

```
[1]: import numpy as np
import pandas as pd
df=pd.read_csv(r"C:\Users\nivet\OneDrive\Documents\Iris (1).csv")
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column             Non-Null Count  Dtype  
---  --
 0   sepal.length       150 non-null   float64
 1   sepal.width        150 non-null   float64
 2   petal.length       150 non-null   float64
 3   petal.width        150 non-null   float64
 4   variety            150 non-null   object  
dtypes: float64(4), object(1)
memory usage: 6.8+ KB
```

```
[2]: df.variety.value_counts()
```

```
[2]: variety
Setosa      50
Versicolor  50
Virginica   50
Name: count, dtype: int64
```

```
[3]: df.head()
```

```
[3]:   sepal.length  sepal.width  petal.length  petal.width  variety
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
```

```
[4]: features=df.iloc[:, :-1].values
label=df.iloc[:, 4].values
```

```
[5]: from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
```

```
[6]: x_train, x_test, y_train, y_test = train_test_split(features, label, test_size=0.2, random_state=0)

model_KNN = KNeighborsClassifier(n_neighbors=5)
model_KNN.fit(x_train, y_train)
```

```
[6]: + KNeighborsClassifier
```

Parameters

```
[8]: print(model_KNN.score(x_train,y_train))
print(model_KNN.score(x_test,y_test))
```

```
0.95
0.9666666666666667
```

```
[9]: from sklearn.metrics import confusion_matrix
confusion_matrix(label,model_KNN.predict(features))
```

```
[9]: array([[50,  0,  0],
        [ 0, 45,  5],
        [ 0,  2, 48]])
```



```
[10]: from sklearn.metrics import classification_report
print(classification_report(label,model_KNN.predict(features)))
```

	precision	recall	f1-score	support
Setosa	1.00	1.00	1.00	50
Versicolor	0.96	0.98	0.93	50
Virginica	0.91	0.96	0.93	50
accuracy			0.95	150
macro avg	0.95	0.95	0.95	150
weighted avg	0.95	0.95	0.95	150

```
[ ]:
```