## Unsupervised Machine Learning

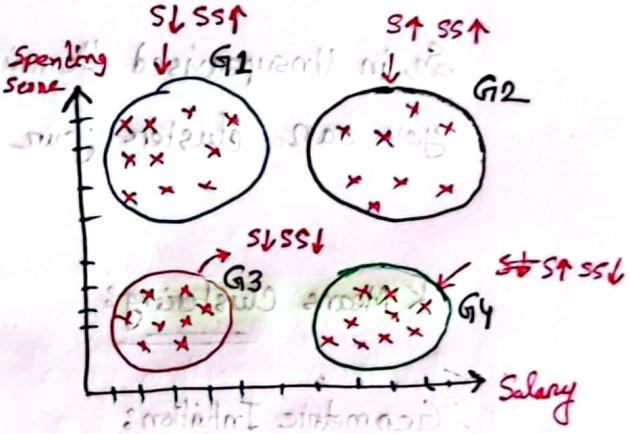
Clustering Algorithms

- K Means Clustering
- Hierarchical Clustering
- DBscan Clustering

In unsupervised machine learning we don't have target feature like supervised ML. We have features ($f_1, f_2, \ldots f_n$) and we make clusters from them.

**Example:** **Dataset:**

| Salary | Spending Score(1-10) |
|---|---|
| – | – |
| – | – |
| – | – |
| – | – |




Suppose, you have mobile shop at a shopping mall and you have to sell the latest Iphone which is 150K (BDT). Using the clustering you can decide how and which way you can sell your phone to which customer. Now, you can make your business strategy to provide discount on your phone to attract the customer and make sure that they buy your product.

For G1 → You can provide more discount as their salary is less but spending is more.

For, G2 → You can provide less discount bcz they have more salary and they also spend more.

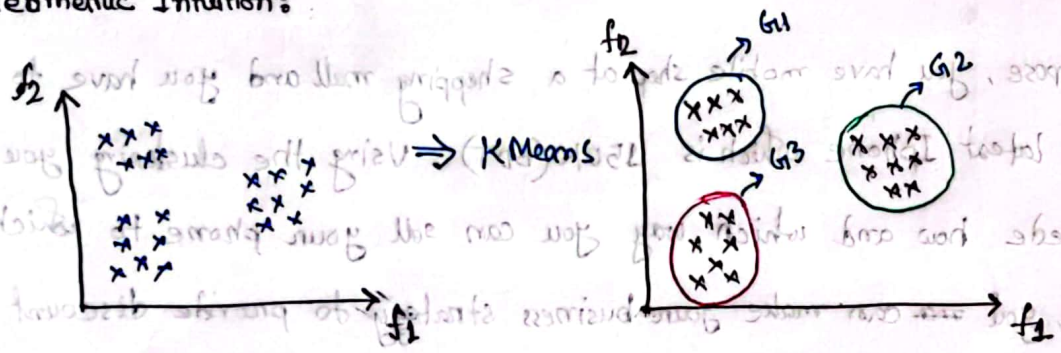
for G3 → You can do nothing. Because there is a very little probability that they can buy the phone as both their salary and spendings are less. You can try other ways like EMI.

For G4 → You can provide the max discount to them because they have higher income/salary but lower spendings. ~~but~~ By providing a great deal you can attract some of them definitely.

So, in Unsupervised Learning, you don't have to predict anything instead you can cluster your data and take important decision from them.

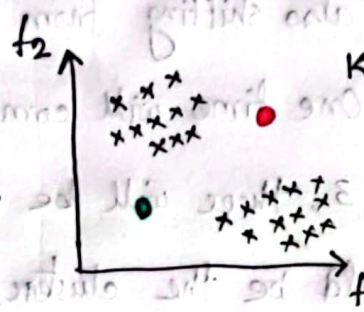
## K Means Clustering:

Geometric Intuition:

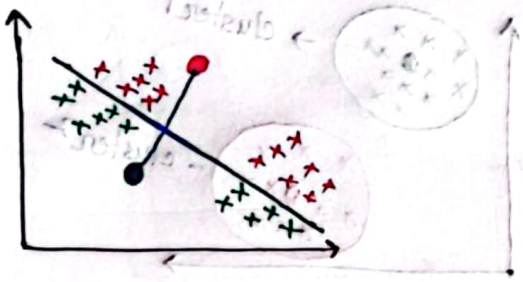


## Mathematical Intuition:

**Step 01:** Initialize some K centroids



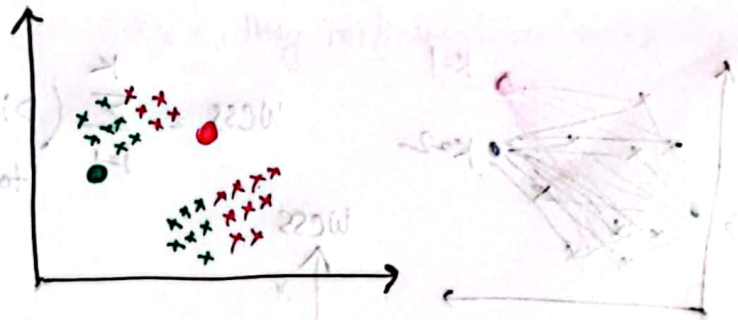

**Step 02:** Points that are nearest to the centroid, group them



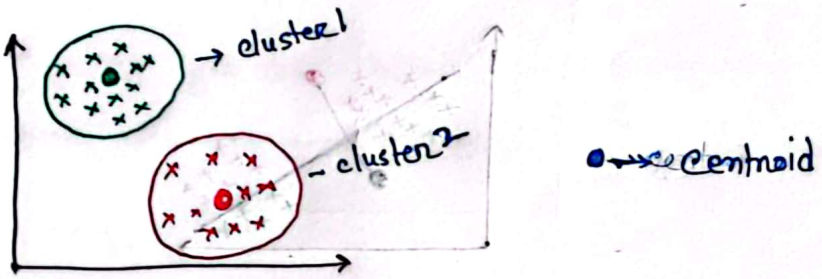
**Step 03:** Move centroids - by calculating the mean of the datapoint.

calculate the means of the red datapoints and move the red centroid according to that

calculate the mean of the green data points and move the green centroid according to that.



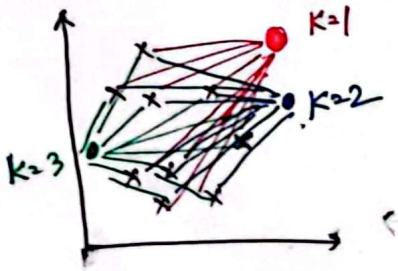
We will repeat the step 2 and step 3 again and again, so the centroid and groups will ~~be~~ keep changing because every time centroid shifting it's place, some data points are also shifting from their previous group to next group or vice versa. One time will come when after repeating the step 2 and step 3, there will be no change in the centroid and group. So that would be the cluster/Group created by K-means.



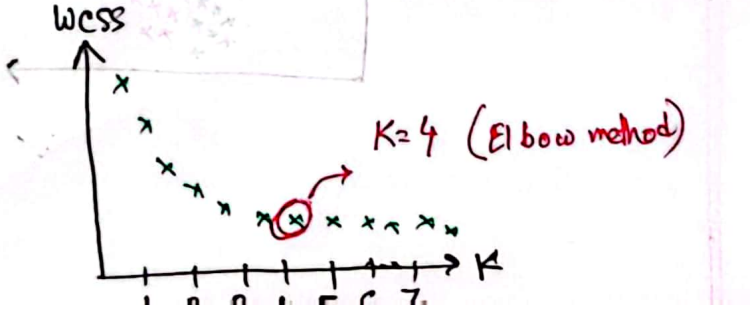

In this algorithm, first we chose K=2. How do we select K in a KMeans Algorithm?

We do that by, WCSS → Within cluster sum of square

initialize, • K = 1 to 20



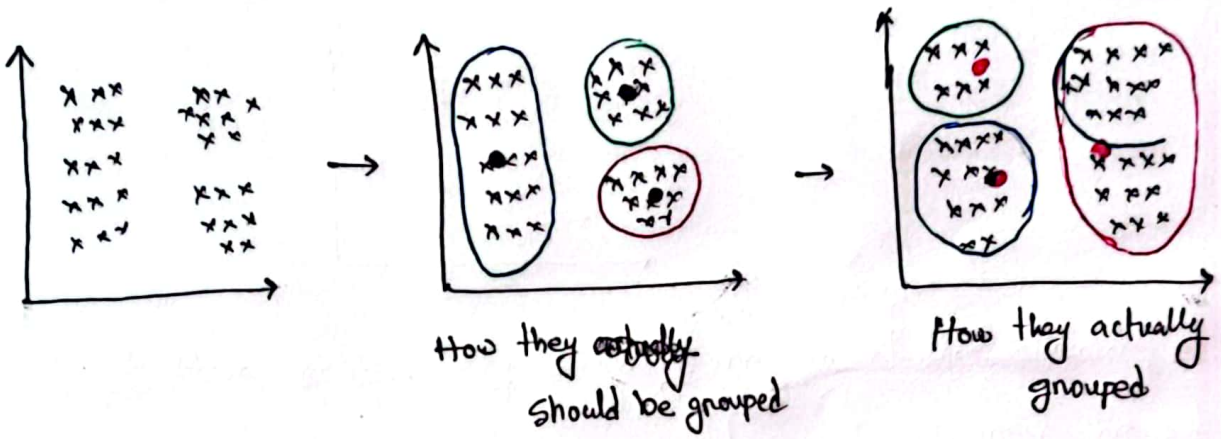

$$WCSS = \sum_{j=1}^{K} (\text{Distance between point to the nearest centroid})^2$$



We will have to take the K value from K vs wcss graph from where the space is becoming stable. For, this graph, after K=4, the ~~space~~ shape of the line is stable. So, we will take K=4. which also called the elbow method.

## Random intialization Trap: (KMeans++)

Sometimes what happens is, the clusters are not built like they should




It happens when the centroids are really close to each other.
So when the centroids are nearly initialized, this type of problem arise.
To solve this we use "KMeans++". It keeps in ~~my~~ mind that, when the centroids are initialized, they initialized in much distance so that this problem doesn't occur.