial Centroids …



Importance of Choosing Initial Centroids …



K-means Objective Function

- A common objective function (used with Euclidean distance measure) is Sum of Squared Error (SSE)
  – For each point, the error is the distance to the nearest cluster center
  – To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

  – x is a data point in cluster C_i and m_i is the centroid (mean) for cluster C_i
  – SSE improves in each iteration of K-means until it reaches a local or global minima.
  – This is called the model’s inertia: the mean squared distance between each instance and its closest centroid.

Unsupervised Measures: Cohesion and Separation

- Cluster Cohesion: Measures how closely related are objects in a cluster
  - Example: SSE
- Cluster Separation: Measure how distinct or well-separated a cluster is from other clusters
- Example: Squared Error
  - Cohesion is measured by the within cluster sum of squares (SSE)

$$SSE = \sum_i \sum_{x \in C_i} (x - m_i)^2$$

  - Separation is measured by the between cluster sum of squares

$$SSB = \sum_i |C_i| (m - m_i)^2$$

Where $|C_i|$ is the size of cluster *i*, *m* is the global average point, m_i is the centroid of cluster *i*, and *x* is a point in the cluster

Determining the Correct Number of Clusters

- SSE is good for comparing two clusterings or two clusters
- SSE can also be used to estimate the number of clusters