ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN CUỐI KỲ
MÔN: TOÁN KHOA HỌC MÁY TÍNH – CS115

A. THÔNG TIN CHUNG

A1. Tên đồ án

-Tên tiếng Việt: Phát hiện bệnh tim.

-Tên tiếng Anh: Heart Disease Detection.

A2. Thành viên thực hiện

TT	Họ tên	MSSV	Khoa/ Bộ Môn
1	Dương Đình Thắng	19522195	Khoa học Máy tính
2	Dương Nguyễn Thuận	19522312	Khoa học Máy tính
3	Phan Minh Nhật	19521956	Khoa học Máy tính
4	Hoàng Ngọc Bá Thi	19522255	Khoa học Máy tính

B. MÔ TẢ ĐỒ ÁN

B1. Giới thiệu về đề tài

Vấn đề về tim mạch luôn là một trong những vấn đề sức khỏe được quan tâm nhất, theo nhận định của WHO (World Health Organization), hiện tại bệnh tim mạch là nguyên nhân hàng đầu gây tử vong trên toàn cầu, chiếm tới 31% tổng số ca tử vong. Tại Việt Nam, bệnh tim mạch chịu trách nhiệm cho 31% tổng số ca tử vong trong năm 2016 tương đương với hơn 170.000 ca tử vong. Trong những năm gần đây, bệnh tim có xu hướng tăng về số lượng người mắc và phức tạp hơn về thành phần người mắc, nghiêm trọng hơn đó là xu hướng trẻ hóa của những căn bệnh về tim.

Để có thể hiểu biết hơn về vấn đề này, cũng như muốn vận dụng những kiến thức cơ bản về Khoa học Máy tính để rèn luyện kỹ năng và học hỏi thêm, nhóm chúng

em đã thống nhất chọn đề tài "**Phát hiện bệnh tim**" (Heart disease detection). Hy vọng nhận được sự đánh giá, hướng dẫn chuyên sâu hơn từ thầy.

B2. Mục tiêu, kế hoạch, nội dung

B2.1. Mục tiêu

- Tìm hiểu được một số kỹ thuật liên quan đến Classification.
- Xây dựng mô hình dự đoán khả năng mắc bệnh tim dựa vào tập dữ liệu có sẵn trên Kaggle (Heart Disease Cleveland UCI)[1]. (ban đầu chúng em thực hiện đồ án này trên dataset Heart Disease UCI, nhưng trong quá trình thực hiện và tham khảo trong phần Discussion[2] trên Kaggle thì phát hiện ra tập dữ liệu có những điểm không đúng so với bản gốc, sau đó chúng em dựa vào đề xuất của 1 người trong phần Discussion (đã được công nhận), chúng em đã sử dụng bộ dataset Heart Disease Cleveland UCI. cách triển khai xử lý bài toán được tham khảo qua một bản notebook được cộng đồng bình chọn cao nhất về tập dữ liệu này [3]).
- Tìm hiểu những tác nhân, yếu tố gây mắc bệnh tim.

B2.2. Kế hoạch

- Mỗi thành viên trong nhóm sẽ:
    + Tự tìm hiểu trước các kiến thức cơ bản cũng như một số mô hình liên quan đến Classification.
    + Nghiên cứu, khảo sát và đánh giá tham khảo một số dự án có sẵn trên cộng đồng về bài toán Heart Disease Detection hoặc những bài có liên quan.
- Phân chia một số công việc cụ thể cho từng thành viên như soạn thảo file báo cáo, làm file PowerPoint và file Code.

B2.3. Nội dung

Nội dung 1: Phân tích bộ dữ liệu lấy từ Kaggle

1.1. Tổng quan về tập dữ liệu

Bộ dữ liệu gồm có 297 mẫu dữ liệu với 13 features.

Ý nghĩa của từng features:

- age (Tuổi)
- sex (Giới tính):
    + 0: female – nữ

+ 1: male - nam

- cp (Chest pain type – Thể loại đau ngực)

+ 0: Typical Angina - Đau thắt ngực thường thấy

+ 1: Atypical Angina - Đau thắt ngực bất thường

+ 2: Asymtomatic – Không có triệu chứng

- trestbps (Resting Blood Pressure – Huyết áp)

- chol (Cholesterol – Mỡ trong máu)

- fbs (Fasting Blood Sugar – Tăng đường huyết)

+ 0: Less than 120mg/ml – dưới 120mg/ml

+ 1: Greater than 120mg/ml – trên 120mg/ml

- restecg (Resting Electrocardiographic Measurement – Chẩn đoán điện tâm đồ)

+ 0: Normal – Bình thường

+ 1: ST-T Wave Abnormality – Sóng ST-T bất thường

+ 2: Left Ventricular Hypertrophy – Mắc bệnh Phì đại tâm thất trái

- thalach (Max Heart Rate Achieved – Nhịp tim cao nhất ghi nhận được)

- exang (Exercise Induced Angina – Chứng đau thắt ngực do luyện tập)

+ 0: Yes – Có

+ 1: No - Không

- oldpeak (ST depression – Độ suy giảm của đoạn ST trong điện tâm đồ)

- slope (Dạng chênh lệch của đoạn ST trong điện tâm đồ)

+ 0: Upsloping – Chênh lên

+ 1: Flat – Phẳng

+ 2: Downsloping – Chênh xuống

- thal (Thalassemia – Bệnh tan máu bẩm sinh)

+ 0: Normal

+ 1: Fixed defect

+ 2: Reversible defect

- ca: Number of Major Vessels – Số mạch chủ quan sát được dưới ánh huỳnh quang

Thống kê dữ liệu ban đầu cho thấy:

- Dữ liệu không nhiều.

- Không có phần tử NULL

- Có 8 thuộc tính dạng rời rạc và 5 thuộc tính dạng liên tục.

- Bảng thống kê các giá trị khác nhau của mỗi thuộc tính

```
# Number of unique train observartions:
train.nunique()
```

```
age                       41
sex                        2
chest_pain_type            4
resting_blood_pressure     50
cholesterol               152
fasting_blood_sugar        2
rest_ecg                   3
max_heart_rate_achieved    91
exercise_induced_angina    2
st_depression              40
st_slope                   3
num_major_vessels          4
thalassemia                3
condition                  2
dtype: int64
```

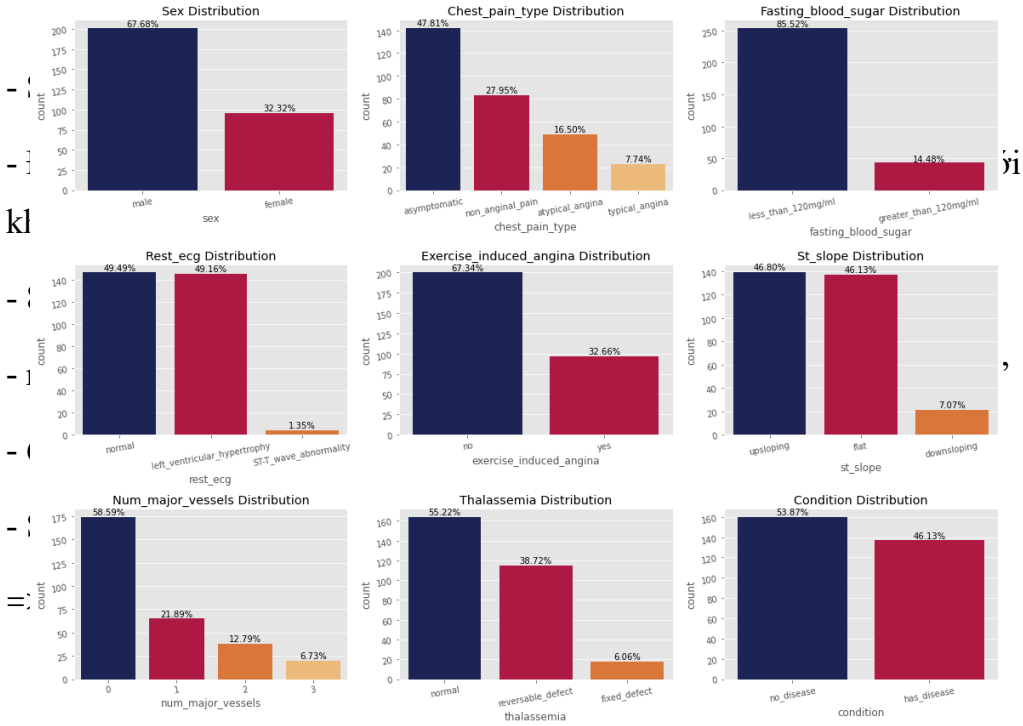
1.2. Phân tích dữ liệu

Để có thể có cái nhìn khái quát hơn về tập dữ liệu trên, ở phần này tụi em đã sử dụng một số các kĩ thuật phân tích dữ liệu cơ bản như **Phân tích đơn biến, Phân tích đa biến**, … từ đó có thể hiểu được các mối liên hệ giữa các features với nhau.

