```
import pandas as pd
from sklearn.datasets import load_wine
```

In [3]:

```
wine = load_wine()
```

In [4]:

```
wine
```

Out[4]:

```
{'data': array([[1.423e+01, 1.710e+00, 2.430e+00, ..., 1.040e+00, 3.920e+00,
         1.065e+03],
        [1.320e+01, 1.780e+00, 2.140e+00, ..., 1.050e+00, 3.400e+00,
         1.050e+03],
        [1.316e+01, 2.360e+00, 2.670e+00, ..., 1.030e+00, 3.170e+00,
         1.185e+03],
        ...,
        [1.327e+01, 4.280e+00, 2.260e+00, ..., 5.900e-01, 1.560e+00,
         8.350e+02],
        [1.317e+01, 2.590e+00, 2.370e+00, ..., 6.000e-01, 1.620e+00,
         8.400e+02],
        [1.413e+01, 4.100e+00, 2.740e+00, ..., 6.100e-01, 1.600e+00,
         5.600e+02]]),
 'target': array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2]),
 'frame': None,
 'target_names': array(['class_0', 'class_1', 'class_2'], dtype='<U7'),
 'DESCR': '.. _wine_dataset:\n\nWine recognition dataset\n------------------------\n\
n**Data Set Characteristics:**\n\n    :Number of Instances: 178\n    :Number of
Attributes: 13 numeric, predictive attributes and the class\n    :Attribute Information:\n
\t\t- Alcohol\n \t\t- Malic acid\n \t\t- Ash\n\t\t- Alcalinity of ash  \n \t\t- Magnesium\
n\t\t- Total phenols\n \t\t- Flavanoids\n \t\t- Nonflavanoid phenols\n \t\t-
Proanthocyanins\n\t\t- Color intensity\n \t\t- Hue\n \t\t- OD280/OD315 of diluted wines\
n \t\t- Proline\n\n    - class:\n            - class_0\n            - class_1\n
- class_2\n\t\t\n    :Summary Statistics:\n    \n    ============================== ====
===== ======= =====\n                                      Min   Max   Mean     SD\n
============================== ==== ===== ======= =====\n    Alcohol:
11.0  14.8    13.0   0.8\n    Malic Acid:                   0.74  5.80    2.34  1.12\n
Ash:                          1.36  3.23    2.36  0.27\n    Alcalinity of Ash:
10.6  30.0    19.5   3.3\n    Magnesium:                    70.0 162.0    99.7  14.3\n
Total Phenols:                0.98  3.88    2.29  0.63\n    Flavanoids:
```

```
0.34  5.08    2.03  1.00\n    Nonflavanoid Phenols:         0.13  0.66    0.36  0.12\n
Proanthocyanins:              0.41  3.58    1.59  0.57\n    Colour Intensity:
1.3  13.0     5.1   2.3\n    Hue:                          0.48  1.71    0.96  0.23\n
OD280/OD315 of diluted wines: 1.27  4.00    2.61  0.71\n    Proline:
278  1680     746   315\n    ============================ ==== ===== ======= =====\n\n
:Missing Attribute Values: None\n    :Class Distribution: class_0 (59), class_1 (71),
class_2 (48)\n    :Creator: R.A. Fisher\n    :Donor: Michael Marshall (MARSHALL
%PLU@io.arc.nasa.gov)\n    :Date: July, 1988\n\nThis is a copy of UCI ML Wine recognition
datasets.\nhttps://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data\n\nThe
data is the results of a chemical analysis of wines grown in the same\nregion in Italy by
three different cultivators. There are thirteen different\nmeasurements taken for
different constituents found in the three types of\nwine.\n\nOriginal Owners: \n\nForina,
M. et al, PARVUS - \nAn Extendible Package for Data Exploration, Classification and
Correlation. \nInstitute of Pharmaceutical and Food Analysis and Technologies,\nVia
Brigata Salerno, 16147 Genoa, Italy.\n\nCitation:\n\nLichman, M. (2013). UCI Machine
Learning Repository\n[https://archive.ics.uci.edu/ml]. Irvine, CA: University of
California,\nSchool of Information and Computer Science. \n\n|details-start|\
n**References**\n|details-split|\n\n(1) S. Aeberhard, D. Coomans and O. de Vel, \
nComparison of Classifiers in High Dimensional Settings, \nTech. Rep. no. 92-02, (1992),
Dept. of Computer Science and Dept. of  \nMathematics and Statistics, James Cook
University of North Queensland. \n(Also submitted to Technometrics). \n\nThe data was used
with many others for comparing various \nclassifiers. The classes are separable, though
only RDA \nhas achieved 100% correct classification. \n(RDA : 100%, QDA 99.4%, LDA 98.9%,
1NN 96.1% (z-transformed data)) \n(All results using the leave-one-out technique) \n\n(2)
S. Aeberhard, D. Coomans and O. de Vel, \n"THE CLASSIFICATION PERFORMANCE OF RDA" \nTech.
Rep. no. 92-01, (1992), Dept. of Computer Science and Dept. of \nMathematics and
Statistics, James Cook University of North Queensland. \n(Also submitted to Journal of
Chemometrics).\n\n|details-end|',
 'feature_names': ['alcohol',
  'malic_acid',
  'ash',
  'alcalinity_of_ash',
  'magnesium',
  'total_phenols',
  'flavanoids',
  'nonflavanoid_phenols',
  'proanthocyanins',
  'color_intensity',
  'hue',
  'od280/od315_of_diluted_wines',
  'proline']}
```

In [8]:

```
df = pd.DataFrame(data = wine.data,columns = wine.feature_names)
```

