

```
In [11]: from pandas import read_csv
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import cross_val_predict
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from sklearn.neural_network import MLPClassifier
import warnings
warnings.filterwarnings("ignore")
```

```
In [12]: df=read_csv('Raisin.csv')
array=df.values
x=array[:,0:-1]
y=array[:, -1]
kfold=KFold(n_splits=10,shuffle=True,random_state=0)
```

```
In [13]: from sklearn.preprocessing import StandardScaler
scaler=StandardScaler().fit(x)
X_scaled=scaler.transform(x)
```

```
In [14]: model=MLPClassifier(hidden_layer_sizes=(10,9),activation='relu',shuffle=True,random_state=0,max_iter=100)
result=cross_val_score(model,X_scaled,y,cv=kfold,scoring='accuracy')
print("(accuracy={:.2f}%)".format(result.mean()*100))
y_pred =cross_val_predict(model,X_scaled,y,cv=kfold)
conf_mat=confusion_matrix(y,y_pred)
print('\n',conf_mat)
report=classification_report(y,y_pred)
print('\n',report)
```

(accuracy=86.67%)

```
[[375  75]
 [ 45 405]]
```

	precision	recall	f1-score	support
Besni	0.89	0.83	0.86	450
Kecimen	0.84	0.90	0.87	450
accuracy			0.87	900
macro avg	0.87	0.87	0.87	900
weighted avg	0.87	0.87	0.87	900

```
In [15]: model=model.fit(x,y)
y_perd=model.predict([[140897,588.0709185,308.0257871,0.851847299,146231,0.733999104,1536.473]])
print(y_perd)

['Besni']
```

```
In [ ]:
```