

K-Nearest Neighbours (KNN) classification model using the Iris dataset to predict the species of iris flowers

October 25, 2024

```
[1]: # Importing necessary libraries
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.datasets import load_iris

# Load the Iris dataset
iris = load_iris()

df = pd.DataFrame(data=iris.data, columns=iris.feature_names)
df['target'] = iris.target

# Features and labels
X = iris.data # Features: Sepal Length, Sepal Width, Petal Length, Petal Width
y = iris.target # Labels: Species of Iris (Setosa, Versicolor, Virginica)

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
                                                    random_state=42)

# Initialize the KNN classifier with K=5
knn = KNeighborsClassifier(n_neighbors=5)

# Train the KNN model
knn.fit(X_train, y_train)

# Make predictions on the test data
y_pred = knn.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy of the KNN model: {accuracy * 100:.2f}%")
```

```
# Display classification report
print("\nClassification Report:")
print(classification_report(y_test, y_pred, target_names=iris.target_names))
```

Accuracy of the KNN model: 100.00%

Classification Report:

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| setosa | 1.00 | 1.00 | 1.00 | 19 |
| versicolor | 1.00 | 1.00 | 1.00 | 13 |
| virginica | 1.00 | 1.00 | 1.00 | 13 |
| accuracy | | | 1.00 | 45 |
| macro avg | 1.00 | 1.00 | 1.00 | 45 |
| weighted avg | 1.00 | 1.00 | 1.00 | 45 |

[]: