

Sci-kit+learn+tutorial

August 6, 2018

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

```
In [2]: iris = load_iris()
```

```
In [3]: print(iris.data)
```

```
[[ 5.1  3.5  1.4  0.2]
 [ 4.9  3.   1.4  0.2]
 [ 4.7  3.2  1.3  0.2]
 [ 4.6  3.1  1.5  0.2]
 [ 5.   3.6  1.4  0.2]
 [ 5.4  3.9  1.7  0.4]
 [ 4.6  3.4  1.4  0.3]
 [ 5.   3.4  1.5  0.2]
 [ 4.4  2.9  1.4  0.2]
 [ 4.9  3.1  1.5  0.1]
 [ 5.4  3.7  1.5  0.2]
 [ 4.8  3.4  1.6  0.2]
 [ 4.8  3.   1.4  0.1]
 [ 4.3  3.   1.1  0.1]
 [ 5.8  4.   1.2  0.2]
 [ 5.7  4.4  1.5  0.4]
 [ 5.4  3.9  1.3  0.4]
 [ 5.1  3.5  1.4  0.3]
 [ 5.7  3.8  1.7  0.3]
 [ 5.1  3.8  1.5  0.3]
 [ 5.4  3.4  1.7  0.2]
 [ 5.1  3.7  1.5  0.4]
 [ 4.6  3.6  1.   0.2]
 [ 5.1  3.3  1.7  0.5]
 [ 4.8  3.4  1.9  0.2]
 [ 5.   3.   1.6  0.2]
 [ 5.   3.4  1.6  0.4]
 [ 5.2  3.5  1.5  0.2]
 [ 5.2  3.4  1.4  0.2]
 [ 4.7  3.2  1.6  0.2]
 [ 4.8  3.1  1.6  0.2]
 [ 5.4  3.4  1.5  0.4]
```

```

[ 5.2  4.1  1.5  0.1]
[ 5.5  4.2  1.4  0.2]
[ 4.9  3.1  1.5  0.1]
[ 5.   3.2  1.2  0.2]
[ 5.5  3.5  1.3  0.2]
[ 4.9  3.1  1.5  0.1]
[ 4.4  3.   1.3  0.2]
[ 5.1  3.4  1.5  0.2]
[ 5.   3.5  1.3  0.3]
[ 4.5  2.3  1.3  0.3]
[ 4.4  3.2  1.3  0.2]
[ 5.   3.5  1.6  0.6]
[ 5.1  3.8  1.9  0.4]
[ 4.8  3.   1.4  0.3]
[ 5.1  3.8  1.6  0.2]
[ 4.6  3.2  1.4  0.2]
[ 5.3  3.7  1.5  0.2]
[ 5.   3.3  1.4  0.2]
[ 7.   3.2  4.7  1.4]
[ 6.4  3.2  4.5  1.5]
[ 6.9  3.1  4.9  1.5]
[ 5.5  2.3  4.   1.3]
[ 6.5  2.8  4.6  1.5]
[ 5.7  2.8  4.5  1.3]
[ 6.3  3.3  4.7  1.6]
[ 4.9  2.4  3.3  1. ]
[ 6.6  2.9  4.6  1.3]
[ 5.2  2.7  3.9  1.4]
[ 5.   2.   3.5  1. ]
[ 5.9  3.   4.2  1.5]
[ 6.   2.2  4.   1. ]
[ 6.1  2.9  4.7  1.4]
[ 5.6  2.9  3.6  1.3]
[ 6.7  3.1  4.4  1.4]
[ 5.6  3.   4.5  1.5]
[ 5.8  2.7  4.1  1. ]
[ 6.2  2.2  4.5  1.5]
[ 5.6  2.5  3.9  1.1]
[ 5.9  3.2  4.8  1.8]
[ 6.1  2.8  4.   1.3]
[ 6.3  2.5  4.9  1.5]
[ 6.1  2.8  4.7  1.2]
[ 6.4  2.9  4.3  1.3]
[ 6.6  3.   4.4  1.4]
[ 6.8  2.8  4.8  1.4]
[ 6.7  3.   5.   1.7]
[ 6.   2.9  4.5  1.5]
[ 5.7  2.6  3.5  1. ]

