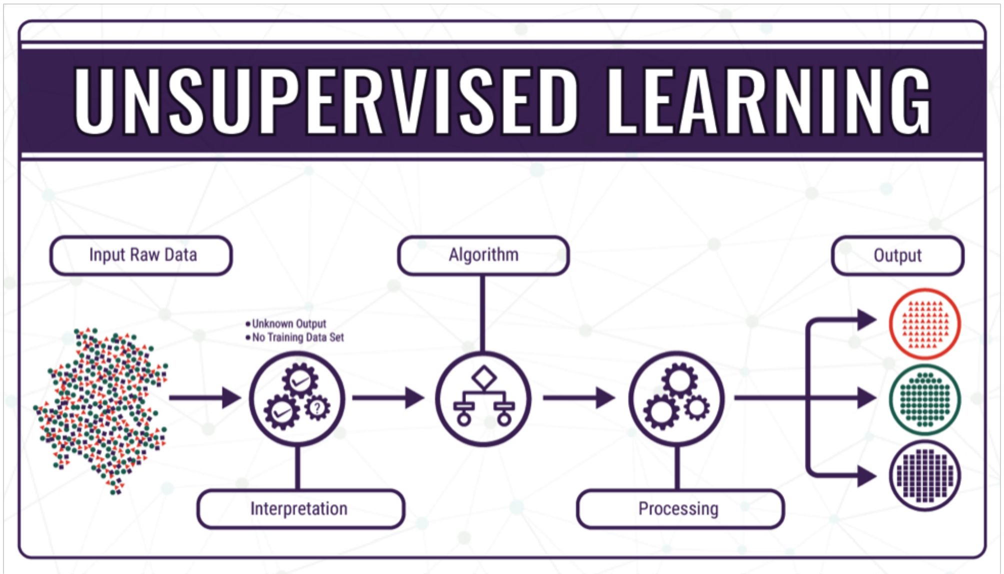
SMAI Scribes - K-Means

Anush Mahajan(20171020), Deepti Dwivedi(2019201027), Gaurav Chaudhari(2019201045)

February 15, 2020

Unsupervised Learning

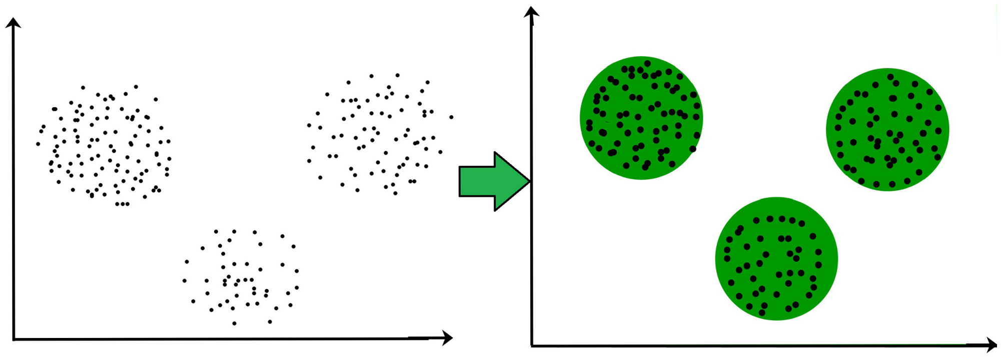
Unsupervised learning is a type of learning in which the algorithm need to draw inferences on it's own as no labelled input is given. It tends to be more challenging than supervised learning as you have no clear objective for the analysis. It is not feasible to do a performance evaluation in unsupervised learning and hence makes the result ambiguous. Even so it has many real life applications. It is used in bioinformatics for sequence analysis,in computer vision for object recognition and in data mining for sequence and pattern mining.



We are going to focus on clustering and specifically k-means clustering.

Clustering

Clustering refers to a broad set of techniques to find subgroups in a dataset.Clustering organizes data into groups such that there is high intra-group similarity and low inter-group similarity. The groups found depend on which features you use while making clusters. For example you can cluster breast cancer patients with the grade of tumour, age of the patients or if the cancer has metastasised or not. The figure below shows the clusters within the data



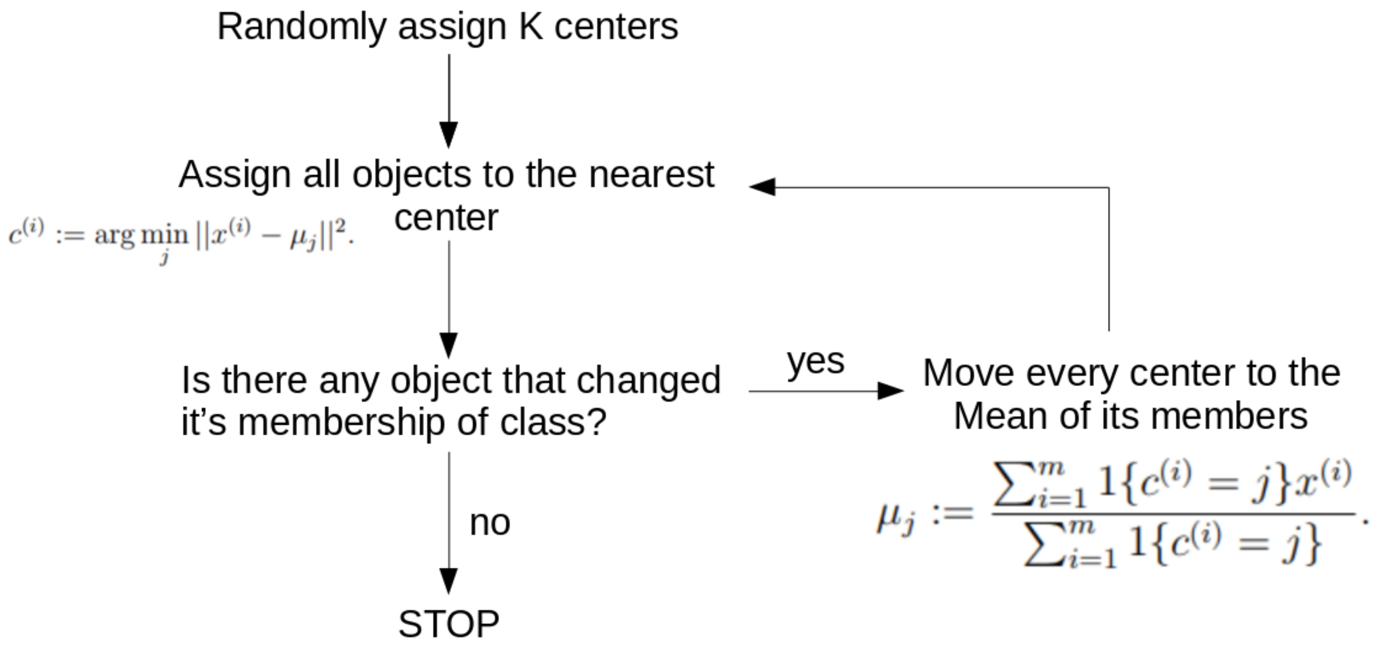
There are different types of clustering methods such as Hierarchical clustering ans K-Means clustering. We will focus on K-Means clustering

K-Means Clustering

In K-means we have assume that we know the number of clusters in the data which is K. The K is often a guess. We then proceed to represent the cluster using a suitable statistic of the cluster in this case the mean (Hence the name K-Means). To begin we choose a value of K and randomly choose a centroid for each cluster. We then implement the following two steps:

1. Assignment Step : Assign each data point to it's nearest centre
2. Update Step: Update the centroid as being the centre for their respective observation.