In [9]:

```
df
```

Out[9]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	h
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68	1
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80	0
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32	1
...	...	...	...	...	...	...	...	...	...	...	..
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	0.52	1.06	7.70	0
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	0.43	1.41	7.30	0
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	0.43	1.35	10.20	0
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	0.53	1.46	9.30	0
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	0.56	1.35	9.20	0

178 rows × 13 columns

In [10]:

```
df['target'] = wine.target
```

In [11]:

```
df
```

Out[11]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	h
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	h
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68	1
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80	0
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32	1
...	...	...	...	...	...	...	...	...	...	...	..
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	0.52	1.06	7.70	0
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	0.43	1.41	7.30	0
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	0.43	1.35	10.20	0
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	0.53	1.46	9.30	0
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	0.56	1.35	9.20	0

178 rows × 14 columns

In [22]:

```
from sklearn.model_selection import train_test_split as Split
x_train,x_test,y_train,y_test =
Split(wine.data,wine.target,stratify=wine.target,random_state=123)

from sklearn.preprocessing import StandardScaler as SS
x_train_std = SS().fit_transform(x_train)
x_test_std = SS().fit_transform(x_test)

from sklearn.neural_network import MLPClassifier as mlp
mlp_model = mlp(hidden_layer_sizes = (10,), max_iter = 5000, learning_rate_init = 0.001)
from time import time
start = time()
mlp_model.fit(x_train_std,y_train)
end = time()
print(f'time taken = {(end-start)*1000}ms')

time taken = 209.75661277770996ms
```

In [28]:

```
from sklearn.metrics import confusion_matrix,classification_report
y_pred = mlp_model.predict(x_test_std)
```

```
confusion = confusion_matrix(y_test,y_pred)
```

In [29]:

```
confusion
```

Out[29]:

```
array([[15,  0,  0],
       [ 1, 17,  0],
       [ 0,  0, 12]], dtype=int64)
```

In [32]:

```
classiReport = classification_report(y_test,y_pred)
```

In [33]:

```
classiReport
```

Out[33]:

```
'              precision    recall  f1-score   support\n\n           0       0.94
1.00      0.97        15\n           1       1.00      0.94      0.97        18\n
2       1.00      1.00      1.00        12\n\n    accuracy                           0.98
45\n   macro avg       0.98      0.98      0.98        45\nweighted avg       0.98
0.98      0.98        45\n'
```

In [34]:

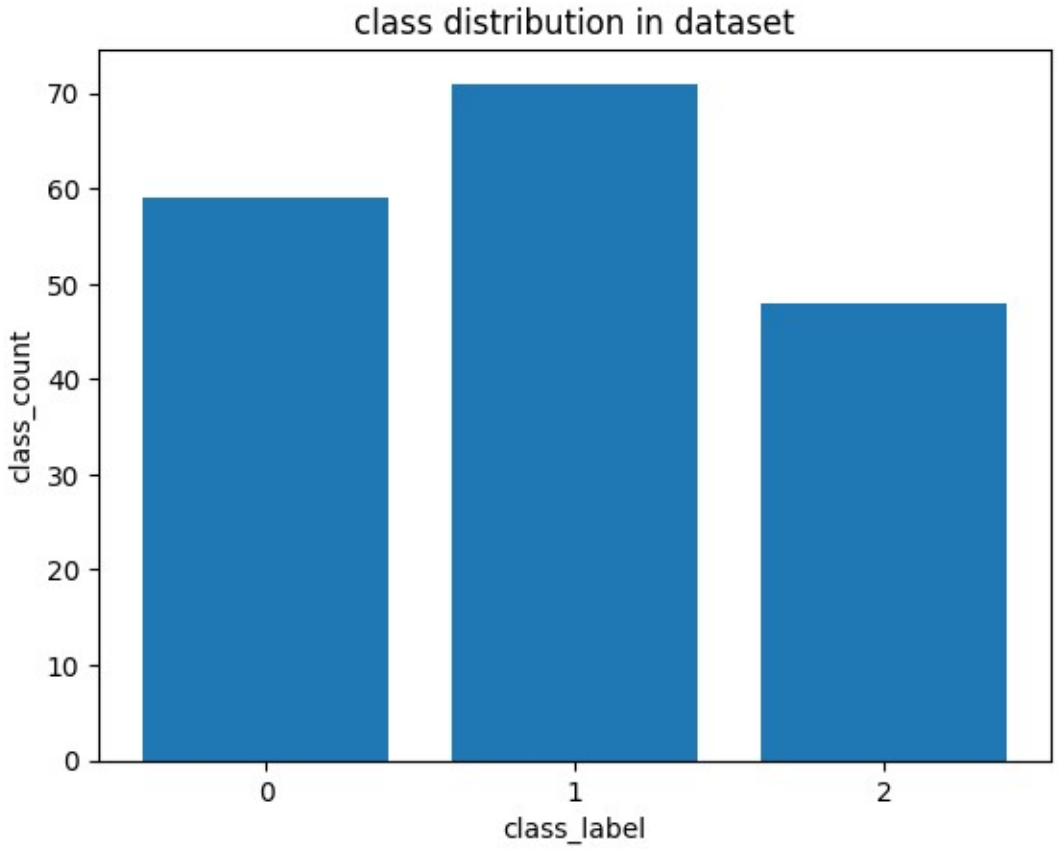
```
#plotting of barchart
```

In [41]:

```
import numpy as np
import matplotlib.pyplot as plt
class_labels = np.unique(wine.target)
class_count = np.bincount(wine.target)
plt.bar(class_labels,class_count,tick_label=class_labels)
plt.xlabel('class_label')
plt.ylabel('class_count')
plt.title('class distribution in dataset')
```

Out[41]:

```
Text(0.5, 1.0, 'class distribution in dataset')
```


class distribution in dataset
class_count
class_label
0
10
20
30
40
50
60
70
0
1
2

In [ ]: