AIM:Program to implement k-NN classification using any standard dataset available in the public domain and find the accuracy of the algorithm.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot asplt
from sklearn.neghbors import KNeighborsClassifier
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
```

In [8]:

```
import pandas as pd
iris=pd.read_csv('Iris.csv')
iris.tail()
```

Out[8]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
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

In [11]:

```
iris['Species'].value_counts()
```

Out[11]:

```
Iris-setosa       50
Iris-versicolor   50
Iris-virginica    50
Name: Species, dtype: int64
```

In [13]:

```
iris.columns
```

Out[13]:

```
Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
       •   'Species'],
    dtype='object')
```

In [14]:

iris.values

Out[14]:

```
array([[1, 5.1, 3.5, 1.4, 0.2, 'Iris-setosa'],
       [2, 4.9, 3.0, 1.4, 0.2, 'Iris-setosa'],
       [3, 4.7, 3.2, 1.3, 0.2, 'Iris-setosa'],
       [4, 4.6, 3.1, 1.5, 0.2, 'Iris-setosa'],
       [5, 5.0, 3.6, 1.4, 0.2, 'Iris-setosa'],
       [6, 5.4, 3.9, 1.7, 0.4, 'Iris-setosa'],
       [7, 4.6, 3.4, 1.4, 0.3, 'Iris-setosa'],
       [8, 5.0, 3.4, 1.5, 0.2, 'Iris-setosa'],
       [9, 4.4, 2.9, 1.4, 0.2, 'Iris-setosa'],
       [10, 4.9, 3.1, 1.5, 0.1, 'Iris-setosa'],
       [11, 5.4, 3.7, 1.5, 0.2, 'Iris-setosa'],
       [12, 4.8, 3.4, 1.6, 0.2, 'Iris-setosa'],
       [13, 4.8, 3.0, 1.4, 0.1, 'Iris-setosa'],
       [14, 4.3, 3.0, 1.1, 0.1, 'Iris-setosa'],
       [15, 5.8, 4.0, 1.2, 0.2, 'Iris-setosa'],
       [16, 5.7, 4.4, 1.5, 0.4, 'Iris-setosa'],
       [17, 5.4, 3.9, 1.3, 0.4, 'Iris-setosa'],
       [18, 5.1, 3.5, 1.4, 0.3, 'Iris-setosa'],
       [19, 5.7, 3.8, 1.7, 0.3, 'Iris-setosa'],
       [20, 5.1, 3.8, 1.5, 0.3, 'Iris-setosa'],
       [21, 5.4, 3.4, 1.7, 0.2, 'Iris-setosa'],
       [22, 5.1, 3.7, 1.5, 0.4, 'Iris-setosa'],
       [23, 4.6, 3.6, 1.0, 0.2, 'Iris-setosa'],
       [24, 5.1, 3.3, 1.7, 0.5, 'Iris-setosa'],
       [25, 4.8, 3.4, 1.9, 0.2, 'Iris-setosa'],
       [26, 5.0, 3.0, 1.6, 0.2, 'Iris-setosa'],
       [27, 5.0, 3.4, 1.6, 0.4, 'Iris-setosa'],
       [28, 5.2, 3.5, 1.5, 0.2, 'Iris-setosa'],
       [29, 5.2, 3.4, 1.4, 0.2, 'Iris-setosa'],
       [30, 4.7, 3.2, 1.6, 0.2, 'Iris-setosa'],
       [31, 4.8, 3.1, 1.6, 0.2, 'Iris-setosa'],
       [32, 5.4, 3.4, 1.5, 0.4, 'Iris-setosa'],
       [33, 5.2, 4.1, 1.5, 0.1, 'Iris-setosa'],
       [34, 5.5, 4.2, 1.4, 0.2, 'Iris-setosa'],
       [35, 4.9, 3.1, 1.5, 0.1, 'Iris-setosa'],
       [36, 5.0, 3.2, 1.2, 0.2, 'Iris-setosa'],
       [37, 5.5, 3.5, 1.3, 0.2, 'Iris-setosa'],
       [38, 4.9, 3.1, 1.5, 0.1, 'Iris-setosa'],
       [39, 4.4, 3.0, 1.3, 0.2, 'Iris-setosa'],
       [40, 5.1, 3.4, 1.5, 0.2, 'Iris-setosa'],
       [41, 5.0, 3.5, 1.3, 0.3, 'Iris-setosa'],
       [42, 4.5, 2.3, 1.3, 0.3, 'Iris-setosa'],
       [43, 4.4, 3.2, 1.3, 0.2, 'Iris-setosa'],
       [44, 5.0, 3.5, 1.6, 0.6, 'Iris-setosa'],
       [45, 5.1, 3.8, 1.9, 0.4, 'Iris-setosa'],
       [46, 4.8, 3.0, 1.4, 0.3, 'Iris-setosa'],
       [47, 5.1, 3.8, 1.6, 0.2, 'Iris-setosa'],
       [48, 4.6, 3.2, 1.4, 0.2, 'Iris-setosa'],
```

```
[49, 5.3, 3.7, 1.5, 0.2, 'Iris-setosa'],
[50, 5.0, 3.3, 1.4, 0.2, 'Iris-setosa'],
[51, 7.0, 3.2, 4.7, 1.4, 'Iris-versicolor'],
[52, 6.4, 3.2, 4.5, 1.5, 'Iris-versicolor'],
[53, 6.9, 3.1, 4.9, 1.5, 'Iris-versicolor'],
[54, 5.5, 2.3, 4.0, 1.3, 'Iris-versicolor'],
[55, 6.5, 2.8, 4.6, 1.5, 'Iris-versicolor'],
[56, 5.7, 2.8, 4.5, 1.3, 'Iris-versicolor'],
[57, 6.3, 3.3, 4.7, 1.6, 'Iris-versicolor'],
[58, 4.9, 2.4, 3.3, 1.0, 'Iris-versicolor'],
[59, 6.6, 2.9, 4.6, 1.3, 'Iris-versicolor'],
[60, 5.2, 2.7, 3.9, 1.4, 'Iris-versicolor'],
[61, 5.0, 2.0, 3.5, 1.0, 'Iris-versicolor'],
[62, 5.9, 3.0, 4.2, 1.5, 'Iris-versicolor'],
[63, 6.0, 2.2, 4.0, 1.0, 'Iris-versicolor'],
[64, 6.1, 2.9, 4.7, 1.4, 'Iris-versicolor'],
[65, 5.6, 2.9, 3.6, 1.3, 'Iris-versicolor'],
[66, 6.7, 3.1, 4.4, 1.4, 'Iris-versicolor'],
[67, 5.6, 3.0, 4.5, 1.5, 'Iris-versicolor'],
[68, 5.8, 2.7, 4.1, 1.0, 'Iris-versicolor'],
[69, 6.2, 2.2, 4.5, 1.5, 'Iris-versicolor'],
[70, 5.6, 2.5, 3.9, 1.1, 'Iris-versicolor'],
[71, 5.9, 3.2, 4.8, 1.8, 'Iris-versicolor'],
[72, 6.1, 2.8, 4.0, 1.3, 'Iris-versicolor'],
[73, 6.3, 2.5, 4.9, 1.5, 'Iris-versicolor'],
[74, 6.1, 2.8, 4.7, 1.2, 'Iris-versicolor'],
[75, 6.4, 2.9, 4.3, 1.3, 'Iris-versicolor'],
[76, 6.6, 3.0, 4.4, 1.4, 'Iris-versicolor'],
[77, 6.8, 2.8, 4.8, 1.4, 'Iris-versicolor'],
[78, 6.7, 3.0, 5.0, 1.7, 'Iris-versicolor'],
[79, 6.0, 2.9, 4.5, 1.5, 'Iris-versicolor'],
[80, 5.7, 2.6, 3.5, 1.0, 'Iris-versicolor'],
[81, 5.5, 2.4, 3.8, 1.1, 'Iris-versicolor'],
[82, 5.5, 2.4, 3.7, 1.0, 'Iris-versicolor'],
[83, 5.8, 2.7, 3.9, 1.2, 'Iris-versicolor'],
[84, 6.0, 2.7, 5.1, 1.6, 'Iris-versicolor'],
[85, 5.4, 3.0, 4.5, 1.5, 'Iris-versicolor'],
[86, 6.0, 3.4, 4.5, 1.6, 'Iris-versicolor'],
[87, 6.7, 3.1, 4.7, 1.5, 'Iris-versicolor'],
[88, 6.3, 2.3, 4.4, 1.3, 'Iris-versicolor'],
[89, 5.6, 3.0, 4.1, 1.3, 'Iris-versicolor'],
[90, 5.5, 2.5, 4.0, 1.3, 'Iris-versicolor'],
[91, 5.5, 2.6, 4.4, 1.2, 'Iris-versicolor'],
[92, 6.1, 3.0, 4.6, 1.4, 'Iris-versicolor'],
[93, 5.8, 2.6, 4.0, 1.2, 'Iris-versicolor'],
[94, 5.0, 2.3, 3.3, 1.0, 'Iris-versicolor'],
[95, 5.6, 2.7, 4.2, 1.3, 'Iris-versicolor'],
[96, 5.7, 3.0, 4.2, 1.2, 'Iris-versicolor'],
[97, 5.7, 2.9, 4.2, 1.3, 'Iris-versicolor'],
[98, 6.2, 2.9, 4.3, 1.3, 'Iris-versicolor'],
[99, 5.1, 2.5, 3.0, 1.1, 'Iris-versicolor'],
```

