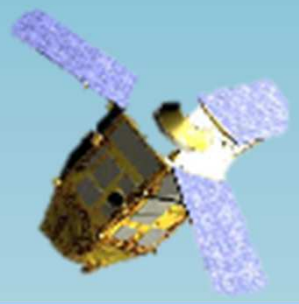
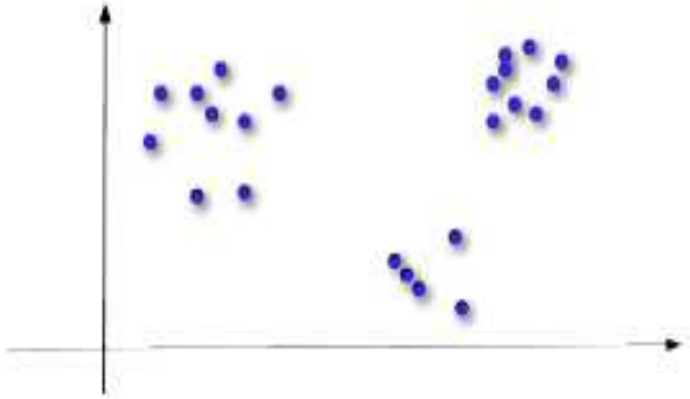
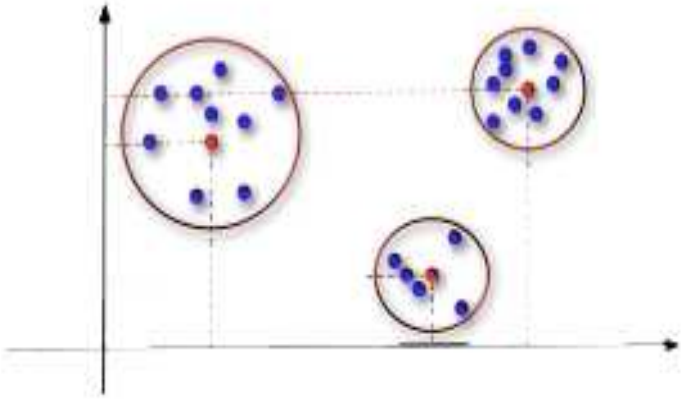
# Clustering



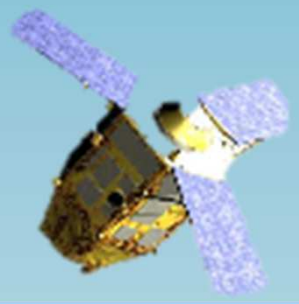
- Clustering aims to **find classes in data**





- Different algorithms exist :
  - K-Means
  - Mean-Shift
  - Expectation-Maximization using Gaussian Mixture Models
  - Etc …

