Unsupervised Learning

Outline

- Clustering
  - Introduction
  - Different Types of Clustering
- Kmeans Clustering
- Case Study
  - Technical Support Data Clustering
  - Retail Customers

We all generate tremendous amount of data

. . .

Unlabeled Data

Photos

Social Media

Patient Data

Survey Data

How do we make sense of Unlabeled Data

Label the Data

Very Expensive

Lacks new Learnings

How to learn from **Unlabeled Data?**

Clustering

Unlabeled Data

Cluster #1

Cluster #2

Unlabeled Data

Vegetables

Fruits

Clustering

Grouping of Similar things

Clustering

No concept of training or test set

1. Clustering is primarily an exploratory technique to discover hidden structures of the data, possibly as a prelude to more focused analysis or decision processes
    a. A way to decompose a data set into subsets with each subset representing a group with similar characteristics
    b. When we cluster observations we seek to partition them into distinct groups such that objects in the same group are more similar to each other in some sense than to objects of different groups
    c. The groups are known as clusters and each cluster gets distinct label called cluster id, the centroid of the cluster, and inertia

2. Clustering is often used as a lead-in to classification. Once the clusters are identified, labels can be applied to each cluster to classify each group based on its characteristics

3. Can also be used to decide whether there is a need for separate models representing each cluster (will be so if clusters are very different on the attributes), or a single model can be reliable. If separate models are required, will the attributes have equal importance

4. Clustering helps simplifying the data representation through the centroid methods.

Some Applications of Clustering

Some specific applications of k-means are image processing, medical, customer segmentation, finance

a. **Image processing** : used to cluster of pixels representing objects in each frame. The attributes of each pixel can include brightness, color, and location, the x and y coordinates in the frame. Successive frames are examined to identify any changes to the clusters. These newly identified clusters may indicate unauthorized access to a facility.

b. **Medical** : Patient attributes such as age, height, weight, systolic and diastolic blood pressures, cholesterol level, and other attributes can identify naturally occurring clusters under various health conditions

c. **Customer segmentation** : Cluster customers on basis of frequency of purchase, recency of purchase, value of purchase and look for common attributes among high value customers. Target all potential customers who have similar attributes

d. **Trading**

How to Cluster Unlabeled data

Clustering Types

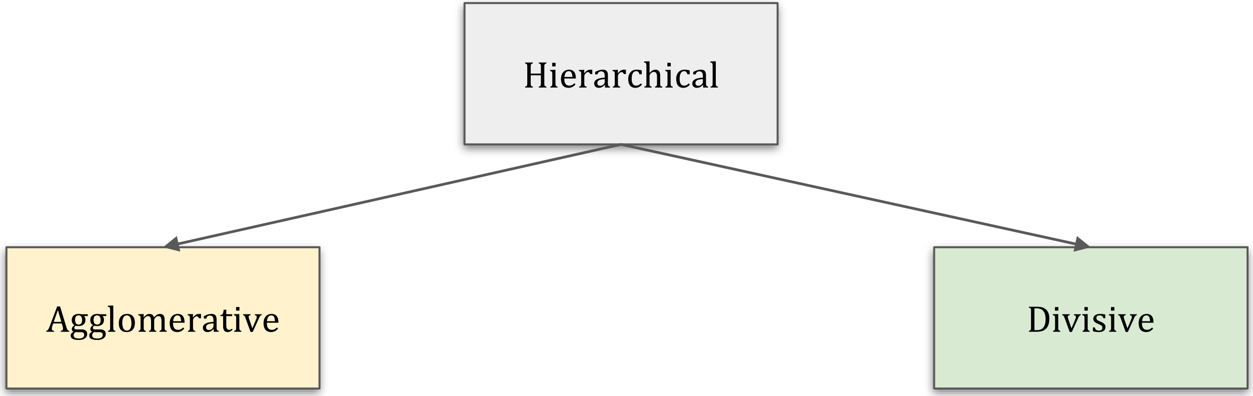
1. Two broad categories of clustering include hierarchical (agglomerative, divisive) and non hierarchical

2. Hierarchical clustering
   a) Agglomerative clustering algorithm uses a bottom-up approach and merges smaller clusters into larger ones
   b) Divisive clustering uses top-bottom approach to break a large cluster into smaller clusters

3. Non-hierarchical / partitional clusters are formed on assumption that the clusters are disjoint and there is no hierarchical relation between them. K Means is an example

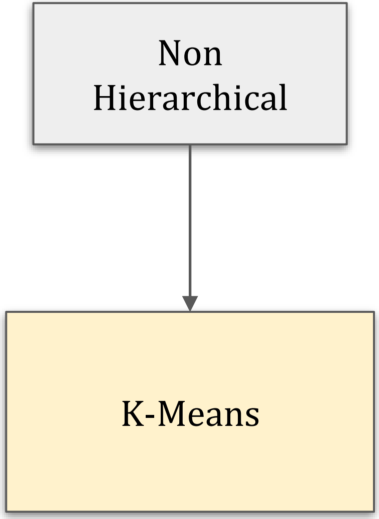
Hierarchical

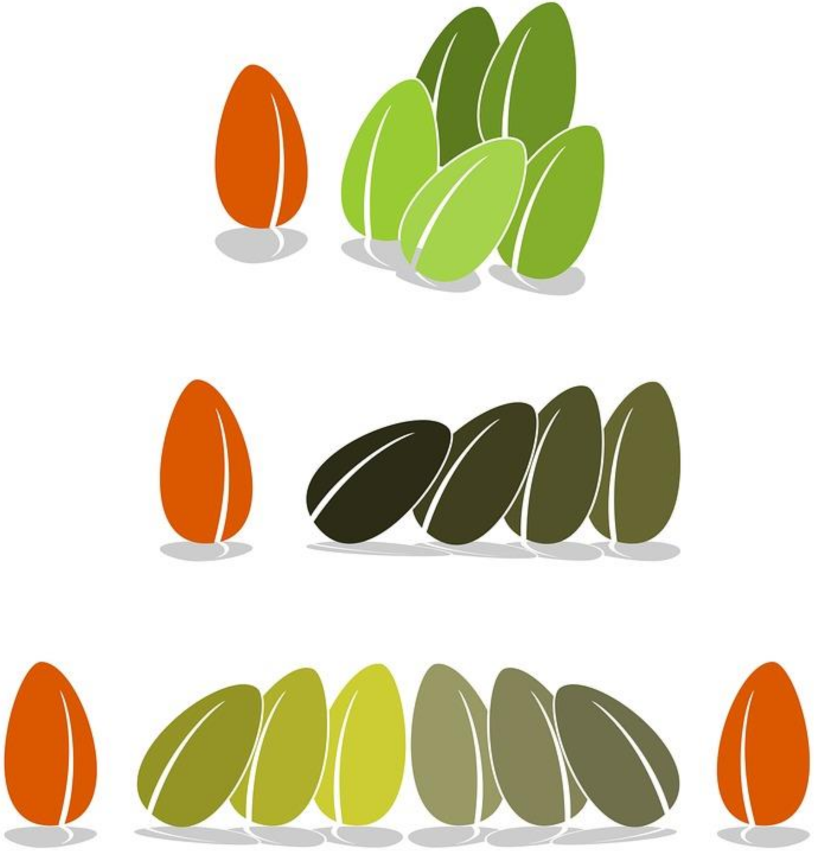
Non Hierarchical


Hierarchical

Agglomerative

Divisive


Non Hierarchical
K-Means



How do we find similar items?

Clustering – Distance Calculations

1. Irrespective of the clustering algorithm, we need a way of defining similarity/ dissimilarity

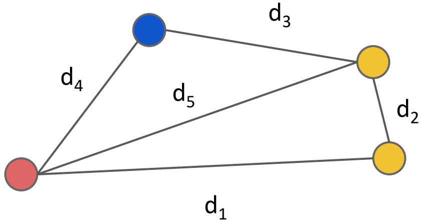
2. Central to all of the goals of cluster analysis is the notion of the degree of similarity (or dissimilarity) between objects being clustered

3. Clustering method attempts to group the objects based on the definition of similarity supplied to it. The definition uses distance calculation functions for the same

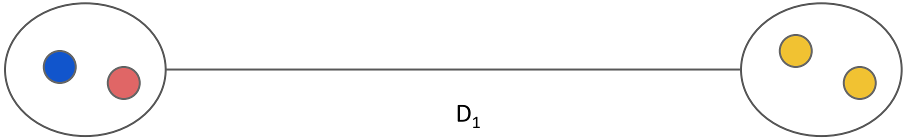
4. We also need a way to define and calculate distance between clusters of objects

5. The lesser the distance, more similar the objects are and more suited to form a larger cluster

6. There are many ways of calculating distance between two points i.e. if d = f(x,y) then there are many ways in which f can be implemented



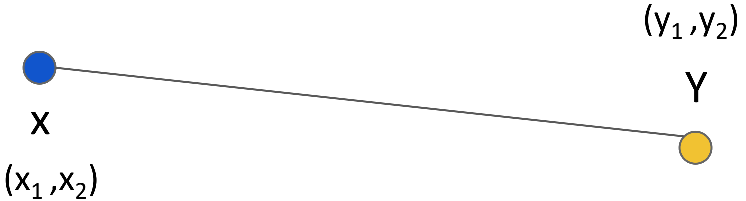
Find distance between data points



Find distance between clusters

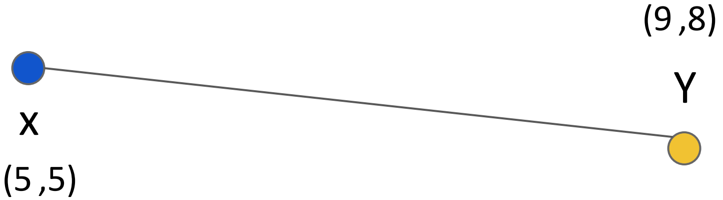
How to calculate distance using Math?

1. Euclidean Distance



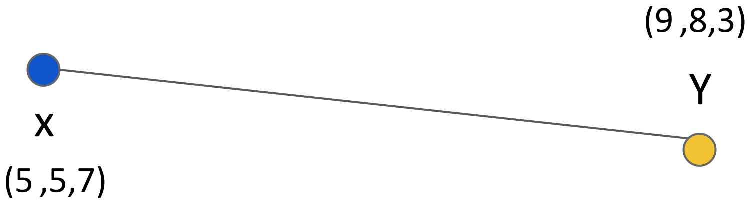
$$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

1. Euclidean Distance



$$d = \sqrt{(5-9)^2 + (5-8)^2} = 5$$

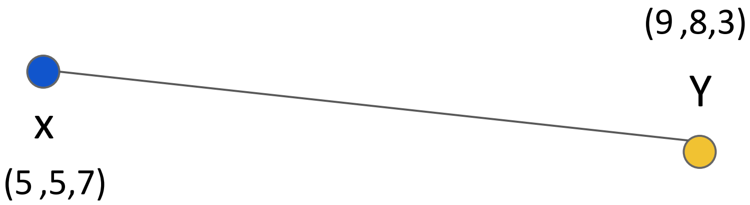
1. Euclidean Distance



$$d = \sqrt{(5-9)^2 + (5-8)^2 + (7-3)^2} = 6.4$$

L2 Norm

1. Euclidean Distance



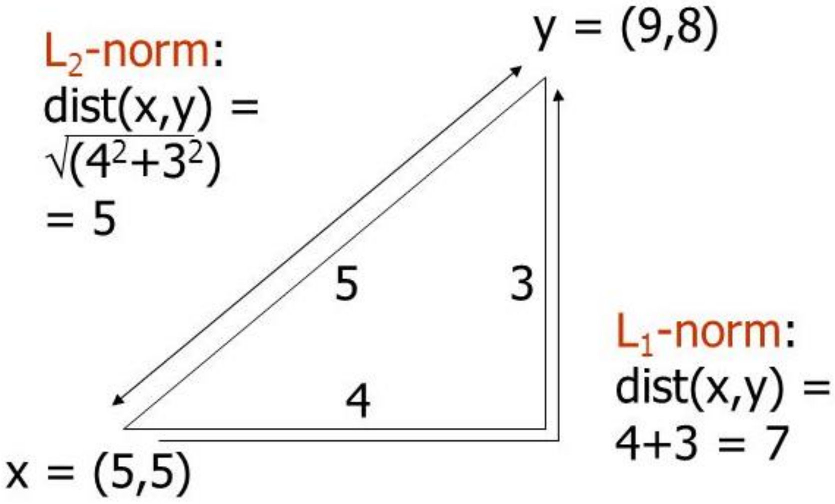
$$d = |5 - 9| + |5 - 8| + |7 - 3| = 11$$

L1 Norm or
Manhattan distance

Clustering – Distance Calculations

1. Euclidean Distance
   a. L2 norm : d(x,y) = square root of the sum of the squares of the differences between x and y in each dimension. The most common notion of “distance”. If there are two dimensions x and y, the distance between two points A and B is –

2. Manhattan / Taxi Distance
   a. L1 norm : sum of the differences in each dimension. Manhattan distance = distance if you had to travel along coordinates only