```
[100, 5.7, 2.8, 4.1, 1.3, 'Iris-versicolor'],
[101, 6.3, 3.3, 6.0, 2.5, 'Iris-virginica'],
[102, 5.8, 2.7, 5.1, 1.9, 'Iris-virginica'],
[103, 7.1, 3.0, 5.9, 2.1, 'Iris-virginica'],
[104, 6.3, 2.9, 5.6, 1.8, 'Iris-virginica'],
[105, 6.5, 3.0, 5.8, 2.2, 'Iris-virginica'],
[106, 7.6, 3.0, 6.6, 2.1, 'Iris-virginica'],
[107, 4.9, 2.5, 4.5, 1.7, 'Iris-virginica'],
[108, 7.3, 2.9, 6.3, 1.8, 'Iris-virginica'],
[109, 6.7, 2.5, 5.8, 1.8, 'Iris-virginica'],
[110, 7.2, 3.6, 6.1, 2.5, 'Iris-virginica'],
[111, 6.5, 3.2, 5.1, 2.0, 'Iris-virginica'],
[112, 6.4, 2.7, 5.3, 1.9, 'Iris-virginica'],
[113, 6.8, 3.0, 5.5, 2.1, 'Iris-virginica'],
[114, 5.7, 2.5, 5.0, 2.0, 'Iris-virginica'],
[115, 5.8, 2.8, 5.1, 2.4, 'Iris-virginica'],
[116, 6.4, 3.2, 5.3, 2.3, 'Iris-virginica'],
[117, 6.5, 3.0, 5.5, 1.8, 'Iris-virginica'],
[118, 7.7, 3.8, 6.7, 2.2, 'Iris-virginica'],
[119, 7.7, 2.6, 6.9, 2.3, 'Iris-virginica'],
[120, 6.0, 2.2, 5.0, 1.5, 'Iris-virginica'],
[121, 6.9, 3.2, 5.7, 2.3, 'Iris-virginica'],
[122, 5.6, 2.8, 4.9, 2.0, 'Iris-virginica'],
[123, 7.7, 2.8, 6.7, 2.0, 'Iris-virginica'],
[124, 6.3, 2.7, 4.9, 1.8, 'Iris-virginica'],
[125, 6.7, 3.3, 5.7, 2.1, 'Iris-virginica'],
[126, 7.2, 3.2, 6.0, 1.8, 'Iris-virginica'],
[127, 6.2, 2.8, 4.8, 1.8, 'Iris-virginica'],
[128, 6.1, 3.0, 4.9, 1.8, 'Iris-virginica'],
[129, 6.4, 2.8, 5.6, 2.1, 'Iris-virginica'],
[130, 7.2, 3.0, 5.8, 1.6, 'Iris-virginica'],
[131, 7.4, 2.8, 6.1, 1.9, 'Iris-virginica'],
[132, 7.9, 3.8, 6.4, 2.0, 'Iris-virginica'],
[133, 6.4, 2.8, 5.6, 2.2, 'Iris-virginica'],
[134, 6.3, 2.8, 5.1, 1.5, 'Iris-virginica'],
[135, 6.1, 2.6, 5.6, 1.4, 'Iris-virginica'],
[136, 7.7, 3.0, 6.1, 2.3, 'Iris-virginica'],
[137, 6.3, 3.4, 5.6, 2.4, 'Iris-virginica'],
[138, 6.4, 3.1, 5.5, 1.8, 'Iris-virginica'],
[139, 6.0, 3.0, 4.8, 1.8, 'Iris-virginica'],
[140, 6.9, 3.1, 5.4, 2.1, 'Iris-virginica'],
[141, 6.7, 3.1, 5.6, 2.4, 'Iris-virginica'],
[142, 6.9, 3.1, 5.1, 2.3, 'Iris-virginica'],
[143, 5.8, 2.7, 5.1, 1.9, 'Iris-virginica'],
[144, 6.8, 3.2, 5.9, 2.3, 'Iris-virginica'],
[145, 6.7, 3.3, 5.7, 2.5, 'Iris-virginica'],
[146, 6.7, 3.0, 5.2, 2.3, 'Iris-virginica'],
[147, 6.3, 2.5, 5.0, 1.9, 'Iris-virginica'],
[148, 6.5, 3.0, 5.2, 2.0, 'Iris-virginica'],
[149, 6.2, 3.4, 5.4, 2.3, 'Iris-virginica'],
[150, 5.9, 3.0, 5.1, 1.8, 'Iris-virginica']], dtype=object)
```

In [15]:

