UnSupervised Machine lerning

UnSupervised machine :- It a type of machine lerning where input data don't have label

OR

a type of machine learning that analyzes and clusters unlabled datasets to discover hidden patterns structures or Relation without human intervation.

Clustering :- It is nothing ~~form~~ gathoring Similar data point from input data.

Mean :- It represent a mathamatical average ~~and also~~

$$\text{mean} = \frac{a_1 + a_2 + a_3 \cdots}{\text{number of value}}$$

Centroid :- The center point (or prototype) of a cluster representing the average position of all data points within the group

Distance :- It determines the distance between 2 data point's and on the base of that ML model perform oproation

Formula :- $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

Variation & Spread :- A mesure of a model's sensitivity to flucutation in the traning data.

Formula

$$\text{Varience} = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 \cdots (x_n - \bar{x})^2}{n}$$

Where $\bar{x}$ = Mean

K-mean's :- It's an unsupervised machine learning algorithum used to partion a dataset into K- distinct, non-overlapping super groups clu sters. The goal is to group data point so that those within the same cluster are as similar as possible while being as diffent as possible from points in other clusters

Step's in k-mean's

1. Initializing :- Choose value of Kluster and random point for serve as initial centroid.
2. Assingment :- Assing each data point to it's nearest centroid, typically using distance.
3. Update :- Recaluate each centroid
4. Repet :- Repet the assingment and update steps until the centroid no longer move or a

Sample code for k-means

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans

data = {
    "Customer" : ["Riya", "Aman", "Faizan", "Nah
                  "Imran", "Senha"],
    "Age" : [20, 30, 40, 22, 38, 45],
    "Spending" : [100, 200, 300, 110, 290, 130]
}

df = pd.DataFrame(data)

X = df[["Age", "Spending"]]

model = KMean(n_clusters=2, random_state=42,
              n_init=10)

df["Group"] = model.fit_predict(X)

plt.figure(figsize=(6,5))

for group in df["Group"].unique():
    group_data = df[df["Group"] == group]
    plt.scatter(group_data["Age"], group_data
                ["Spending"], label=f'Group{group}')

plt.xlabel("Age")
plt.ylabel("Spending")
plt.title("Customer Segmentation")
plt.legend()
plt.show

# Imported library
# Sample data
# Data frame
# Input for model
# defined model
    n_cluster = No. of cluster
    Random State = select the state 42 for constant
                   Result, in
    n_init = No of time it should be repeated
# Model traning
# Size of Graph

for loop in that passed df["Group"].unique
it means for each cluster state a unique value
and store it in group.

Note the decision of wheter it's going to be 0 or
one is decided in traning itself it's just for
the visulization purpose, that why we are using
```

Same

The for loop & differntiat with Group (A) & Gro
)
with

group_data = df[df["Group"] == group]

So If it's 0 then Group 0
If it's 1 then Group 1

and hence Graph is plotted