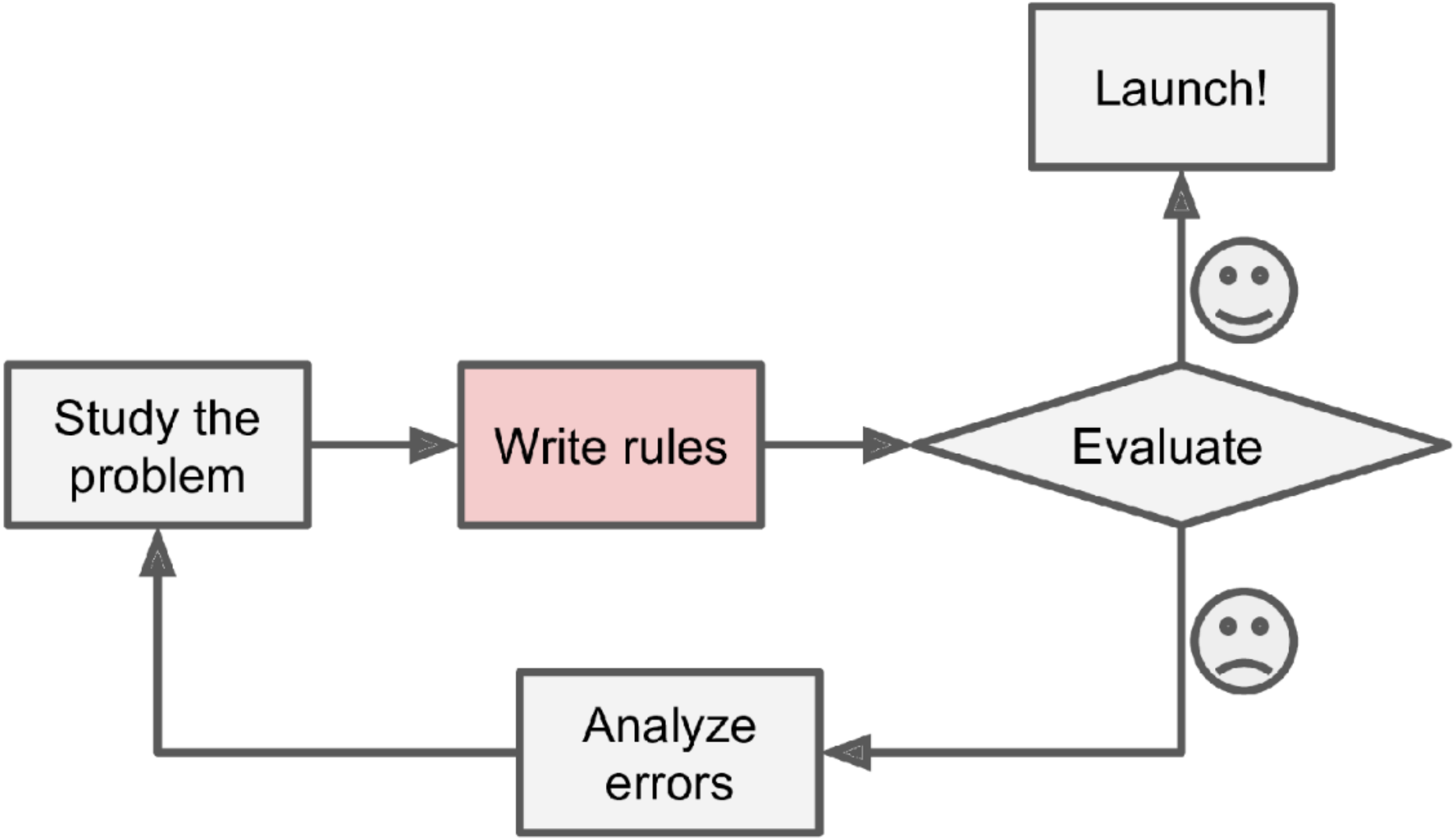
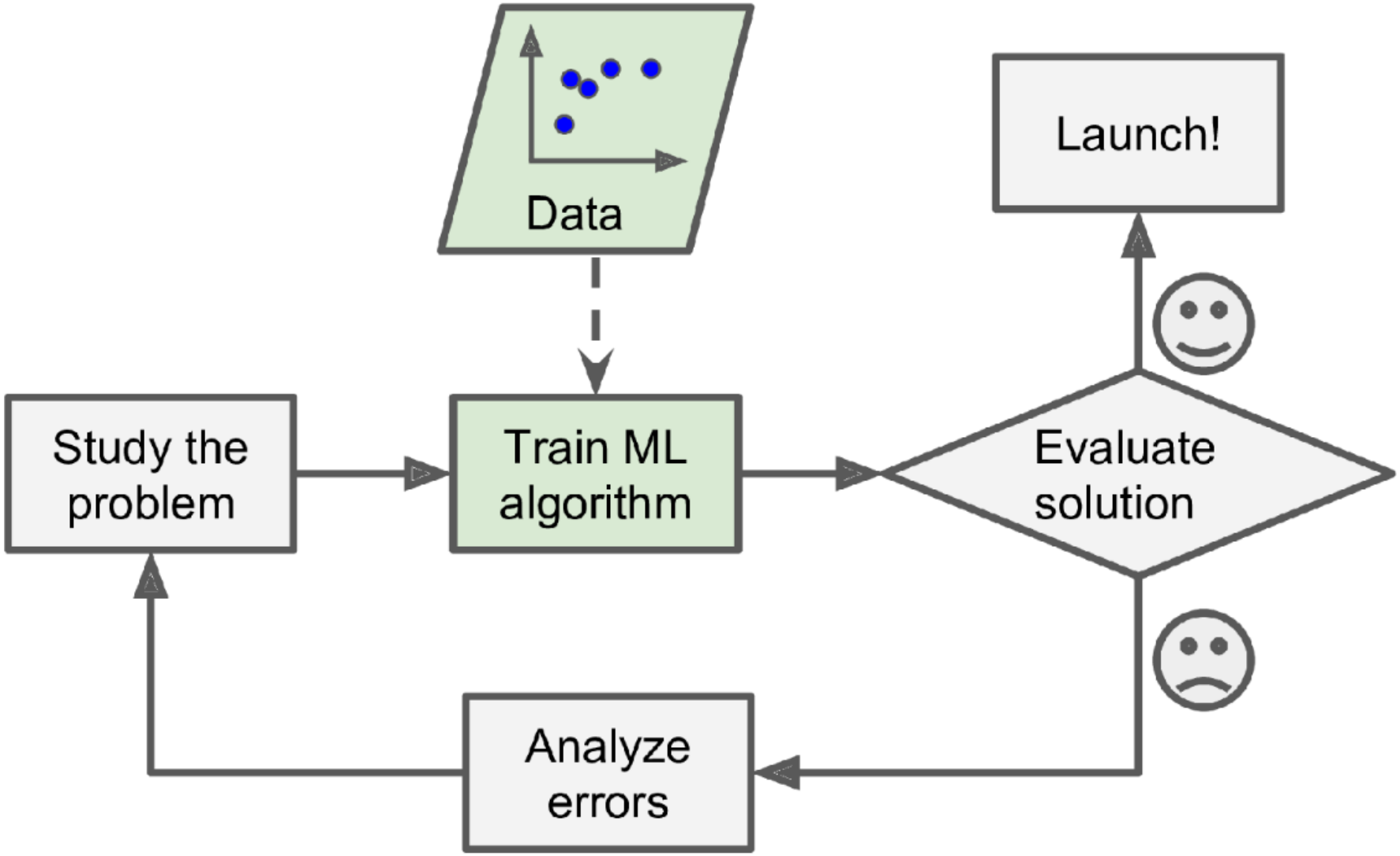
Machine Learning

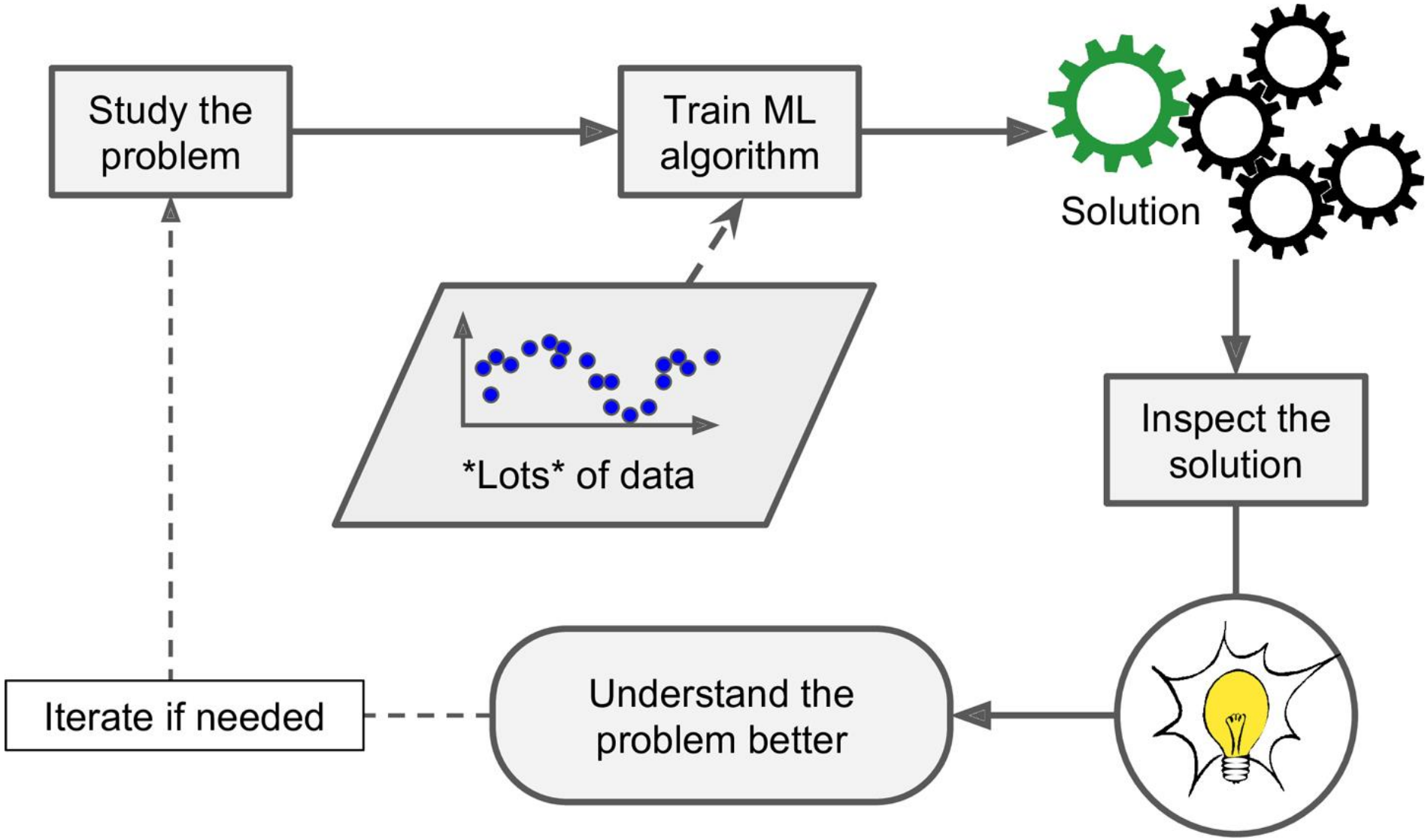
Traditional Approach



Machine Learning Approach



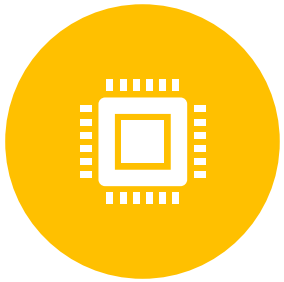
Machine Learning can help humans learn



Machine Learning advantages

Problems for which existing solutions require a lot of fine-tuning or long lists of rules: one Machine Learning algorithm can often simplify code and perform better than the traditional approach.

Complex problems for which using a traditional approach yields no good solution: the best Machine Learning techniques can perhaps find a solution.



Fluctuating environments: a Machine Learning system can adapt to new data.

Getting insights about complex problems and large amounts of data.

Examples of Machine Learning Applications

1. Detecting credit card fraud. This is anomaly detection.

2. Predicting whether a customer will cancel their hotel booking or not based on their previous booking history. This is classification.

3. Predicting the resale value of the used phones based on the phone attributes and how long they have been used. This is regression.

4. Automatically classifying news articles. This is natural language processing (NLP), and more specifically text classification, which can be tackled using recurrent neural networks (RNNs), CNNs, or Transformers.

