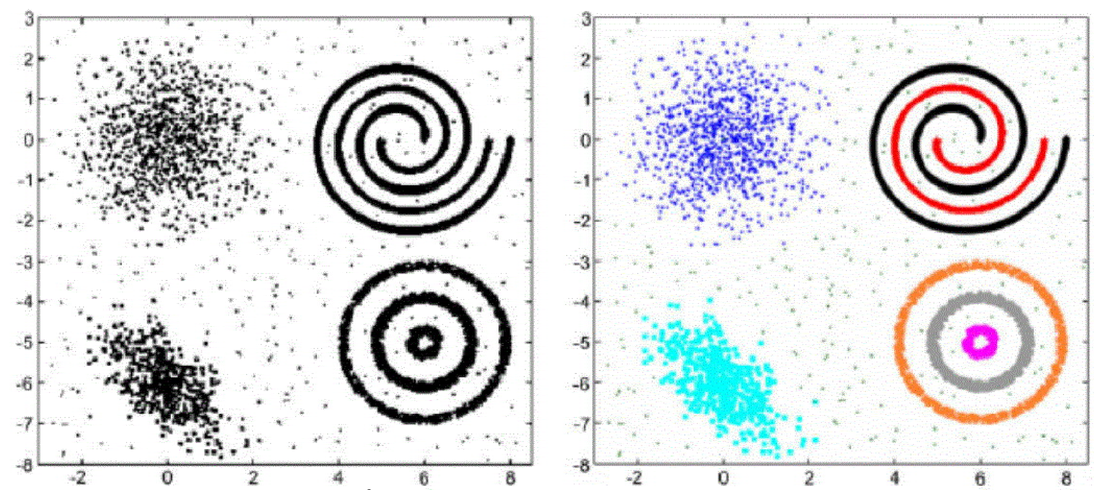
Clustering = Unsupervised Learning

↳ unlabeled data

★ Why Data Clustering?



(a) Input data in 2D (b) Desired clustering

From: A.K. Jain, "Data clustering: 50 years beyond K-means," Pattern Recognition Letters, vol. 31, pp. 651–666, 2010.

- Underlying structure: to gain insight into data, generate hypotheses, detect anomalies, and identify salient features.
- Natural classification: to identify the degree of similarity among forms or organisms (phylogenetic relationship).
- Compression: as a method for organizing the data and summarizing it through cluster prototypes.

☆ The K-Means Algorithm

- Most popular
- Simplest
- Still being used after all these years and after hundreds of clustering algorithms were proposed.

- **Set up**

  Let $X = \{x_1, \cdots, x_n\}$, $x_j \in \mathbb{R}^d$, $1 \le j \le n$

  Suppose we want to cluster (group) them into a set of K clusters, $C = \{c_1, \cdots, c_K\}$, $1 < K \ll n$. Each c_j contains some data vectors in X.

- K-means algorithm finds a partition s.t. the squared error between the empirical mean of a cluster and the points in the cluster is minimized. More precisely, let $\mu_k :=$ the mean of cluster c_k and define

$$J(c_k) := \sum_{x_j \in c_k} \| x_j - \mu_k \|^2$$

and

$$J(C) := \sum_{k=1}^{K} J(c_k)$$

K-means tries to find a partition (clustering) C s.t. $J(C) \to \min$.

- This minimization problem is known to be NP-hard (non-deterministic polynomial-time hard, i.e., at least as hard as any NP problem, e.g. might require the exhaustive search or trials)
- Hence, the result of the K-means may be just a local minimum, not necessarily the global minimum of $J(C)$.
- $J(C)$ always decreases if K increases. In fact, if $K = n$, then $J(C) = 0$! So, we should fix K as $1 < K \ll n$.

