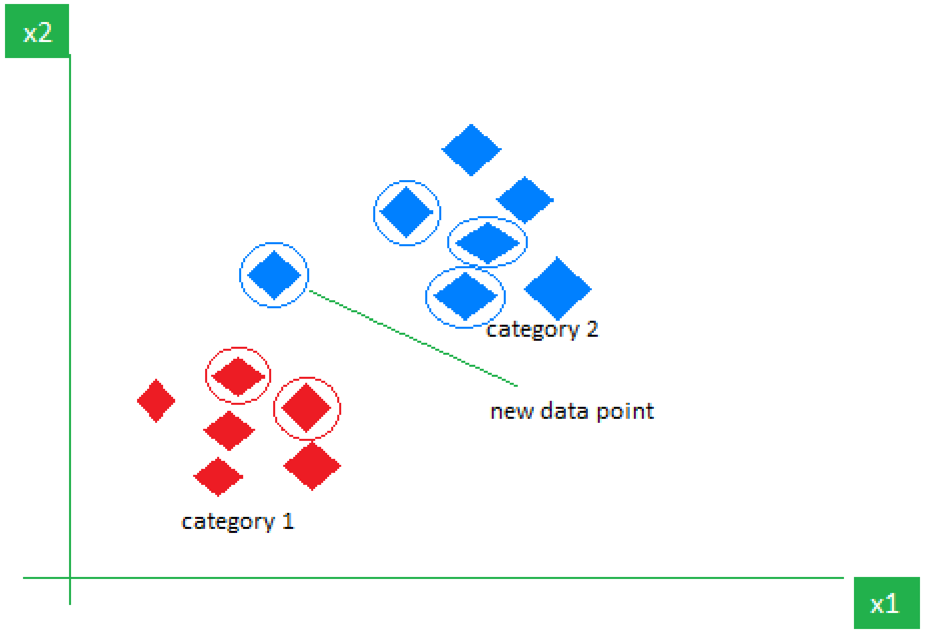
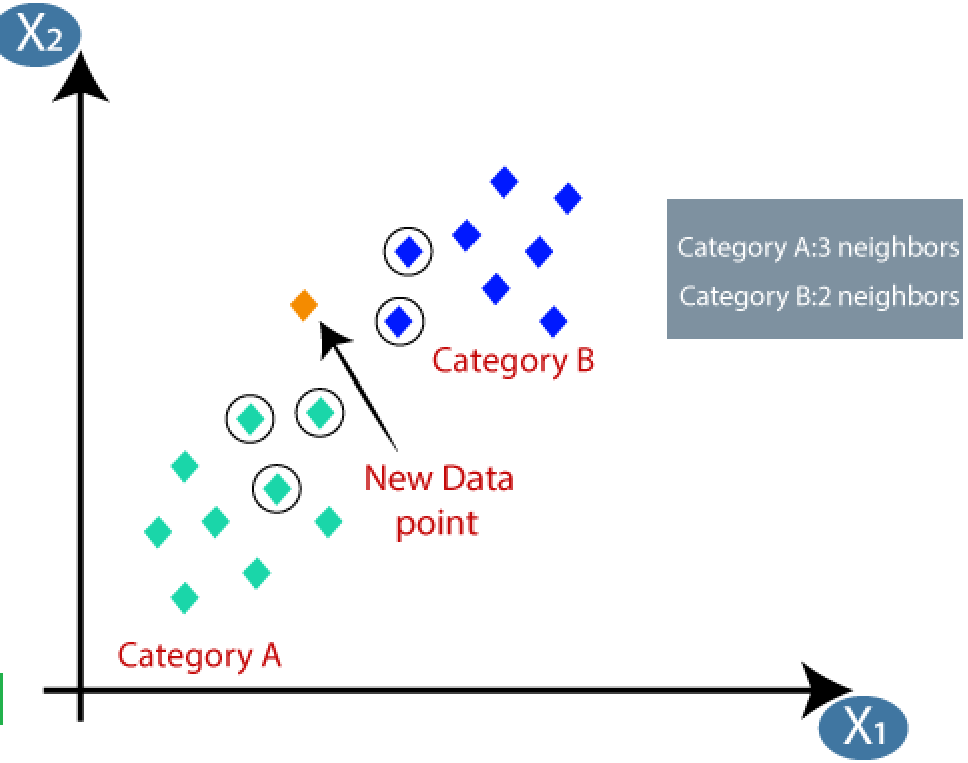
AI Algorithms – 4:
KNN, K-Means

- **Sayed Ahmed**
- PhD Studies in Electrical and Computer Eng. (McMaster University) (Partially Complete)
- Master of Engineering in Electrical and Computer Engineering (McMaster University)
- MSc in Data Science and Analytics (Toronto Metropolitan University/Ryerson)
- MSc in Computer Science (U of Manitoba)
- BSc. Engineering in Computer Science and Engineering (BUET)
- Extensive experience in Software Development and Engineering (primarily in Canada)
- Significant experience in Teaching
- Taught in Universities, Colleges, and Training Institutes

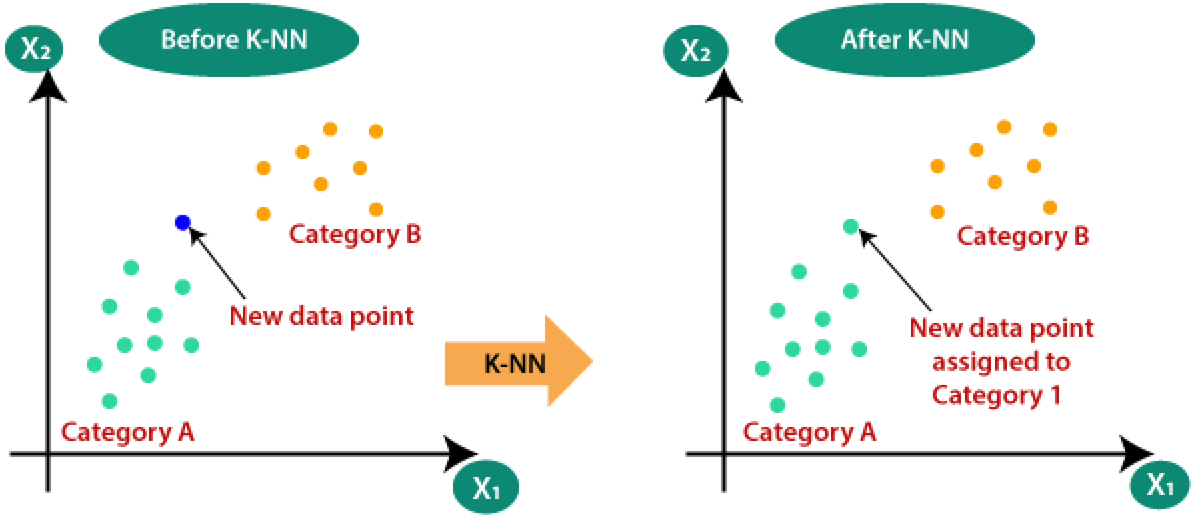
What is KNN?





- It’s about: Classification, and Clustering: https://www.geeksforgeeks.org/k-nearest-neighbours/ https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

What is KNN?

