KNN

→ One of simplest ml model.

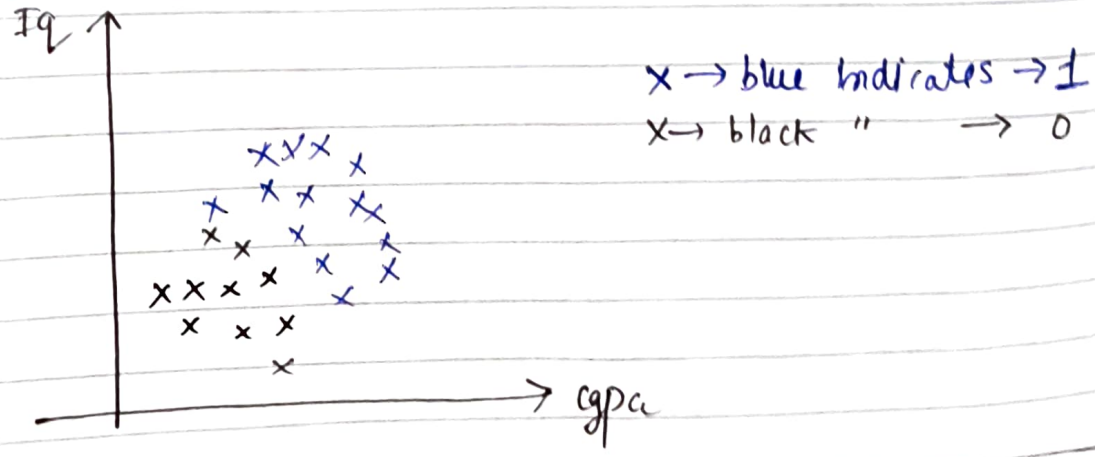
→ Understanding through example

- We have data:

cgpa	iq	placement
80	80	1
7	70	1
⋮	⋮	⋮ (100 pts)

- Make KNN model which predicts the placement value based on cgpa & iq.
- Works based on quote: "You are average of 5 people you spend the most time with."

→ plotting cgpa & iq data we have.



- KNN works on principle → ie you are like your neighbour.

Steps:

start with

- We deciding the value of K.
  - Let's say $K = 3$
- $K = 3$, means, we are considering 3 ~~neight~~ nearest neighbors
- Let's say we got a query with cgpa and iq value, we have to predict whether it will fall in blue class (1) or black class (0)
- Now, we will calculate the distance of query pt we got from each point in training data
- Sort in ascending order, benefit?
  - closest distance training point will be at top
- Since $K = 3$, we will select 3 closest distance points and apply majority count rule.
- We will look for 3 points, that in which class they fall. In blue (1) or black (0) class.
- Lets say → the 3 neighbors classes are:-

Neighbor	class
1st	1
2nd	1
3rd	0

(count) % 2 → 1

count 1 → 0

- As blue class (1) is in majority
- ∴ We will say, the given query pt will fall in blue class (1)

Note:- (1) Generally distance calculated is Euclidient distance. But there are other types of distances are too.

(2) Works in more than 2 dimensions too.

→ KNN in higher dimension

- x_1 x_2 x_3 x_4 ..... n dimension

- In this, we will have n-dimension coordinate system
- Each student will be one vector with dimensions → $x_1, x_2, x_3 \cdots x_n$ in that coordinate system
- We will get a query, We will treat it as vector too.
- And find the distance of query vector from each vector
- Choose 3 vectors with closest distance

to query vector

- check the 3 neighbor vector classes and apply majority count rule