

Q1:

```
import pandas as pd
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import LabelEncoder

data = {
    'Gender': ['Male', 'Male', 'Male', 'Male', 'Female', 'Female', 'Female', 'Female'],
    'Height': [6.00, 5.92, 5.58, 5.92, 5.00, 5.50, 5.42, 5.75],
    'Weight': [180, 190, 170, 165, 100, 150, 130, 150],
    'Foot_Size': [12, 11, 12, 10, 6, 8, 7, 9],
    'Target': [1, 0, 0, 1, 1, 0, 1, 0] # Assuming Target as the Label column for prediction
}

df = pd.DataFrame(data)

le_gender = LabelEncoder()
df['Gender'] = le_gender.fit_transform(df['Gender'])

X = df[['Gender', 'Height', 'Weight', 'Foot_Size']]
y = df['Target']

dt = DecisionTreeClassifier()

dt.fit(X, y)

new_entry = {
    'Gender': 'Male',
    'Height': 175,
    'Weight': 68,
    'Foot_Size': 9
}

new_entry_encoded = [le_gender.transform([new_entry['Gender']])[0], new_entry['Height'], new_entry['Weight'], new_entry['Foot_Size']]

prediction = dt.predict([new_entry_encoded])

predicted_label = 'Play' if prediction[0] == 1 else 'Not Play'

print(f"Prediction for new entry: The person will {'play' if predicted_label == 'Play' else 'not play'}.")
```

Prediction for new entry: The person will play.

Q2:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier

data = {
    'Gender': ['Male', 'Female', 'Female', 'Male'],
    'Height': [180, 165, 170, 175],
    'Weight': [75, 60, 65, 70],
    'Foot_Size': [10, 8, 9, 9],
    'Result': [1, 0, 0, 1] # Example target variable
}

df = pd.DataFrame(data)

df['Gender'] = df['Gender'].map({'Male': 0, 'Female': 1})

X = df[['Gender', 'Height', 'Weight', 'Foot_Size']]
y = df['Result']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

dt = DecisionTreeClassifier()
dt.fit(X_train, y_train)

new_entry = {
    'Gender': 'Male',
    'Height': 178,
    'Weight': 72,
    'Foot_Size': 9,
    'Result': 1
}

new_entry['Gender'] = 0 if new_entry['Gender'] == 'Male' else 1

new_entry_df = pd.DataFrame([new_entry])

df = pd.concat([df, new_entry_df], ignore_index=True)

X = df[['Gender', 'Height', 'Weight', 'Foot_Size']]
y = df['Result']

dt.fit(X, y)

input_transformed = [
    new_entry['Gender'],
    new_entry['Height'],
    new_entry['Weight'],
    new_entry['Foot_Size']
]

prediction = dt.predict([input_transformed])

print(f"Prediction for the new entry (Gender: {new_entry['Gender']}, Height: {new_entry['Height']}, Weight: {new_entry['Weight']}, Foot_Size: {new_entry['Foot_Size']})")
print("\nUpdated Dataset:")
print(df)
```

Prediction for the new entry (Gender: 0, Height: 178, Weight: 72, Foot_Size: 9): 1

Updated Dataset:

	Gender	Height	Weight	Foot_Size	Result
0	0	180	75	10	1
1	1	165	60	8	0
2	1	170	65	9	0
3	0	175	70	9	1
4	0	178	72	9	1

Q3:

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier

data = np.array([
    [0, 180, 75, 10, 1],
    [1, 165, 60, 8, 0],
    [1, 170, 65, 9, 0],
    [0, 175, 70, 9, 1]
])

X = data[:, :-1]
y = data[:, -1]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

dt = DecisionTreeClassifier()
dt.fit(X_train, y_train)

new_entry = np.array([[0, 178, 72, 9, 1]])

data = np.vstack([data, new_entry])

X = data[:, :-1]
y = data[:, -1]

dt.fit(X, y)

input_transformed = np.array([[0, 178, 72, 9]])

prediction = dt.predict(input_transformed)

print(f"Prediction for the new entry (Gender: 0, Height: 178, Weight: 72, Foot_Size: 9): {'1' if prediction[0] == 1 else '0'}")

print("\nUpdated Dataset:")
print(data)
```

Prediction for the new entry (Gender: 0, Height: 178, Weight: 72, Foot_Size: 9): 1

Updated Dataset:

```
[[ 0 180 75 10 1]
 [ 1 165 60  8 0]
 [ 1 170 65  9 0]
 [ 0 175 70  9 1]
 [ 0 178 72  9 1]]
```

HomeTask:



AhmedHehe-24 Add files via upload



AI LAB#4 COMPLETED.pdf

Add files via upload