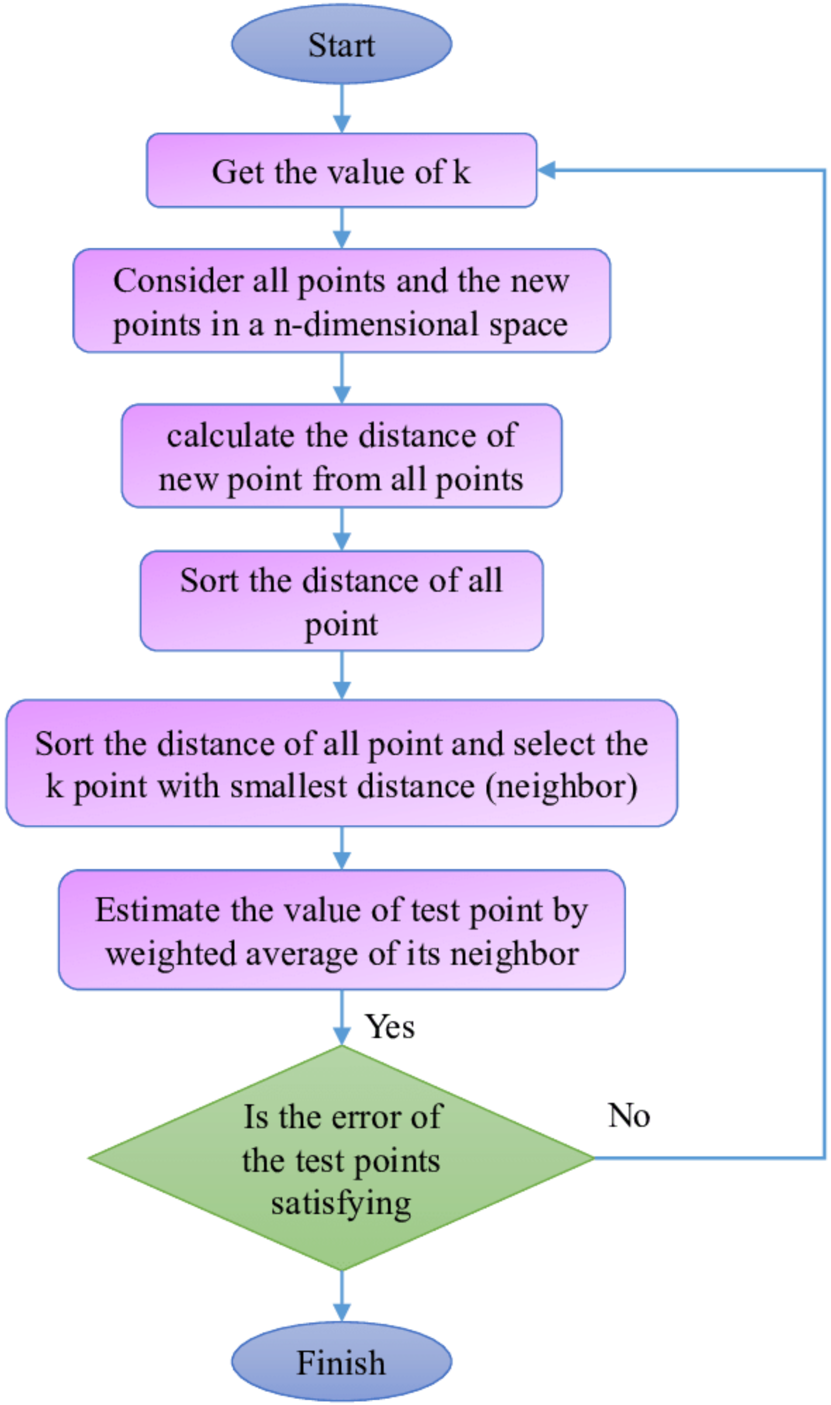


- https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

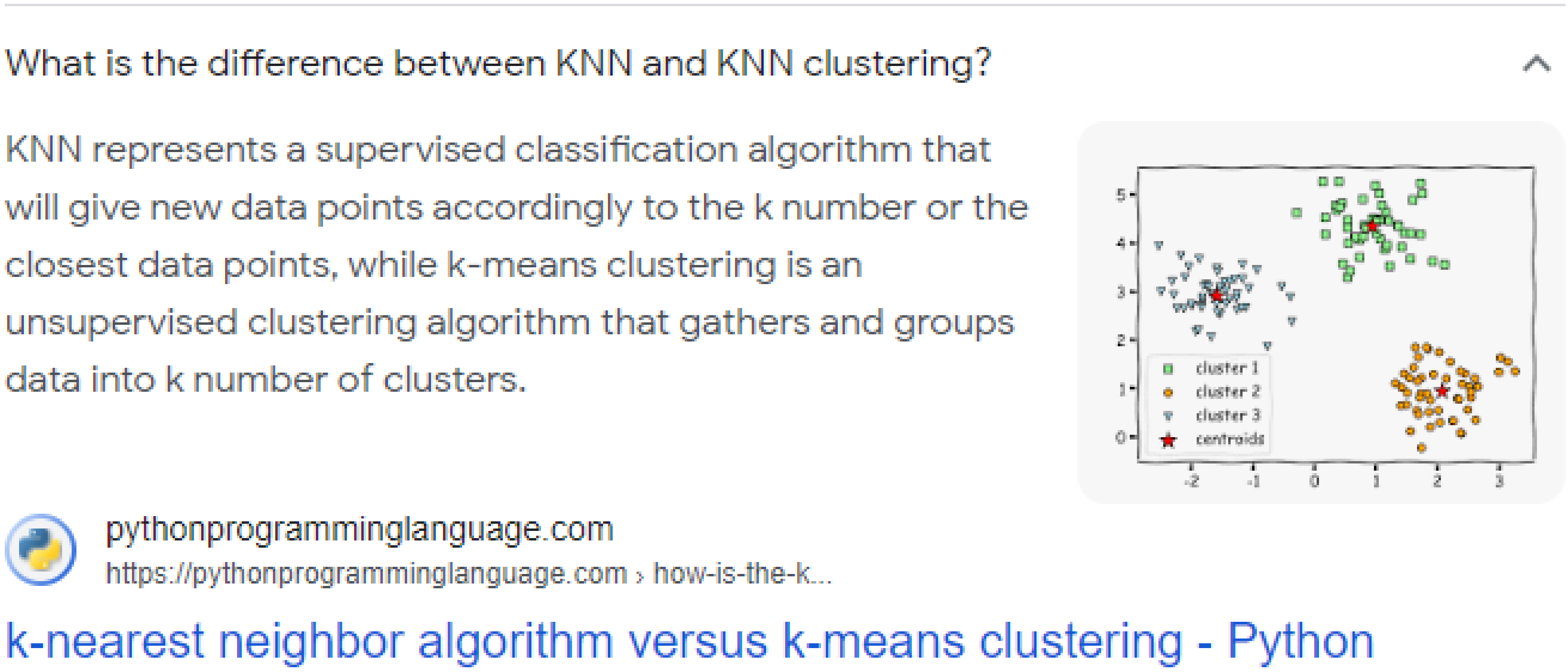
KNN Modeling/Algorithm

Assign class to a new point

- May not be perfect. https://www.researchgate.net/figure/A-simple-flowchart-for-the-| modeling_fig1_346429285


Start
Get the value of k
Consider all points and the new points in a n-dimensional space
calculate the distance of new point from all points
Sort the distance of all point
Sort the distance of all point and select the k point with smallest distance (neighbor)
Estimate the value of test point by weighted average of its neighbor
Yes
Is the error of the test points satisfying
No
Finish

KNN-Algorithm vs KNN-Clustering



What is the difference between KNN and KNN clustering?

KNN represents a supervised classification algorithm that will give new data points accordingly to the k number or the closest data points, while k-means clustering is an unsupervised clustering algorithm that gathers and groups data into k number of clusters.

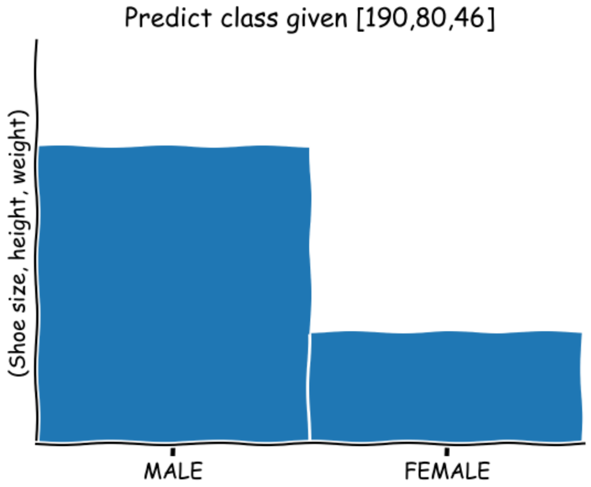
pythonprogramminglanguage.com
https://pythonprogramminglanguage.com › how-is-the-k...

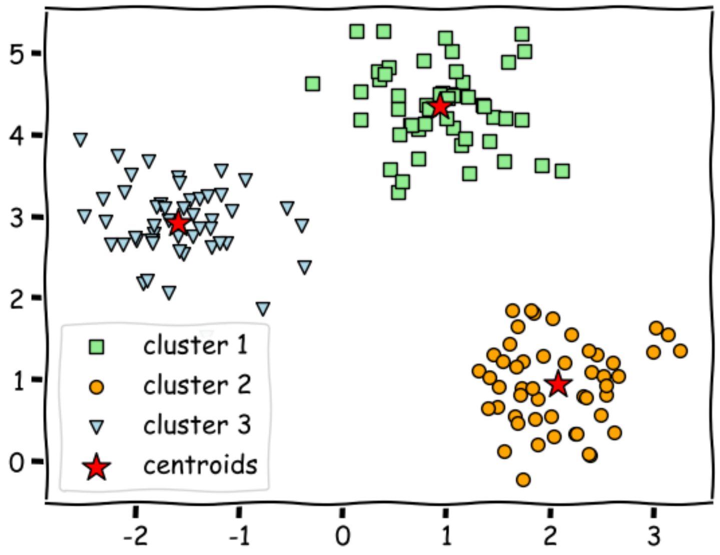
k-nearest neighbor algorithm versus k-means clustering - Python

- https://pythonprogramminglanguage.com/how-is-the-k-nearest-neighbor-algorithm-different-from-k-means-clustering

K-means Classification vs Clustering

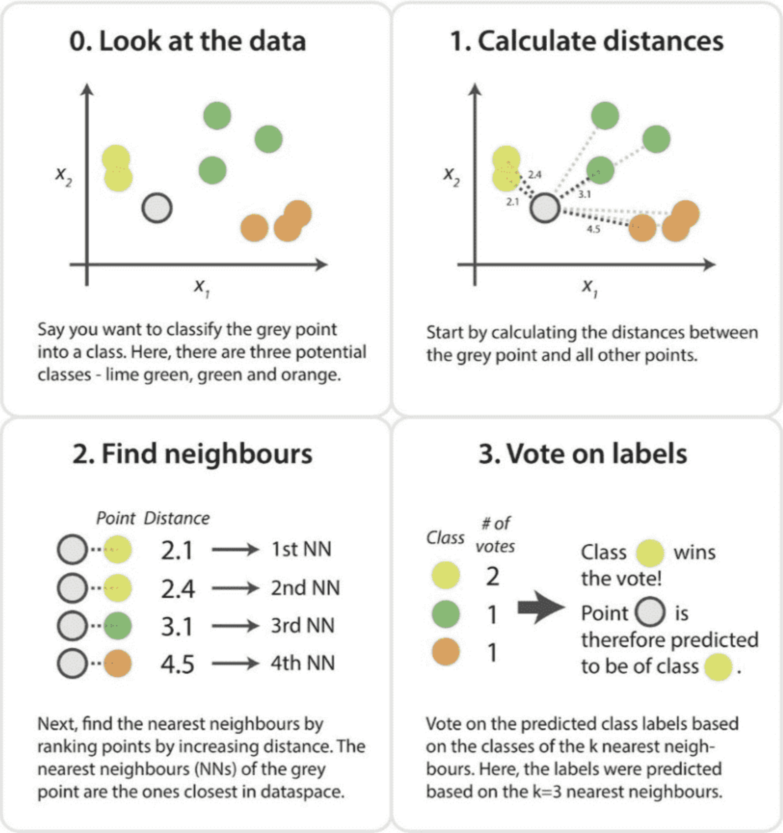
Thus, k-kmeans needs training data to make predictions.





- https://pythonprogramminglanguage.com/how-is-the-k-nearest-neighbor-algorithm-different-from-k-means-clustering

KNN - Classification



0. Look at the data

Say you want to classify the grey point into a class. Here, there are three potential classes - lime green, green and orange.

1. Calculate distances

Start by calculating the distances between the grey point and all other points.

2. Find neighbours

Point	Distance	
○··●	2.1	→ 1st NN
○··●	2.4	→ 2nd NN
○··●	3.1	→ 3rd NN
○··●	4.5	→ 4th NN

Next, find the nearest neighbours by ranking points by increasing distance. The nearest neighbours (NNs) of the grey point are the ones closest in dataspace.

3. Vote on labels

Class	# of votes
●	2
●	1
●	1

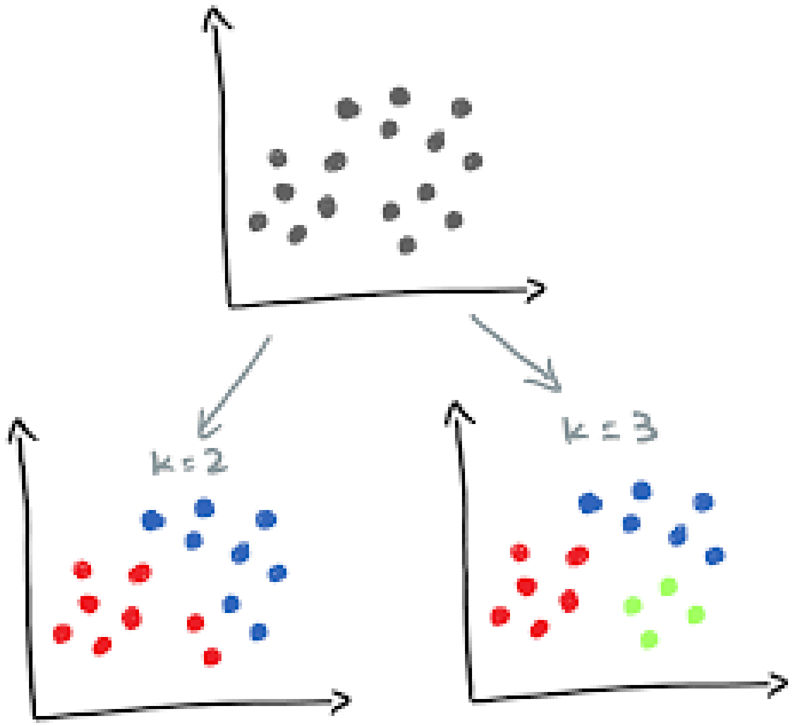
Class ● wins the vote!

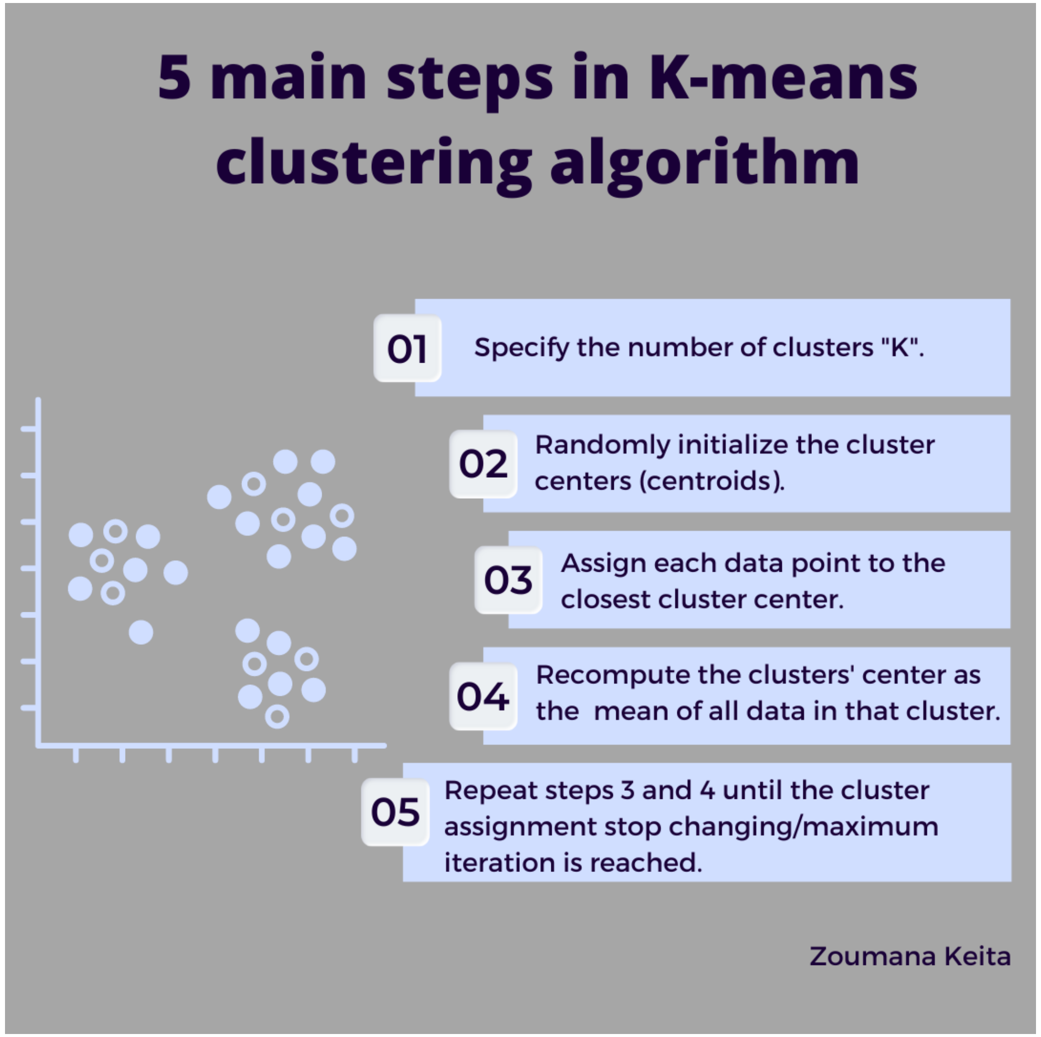
Point ○ is therefore predicted to be of class ●.

