EXPERIMENT 7

Aim:

To build a classification model using the K-Nearest Neighbors (KNN) algorithm to classify iris flower species based on their sepal and petal measurements.

Algorithm:

1. Import necessary libraries such as NumPy, Pandas, and Scikit-learn.
2. Load the Iris dataset using Pandas.
3. Display dataset information using info() and check class distribution using value_counts().
4. Split the dataset into features and labels.
5. Divide the data into training and testing sets using train_test_split().
6. Initialize the KNN classifier with n_neighbors=5.
7. Train the classifier using the training data.
8. Evaluate model accuracy on training and testing data using score().
9. Generate the confusion matrix to evaluate predictions.
10. Display the classification report showing precision, recall, and f1-score.

Code:

```
import numpy as np
import pandas as pd
df = pd.read_csv('Iris (1) - Iris (1).csv')
df.info()
```

Output:

```
<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: 6.0+ KB

---

df.variety.value_counts()

Output:

variety

Setosa 50

Versicolor 50

Virginica 50

Name: count, dtype: int64

---

df.head()

Output:

	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

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

from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

xtrain, xtest, ytrain, ytest = train_test_split(features, label, test_size=0.2, random_state=42)
```

```
model_KNN = KNeighborsClassifier(n_neighbors=5)
model_KNN.fit(xtrain, ytrain)
```

(No output for this cell)

```
print(model_KNN.score(xtrain, ytrain))
print(model_KNN.score(xtest, ytest))
```

Output:

```
0.9666666666666667
1.0
```

```
from sklearn.metrics import confusion_matrix
confusion_matrix(label, model_KNN.predict(features))
```

Output:

```
array([[50,  0,  0],
       [ 0, 47,  3],
       [ 0,  1, 49]])
```

```
from sklearn.metrics import classification_report
print(classification_report(label, model_KNN.predict(features)))
```

Output:

```
              precision    recall  f1-score   support

      Setosa       1.00      1.00      1.00        50
  Versicolor       0.98      0.94      0.96        50
   Virginica       0.94      0.98      0.96        50

    accuracy                           0.97       150
   macro avg       0.97      0.97      0.97       150
weighted avg       0.97      0.97      0.97       150
```

Result:

Thus, the Python program to classify iris flower species using the K-Nearest Neighbors (KNN) algorithm was executed successfully. The model achieved an overall accuracy of **97%**, indicating effective performance in classifying the flower species based on petal and sepal dimensions.