```
iris.describe(include='all')
```

Out[15]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
count	150.000000	150.000000	150.000000	150.000000	150.000000	150
unique	NaN	NaN	NaN	NaN	NaN	3
top	NaN	NaN	NaN	NaN	NaN	Iris-setosa
freq	NaN	NaN	NaN	NaN	NaN	50
mean	75.500000	5.843333	3.054000	3.758667	1.198667	NaN
std	43.445368	0.828066	0.433594	1.764420	0.763161	NaN
min	1.000000	4.300000	2.000000	1.000000	0.100000	NaN
25%	38.250000	5.100000	2.800000	1.600000	0.300000	NaN
50%	75.500000	5.800000	3.000000	4.350000	1.300000	NaN
75%	112.750000	6.400000	3.300000	5.100000	1.800000	NaN
max	150.000000	7.900000	4.400000	6.900000	2.500000	NaN

In [17]:

```
x=iris.iloc[:,:4]
x.head()
```

Out[17]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm
0	1	5.1	3.5	1.4

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm
1	2	4.9	3.0	1.4
2	3	4.7	3.2	1.3
3	4	4.6	3.1	1.5
4	5	5.0	3.6	1.4

In [18]:

```
y=iris.iloc[:,-1]
y.head()
```

Out[18]:

```
0    Iris-setosa
1    Iris-setosa
2    Iris-setosa
3    Iris-setosa
4    Iris-setosa
Name: Species, dtype: object
```

In [22]:

```
from sklearn import preprocessing
x=preprocessing.StandardScaler().fit_transform(x)
x[0:4]
```

Out[22]:

```
array([[-1.72054204, -0.90068117,  1.03205722, -1.3412724 ],
       [-1.69744751, -1.14301691, -0.1249576 , -1.3412724 ],
       [-1.67435299, -1.38535265,  0.33784833, -1.39813811],
       [-1.65125846, -1.50652052,  0.10644536, -1.2844067 ]])
```

In [26]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=1)
y_test.shape
```