2. Jaccard Distance

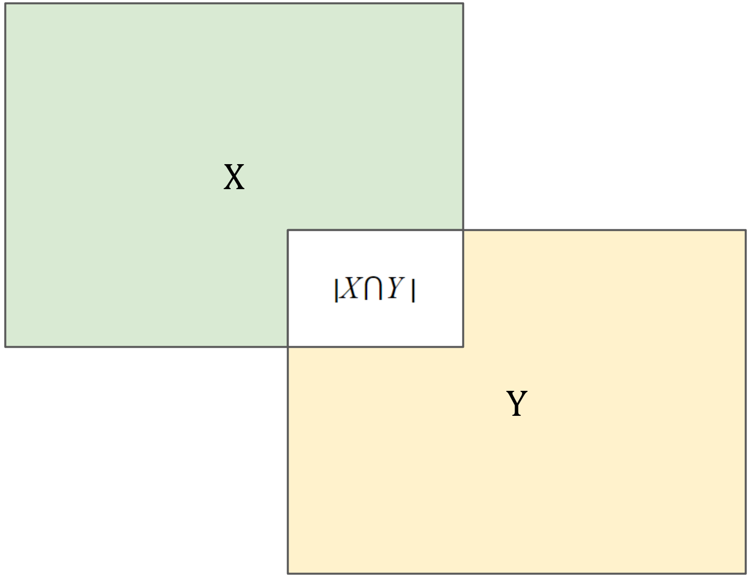
Intersection

$$jac(x,y) = \frac{|X \cap Y|}{|X| + |Y| - |X \cap Y|}$$

Union

$$d = 1 - jac(x,y)$$

2. Jaccard Distance



2. Jaccard Distance

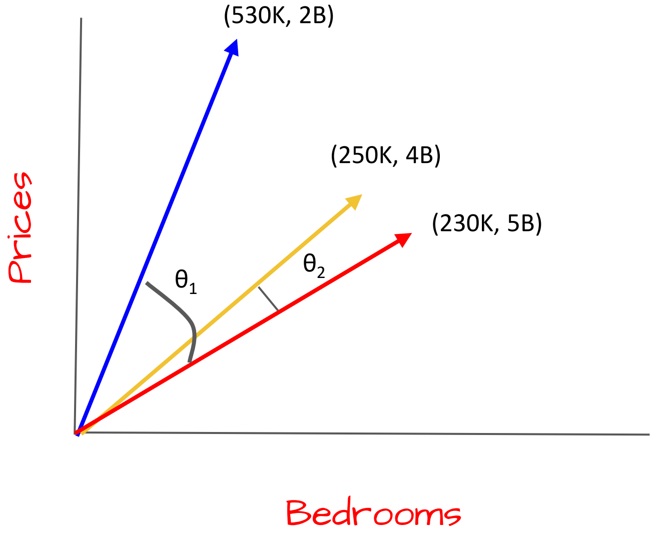
X	1	0	0	1	1	1
Y	0	1	1	0	1	0

$$|x \cap y| = 1, \quad |x| = 4, \quad |y| = 3$$

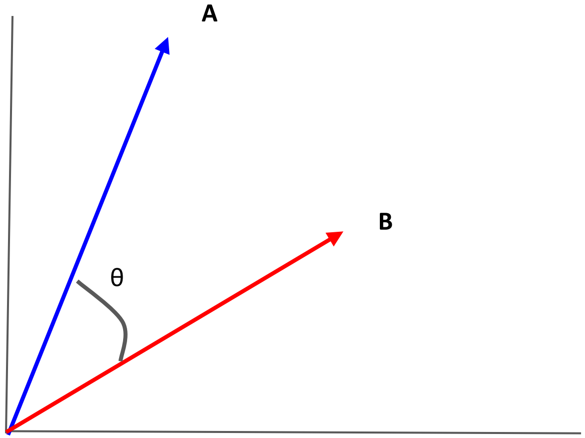
$$jac(x, y) = \frac{1}{4 + 3 - 1} = \frac{1}{6}$$

$$d = 1 - \tfrac{1}{6} = \tfrac{5}{6}$$

3. Cosine Distance



3. Cosine Distance



$$similarity = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|_2 \|\mathbf{B}\|_2} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

$$d = 1 - cos(\theta)$$

4. Edit Distance

Given two vectors, find minimum number of operations to transform one vector into another.

4. Edit Distance

Kitten -> Sitting

Kitten -> **S**itten

Sitt**e**n -> Sitt**i**n

Sittin -> Sittin**g**

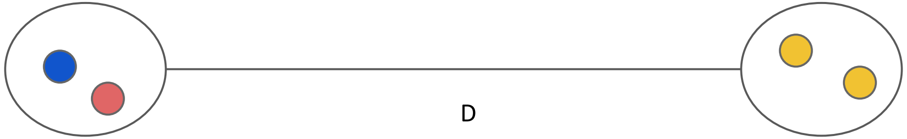
3 operations - substitution and addition

Clustering – Distance Calculations

1. Non Euclidean Distance
    a. Jaccard distance for sets = 1 minus ratio of sizes of intersection and union.
    b. Cosine distance = angle between vectors from the origin to the points in question.
    c. Edit distance = number of inserts and deletes to change one string into another
    d. Mahalanobis distance – takes into account the covariance between attributes (Ref: http://mccormickml.com/2014/07/22/mahalanobis-distance/)

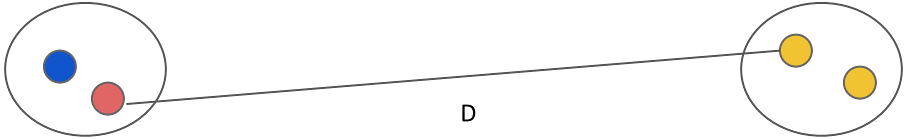
2. Euclidian Distance … Some points
    a. The measures computed in Euclidian methods are highly influenced by the scale of each variable
    b. Variables with larger scale have much greater influence over the total distance. This may or may not be good for clustering

Calculating distance between Clusters



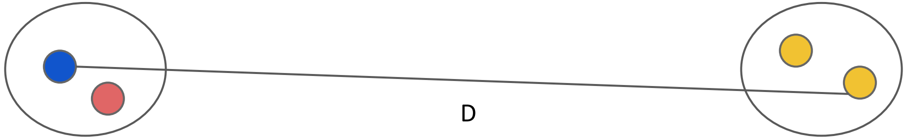
1. Minimum Distance

(Single Linkage)



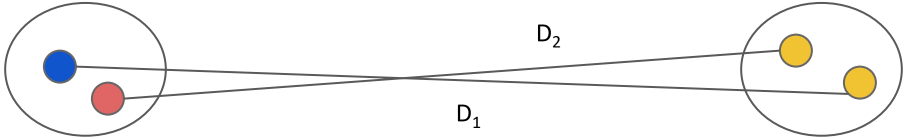
2. Maximum Distance

(Complete Linkage)



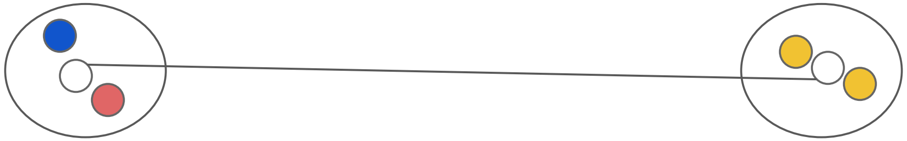
3. Average Distance

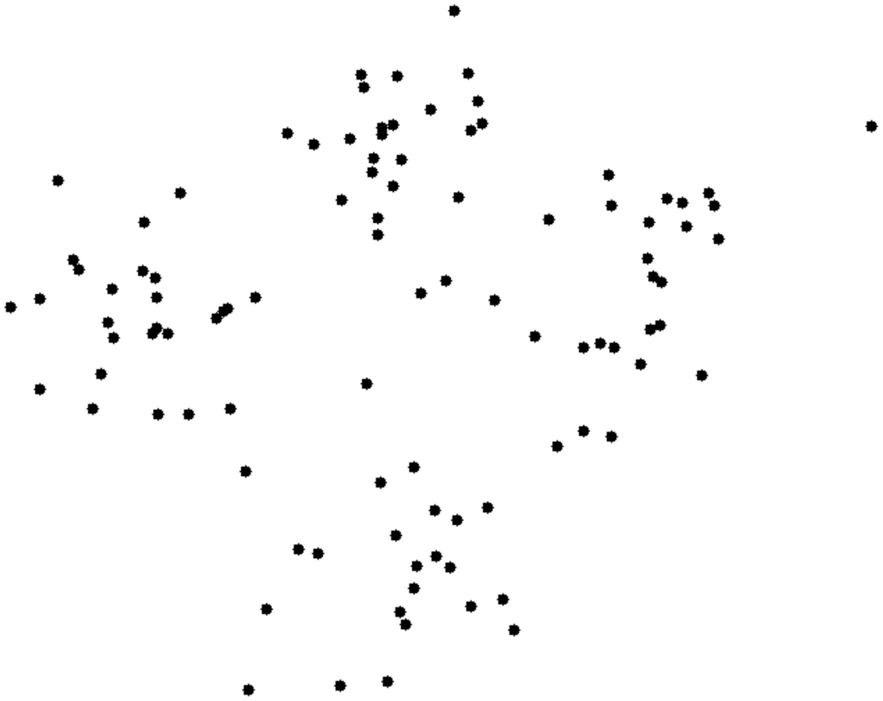
(Average Linkage)



$(D_1 + D_2)/2$

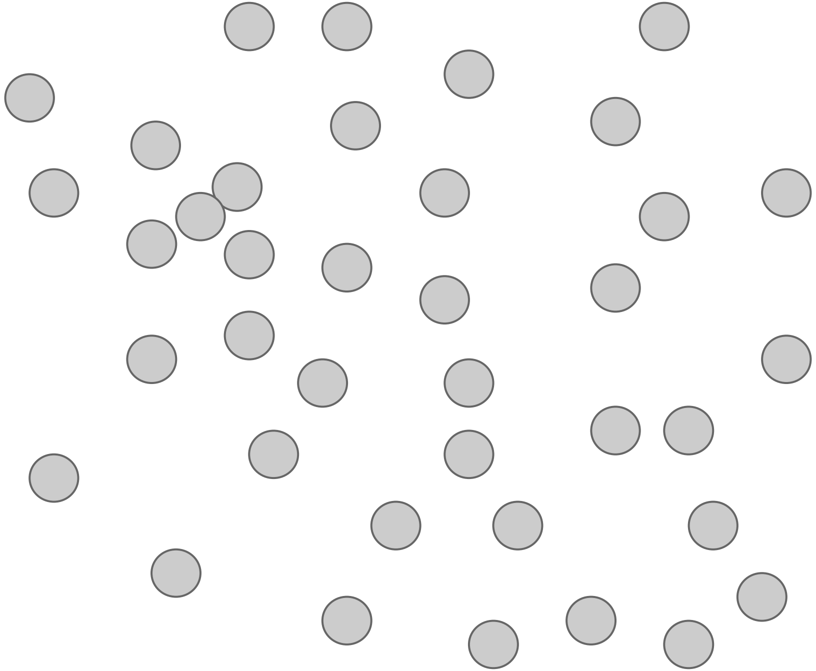
4. Centroid Distance



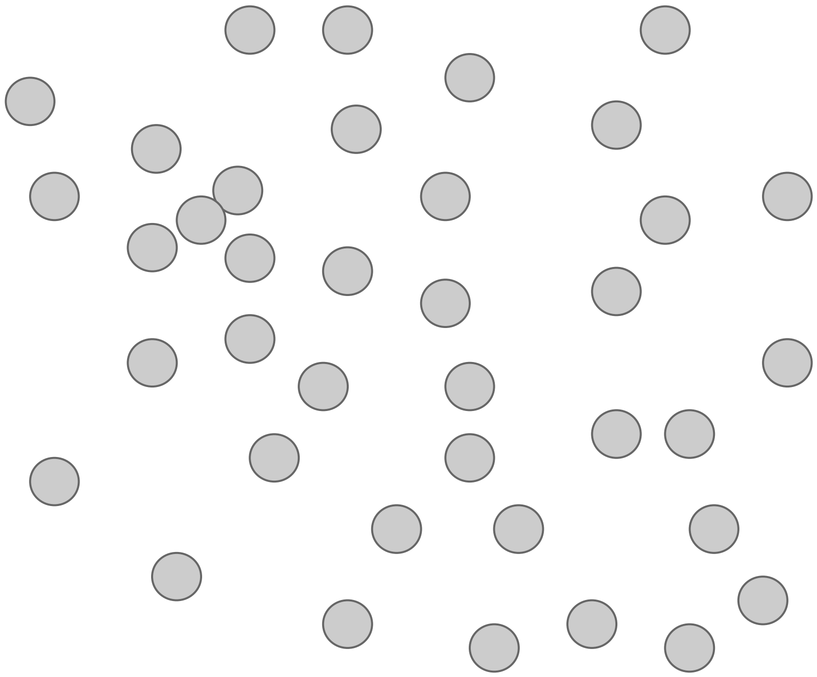


Credit Andrey A. Shabalin

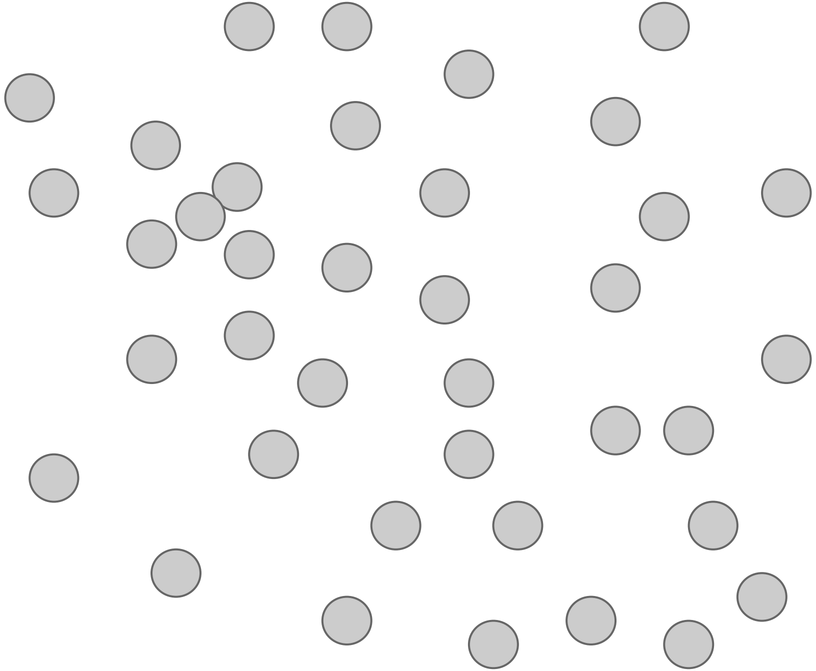
K-Means Clustering



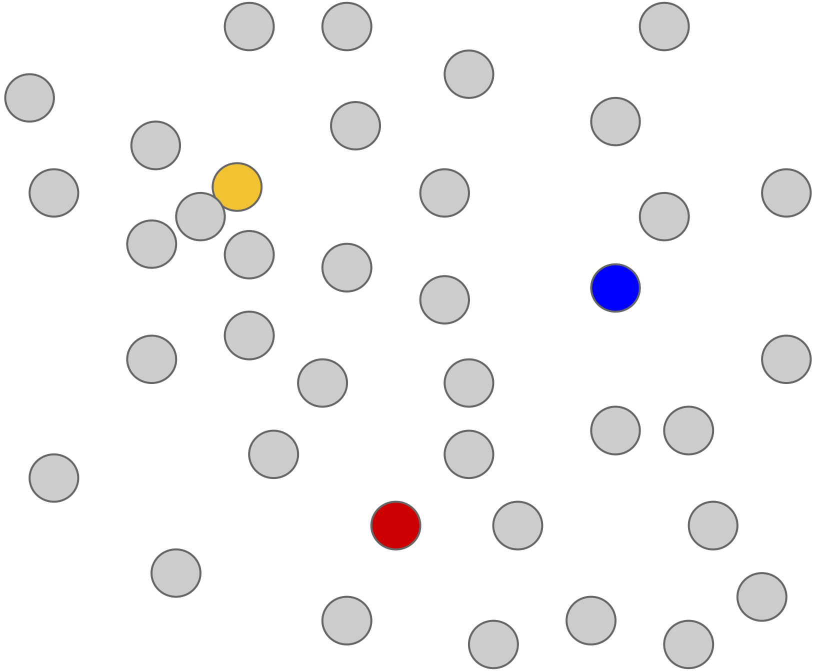
Define number of Clusters ‘K’



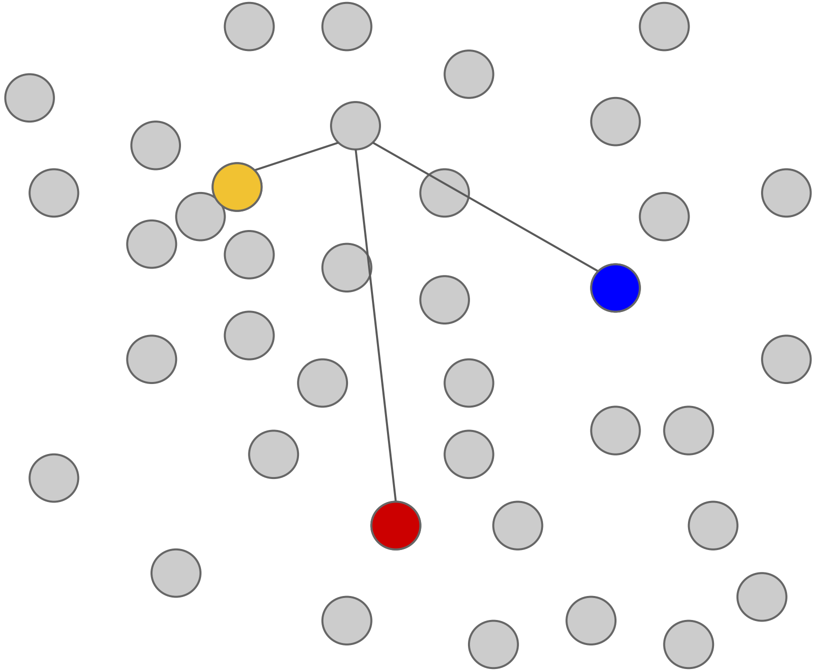
What will be center point for each Cluster *i.e* Centroid?



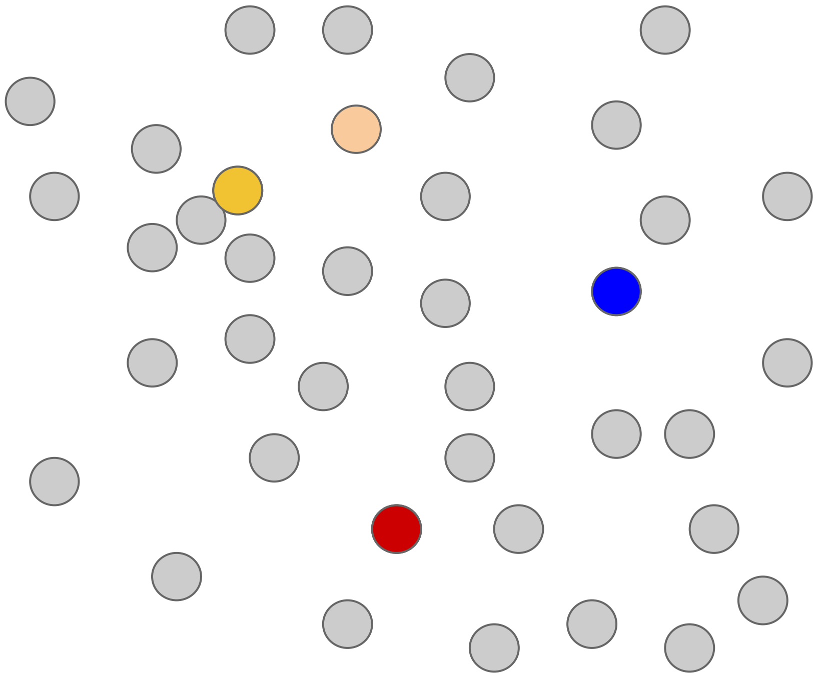
Randomly choose K data points



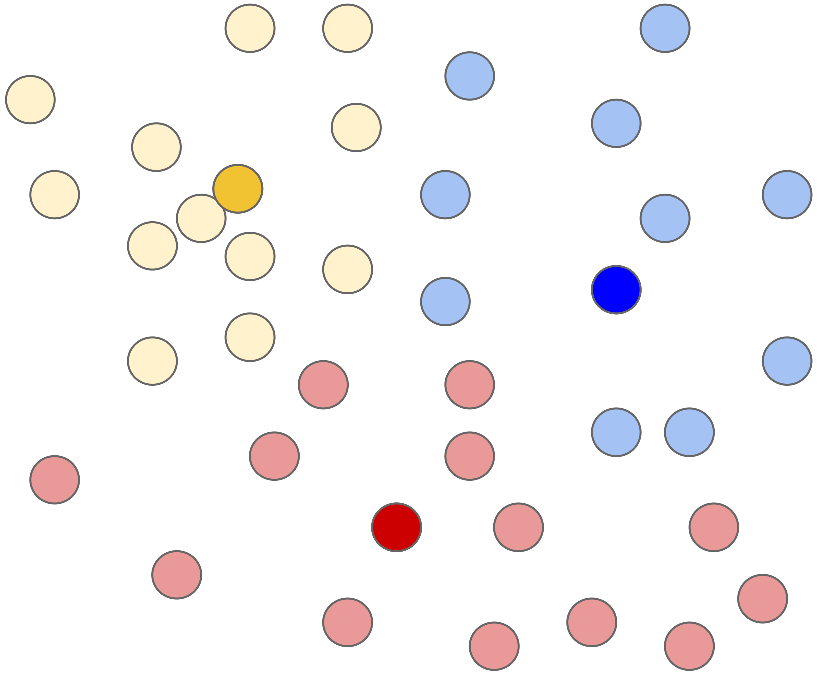
Randomly choose K data points



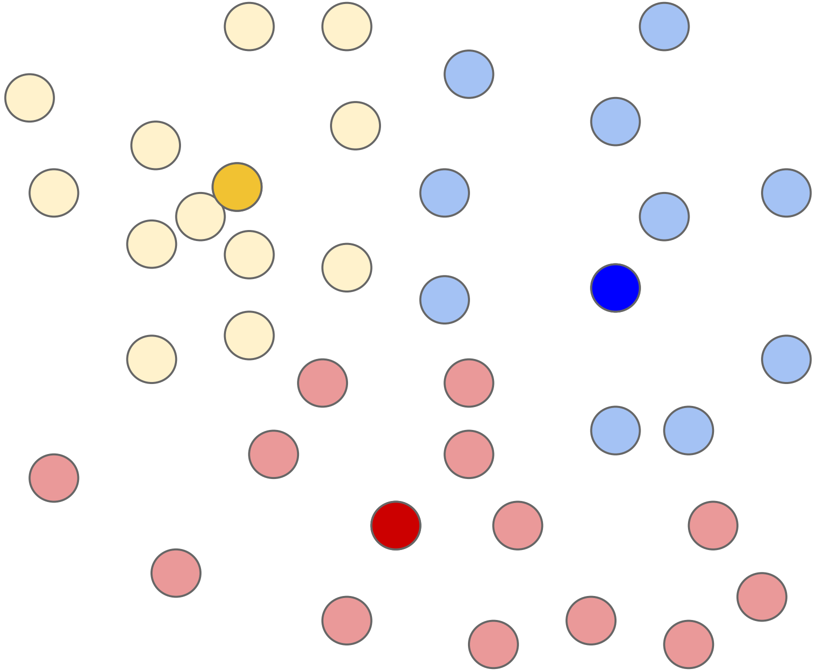
Assign each Data point to a Cluster



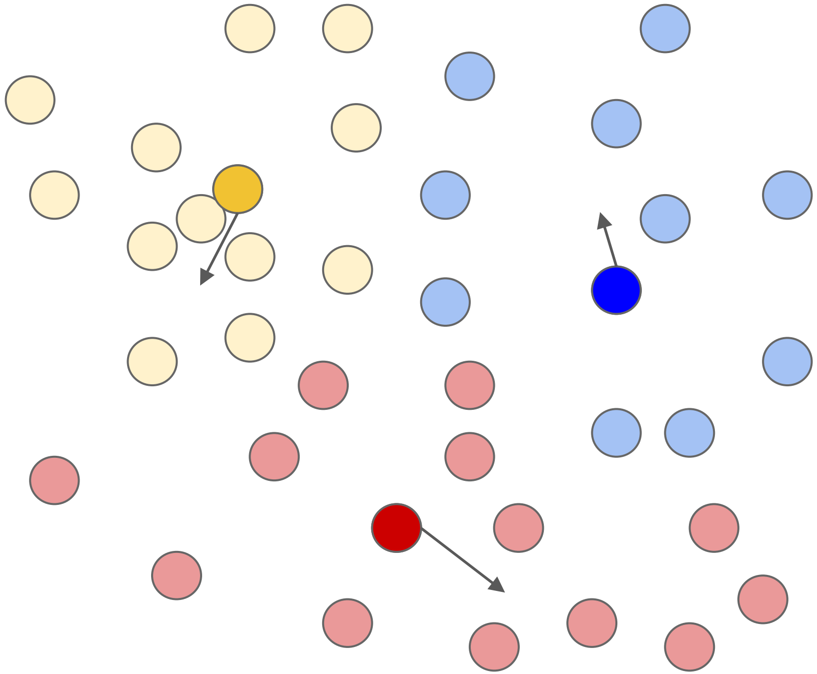
Assign each Data point to a Cluster



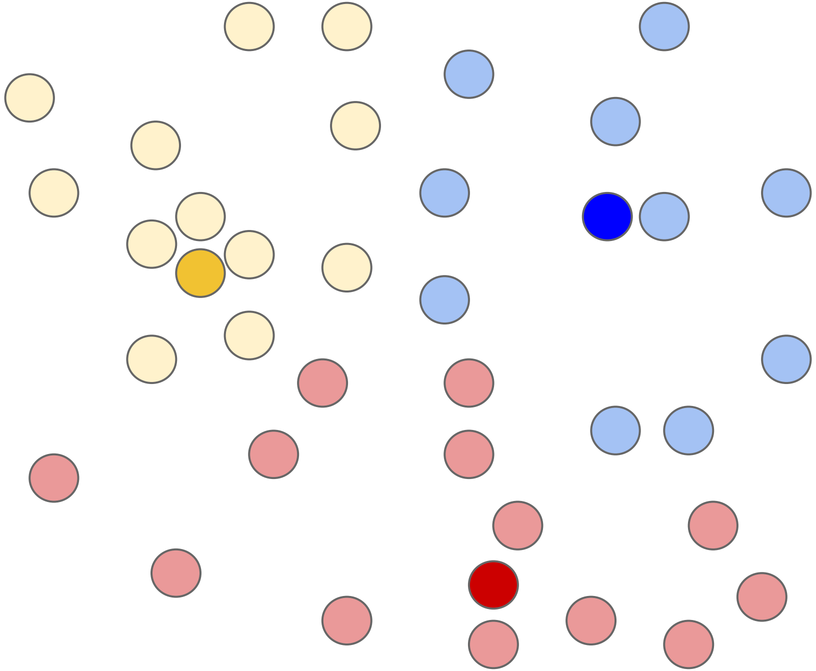
Repeat this for each point



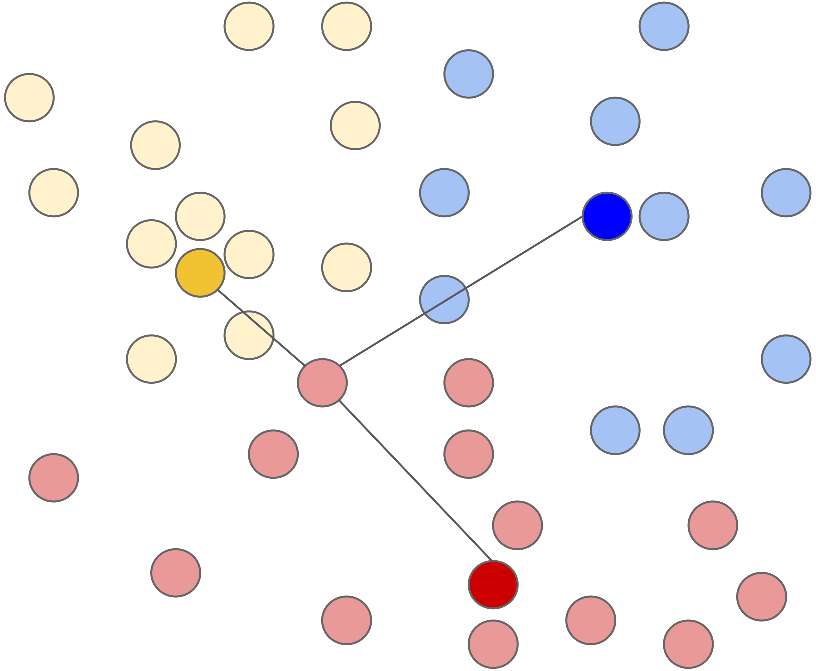
Calculate Mean of Data points in Each Cluster



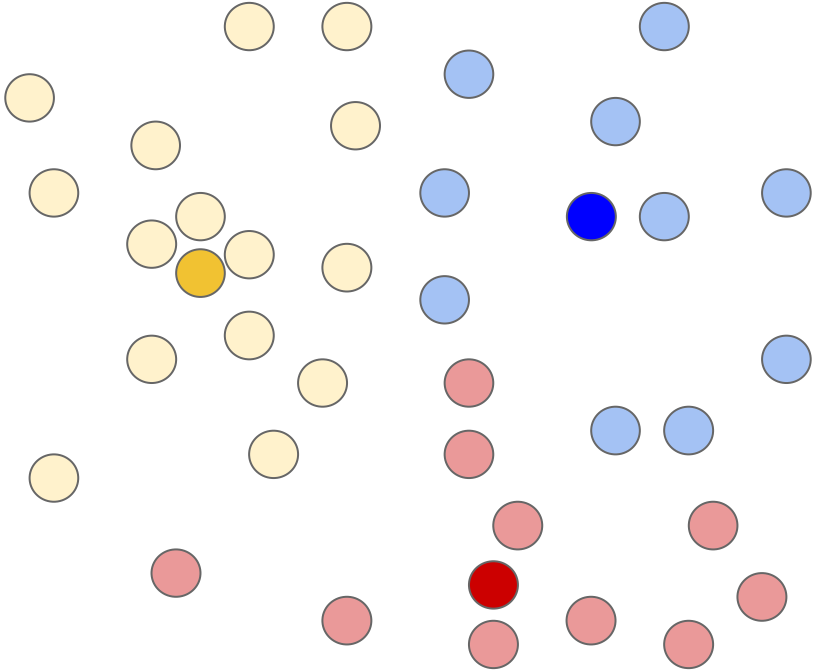
Move Centroid to Mean for each Cluster



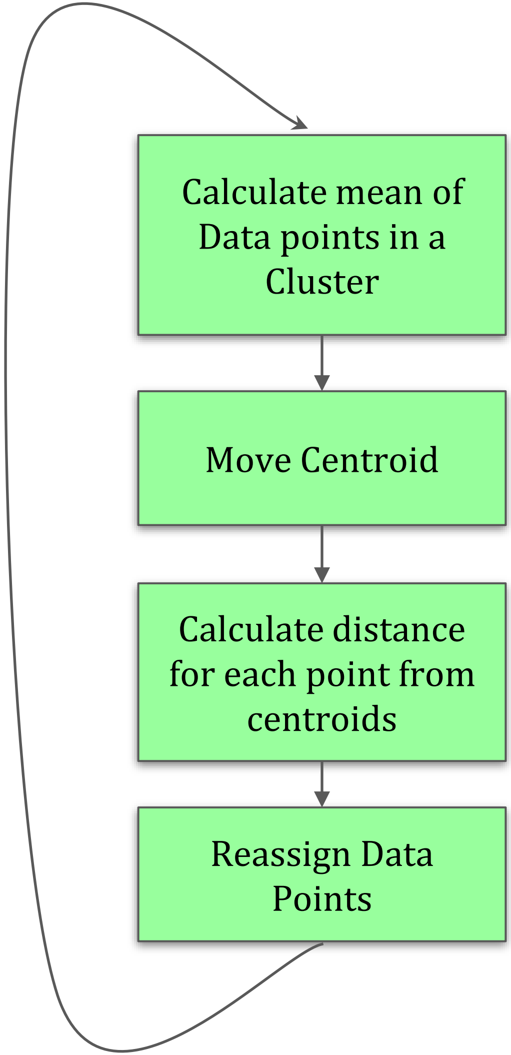
Move Centroid to Mean for each Cluster



Calculate Distance to Centroid for each point



Reassign Each point to cluster



Calculate mean of Data points in a Cluster

Move Centroid

Calculate distance for each point from centroids

Reassign Data Points

..till data points stop changing Cluster

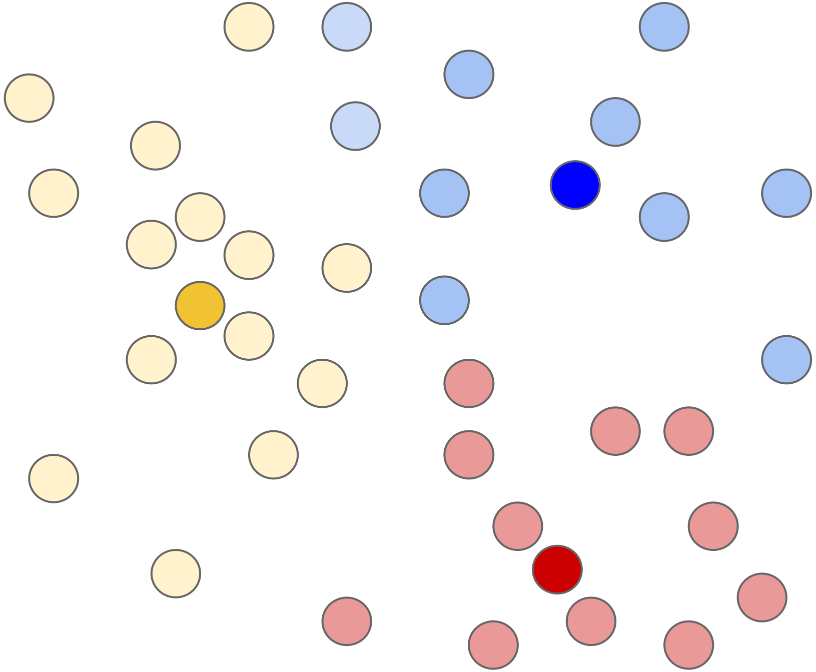
K-Means Goal

Minimize

$$\sum_{i=1}^{n} (x_i - \mu_{c(x_i)})^2$$

Centroid

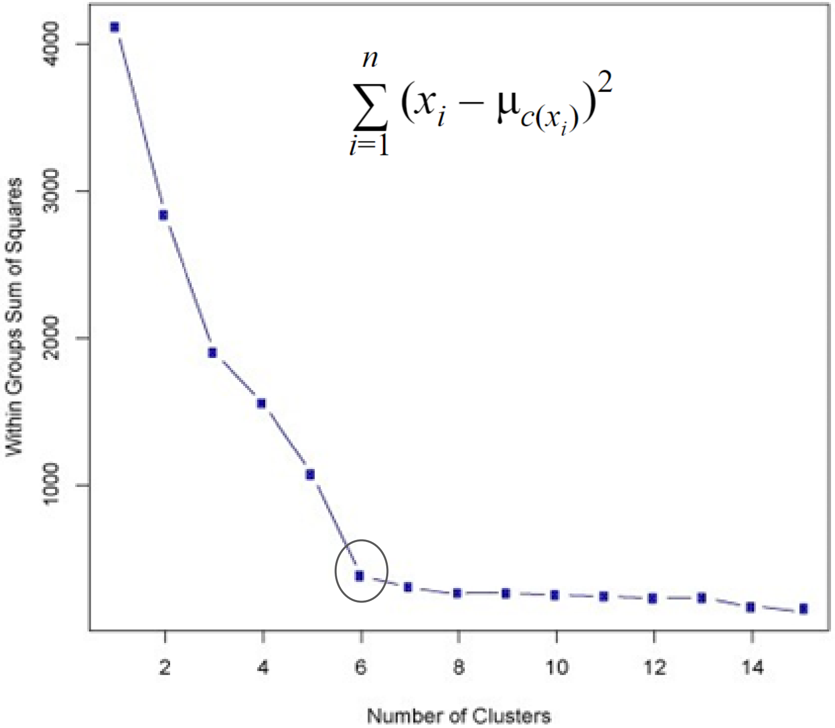
$$c(x_i) \in \{1 \ldots k\}$$



Prediction for a new Data point?

Check distance from Centroids

What should be value of ‘K’?



$$\sum_{i=1}^{n}(x_i - \mu_{c(x_i)})^2$$

Elbow method

9. Given that the total variance T across the entire data set is a constant, T = BC + WC (T = between cluster variations (BC) and within cluster variance (WC)

10. Thus WC = T – BC

11. Minimizing WC = Maximizing BC. Thus the objective of clustering is to get tightest and farthest clusters. Achieving one will automatically get the other i.e. tightest will make the clusters farthest. In the process ensure the clusters are natural, meaningful

12. Unfortunately, there is no well defined algorithm to achieve this. The problem of finding the tightest and farthest clusters in a data set belongs to a family of problems called the NP Hard problems (Non deterministic, Polynomial-time Hard problems). Ref: http://jeffe.cs.illinois.edu/teaching/algorithms/notes/30-nphard.pdf

13. For this reason, the clustering algorithms usually converge to sub-optimal solutions or converge to local optima. However, they are still powerful techniques in highlighting hidden structures

1. Distance measures and some key points:
    a. Choice of distance measures play a key role in cluster analysis
    b. Knowledge of the distribution of data (gaussian or otherwise) will help
    c. Are the various attributes independent or influence each other
    d. Are their outliers in the data on the various dimensions
    e. Though Euclidian distance is the most commonly used distance metric, it has three main features that should be kept in view
        a. It is highly scale dependent. Changing the units of one variable can have a huge influence on the results. Hence standardizing the dimensions is a good practice
        b. It completely ignores the relationship between measurements (Refer to Mahalanobis distance diagram)
        c. It is sensitive to outliers. If the data has outliers that cannot be handled or removed, use of Manhattan distance is preferred
    f. KMeans algorithm implements only Euclidian distance

Machine Learning (K Means Clustering – Some considerations)

1. K-Means (a.k.a Lloyd’s algorithm) clusters data by separating data points into groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squared errors

2. It requires the number of clusters to be specified, hence the term “K” in its name

3. It divides the samples into K disjoint clusters Ci, each described by the mean of the samples in the cluster. The means are commonly called “centroids” (they are not the points from the data)

4. The K-Means algorithm chooses centroids that minimizes the within cluster inertia (variations) across all the clusters

Machine Learning (K Means Clustering – Some considerations)

5. From a computational perspective, the k-means algorithm is indifferent to the units of measure for a given attribute (for example, meters or centimeters for a patient's height). However, the algorithm will identify different clusters depending on the choice of the units of measure.

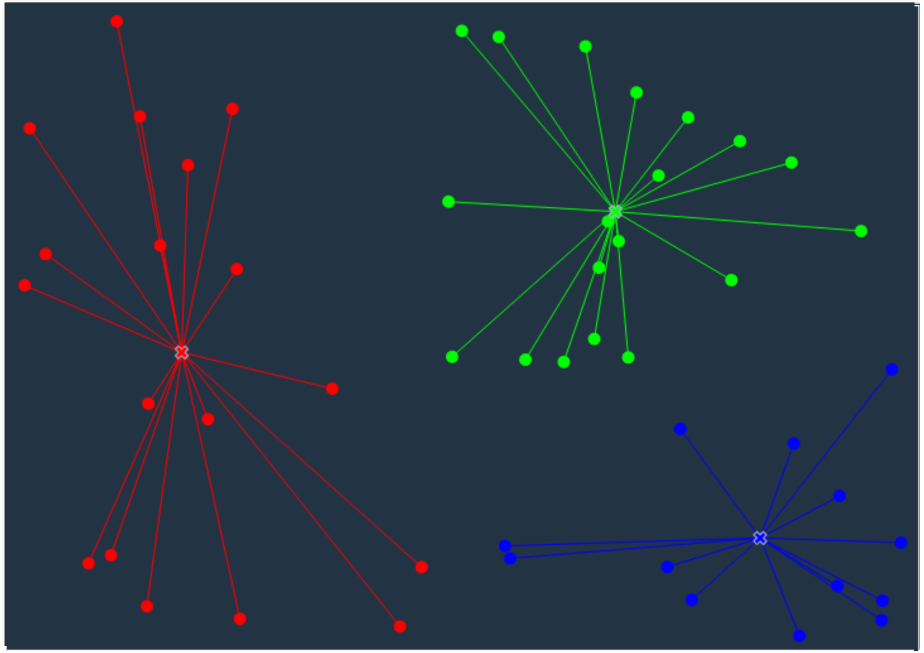
6. Choosing different starting points can result in different clusters. The algorithm is sensitive to the initial starting condition

7. Given enough time, K-means will always converge, however this may be a local minimum. This is highly dependent on the initialization of the centroids

8. Scikit-learn has implemented K-mean++ initialization scheme, which initializes centroids to be distant to one another which provably leads to better results

Machine Learning (K-means Clustering)

K-means treats feature values as coordinates in the multi-dimensional feature space



In the update phase the average position of each cluster is calculated (based on position of the points in the cluster)

Note : based on the distance of some boundary points from the new centroid, they get reallocated to different cluster

Ref: http://tech.nitoyon.com/en/blog/2013/11/07/k-means/

Machine Learning (K-means Clustering)

K Means Algorithm

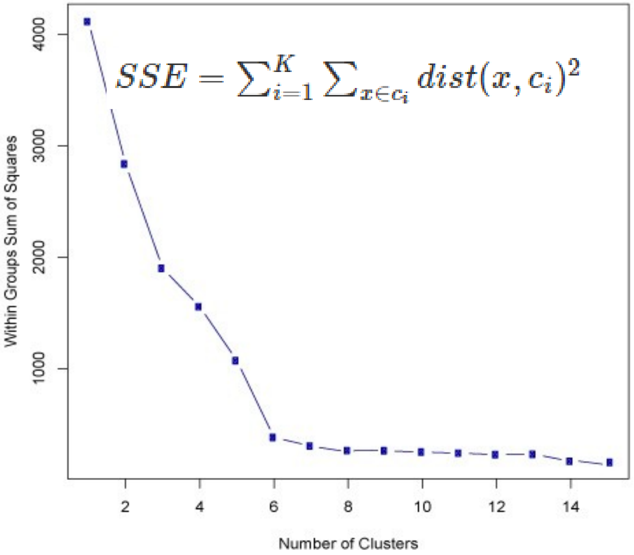
1. Randomly pick K points in the feature space as initial centroids. Or randomly assign values 1 – K to each data point. This is the first version of K clusters. Obviously the worst clusters

2. Iterate until centroids stop moving or cluster assignment stops changing –
   a. For each of the K clusters, compute the centroid i.e. avg on each attribute. The point in the mathematical space where these averages meet, is the new centroid. The Kth cluster centroid is thus a vector of P feature means for the observations in the Kth cluster

   b. Re assign each data point to the centroid which is closest to them. Closeness is defined using Euclidian distance

Machine Learning (K-means Clustering)

Without apriori knowledge, one can use elbow method that measures the homogeneity or heterogeneity within clusters as the number of clusters change (i.e. K is changed). One way to measure is use sum of square errors in each cluster

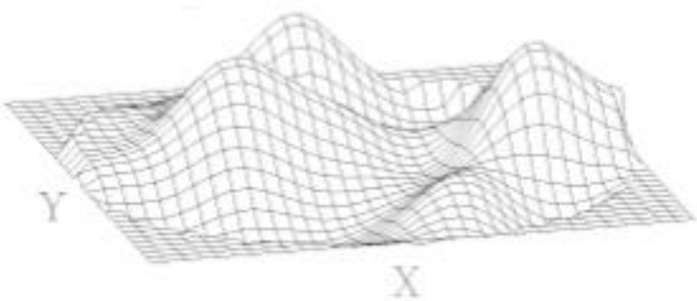


$$SSE = \sum_{i=1}^{K} \sum_{x \in c_i} dist(x, c_i)^2$$

Machine Learning (K-means Clustering)