5. Forecasting your company's weekly revenue or budget for next year. This is time-series analysis.

Types of Machine Learning

Whether or not they are trained with human supervision (supervised, unsupervised, semi supervised, and Reinforcement Learning)

Whether or not they can learn incrementally on the fly (online versus batch learning)

Whether they work by simply comparing new data points to known data points, or instead by detecting patterns in the training data and building a predictive model, much like scientists do (instance-based versus model-based learning)

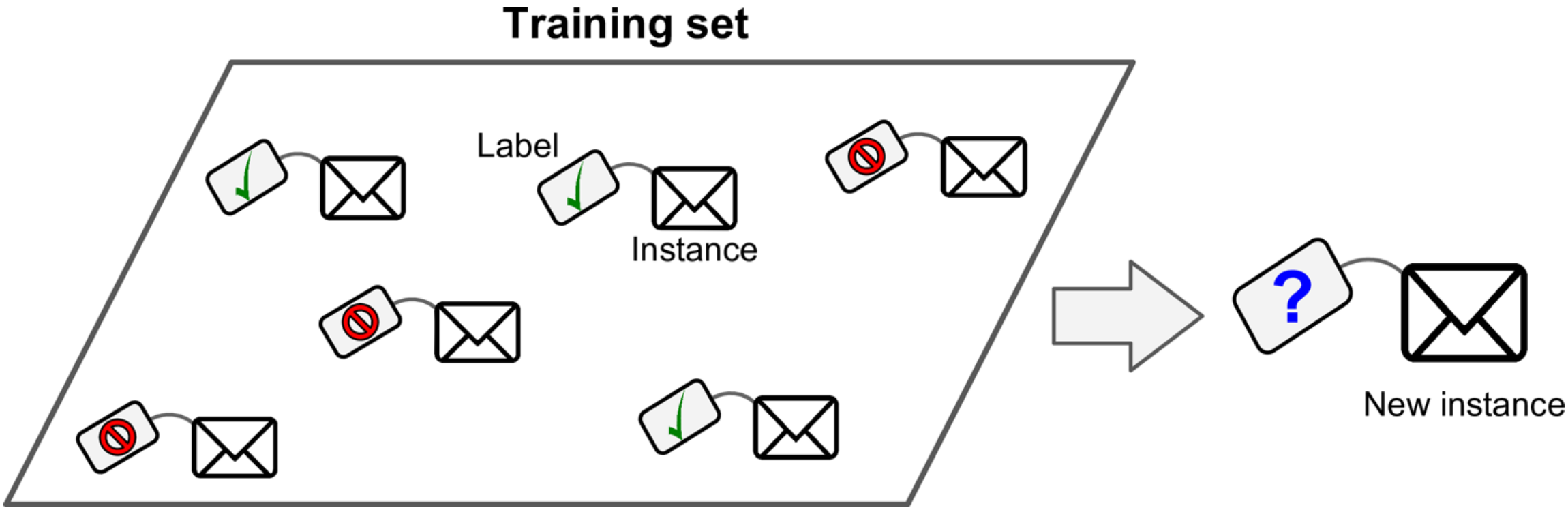
Types of Machine Learning

Whether or not they are trained with human supervision (**supervised**, **unsupervised**, semi supervised, and Reinforcement Learning)

Whether or not they can learn incrementally on the fly (online versus batch learning)

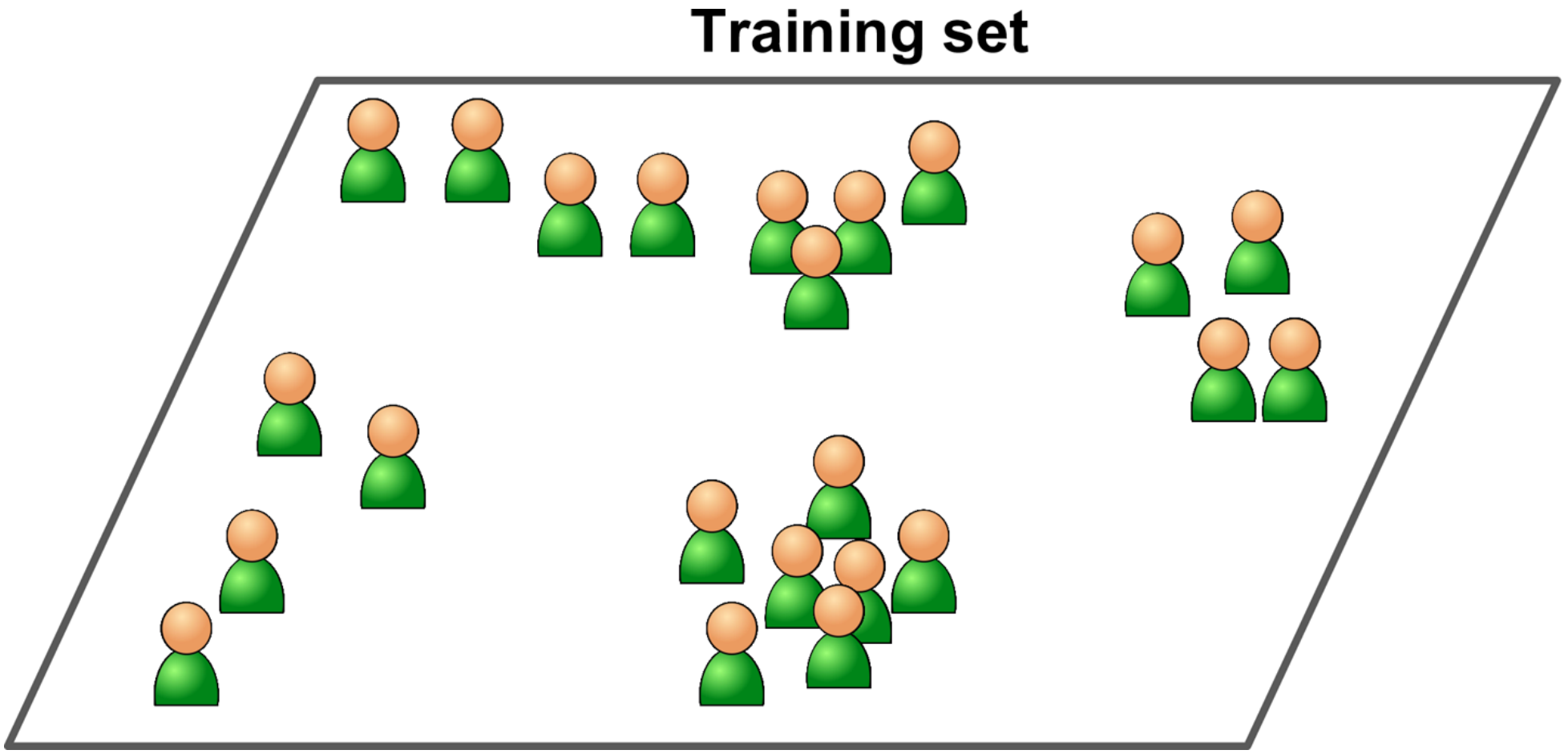
Whether they work by simply comparing new data points to known data points, or instead by detecting patterns in the training data and building a predictive model, much like scientists do (instance-based versus model-based learning)

Supervised Learning



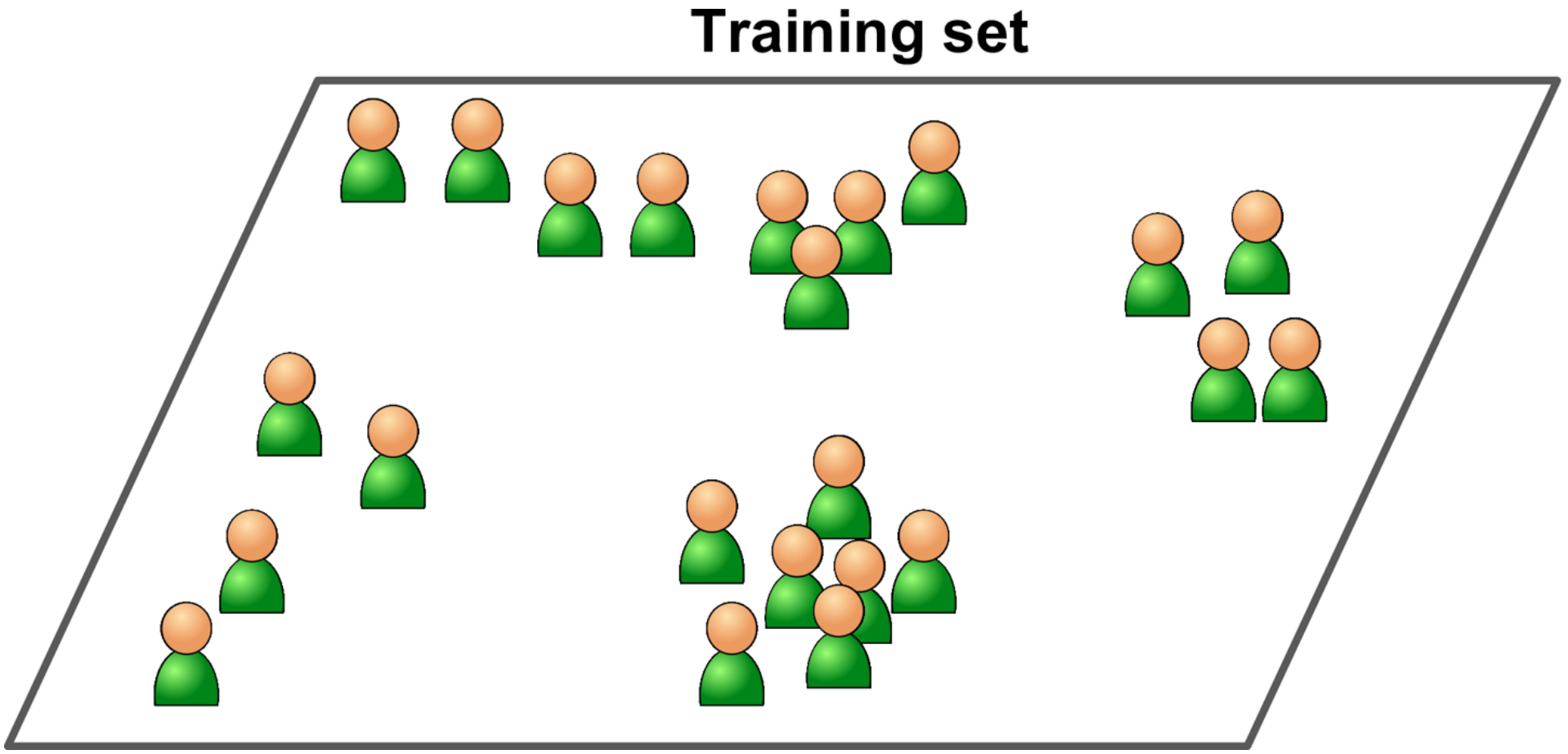
- Each instance of the training dataset has a label (answer) attached to it.
- A label is the thing that we are trying to learn using machine learning. E.g. if we are interested in training an ML model to predict whether an email is a spam or not, then the span/no-spam information is the label.

Unsupervised Learning



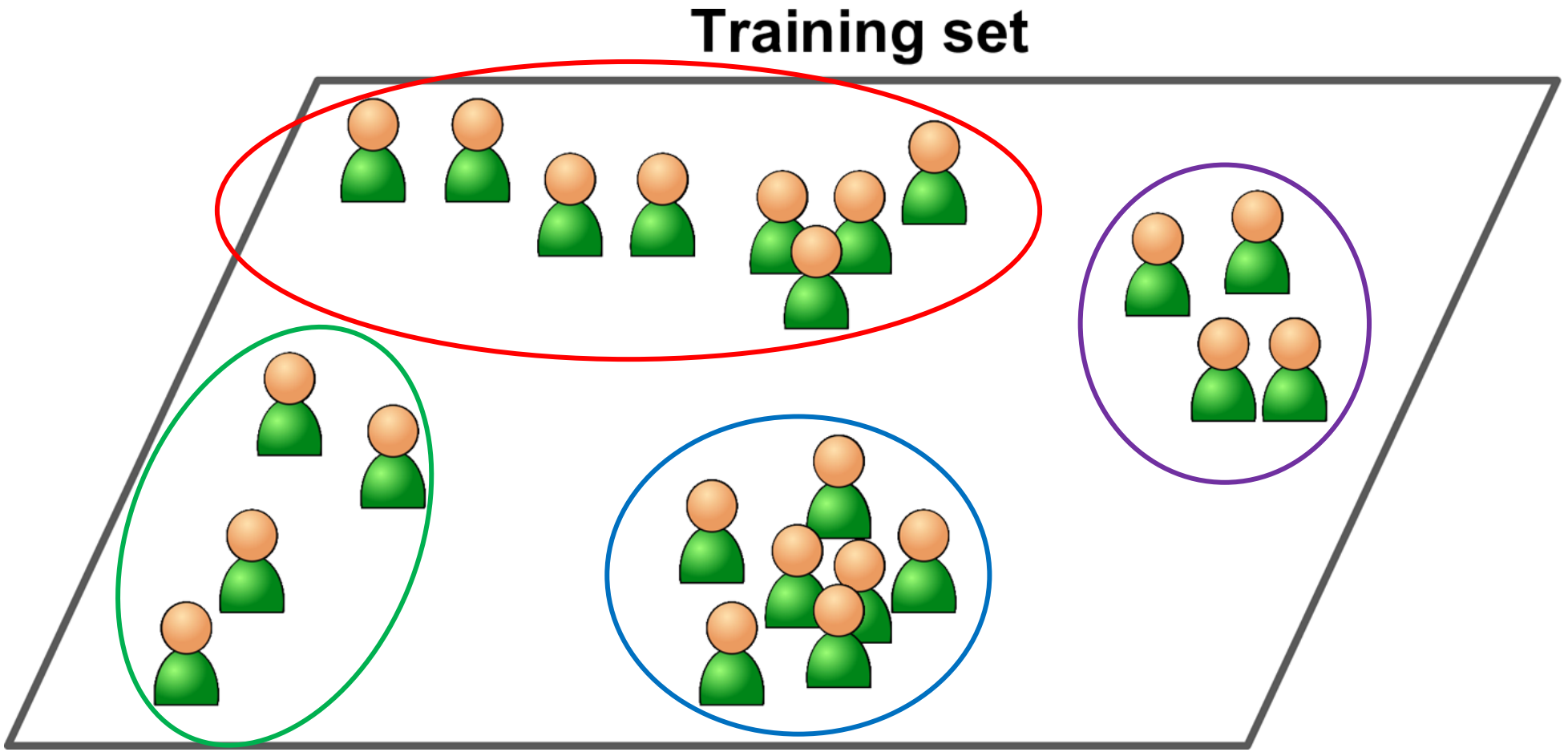
- The instances of the training dataset don’t have labels (answer) attached to them.

Unsupervised Learning



- The instances of the training dataset don’t have labels (answer) attached to them.
- So, what would an ML model learn from such an unlabeled dataset?

Unsupervised Learning



- The instances of the training dataset don't have labels (answer) attached to them.

- So, what would an ML model learn from such an unlabeled dataset?
  - Create clusters/groups of similar instances! -> Clustering Analysis

Clustering Analysis

- Cluster: A collection of data objects
  - similar (or related) to one another within the same group
  - dissimilar (or unrelated) to the objects in other groups
- Cluster analysis (or *clustering*, *data segmentation, …*)
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Unsupervised learning: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

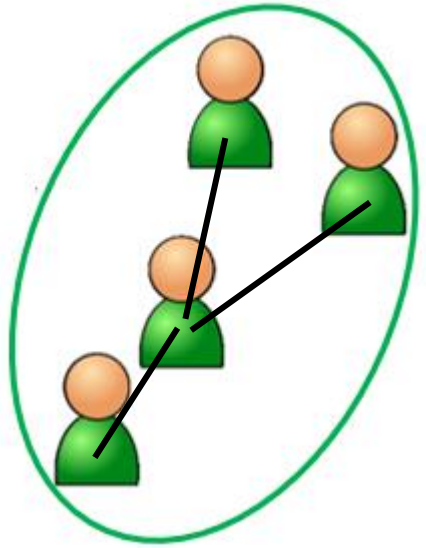
Clustering Analysis Applications

- Biology: taxonomy of living things: kingdom, phylum, class, order, family, genus and species
- Information retrieval: document clustering
- Land use: Identification of areas of similar land use in an earth observation database
- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Finance: Grouping securities in portfolios
- Climate: understanding earth climate, find patterns of atmospheric and ocean
- Economic Science: market research, grouping firms for structural analysis of economy

Quality: What is Good Clustering?

- A good clustering method will produce high quality clusters
  - high intra-class similarity: cohesive within clusters
  - low inter-class similarity: distinctive between clusters
- The quality of a clustering method depends on
  - the similarity measure used by the method
  - its implementation, and
  - Its ability to discover some or all of the hidden patterns

Similarity measure
(Measuring distance between records)



- **Euclidean Distance** (most popular)

	x1	x2	x3
i	43	23.5	32
j	34	35	35.4

$$d_{ij} = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ip} - x_{jp})^2}$$

$$= \sqrt{(43 - 34)^2 + (23.5 - 35)^2 + (32 - 35.4)^2} = 14.99$$

- **Problem**: Raw distance measures are highly influenced by scale of measurements
- **Solutions**:

Standardization

$$x_{new} = \frac{x - mean(x)}{std(x)}$$

Normalization

$$x_{new} = \frac{x - min(x)}{max(x) - \min(x)}$$

Similarity measure
(Measuring distance between records)

- For Categorical Data:

	x1	x2	x3	x4	x5
i	1	0	0	1	1
j	0	0	1	1	1

		Record j: 0	Record j: 1	
Record i	0	a	b	$a+b$
	1	c	d	$c+d$
		$a+c$	$b+d$	n

		Record *j*: 0	Record *j*: 1	
Record *i*	0	1	1	2
	1	1	2	3
		2	3	5

- Matching coefficient: $\frac{a+d}{n} = \frac{1+2}{5} = 0.6$
- Jaquard's coefficient: $\frac{d}{b+c+d} = \frac{2}{1+1+2} = 0.5$
  - Ignores zero matches. Use in cases where matching "1" is much greater evidence of similarity than matching "0" (e.g. "owns Corvette")

Similarity measure
(Measuring distance between records)

- Gower's similarity (for mixed variable types: continuous & categorical)
  - Weighted average of the distances computed for each variable.

$$s_{ij} = \frac{\sum_{m=1}^{p} w_{ijm} s_{ijm}}{\sum_{m=1}^{p} w_{ijm}},$$

where s_{ijm} is the similarity between records i and j on measurement m, and w_{ijm} is a binary weight given to the corresponding distance.

The similarity measures s_{ijm} and weights w_{ijm} are computed as follows:

1. For continuous measurements, $s_{ijm} = 1 - \frac{|x_{im} - x_{jm}|}{\max(x_m) - \min(x_m)}$ and $w_{ijm} = 1$ unless the value for measurement m is unknown for one or both of the records, in which case $w_{ijm} = 0$.
2. For binary measurements, $s_{ijm} = 1$ if $x_{im} = x_{jm} = 1$ and 0 otherwise. $w_{ijm} = 1$ unless $x_{im} = x_{jm} = 0$.
3. For nonbinary categorical measurements, $s_{ijm} = 1$ if both records are in the same category, and otherwise $s_{ijm} = 0$. As in continuous measurements, $w_{ijm} = 1$ unless the category for measurement m is unknown for one or both of the records, in which case $w_{ijm} = 0$.

Similarity measure
(Measuring distance between records)

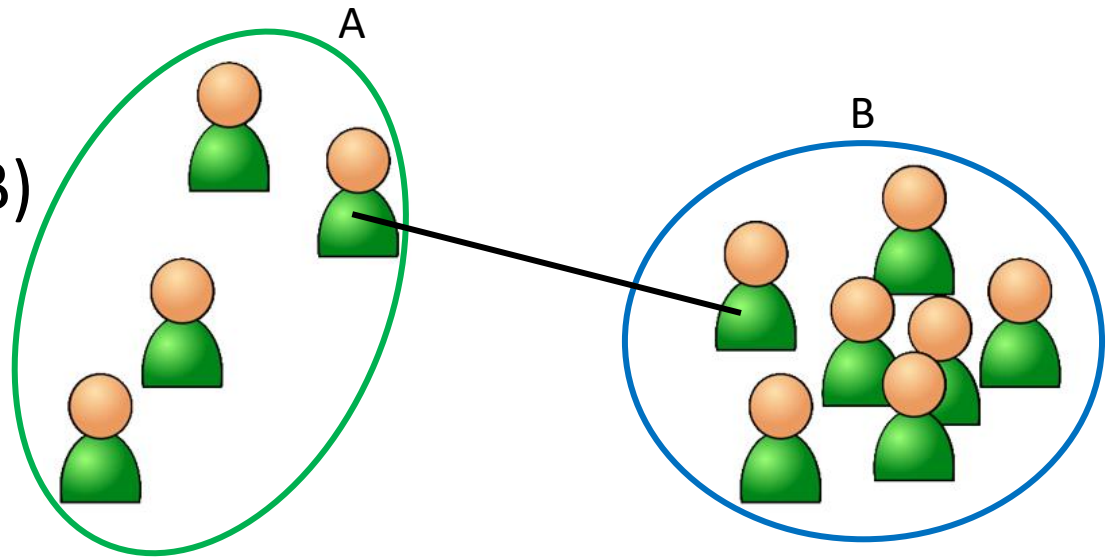
- Gower's similarity (for mixed variable types: continuous & categorical)
  - Weighted average of the distances computed for each variable.

$$s_{ij} = \frac{\sum_{m=1}^{p} w_{ijm} s_{ijm}}{\sum_{m=1}^{p} w_{ijm}},$$

	x1	x2	x3	x4	x5
e	0	0	56	0	32
g	1	1	43	1	35
h	1	1	55	0	22
i	1	0	43	1	23
j	0	0	53	1	31
k	0	1	54	0	43
max	1	1	56	1	43
min	0	0	43	0	22
S_{ijm}	0	0	0.23	1	0.62
w_{ijm}	1	0	1	1	1
S_{ijm}	1.46				

Similarity measure
(Measuring distance between clusters)

- **Minimum Distance** (Cluster A to Cluster B)

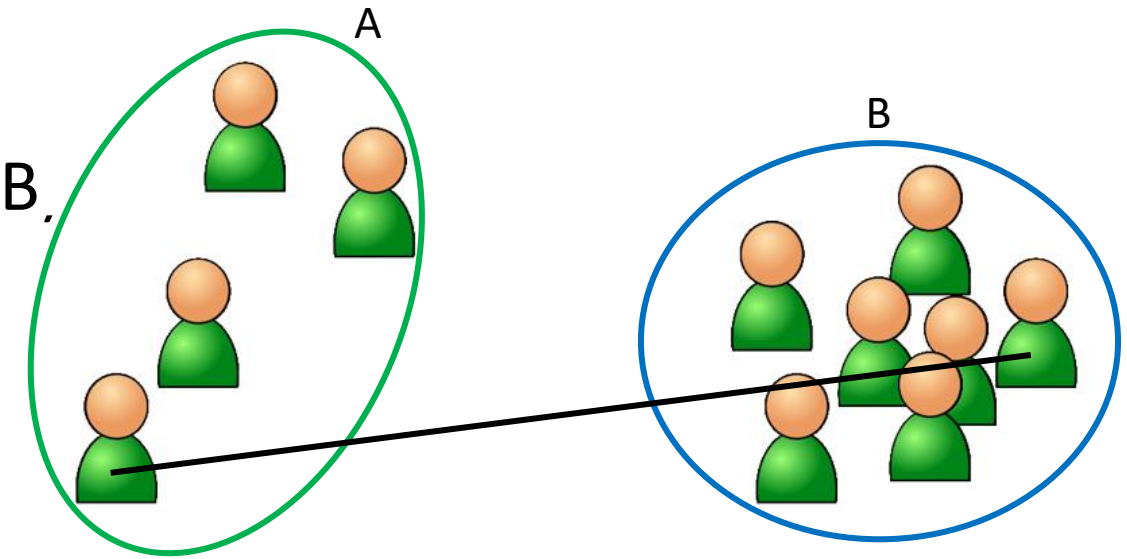


- Also called **single linkage**

- Distance between two clusters is the distance between the pairs of records A_i and B_j that are closest

Similarity measure
(Measuring distance between clusters)

- **Maximum Distance** (Cluster A to Cluster B)

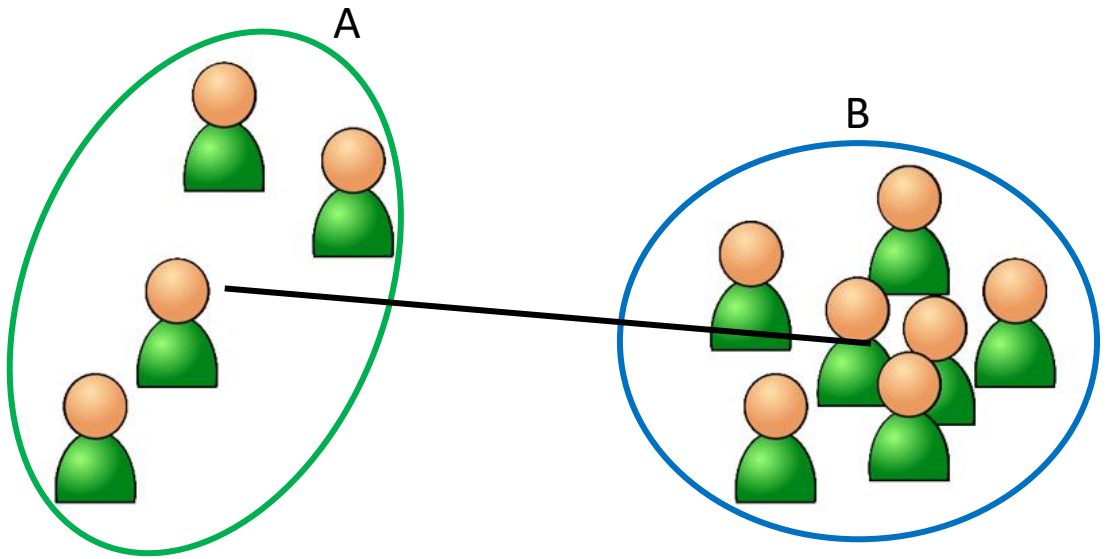


- Also called **complete linkage**

- Distance between two clusters is the distance between the pairs of records A_i and B_j that are farthest from each other

Similarity measure
(Measuring distance between clusters)

- **Average Distance** (Cluster A to Cluster B)

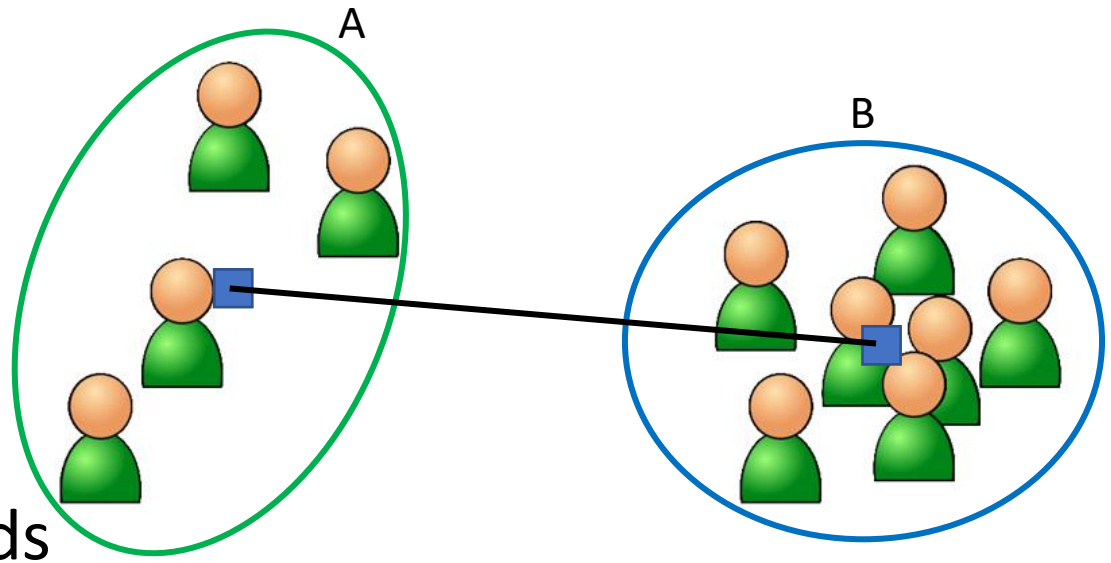


- Also called **average linkage**

- Distance between two clusters is the average of all possible pair-wise distances

Similarity measure
(Measuring distance between clusters)

- **Centroid Distance** (Cluster A to Cluster B)



- Distance between two clusters is the distance between the two cluster centroids

- Centroid is the vector of variable averages for all records in a cluster

Major Clustering Approaches

- Partitioning approach:
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  - Typical methods: k-means, k-medoids, CLARANS

- Hierarchical approach:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Diana, Agnes, BIRCH, CAMELEON