How K-means works


Randomly assign K centers
Assign all objects to the nearest center
$c^{(i)} := \arg\min_j ||x^{(i)} - \mu_j||^2.$
Is there any object that changed it's membership of class?
yes
Move every center to the Mean of its members
$\mu_j := \frac{\sum_{i=1}^m 1\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^m 1\{c^{(i)} = j\}}.$
no
STOP

Why K-means works

- We run the algorithm until we find at-least one object changing it's membership of class. An object changes it's membership only when it is nearer to another cluster center (after the centers were moved to the mean of members).
- A scope for improvement is seen when there is some reduction in distance between an object and center of a cluster. Hence, we define the objective function we are trying to minimize as.

$$J = \sum_{k=1}^{K} \sum_{i=1}^{n_k} \left\| x_{ki} - \mu_k \right\|^2$$

- J denotes the sum of squared distances between a data point and it's assigned cluster center. The algorithm, in finite number of steps, is actually minimizing the distance constantly so as to get J to minimum possible value.
- K-Means doesn't necessarily converge to global optimum. The initial positions for the centroids determine which locally optimum solution algorithm reaches. i.e. if you start with different positions, you can easily end up in different final places.

Algorithm K-Means

K- Means algorithm is an iterative algorithm that tries to partition the dataset into K predefined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. It tries to make the inter-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

The way k- Means algorithm works is as follows:

1. Specify number of clusters K.

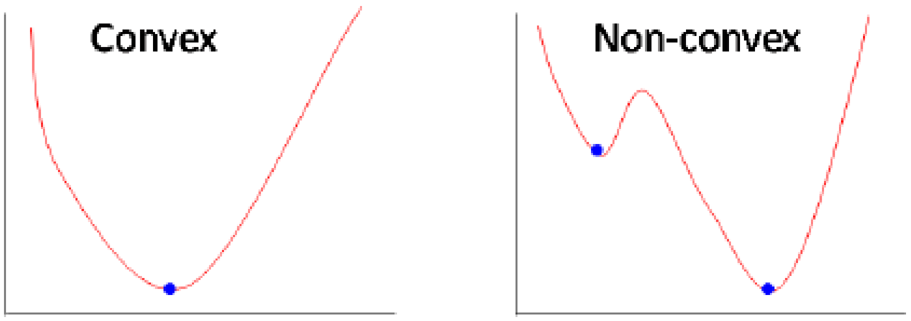
2. Initialize centroid by first shuffling the dataset and then randomly selecting K data points for the centrid without replacement.

3. keep iterating until there is no change to the centroids. i.e. assignment of data points to clusters isn't changing.

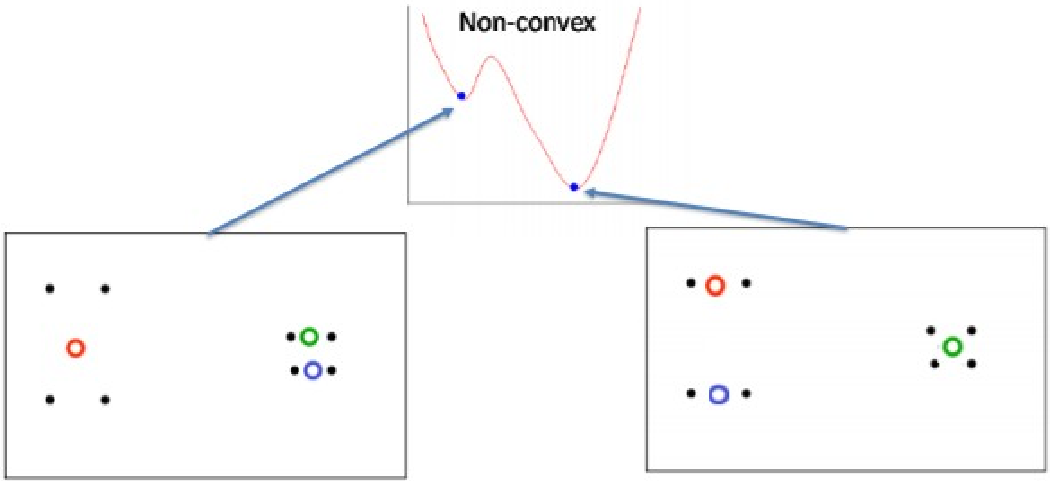
    - Compute the sum of the squared distance between data points and all centroids.
    - Assign each data point to the closest cluster (centroid).
    - Compute the centroids for the clusters by taking the average of all data points that belong to each cluster.

Convex and Non-Convex functions

In terms of cost function with a convex type you are always guaranteed to have a global minimum, whilst for a non convex only local minima.

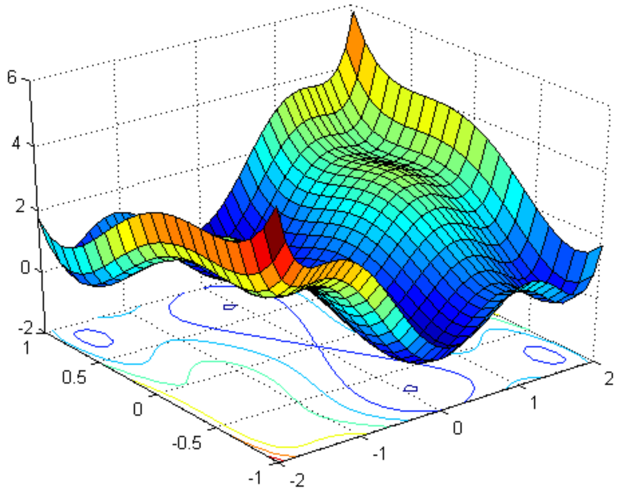


## **Objective function for k-means is non-convex.**



Let $\vec{x_i}$, i=1,2,...,n be the data points and $\vec{\mu}_i$, j=1,2,...,k be the k mean values. minimize

$$\sum_{i=1}^{n} \min_{j=1...k} ||\vec{x_i} - \vec{\mu_j}||^2$$

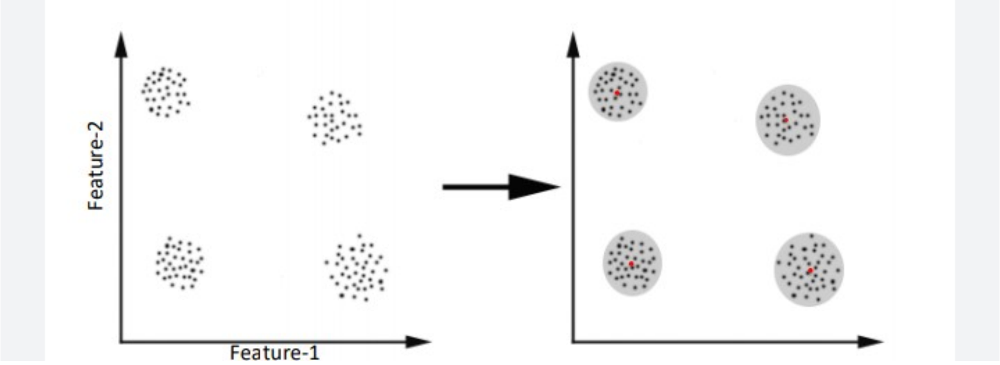


K-means++: Improving K-means Initialization

- Common way to improve k-means - smart initialization!
- General idea - try to get good coverage of the data.
- K-means++ Algorithm:

1. Pick the first center randomly.
2. For all points $x^{(n)}$ set $d^{(n)}$ to be the distance to closest center.
3. Pick the new center to be at $x^{(n)}$ with probability proportional to $d^{(n)2}$
4. Repeat steps 2+3 until you have k centers

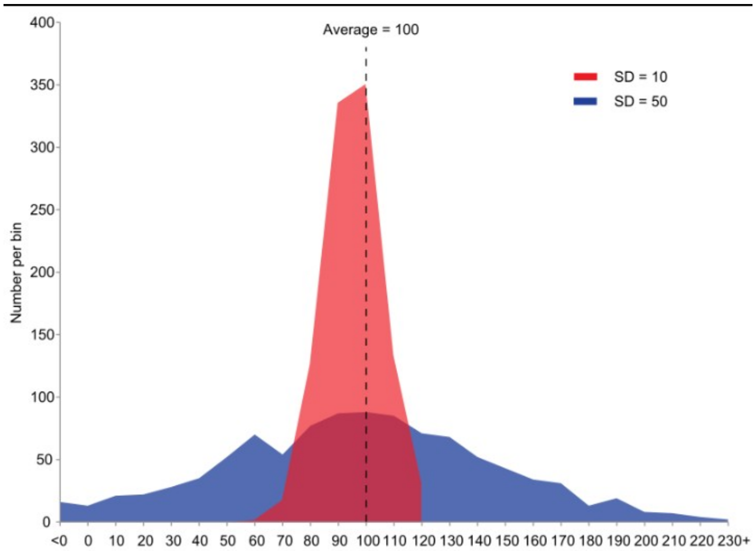
Perspective: Clustering as a 'summary' of input data



Output of k-means = 'centers' ... but only these are not sufficient to summarize

Mean, Standard Deviation and Variance (1-D)

In statistics, dispersion (also called variability, scatter, or spread) is the extent to which a distribution is stretched or squeezed.Common examples of measures of statistical dispersion are the variance, standard deviation etc.



above is the example of samples from two populations with the same mean but different dispersion. The blue population is much more dispersed than the red population.

Formula for Sample Variance: $\sigma^2 = \frac{\displaystyle\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}$

Formula for Sample Standard Deviation: (encodes spread with respect to mean) $\sigma = \sqrt{\frac{\displaystyle\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}}$

Covariance

covariance is a property of two random variables, which is a rough meansure of how much chnging one affects the other.

Variance:

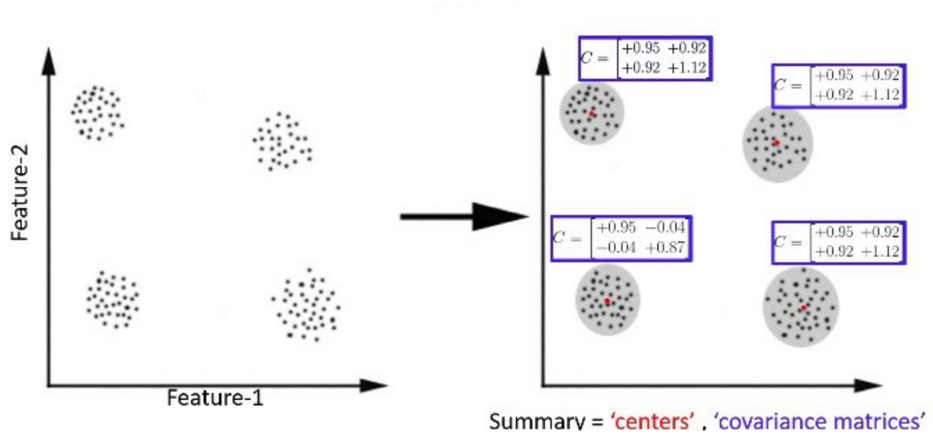
$\sigma^2 = \frac{\displaystyle\sum_{i=1}^{n}(x_i - \mu)^2}{n}$

Covariance:

$cov_{X,Y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{N-1}$

$cov_{Ma,Mb} = \frac{\sum_{i=1}^{m}(q_{i,a} - \bar{q_a})(q_{i,b} - \bar{q_b})}{m}$

Perspective: Clustering as a 'Summary' of input data version-2


Feature-2
Feature-1
$C = \begin{bmatrix} +0.95 & +0.92 \\ +0.92 & +1.12 \end{bmatrix}$
$C = \begin{bmatrix} +0.95 & +0.92 \\ +0.92 & +1.12 \end{bmatrix}$
$C = \begin{bmatrix} +0.95 & -0.04 \\ -0.04 & +0.87 \end{bmatrix}$
$C = \begin{bmatrix} +0.95 & +0.92 \\ +0.92 & +1.12 \end{bmatrix}$
Summary = 'centers' , 'covariance matrices'

Summary: K-Means

- **Strength:**

  1. Simple, easy to implement and debug

    2. Intuitive objective function: optimizes intra-cluster similarity
    3. Relatively efficient: O(tkn), where n is number of objects, k is the number of clusters, and t is the iterations. Normally, k,t¡¡n

- **Weakness:**

    1. Applicable only when mean is defined, what about the categorical data?
    2. Often terminates at a local optimum. Initialization is important.
    3. Need to specify K, the number of clusters, in advance
    4. Unable to handle noisy data and outliers
    5. Not suitable to discover clusters with non-convex shapes

- **Summary:**

    1. Assign members based on current centers
    2. Re-estimate centers based on current assignment