Vote on the predicted class labels based on the classes of the k nearest neighbours. Here, the labels were predicted based on the k=3 nearest neighbours.

- https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4

KNN Clustering





5 main steps in K-means clustering algorithm

01 Specify the number of clusters "K".

02 Randomly initialize the cluster centers (centroids).

03 Assign each data point to the closest cluster center.

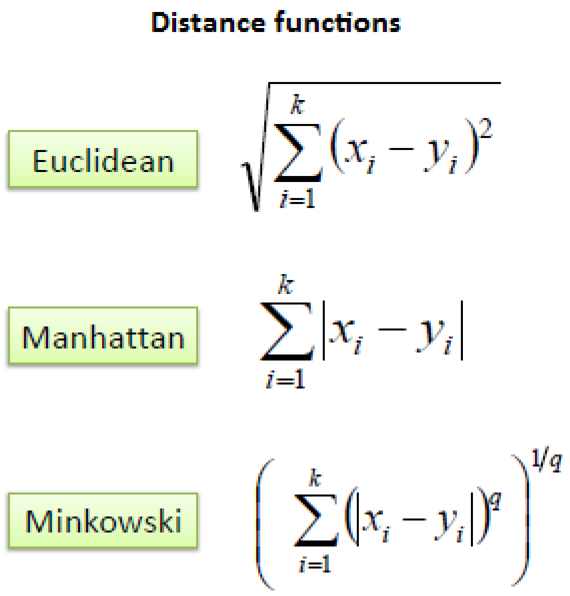
04 Recompute the clusters' center as the mean of all data in that cluster.

05 Repeat steps 3 and 4 until the cluster assignment stop changing/maximum iteration is reached.

Zoumana Keita

- https://ealizadeh.com/blog/knn-and-kmeans/ , https://towardsdatascience.com/how-to-perform-kmeans-clustering-using-python-7cc296cec092

Distance for KNN Classification



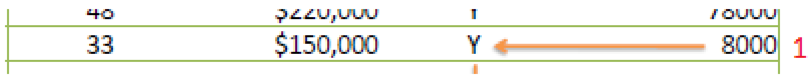
Distance functions

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$

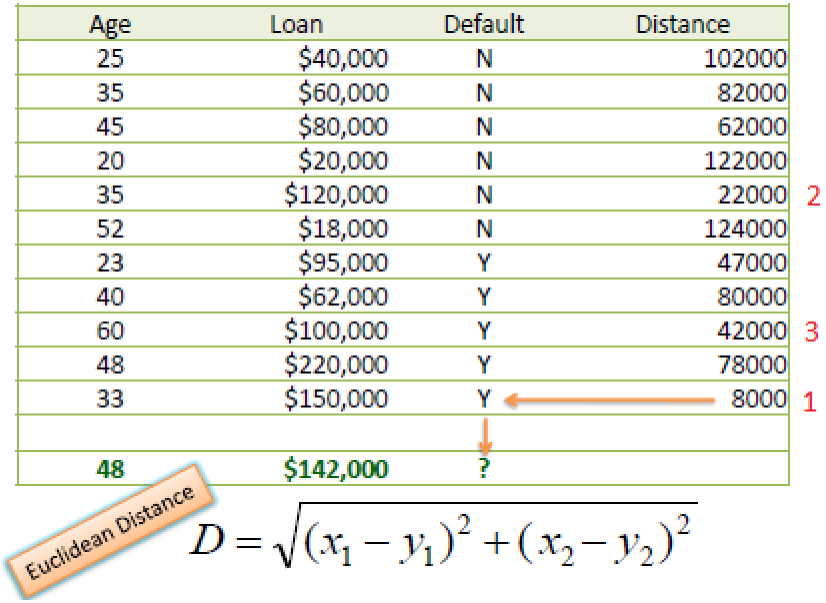
Minkowski $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

Distance from



33	$150,000	Y	8000	1

D = Sqrt[(48-33)^2 + (142000-150000)^2] = 8000.01 >> Default=Y



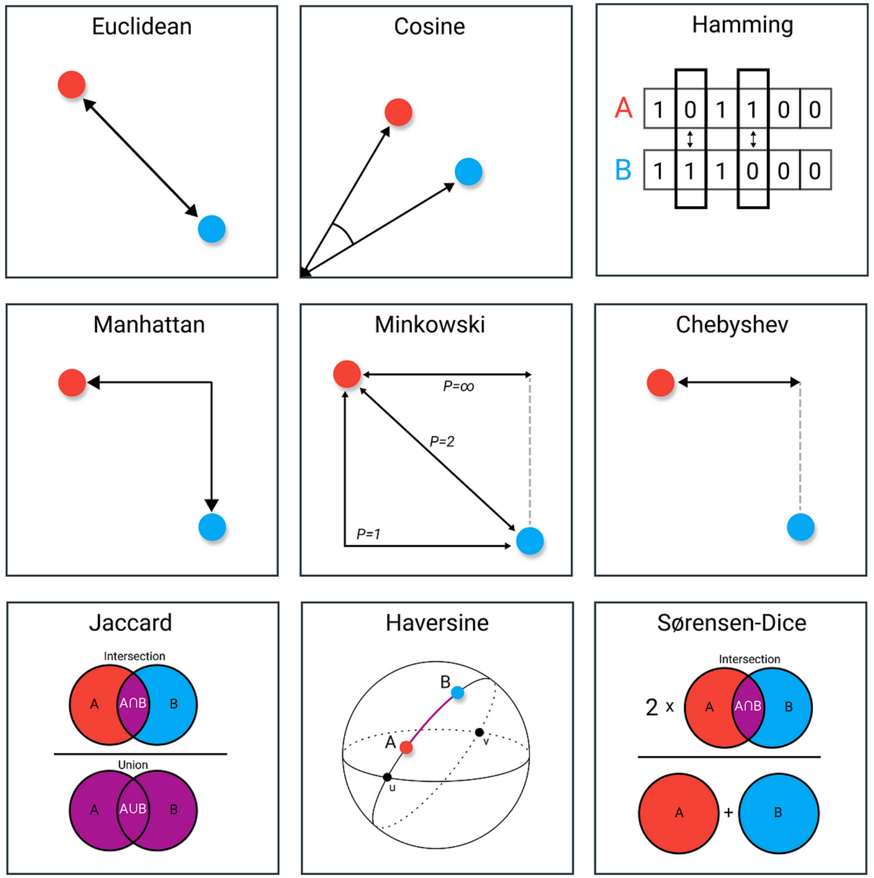
Age	Loan	Default	Distance	
25	$40,000	N	102000	
35	$60,000	N	82000	
45	$80,000	N	62000	
20	$20,000	N	122000	
35	$120,000	N	22000	2
52	$18,000	N	124000	
23	$95,000	Y	47000	
40	$62,000	Y	80000	
60	$100,000	Y	42000	3
48	$220,000	Y	78000	
33	$150,000	Y	8000	1
48	**$142,000**	**?**		

