In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
data = pd.read_csv("Iris.csv")
data
```

Out[2]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [3]:
```
data = data.drop("Id",axis=1)
data
```

Out[3]:

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

In [4]:
```
# 3 label classes --> Iris-setosa = 0, Iris-versicolor = 1, Iris-virginica = 2
```

In [5]:
```
Y = data["Species"].values
data = data.drop("Species",axis=1)
```

In [6]:
```
from sklearn import preprocessing
encoder = preprocessing.LabelEncoder()
encoder.fit(Y)
Y_numbers = encoder.transform(Y)
Y_numbers
```

Out[6]:
```
array([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, 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, 2, 2])
```

In [7]:
```
Y_num_serisi = pd.Series(Y_numbers, name="Class")
dataframe = pd.concat([data,Y_num_serisi],axis=1)
dataframe
```

Out[7]:

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Class
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
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	2
146	6.3	2.5	5.0	1.9	2
147	6.5	3.0	5.2	2.0	2
148	6.2	3.4	5.4	2.3	2
149	5.9	3.0	5.1	1.8	2

150 rows × 5 columns

In [8]:
```
Y = dataframe["Class"].values
X_1234 = dataframe.drop("Class",axis=1)

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X_1234,Y,test_size=0.33)
```

In [9]:
```
from sklearn.neighbors import KNeighborsClassifier
#n_neighbors = int(sqrt(len(x_train))/2)
knn = KNeighborsClassifier(n_neighbors=5,metric="euclidean")
knn.fit(x_train,y_train)
y_pred = knn.predict(x_test)
```

In [10]:
```
y_pred_seri = pd.Series(y_pred)
y_test_seri = pd.Series(y_test)
a = dict(Gercek_Sinif=y_test_seri, Tahmin_Sinif=y_pred_seri)
```

```
dataFrame = pd.DataFrame(a)
dataFrame
```

Out[10]:

	Gercek_Sinif	Tahmin_Sinif
0	1	1
1	2	2
2	1	1
3	1	1
4	1	1
5	2	2
6	0	0
7	0	0
8	0	0
9	2	2
10	0	0
11	2	2
12	2	2
13	2	2
14	1	1
15	2	2
16	2	2
17	0	0
18	1	1
19	2	2
20	2	2
21	0	0
22	0	0
23	1	1
24	0	0
25	2	2
26	1	1
27	2	2
28	0	0
29	1	1
30	1	1
31	2	2
32	1	1
33	2	2
34	0	0

	Gercek_Sinif	Tahmin_Sinif
35	1	2
36	2	2
37	0	0
38	1	1
39	0	0
40	2	2
41	1	1
42	2	2
43	1	1
44	2	2
45	2	2
46	0	0
47	1	1
48	0	0
49	1	1

In [11]:
```
from sklearn.metrics import confusion_matrix
result = confusion_matrix(y_test,y_pred)
result
```

Out[11]:
```
array([[14,  0,  0],
       [ 0, 16,  1],
       [ 0,  0, 19]], dtype=int64)
```

In [13]:
```
accuracy = (14+16+19)/(14+16+19+1)
accuracy = accuracy*100
accuracy  # % doğruluk
```

Out[13]:
```
98.0
```

In [ ]:
```

```