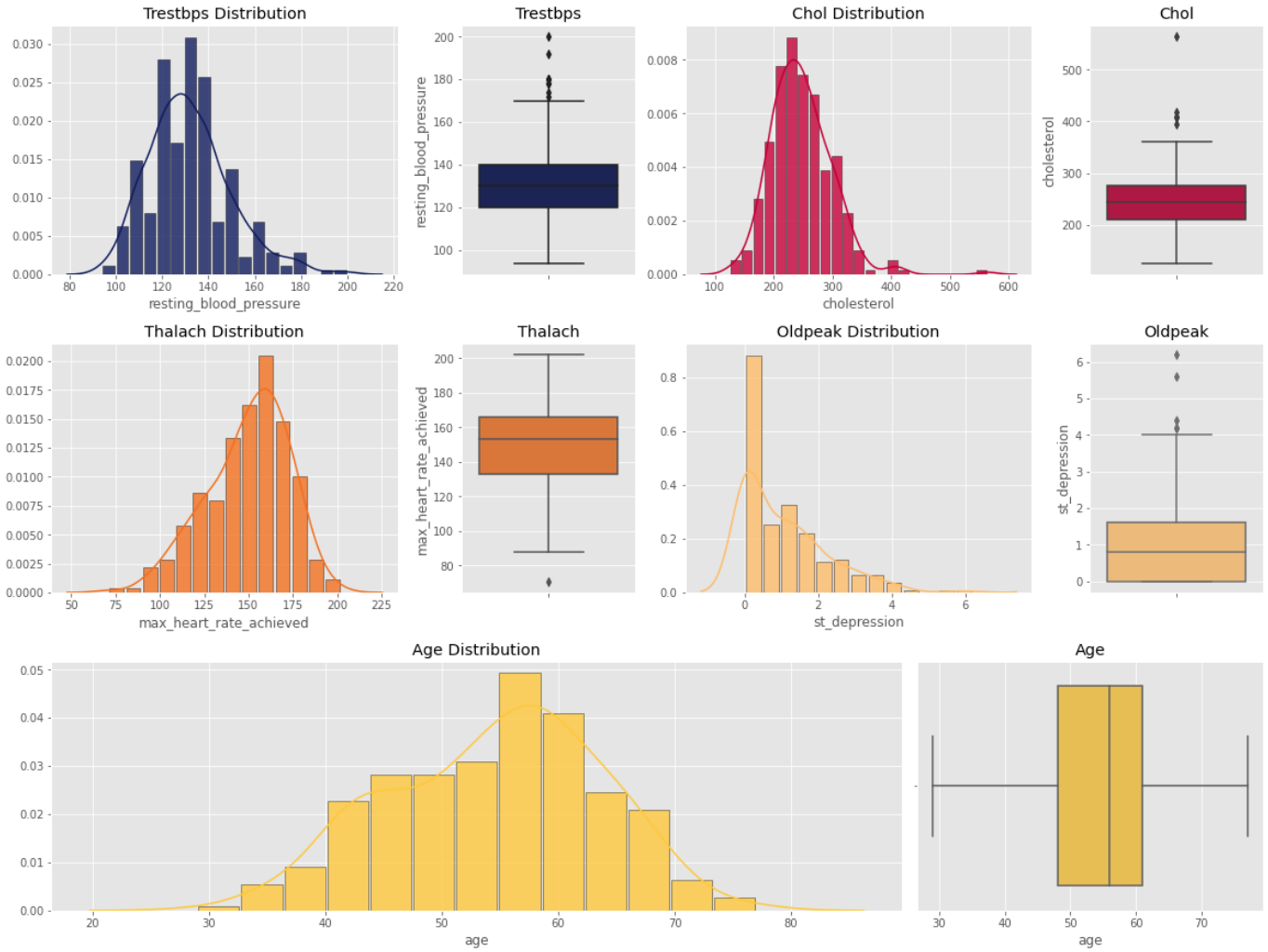
Việc này vừa có thể giúp ích được cho ta trong việc tiền xử lý dữ liệu về sau để tăng khả năng của các mô hình cũng như mang lại một ít những kiến thức đặc thù về ngành y đằng sau tập dữ liệu này.

1.2.1. Univariate Analysis – Phân tích đơn biến

a. Categorical data (Discrete data)



b. Numerical data (Continuous data

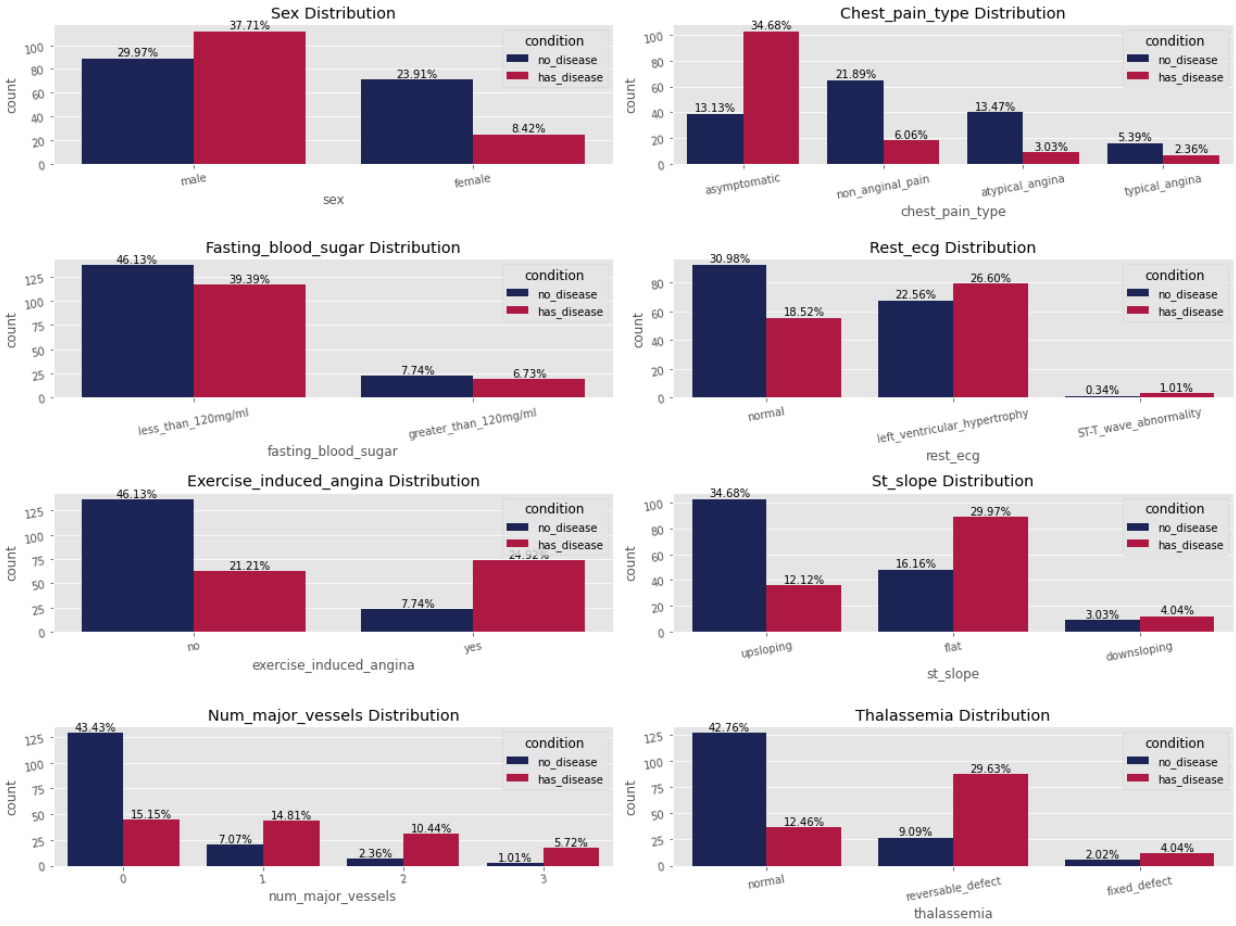


- Đa phần các biến liên tục đều tiệm cận với phân phối Gaussian và không quá nghiêng về bên trái hay bên phải, ngoại trừ biến “oldpeak”.

- Có một số trường hợp ngoại lệ, đặc biệt là ở “cholesterol”.

1.2.2. Bivariate Analysis – Phân tích lưỡng biến

a. Categorical data vs. Label



- Nam giới mắc bệnh tim nhiều hơn Nữ giới

- Thể loại đau ngực có vẻ mang tính chủ quan và không có mối liên hệ trực tiếp đến kết quả, khi mà không có triệu chứng (asymptomatic) lại có kết quả mắc bệnh nhiều nhất.

- Đường huyết không ảnh hưởng trực tiếp đến bệnh.

- Kết quả RestECG không cho kết quả trực tiếp nhưng có dạng điện tâm đồ bình thường là một dấu hiệu tốt. Mặc dù nó khá hiếm trong dữ liệu, nhưng nếu sóng ST-T bất thường thì nguy cơ mắc bệnh tim cao gấp 3 lần.

