

k_nearest_neighbours

January 27, 2022

```
[ ]: import pandas as pd
df = pd.read_csv("mldata_dtc.csv")
df.head(1)
```

```
[ ]:   age  height  weight gender likeness
0    27  170.688    76.0   Male  Biryani
```

0.0.1 Convert gender (M/F) to 1 and 0

```
[ ]: df['gender'] = df['gender'].replace("Male",1)
df['gender'] = df['gender'].replace("Female",0)
df.tail(1)
```

```
[ ]:   age  height  weight  gender likeness
244   33   157.0    56.0         0  Samosa
```

```
[ ]: X=df[['weight','gender','age']]
#print("the value in X feature is ",X.head(3))
y=df['likeness']
#print("the value in y output is ",y.head(3))
```

```
[ ]: #machine learning algorithm
from sklearn.neighbors import KNeighborsClassifier
# create and fit model
model = KNeighborsClassifier(n_neighbors=5).fit(X,y)
#Prediction
model.predict([[23,0,23]])
```

```
C:\Users\dell7450\AppData\Local\Programs\Python\Python310\lib\site-
packages\sklearn\base.py:450: UserWarning: X does not have valid feature names,
but KNeighborsClassifier was fitted with feature names
warnings.warn(
```

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

0.0.2 accuracy evaluation (SPlit 80-20)

Metrics for Evaluation of Classification Data

```
[ ]: # accuracy by splitting
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# split syntax
X_train,X_test,y_train,y_test =train_test_split(X, y, test_size=0.2)
#Creating and model fitting
model = KNeighborsClassifier(n_neighbors=5)
model.fit(X_train,y_train)
# checking predicted values with input test data
predicted_values = model.predict(X_test)
print("The predicted values from 20% of test input is",predicted_values,"\n")
```

```
The predicted values from 20% of test input is ['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' 'Biryani'
'Biryani' 'Biryani' 'Biryani' 'Biryani' 'Biryani' 'Biryani' 'Biryani'
'Biryani' 'Biryani' 'Biryani' 'Pakora' 'Biryani' 'Biryani' 'Biryani']
```

0.0.3 Accuracy Score Checking

```
[ ]: #Now compare y_test values with the values of y_test(predicted)
score = accuracy_score(y_test,predicted_values)
print("The accuracy score of model when compared with twenty percent test_
↪values is",score)
```

The accuracy score of model when compared with twenty percent test values is
0.6122448979591837

0.0.4 Top k-Accuracy Score Checking

```
[ ]: # from sklearn.metrics import top_k_accuracy_score
# top_k_accuracy_score(y_test, predicted_values, k=2, normalize=False)

# ## this will generate error because it work on Numeric Data. I need to_
↪convert Biryani Samosa Pakora into 1,2,3
# # This is my future task
```

0.0.5 Confusion Matrix

It is really a big confusion

```
[ ]: from sklearn.metrics import confusion_matrix
      confusion_matrix(y_test, predicted_values)
```

```
[ ]: array([[29,  0,  0],
           [ 6,  1,  0],
           [13,  0,  0]], dtype=int64)
```

```
[ ]: tn, fp, fn, tp ,tn1, fp1, fn1, tp1, tp2 = confusion_matrix(y_test,
      ↪predicted_values).ravel()
      tn, fp, fn, tp ,tn1, fp1, fn1, tp1, tp2
```

```
[ ]: (29, 0, 0, 6, 1, 0, 13, 0, 0)
```

```
[ ]:
```