***K Nearest Neighbour***

***(Best part ki ishe hum Classification & Regression dono mai use krskte )***

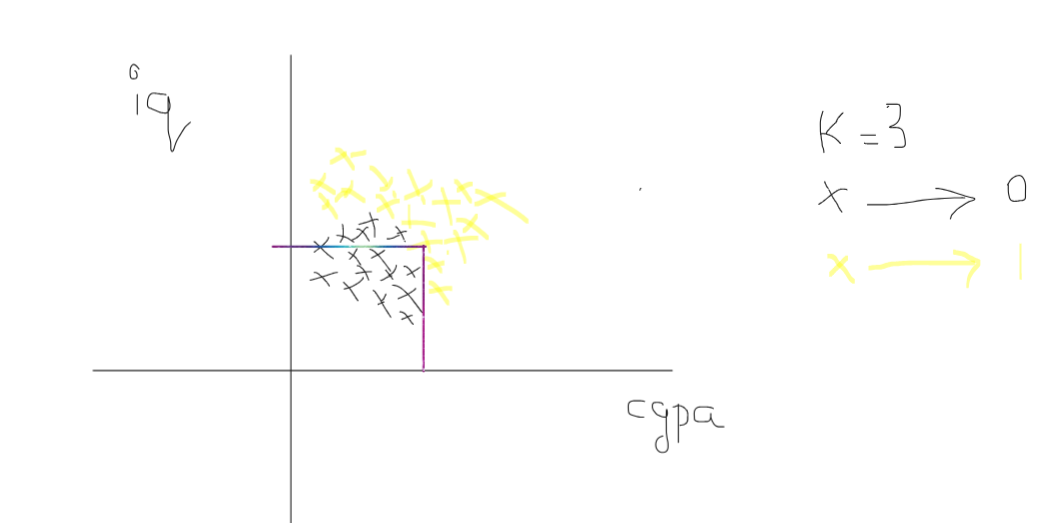
***Intuition***

**You are the average of the 5 people you spend the most time with**

* **Jim Rohn**

**Yeahi beautiful philosophy pe based hai KNN**

**It is one of the simplest & elegant ML algorithm.**

****

Abhi hum Eucledian distance nikalte hai

Humne saare points kesath Eucledian distance Calculate kiye & then we will sort all the distances.

In ascending order, jo closest point hai vo upar aajayenge

Suppose upar 3 point hai koise jo sbhse closest hai then at that time

We apply Rule known as Majority Count

It’s lyk democracy kindoff voting

Jaise closest point ka class hai

1,1,0

So the majority is 1

***So the given query point will be labelled as 1***

X1, X2, X3 ………………………… Xn

3 Vectors 🡪 Majority Count the class Label of the Query Point

Jbh bhi KNN use karo tou yaadse

Scale pe laao sbh Data ko coz, KNN pura reliye krta hai Distance pe

***How to Select K ?***

Alag alag data pe alag alag k value acha work krta hai

But proper K ko kitna rkhna hai vo humme 2 tarike se pata krskte

1. *Heuristic*
2. *Experiementation*

*Better is to Find it through Experimentation*

*Lyk through Cross Validation*

*1000 = n*

*| |*

*| |*

*800 200*

*(train) (test)*

*Abh alag alag KNN Models banayenge*

*Lyk 1 knn model hoga jiska n =1 hoga*

*& kisi KNN mai n ka value = 2 hoga & kismet 3 & so on.*

*& sbhka accuracy nikalenge*

*& jinka accuracy best hoga*

*Usko select karenge.*

***Decision Surface or Decision Boundary***

*Yeah tool har classification technique mai use mai aayega even neural networks ke tym bhi*

*Yeah Decision Surface jo hai pura coordinate system ko lyk classes mai define kardeta hai*

*& predict krneka need nahi pdta humme direct Dikh jaata hai*

*Steps ot generate this*

1. *Plot yr training data*
2. *Generate a numpy messgrid*

***There is a library called mlxtend use krke***

***We can plot decision surface.***

***Overfitting & Underfitting in KNN***

*Minor changes bhi hota hai tou kindoff rattelete hai*

*So overfitting problem aajati hai*

*That’s why if K Value is Very less then Overfitting rehne ka chance rehta hai*

*Max Value utna hie hoga jitna n hai ,*

*Let’s assume ki K = 200*

*So Hum har point ko consider krenge*

*& suppose jo Blue point hai vo zyaada hai black point se*

*So hum kahi pe bhi Query Point daalde*

*Output hamesha Blue hie aayega. As blue hamesa majority count mai jeetega’*

*This is tendency of Underfitting*

K ki bht low value will give Overfitting &

K ki bht high Value will give Underfitting

Humko kya krna hai Intermediate Value Dhundna hogga

Jo hum Experimentation ke through hie Nikaal paate hai.

**The Limitation of KNN Algorithm**

**Failure Case of KNN**

1. **Jbh we have large datasets**

E.g number of observation 5 lakh hai & number of columns 100 hai

Problem kya hai isme ?

As KNN is a lazy learning technique

Lazy learning technique means:

During training Phase koi kaam nahi hota , saara kaam prediction phase mai hota hai

Lyk in KNN what we do is saare points ko plot krdete hai

& fir 1 naya Query Point aata hai

Hum uska distance sbhkesath nikalte hai

& k ke hisabse Nearest Neighbours calculate krte hai

Then hum Majority count apply karte hai.

Yeahsbh hum tbh krte hai jbh humare pass 1 Query point hota hai

Query point mtlb jbh Hum Prediction phase mai hai

So here we see training phase mai we actually do Nothing. !

That’s why it is known as Lazy learning technique

**Problem yeah hai isme ki**

**Jo Prediction hai vo slow hojata hai when dataset bada hai.**

1. **High Dimensional Data**

Suppose High Dimension ka data hai E.g number of columns are 500

Then Curse of Dimensionality Kick maarta hai ,

Curse Dimenosnality yeahi bolta hai ki

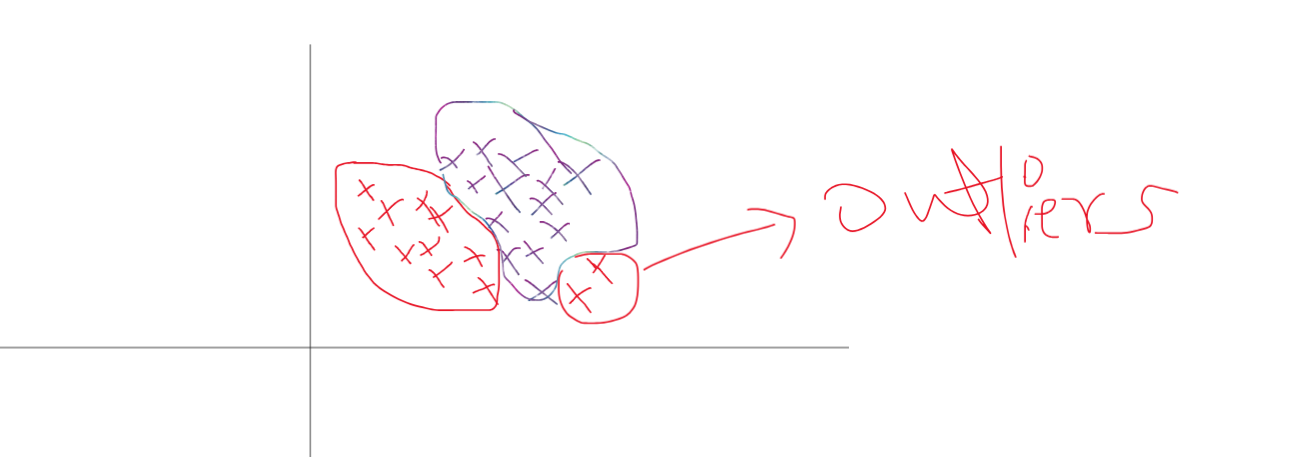
Higher dimension mai jbh Hum distances Calculate krte hai let’s say Euclidean distance

Since KNN completely relies on distance

Agar distance hie Sahi se calculate nahi hopayega then KNN ka result bhi reliable nahi hoga.

1. **Outlier Kesath acha kaam nahi krta hai KNN**

Outliers ke vajese overfitting hojayega.

****

1. **Non Homogeneous Features Scales**

**Assume There is one data lyk this**

**Experience | salary | Fire or Not**

Yaha experience 0 se 25 ka hoga but salary ka range kitna bhi hoskta hai

Lyk 1 lakh se 10 crore tak

Jbh hum graph plot krenge distance nikalneko 2 point ka

Tbhi salary vala dominate karjaayega Experience vale component ka

So the distance will not be reliable.

That’s Why Use Scaling via Standard scalar from Sklearn

1. Imbalanced Dataset

E.g:

Yes 🡪 98 %

No 🡪 2%

🡪 biased

Yaha KNN fail krjaayega bekar result dega

1. KNN isn’t good for Inference but is quite good with Prediction