Visual Analysis for Clustering

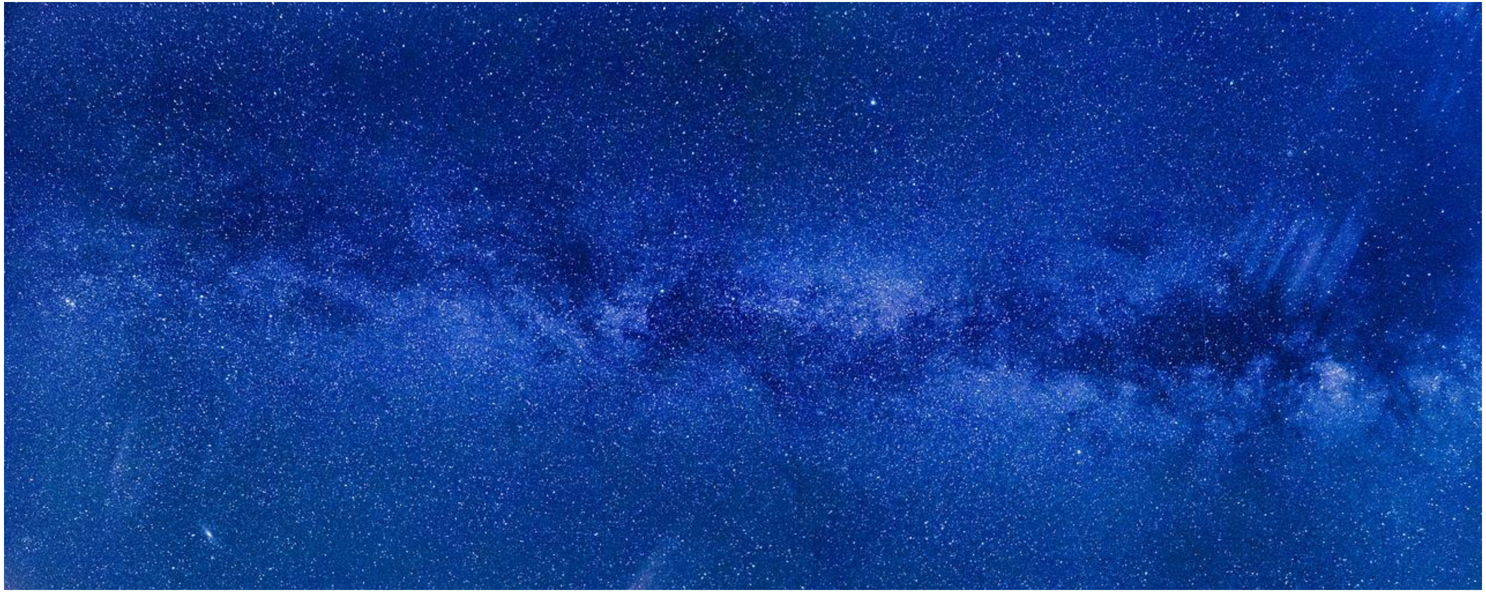
1. Visual analysis of the attributes selected for the clustering may give an idea of the range of values that K should be evaluated in



2. Identifying the attributes on which clusters are clearly demarcated and using them in incremental order to build the multi-dimensional clusters likely to give much better clusters than using all the attributes at one go

Machine Learning (K-Means Clustering)

Strengths	Weakness
Use simple principles without the need for any complex statistical terms	Computationally intensive How to fix K?
Once clusters and their associated centroids are identified, it is easy to assign new objects (for example, new customers) to a cluster based on the object's distance from the closest centroid	The k-means algorithm is sensitive to the starting positions of the initial centroid. Thus, it is important to rerun the k-means analysis several times for a particular value of k to ensure the cluster results provide the overall minimum WSS
Because the method is unsupervised, using k-means helps to eliminate subjectivity from the analysis.	Susceptible to curse of dimensionality



Technical Support Analysis with K-Means

Lab1_tech_supp_analysis.ipynb

Customer Segmentation

Lab2_HandsOn_Customer Segmentation using K-Means.ipynb

Context

In today’s competitive world, it is crucial to understand customer behavior and categorize customers to tailor promotional, marketing and product development strategies.

Divide customers into groups...

- that share certain characteristics
  - Geographic regions
  - Demographics e.g age, gender, marital status, income etc
  - Psychographics e.g values, interests, lifestyle etc
  - Purchase behaviour e.g previous purchases, page views, shipping preferences etc

Can be difficult to do...

- Almost infinite characteristics which can be used
- No single, correct way

Online Retail Dataset

https://archive.ics.uci.edu/ml/datasets/Online+Retail

Goal

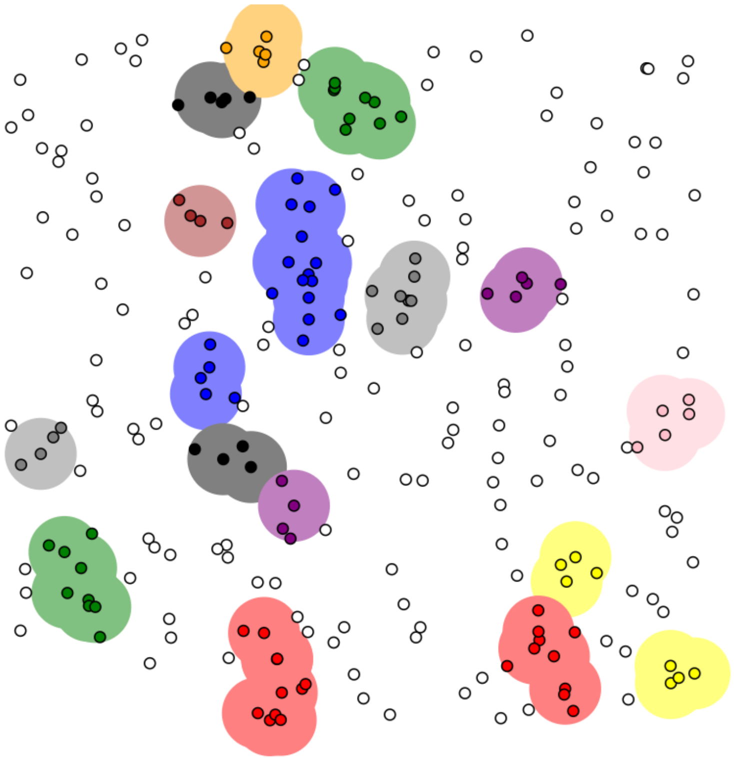
Identify High and Low Value Customers for marketing purposes

What will be our features?

1. Recency of the Purchase
2. Frequency of the Purchases
3. Monetary value of Purchases

Customer Segmentation using K-Means in Scikit-Learn

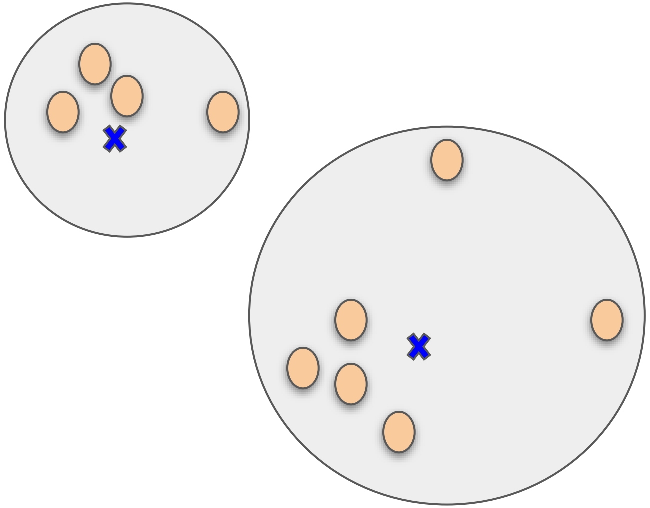
Questions?



https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

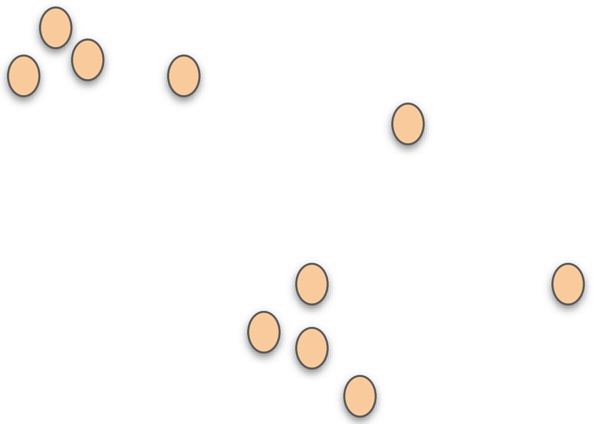
DBSCAN

Density Based Spatial Clustering of Applications with Noise



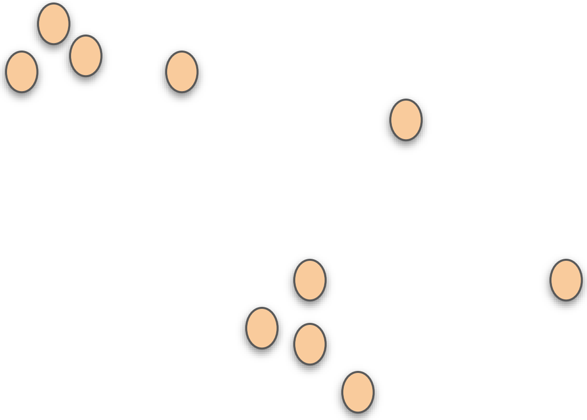
How would K-Means Cluster these points?

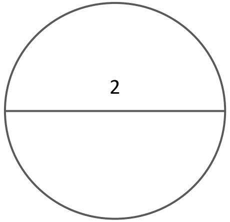
- Number of clusters decided by Data, unlike K-Means
- Some data points may not be part of any Cluster
- Views clusters as area of high density, separated by areas of low density
- Uses two parameters to define density
    - min_samples
    - eps



How would DBSCAN Cluster the points?

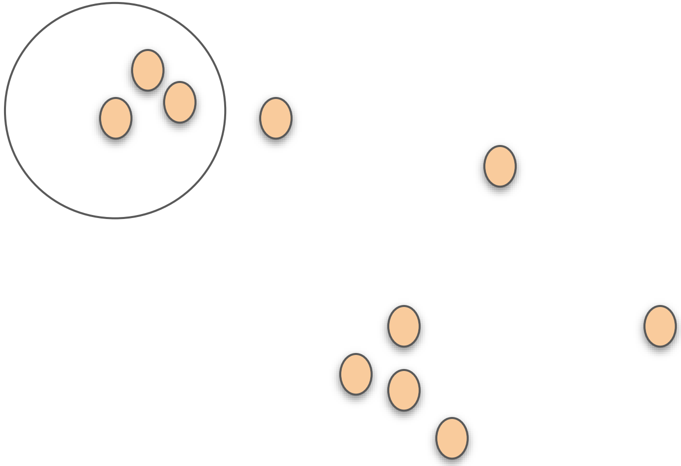
Eps = 1
Min_samples = 4

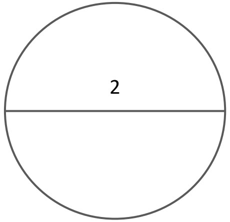




How would DBSCAN Cluster the points?

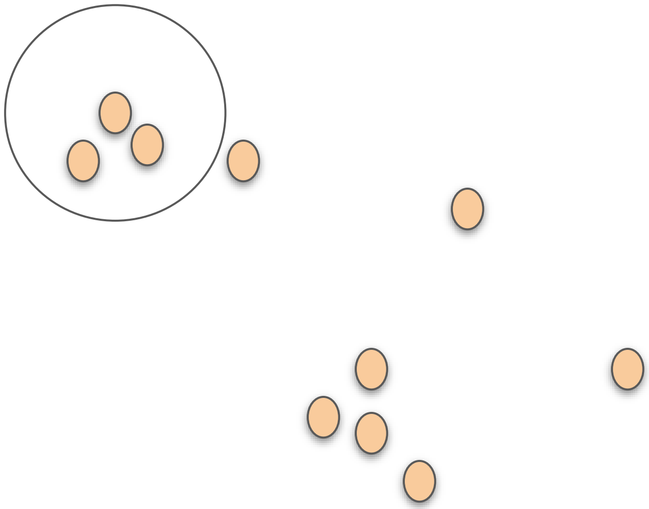
Eps = 1
Min_samples = 4

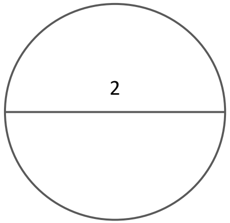




Is this a Cluster?

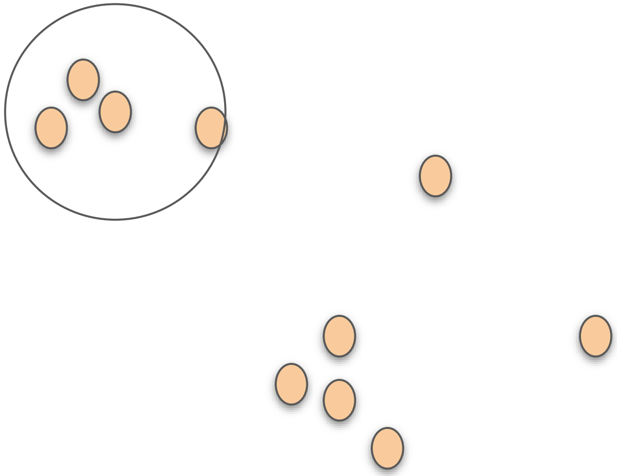
Eps = 1
Min_samples = 4

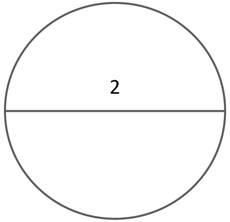




Is this a Cluster?

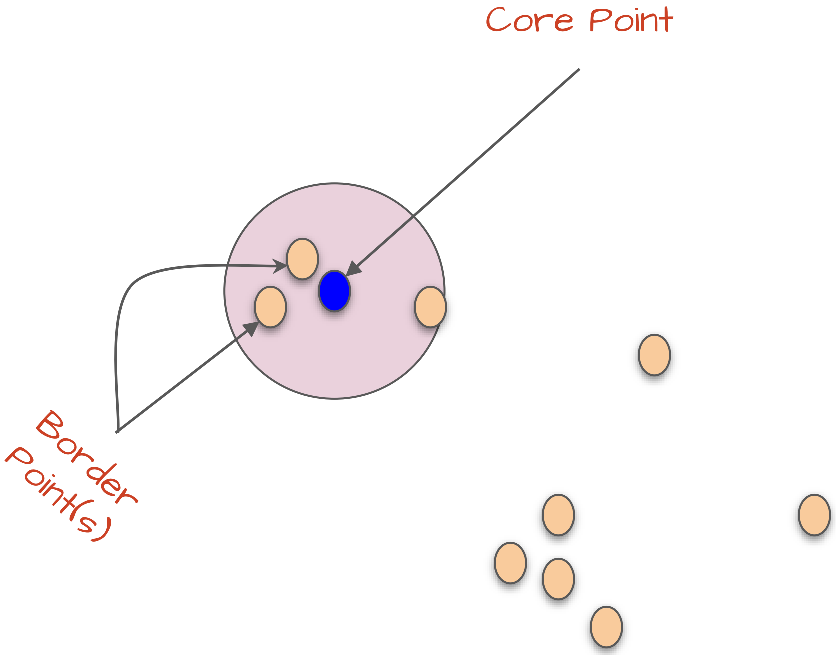
Eps = 1
Min_samples = 4

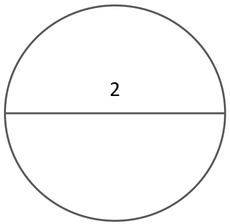




Is this a Cluster?