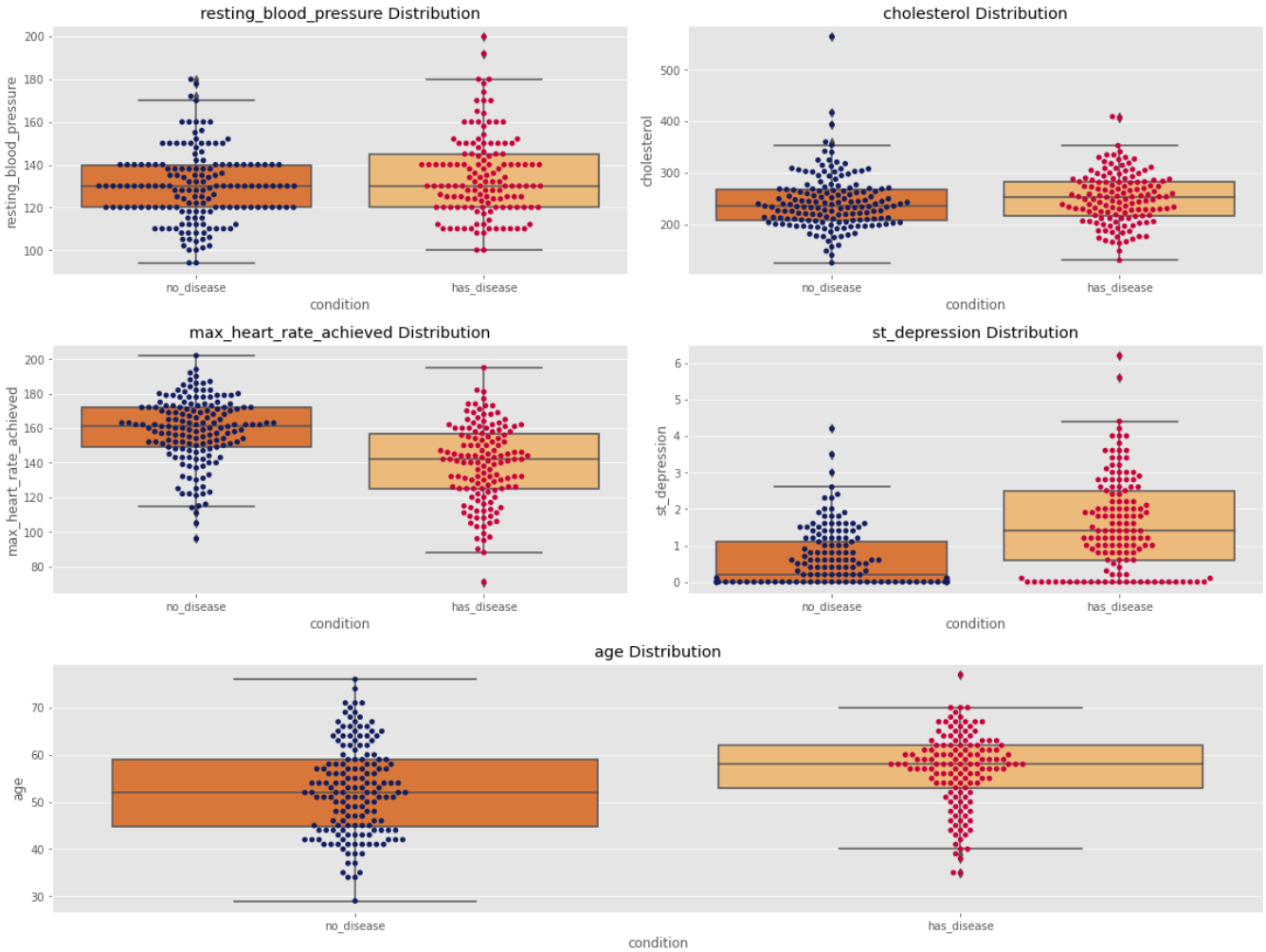
- Đau thắt ngực do luyện tập là dấu hiệu khá rõ ràng của bệnh tim, bệnh nhân có nguy cơ mắc bệnh cao gấp 3 lần nếu họ bị đau thắt ngực do luyện tập. Trong khi đó, gần một nửa số người không bị bệnh tim là không bị đau thắt ngực.

- Người có dạng chênh sóng ST (slope) là phẳng (Flat) thì có khả năng sẽ mắc bệnh tim.

- Số lượng mạch chủ quan sát được (ca) có vẻ không thể hiện rõ sự liên quan đến kết quả nhưng nếu số lượng là 0 thì đó là dấu hiệu tốt cho việc không mắc bệnh tim.

- Phát hiện có bệnh Tan máu bẩm sinh là dấu hiệu khá nghiêm trọng cho thấy khả năng cao bị bệnh tim.

b. Numerical data vs. Label



- Có huyết áp cao cho thấy khả năng mắc bệnh tim cao hơn một chút.

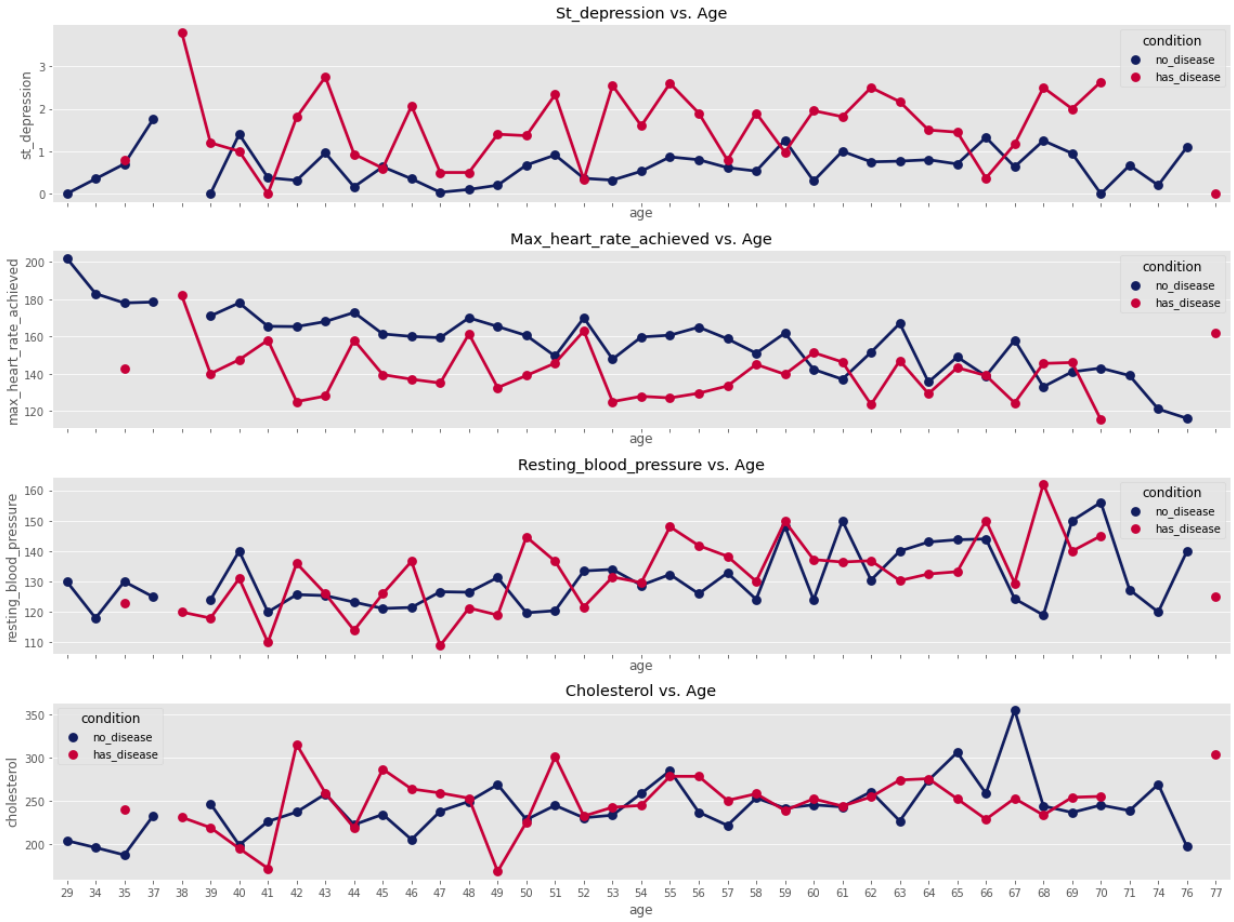
- Cholesterol cũng thế, tuy không mang tính quyết định nhưng những bệnh nhân có lượng Cholesterol cao có vẻ sẽ mắc bệnh tim. Tuy nhiên lại có trường hợp ngoại lệ khi Cholesterol ở mức quá cao nhưng vẫn không bị bệnh tim.

- Khả năng mắc bệnh tim cũng tăng theo Độ suy giảm đoạn ST (ST Depression).

- Người lớn tuổi sẽ có nguy cơ mắc bệnh tim cao hơn.

1.2.3. Multivariate Analysis – Phân tích đa biến

- Theo các biểu đồ chỉ ra thì độ tuổi có sự phân biệt rõ giữa hai trường hợp mắc bệnh và không mắc bệnh. Vì thế chúng em tiến hành tìm hiểu sâu hơn về yếu tố này.



Tuy nhiên, sự liên kết giữa độ tuổi với các yếu tố khác lại không cho thấy những quy luật rõ rệt nào.

1.2.4. Correlations - Hệ số tương quan

Chúng em dùng tương quan Pearson để tìm ra mối liên hệ tuyến tính giữa các thuộc tính, sử dụng bản đồ nhiệt để thể hiện mức độ của những mối liên hệ này.