Euclidean Distance

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

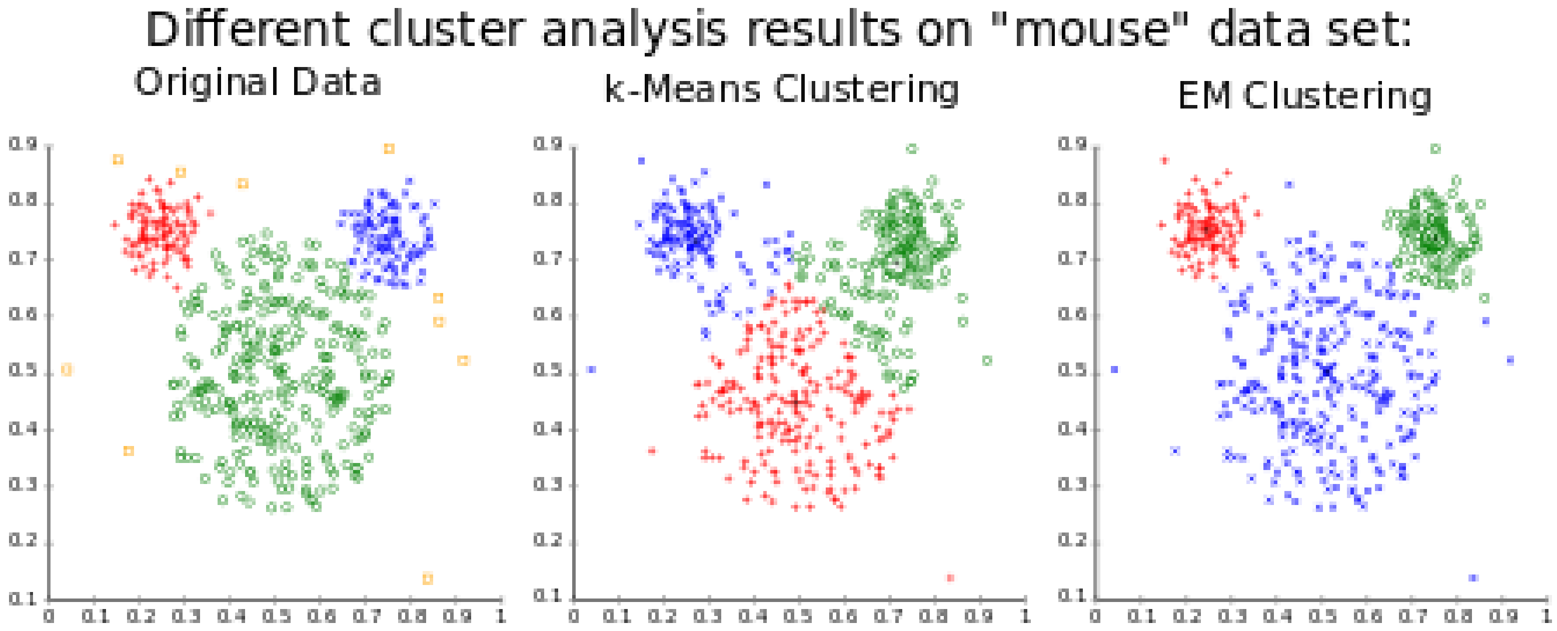
- https://www.saedsayad.com/k_nearest_neighbors.htm

9 Distance Measures



- https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa

What is K-Means?

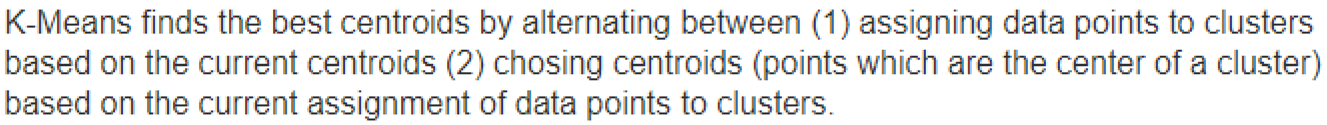


Different cluster analysis results on "mouse" data set:

Original Data

k-Means Clustering

EM Clustering



K-Means finds the best centroids by alternating between (1) assigning data points to clusters based on the current centroids (2) chosing centroids (points which are the center of a cluster) based on the current assignment of data points to clusters.

https://stanford.edu/~cpiech/cs221/handouts/kmeans.html

- https://en.wikipedia.org/wiki/K-means_clustering
- “K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science” https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning

K-Means Algorithm

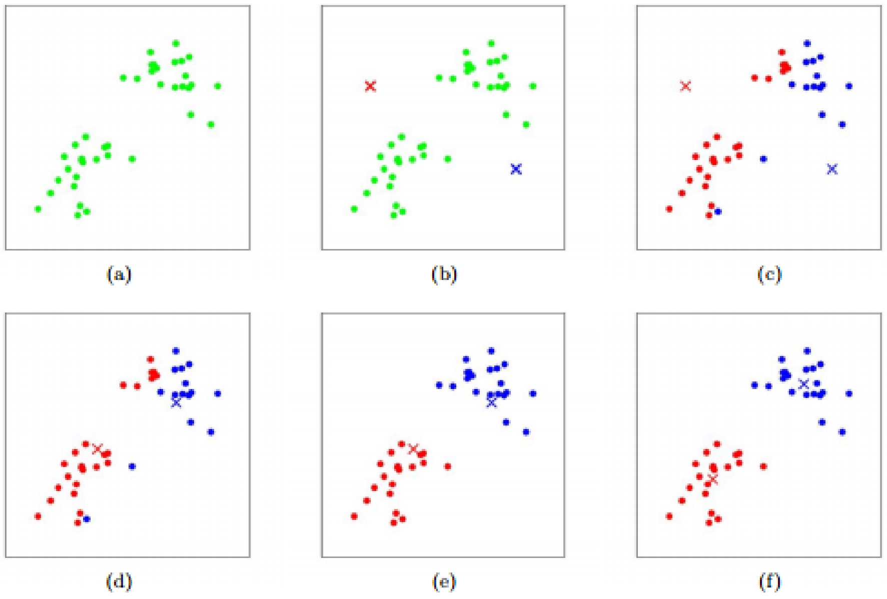
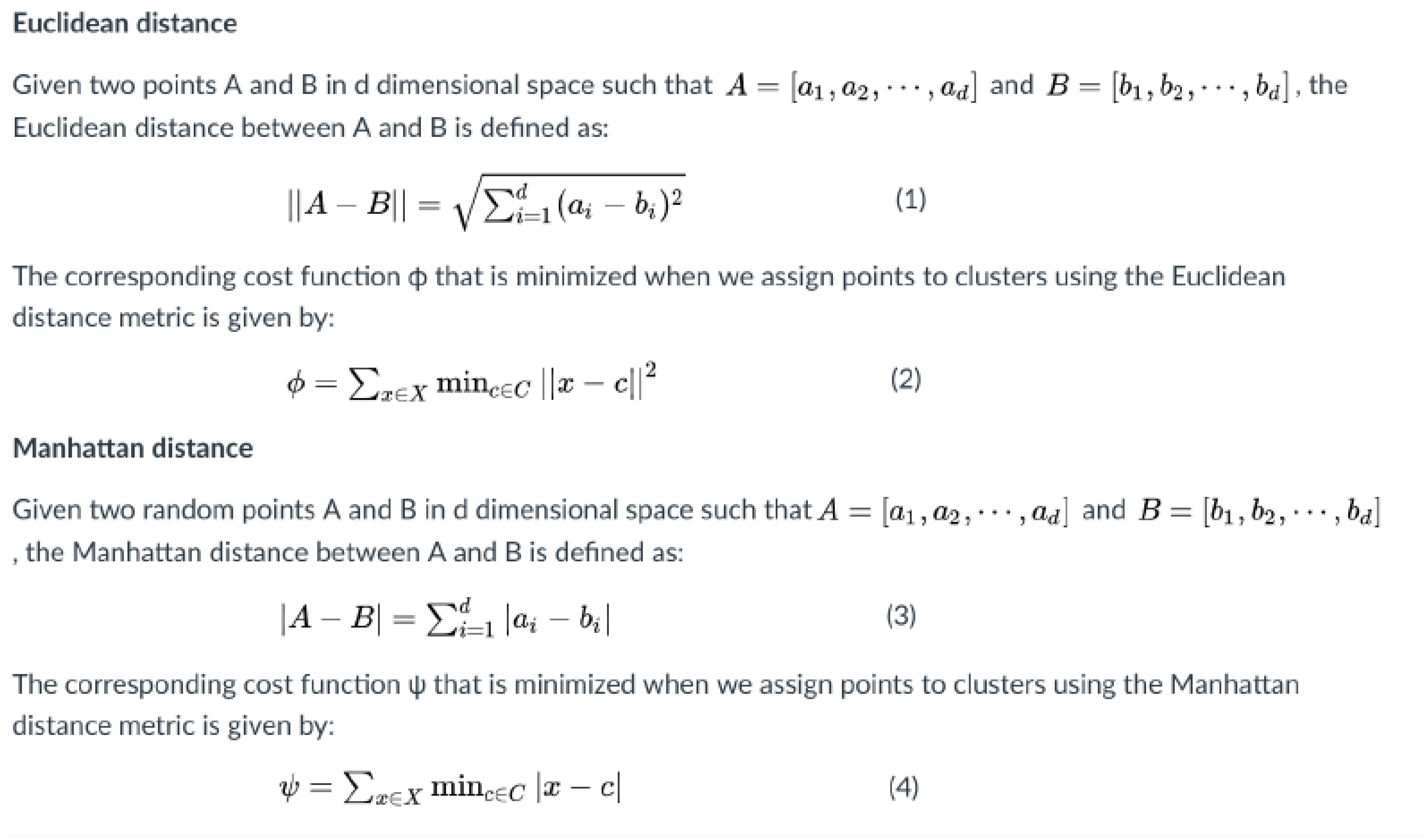


Figure 1: K-means algorithm. Training examples are shown as dots, and cluster centroids are shown as crosses. (a) Original dataset. (b) Random initial cluster centroids. (c-f) Illustration of running two iterations of k-means. In each iteration, we assign each training example to the closest cluster centroid (shown by "painting" the training examples the same color as the cluster centroid to which is assigned); then we move each cluster centroid to the mean of the points assigned to it. Images courtesy of Michael Jordan.

- https://stanford.edu/~cpiech/cs221/handouts/kmeans.html

Distances used in K-Means



Euclidean distance

Given two points A and B in d dimensional space such that $A = [a_1, a_2, \cdots, a_d]$ and $B = [b_1, b_2, \cdots, b_d]$, the Euclidean distance between A and B is defined as:

$$||A - B|| = \sqrt{\sum_{i=1}^{d} (a_i - b_i)^2} \qquad (1)$$

The corresponding cost function ϕ that is minimized when we assign points to clusters using the Euclidean distance metric is given by:

$$\phi = \sum_{x \in X} \min_{c \in C} ||x - c||^2 \qquad (2)$$

Manhattan distance

Given two random points A and B in d dimensional space such that $A = [a_1, a_2, \cdots, a_d]$ and $B = [b_1, b_2, \cdots, b_d]$, the Manhattan distance between A and B is defined as:

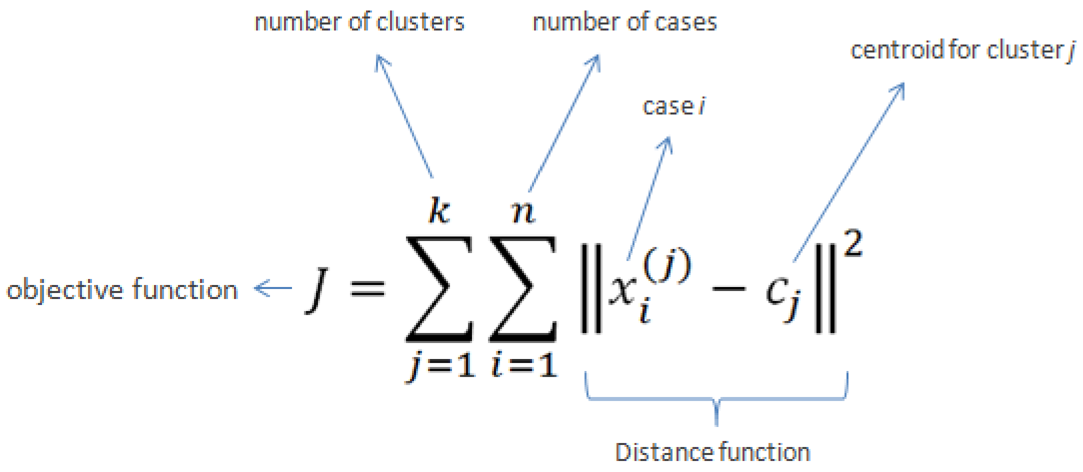
$$|A - B| = \sum_{i=1}^{d} |a_i - b_i| \qquad (3)$$

The corresponding cost function ψ that is minimized when we assign points to clusters using the Manhattan distance metric is given by:

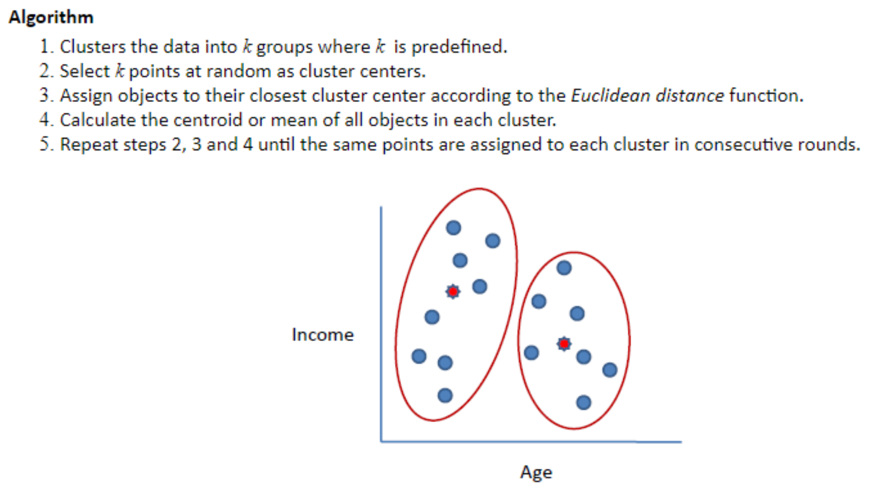
$$\psi = \sum_{x \in X} \min_{c \in C} |x - c| \qquad (4)$$

