

```
import numpy as np
import pandas as pd
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

```
data = pd.read_csv('iris.csv')
data.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
data.isna().values.any()
```

```
False
```

```
encoder = preprocessing.LabelEncoder()
```

```
data['species'] = encoder.fit_transform(data['species'])
data.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	8	14	4	1	0
1	6	9	4	1	0
2	4	11	3	1	0
3	3	10	5	1	0
4	7	15	4	1	0

```
data.describe()
```

	sepal_length	sepal_width	petal_length	petal_width	species
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	15.386667	9.540000	18.200000	8.986667	1.000000
std	8.175743	4.300195	11.648666	6.405730	0.819232
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	8.000000	7.000000	6.000000	2.000000	0.000000
50%	15.000000	9.000000	19.500000	9.000000	1.000000
75%	21.000000	12.000000	27.000000	14.000000	2.000000
max	34.000000	22.000000	42.000000	21.000000	2.000000

```
x = data.drop('species', axis = 1)
y = data.species
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 0)
```

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix, accuracy_score
```

```
knn = KNeighborsClassifier(n_neighbors = 3)
knn.fit(x_train,y_train)
```

```
KNeighborsClassifier(n_neighbors=3)
```

```
y_pred = knn.predict(x_test)
y_pred
```

```
array([2, 1, 0, 2, 0, 2, 0, 1, 1, 1, 2, 1, 1, 1, 2, 0, 1, 1, 0, 0, 2, 1,
       0, 0, 2, 0, 0, 1, 1, 0])
```

```
y_pred == y_test
```

```
114    True
62     True
33     True
107    True
7      True
100    True
40     True
86     True
76     True
71     True
134    True
51     True
73     True
54     True
63    False
37     True
78     True
90     True
45     True
16     True
121    True
66     True
24     True
8      True
126    True
22     True
44     True
97     True
93     True
26     True
Name: species, dtype: bool
```

```
print("Accuracy : %.2f"%(accuracy_score(y_test,y_pred)))
```

```
Accuracy : 0.97
```

```
confusion_matrix(y_true = y_test, y_pred = y_pred, labels = (2,1,0))
```

```
array([[ 6,  0,  0],
       [ 1, 12,  0],
       [ 0,  0, 11]])
```