Công thức tính hệ số tương quan Pearson là:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

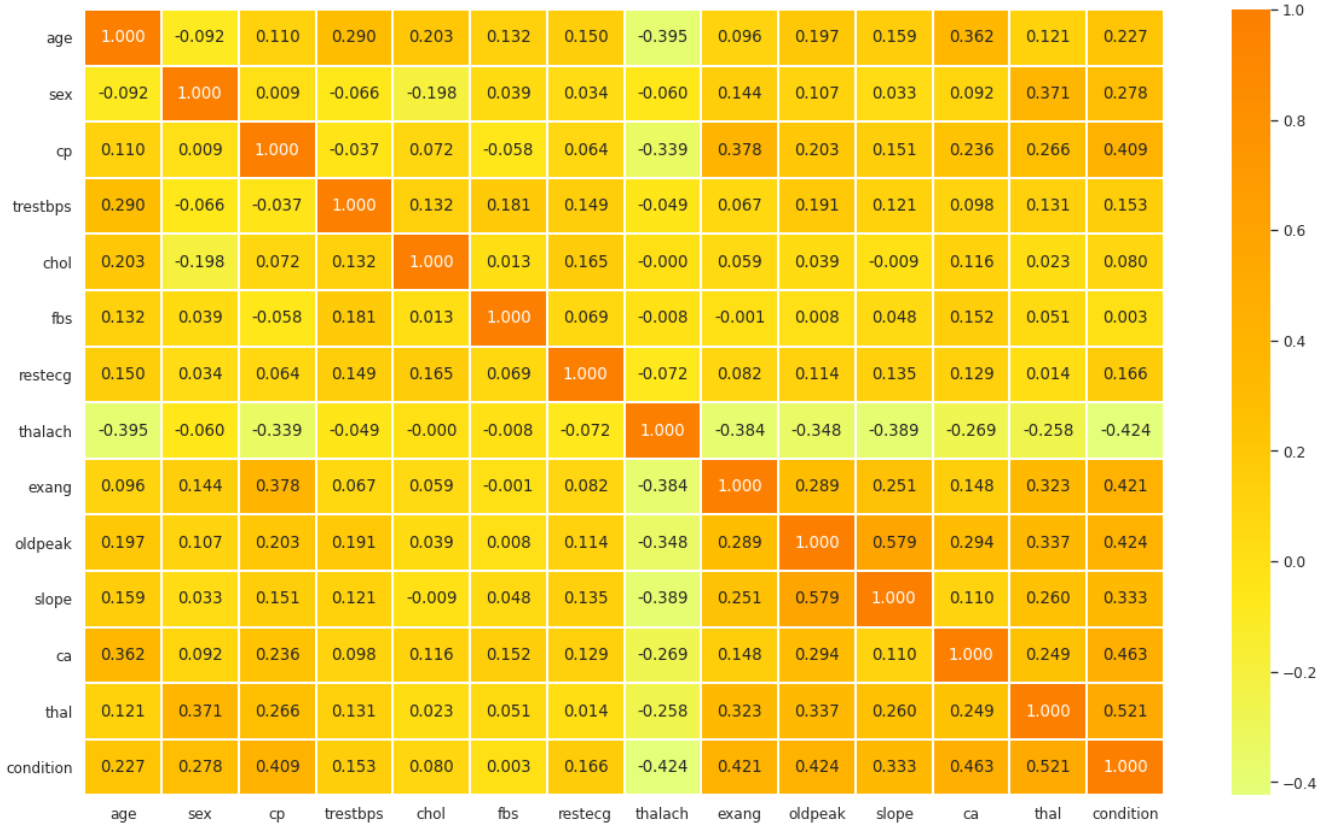
r = correlation coefficient

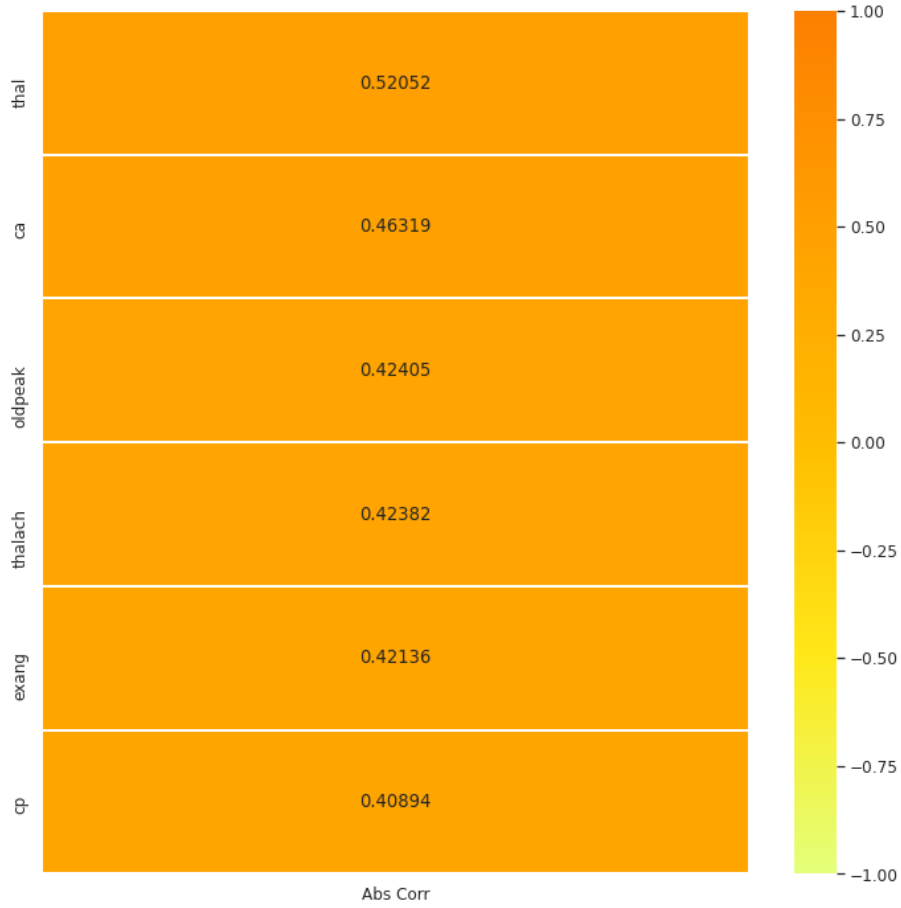
x_i = values of the x-variable in a sample

$\bar{x}$ = mean of the values of the x-variable

y_i = values of the y-variable in a sample

$\bar{y}$ = mean of the values of the y-variable


age 1.000 -0.092 0.110 0.290 0.203 0.132 0.150 -0.395 0.096 0.197 0.159 0.362 0.121 0.227
sex -0.092 1.000 0.009 -0.066 -0.198 0.039 0.034 -0.060 0.144 0.107 0.033 0.092 0.371 0.278
cp 0.110 0.009 1.000 -0.037 0.072 -0.058 0.064 -0.339 0.378 0.203 0.151 0.236 0.266 0.409
trestbps 0.290 -0.066 -0.037 1.000 0.132 0.181 0.149 -0.049 0.067 0.191 0.121 0.098 0.131 0.153
chol 0.203 -0.198 0.072 0.132 1.000 0.013 0.165 -0.000 0.059 0.039 -0.009 0.116 0.023 0.080
fbs 0.132 0.039 -0.058 0.181 0.013 1.000 0.069 -0.008 -0.001 0.008 0.048 0.152 0.051 0.003
restecg 0.150 0.034 0.064 0.149 0.165 0.069 1.000 -0.072 0.082 0.114 0.135 0.129 0.014 0.166
thalach -0.395 -0.060 -0.339 -0.049 -0.000 -0.008 -0.072 1.000 -0.384 -0.348 -0.389 -0.269 -0.258 -0.424
exang 0.096 0.144 0.378 0.067 0.059 -0.001 0.082 -0.384 1.000 0.289 0.251 0.148 0.323 0.421
oldpeak 0.197 0.107 0.203 0.191 0.039 0.008 0.114 -0.348 0.289 1.000 0.579 0.294 0.337 0.424
slope 0.159 0.033 0.151 0.121 -0.009 0.048 0.135 -0.389 0.251 0.579 1.000 0.110 0.260 0.333
ca 0.362 0.092 0.236 0.098 0.116 0.152 0.129 -0.269 0.148 0.294 0.110 1.000 0.249 0.463
thal 0.121 0.371 0.266 0.131 0.023 0.051 0.014 -0.258 0.323 0.337 0.260 0.249 1.000 0.521
condition 0.227 0.278 0.409 0.153 0.080 0.003 0.166 -0.424 0.421 0.424 0.333 0.463 0.521 1.000
age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca thal condition
1.0 0.8 0.6 0.4 0.2 0.0 −0.2 −0.4


thal 0.52052
ca 0.46319
oldpeak 0.42405
thalach 0.42382
exang 0.42136
cp 0.40894
Abs Corr
1.00 0.75 0.50 0.25 0.00 −0.25 −0.50 −0.75 −1.00

=> Thông qua bản đồ nhiệt trên thì thuộc tính "thal" – bệnh Tan máu bẩm sinh là có tính tương quan nhất.

1.3. Đánh giá sơ bộ

Thông qua việc phân tích bộ dữ liệu bằng một vài kĩ thuật phân tích cơ bản, nhóm chúng em đã có một vài đánh giá sơ bộ như sau:

- Vấn đề lớn nhất của chúng ta đó chính là **quá ít dữ liệu**, không những có thể sẽ khiến cho một số mô hình không thể phát huy hết khả năng nội tại của nó mà còn với một bài toán đặc thù mang tính cộng đồng như trên cũng sẽ ảnh hưởng nhiều đến tính thực tế, mức độ rủi ro, ... vì nó cần độ chính xác gần như tuyệt đối.

- Dựa vào các kết quả phân tích ban đầu của từng kĩ thuật, ta có thể thấy có một số features có sự ảnh hưởng lớn hơn các features khác về việc bệnh nhân có mắc bệnh hay không như **Thal (Bệnh tan máu bẩm sinh), ca (Số mạch chủ quan sát được dưới ánh huỳnh quang), age (Độ tuổi),** ... Nhưng với lý do dữ liệu ít kèm với việc chưa thật sự chắc chắn về các kết quả phân tích trên vì ở mỗi kiểu phân tích lại cho ta một kết quả khác nhau nên tụi em đã quyết định không bỏ một features nào để training.

Nội dung 2: Xây dựng mô hình dự đoán khả năng mắc bệnh tim.

Trong đồ án này, chúng em sẽ dùng những mô hình bao gồm Support Vector Machine (SVM), Naive Bayes, Logistics Regression và Decision Tree để giải quyết bài toán.

Ngoài việc chỉ sử dụng các classifiers trên, chúng em cũng đã cố gắng sử dụng thêm một số các phương pháp, kĩ thuật nhằm nâng cao khả năng của các mô hình một cách tối ưu nhất dựa trên những modules có sẵn trong thư viện scikit-learn.

2.1. Tổng quan về các mô hình.

a. Support Vector Machine (SVM)

Support Vector Machine (SVM) là một tập hợp các phương pháp học có giám sát được sử dụng để phân loại, hồi quy và phát hiện ngoại lệ.

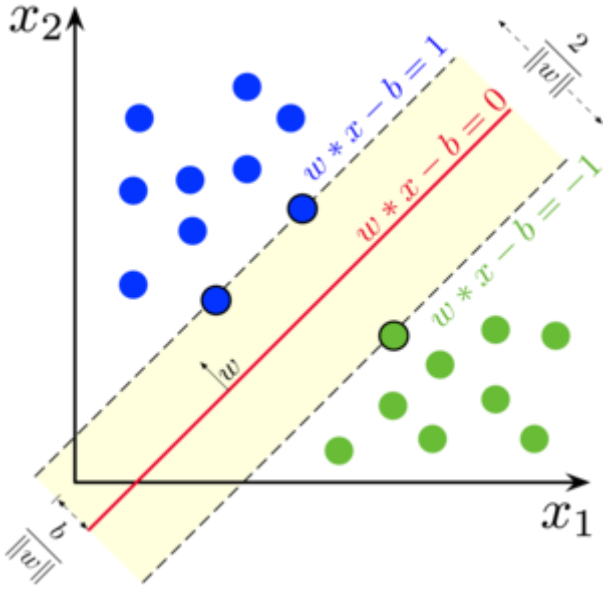
- Ưu điểm của SVM là:

+ Hiệu quả trong không gian nhiều chiều.

+ Vẫn hiệu quả trong trường hợp số chiều lớn hơn số lượng mẫu dữ liệu.

+ Sử dụng một tập con các điểm huấn luyện (training points) trong hàm quyết định (gọi là support vector), do đó, nó cũng đem lại hiệu quả về bộ nhớ.

+ Đa năng: các hàm Kernel khác nhau có thể được dùng cho hàm quyết định.

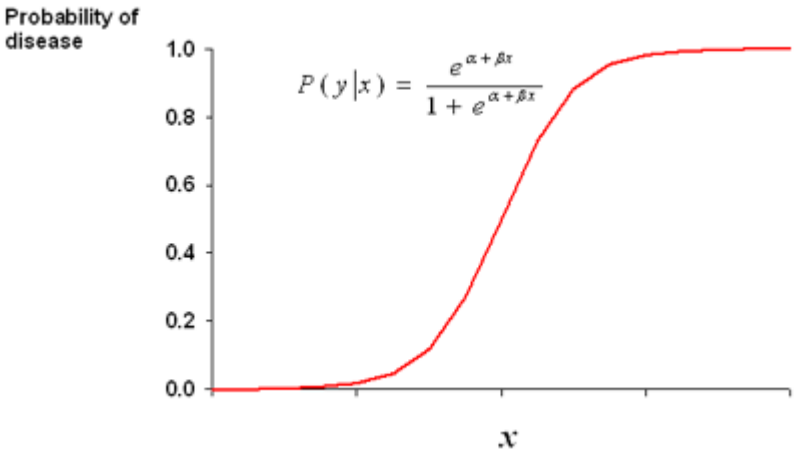


(Support-vector machine, n.d.)

b. Logistic Regression

Logistic regression phù hợp để áp dụng cho bài toán khi biến phụ thuộc có dạng nhị phân. Giống như tất cả các thuật toán phân tích hồi quy, logistic Regression là một loại phân tích dự đoán.

Logistic Regression được sử dụng để mô tả dữ liệu và giải thích mối quan hệ giữa một biến nhị phân phụ thuộc với một hoặc nhiều biến độc lập như thứ tự, khoảng hoặc tỷ lệ.



(Lesson 12: Statistical Methods (2) Logistic Regression, Poisson Regression, n.d.)

c. Naive Bayes

Naive Bayes là một thuật toán học đơn giản sử dụng quy tắc Bayes cùng với một nhận định rằng các thuộc tính là độc lập có điều kiện.

Trong khi nhận định về tính độc lập này thường bị vi phạm trong thực tế, tuy nhiên Bayers đem lại độ chính xác cho các bài toán phân loại. Cùng với hiệu quả tính toán và nhiều tính năng khác, Naive Bayers đang được áp dụng rộng rãi trong thực tế.

$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$$

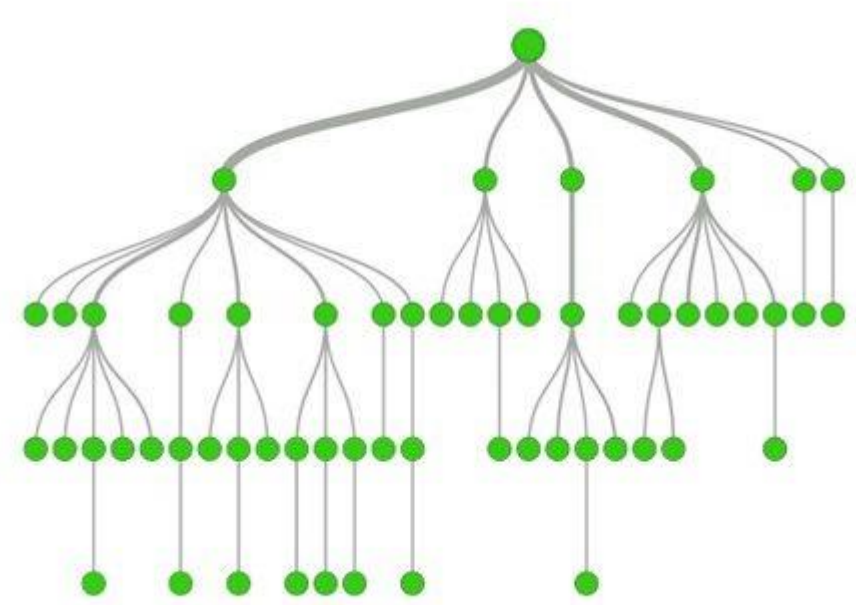
(Naive Bayes Theorem, n.d.)

d. Decision Tree

Decision Tree là một phương pháp học tập có giám sát phi tham số được sử dụng để phân loại và hồi quy.

Mục đích là tạo ra một mô hình có thể dự đoán giá trị của một biến mục tiêu bằng cách tìm hiểu các quy tắc quyết định đơn giản được suy ra từ dữ liệu của các thuộc tính.

Decision Tree học từ dữ liệu để tính gần đúng đường cong sin với một tập hợp các quy tắc quyết định if-then-else. Cây càng sâu, các quy tắc quyết định càng phức tạp và mô hình càng hoàn thiện.



(Decision Trees Tutorial, n.d.)

2.2. Xây dựng và cải thiện mô hình

a. Train lần đầu

Sau khi đã tiền xử lý dữ liệu và train model, chúng em có bảng kết quả ban đầu như sau:

	Model Name	Train Accuracy Mean	Test Accuracy Mean	Train Precision Mean	Test Precision Mean	Train Recall Mean	Test Recall Mean	Train F1 Mean	Test F1 Mean	Time
1	SVC	0.678453	0.666836	0.577783	0.562758	0.482061	0.465777	0.577783	0.562758	0.011796
0	DecisionTreeClassifier	1.000000	0.730395	1.000000	0.706855	1.000000	0.712227	1.000000	0.706855	0.007125
2	GaussianNB	0.836702	0.811695	0.814794	0.790126	0.783448	0.772559	0.814794	0.790126	0.006909
3	LogisticRegression	0.872070	0.851921	0.856401	0.833685	0.826599	0.809596	0.856401	0.833685	0.057686

Có thể thấy SVC (SVM) có kết quả thấp nhất và Decision Tree mắc phải vấn đề Overfitting.

b. Tinh chỉnh model

*__Tiền xử lý dữ liệu trước khi tinh chỉnh: c__húng em quyết định sẽ loại bỏ những điểm ngoại lệ hay nhiễu loạn.

Isolation Forest

Isolation Forest 'cô lập' các quan sát bằng cách chọn ngẫu nhiên một thuộc tính và sau đó chọn ngẫu nhiên một giá trị phân tách giữa các giá trị lớn nhất và nhỏ nhất của thuộc tính đã chọn.

Dưới đây là bảng kết quả, có thể thấy score vẫn chưa được cải thiện tốt.

	Model Name	Train Accuracy Mean	Test Accuracy Mean	Train Precision Mean	Test Precision Mean	Train Recall Mean	Test Recall Mean	Train F1 Mean	Test F1 Mean	Time
1	SVC	0.676034	0.655625	0.538764	0.510568	0.429837	0.404761	0.538764	0.510568	0.006366
0	DecisionTreeClassifier	1.000000	0.756674	1.000000	0.720922	1.000000	0.714967	1.000000	0.720922	0.004861
2	GaussianNB	0.834259	0.808945	0.817561	0.793670	0.828498	0.816967	0.817561	0.793670	0.003673
3	LogisticRegression	0.874543	0.850105	0.854300	0.818976	0.820250	0.790663	0.854300	0.818976	0.029504

Elliptic Envelope

Chúng em nhận định rằng các thuộc tính đều có phân phối Gaussian nên đã dùng đường cong Elliptic để xử lý dữ liệu với phương pháp MCD.

MCD (The Minium Covariance Determinant) là một công cụ đo đạc rất tốt về vị trí và độ phân tán đa biến. Nó cũng đóng vai trò như một công cụ tiện lợi và hiệu quả để phát hiện các điểm ngoại lệ.

So với Isolation Forest thì kết quả đã tốt hơn:

	Model Name	Train Accuracy Mean	Test Accuracy Mean	Train Precision Mean	Test Precision Mean	Train Recall Mean	Test Recall Mean	Train F1 Mean	Test F1 Mean	Time
1	SVC	0.683533	0.639972	0.584611	0.511984	0.484960	0.412848	0.584611	0.511984	0.009940
0	DecisionTreeClassifier	1.000000	0.757023	1.000000	0.708480	1.000000	0.694421	1.000000	0.708480	0.009505
2	GaussianNB	0.850226	0.809154	0.831201	0.790047	0.811958	0.802283	0.831201	0.790047	0.005228
3	LogisticRegression	0.889513	0.865339	0.875353	0.844419	0.843333	0.826622	0.875353	0.844419	0.049127

Discretization

Discretization (rời rạc hóa hay lượng tử hóa) là phương pháp để phân chia các thuộc tính liên tục thành các giá trị rời rạc.

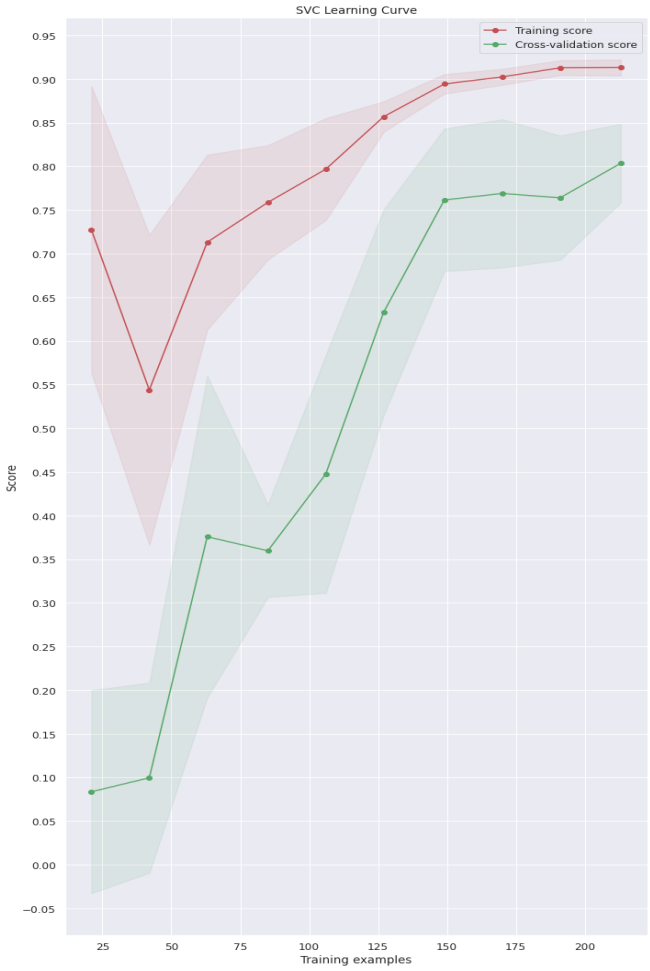
K-Bbins Discretization: Phương pháp này rời rạc hóa các thuộc tính liên tục thành các ‘k’ bin. Chúng em dùng tham số 'k-means' để xác định các bin dựa vào quy trình phân cụm k-mean được thực hiện trên từng thuộc tính một cách độc lập.

Dưới đây là bảng kết quả, có thể thấy SVM đã được cải thiện rõ rệt, tuy nhiên Decision Tree vẫn bị overfitting.

	Model Name	Train Accuracy Mean	Test Accuracy Mean	Train Precision Mean	Test Precision Mean	Train Recall Mean	Test Recall Mean	Train F1 Mean	Test F1 Mean	Time
2	GaussianNB	0.791225	0.763802	0.731303	0.692745	0.633962	0.605826	0.731303	0.692745	0.005515
0	DecisionTreeClassifier	1.000000	0.756604	1.000000	0.729606	1.000000	0.722262	1.000000	0.729606	0.007175
3	LogisticRegression	0.897008	0.839203	0.884743	0.808287	0.861297	0.789717	0.884743	0.808287	0.022548
1	SVC	0.934452	0.839203	0.928033	0.814671	0.916766	0.794479	0.928033	0.814671	0.009068

*Mô tả quá trình học của model: Learning Curve

Learning Curve có thể chỉ ra tiến độ học của model và đặc biệt là cách xử lý với dữ liệu của model đó, từ đó chúng ta có thể đưa ra những quyết định khác nếu model cần nhiều dữ liệu hơn để có kết quả tốt hơn.



=> Mô hình sử dụng Decision Tree bị overfitting và các mô hình khác có thể sẽ kết quả tốt hơn khi ta có nhiều dữ liệu hơn

***Tinh chỉnh mô hình sử dụng RandomizedSearchCV**

Để khắc phục tình trạng overfitting của mô hình Decision Tree cũng như cải thiện thêm kết quả của các mô hình khác, chúng em đã quyết định tiến hành tinh chỉnh lại các tham số đầu vào của tất cả models nhằm tìm ra một bộ tham số phù hợp nhất. Các tham số đầu vào mà ta đã sử dụng của từng models như sau:

```
SVC params used:  {'C': 1.0, 'break_ties': False, 'cache_size': 200, 'class_weight': None, 'coef0': 0.0, 'decision_function_shape': 'ov
r', 'degree': 3, 'gamma': 'scale', 'kernel': 'rbf', 'max_iter': -1, 'probability': False, 'random_state': None, 'shrinking': True, 'to
l': 0.001, 'verbose': False}

Decision Tree params used:  {'ccp_alpha': 0.0, 'class_weight': None, 'criterion': 'gini', 'max_depth': None, 'max_features': None, 'max
_leaf_nodes': None, 'min_impurity_decrease': 0.0, 'min_impurity_split': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'min_weigh
t_fraction_leaf': 0.0, 'presort': 'deprecated', 'random_state': 42, 'splitter': 'best'}

Gaussian Naive Bayes params used:  {'priors': None, 'var_smoothing': 1e-09}

Logistic Regression params used:  {'C': 1.0, 'class_weight': None, 'dual': False, 'fit_intercept': True, 'intercept_scaling': 1, 'l1_ra
tio': None, 'max_iter': 100, 'multi_class': 'auto', 'n_jobs': None, 'penalty': 'l2', 'random_state': None, 'solver': 'lbfgs', 'tol': 0.
0001, 'verbose': 0, 'warm_start': False}
```

Sau khi tham khảo qua một số tài liệu [4][5][6][7], tụi em đã quyết định chọn một vài các tham số đầu vào như hình như sau kèm với một vài các giá trị cụ thể để xem rằng, bộ thông số nào sẽ là tốt nhất cho các mô hình của chúng ta.

```
====================
Updated Parameters for SVC
Cross Val Mean: 0.797, Cross Val Stdev: 0.056
Best Score: 0.797
Best Parameters: {'shrinking': True, 'kernel': 'sigmoid', 'gamma': 0.01, 'C': 10}
Elapsed Time: 00:00:11
====================
====================
Updated Parameters for DecisionTreeClassifier
Cross Val Mean: 0.730, Cross Val Stdev: 0.081
Best Score: 0.764
Best Parameters: {'min_samples_split': 7, 'min_samples_leaf': 4, 'max_depth': 3, 'criterion': 'entropy'}
Elapsed Time: 00:00:05
====================
====================
Updated Parameters for GaussianNB
Cross Val Mean: 0.772, Cross Val Stdev: 0.055
Best Score: 0.783
Best Parameters: {'var_smoothing': 0.08111308307896872}
Elapsed Time: 00:00:04
====================
====================
Updated Parameters for LogisticRegression
Cross Val Mean: 0.795, Cross Val Stdev: 0.049
Best Score: 0.803
Best Parameters: {'C': 10}
Elapsed Time: 00:00:04
====================
```

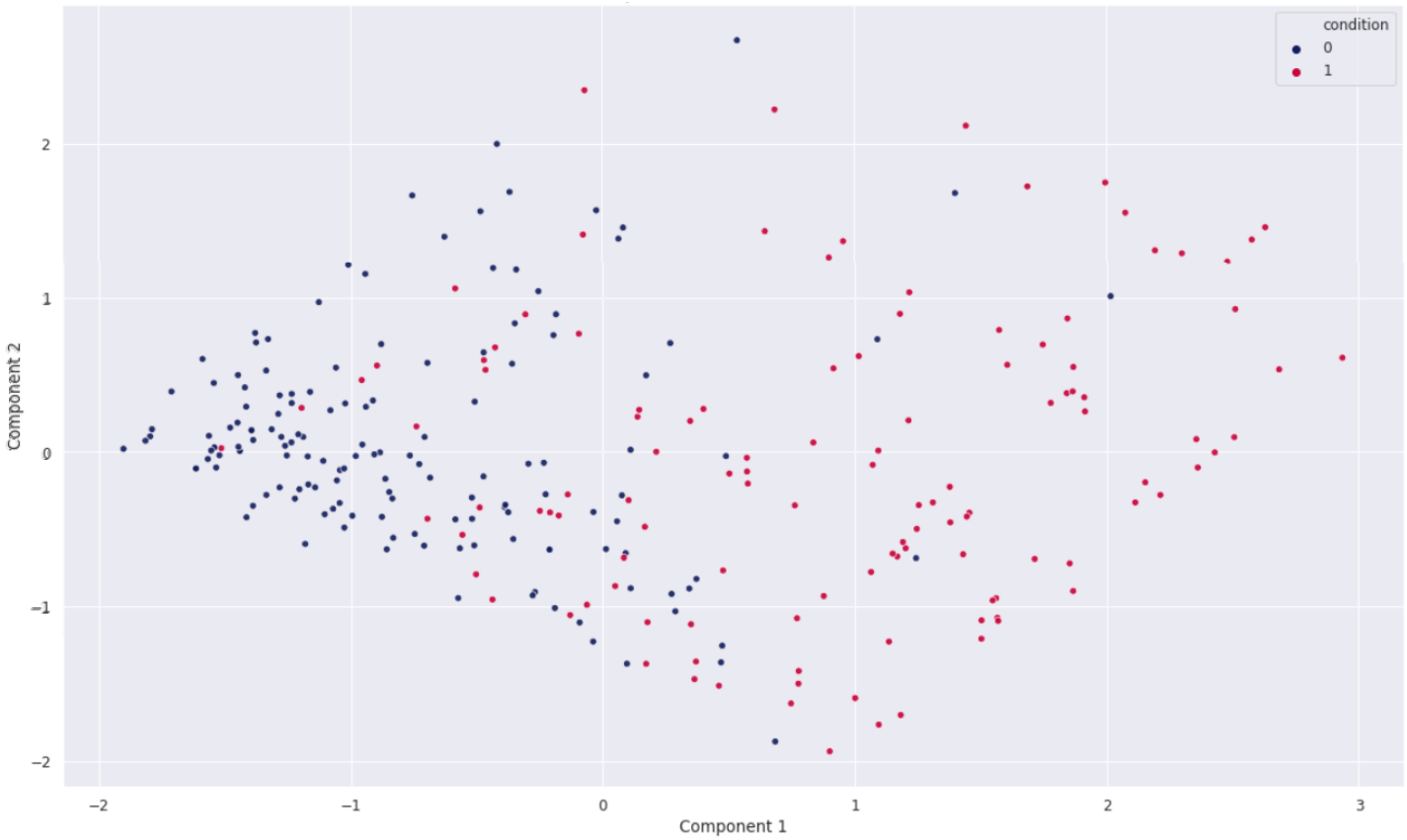
Kết quả sau khi tinh chỉnh:

	Model Name	Train Accuracy Mean	Test Accuracy Mean	Train Precision Mean	Test Precision Mean	Train Recall Mean	Test Recall Mean	Train F1 Mean	Test F1 Mean	Time
0	DecisionTreeClassifier	0.853003	0.805031	0.825365	0.763980	0.788585	0.727746	0.825365	0.763980	0.004937
2	GaussianNB	0.840841	0.820335	0.814582	0.783239	0.793942	0.758968	0.814582	0.783239	0.004072
1	SVC	0.867053	0.831377	0.840154	0.797228	0.792161	0.760413	0.840154	0.797228	0.007514
3	LogisticRegression	0.890444	0.831307	0.871579	0.803208	0.840949	0.784968	0.871579	0.803208	0.022192

Có thể thấy mặc dù chỉ với việc điều chỉnh lại một vài thông số thôi những chúng ta đã khắc phuc được tình trạng overfitting của mô hình Decision Tree.

c. Giảm chiều dữ liệu sử dụng phương pháp PCA

Principal Components Analysis (PCA) là một thuật toán thống kê sử dụng phép biến đổi trực giao để biến đổi một tập hợp dữ liệu từ một không gian nhiều chiều sang một không gian mới ít chiều hơn (2 hoặc 3 chiều) nhằm tối ưu hóa việc thể hiện sự biến thiên của dữ liệu.



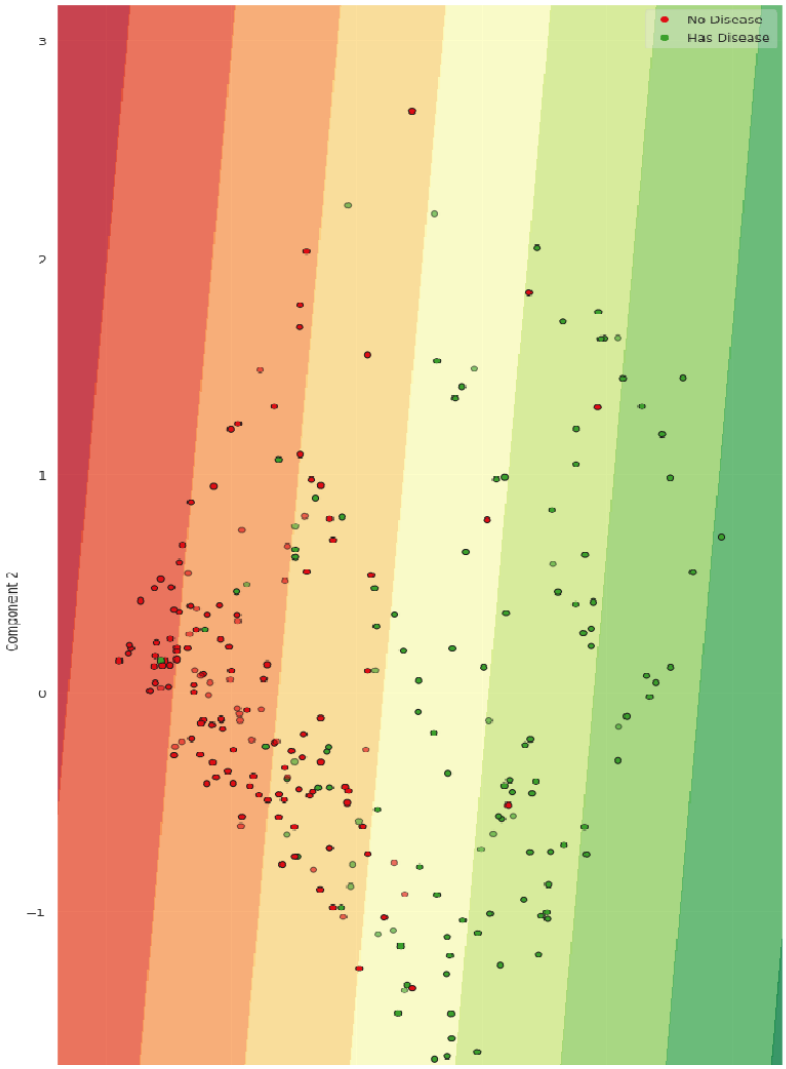
Kết quả sau khi giảm chiều dữ liệu với tham số PCA = 2

	Model Name	Train Accuracy Mean	Test Accuracy Mean	Train Precision Mean	Test Precision Mean	Train Recall Mean	Test Recall Mean	Train F1 Mean	Test F1 Mean	Time
0	DecisionTreeClassifier	0.889505	0.839203	0.874211	0.809155	0.833363	0.775069	0.874211	0.809155	0.002397
2	GaussianNB	0.849265	0.850314	0.817168	0.810715	0.735099	0.742662	0.817168	0.810715	0.001790
3	LogisticRegression	0.855820	0.854018	0.827329	0.819446	0.753440	0.760881	0.827329	0.819446	0.004640
1	SVC	0.854881	0.857722	0.826275	0.822169	0.753401	0.752185	0.826275	0.822169	0.003215

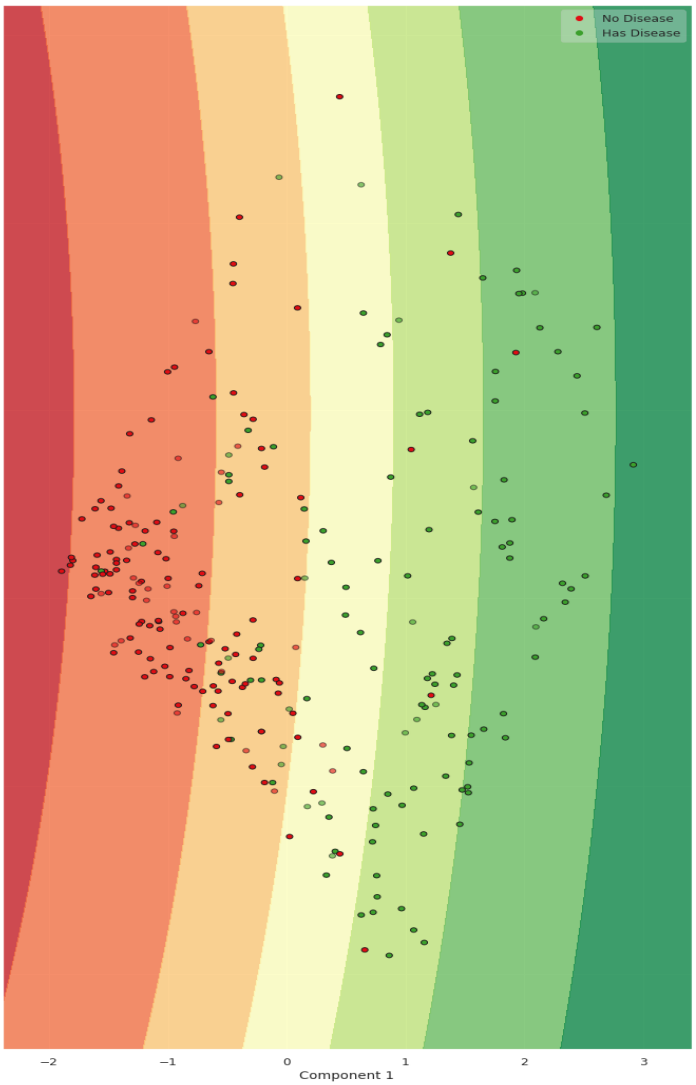
d. Decision Regions

Decision Tree

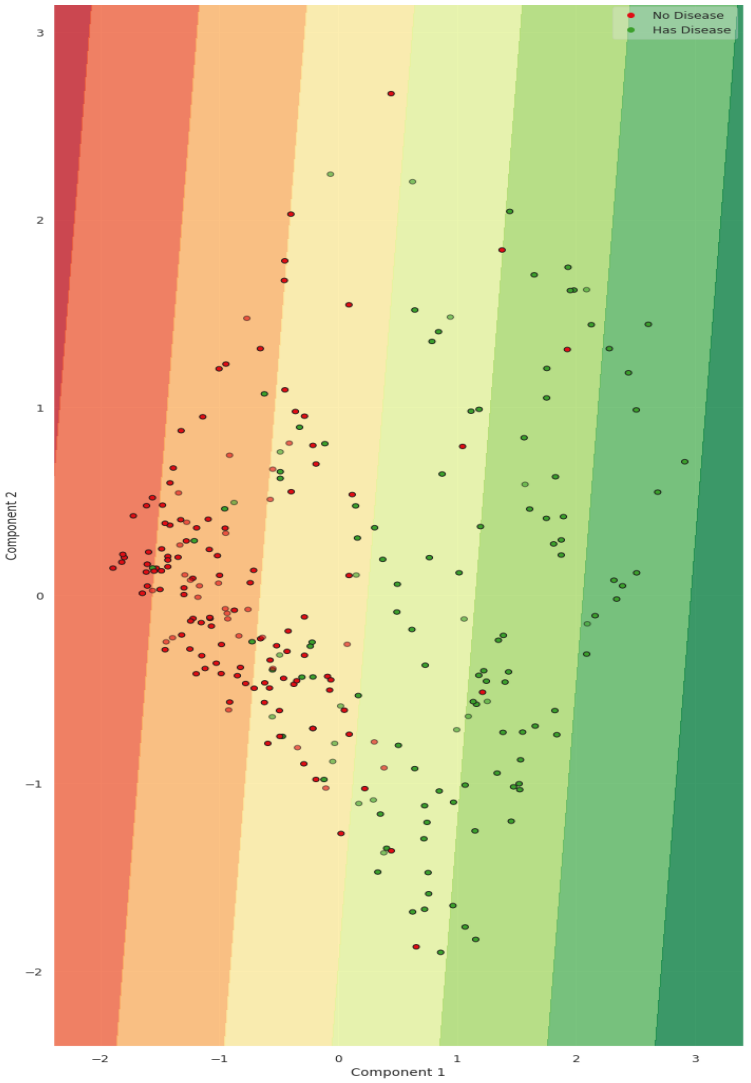
Support Vector Machine



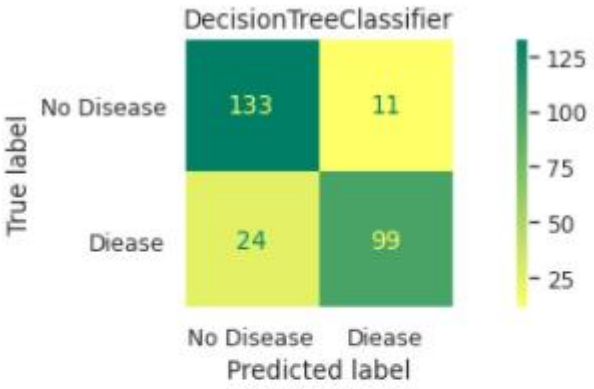
Naïve Bayes

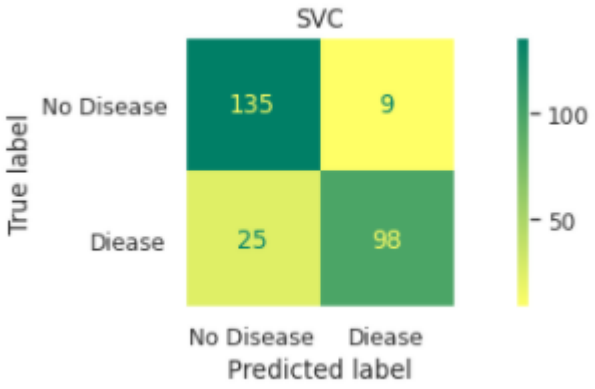


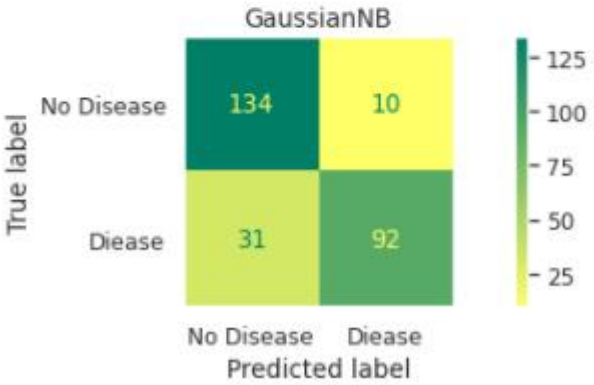
Logistic Regression



e. Confusion Matrix







Nội dung 3: Tổng kết đồ án

Sau quá trình phân tích dữ liệu thì chúng em kết luận rằng bệnh Tan máu bẩm sinh và Tuổi tác là 2 yếu tố liên quan nhất đến việc một người có mắc bệnh tim hay không, có bệnh tan máu bẩm sinh và tuổi tác cao sẽ tăng nguy cơ mắc bệnh tim hơn hẳn những yếu tố khác.

Chúng em đã tìm hiểu được một số cách xử lý dữ liệu nhằm tối ưu hóa model.

Từ bảng kết quả cuối cùng, chúng em rút ra nhận định rằng mô hình SVM và Logistic Regression xử lý tốt trên bộ dữ liệu đã cho và có độ chính xác tốt nhất.

Mô hình đã xây dựng cho kết quả dự đoán bệnh tim ở mức chấp nhận được.

B3. Kết quả

- File trình chiếu PowerPoint
- File Code
- File báo cáo PDF

B4. Tài liệu tham khảo

[1] “Heart Disease Cleveland UCI.” https://kaggle.com/cherngs/heart-disease-cleveland-uci (accessed Jan. 27, 2021).
[2] “Heart Disease UCI.” https://kaggle.com/ronitf/heart-disease-uci (accessed Jan. 27, 2021).
[3] “Heart Disease and Some scikit-learn Magic.” https://kaggle.com/datafan07/heart-disease-and-some-scikit-learn-magic (accessed Jan. 27, 2021).
[4] S. Yıldırım, “Hyperparameter Tuning for Support Vector Machines — C and Gamma Parameters,” *Medium*, Jun. 01, 2020. https://towardsdatascience.com/hyperparameter-tuning-for-support-vector-machines-c-and-gamma-parameters-6a5097416167 (accessed Jan. 27, 2021).
[5] “2. Tuning parameters for logistic regression.” https://kaggle.com/joparga3/2-tuning-parameters-for-logistic-regression (accessed Jan. 27, 2021).
[6] M. Mithrakumar, “How to tune a Decision Tree?,” *Medium*, Nov. 12, 2019. https://towardsdatascience.com/how-to-tune-a-decision-tree-f03721801680 (accessed Jan. 27, 2021).
[7] “machine learning - How to tune GaussianNB?,” *Stack Overflow*. https://stackoverflow.com/questions/39828535/how-to-tune-gaussiannb (accessed Jan. 27, 2021).

--Hết--