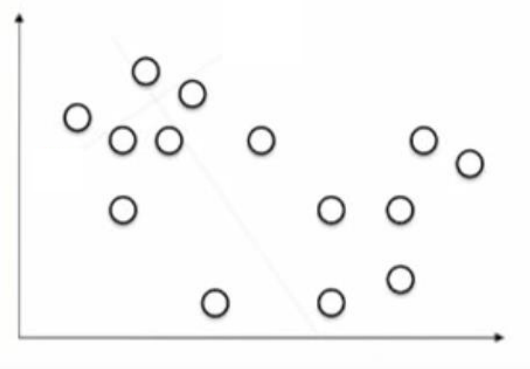
K-Means Clustering Algorithm

1. Choose # of clusters desired, *k*
2. Start with a partition into k clusters
   Often based on random selection of k centroids
3. At each step, move each record to cluster with closest centroid
4. Recompute centroids, repeat step 3
5. Stop when moving records increases within-cluster dispersion

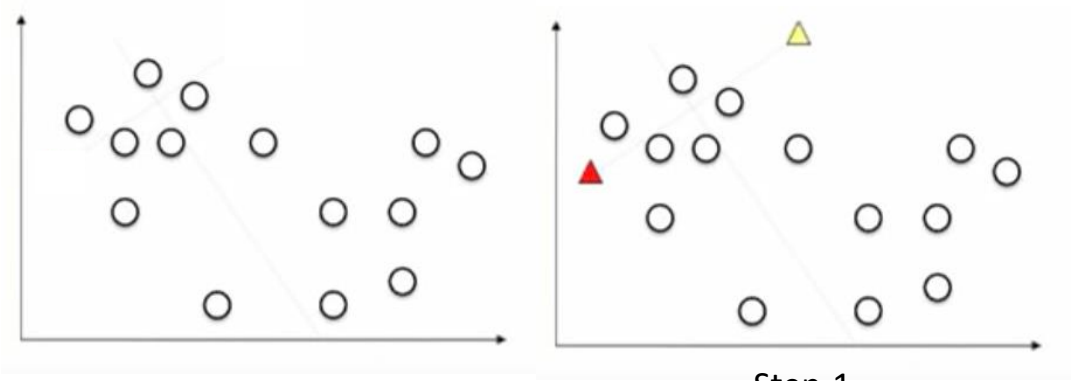
K-Means Algorithm: Choosing k and Initial Partitioning

- Choose *k* based on the how results will be used
  - e.g. “How many market segments do we want?”
- Also experiment with slightly different *k*’s
- Initial partition into clusters can be random, or based on domain knowledge
  - If random partition, repeat the process with different random partitions

K-Means Clustering (step-by-step)

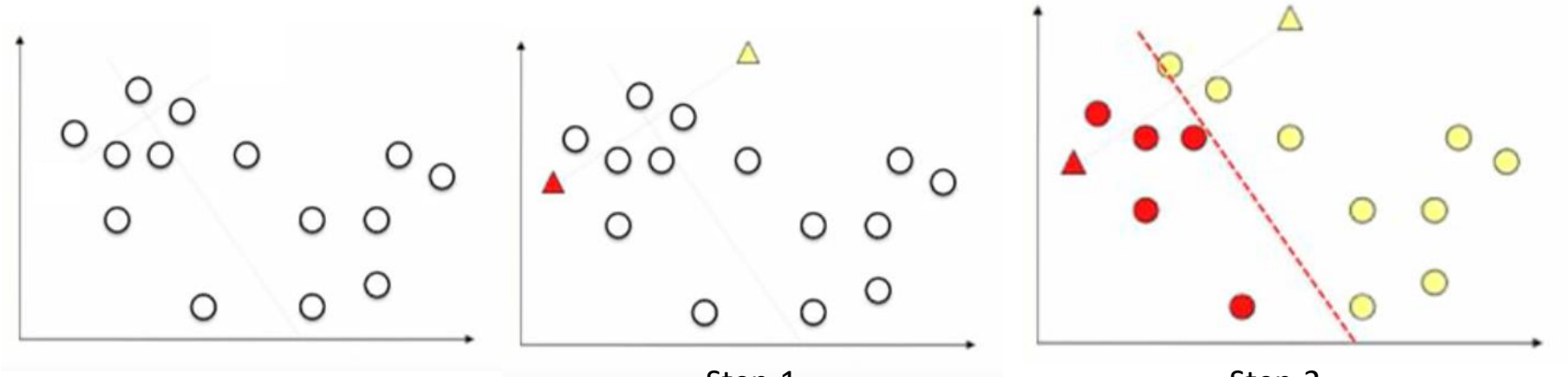


K-Means Clustering (step-by-step)



Step-1

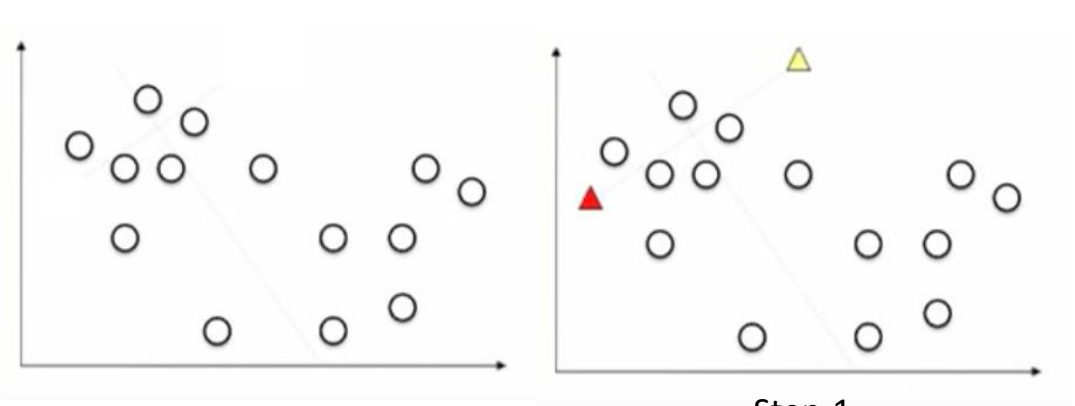
K-Means Clustering (step-by-step)



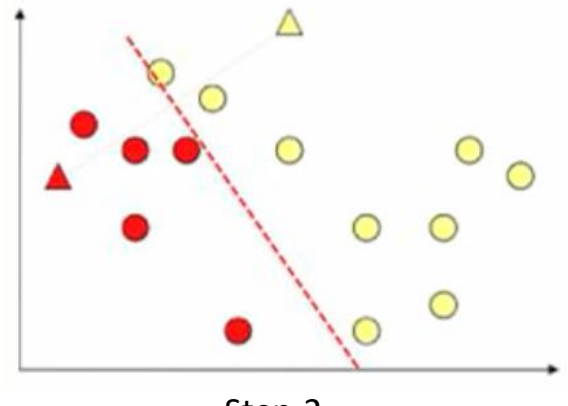
Step-1

Step-2

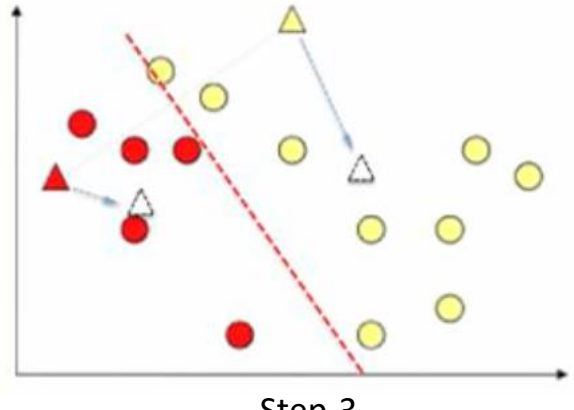
K-Means Clustering (step-by-step)



Step-1

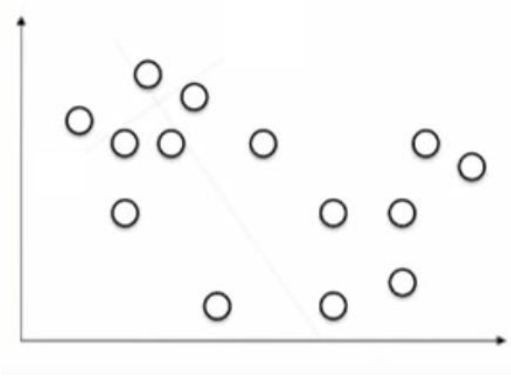


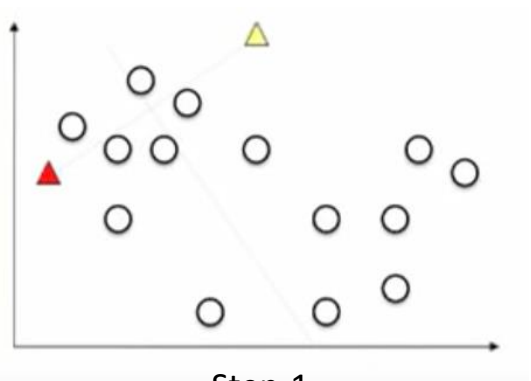
Step-2



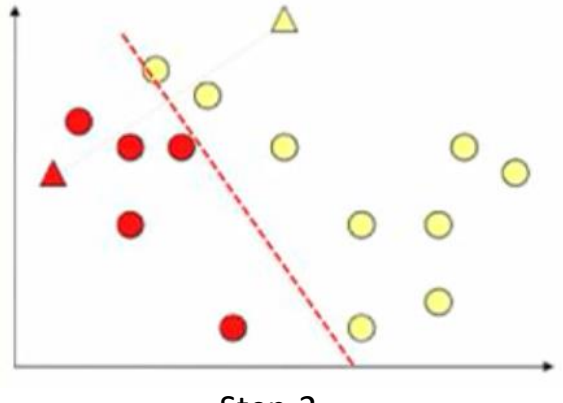
Step-3

K-Means Clustering (step-by-step)

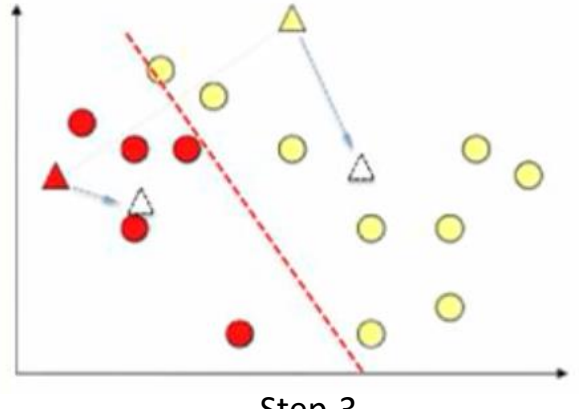




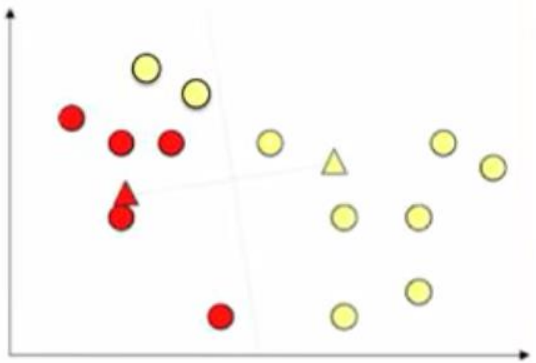
Step-1



Step-2

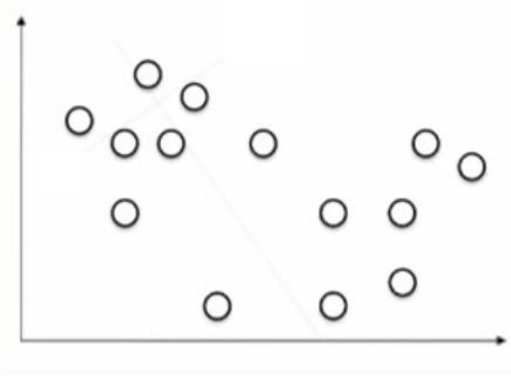


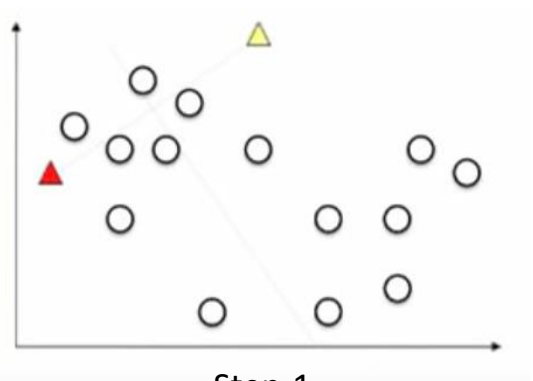
Step-3



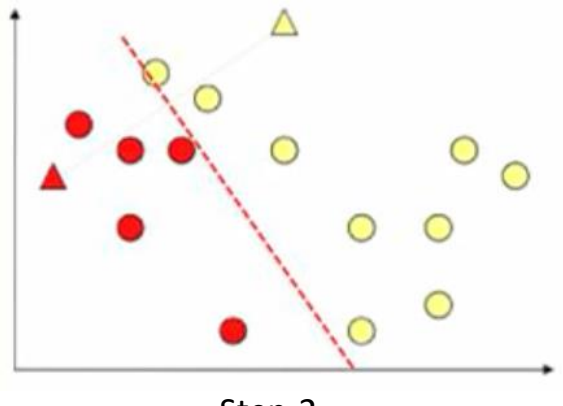
Step-4

K-Means Clustering (step-by-step)

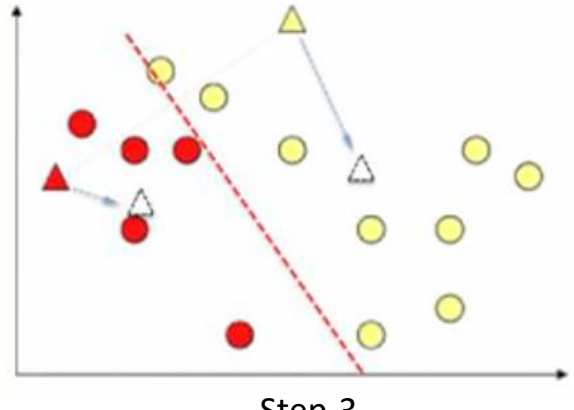




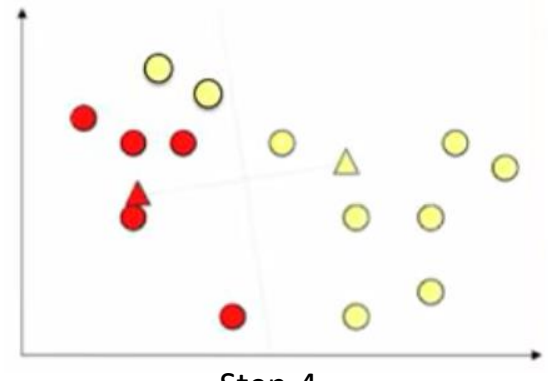
Step-1



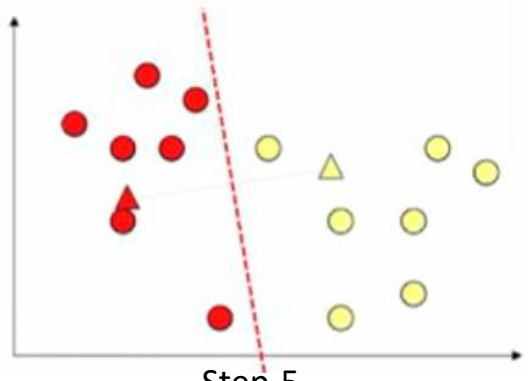
Step-2



Step-3

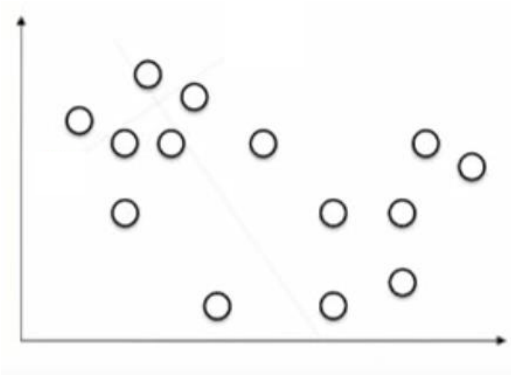


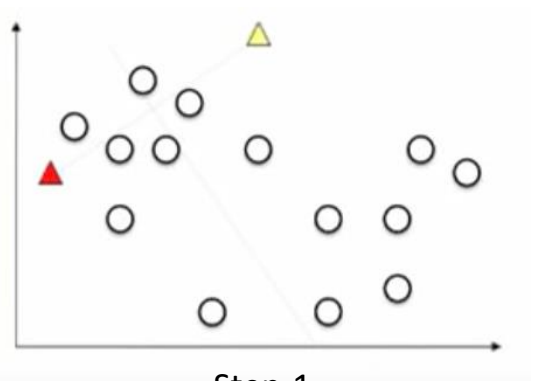
Step-4



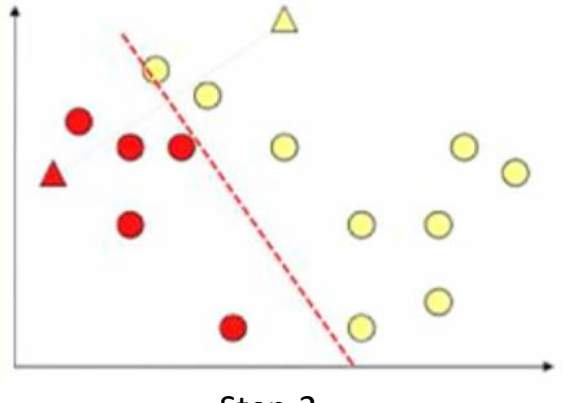
Step-5

K-Means Clustering (step-by-step)

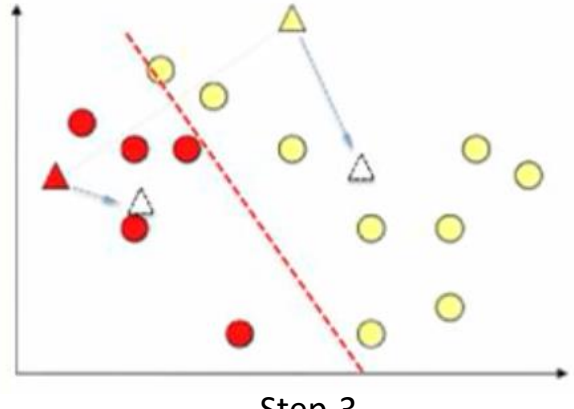




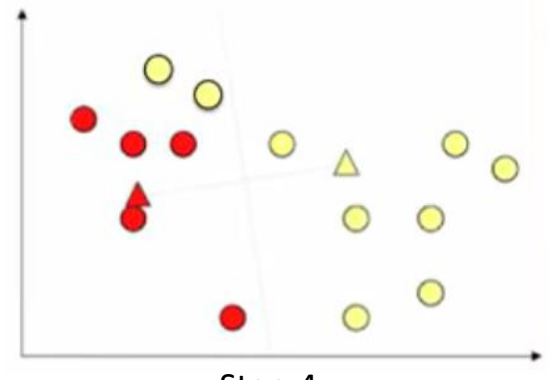
Step-1



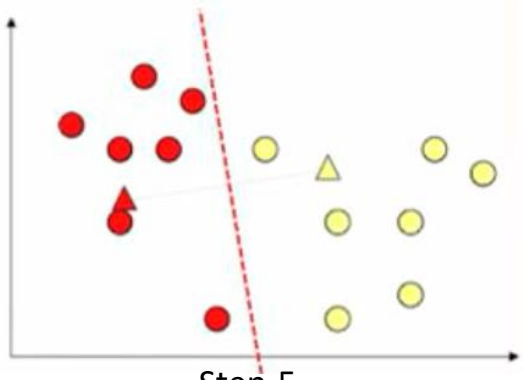
Step-2



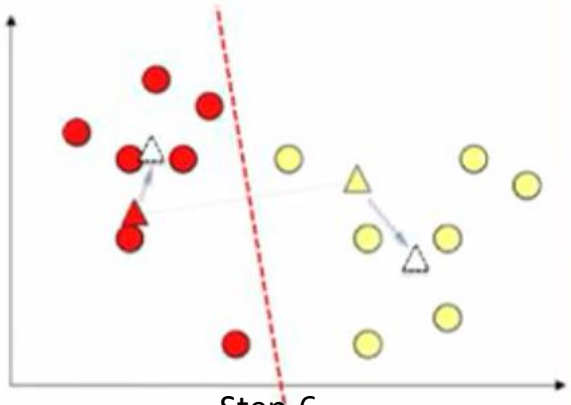
Step-3



Step-4

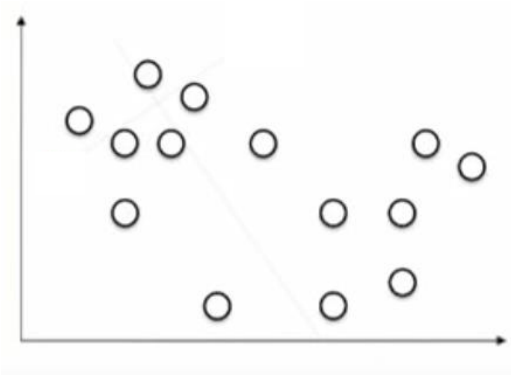


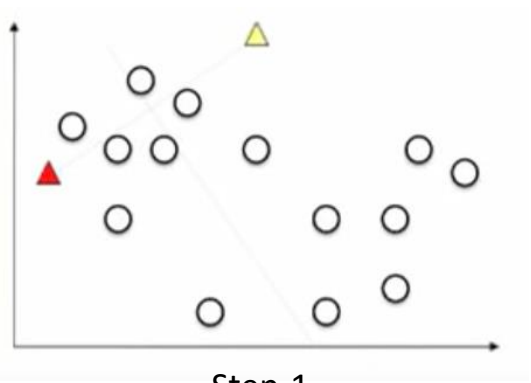
Step-5



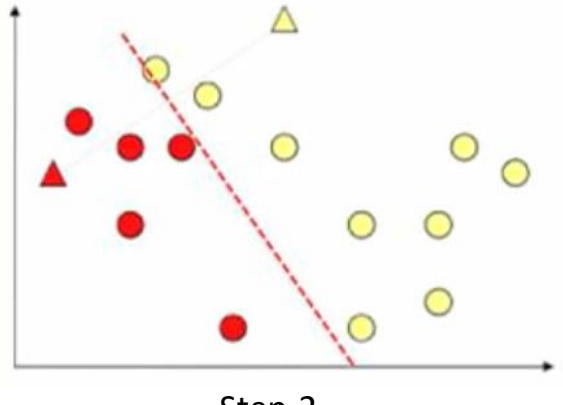
Step-6

K-Means Clustering (step-by-step)

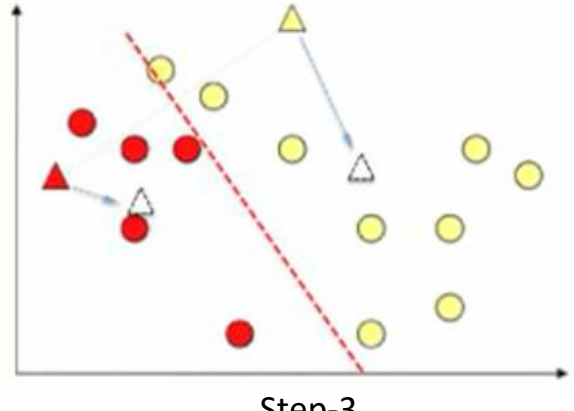




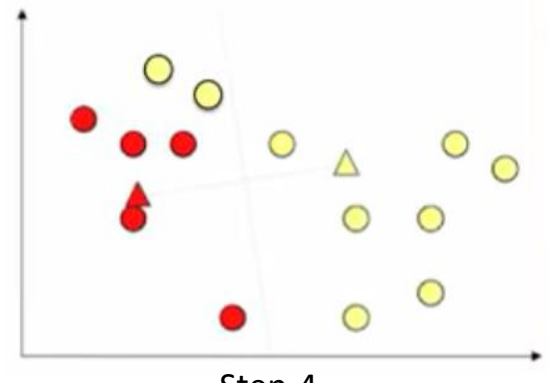
Step-1



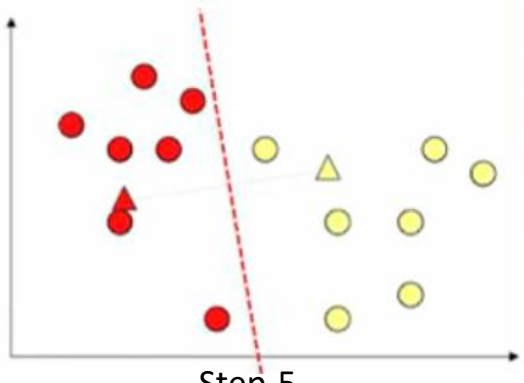
Step-2



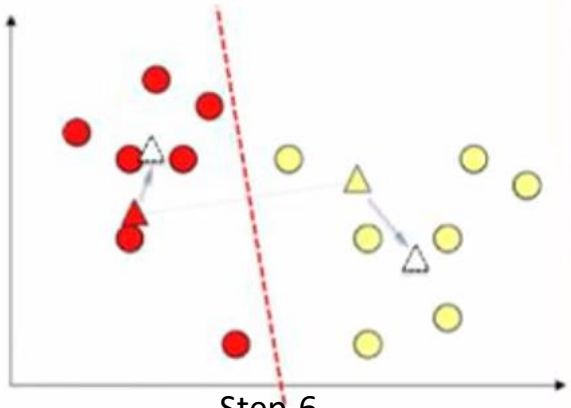
Step-3



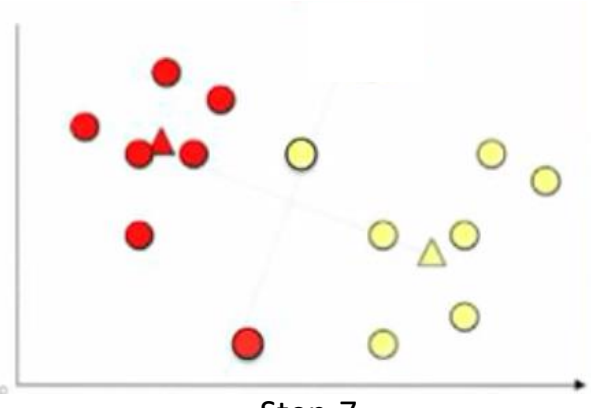
Step-4



Step-5

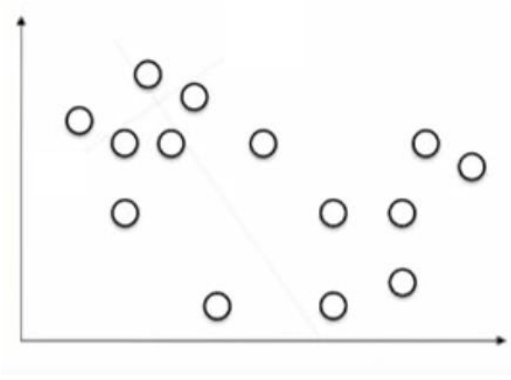


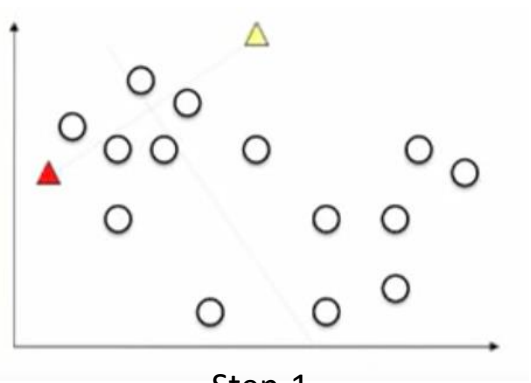
Step-6



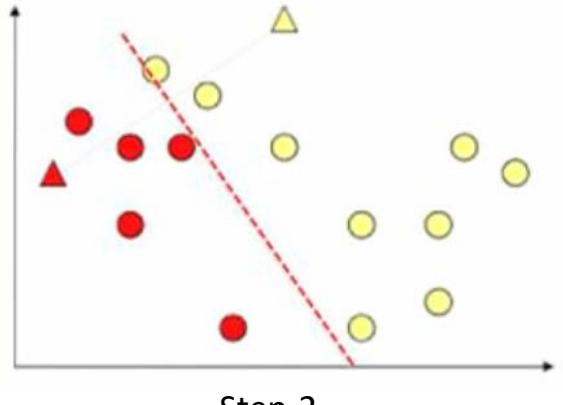
Step-7

K-Means Clustering (step-by-step)