Out[26]:

```
(45,)
```

In [39]:

```
from sklearn.neighbors import KNeighborsClassifier
knnmodel=KNeighborsClassifier(n_neighbors=3)
knnmodel.fit(x_train,y_train)
```

Out[39]:

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

In [44]:

```
from sklearn.neighbors import KNeighborsClassifier
KNeighborsClassifier(algorithm='auto',leaf_size=30,metric='minkowski',metric_params=None,n_jobs
=None,n_neighbors=3,p=2,weights='uniform')
```

Out[44]:

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

In [48]:

```
y_predict1=knnmodel.predict(x_test)
```

In [49]:

```
from sklearn.metrics import accuracy_score
```

In [50]:

```
acc=accuracy_score(y_test,y_predict1)
acc
```

Out[50]:

```
1.0
```

In [52]:

```
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test.values,y_predict1)
cm
```

Out[52]:

```
array([[14,  0,  0],
       [ 0, 18,  0],
       [ 0,  0, 13]], dtype=int64)
```

In [53]:

```
cm1=pd.DataFrame(data=cm,index=['setosa','versicolor','verginica'],columns=['setosa','versicolor','ver
ginica'])
cm1
```

Out[53]:

	setosa	versicolor	verginica
setosa	14	0	0
versicolor	0	18	0
verginica	0	0	13

In [56]:

```
prediction_output=pd.DataFrame(data=[y_test.values,y_predict1],index=['y_test','y_predict1'])
```

In [57]:

```
prediction_output.transpose()
```

Out[57]:

	y_test	y_predict1
0	Iris-setosa	Iris-setosa
1	Iris-versicolor	Iris-versicolor
2	Iris-versicolor	Iris-versicolor
3	Iris-setosa	Iris-setosa
4	Iris-virginica	Iris-virginica
5	Iris-versicolor	Iris-versicolor
6	Iris-virginica	Iris-virginica
7	Iris-setosa	Iris-setosa
8	Iris-setosa	Iris-setosa
9	Iris-virginica	Iris-virginica
10	Iris-versicolor	Iris-versicolor
11	Iris-setosa	Iris-setosa
12	Iris-virginica	Iris-virginica
13	Iris-versicolor	Iris-versicolor
14	Iris-versicolor	Iris-versicolor
15	Iris-setosa	Iris-setosa

	y_test	y_predict1
16	Iris-versicolor	Iris-versicolor
17	Iris-versicolor	Iris-versicolor
18	Iris-setosa	Iris-setosa
19	Iris-setosa	Iris-setosa
20	Iris-versicolor	Iris-versicolor
21	Iris-versicolor	Iris-versicolor
22	Iris-versicolor	Iris-versicolor
23	Iris-setosa	Iris-setosa
24	Iris-virginica	Iris-virginica
25	Iris-versicolor	Iris-versicolor
26	Iris-setosa	Iris-setosa
27	Iris-setosa	Iris-setosa
28	Iris-versicolor	Iris-versicolor
29	Iris-virginica	Iris-virginica
30	Iris-versicolor	Iris-versicolor
31	Iris-virginica	Iris-virginica
32	Iris-versicolor	Iris-versicolor

	y_test	y_predict1
33	Iris-virginica	Iris-virginica
34	Iris-virginica	Iris-virginica
35	Iris-setosa	Iris-setosa
36	Iris-versicolor	Iris-versicolor
37	Iris-setosa	Iris-setosa
38	Iris-versicolor	Iris-versicolor
39	Iris-virginica	Iris-virginica
40	Iris-virginica	Iris-virginica
41	Iris-setosa	Iris-setosa
42	Iris-virginica	Iris-virginica
43	Iris-virginica	Iris-virginica
44	Iris-versicolor	Iris-versicolor

In [58]:

```
prediction_output.iloc[0,:].value_counts()
```

Out[58]:

```
Iris-versicolor    18
Iris-setosa        14
Iris-virginica     13
Name: y_test, dtype: int64
```

In [ ]: