

```

import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
# Importing the dataset
data = pd.read_csv('train.csv')

# Splitting the dataset into Training set and Test set
from sklearn.model_selection import train_test_split
Y = data.fetal_health
X = data.drop('fetal_health',axis=1)
X_train, X_test, Y_train, Y_test = train_test_split( X ,Y, test_size = 0.2, random_state = 10

from sklearn.neighbors import KNeighborsClassifier
model = KNeighborsClassifier(n_neighbors=7)
model.fit(X_train, Y_train)
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=7, p=2,
                     weights='uniform')

☞ KNeighborsClassifier(n_neighbors=7)

print('X_train', X_train.shape)
print('y_train', Y_train.shape)

print('X_test', X_test.shape)
print('y_test', Y_test.shape)

X_train (1360, 21)
y_train (1360,)
X_test (340, 21)
y_test (340,)

#Predict the response for test dataset
y_pred =model.predict(X_test)

y_pred

```

```

array([1., 3., 2., 1., 1., 2., 2., 2., 2., 1., 1., 1., 2., 1., 1., 1., 1.,
       1., 1., 2., 2., 1., 1., 2., 2., 1., 1., 1., 1., 1., 1., 2., 1.,
       1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1., 1., 1., 2., 1., 3.,
       1., 1., 1., 1., 1., 1., 3., 1., 1., 1., 2., 2., 2., 1., 1., 1., 1.,
       1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 2., 1., 1., 1., 2., 1., 1.,
       1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 2., 1., 1., 3., 2., 1.,

```

```

3., 1., 1., 1., 1., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
2., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 2., 3., 3., 1., 1.,
1., 1., 1., 1., 3., 1., 1., 1., 1., 1., 2., 1., 2., 1., 2., 1., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1.,
1., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 3., 2., 1., 1., 2.,
1., 3., 3., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 2., 1.,
3., 1., 3., 1., 1., 2., 2., 1., 1., 1., 2., 1., 1., 1., 3., 1., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 3., 1., 2., 1., 1., 1., 1., 3.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2.,
1., 2., 2., 1., 1., 3., 2., 1., 1., 1., 1., 3., 3., 1., 1., 1., 1.,
1., 2., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 2., 1., 1., 1., 1.,
1., 1., 1., 1., 1., 3., 1., 1., 1., 3., 1., 1., 1., 3., 1., 1., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1.,
1., 1., 2., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1., 1., 1., 1., 1.]

```

```

#Import scikit-learn metrics module for accuracy calculation
from sklearn.metrics import accuracy_score
# Model Accuracy, how often is the classifier correct?
print("Accuracy: {:.0f}".format(accuracy_score(Y_test, y_pred)*100))

```

Accuracy: 90

```

Validation_test = pd.read_csv('test.csv')
print('Validation_test',Validation_test.shape)

```

Validation_test (426, 21)

```
val_pred =model.predict(Validation_test)
```

val_pred

```

array([1., 1., 2., 1., 2., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2.,
3., 1., 2., 1., 3., 2., 1., 1., 3., 1., 1., 1., 3., 2., 1., 1., 2.,
3., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1., 1., 3., 1., 1., 1.,
1., 1., 1., 1., 1., 1., 1., 3., 2., 1., 1., 1., 1., 1., 2., 1., 1.,
1., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 3., 1., 1., 1.,
1., 1., 1., 1., 1., 3., 1., 1., 2., 1., 2., 1., 3., 1., 1., 3., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1., 1., 1., 1., 1.,
1., 1., 1., 1., 3., 1., 1., 1., 1., 1., 2., 1., 1., 2., 2., 1., 1.,
1., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 3., 1.,
2., 1., 3., 2., 1., 1., 1., 1., 3., 1., 2., 1., 1., 1., 1., 1.,
3., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1.,
1., 1., 3., 2., 1., 2., 1., 1., 1., 2., 1., 1., 3., 1., 1., 1., 2.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 3., 1., 1., 1.,
1., 2., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 1., 1., 1., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
3., 1., 3., 1., 1., 1., 1., 1., 1., 1., 2., 1., 1., 1., 3., 1.,
1., 1., 1., 1., 2., 1., 1., 2., 1., 1., 1., 1., 3., 1., 1.,
1., 1., 1., 1., 1., 3., 1., 2., 1., 1., 3., 1., 1., 1., 3., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,

```

```
1., 1., 2., 1., 1., 1., 2., 1., 2., 1., 1., 2., 1., 1., 1., 1., 1.,
1., 1., 1., 2., 1., 1., 2., 1., 2., 1., 2., 1., 1., 1., 1., 1.,
1., 3., 3., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 1.,
2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2.,
1., 1., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 3., 1., 1., 1., 1.,
1., 1., 1., 1., 1., 1., 1., 1., 3., 1., 1., 1., 3., 1., 1., 1., 1.,
1.]])
```

```
output=[val_pred]
df=pd.DataFrame(output)
new=df.to_csv(r'output.csv',index=None,header=True)
print(df)
```

```
      0      1      2      3      4      5      6      7      8      9      ...    416  417  418  419  \
0  1.0  1.0  2.0  1.0  2.0  2.0  1.0  1.0  1.0  1.0  ...    3.0  1.0  1.0  1.0

      420  421  422  423  424  425
0  3.0  1.0  1.0  1.0  1.0  1.0

[1 rows x 426 columns]
```

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

✓ 0s completed at 20:44

