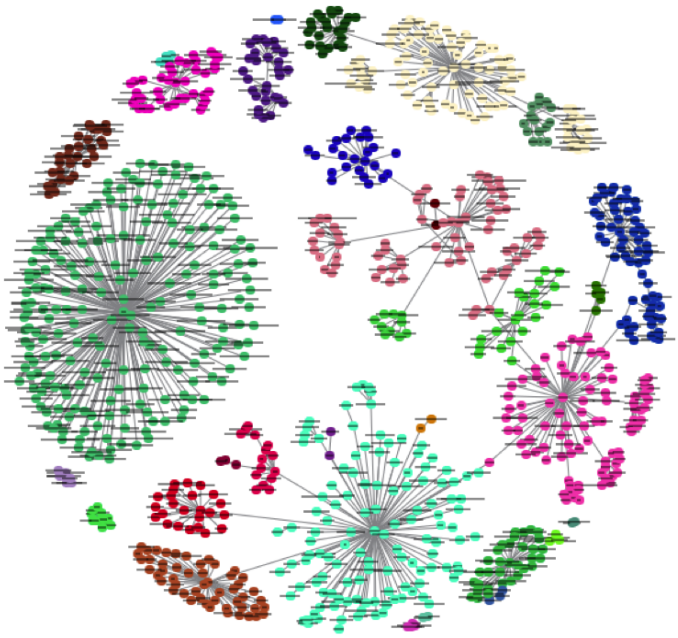
TRƯỜNG THPT CHUYÊN TRẦN ĐẠI NGHĨA



DỰ ĐOÁN SỰ XUẤT HIỆN BỆNH TIM MẠCH Ở NGƯỜI DỰA VÀO CÁC CHỈ SỐ CƠ THỂ VỚI THUẬT TOÁN KNN

Giáo viên hướng dẫn: Thạc sĩ Hồ Ngọc Lâm

Nhóm học sinh thực hiện:

1. Huỳnh Chí Tài
2. Nguyễn Quang Anh
3. Văn Công Minh Triết
4. Phạm Hoàng Phúc
5. Phan Đình Minh Quân

Học sinh lớp 12CTin - Khóa 2021-2024

Mục lục

1. Giới thiệu

1.1. Đặt vấn đề

Bệnh tim mạch là một trong những nguyên nhân hàng đầu gây tử vong trên toàn cầu. Việc phát hiện và dự đoán sự xuất hiện của bệnh từ sớm là rất quan trọng để có thể cung cấp điều trị và chăm sóc sức khỏe phù hợp. Đó là lí do ứng dụng này được phát triển nhằm dự đoán sớm một phần nguy cơ mắc bệnh tim mạch dựa trên các chỉ số cơ thể của người dùng.

1.2. Các thông số sức khỏe

- Tuổi tác: trong dataset này sẽ bao gồm dữ liệu của những người có độ tuổi từ 29 đến 77.
- Giới tính: Nam (1) hay Nữ (0).
- Các kiểu đau ngực: trong dữ liệu này sẽ có 4 loại:
    - Đau ngực do tim: 1
    - Đau cơ xương: 2
    - Đau ngực đường tiêu hóa: 3
    - Đau ngực phổi: 4
- Huyết áp nghỉ: là huyết áp khi đang nghỉ ngơi.
- Nồng độ cholesterol trong huyết thanh máu (mg/dl).
- FBS: chỉ số y tế dùng để đo lượng đường trong máu sau khi nhịn đói trong khoảng thời gian nhất định (thường là 8 - 12 giờ), dữ liệu này sẽ chỉ quan tâm đến việc chỉ số FBS có lớn hơn 120 mg/dl hay không (nếu lớn hơn 120 mg/dl là bị tiểu đường, 1 = True; 0 = False).
- Resting ECG: là phương pháp đánh giá hoạt động điện của tim mạch trong trạng thái nghỉ ngơi, dữ liệu này sẽ chia ra 3 loại (0, 1, 2) (trong đó 0 = bình thường; 1 = có bất thường sóng ST-T; 2 = cho thấy phì đại thất trái có thể xảy ra hoặc xác định theo tiêu chí của Estes).
- Nhịp tim tối đa: là nhịp tim tối đa mà một người có sau một khoảng thời gian đo nhất định.
- Đau thắt ngực do vận động: là cảm giác đau, không thoải mái của bệnh nhân khi hoạt động thể chất, dữ liệu sẽ quan tâm bệnh nhân có hoặc không đang gặp tình trạng này (1 = Có; 0 = Không).

- So sánh giữa biểu hiện điện tín hiệu trong ECG giữa lúc nghỉ ngơi và sau vận động để thấy được độ suy giảm điện thế.
- Độ dốc của đoạn ST tập luyện đỉnh cao (1: Dốc lên, 2: Phẳng, 3: Dốc xuống)
- Số mạch chính 0 - 3
- Thal: bệnh về máu có tên đầy đủ là Thalassemia (bệnh thiếu máu Địa Trung Hải): 3 = bình thường; 6 = có bệnh; 7 = có bệnh và có thể chữa được.