# Clustering : K-Means

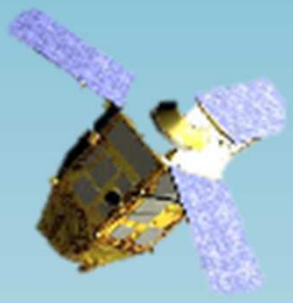


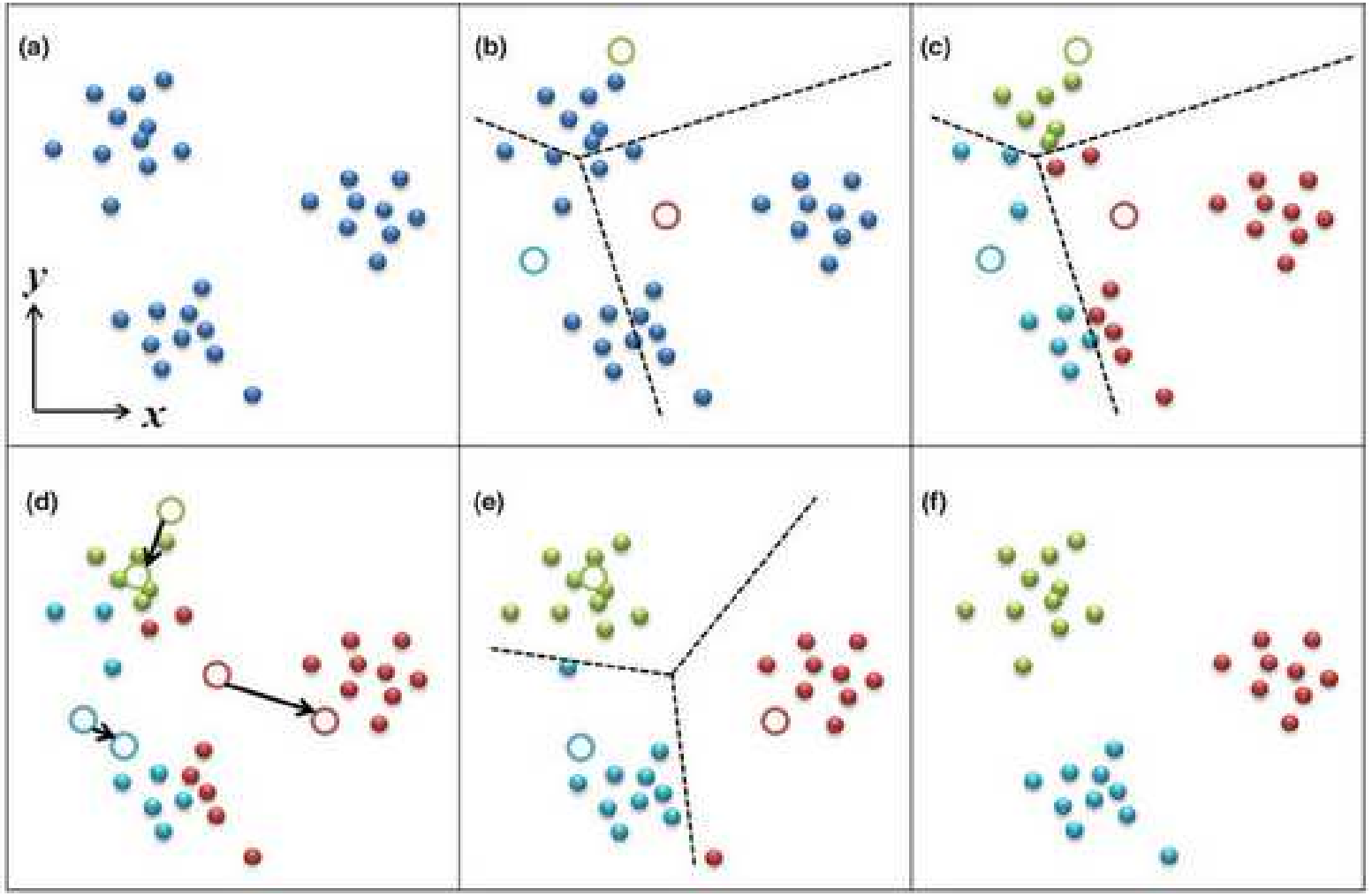
- K-Means is the **most famous** Clustering algorithm
- K-Means is composed of the following steps :
  - Select a number of classes and randomly initialize their center points
  - Classify each data by computing the distance between that data and each group center points
  - Recompute group center points by taking the mean of data in each class
  - Repeat the steps 2 and 3 for n iterations or until group center points do not change much

→ **Drawbacks :**

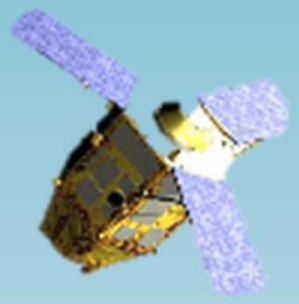
- ✓ Choose the number of classes
- ✓ Start with a random choice so can yield to different clustering results

# Clustering :
# K-Means example





# Clustering : Mean-Shift



- Mean-Shift is composed of the following steps :
  - Select a random set of points and a window size *(windows will be centered on each point)*
  - Iteratively:
    - Compute center points by taking the mean of data in each window
    - Move windows on these new center points
  - Once windows do not move anymore, remove overlap windows by keeping the one with the most data
  - Classify each data by computing the distance between that data and each center points

