"KNN - k-Nearest Neighbor Algorithm:-"

* KNN is a simple algorithm for classification

* It is lazy learner algorithm, stores all available cases & classifies new data point based on similarity measure.

→ "k" in KNN denotes no/. of nearest neighbor which are voting class of new data/test data.

(*) "Two key points":-

→ Similarity calculation - How & what metrics (Euclidean, Manhattan, Minkowski, Hamming)

→ How many similar elements should be considered for deciding the class label of each test data element?

Euclidean measure:

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

ex: $(x_1, y_1) = (3, 4)$

$(x_2, y_2) = (4, 7)$

$$(D) = \sqrt{(4-3)^2 + (7-4)^2}$$

$$(D) = \sqrt{1 + (3)^2}$$

$$(D) = \sqrt{1+9}$$

$$(D) = \sqrt{10} = 3.$$

"Algorithm":-

Input:- Training dataset, test dataset (or data points), value of 'k' to be considered.

steps:

Do for all test data points

- calculate the distance (euclidean) of test data/query point from different training data points.
- Find closest (k) training data points ie; training data points whose distances are least from test data points.

If k=1

then assign class label of training data point to test point

whichever class label is frequently present in training data points
assign class label to the test data Point.
End

Advantages of KNN :

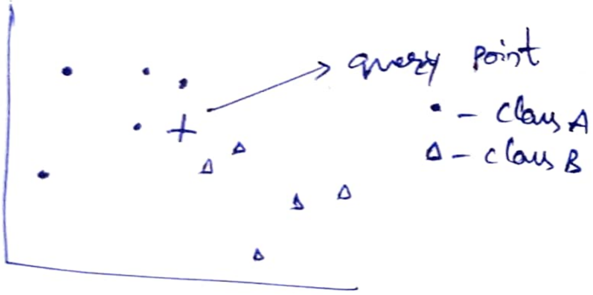
-> Simple & easy to implement
-> Very effective - recommender systems
-> very fast as no much time in training phase.

Disadvantages :-

-> Donot learn any patterns except distance based assigning of labels for classification task. data in training is important
-> large computing space for loading data.

Applications :-

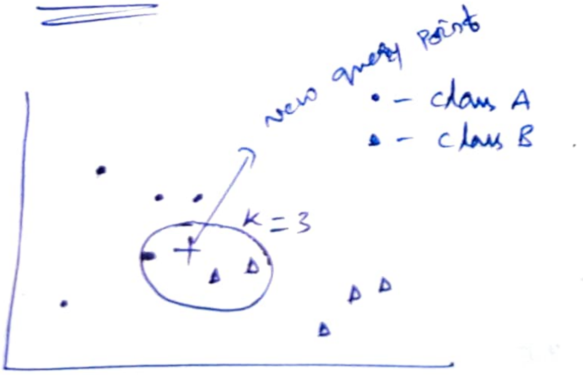
Recommender systems (Amazon) based on search.
↳ 35% of revenue from recom. systems



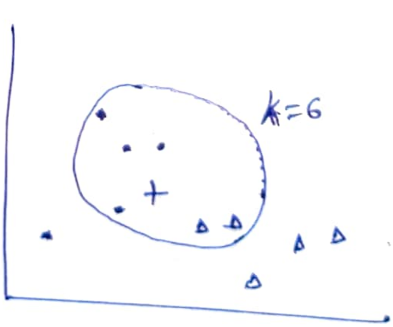
-> (K in KNN is no. of nearest neighbors)

-> KNN uses least distance measure to find the nearest neighbors.

K = 3



K = 6



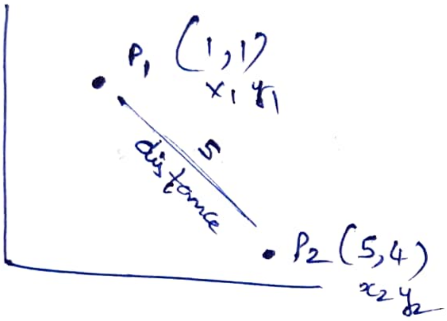
So no. of frequency is class B,
so query Point assigned label (B).

Class A is assigned to query pt.

Euclidean distance :-

- Square root of sum of differences b/w new point (x) and Existing pt/. (y).

(Ex) .



Euclidean distance (direct distance b/w 2 pts/.)

$$\sqrt{(5-1)^2+(4-1)^2} = 5$$

Manhattan distance :-
Real vector using sum of their ~~ab~~ absolute difference

$$|5-1| + |4-1| = 3+4 = 7.$$

(Ex) :-

Height (cms)	weight (kgs)	Decision (Label)	Euclidean
158	58	N	4.24
158	59	N	.
158	63	N	.
160	59	N	.
160	60	N	.
163	60	N	.
163	61	N	.
160	64	N	.
163	64	Y	.
165	61	Y	.
165	62	Y	.
165	65	Y	.
168	62	Y	.
168	63	Y	.
168	66	Y	.
170	63	Y	.
170	64	Y	.
170	68	Y	

Similarity dist. is computed for all points

Query point = Height (161) Weight (61) $\sqrt{(161-158)^2+(61-58)^2} = 4.24.$

Lets assume (k = 5) in KNN, Considers top (5) in the order.

So; for k = 5, we get (4) data points of labeling as (N), and (1) datapoint as labeling (Y).

→ Given, the query point (161 H & 61 W) → assigned labeling as(N

(Q) what if k = 6, k = 9 ....

(*) (KNN as lazy learner) :-

→ No learning is happening except storing the training data.

→ Memorizes the training data.

(Steps) of KNN :- (Coding)

→ Handle data (load) & split train & test part

→ Compute similarity based on metric chosen

→ Compute neighbors based on (k) value, locate (k) most similar data points

→ Generate the label response

→ Accuracy - summarize accuracy of predictions.