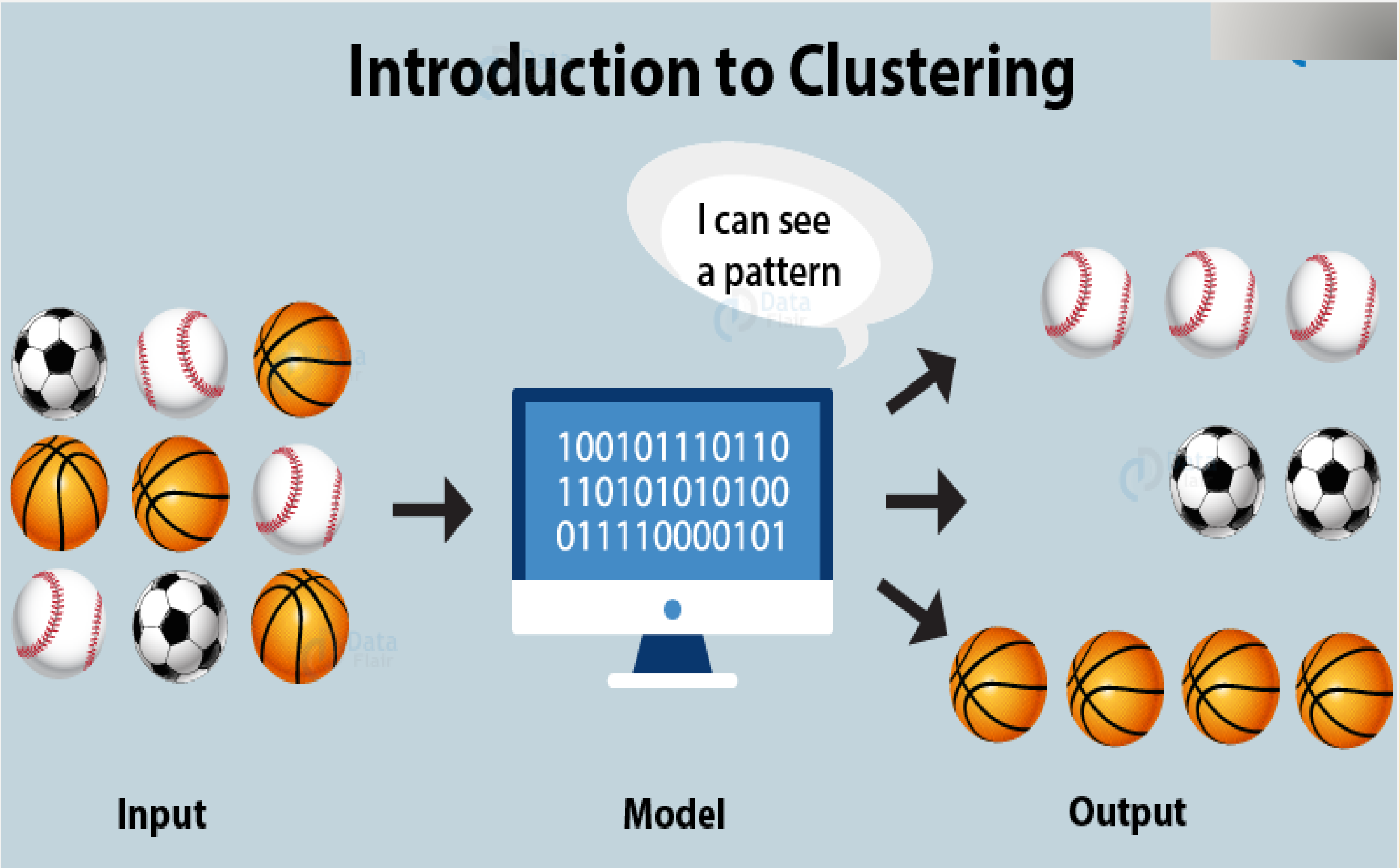
CLUSTERING IN MACHINE LEARNING

TABLE OF CONTENTS

INTRODUCTION OF CLUSTERING

- **Clustering** is the task of dividing the data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.

- **Clustering** is a method of unsupervised **learning**, and a common technique for statistical data analysis used in many fields.


Introduction to Clustering
I can see
a pattern
100101110110
110101010100
011110000101
Input
Model
Output

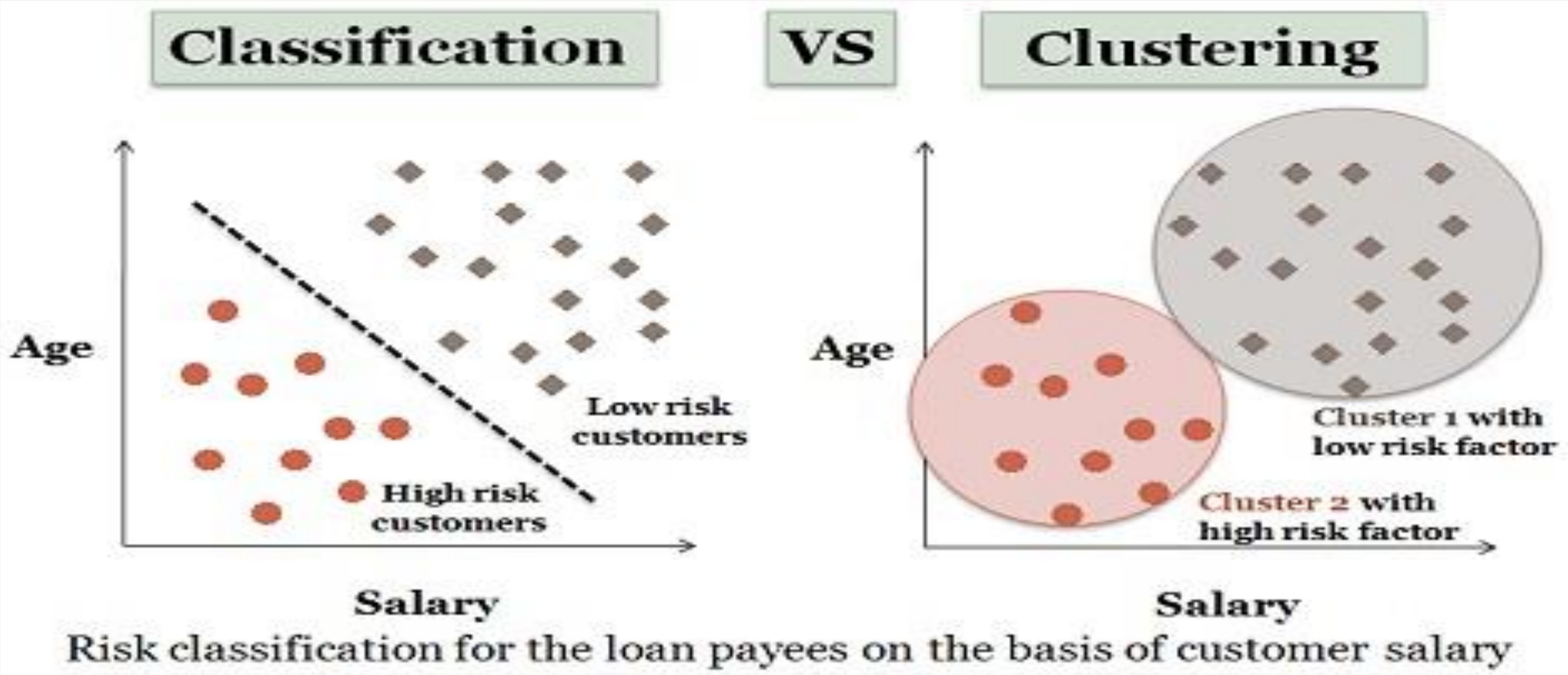
WHY DO WE USE CLUSTERING IN ML?

In basic terms, the objective of **clustering is** to find different groups within the elements in the data. To **do** so, **clustering** algorithms find the structure in the data so that elements of the same **cluster** (or group) **are** more similar to each other than to those from different **clusters**

DIFFERENCE BETWEEN CLUSTERING AND CLASSIFICATION?

- Classification and Clustering are the two types of learning methods which characterize objects into groups by one or more
- **Classification** is used in supervised learning technique where predefined labels are assigned to instances by properties, on the contrary,
- **Clustering** is used in unsupervised learning where similar instances are grouped, based on their features or properties.

DIFFERENCE BETWEEN CLUSTERING AND CLASSIFICATION?



APPLICATIONS OF CLUSTERING IN IR

- Whole corpus analysis/navigation
  - Better user interface: search without typing
- For improving recall in search applications
  - Better search results (like pseudo RF)
- For better navigation of search results
  - Effective "user recall" will be higher
- For speeding up vector space retrieval
  - Cluster-based retrieval gives faster search

CLUSTERING ALGORITHMS

- Flat algorithms
  - Usually start with a random (partial) partitioning
  - Refine it iteratively
    - *K* means clustering
    - (Model based clustering)
- Hierarchical algorithms
  - Bottom-up, agglomerative
  - (Top-down, divisive)

HARD VS. SOFT CLUSTERING

- **Hard clustering**: Each document belongs to exactly one cluster.
- **Soft clustering**: A document can belong to more than one cluster.

  Makes more sense for applications like creating browsable hierarchies

  You may want to put a pair of sneakers in two clusters: (i) sports apparel and (ii) shoes

  You can only do that with a soft clustering approach.

PARTITIONING ALGORITHMS

- Partitioning method: Construct a partition of *n* documents into a set of *K* clusters
- Given: a set of documents and the number *K*
- Find: a partition of *K* clusters that optimizes the chosen partitioning criterion
  - Globally optimal
    - Intractable for many objective functions
    - Ergo, exhaustively enumerate all partitions
- Effective heuristic methods: K-means and K-medoids algorithms

K-MEANS

- Assumes documents are real-valued vectors.
- Clusters based on *centroids* (aka the *center of gravity* or mean) of points in a cluster, *c*:

$$\vec{\mu}(\mathrm{c}) = \frac{1}{|c|}\sum_{\vec{x}\in c}\vec{x}$$

- Reassignment of instances to clusters is based on distance to the current cluster centroids.
  - (Or one can equivalently phrase it in terms of similarities)

