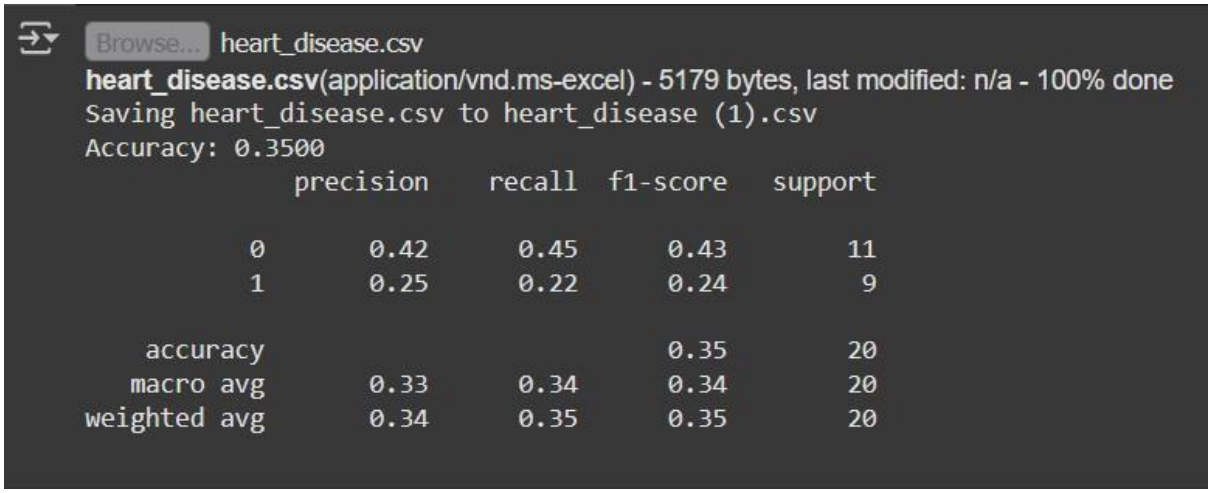
Heart Disease Prediction

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from google.colab import files
uploaded = files.upload()  # Upload heart_disease.csv here
data = pd.read_csv('heart_disease.csv')
data = data.dropna(subset=['target'])
for col in data.select_dtypes(include=['number']):
    data[col] = data[col].fillna(data[col].mean())
categorical_cols = data.select_dtypes(exclude=['number']).columns
data = pd.get_dummies(data, columns=categorical_cols, drop_first=True)
X = data.drop('target', axis=1)
y = data['target']
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy:.4f}")
print(classification_report(y_test, y_pred))
```

Output


Browse...
heart_disease.csv
heart_disease.csv(application/vnd.ms-excel) - 5179 bytes, last modified: n/a - 100% done
Saving heart_disease.csv to heart_disease (1).csv
Accuracy: 0.3500
precision recall f1-score support
0 0.42 0.45 0.43 11
1 0.25 0.22 0.24 9
accuracy 0.35 20
macro avg 0.33 0.34 0.34 20
weighted avg 0.34 0.35 0.35 20