

Untitled

April 15, 2023

```
[1]: from sklearn.datasets import load_iris
```

```
[12]: import pandas as pd  
import numpy as np
```

```
[13]: data = load_iris()
```

```
[15]: df = pd.DataFrame(data = data.data , columns = data.feature_names)  
df.head()
```

```
[15]:      sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)  
0              5.1             3.5             1.4             0.2  
1              4.9             3.0             1.4             0.2  
2              4.7             3.2             1.3             0.2  
3              4.6             3.1             1.5             0.2  
4              5.0             3.6             1.4             0.2
```

```
[16]: df = pd.DataFrame(data = data.data , columns =  
    ↪ ['Slength', 'Swidth', 'Plength', 'Pwidth'])  
df.head()
```

```
[16]:      Slength  Swidth  Plength  Pwidth  
0         5.1      3.5       1.4      0.2  
1         4.9      3.0       1.4      0.2  
2         4.7      3.2       1.3      0.2  
3         4.6      3.1       1.5      0.2  
4         5.0      3.6       1.4      0.2
```

```
[17]: df['target'] = pd.DataFrame(data = data.target)  
df.head()
```

```
[17]:      Slength  Swidth  Plength  Pwidth  target  
0         5.1      3.5       1.4      0.2       0  
1         4.9      3.0       1.4      0.2       0  
2         4.7      3.2       1.3      0.2       0  
3         4.6      3.1       1.5      0.2       0  
4         5.0      3.6       1.4      0.2       0
```

```
[18]: df['target'].nunique()
```

```
[18]: 3
```

```
[19]: X = pd.DataFrame(data = data.data , columns =  
      ↪ ['Slength', 'Swidth', 'Plength', 'Pwidth'])  
y = pd.DataFrame(data = data.target)
```

```
[21]: from sklearn.neighbors import KNeighborsClassifier  
neigh = KNeighborsClassifier(n_neighbors=3)  
neigh.fit(X, y)
```

```
/home/h/anaconda3/lib/python3.9/site-  
packages/sklearn/neighbors/_classification.py:198: DataConversionWarning: A  
column-vector y was passed when a 1d array was expected. Please change the shape  
of y to (n_samples,), for example using ravel().  
    return self._fit(X, y)
```

```
[21]: KNeighborsClassifier(n_neighbors=3)
```

```
[24]: print(neigh.predict([[5.1,3.5,1.4,0.2]]))
```

```
[0]
```

```
/home/h/anaconda3/lib/python3.9/site-packages/sklearn/base.py:450: UserWarning:  
X does not have valid feature names, but KNeighborsClassifier was fitted with  
feature names  
    warnings.warn(  
/home/h/anaconda3/lib/python3.9/site-  
packages/sklearn/neighbors/_classification.py:228: FutureWarning: Unlike other  
reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode`  
typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will  
change: the default value of `keepdims` will become False, the `axis` over which  
the statistic is taken will be eliminated, and the value None will no longer be  
accepted. Set `keepdims` to True or False to avoid this warning.  
    mode, _ = stats.mode(_y[neigh_ind, k], axis=1)
```

```
[27]: print(neigh.predict([[6.7,3.0,5.2,2.3]]))
```

```
[2]
```

```
/home/h/anaconda3/lib/python3.9/site-packages/sklearn/base.py:450: UserWarning:  
X does not have valid feature names, but KNeighborsClassifier was fitted with  
feature names  
    warnings.warn(  
/home/h/anaconda3/lib/python3.9/site-  
packages/sklearn/neighbors/_classification.py:228: FutureWarning: Unlike other  
reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode`  
typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will
```

change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.

```
mode, _ = stats.mode(_y[neigh_ind, k], axis=1)
```

```
[28]: X
```

```
[28]:
```

	Slength	Swidth	Plength	Pwidth
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
..
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

[150 rows x 4 columns]

```
[29]: y
```

```
[29]:
```

	0
0	0
1	0
2	0
3	0
4	0
..	..
145	2
146	2
147	2
148	2
149	2

[150 rows x 1 columns]

```
[31]: A = neigh.kneighbors_graph(X)
A.toarray()
A
```

```
[31]: <150x150 sparse matrix of type '<class 'numpy.float64'>'
      with 450 stored elements in Compressed Sparse Row format>
```

```
[32]: print(A)
```

(0, 0)	1.0
(0, 17)	1.0
(0, 4)	1.0
(1, 1)	1.0
(1, 34)	1.0
(1, 45)	1.0
(2, 2)	1.0
(2, 47)	1.0
(2, 3)	1.0
(3, 3)	1.0
(3, 47)	1.0
(3, 29)	1.0
(4, 4)	1.0
(4, 37)	1.0
(4, 0)	1.0
(5, 5)	1.0
(5, 18)	1.0
(5, 10)	1.0
(6, 6)	1.0
(6, 47)	1.0
(6, 2)	1.0
(7, 7)	1.0
(7, 39)	1.0
(7, 49)	1.0
(8, 8)	1.0
:	:
(141, 139)	1.0
(142, 142)	1.0
(142, 101)	1.0
(142, 113)	1.0
(143, 143)	1.0
(143, 120)	1.0
(143, 124)	1.0
(144, 144)	1.0
(144, 140)	1.0
(144, 120)	1.0
(145, 145)	1.0
(145, 141)	1.0
(145, 147)	1.0
(146, 146)	1.0
(146, 123)	1.0
(146, 111)	1.0
(147, 147)	1.0
(147, 110)	1.0
(147, 111)	1.0
(148, 148)	1.0
(148, 136)	1.0
(148, 115)	1.0

```
(149, 149)    1.0
(149, 127)    1.0
(149, 138)    1.0
```

```
[33]: print(neigh.kneighbors([[6.7,3.0,5.2,2.3]]))
```

```
(array([[0.          , 0.24494897, 0.36055513]]), array([[145, 141, 147]]))
/home/h/anaconda3/lib/python3.9/site-packages/sklearn/base.py:450: UserWarning:
X does not have valid feature names, but KNeighborsClassifier was fitted with
feature names
  warnings.warn(
```

```
[ ]:
```