```

```

[ 5.5  2.4  3.8  1.1]
[ 5.5  2.4  3.7  1. ]
[ 5.8  2.7  3.9  1.2]
[ 6.   2.7  5.1  1.6]
[ 5.4  3.   4.5  1.5]
[ 6.   3.4  4.5  1.6]
[ 6.7  3.1  4.7  1.5]
[ 6.3  2.3  4.4  1.3]
[ 5.6  3.   4.1  1.3]
[ 5.5  2.5  4.   1.3]
[ 5.5  2.6  4.4  1.2]
[ 6.1  3.   4.6  1.4]
[ 5.8  2.6  4.   1.2]
[ 5.   2.3  3.3  1. ]
[ 5.6  2.7  4.2  1.3]
[ 5.7  3.   4.2  1.2]
[ 5.7  2.9  4.2  1.3]
[ 6.2  2.9  4.3  1.3]
[ 5.1  2.5  3.   1.1]
[ 5.7  2.8  4.1  1.3]
[ 6.3  3.3  6.   2.5]
[ 5.8  2.7  5.1  1.9]
[ 7.1  3.   5.9  2.1]
[ 6.3  2.9  5.6  1.8]
[ 6.5  3.   5.8  2.2]
[ 7.6  3.   6.6  2.1]
[ 4.9  2.5  4.5  1.7]
[ 7.3  2.9  6.3  1.8]
[ 6.7  2.5  5.8  1.8]
[ 7.2  3.6  6.1  2.5]
[ 6.5  3.2  5.1  2. ]
[ 6.4  2.7  5.3  1.9]
[ 6.8  3.   5.5  2.1]
[ 5.7  2.5  5.   2. ]
[ 5.8  2.8  5.1  2.4]
[ 6.4  3.2  5.3  2.3]
[ 6.5  3.   5.5  1.8]
[ 7.7  3.8  6.7  2.2]
[ 7.7  2.6  6.9  2.3]
[ 6.   2.2  5.   1.5]
[ 6.9  3.2  5.7  2.3]
[ 5.6  2.8  4.9  2. ]
[ 7.7  2.8  6.7  2. ]
[ 6.3  2.7  4.9  1.8]
[ 6.7  3.3  5.7  2.1]
[ 7.2  3.2  6.   1.8]
[ 6.2  2.8  4.8  1.8]
[ 6.1  3.   4.9  1.8]

```



```
In [8]: print (type(iris.data))
```

```
<class 'numpy.ndarray'>
```

```
In [9]: print(type(iris.target))
```

```
<class 'numpy.ndarray'>
```

```
In [10]: print(iris.data.shape)
```

```
(150, 4)
```

```
In [11]: prrin(iris.data.target)
```

```
-----  
NameError
```

```
Traceback (most recent call last)
```

```
<ipython-input-11-d1558a8449fe> in <module>()  
----> 1 prrin(iris.data.target)
```

```
NameError: name 'prrin' is not defined
```

```
In [12]: print(iris.data.target)
```

```
-----  
AttributeError
```

```
Traceback (most recent call last)
```

```
<ipython-input-12-ee5d529aa8c8> in <module>()  
----> 1 print(iris.data.target)
```

```
AttributeError: 'numpy.ndarray' object has no attribute 'target'
```

```
In [13]: print(iris.target.shape)
```

```
(150,)
```

```
In [14]: X = iris.data
```

```

In [15]: y = iris.target

In [16]: from sklearn.neighbors import KNeighborsClassifier

In [17]: knn = KNeighborsClassifier()

In [18]: print(knn)

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                    metric_params=None, n_jobs=1, n_neighbors=5, p=2,
                    weights='uniform')

In [19]: knn = KNeighborsClassifier(n_neighbors = 1)

In [20]: print(knn)

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                    metric_params=None, n_jobs=1, n_neighbors=1, p=2,
                    weights='uniform')

In [21]: knn.fit(X,y)

Out[21]: KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                             metric_params=None, n_jobs=1, n_neighbors=1, p=2,
                             weights='uniform')

In [26]: import numpy as np

In [28]: test = np.array([1,2,3,4])

In [30]: knn.predict(test.reshape(1,-1))

Out[30]: array([2])

In [34]: print(iris.target_names)

['setosa' 'versicolor' 'virginica']

In [35]: from sklearn.cross_validation import train_test_split

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\cross_validation.py:41: DeprecationWarning:
  "This module will be removed in 0.20.", DeprecationWarning)

In [36]: from sklearn.model_selection import train_test_split

In [37]: X_train, X_test, y_train, y_test = train_test_split(X,y)

```

```

In [38]: knn = KNeighborsClassifier()

In [39]: print(knn)

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                    metric_params=None, n_jobs=1, n_neighbors=5, p=2,
                    weights='uniform')

In [41]: knn.fit(X_train, y_train)

Out[41]: KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                    metric_params=None, n_jobs=1, n_neighbors=5, p=2,
                    weights='uniform')

In [42]: knn.predict(X_test)

Out[42]: array([2, 2, 2, 2, 1, 2, 0, 2, 1, 0, 2, 2, 1, 1, 2, 2, 0, 2, 1, 0, 0, 0, 2,
                1, 2, 1, 0, 1, 1, 1, 0, 2, 0, 2, 2, 0, 0, 1])

In [43]: y_pred = knn.predict(X_test)

In [46]: from sklearn.metrics import accuracy_score

In [47]: print(accuracy_score(y_test, y_pred))

0.973684210526

In [56]: import numpy as np
         f = open('C:\\Users\\lab10-pc-11\\Downloads\\pima-indians-diabetes.data', 'r')
         dataset = np.loadtxt(f, delimiter=",")
         print(dataset.shape)
         # separate the data from the target attributes
         X = dataset[:,0:7]
         y = dataset[:,8]

(768, 9)

In [57]: print(type(y))

<class 'numpy.ndarray'>

In [58]: print(X.shape)

(768, 7)

In [59]: print(y.shape)

(768,)

```