파이썬을 이용한 데이터과학과 머신러닝 입문

Iris Dataset

August, 2019

Ein Intelligence

1

Machine Learning Using Python

Scikit Learn Library

- Scikit Learn Examples

  https://scikit-learn.org/stable/auto_examples/index.html

- Iris Dataset

  https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_dataset.html

2

KNN Classification

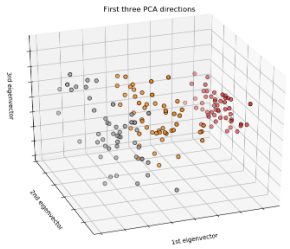
- **The Iris Dataset** **https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_dataset.html#sphx-glr-auto-examples-datasets-plot-iris-dataset-py**

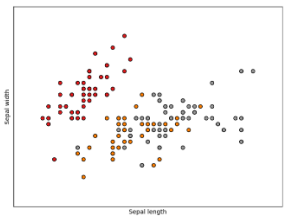
The Iris Dataset

This data sets consists of 3 different types of irises' (Setosa, Versicolour, and Virginica) petal and sepal length, stored in a 150x4 numpy.ndarray

The rows being the samples and the columns being: Sepal Length, Sepal Width, Petal Length and Petal Width.

The below plot uses the first two features. See here for more information on this dataset.






Setosa


Versicolour


Virginica

* Petal : 꽃잎
Sepal : 꽃받침


```
[ ]  import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     from sklearn.model_selection import train_test_split
     from sklearn import datasets
     %matplotlib inline
```

```
[ ]  iris = datasets.load_iris()
```

```
[ ]  print (iris.keys)
     print (format(iris.keys()))
```

```
<built-in method keys of Bunch object at 0x7f4daf0296d0>
dict_keys(['data', 'target', 'target_names', 'DESCR', 'feature_names', 'filename'])
```

```
[ ]  print (iris["DESCR"])
```

```
[ ]  print (iris['target_names'])
     print (iris['feature_names'])
     iris.feature_names
```

```
['setosa' 'versicolor' 'virginica']
['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']
['sepal length (cm)',
 'sepal width (cm)',
 'petal length (cm)',
 'petal width (cm)']
```

```
[ ]  print (format(type(iris['data'])))
     print (format(iris['data'].shape))
     print (format(iris['data'][0:5]))
```

```
<class 'numpy.ndarray'>
(150, 4)
[[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]]
```

```
[ ]  print (format(type(iris['target'])))
     print (format(iris['target'].shape))
     print (format(iris['target'][:]))
```

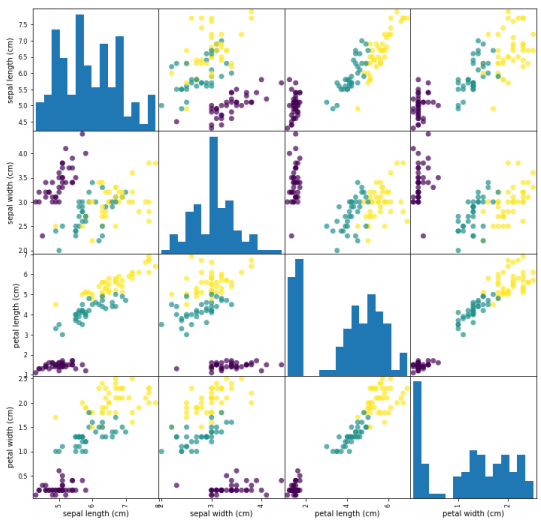
```
<class 'numpy.ndarray'>
(150,)
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2]
```

```
[ ]  X = iris['data']
     y = iris['target']
     X_train, X_test, y_train, y_test = train_test_split(X, y, random_state = 0)
     print (format(X_train.shape))
     print (format(X_test.shape))
     print (format(y_train.shape))
     print (format(y_test.shape))
```

```
(112, 4)
(38, 4)
(112,)
(38,)
```

5

```
[ ]  df_iris = pd.DataFrame(X_train, columns=iris.feature_names)
     pd.plotting.scatter_matrix(df_iris, c=y_train, figsize=(12,12), marker='o', hist_kwds={'bins': 15}, s=49, alpha=0.7)
```



6

```
[ ]  from sklearn.neighbors import KNeighborsClassifier
     knn = KNeighborsClassifier(n_neighbors=5)

[ ]  knn.fit(X_train, y_train)

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

[ ]  X_new = np.array([[5, 2.9, 1, 0.2]])
     print("X_new.shape: {}".format(X_new.shape))

     X_new.shape: (1, 4)

[ ]  prediction = knn.predict(X_new)
     print (format(prediction))
     print (format(iris.target_names[prediction]))

     [0]
     ['setosa']

[ ]  y_pred = knn.predict(X_test)
     print (format(y_pred))
     print ("accuracy {}".format(np.mean(y_pred == y_test)))
     knn.score(X_test, y_test)

     [2 1 0 2 0 2 0 1 1 1 2 1 1 1 1 0 1 1 0 0 2 1 0 0 2 0 0 1 1 0 2 1 0 2 2 1 0
      2]
     accuracy 0.9736842105263158
     0.9736842105263158
```

** Ref) https://tensorflow.blog/파이썬-머신러닝/1-7-첫-번째-애플리케이션-붓꽃의-품종-분류/*

7

Let's Practice

8