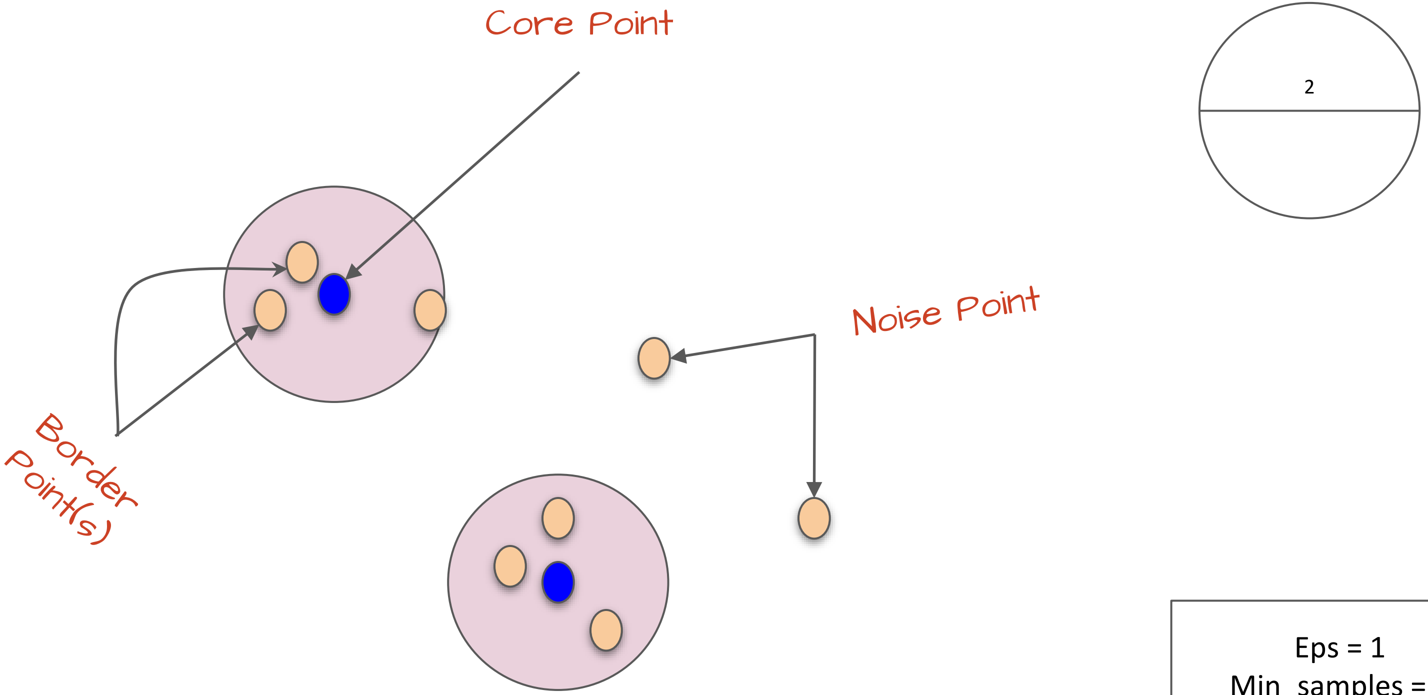
Eps = 1
Min_samples = 4



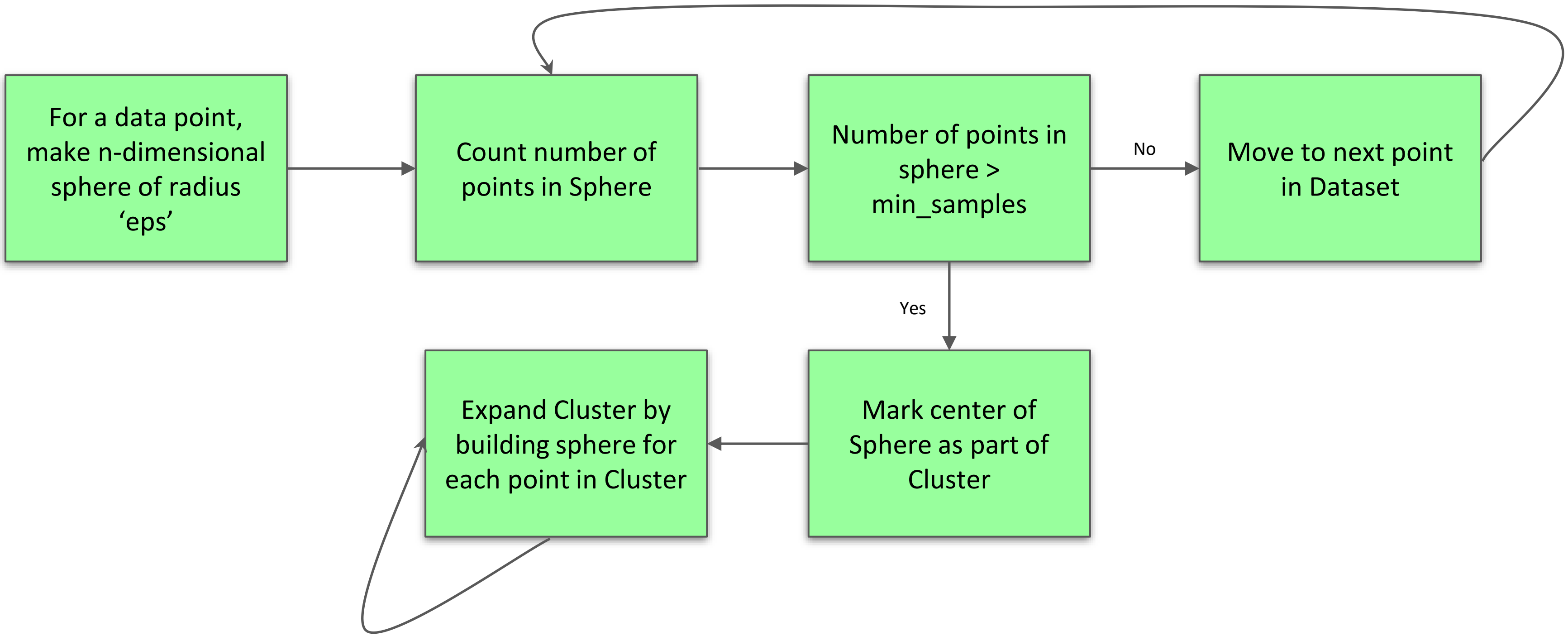


Is this a Cluster?
Yes

Eps = 1
Min_samples = 4



Eps = 1
Min_samples = 4



For a data point, make n-dimensional sphere of radius ‘eps’

Count number of points in Sphere

Number of points in sphere > min_samples

Move to next point in Dataset

Mark center of Sphere as part of Cluster

Expand Cluster by building sphere for each point in Cluster

Visualizing DBSCAN

https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

Implementing DBSCAN

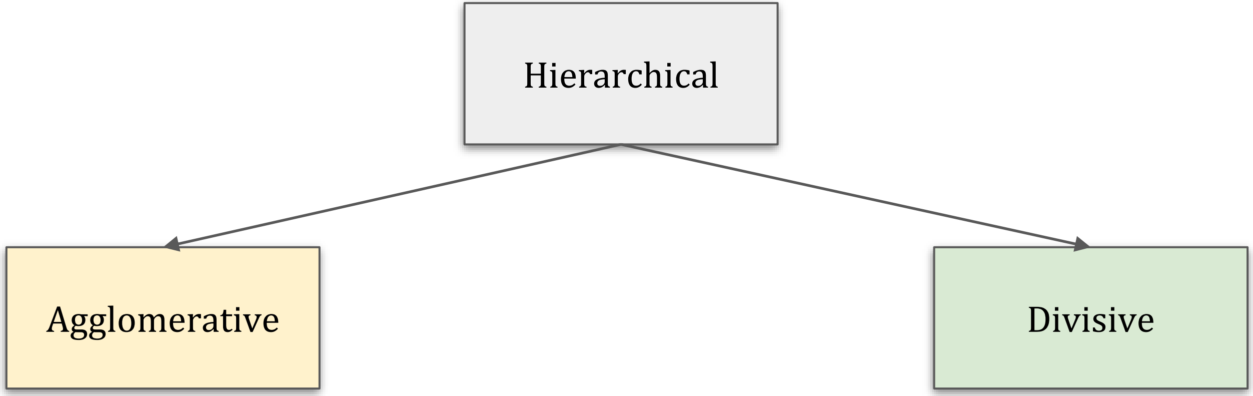
Strengths

- No pre-defined K required
- Can find arbitrary shaped clusters
- Robust to outliers

Weakness

- ‘eps’ can be difficult to choose
- May not work well with large differences in densities

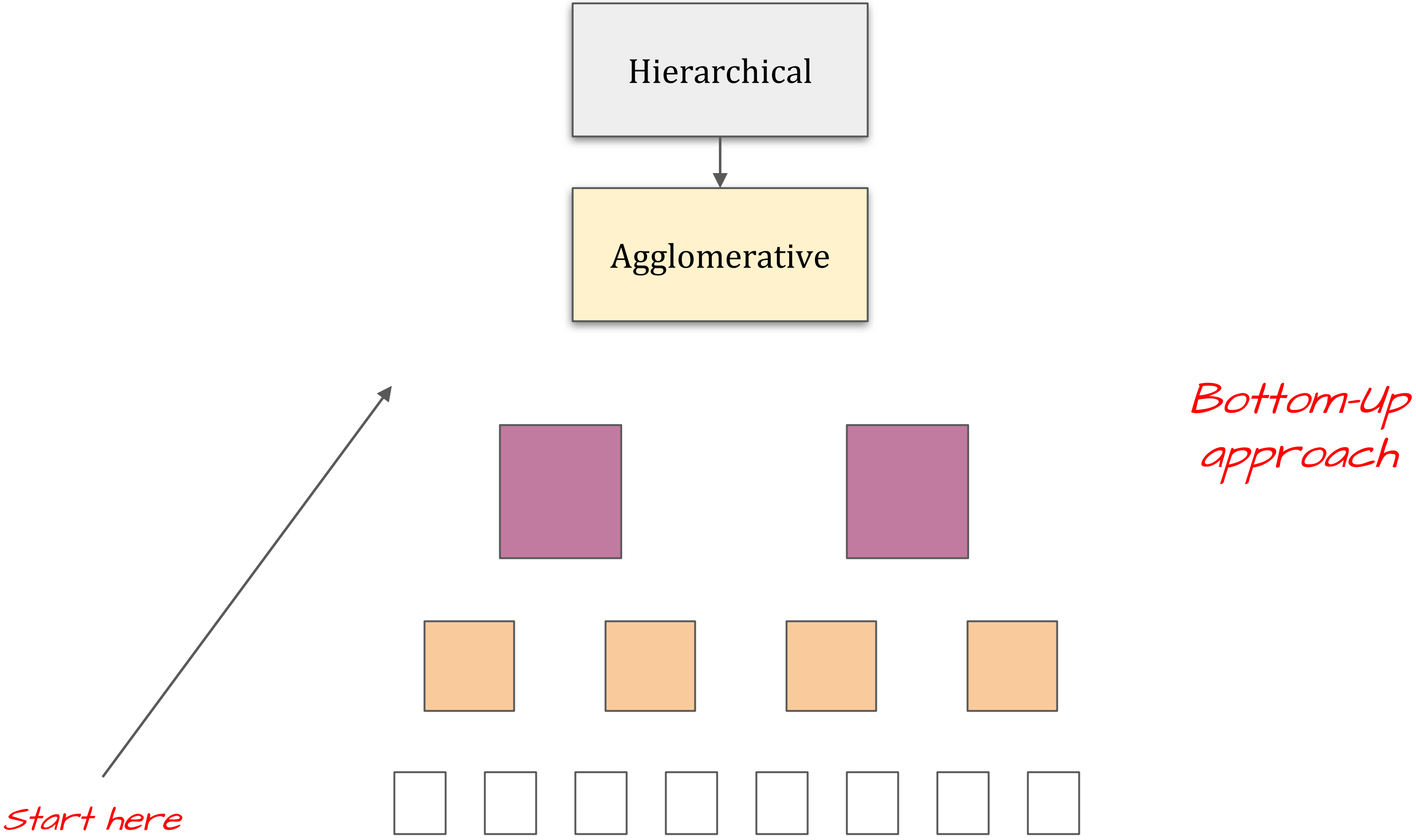
Hierarchical Clustering

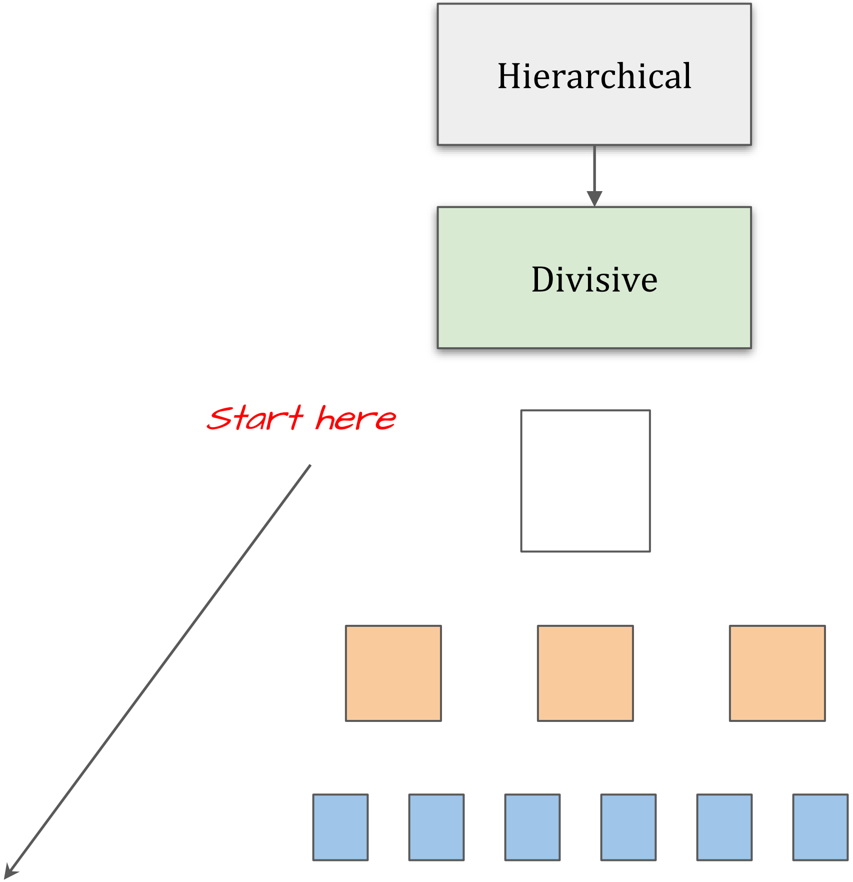


Hierarchical

Agglomerative

Divisive


Hierarchical
Agglomerative
Bottom-Up approach
Start here



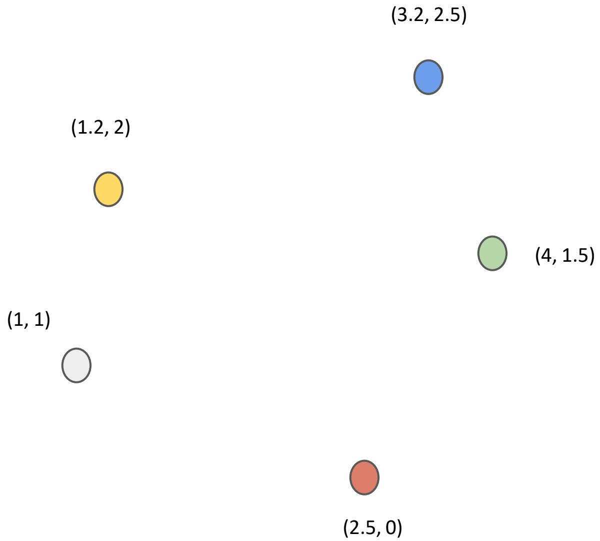
Hierarchical

Divisive

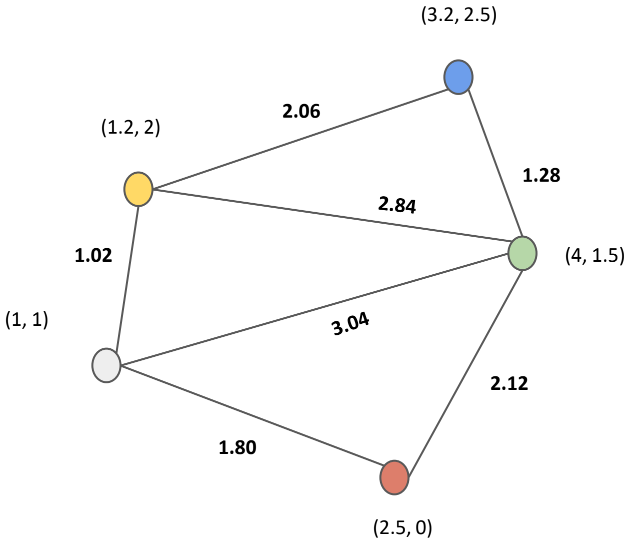
Start here

Top Down approach

Agglomerative Clustering

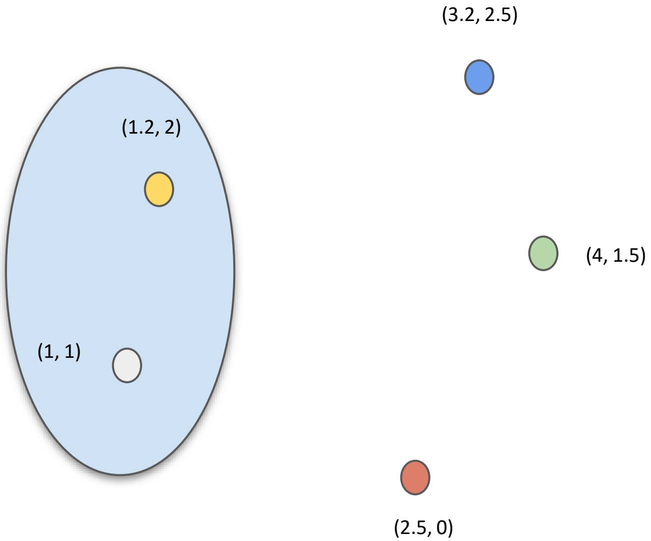


Each point is a Cluster

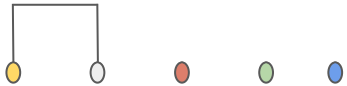


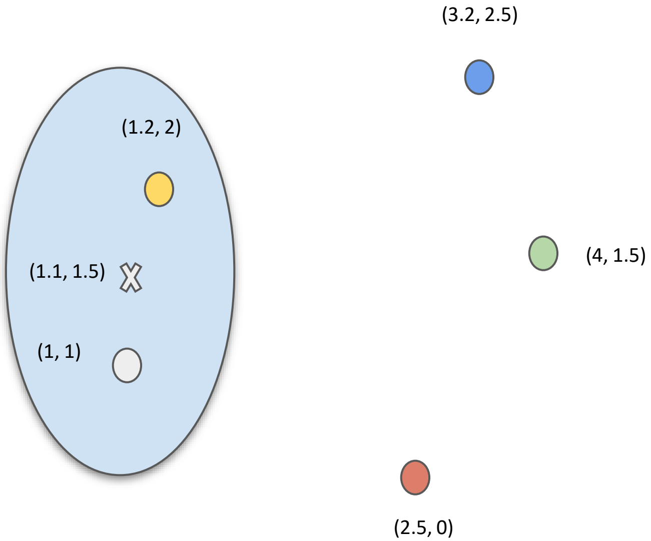
Calculate Distance between Each Cluster

● ○ ● ● ●

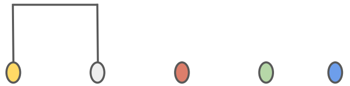


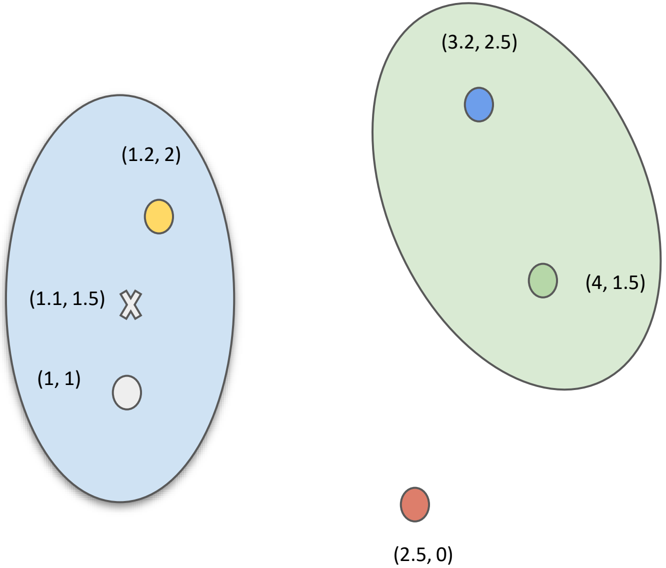
Join nearest Clusters



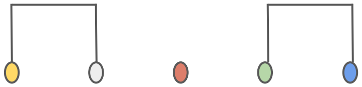


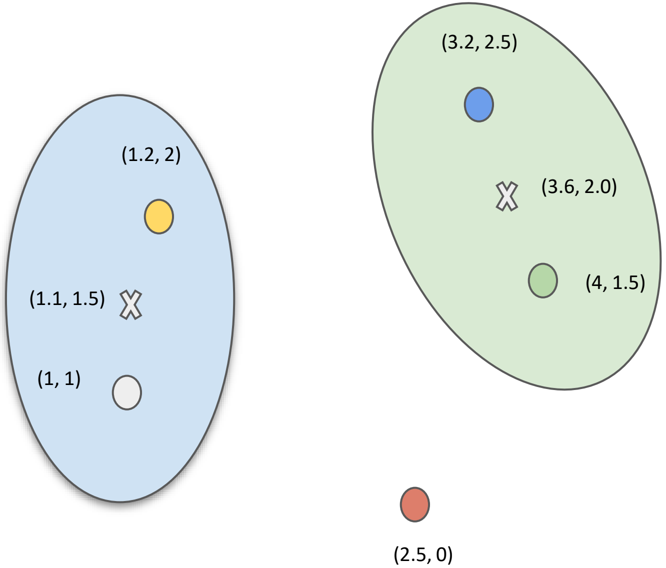
Find Centroid of new Cluster



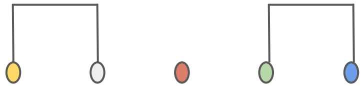


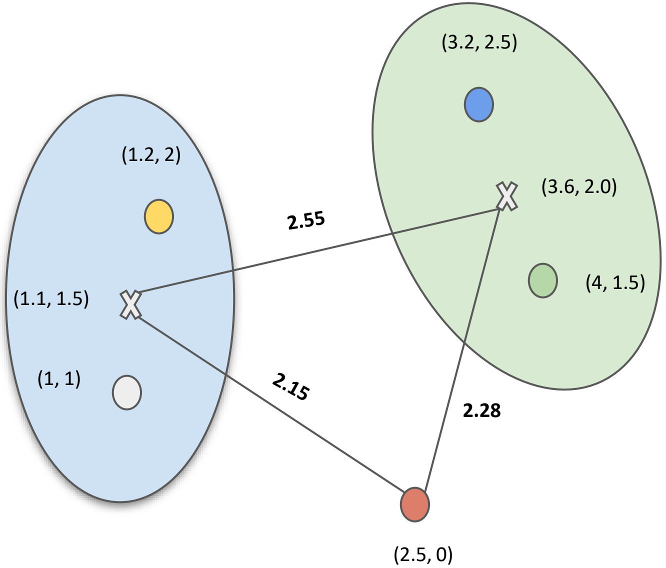
Join Clusters



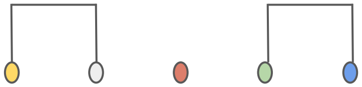


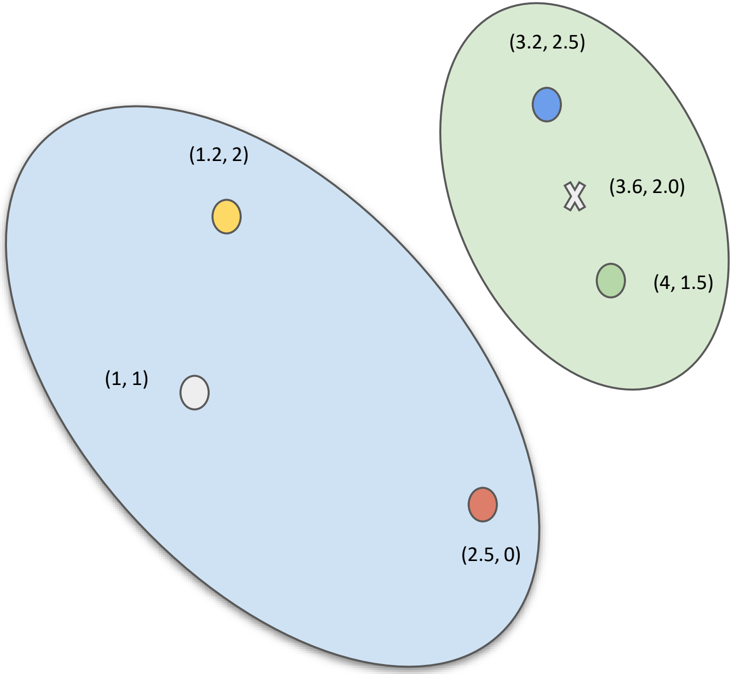
Identify Centroid



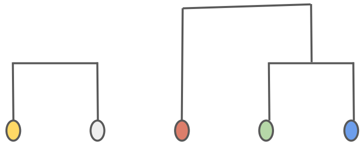


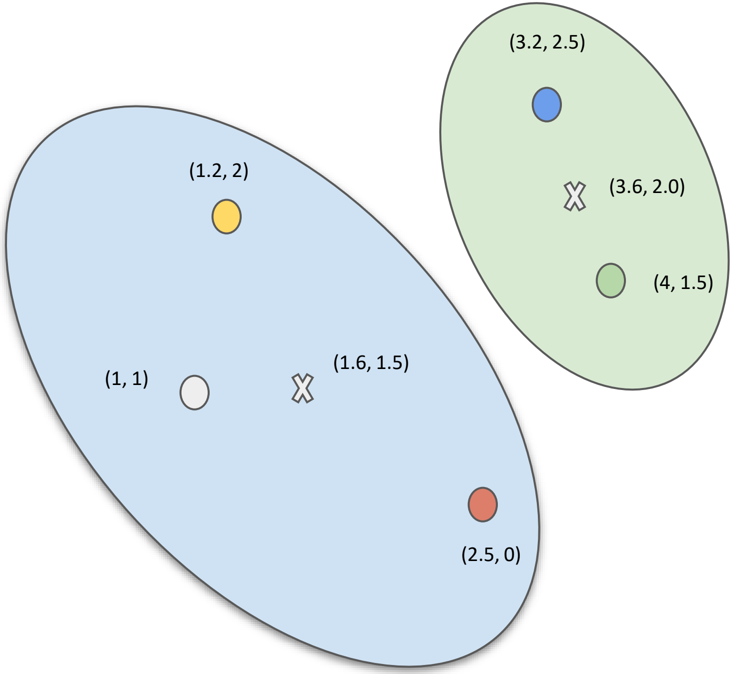
Measure Distance between Clusters



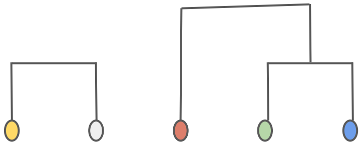


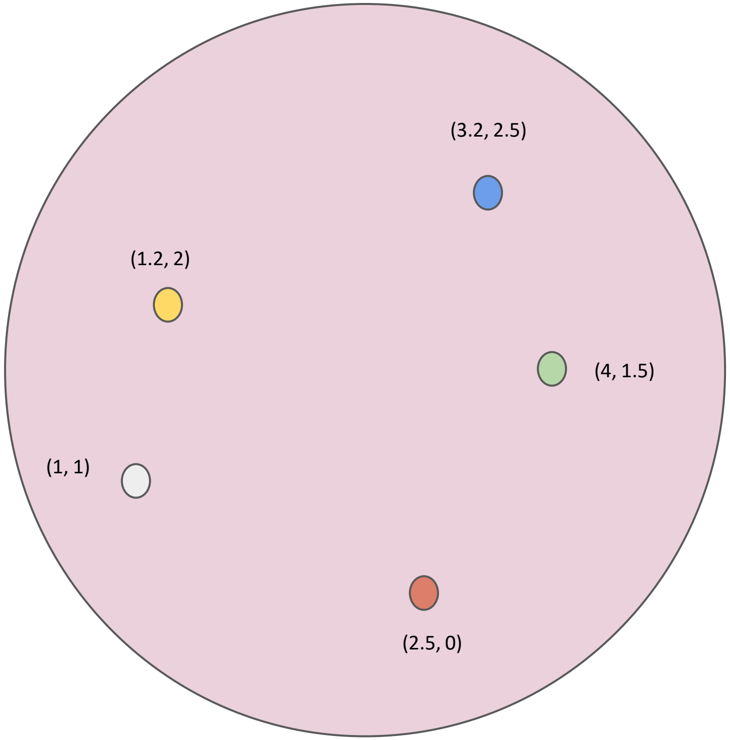
Identify Centroid



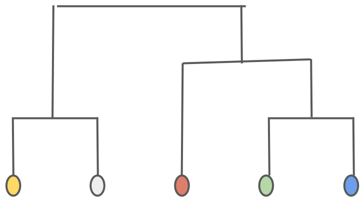


Identify Centroid





Join Clusters



Dendrogram

Machine Learning (Hierarchical (Agglomerative) Clustering)

1. The agglomerative clustering starts with each cluster comprising exactly one data point in the feature space

2. It progressively agglomerates / combines the two nearest clusters until there is one grand cluster left in the feature space

3. For the closest cluster analysis, each of the inter cluster distance measurement techniques (single link, complete link, average link, centroid distance) can be implemented

    a. In single linkage method, the minimum distance between nearest points from the two clusters is used to consolidate clusters

    b. In complete linkage, distance between two farthest points from each cluster is considered

    c. Group average clustering is based on the average distance between clusters

4. Prior domain knowledge helps in deciding the inter cluster distance metric selection. If the clusters are likely to be in long chain or sausage like, minimum distance (single linkage) would be a good choice

5. Complete and average linkage are better choice if the clusters are likely to be spherical

Machine Learning (Clustering – Measuring distance between clusters)

1. Ideally, a good clustering should result in compact clusters separated from one another by maximal distance. This calls for measuring the distance between cluster. The most widely used methods include :

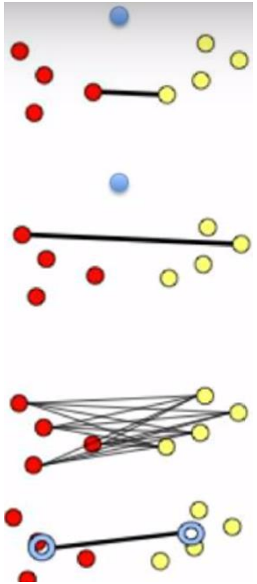
    a. Minimum distance(single linkage) – is the distance between pair of records Ai and Bj that belong to clusters A and B respectively and are closest

    b. Maximum distance(complete linkage) – is the largest distance between the pair of records Ai and Bj that belong to cluster A and B respectively

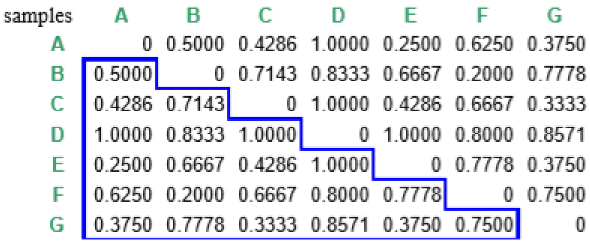
    c. Average distance (average linkage) - average distance of all possible distances between records in one cluster to records in other cluster

    d. Centroid distance - the distance between centroids of the different clusters.

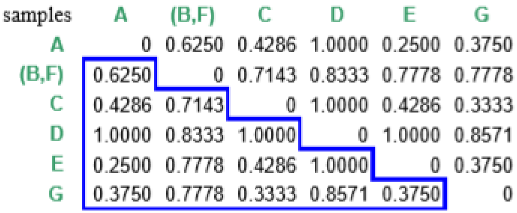
2. Distance between clusters is used in hierarchical clustering



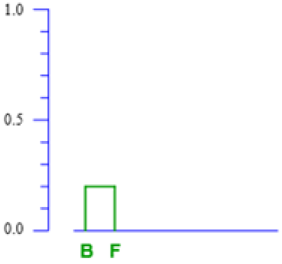
Complete Linkage

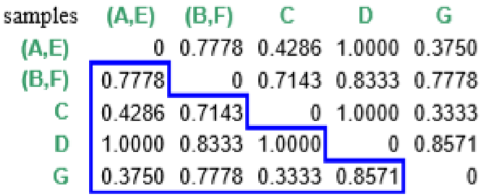


samples	A	B	C	D	E	F	G
A	0	0.5000	0.4286	1.0000	0.2500	0.6250	0.3750
B	0.5000	0	0.7143	0.8333	0.6667	0.2000	0.7778
C	0.4286	0.7143	0	1.0000	0.4286	0.6667	0.3333
D	1.0000	0.8333	1.0000	0	1.0000	0.8000	0.8571
E	0.2500	0.6667	0.4286	1.0000	0	0.7778	0.3750
F	0.6250	0.2000	0.6667	0.8000	0.7778	0	0.7500
G	0.3750	0.7778	0.3333	0.8571	0.3750	0.7500	0

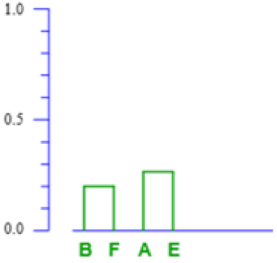


samples	A	(B,F)	C	D	E	G
A	0	0.6250	0.4286	1.0000	0.2500	0.3750
(B,F)	0.6250	0	0.7143	0.8333	0.7778	0.7778
C	0.4286	0.7143	0	1.0000	0.4286	0.3333
D	1.0000	0.8333	1.0000	0	1.0000	0.8571
E	0.2500	0.7778	0.4286	1.0000	0	0.3750
G	0.3750	0.7778	0.3333	0.8571	0.3750	0

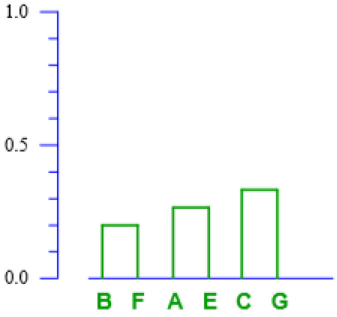


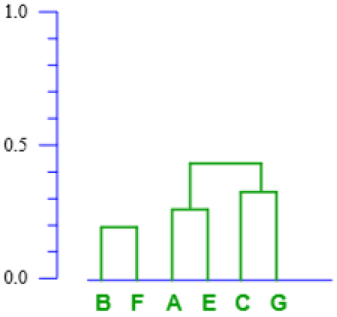


samples	(A,E)	(B,F)	C	D	G
(A,E)	0	0.7778	0.4286	1.0000	0.3750
(B,F)	0.7778	0	0.7143	0.8333	0.7778
C	0.4286	0.7143	0	1.0000	0.3333
D	1.0000	0.8333	1.0000	0	0.8571
G	0.3750	0.7778	0.3333	0.8571	0



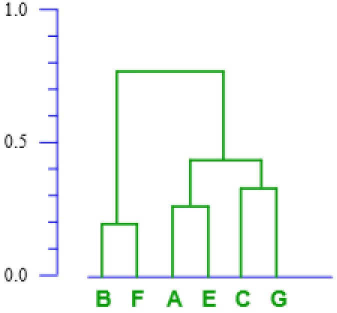
samples	(A,E)	(B,F)	(C,G)	D
(A,E)	0	0.7778	0.4286	1.0000
(B,F)	0.7778	0	0.7778	0.8333
(C,G)	0.4286	0.7778	0	1.0000
D	1.0000	0.8333	1.0000	0

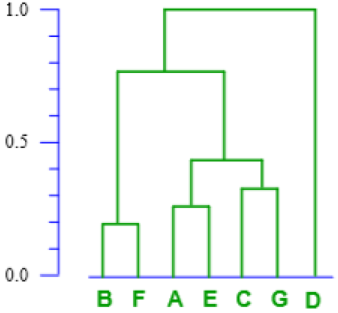




samples	(A,E,C,G)	(B,F)	D
(A,E,C,G)	0	0.7778	1.0000
(B,F)	0.7778	0	0.8333
D	1.0000	0.8333	0

samples	(A,E,C,G,B,F)	D
(A,E,C,G,B,F)	0	1.0000
D	1.0000	0



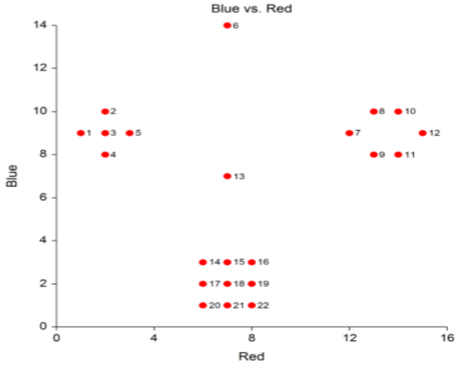


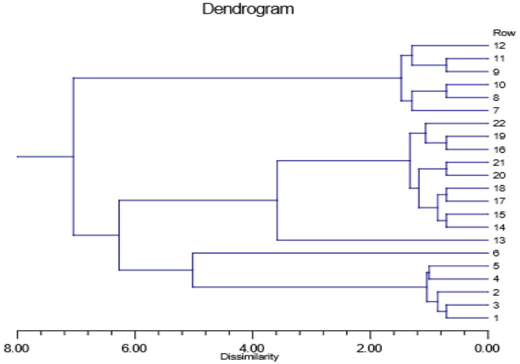
Machine Learning (Hierarchical (Agglomerative) Clustering)

1. Dendrogram is a tree like diagram that summarizes the process of clustering. At the leaf are the records representing the data points while root node is the entire data. The intermediate nodes have two daughter nodes representing sub groups

2. Similar records are joined by lines who’s vertical length reflects the relative distance between the data points

3. When viewed bottom up, the tree posses a monotonicity property. Dissimilarity between the merged clusters is monotone increasing with the level of merger





Source: https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Hierarchical_Clustering-Dendrograms.pdf

Machine Learning (Hierarchical (Agglomerative) Clustering)

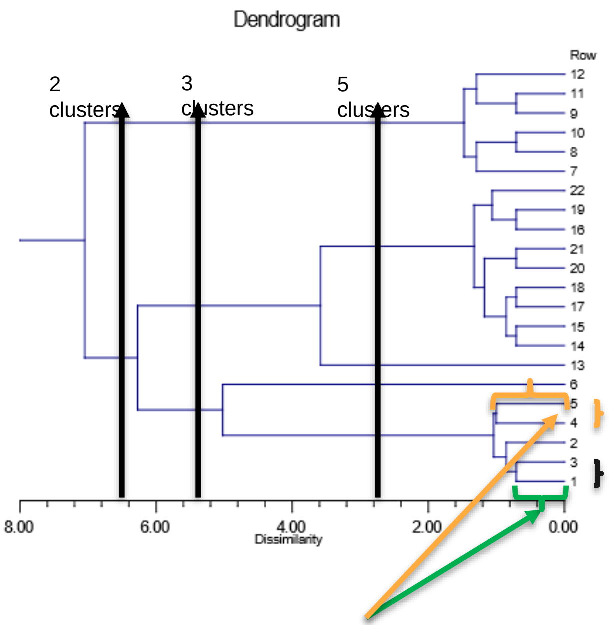
1. The horizontal axis of the dendrogram represents the distance or dissimilarity between clusters (the scale is in reverse order)

2. The vertical axis represents the objects and clusters.

3. Each fusion of two clusters is represented on the graph by the splitting of a horizontal line

4. The horizontal position of the split, shown by the short vertical bar, gives the distance (dissimilarity) between the two clusters

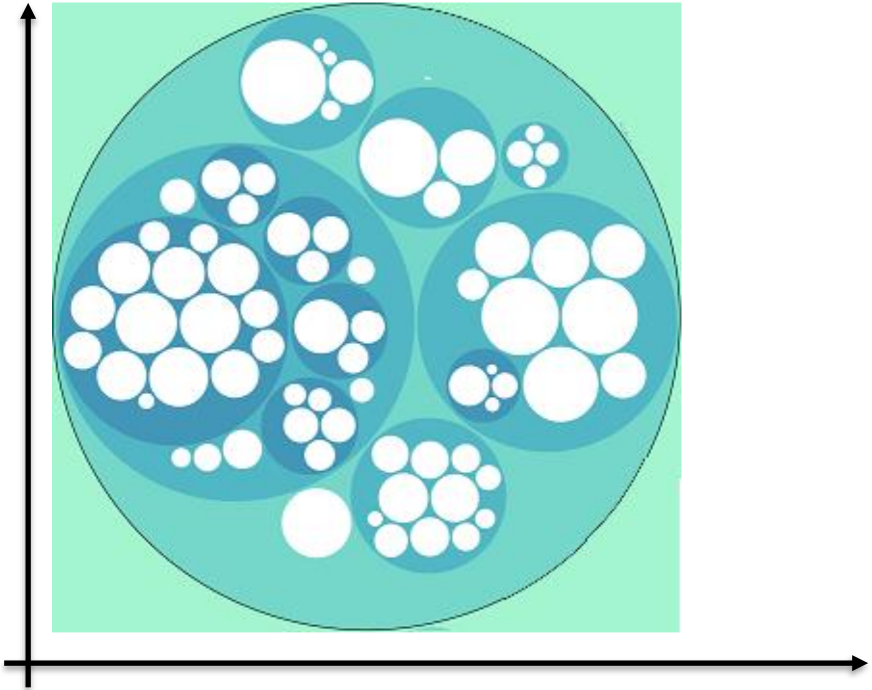
5. When we draw a vertical at any point on the X axis, the number of lines it cuts indicates number of clusters at that value of dissimilarity



Distance/ dissimilarity between 1,3 is less than between 4 and 5. This is reflected in the length of the horizontal bar which is longer for 4,5 compared to 1,3

Source: https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Hierarchical_Clustering-Dendrograms.pdf

Clusters within clusters



How many Clusters to have?

Questions?