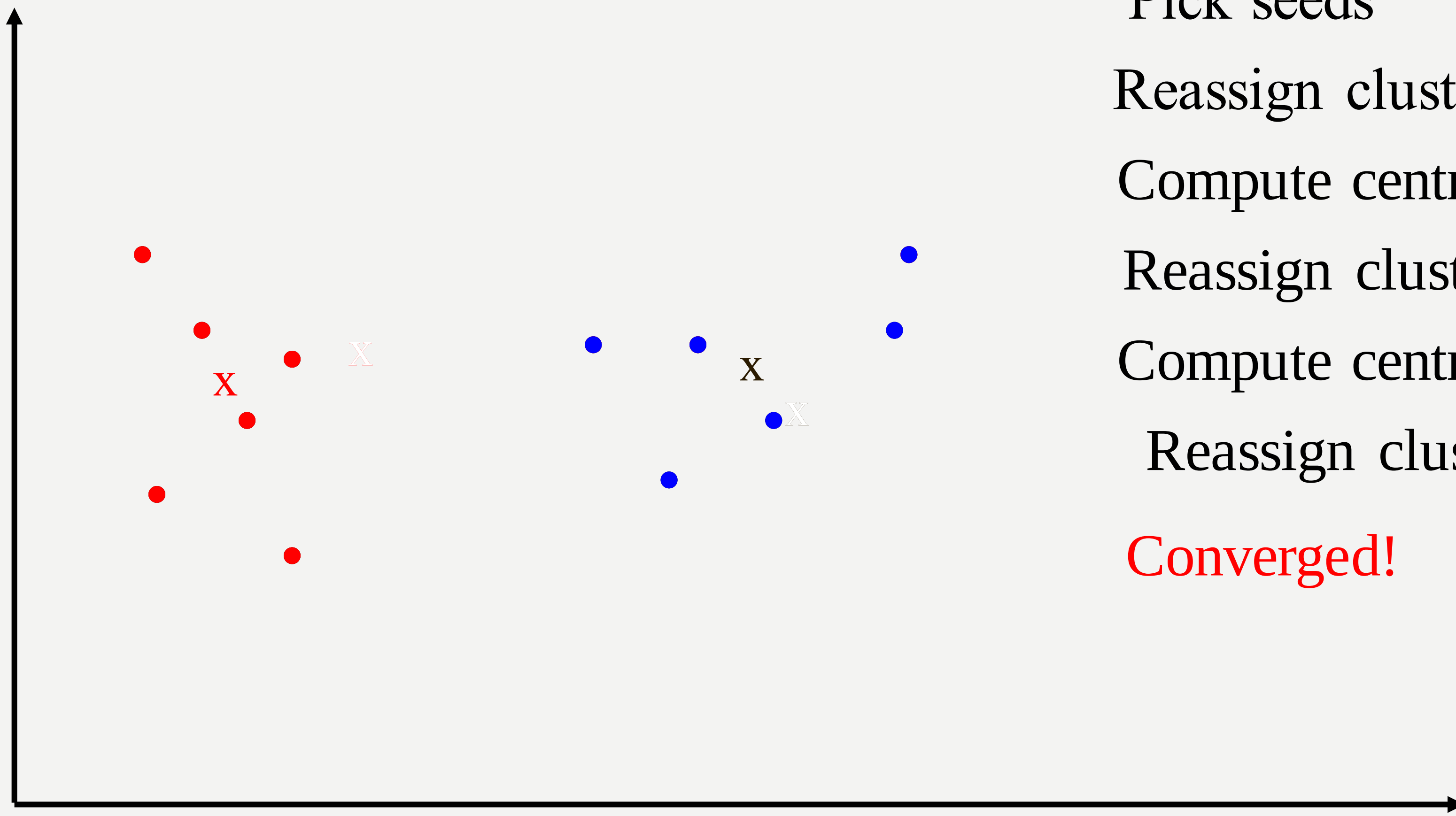
K-MEANS ALGORITHM

- Select K random docs $\{s_1, s_2, \ldots s_K\}$ as seeds.
- Until clustering converges (or other stopping criterion):

- For each doc d_i:
- Assign d_i to the cluster c_j such that $dist(x_i, s_j)$ is minimal.
- (Next, update the seeds to the centroid of each cluster)

- For each cluster c_j
- $s_j = \mu(c_j)$

K MEANS EXAMPLE (*K*=2)



Pick seeds

Reassign clusters

Compute centroids

Reassign clusters

Compute centroids

Reassign clusters

Converged!

TERMINATION CONDITIONS

- Several possibilities, e.g.,
  - A fixed number of iterations.
  - Doc partition unchanged.
  - Centroid positions don't change.

Does this mean that the docs in a cluster are unchanged?

CONVERGENCE

- Why should the *K*-means algorithm ever reach a *fixed point?*
  - A state in which clusters don't change.
- *K*-means is a special case of a general procedure known as the *Expectation Maximization (EM) algorithm.*
  - EM is known to converge.
  - Number of iterations could be large.
    - But in practice usually isn't

CONVERGENCE OF *K*-MEANS

- Define goodness measure of cluster *k* as sum of squared distances from cluster centroid:
  - $G_k = \Sigma_i (d_i – c_k)^2$ (sum over all d_i in cluster *k*)
- G = Σk Gk
- Reassignment monotonically decreases G since each vector is assigned to the closest centroid.

CONVERGENCE OF *K*-MEANS

- Recomputation monotonically decreases each Gk since (mk is number of members in cluster k):
  - $\Sigma (d_i - a)^2$ reaches minimum for:
  - $\Sigma -2(d_i - a) = 0$
  - $\Sigma d_i = \Sigma a$
  - $m_K a = \Sigma d_i$
  - $a = (1/ m_k) \Sigma d_i = c_k$
- K-means typically converges quickly

TIME COMPLEXITY

- Computing distance between two docs is O(*M*) where *M* is the dimensionality of the vectors.
- Reassigning clusters: O(*KN*) distance computations, or O(*KNM*).
- Computing centroids: Each doc gets added once to some centroid: O(*NM*).
- Assume these two steps are each done once for *I* iterations: O(*IKNM*).

THANKS!!!