- Stackoverflow

Objective Function (to minimize) for K-Means



$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

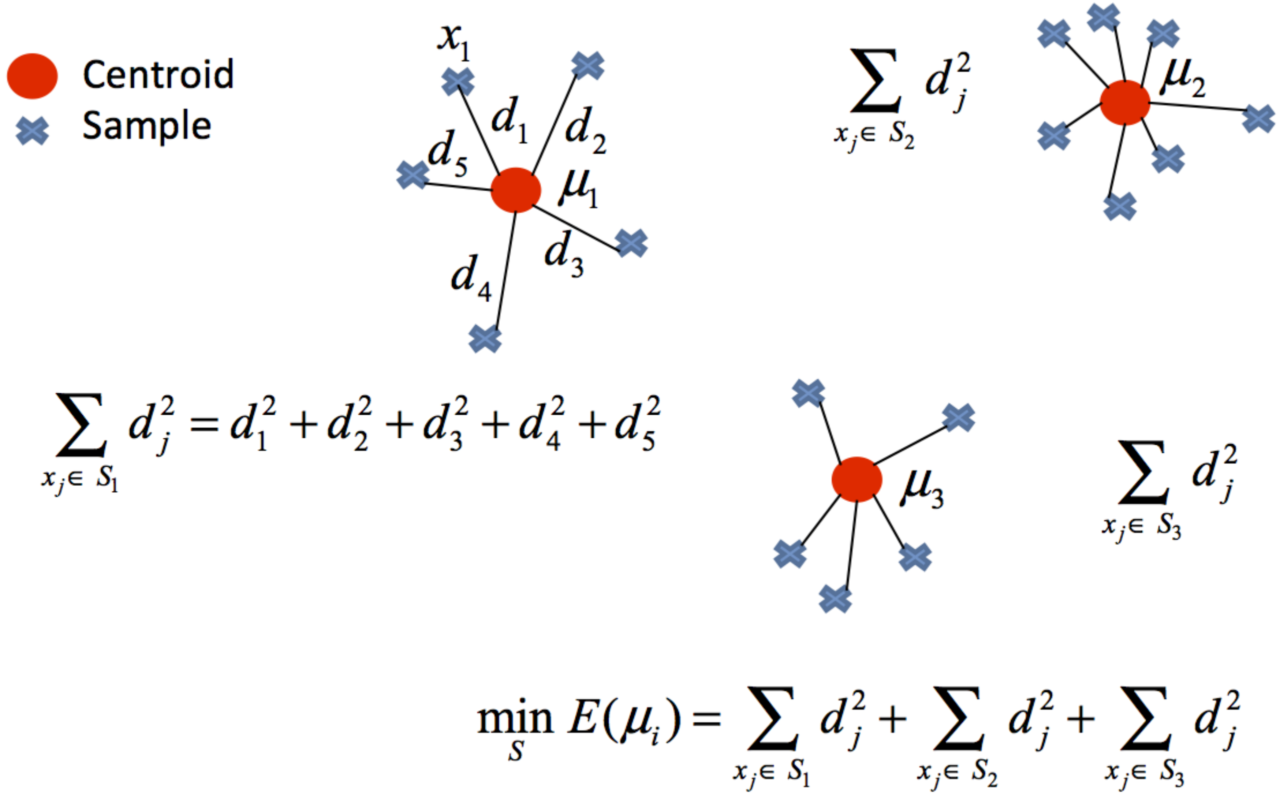


Algorithm

1. Clusters the data into *k* groups where *k* is predefined.
2. Select *k* points at random as cluster centers.
3. Assign objects to their closest cluster center according to the *Euclidean distance* function.
4. Calculate the centroid or mean of all objects in each cluster.
5. Repeat steps 2, 3 and 4 until the same points are assigned to each cluster in consecutive rounds.

- https://www.saedsayad.com/clustering_kmeans.htm

K-Means, Centroid, Minimize Objective Function



$$\sum_{x_j \in S_2} d_j^2$$

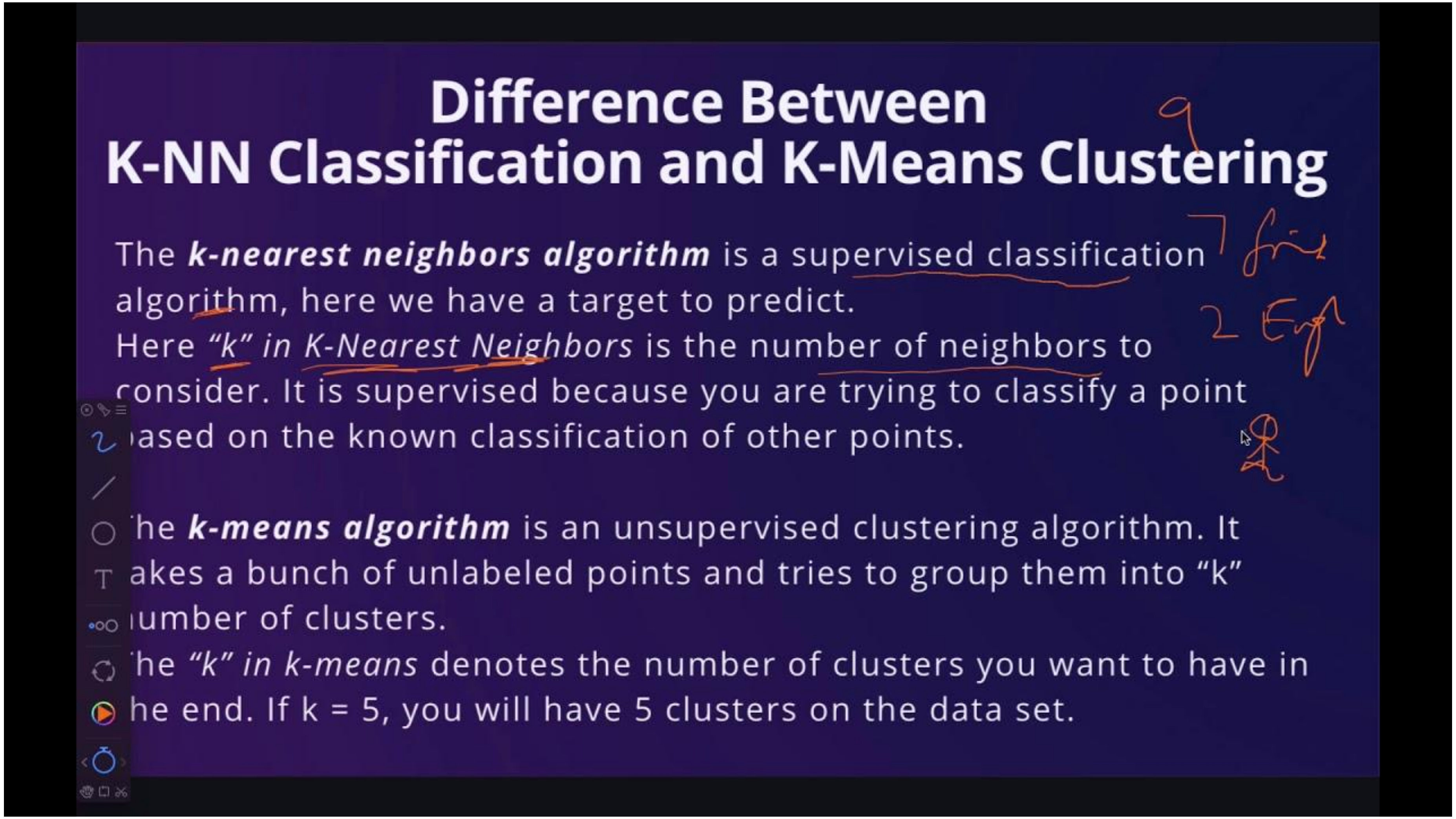
$$\sum_{x_j \in S_1} d_j^2 = d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2$$

$$\sum_{x_j \in S_3} d_j^2$$

$$\min_S E(\mu_i) = \sum_{x_j \in S_1} d_j^2 + \sum_{x_j \in S_2} d_j^2 + \sum_{x_j \in S_3} d_j^2$$

- https://www.unioviedo.es/compnum/labs/PYTHON/kmeans.html

KNN Classification vs K-Means Clustering



Difference Between K-NN Classification and K-Means Clustering

The ***k-nearest neighbors algorithm*** is a supervised classification algorithm, here we have a target to predict.
Here *“k” in K-Nearest Neighbors* is the number of neighbors to consider. It is supervised because you are trying to classify a point ased on the known classification of other points.

he ***k-means algorithm*** is an unsupervised clustering algorithm. It akes a bunch of unlabeled points and tries to group them into “k” umber of clusters.
he *“k” in k-means* denotes the number of clusters you want to have in he end. If k = 5, you will have 5 clusters on the data set.

