assignment 1

[ ] ↳ *5 cells hidden*

assignment 2

[ ] ↳ *12 cells hidden*

assignment 01

```
import pandas as pd
from sklearn.datasets import load_iris

# Load the Iris dataset
iris = load_iris()
df = pd.DataFrame(iris.data, columns=iris.feature_names)
df['target'] = iris.target

# Check for missing values
print(df.isnull().sum())
```

```
sepal length (cm)    0
sepal width (cm)     0
petal length (cm)    0
petal width (cm)     0
target               0
dtype: int64
```

```
import numpy as np

# Convert a column to a NumPy array
sepal_length_np = df['sepal length (cm)'].values

# Calculate basic statistics
print("Mean sepal length:", np.mean(sepal_length_np))
print("Median sepal length:", np.median(sepal_length_np))
```

```
Mean sepal length: 5.843333333333334
Median sepal length: 5.8
```

```
from sklearn.model_selection import train_test_split

X = df.drop('target', axis=1)
y = df['target']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score

# Create a KNN classifier
knn = KNeighborsClassifier()

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

# Make predictions
y_pred = knn.predict(X_test)

# Evaluate accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

```
Accuracy: 1.0
```

Report Dataset Analysis

The Iris dataset was used for this analysis. It contains information about sepal length, sepal width, petal length, petal width, and target species.

Data Preprocessing

No data cleaning was required as there were no missing values.

Model Building and Evaluation

A K-Nearest Neighbors (KNN) classifier was used for classification. The model achieved an accuracy of [insert accuracy value] on the testing set.

Conclusion

The KNN model performed well on the Iris dataset, demonstrating the effectiveness of this simple algorithm for classification tasks.