

▼ K_Nearest Neighbour

```
import pandas as pd
df = pd.read_csv("mldata1.csv")
df["gender"] = df["gender"].replace("Male",1)
df["gender"] = df["gender"].replace("Female",0)
df.head()
```

| | age | height | weight | gender | likeness |
|---|-----|---------|--------|--------|----------|
| 0 | 27 | 170.688 | 76.0 | 1 | Biryani |
| 1 | 41 | 165 | 70.0 | 1 | Biryani |
| 2 | 29 | 171 | 80.0 | 1 | Biryani |
| 3 | 27 | 173 | 102.0 | 1 | Biryani |
| 4 | 29 | 164 | 67.0 | 1 | Biryani |

```
# selection of input and output variable
X = df[["weight","gender"]]
y = df["likeness"]
```

```
# Machine learning algorithm
from sklearn.neighbors import KNeighborsClassifier
# Create and fit our model
model = KNeighborsClassifier(n_neighbors=9)
model.fit(X,y)
# predict the result
predicted =model.predict([[59,1]]) # 70 Weight, 1 Male
predicted
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but KNeighborsC
warnings.warn(
array(['Biryani'], dtype=object)
```

```
# How to measure the accuracy of model
# 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 and fit a model
model = KNeighborsClassifier(n_neighbors=9).fit(X_train,y_train)
# predicting output
predicted_values = model.predict(X_test)
predicted_values
```

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

```
# checking score
score = accuracy_score(y_test, predicted_values)
score
```

```
0.6326530612244898
```

[Colab paid products](#) - [Cancel contracts here](#)

✓ 0s completed at 1:15 AM