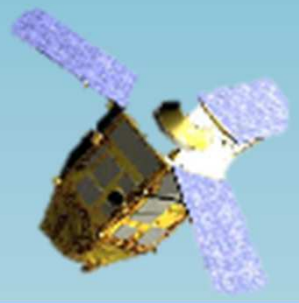
**→Advantage :**

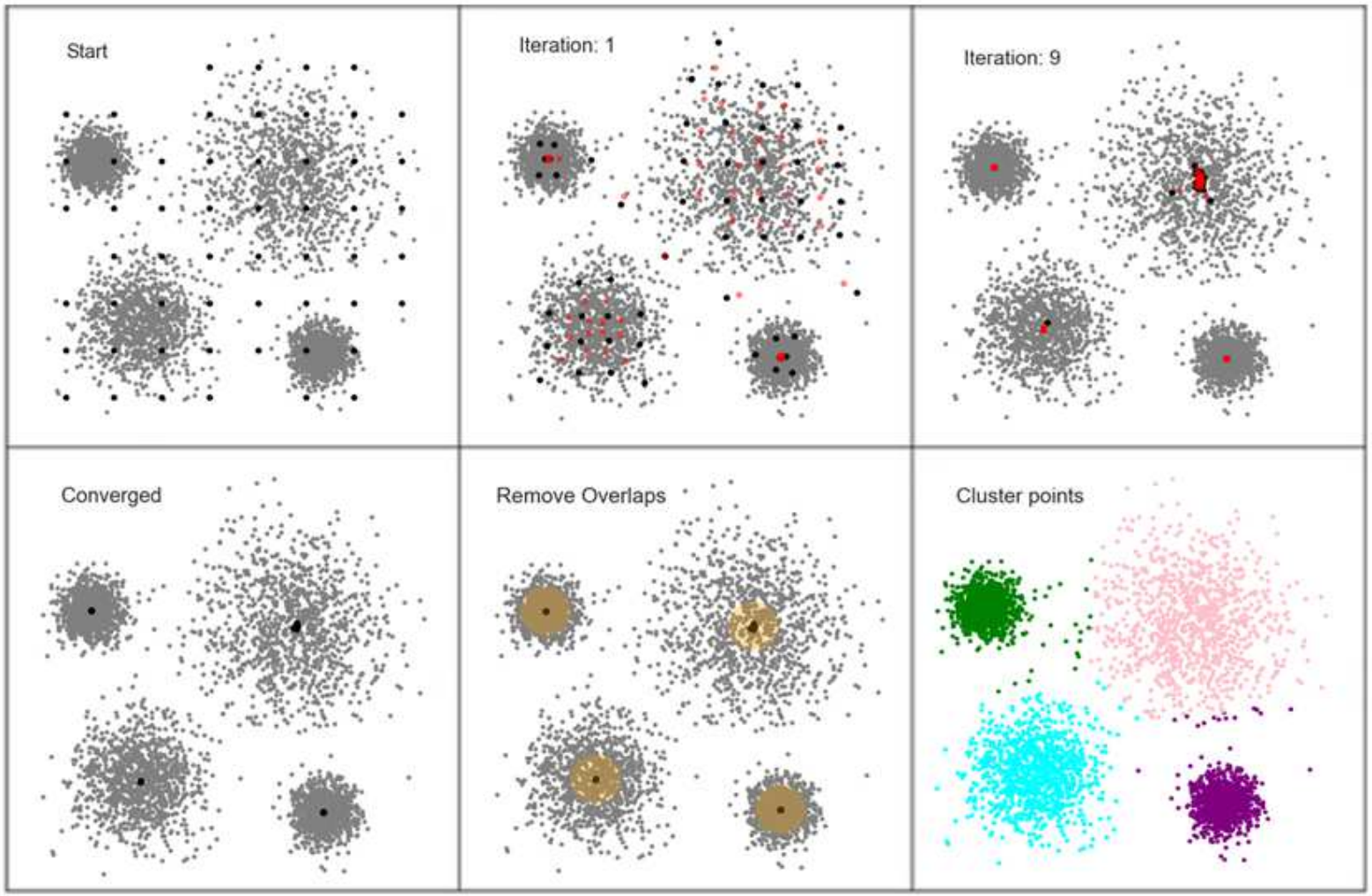
- ✓ Do not choose the number of classes

**→ Drawbacks :**

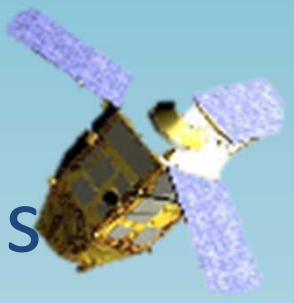
- ✓ Choose the window size
- ✓ Start with a random choice so can yield to different clustering results

# Clustering :
# Mean-Shift example

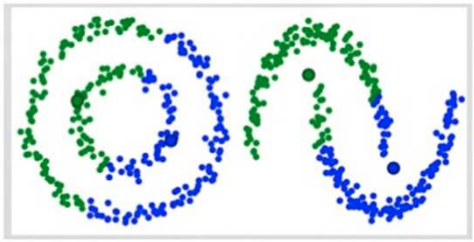




# Clustering :
## Expectation-Maximization using GMMs



- Gaussian Mixture Models (GMMs) assume that data points are Gaussian distributed
  - → This is less restrictive assumption than saying data are circular



- Expectation-Maximization is composed of the same steps than K-Means :
  - Select a number of classes and randomly initialize **the Gaussian distribution parameters**
  - Classify each data by computing **the probability that data belongs to each cluster**
  - Recompute **the Gaussian distribution parameters** for each cluster
  - Repeat the steps 2 and 3 for n iterations or until **Gaussian distribution parameters** do not change much