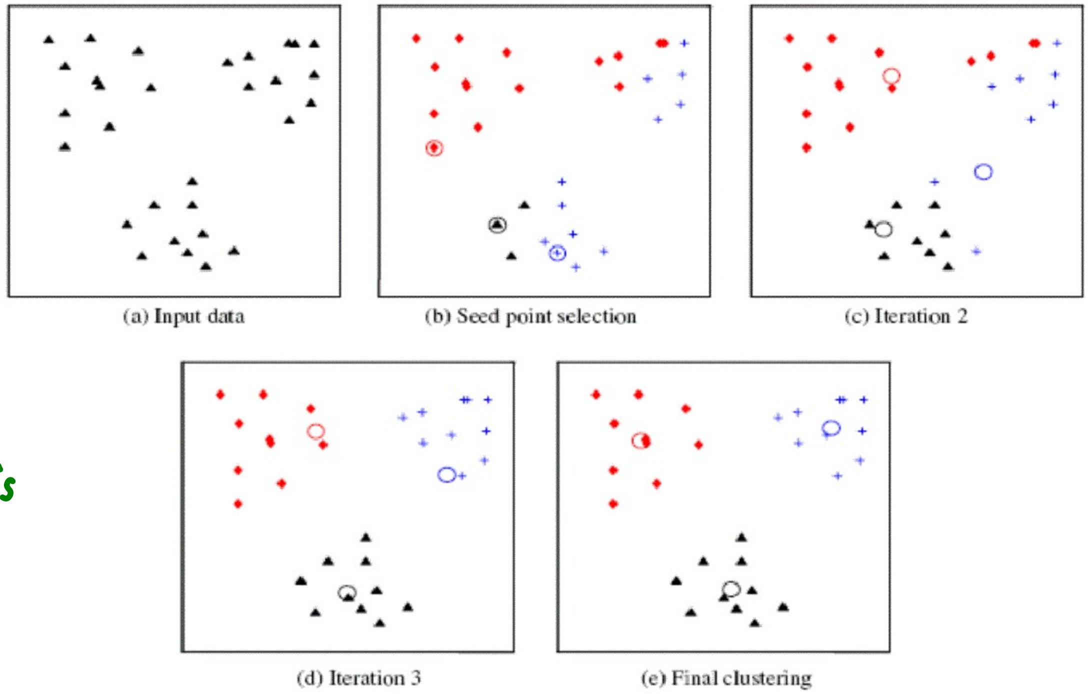
Here are the main steps of the K-means algorithm.

<u>Step 1</u>: Select an initial partition with K clusters; repeat Steps 2 & 3 until cluster membership stabilizes.

<u>Step 2</u>: Generate a new partition by assigning each point (vector) to its closest cluster center.

<u>Step 3</u>: Compute new cluster centers.

K-means in action!

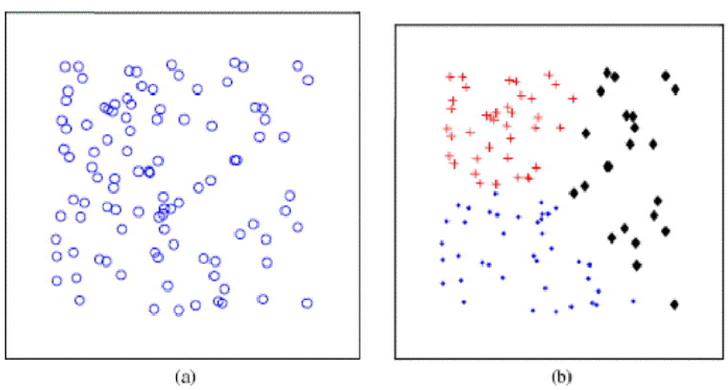


(a) Input data (b) Seed point selection (c) Iteration 2

(d) Iteration 3 (e) Final clustering

Both from A.K.Jain's paper

- Problems of the K-means Alg.
  - How to preset K?



(a) (b)

← K = 3

  - The computed clusters are just local minimum of J(C).

    ⇒ one option would be to run the K-means algorithm (with fixed K) several times, and pick the best one.

Two MATLAB Demonstrations

(1) Breast Cancer Dataset from UC Irvine Machine Learning Repository

$d = 9$, $n = 683$ (after removing patients of some missing measurements)

↓

measurements based on cytological images of breast cells including: clump thickness; uniformity of cell size; uniformity of cell shape; etc.

Out of 683 subjects, 444: benign
239: malignant

Suppose we do not know these diagnostic results, and use the K-means alg. with $K = 2$ on this data matrix $X \in \mathbb{R}^{9 \times 683}$. Can we classify benign & malignant cells correctly?

(2) Using the K-means alg. to binarize a face image.

Here $d = 1$ (pixel value), $n = 128^2$ (# of pixels)

also $K = 2$.

We can also use $K = 3, 4, 5, \cdots$ to see how the image looks like after replacing the true pixel values by the cluster center values.