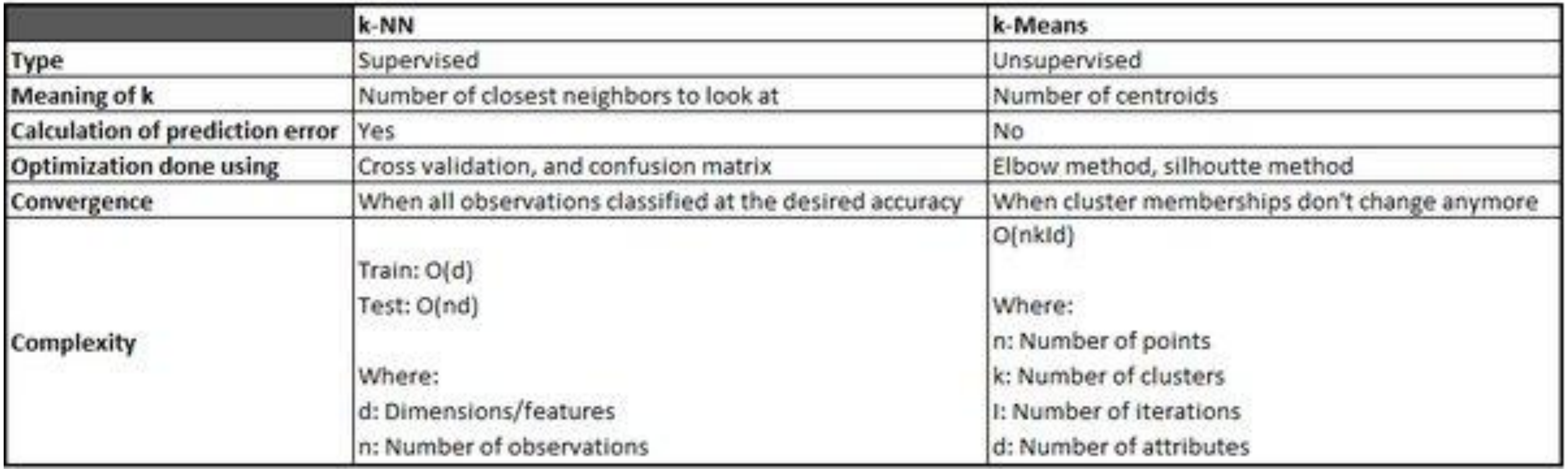
9

7 frnd

2 Engl

- https://www.youtube.com/watch?v=QMWMc5Kzq0o

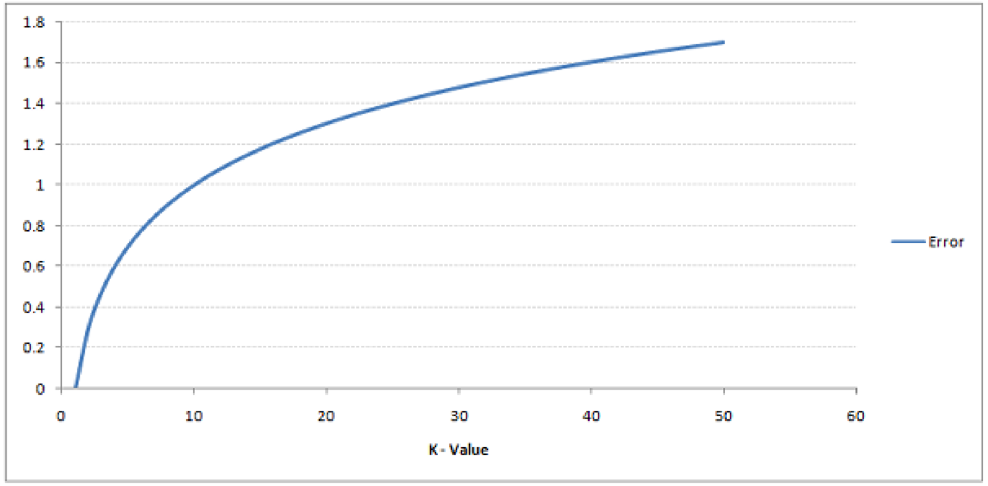
KNN Classification vs K-Means Clustering

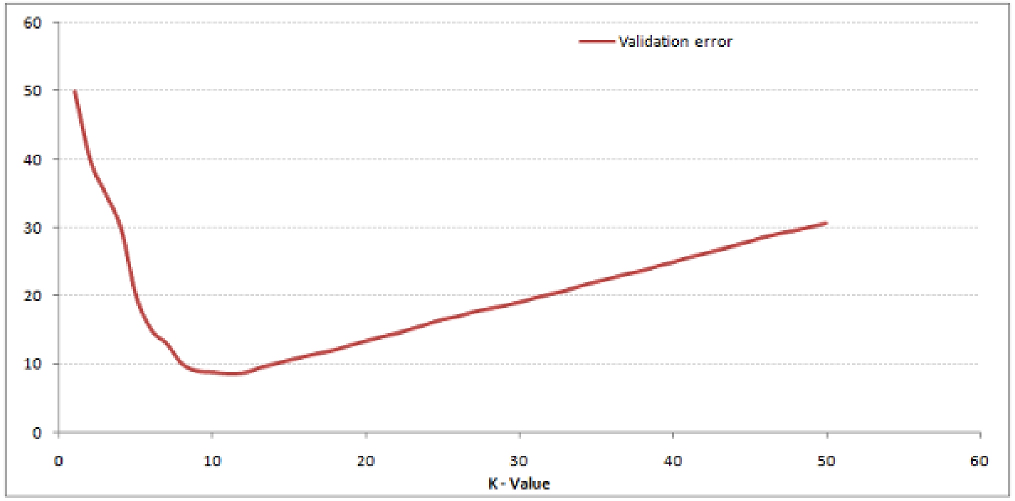


	k-NN	k-Means
Type	Supervised	Unsupervised
Meaning of k	Number of closest neighbors to look at	Number of centroids
Calculation of prediction error	Yes	No
Optimization done using	Cross validation, and confusion matrix	Elbow method, silhoutte method
Convergence	When all observations classified at the desired accuracy	When cluster memberships don't change anymore
Complexity	Train: O(d) Test: O(nd) Where: d: Dimensions/features n: Number of observations	O(nkId) Where: n: Number of points k: Number of clusters I: Number of iterations d: Number of attributes

- https://www.quora.com/What-is-the-difference-between-a-KNN-algorithm-and-a-k-means-algorithm

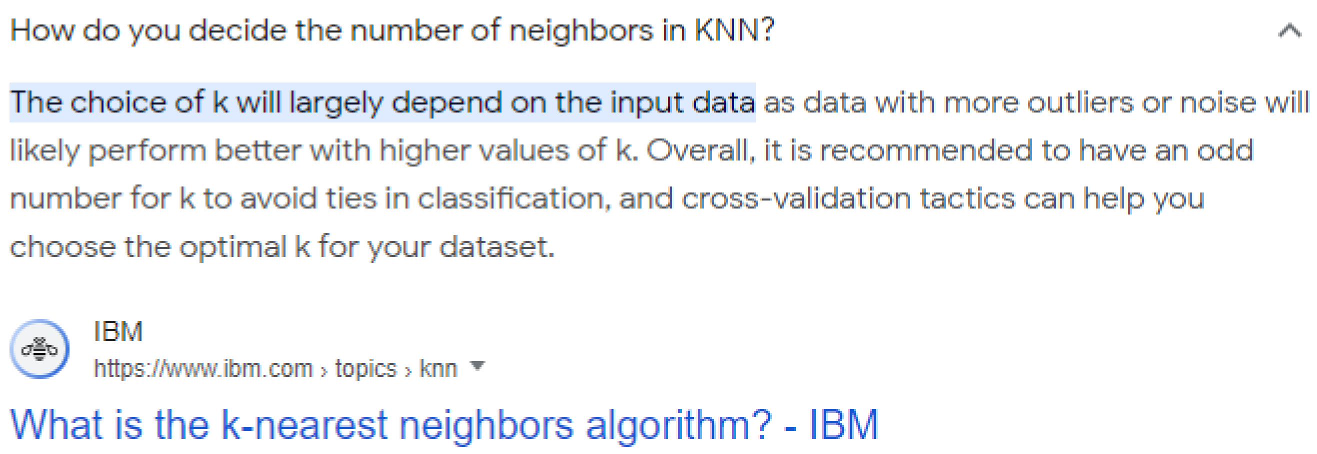
Training and Validation Error Rate with increasing K for KNN





- https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/

KNN Deciding about Best K



How do you decide the number of neighbors in KNN?

The choice of k will largely depend on the input data as data with more outliers or noise will likely perform better with higher values of k. Overall, it is recommended to have an odd number for k to avoid ties in classification, and cross-validation tactics can help you choose the optimal k for your dataset.

IBM

https://www.ibm.com › topics › knn

What is the k-nearest neighbors algorithm? - IBM

- https://www.ibm.com/topics/knn

References

- As provided in the slides
- Google Images
- Internet