

National University of Computer & Emerging Sciences



Lab Manual

CS461: Artificial Intelligence Lab

Course Instructor	Dr. Hafeez-Ur-Rehman
Lab Instructor	Muhammad Hamza
Semester	Spring 2022

K- Means Clustering

There are two main types of machine learning methods. Supervised learning techniques require labeled training data. Unsupervised learning techniques do not need label instances. Instead, they try to find patterns within the data itself.



Supervised

Labels associated with the training data is used to correct the algorithm

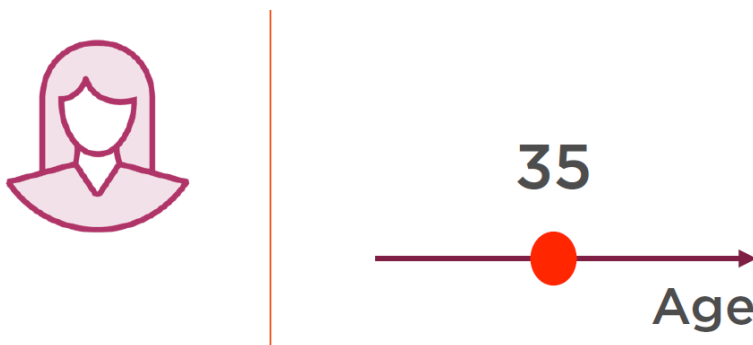


Unsupervised

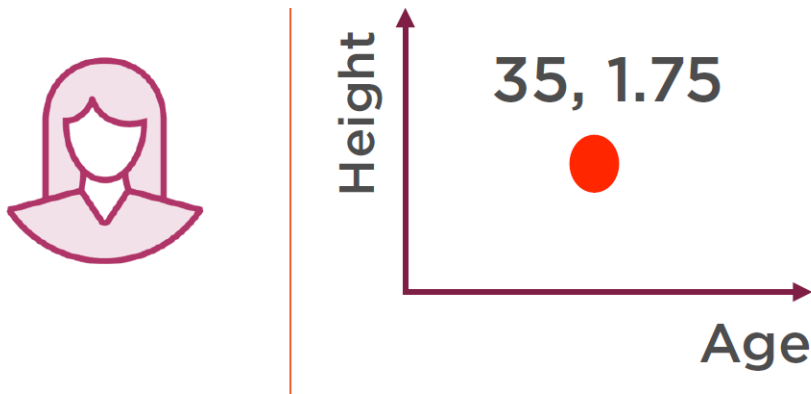
The model has to be set up right to learn structure in the data

Clustering:

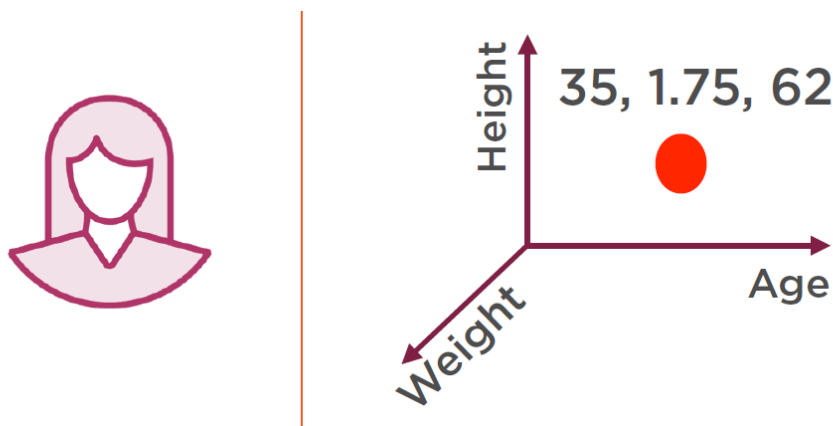
Clustering is a popular unsupervised learning technique which helps find patterns in the underlying data. Clustering does not use any Y variables or labels on the data. It looks at the data structure itself. Let's first understand how clustering works and how we can use it with any kind of data. The important principle behind clustering is that anything can be represented by a set of numbers. Whether it's an object, a person, a document, or a webpage, all of these can be represented in some numeric form. Let's consider a person. A person is of a certain age that can be represented on a number line.



A person may be of a certain height. All you need to do then is to represent this information in two dimensions. The person is a point on this plane.

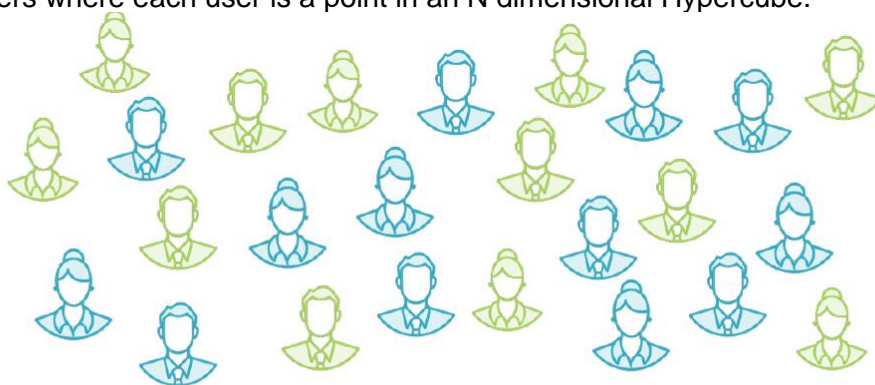


Let's say you were to add a third dimension. A person has a certain weight. Now this individual is represented using three distinct pieces of data.



Now, assume you have a whole bunch of other information about this person, you could then use an N dimensional Hypercube to represent the set of N numbers. The basic principle is that all the information about a particular person can be represented in **numeric form**.

Now let's take the example of Facebook users. Facebook users have certain characteristics. Different users have different characteristics. Hypothetically, you could have a set of Facebook users where each user is a point in an N dimensional Hypercube.



A set of points, each representing a Facebook user

Clustering involves finding groups of people within this data who have the same characteristics. What those characteristics are can differ. It could be that they like the same music, they went to the same high school, anything. Clustering results in the formation of groups within the data where people within the same group are similar. People who are in different groups are different.



Let's say you were to change the features based on which you performed clustering. You could end up with a completely different set of groups. One of these groups could be parents with children under five. Another group could be parents of teenagers.

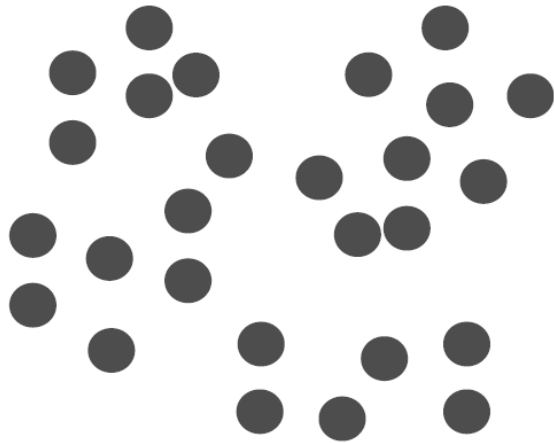


If you think about the Facebook example, clustering of users is important because then you can target specific ads to specific groups. So, how well did your algorithm cluster the underlying data? This can be measured by considering the distance between individual points in a cluster. Smaller this distance, better the clustering. The distance between users in a cluster is a measure of how similar the users are, and the goal of clustering is to maximize intra-cluster similarity. In addition, we also want our clustering algorithm to ensure that the distance between users who are in different clusters is as large as possible. We want to minimize inter-cluster similarity. A good clustering algorithm will try and achieve both objectives to the best of its ability, maximize intra-cluster similarity, and minimize inter-cluster similarity.

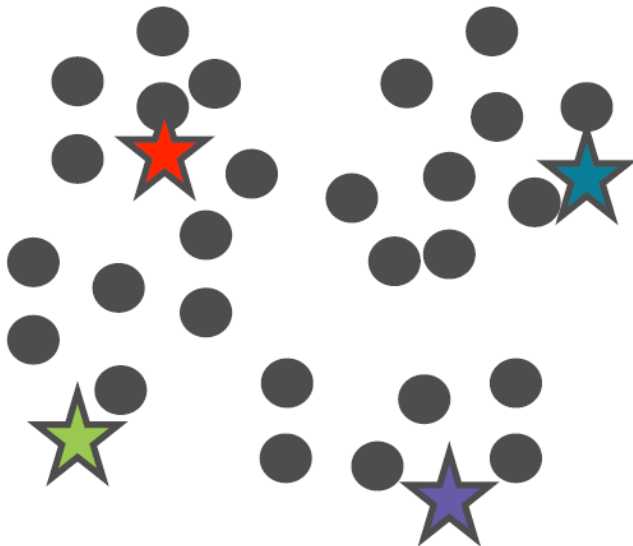
K - Mean Clustering:

