```
# モジュールのインポート
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn import cluster, preprocessing, datasets
from sklearn.cluster import KMeans
```

```
# The data is the results of a chemical analysis of wines grown in the same
# region in Italy by three different cultivators. There are thirteen different
# measurements taken for different constituents found in the three types of
# wine.
wine = datasets.load_wine()
```

```
print(wine['DESCR'])
```

```
        Total Phenols:            0.98  3.88    2.29  0.63

        Flavanoids:               0.34  5.08    2.03  1.00
        Nonflavanoid Phenols:       0.13  0.66    0.36  0.12
        Proanthocyanins:          0.41  3.58    1.59  0.57
        Colour Intensity:           1.3  13.0     5.1   2.3
        Hue:                    0.48  1.71    0.96  0.23
        OD280/OD315 of diluted wines: 1.27  4.00    2.61  0.71
        Proline:                  278  1680     746   315
        ============================== ==== ===== ======= =====

        :Missing Attribute Values: None
        :Class Distribution: class_0 (59), class_1 (71), class_2 (48)
        :Creator: R.A. Fisher
        :Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
        :Date: July, 1988

    This is a copy of UCI ML Wine recognition datasets.
    https://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data

    The data is the results of a chemical analysis of wines grown in the same
    region in Italy by three different cultivators. There are thirteen different
    measurements taken for different constituents found in the three types of
    wine.

    Original Owners:

    Forina, M. et al, PARVUS -
    An Extendible Package for Data Exploration, Classification and Correlation.
    Institute of Pharmaceutical and Food Analysis and Technologies,
    Via Brigata Salerno, 16147 Genoa, Italy.

    Citation:

    Lichman, M. (2013). UCI Machine Learning Repository
    [https://archive.ics.uci.edu/ml]. Irvine, CA: University of California,
    School of Information and Computer Science.

    .. topic:: References

      (1) S. Aeberhard, D. Coomans and O. de Vel,
      Comparison of Classifiers in High Dimensional Settings
```

Comparison of Classifiers in High Dimensional Settings,
Tech. Rep. no. 92-02, (1992), Dept. of Computer Science and Dept. of
Mathematics and Statistics, James Cook University of North Queensland.
(Also submitted to Technometrics).

The data was used with many others for comparing various
classifiers. The classes are separable, though only RDA
has achieved 100% correct classification.
(RDA : 100%, QDA 99.4%, LDA 98.9%, 1NN 96.1% (z-transformed data))
(All results using the leave-one-out technique)

(2) S. Aeberhard, D. Coomans and O. de Vel,
"THE CLASSIFICATION PERFORMANCE OF RDA"
Tech. Rep. no. 92-01, (1992), Dept. of Computer Science and Dept. of
Mathematics and Statistics, James Cook University of North Queensland.
(Also submitted to Journal of Chemometrics).

```
# 説明変数
print(wine['feature_names'])
```

```
['alcohol', 'malic_acid', 'ash', 'alcalinity_of_ash', 'magnesium', 'total_phenols', 'flavanoids', 'nonfl
```

```
X = wine.data
```

```
X.shape
```

```
(178, 13)
```

```
print(X[:5])
```

```
[[1.423e+01 1.710e+00 2.430e+00 1.560e+01 1.270e+02 2.800e+00 3.060e+00
  2.800e-01 2.290e+00 5.640e+00 1.040e+00 3.920e+00 1.065e+03]
 [1.320e+01 1.780e+00 2.140e+00 1.120e+01 1.000e+02 2.650e+00 2.760e+00
  2.600e-01 1.280e+00 4.380e+00 1.050e+00 3.400e+00 1.050e+03]
 [1.316e+01 2.360e+00 2.670e+00 1.860e+01 1.010e+02 2.800e+00 3.240e+00
  3.000e-01 2.810e+00 5.680e+00 1.030e+00 3.170e+00 1.185e+03]
 [1.437e+01 1.950e+00 2.500e+00 1.680e+01 1.130e+02 3.850e+00 3.490e+00
  2.400e-01 2.180e+00 7.800e+00 8.600e-01 3.450e+00 1.480e+03]
 [1.324e+01 2.590e+00 2.870e+00 2.100e+01 1.180e+02 2.800e+00 2.690e+00
  3.900e-01 1.820e+00 4.320e+00 1.040e+00 2.930e+00 7.350e+02]]
```

```
# 目的変数
y = wine.target
```

```
y.shape
```

```
(178,)
```

```
wine.target_names
```

```
array(['class_0', 'class_1', 'class_2'], dtype='<U7')
```

```
# kmeans法
model = KMeans(n_clusters=3)
labels = model.fit_predict(X)
```

```
df = pd.DataFrame({'labels': labels})
```

```
def species_label(theta):
  if theta == 0:
    return wine.target_names[0]
  if theta == 1:
    return wine.target_names[1]
  if theta == 2:
    return wine.target_names[2]
```

```
df['species'] = [species_label(theta) for theta in wine.target]
```

```
# 正解データと比較
pd.crosstab(df['labels'], df['species'])
```

species labels	class_0	class_1	class_2
0	46	1	0
1	13	20	29
2	0	50	19