Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение высшего образования «Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Лабораторная работа №2

Выполнил: студент IV курса ИВТ,

гр. ИП-713

Михеев Н.А.

Проверила: ассистент кафедры ПМиК

Морозова К.И.

Новосибирск, 2020 г.

Цель

В приложенном файле (heart_data.csv) располагаются реальные данные по сердечной заболеваемости, собранные различными медицинскими учреждениями. Каждый человек представлен 13-ю характеристиками и полем goal, которое показывает наличие болезни сердца, поле принимает значение 0 или 1 (0 – нет болезни, 1 - есть). Символ ‘?’ в каком-либо поле означает, что для конкретного человека отсутствуют данные в этом поле (либо не производились замеры, либо не записывались в базу). Требуется имеющиеся данные разбить на обучающую и тестовую выборки в процентном соотношении 70 к 30. После чего по обучающей выборке необходимо построить решающее дерево. Для построения дерева можно пользоваться любыми существующими средствами. Кроме того, для построения дерева необходимо будет решить задачу выделения информативных решающих правил относительно имеющихся числовых признаков. Разрешается использовать уже реализованные решающие деревья из известных библиотек (например, scikit-learn для Python), либо реализовывать алгоритм построения дерева самостоятельно (все необходимые алгоритмы представлены в теории по ссылке). В качестве результата работы необходимо сделать не менее 10 случайных разбиений исходных данных на обучающую и тестовую выборки, для каждой построить дерево и протестировать, после чего построить таблицу, в которой указать процент правильно классифицированных данных. Полученную таблицу необходимо включить в отчёт по лабораторной работе.

Результат работы

Для выполнения данного задания была разработана программа с использованием библиотеки sklearn. Были считаны данные, очищены все строки в которых есть неопределенные значения. Далее в программе был запущен цикл, в котором идет разбиение на обучающую и тестовую выборки по 0.7 и 0.3 от общего количества данных соответственно. Инициализируется классификатор решающего дерева с глубиной 6 и максимальным количеством «нод» для листьев – 12 – эти значения были подобраны в ходе тестовых проверок. Идет обучения нашего классификатора по обучающей выборке и дальше следует проверка нашей тестовой выборки. С каждым шагом цикла идет вывод точности работы нашего классификатора. За все прогоны программы точность не опускалась ниже 70%, а зачастую точность приближена к 80%.

```
Accuracy: 75.0%
Accuracy: 77.2%
Accuracy: 83.7%
Accuracy: 75.0%
Accuracy: 83.7%
Accuracy: 79.3%
Accuracy: 80.4%
Accuracy: 79.3%
Accuracy: 85.9%
Accuracy: 77.2%
```

Рис.1 – Демонстрация точности работы программы.

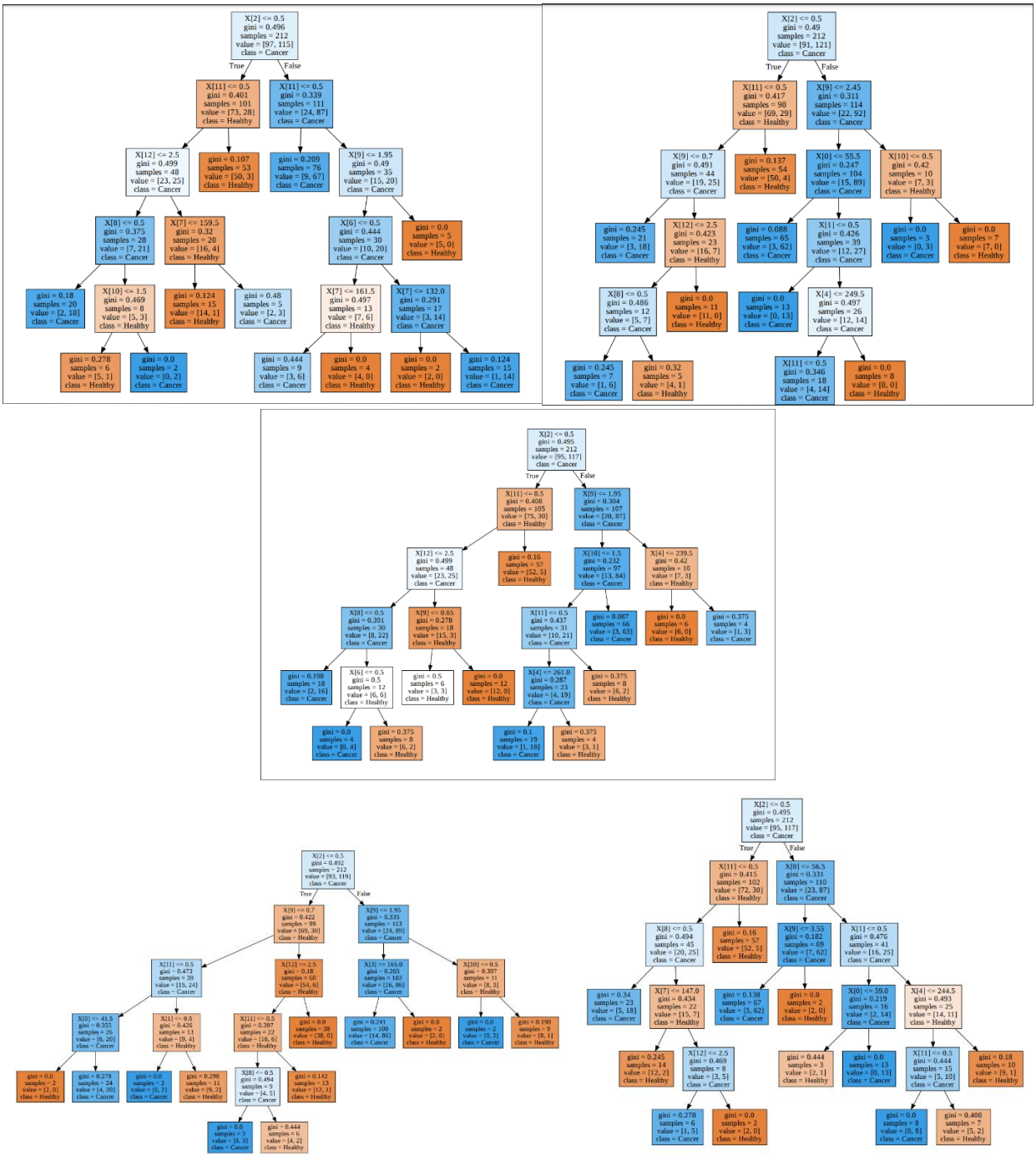

Рис.2 – примеры построенных деревьев.

Листинг программы

```
from random import randint

import pandas as pd
import graphviz
from sklearn import tree
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split


if __name__ == '__main__':
    nan_value = float("NaN")
    data = pd.read_csv('ac_heart_data.csv', header=None)
    data.replace("?", nan_value, inplace=True)
    data.dropna(inplace=True)
    x_data = data.loc[1:, 0:12]
    y_data = data.loc[1:, 13]

    for i in range(10):
        X_train, X_test, Y_train, Y_test = train_test_split(x_data, y_data,
test_size=0.3, random_state=randint(0, 10000))
        clf = tree.DecisionTreeClassifier(random_state=0, max_depth=6,
max_leaf_nodes=12)
        clf.fit(X_train, Y_train)
        prediction = clf.predict(X_test)
        print(f"Accuracy: {accuracy_score(Y_test, prediction) * 100:.3}%")

        dot_data = tree.export_graphviz(clf, out_file=None,
                                        class_names=['Healthy', 'Cancer'],
                                        filled=True)
        graph = graphviz.Source(dot_data)
        graph.render('test-output/round-table.gv', view=True)
```