One of the most popular machine learning algorithms to perform clustering which allows us to maximize inter-cluster similarity and minimize intra-cluster similarity is the K-means clustering algorithm.

Let's say we have several points in two-dimensional space. This can be extended to N dimensional space. We'll work with two dimensions because that's simpler to visualize. We start off by initializing K centroids or the K-means of the clusters.



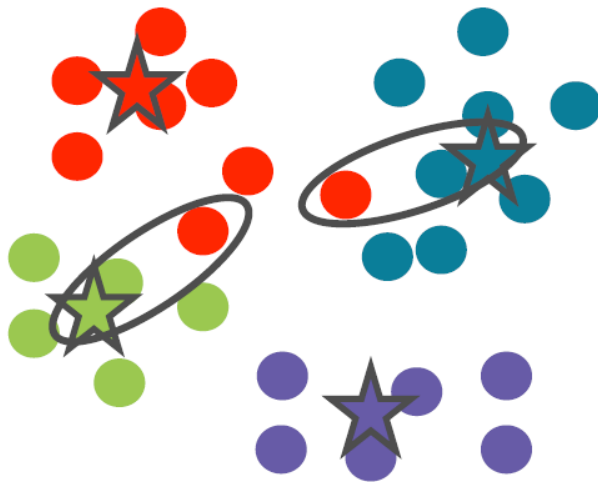
In K-means clustering you must specify this value of **K** up front, how many clusters you want your data to be divided into. Let's assume 4 different centroids in our example.



Once you have K cluster centers assign each point to a particular cluster. To do this, we calculate the distance between every point and every cluster center. A point is assigned to that cluster whose cluster center it is the closest to.



Once you've assigned all the points, you'll see cluster set up like this. At this point in time, use the existing points in each cluster to recalculate the mean for each cluster. Once the cluster centers have been recalculated, you'll find that certain points will move to another cluster.



We recalculate the distance from all cluster centers and reassign the points. This process of recalculating the means of each cluster and then reassigning the points once the new means have been calculated continues till the points reach their final position. When the cluster centers and the corresponding points don't move anymore, that's when the algorithm has converged. After convergence, you can think of every cluster being represented by a single point and this point is the reference vector. This reference vector is the center of the cluster and because it is calculated as an average of all points that belong to a cluster it's called the centroid of the cluster.

Step wise K - Means Clustering:

Step 1:

Choose the number of clusters k

- K should be less than total number of datapoints (m)
- You can visualize the data to decide the number of K .

Step 2:

Select K random points from the data as centroids for each cluster. Let's say we want to have 2 clusters, so K is equal to 2 here.

Step 3:

Find distance of all the datapoints with all the centroid using any point-to-point distance formula e-g Euclidean distance measure. Add the point to the cluster having minimum centroid distance.

Step 4:

Recompute the centroids of newly formed clusters. Now, once we have assigned all the points to either cluster, the next step is to compute the centroids of newly formed clusters.

- You can find the new centroid of a cluster by taking mean of all its datapoints w.r.t. each feature. Let's say we have 2 features per datapoint. Take mean of 1st feature and then 2nd feature.

Step 5:

Repeat steps 3 and 4 The step of computing the centroid and assigning all the points to the cluster based on their distance from the centroid is a single iteration. But wait – when should we stop this process? It can't run till eternity, right?

Stopping Criteria for K-Means Clustering

There are essentially three stopping criteria that can be adopted to stop the K-means algorithm:

1. Centroids of newly formed clusters do not change.
2. Points remain in the same cluster.
3. Maximum number of iterations are reached.

Task:

- Download fruit dataset, previously provided.
- Remove the columns containing class labels to make it unlabeled dataset.
- Implement K-means clustering from scratch and apply it on the dataset.