1.3. Cách sử dụng

Để sử dụng ứng dụng, người dùng chỉ cần cung cấp các thông tin về chiều cao, cân nặng, độ tuổi và một số thông số khác liên quan đến sức khỏe của họ. Sau đó, ứng dụng sẽ tính toán điểm tương đồng giữa dữ liệu được cung cấp với dữ liệu của những người khác trong cơ sở dữ liệu và dự đoán nguy cơ mắc bệnh tim mạch.

2. Sản phẩm

Xây dựng một mô hình KNN để dự đoán nguy cơ mắc bệnh tim dựa trên các đặc trưng y tế như tuổi, giới tính, huyết áp, hàm lượng cholesterol, và các chỉ số khác.
Chọn giá trị k phù hợp và chuẩn hóa dữ liệu để tối ưu hóa hiệu suất của mô hình.
Sử dụng Euclidean Distance và xét k giá trị để dự đoán.
Đánh giá mô hình bằng cách sử dụng các phương pháp đánh giá mô hình như đánh giá đúng/sai, ma trận nhầm lẫn (confusion matrix), precision, recall, và F1-score.

3. Quá trình phát triển

3.1. Xây dựng mô hình

3.1.1. Tìm dữ liệu

https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset/

3.1.2. Xử lý dữ liệu

Xử lý các giá trị cột cp, thal, và slope

```
a = pd.get_dummies(df['cp'], prefix = "cp")
b = pd.get_dummies(df['thal'], prefix = "thal")
c = pd.get_dummies(df['slope'], prefix = "slope")
frames = [df, a, b, c]
```

```
df = pd.concat(frames, axis = 1)
df = df.drop(columns = ['cp', 'thal', 'slope'])
y = df.target.values
x_data = df.drop(['target'], axis = 1)
```

Sử dụng MinMaxScaler để scale lại dữ liệu

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
x = scaler.fit_transform(x_data)
x = pd.DataFrame(x,
columns=['age','sex','trestbps','chol','fbs','restecg','thalach','exang','oldpeak','ca','cp_0','cp_1','cp_2','cp_3','tha
l_0','thal_1','thal_2','thal_3','slope_0','slope_1','slope_2'])
```

3.1.3. Tách bộ dữ liệu

```
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.2,random_state=0)
```

3.1.4. Tìm giá trị K tốt nhất và huấn luyện model

```
from sklearn.neighbors import KNeighborsClassifier
scoreList = []
for i in range(2,20):
    knn2 = KNeighborsClassifier(n_neighbors = i)  # n_neighbors means k
    knn2.fit(x_train, y_train)
    scoreList.append(knn2.score(x_test, y_test))

plt.plot(range(2,20), scoreList)
plt.xticks(np.arange(1,20,1))
plt.xlabel("K value")
plt.ylabel("Score")
plt.show()

acc = max(scoreList)*100
print("Maximum KNN Score is {:.2f}%".format(acc))
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 2)  # n_neighbors means k
knn.fit(x_train, y_train)
```

3.2. Tạo giao diện

Sử dụng web framework Flask và html để xây dựng trang web bao gồm giao diện hướng dẫn sử dụng, giao diện để nhập thông tin người dùng, và giao diện thông báo kết quả.

4. Kết luận

Ứng dụng dự đoán sự xuất hiện bệnh tim mạch dựa vào các chỉ số cơ thể với thuật toán KNN là một công cụ hữu ích để phần nào đánh giá nguy cơ mắc bệnh và giúp người dùng bảo vệ sức khỏe bản thân. Bằng cách phân loại người dùng vào các nhóm có thể mắc bệnh tim mạch hoặc

không, ứng dụng cung cấp thông tin cần thiết để họ có thể thực hiện các biện pháp phòng ngừa và thay đổi lối sống, từ đó giảm nguy cơ mắc bệnh tim mạch.

5. Tài liệu tham khảo

[1] https://viblo.asia/p/knn-k-nearest-neighbors-1-djeZ14ejKWz, truy cập lần cuối 11h03 ngày 15/11/2023.
[2] https://codelearn.io/sharing/thuat-toan-k-nearest-neighbors-knn, truy cập lần cuối 11h10 ngày 18/11/2023.
[3] https://machinelearningcoban.com/2017/01/08/knn/, truy cập lần cuối 11h18 ngày 8/12/2023.
[4] https://www.ibm.com/topics/knn, truy cập lần cuối 11h27 ngày 9/12/2023.
[5] https://www.geeksforgeeks.org/k-nearest-neighbours/, truy cập lần cuối 22h32 ngày 10/12/2023.