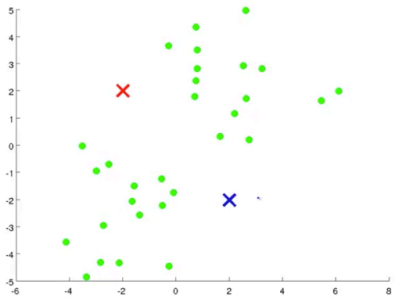
Clustering (Unsupervised Learning)

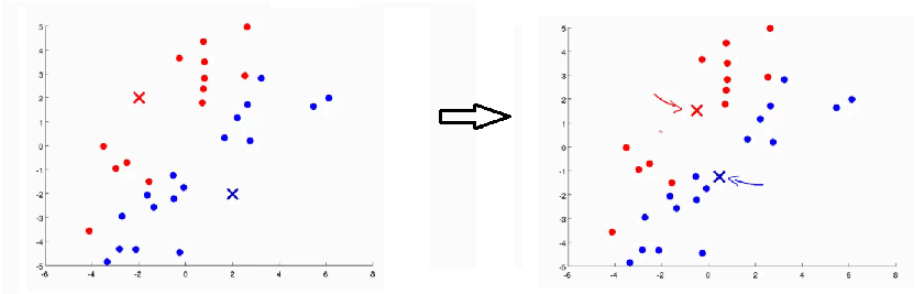
Clustering is an unsupervised learning, meaning we have unlabeled dataset. In these types of data structure, we are trying to find structure. In clustering, we group data based on data features. This has many applications in real world such as organizing computer clusters, astronomical data analysis, social network, and market segmentation.

K mean algorithm

The most popular algorithm in unsupervised learning. It is best to describe it by plots. Imagine we have the green datasets. Here is how K-mean works: (*k-mean is an iterative algorithm*)



1. Randomly allocate the cluster centroids as cluster (shown in blue and red)
2. Cluster assignment step: In this step, we go through all of our example and assign them to the blue or red line depending on their distance from cluster centroids: $min_c \|x^i - \mu_k\|^2$
3. Move centroid step: we move the centroid step to the average of the corresponding assigned data points.

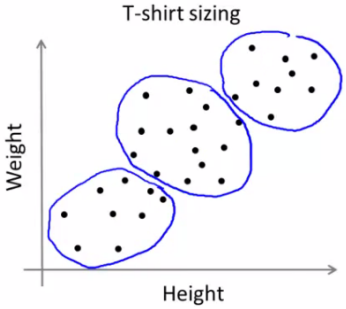


4. Repeat 2 and 3 until it converges

K in. k-mean is the number of clusters in the dataset.

K mean for non-separated clusters

Sometimes the clusters might not be well defined. For example, we have a T-shirt sizes based on size and height. It's not very obvious how to do clustering. In this case, we build products which suits our needs in subpopulations.



Optimization objective

k-mean has an optimization or cost function we are trying to solve. Optimization help us to debug the algorithm better. Let's define some parameters first, then we can define the cost function.

Clustering (Unsupervised Learning)

→ $c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently assigned

→ μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$)

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned

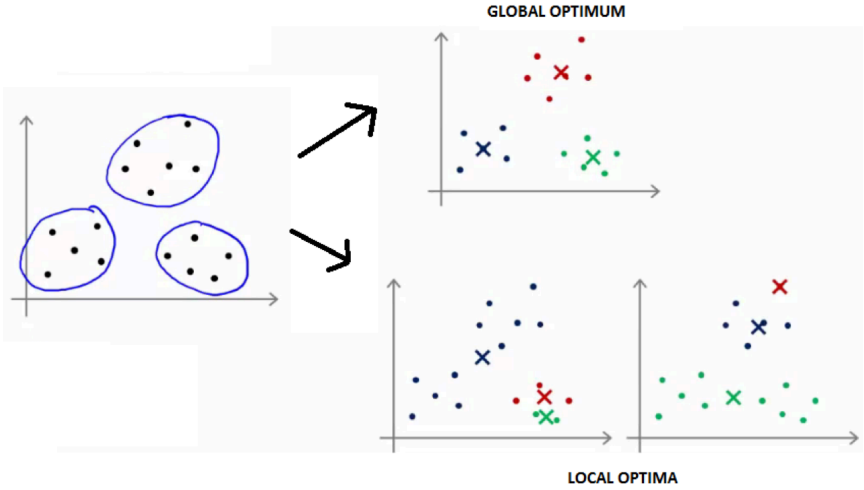
Optimization objective:

$$J(c^{(1)},\dots,c^{(m)},\mu_1,\dots,\mu_K) = \frac{1}{m}\sum_{i=1}^{m}||x^{(i)} - \mu_{c^{(i)}}||^2$$

$$\min_{\substack{c^{(1)},\dots,c^{(m)},\\ \mu_1,\dots,\mu_K}} J(c^{(1)},\dots,c^{(m)},\mu_1,\dots,\mu_K)$$

One of questions that comes to mind is how to initialize the values randomly for the centroid clusters. First of all, we need to set number of clusters less than number of examples (k<m), then randomly pick k training examples and set them as cluster centroids.

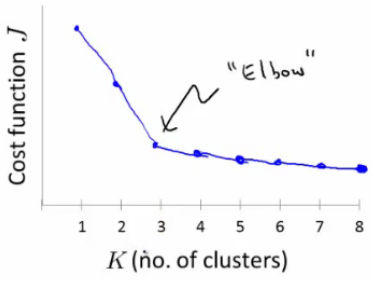
K-mean may end up to different solution depends on initialization setup. Therefore, there is a risk of local optima.



In this regard, we may do multiple random initialization (50- 10,000 times) and see the result. Many result most likely will end up to global optimum. Finally, we pick the one with lowest cost function.

Number of clustering

Elbow method: as k increases, the cost function should decrease. If we plot the K vs J, we should be able to see the elbow on the graph. However, normally we are not able to see the elbow clearly. Therefore, this method is not real helpful in real world dataset.



Visualization: Like the T-shirt example how to choose the k-mean based on what serves our needs the best.

Clustering (Unsupervised Learning)