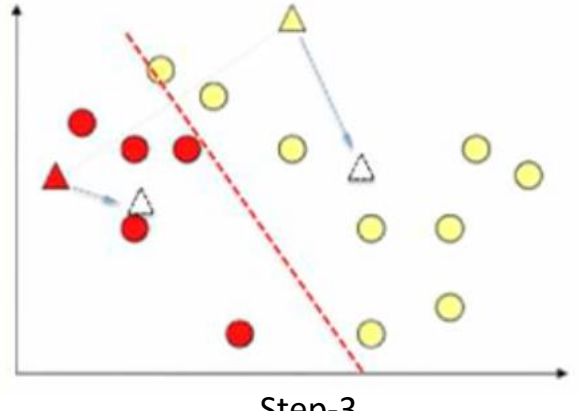




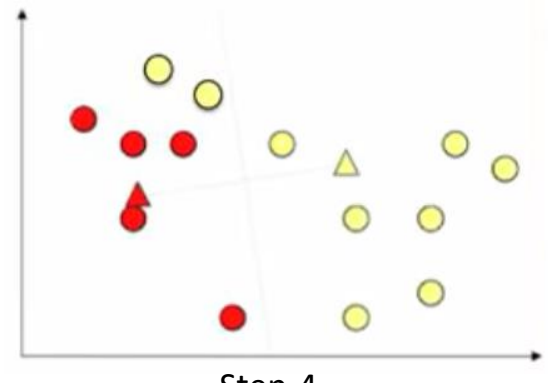
Step-1



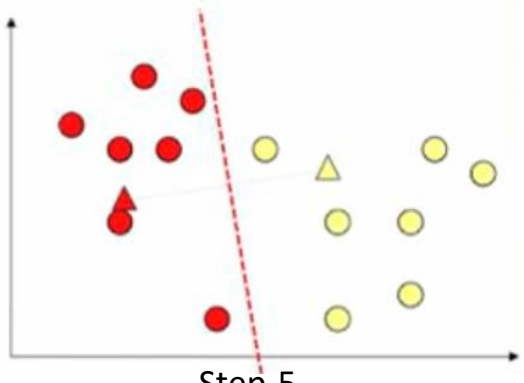
Step-2



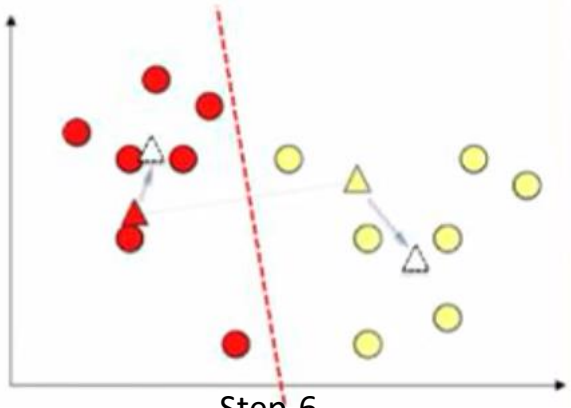
Step-3



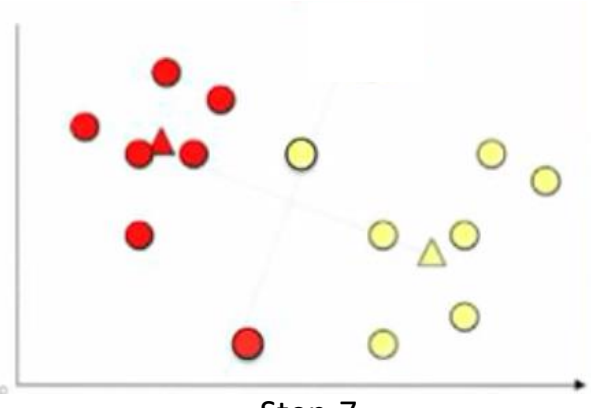
Step-4



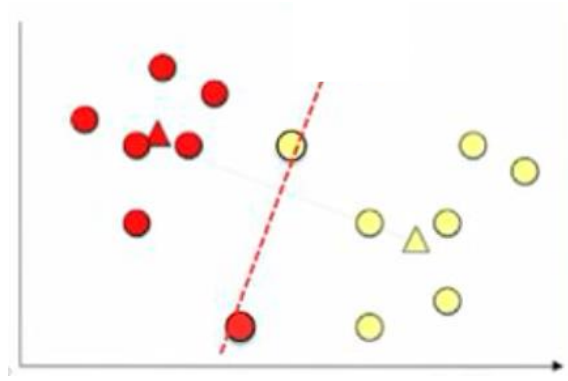
Step-5



Step-6

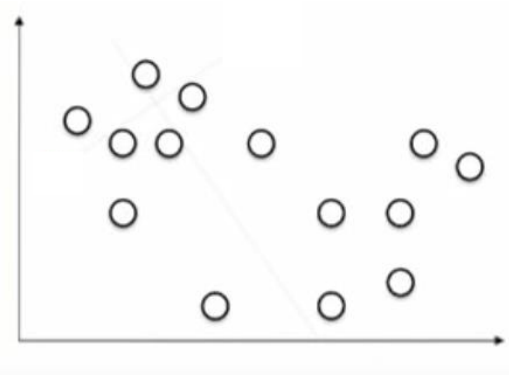


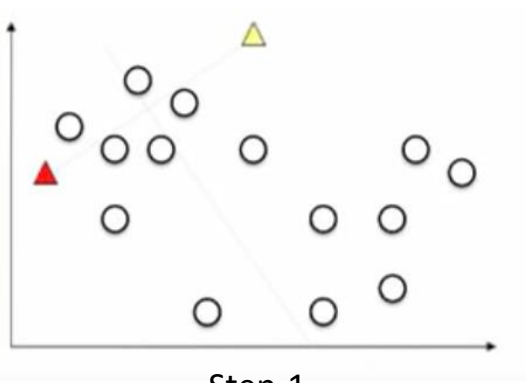
Step-7



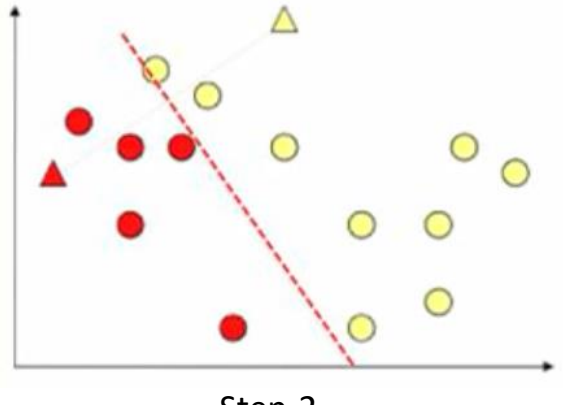
Step-8

K-Means Clustering (step-by-step)

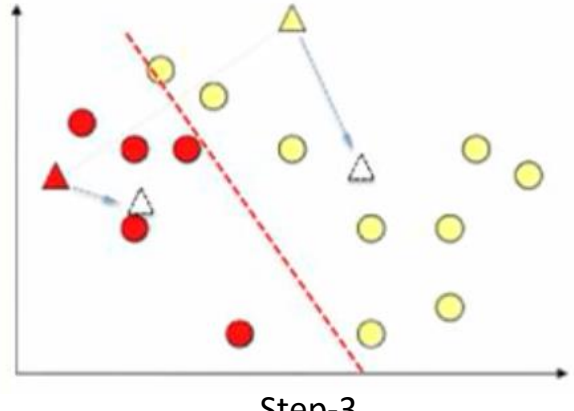




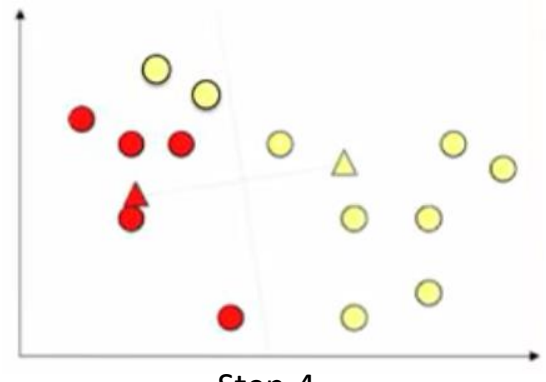
Step-1



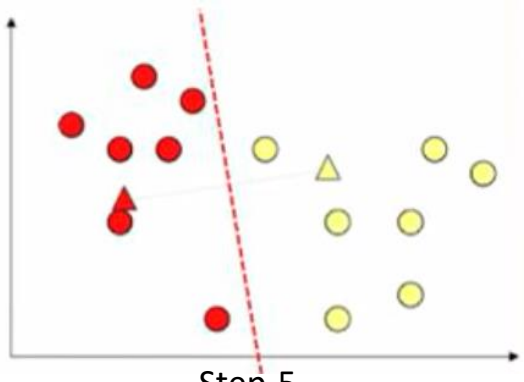
Step-2



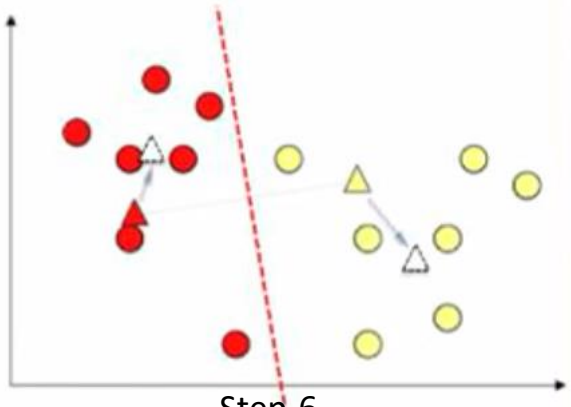
Step-3



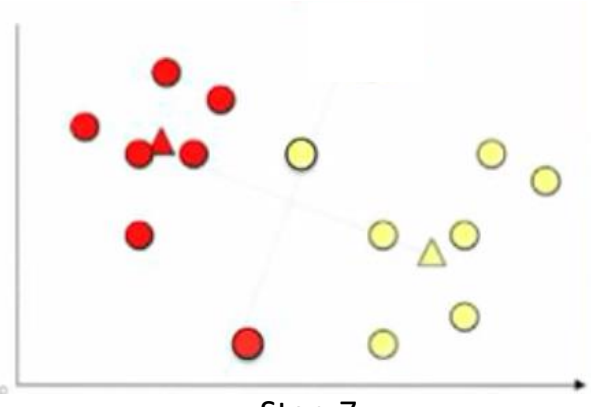
Step-4



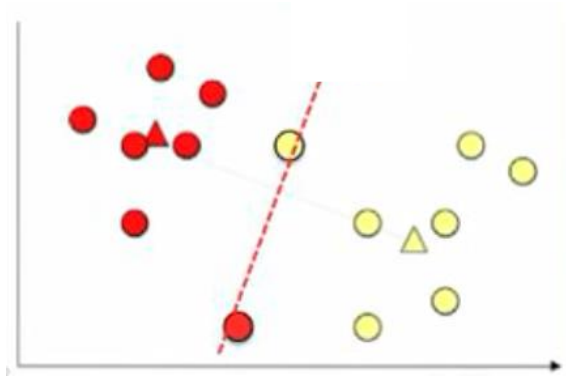
Step-5



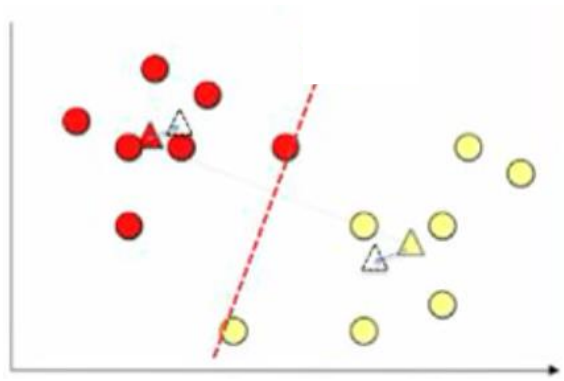
Step-6



Step-7

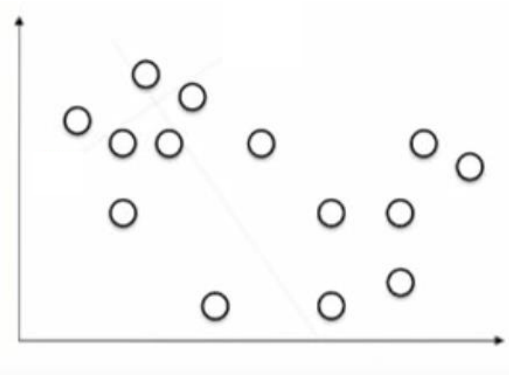


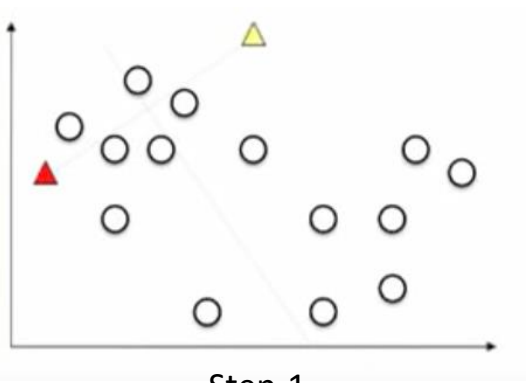
Step-8



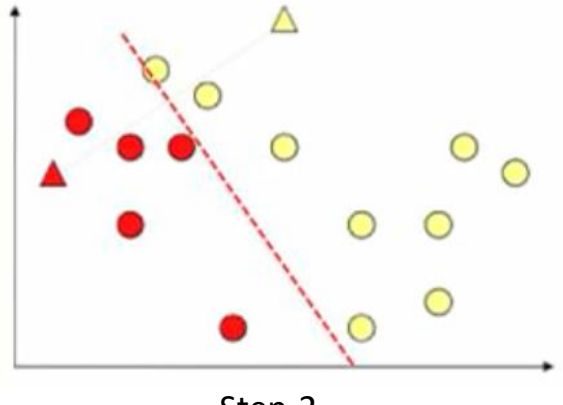
Step-9

K-Means Clustering (step-by-step)

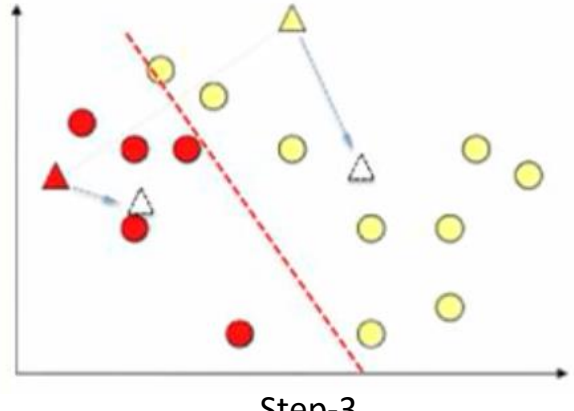




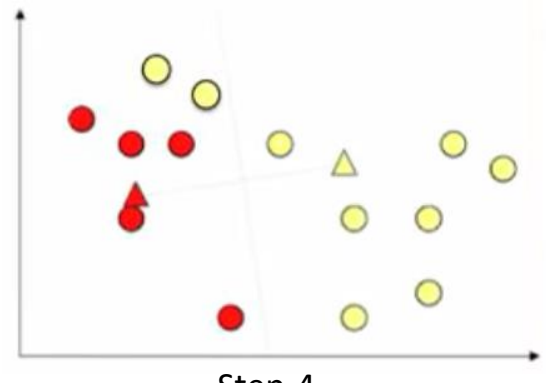
Step-1



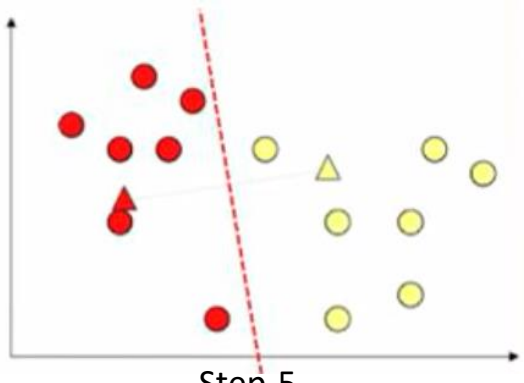
Step-2



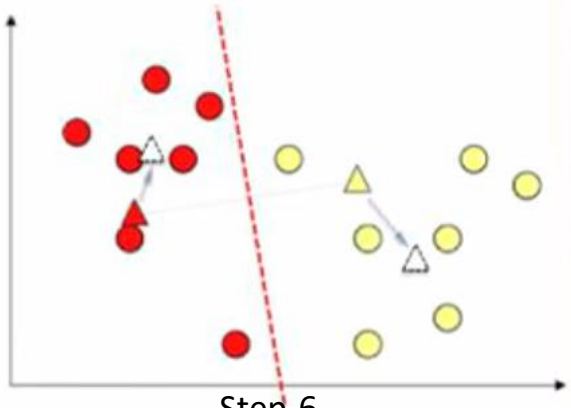
Step-3



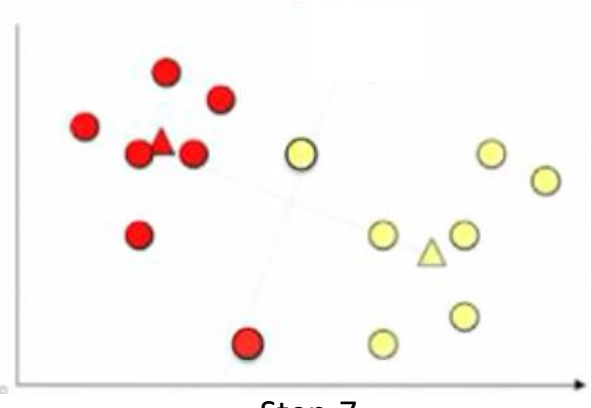
Step-4



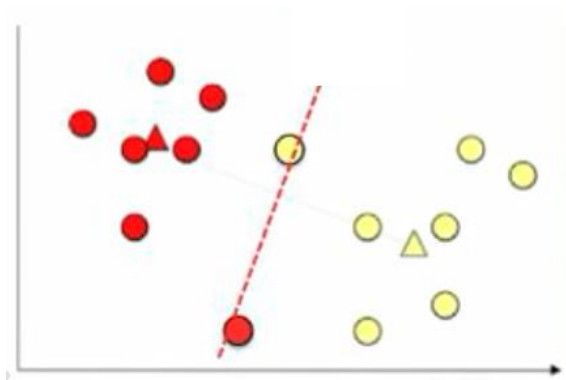
Step-5



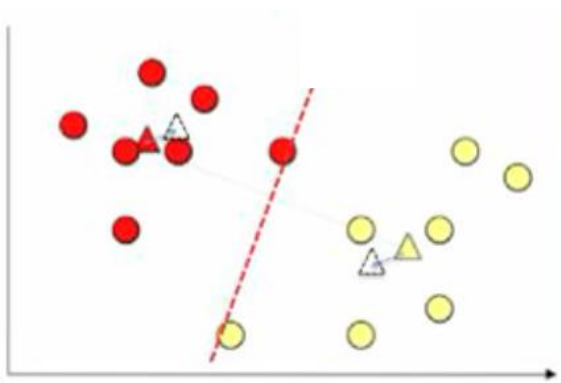
Step-6



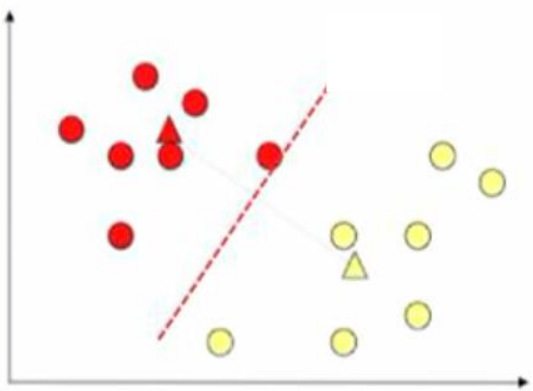
Step-7



Step-8



Step-9



Step-10

K-Means: Pros and Cons

Pros:

- Can be implemented with ease and it is faster than other clustering algorithms
- Works great on large scale data
- Results guarantee convergence
- Easily works with new examples

Cons:

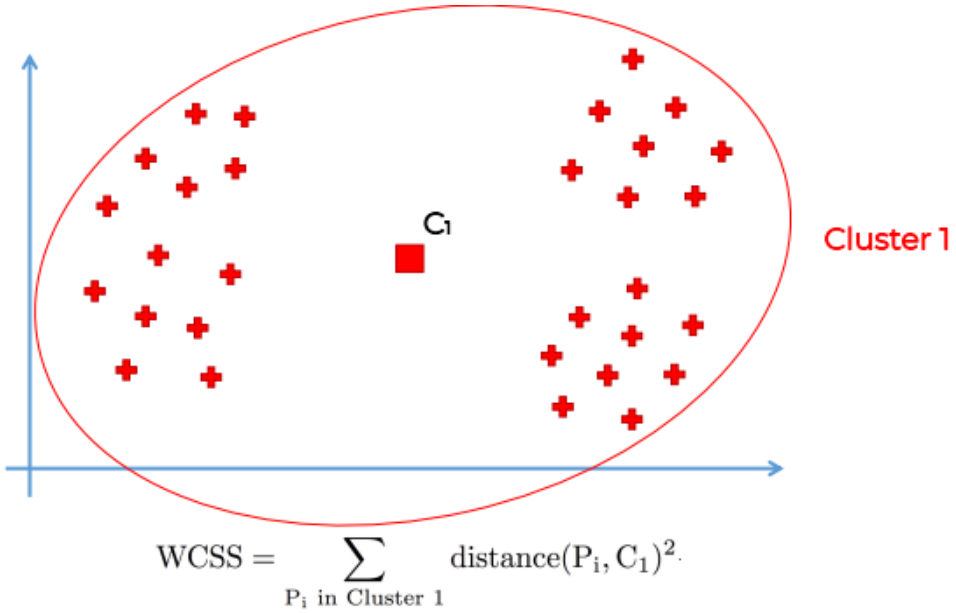
- Sensitive to outliers
- Quite difficult to determine the number of clusters
- Sensitive to initialization of cluster centers

The Elbow Method (Finding optimal # of clusters)

Within Cluster Sum of Squares:

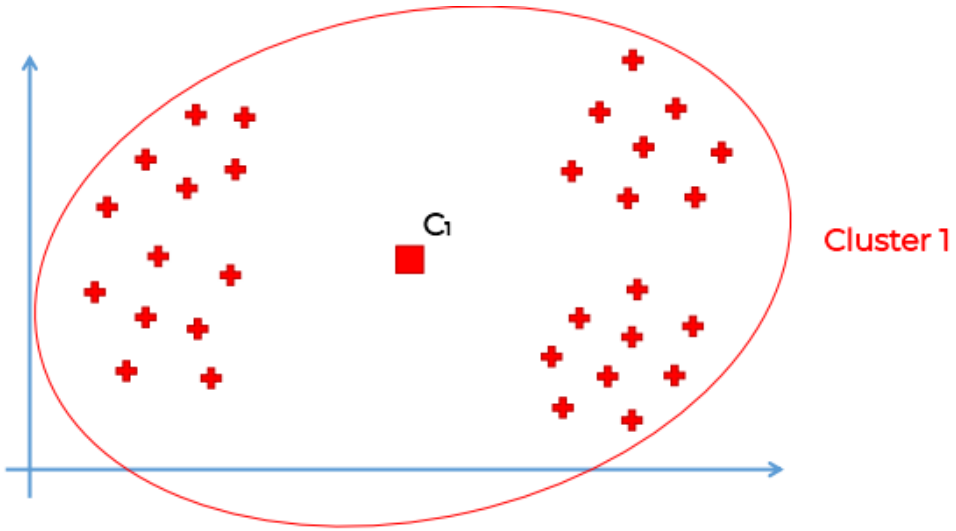
$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2 + \ldots$$

The Elbow Method (Finding optimal # of clusters)

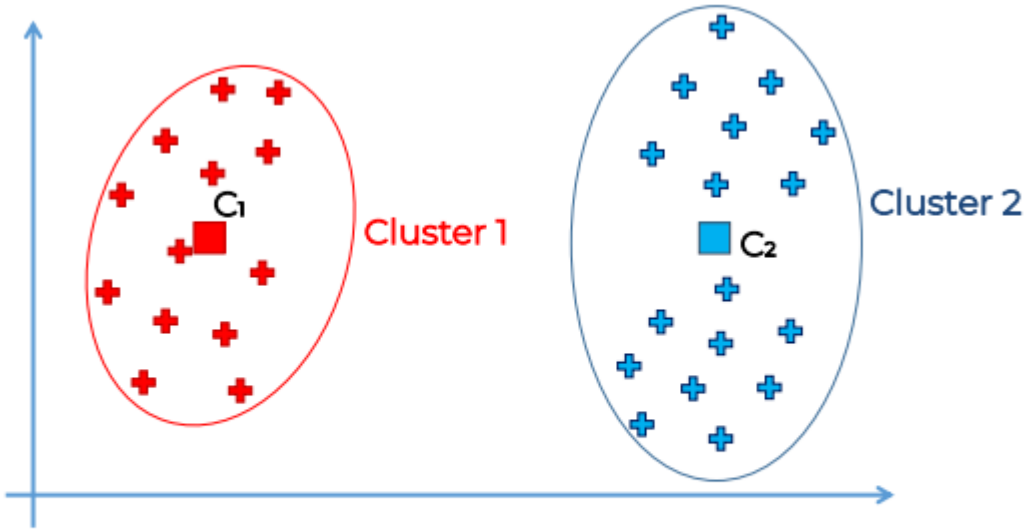


$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2$$

The Elbow Method (Finding optimal # of clusters)

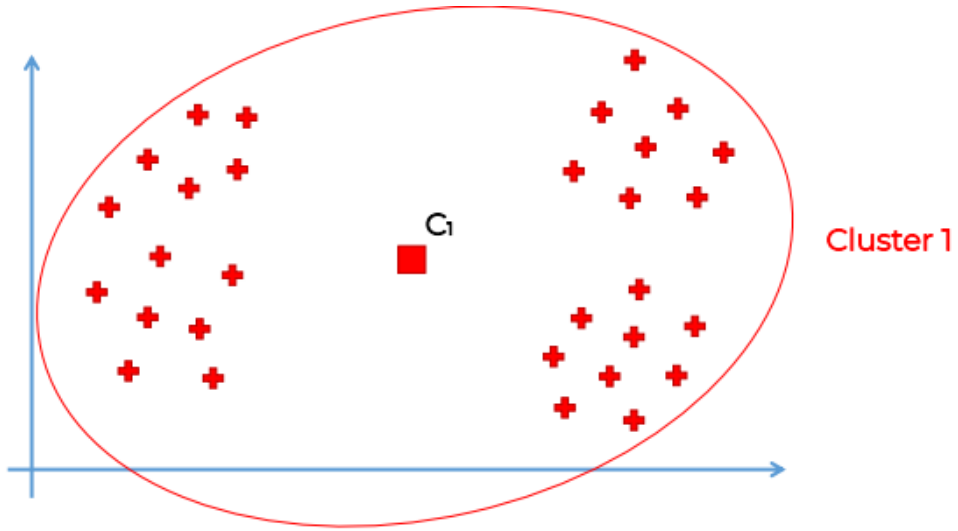


$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2$$

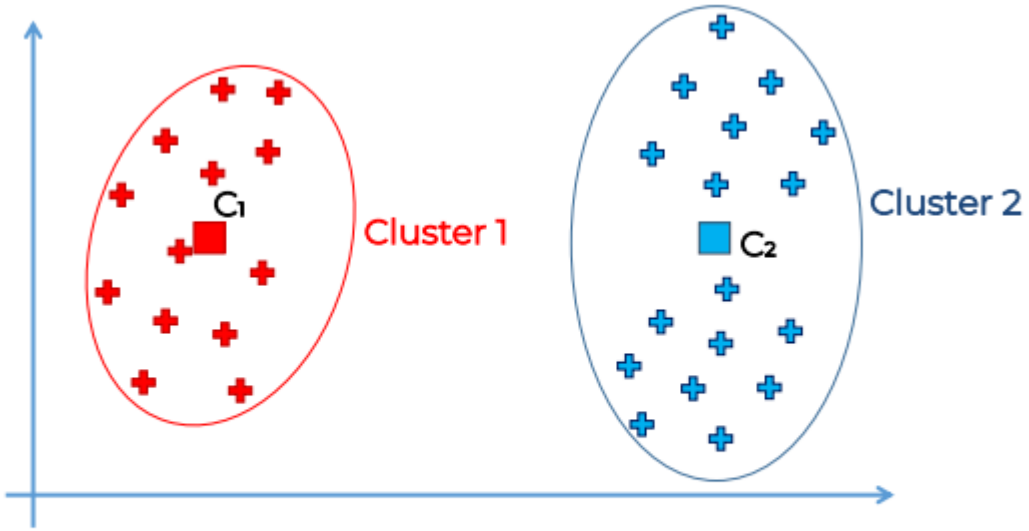


$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2$$

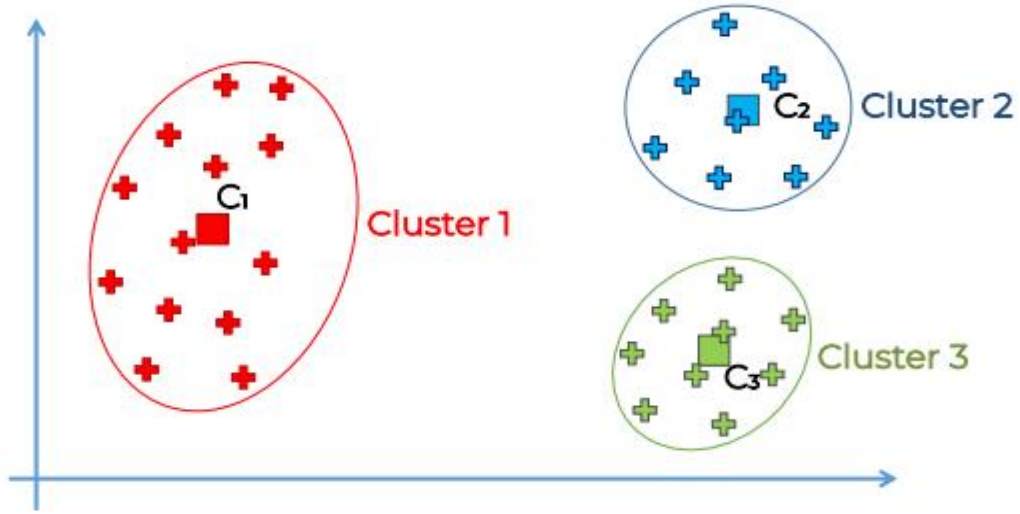
The Elbow Method (Finding optimal # of clusters)



$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2$$

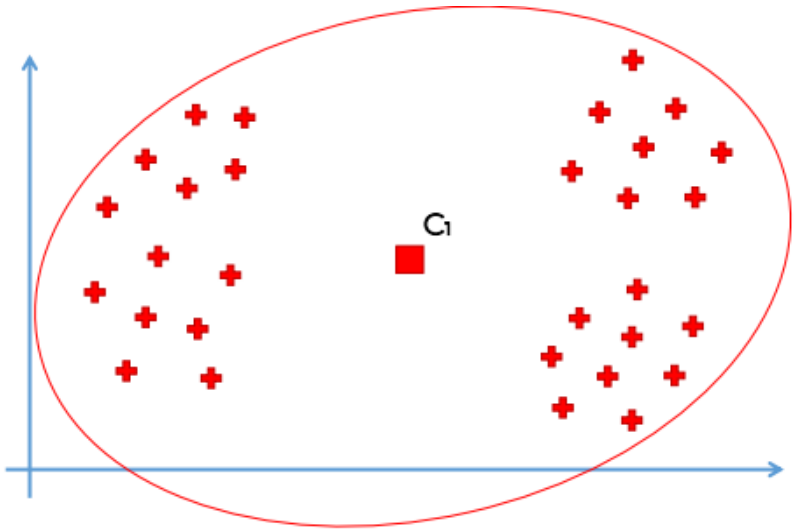


$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2$$

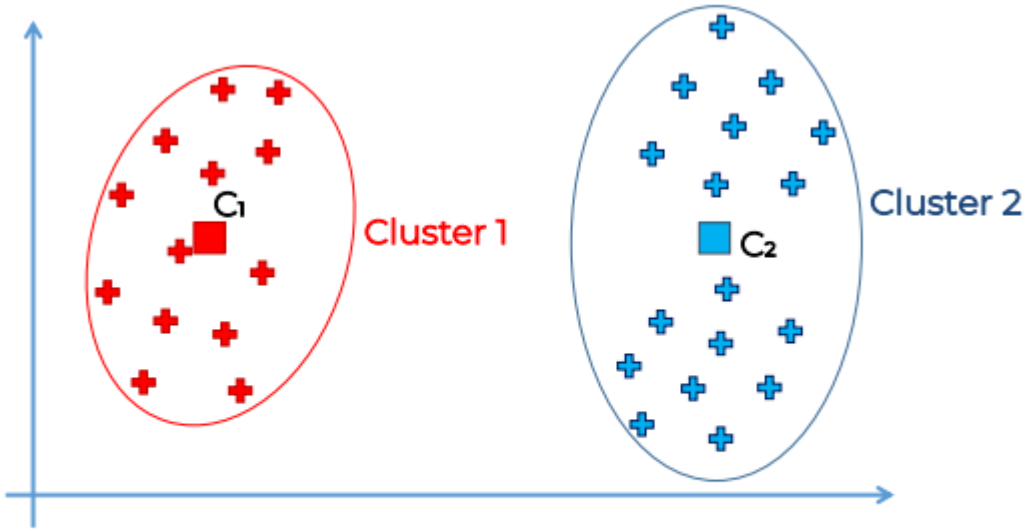


$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2 + \sum_{P_i \text{ in Cluster 3}} \text{distance}(P_i, C_3)^2$$

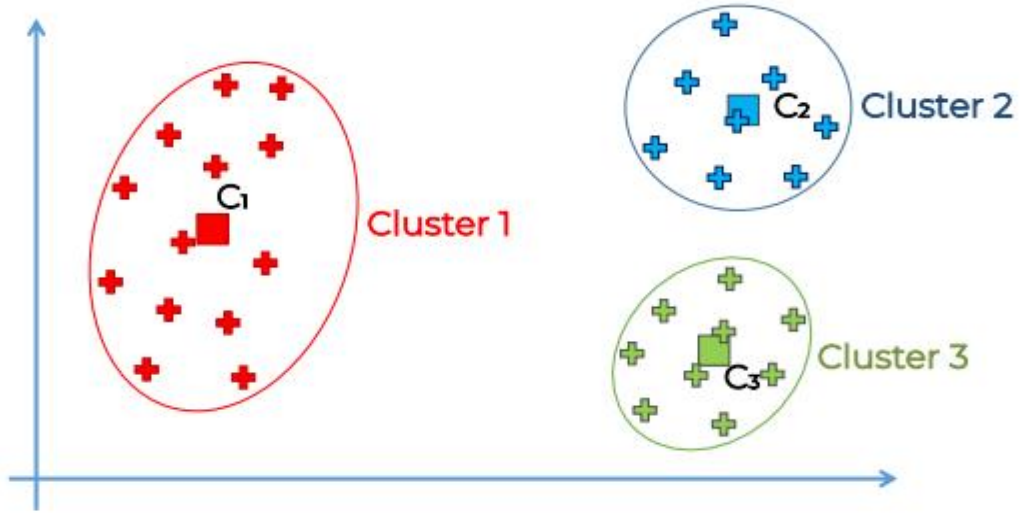
The Elbow Method (Finding optimal # of clusters)



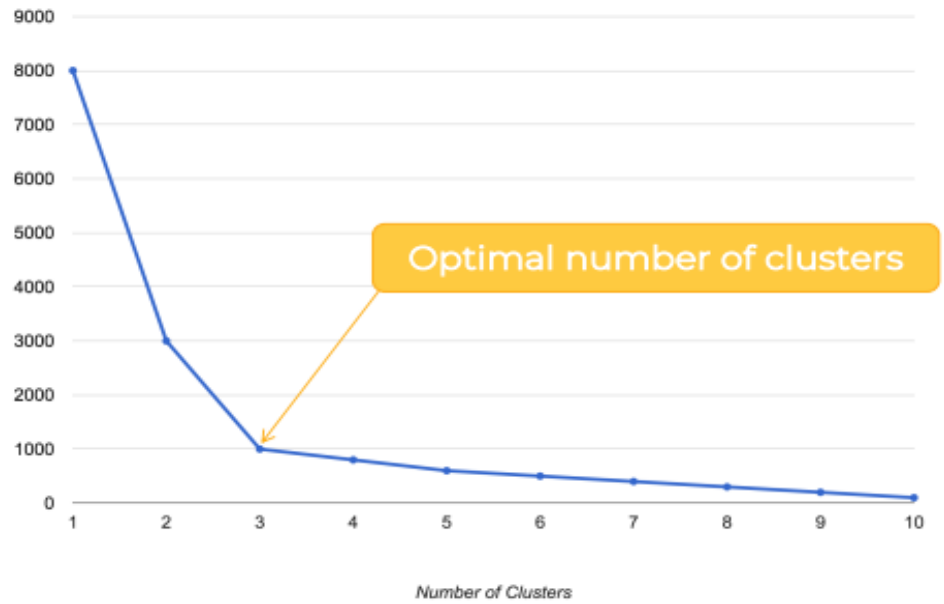
$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2$$



$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2$$



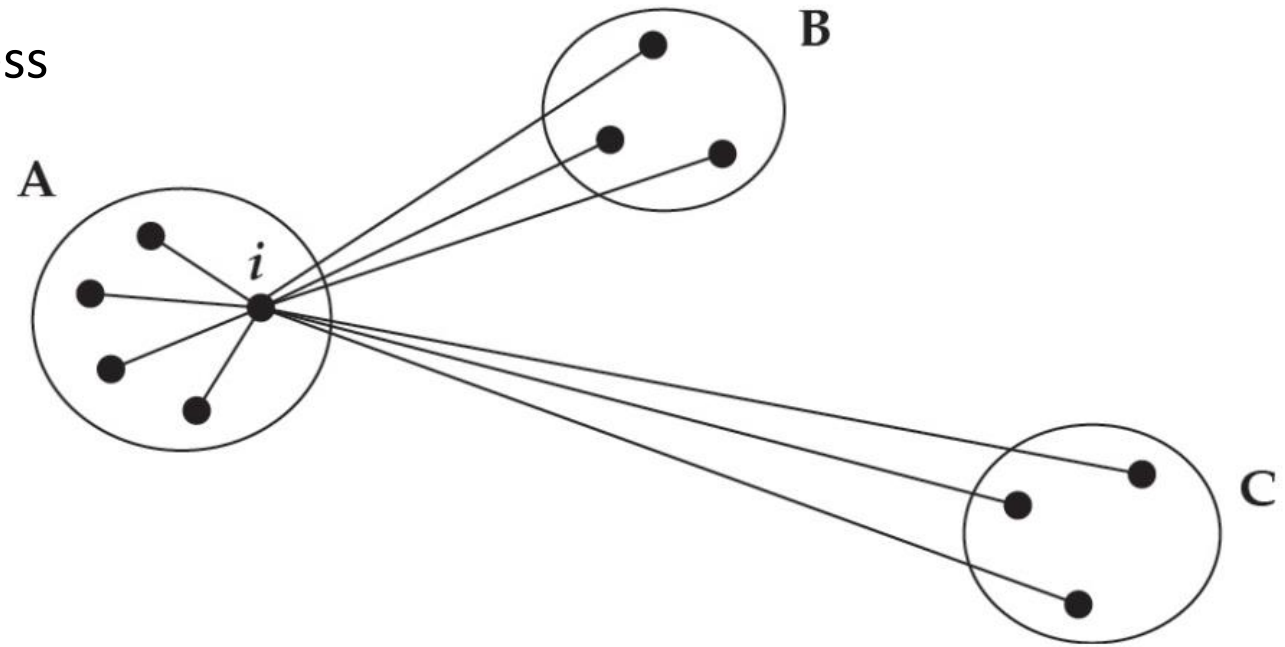
$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2 + \sum_{P_i \text{ in Cluster 3}} \text{distance}(P_i, C_3)^2$$



Silhouette Scores (Finding optimal # of clusters)

- Silhouette score: metric to measure the goodness of clustering algorithms

1. Find the average distance of the data point to each other point in the same cluster.
2. Find the nearest cluster (that the point itself is not a part of) by comparing the distance of that point to each of the other centroids.
3. Find the point's average distance to each point in that cluster



$$\frac{Avg.dist.to\ those\ in\ the\ nearest\ neighboring\ cluster\ - Avg.dist.to\ those\ in\ own\ cluster}{The\ maximum\ of\ those\ two\ averages}$$

It's value ranges between -1 to +1

- 1 indicates tight, well-separated clusters
- 0 indicates clusters not well separable
- -1 indicates data points of one cluster is closer to centroid of another cluster than the centroid of its own clusters