NAME : SYED RIAZ ALI

EMAIL : syedriazali1997@gmail.com

WHATSAPP : 03002502513

Supervised Machine Learning (K - Nearest Neighbor Algorithm)

In [2]:
```
# importing libraries

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# Loading the data set

df = pd.read_csv("mldata_Day_22.csv")
df.head()
```

Out[2]:

	age	height	weight	gender	likeness
0	27	170.688	76.0	Male	Biryani
1	41	165.000	70.0	Male	Biryani
2	29	171.000	80.0	Male	Biryani
3	27	173.000	102.0	Male	Biryani
4	29	164.000	67.0	Male	Biryani

In [3]:
```
# Replacing categorical data with numerical data in gender column by assigning value of 1 to Male and 0 to Female

df["gender"] = df["gender"].replace("Male", 1)
df["gender"] = df["gender"].replace("Female", 0)

df.head()
```

Out[3]:

	age	height	weight	gender	likeness
0	27	170.688	76.0	1	Biryani
1	41	165.000	70.0	1	Biryani
2	29	171.000	80.0	1	Biryani
3	27	173.000	102.0	1	Biryani
4	29	164.000	67.0	1	Biryani

In [4]:
```
# Selecting the input values for our Model

X = df[["age", "height", "weight", "gender"]]
y = df["likeness"]
```

In [5]:
```
# Model and prediction

from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier(n_neighbors= 5)
model.fit(X, y)
```

Out[5]:
```
KNeighborsClassifier()
```

In [6]:
```
model.predict([[24, 173, 68, 1]])
```

```
C:\Users\syedriaz\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but KNeighborsClassifier was fitted with f
eature names
  warnings.warn(
```

Out[6]:
```
array(['Biryani'], dtype=object)
```

In [7]:
```
# metrices for evaluation
# split data into test and train (80/20)

from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

X_train, X_test, y_train, y_test =  train_test_split(X, y, test_size=0.2)

# Create a model
model = KNeighborsClassifier()

# Fitting a model
model.fit(X_train, y_train)

predict_value = model.predict(X_test)
predict_value
```

Out[7]:
```
array(['Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani',
       'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Samosa', 'Biryani',
       'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani',
       'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani',
       'Samosa', 'Biryani', 'Biryani', 'Samosa', 'Biryani', 'Biryani',
       'Biryani', 'Biryani', 'Biryani', 'Samosa', 'Biryani', 'Biryani',
       'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani',
       'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani', 'Biryani',
       'Biryani'], dtype=object)
```

In [8]:
```
score = accuracy_score(y_test, predict_value)
print("The accuracy score of our model is = ", score)
```

```
The accuracy score of our model is =  0.6122448979591837
```

ASSIGNMENT

In [9]:
```
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import cross_val_score

confusion_matrix(y_test, predict_value)
```

Out[9]:
```
array([[29,  0,  2],
       [ 7,  0,  1],
       [ 9,  0,  1]], dtype=int64)
```

In [10]:
```
classification_report(y_test, predict_value)
```

```
C:\Users\syedriaz\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined and being
set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\syedriaz\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined and being
set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\syedriaz\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined and being
set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

Out[10]:
```
'              precision    recall  f1-score   support\n\n     Biryani       0.64      0.94      0.76        31\n      Pakora       0.00      0.00      0.00         8\n      Samosa
0.25      0.10      0.14        10\n\n    accuracy                           0.61        49\n   macro avg       0.30      0.35      0.30        49\nweighted avg       0.46      0.61
0.51        49\n'
```

In [ ]:

In [ ]: