

**BỘ GIÁO DỤC VÀ ĐÀO TẠO      BỘ NÔNG NGHIỆP VÀ PTNT**  
**TRƯỜNG ĐẠI HỌC THỦY LỢI**



**ĐÀO TRUNG HIẾU**

**ỨNG DỤNG MÔ HÌNH HỒI QUY TRONG HỖ TRỢ CHẨN  
ĐOÁN BỆNH TIM**

**ĐỒ ÁN TỐT NGHIỆP**

**HÀ NỘI, NĂM 2022**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO    BỘ NÔNG NGHIỆP VÀ PTNT**  
**TRƯỜNG ĐẠI HỌC THỦY LỢI**

**ĐÀO TRUNG HIẾU**

**ỨNG DỤNG MÔ HÌNH HỒI QUY TRONG HỖ TRỢ CHẨN  
ĐOÁN BỆNH TIM**

Ngành: Kỹ thuật phần mềm

Mã số: 7480103

NGƯỜI HƯỚNG DẪN 1. TS Lương Thị Hồng Lan

HÀ NỘI, NĂM 2022



**CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập - Tự do - Hạnh phúc**



**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**

**Họ tên sinh viên:** ĐÀO TRUNG HIẾU

**Hệ đào tạo:** Đại học chính quy

**Lớp:** 59PM1

**Ngành:** Kỹ thuật phần mềm

**Khoa:** Công nghệ thông tin

**1- TÊN ĐỀ TÀI:**

Ứng dụng mô hình hồi quy trong hỗ trợ chẩn đoán bệnh tim.

**2- CÁC TÀI LIỆU CƠ BẢN:**

**3 - NỘI DUNG CÁC PHẦN THUYẾT MINH VÀ TÍNH TOÁN:**

Tỷ lệ %

**5. GIÁO VIÊN HƯỚNG DẪN TỪNG PHẦN**

## **6. NGÀY GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**

Ngày ..... tháng ..... năm 2022

**Trưởng Bộ môn**

*(Ký và ghi rõ Họ tên)*

**Giáo viên hướng dẫn chính**

*(Ký và ghi rõ Họ tên)*

Nhiệm vụ Đồ án tốt nghiệp đã được Hội đồng thi tốt nghiệp của Khoa thông qua.

Ngày...tháng...năm 2022

**Chủ tịch Hội đồng**

*(Ký và ghi rõ Họ tên)*

Sinh viên đã hoàn thành và nộp bản Đồ án tốt nghiệp cho Hội đồng thi ngày... tháng... năm 2022

**Sinh viên làm Đồ án tốt nghiệp**

*(Ký và ghi rõ Họ tên)*



TRƯỜNG ĐẠI HỌC THỦY LỢI  
KHOA CÔNG NGHỆ THÔNG TIN

BẢN TÓM TẮT ĐỀ CƯƠNG ĐỒ ÁN TỐT NGHIỆP

**TÊN ĐỀ TÀI:** ỨNG DỤNG MÔ HÌNH HỒI QUY TRONG HỖ TRỢ CHẨN ĐOÁN BỆNH TIM

*Sinh viên thực hiện:* Đào Trung Hiếu

*Lớp:* 59PM1

*Giáo viên hướng dẫn:* TS Lương Thị Hồng Lan.

**TÓM TẮT ĐỀ TÀI**

Theo Health Việt Nam hiện nay bệnh tim mạch đã trở thành nguyên nhân gây tử vong hàng đầu trên toàn cầu. Những năm đầu thế kỷ 21, số lượng người mắc và tử vong do bệnh tim mạch ước tính khoảng gần 18 triệu người trên toàn thế giới và chiếm hơn 30% nguyên nhân gây tử vong. Điều đáng lo ngại là số lượng người mắc và chết do các bệnh tim mạch vẫn tiếp tục gia tăng rất nhanh và chiếm tỷ lệ rất lớn ở các nước đang phát triển có thu nhập trung bình - thấp.

Bài toán dự báo chẩn đoán bệnh tim là một chủ đề rất quan trọng trong lĩnh vực y học. Hiện nay việc áp dụng các công nghệ trong khám và điều trị rất được quan tâm và được nghiên cứu rộng rãi. Việc dự báo chẩn đoán hiệu quả sẽ giúp ngành y học và mỗi người có được chiến lược phòng bệnh tối ưu. Nhận thức từ ý nghĩa thực tiễn, em quyết định chọn đề tài : “*Ứng dụng mô hình hồi quy trong hỗ trợ chẩn đoán bệnh tim*”.

**CÁC MỤC TIÊU CHÍNH**

**CÁC MỤC TIÊU CHÍNH**

- Tìm hiểu về các mô hình phân lớp, mô hình hồi quy
- Tìm hiểu về thuật toán hồi quy tuyến tính, hồi quy logic và cây quyết định.
- Tìm hiểu về bài toán chẩn đoán bệnh tim.
- Ứng dụng mô hình hồi quy trong chẩn đoán bệnh tim mạch.
- Tìm hiểu ngôn ngữ Python và các thư viện hỗ trợ

## KẾT QUẢ DỰ KIẾN

- Hiểu được về các mô hình phân lớp, mô hình hồi quy
- Hiểu về các thuật toán: hồi quy tuyến tính, hồi quy logic và cây quyết định.
- Nắm được bài toán chẩn đoán bệnh tim mạch nói chung.
- Hiểu và sử dụng được ngôn ngữ Python trong xây dựng thuật toán hồi quy đối với bài toán chẩn đoán bệnh tim mạch.
- Tập dữ liệu được lấy từ Heart-2020-cleaned.csv có 300000+ mẫu gồm 18 thuộc tính là :
  - (1) 9 booleans( HeartDisease, Smoking , AlcoholDrinking, Stroke, DiffWalking, Diabetic, PhysicalActivity, Asthma, KidneyDisease, SkinCancer).
  - (2) 5 strings(Sex , AgeCategoryRace, Race, Diabetic,Genhealth).
  - (3) 4 decimals (BMI, PhysicalHealth, MentalHealth, SleepTime)

## **LỜI CAM ĐOAN**

Tác giả xin cam đoan đây là Đồ án tốt nghiệp của bản thân tác giả. Các kết quả trong Đồ án tốt nghiệp này là trung thực và không sao chép từ bất kỳ một nguồn nào và dưới bất kỳ hình thức nào. Việc tham khảo các nguồn tài liệu (nếu có) đã được thực hiện trích dẫn và ghi nguồn tài liệu tham khảo đúng quy định.

**Tác giả ĐATN/KLTN**

**Đào Trung Hiếu**

## **LỜI CẢM ƠN**

Trong quá trình thực hiện luận văn này, em đã nhận được rất nhiều sự động viên, giúp đỡ của nhiều cá nhân và tập thể.

Trước tiên, em xin bày tỏ lòng cảm ơn sâu sắc tới các thầy cô trong khoa Công nghệ thông tin trường Đại học Thủy Lợi đã cung cấp kiến thức, kỹ năng và truyền dạy kinh nghiệm cho em trong suốt quá trình học tập tại trường. Đặc biệt em xin được gửi lời cảm ơn TS Lương Thị Hồng Lan, đã nhiệt tình hướng dẫn, góp ý và tạo điều kiện thuận lợi để em có thể hoàn thành Đồ án tốt nghiệp một cách tốt nhất.

Cuối cùng, em cũng xin chân thành cảm ơn các anh, các chị và các bạn học lớp 59PM1 trường Đại học Thủy Lợi đã luôn động viên, giúp đỡ và nhiệt tình chia sẻ với em những kinh nghiệm học tập, công tác trong suốt khoá học.

*Hà Nội, ngày ... tháng ... năm 2022*

**Sinh viên thực hiện**

**Đào Trung Hiếu**



## MỤC LỤC

MỤC LỤC .....	iii
DANH MỤC CÁC HÌNH ẢNH .....	vi
DANH MỤC BẢNG BIỂU.....	viii
DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ .....	ix
MỞ ĐẦU .....	10
CHƯƠNG 1 TỔNG QUAN CƠ SỞ LÝ THUYẾT.....	12
1.1 Học máy (Machine learning)[2] .....	12
1.1.1 Tổng quan về học máy .....	12
1.1.2 Học có giám sát (Supervised learning).....	13
1.1.2.1 Classification (Phân lớp) .....	14
1.1.2.2 Regression (Hồi quy).....	15
1.1.3 Học không giám sát (Unsupervised learning) .....	16
1.1.3.1 Clustering (phân cụm) .....	17
1.1.3.2 Association .....	17
1.2 Tổng quan lý thuyết về thuật toán áp dụng[3] .....	18
1.2.1 Mô hình hồi quy tuyến tính .....	18
1.2.2 Mô hình hồi quy logic .....	18
1.2.3 Mô hình cây quyết định .....	19
CHƯƠNG 2 PHÂN TÍCH DỮ LIỆU TIỀN XỬ LÝ.....	20
2.1 Dữ liệu về BMI.....	20
2.2 Dữ liệu về người hút thuốc.....	20
2.3 Dữ liệu về người uống rượu .....	21
2.4 Dữ liệu về người bị đột quỵ .....	21
2.5 Dữ liệu về sức khỏe thể chất .....	22
2.6 Dữ liệu về sức khỏe tinh thần.....	22
2.7 Dữ liệu về những người đi bộ gặp khó khăn.....	23
2.8 Dữ liệu về giới tính.....	23
2.9 Dữ liệu về sắc tộc .....	24

2.10 Dữ liệu về người bị tiểu đường .....	24
2.11 Dữ liệu về người có tập thể dục .....	25
2.12 Dữ liệu về người mắc bệnh hen suyễn .....	25
2.13 Dữ liệu về người bị mắc bệnh về thận .....	26
2.14 Dữ liệu về người bị ung thư .....	26
CHƯƠNG 3 TIỀN XỬ LÝ DỮ LIỆU .....	27
3.1 Chuyển đổi BMI .....	27
3.2 Chuyển đổi độ tuổi thành các nhóm tuổi.....	29
3.3 Chuyển đổi thời gian ngủ .....	31
3.4 Chuyển đổi các giá trị dạng chữ thành dạng số. ....	32
CHƯƠNG 4 XÂY DỰNG HỆ THỐNG DỰ ĐOÁN VÀ KẾT QUẢ THỰC NGHIỆM.....	33
4.1 Ngôn ngữ và thư viện sử dụng .....	33
4.2 Cài đặt ngôn ngữ và thư viện cần thiết.....	34
4.2.1 Cài đặt python.....	34
4.2.2 Cài đặt các thư viện cần thiết.....	35
4.3 Chạy chương trình tiền xử lý dữ liệu .....	36
4.4 Triển khai các thuật toán .....	37
4.4.1 Thuật toán hồi quy tuyến tính.....	37
4.4.2 Thuật toán hồi quy logic.....	38
4.4.3 Thuật toán cây quyết định .....	39
4.4.4 So sánh các kết quả và chọn ra thuật toán tốt nhất.....	40
CHƯƠNG 5 TRIỂN KHAI MÔ HÌNH SẢN PHẨM LÊN WEB DEMO .....	41
5.1 Cài đặt môi trường cần thiết.....	41
5.1.1 Cài đặt NodeJS .....	41
5.1.2 Cài đặt Xampp.....	43

5.1.3 Cài đặt mô hình sản phẩm .....	44
5.2 Kết quả cài đặt.....	47
5.2.1 Hình ảnh trang web.....	47
5.2.2 Kết quả thử nghiệm .....	48
KẾT LUẬN .....	49
TÀI LIỆU THAM KHẢO.....	51

## DANH MỤC CÁC HÌNH ẢNH

Hình 1. 1: Supervised learning và Unsupervised learning .....	13
Hình 1. 2: Mô hình học có giám sát – Supervised Learning .....	13
Hình 1. 3: Bài toán Classification .....	15
Hình 1. 4: Mô hình học không giám sát – Unsupervised Learning .....	16
Hình 1. 5: Mô hình học phân nhóm - Clustering .....	17
Hình 1.2.1.1 Biểu diễn mô hình hồi quy tuyến tính .....	18
Hình 1.2.2.1 Biểu diễn mô hình hồi quy logic .....	19
Hình 1.2.3.1 Biểu diễn mô hình cây quyết định.....	19
Hình 2.1.1 Dữ liệu về BMI.....	20
Hình 2.2.1: Dữ liệu về người hút thuốc.....	20
Hình 2.3.1 Dữ liệu về người uống rượu .....	21
Hình 2.4.1 Dữ liệu về người bị đột quỵ .....	21
Hình 2.5.1 Dữ liệu về sức khoẻ thể chất .....	22
Hình 2.6.1 Dữ liệu về sức khoẻ tinh thần.....	22
Hình 2.7.1 Dữ liệu về những người đi bộ gặp khó khăn.....	23
Hình 2.8.1 Dữ liệu về giới tính.....	23
Hình 2.10.1 Dữ liệu về người bị tiểu đường .....	24
Hình 2.11.1 Dữ liệu về người có tập thể dục .....	25
Hình 2.12.1 Dữ liệu về người mắc bệnh hen suyễn .....	25
Hình 2.13.1 Dữ liệu về người bị mắc bệnh về thận .....	26
Hình 2.14.1 Dữ liệu về người bị ung thư da .....	26
Hình 4.2.1.1 Giao diện trang web .....	34
Hình 4.2.1.2 Giao diện trang download .....	34
Hình 4.2.1.3 Giao diện cài đặt.....	35
Hình 4.2.2.1 Giao diện mở hộp thoại Run .....	35
Hình 4.2.2.2 Giao diện terminal .....	36
Hình 4.3.1 Folder chứa code .....	36
Hình 4.3.2 Giao diện terminal .....	36
Hình 4.4.1.1 Độ chính xác của thuật toán .....	37

Hình 4.4.2.1 Độ chính xác của thuật toán .....	38
Hình 4.4.3.1 Độ chính xác của thuật toán .....	39
Hình 5.1.1.1 Giao diện trang web .....	41
Hình 5.1.1.2 Giao diện trang download .....	41
Hình 5.1.1.3 Giao diện cài đặt.....	42
Hình 5.1.2.1 Giao diện trang web .....	43
Hình 5.1.2.2 Giao diện trang download .....	43
Hình 5.1.2.3 Giao diện cài đặt.....	44
Hình 5.1.3.1 Giao diện folder chứa code .....	44
Hình 5.1.3.2 Giao diện terminal .....	45
Hình 5.1.3.3 Giao diện terminal .....	45
Hình 5.1.3.4 Giao diện terminal .....	45
Hình 5.1.3.5 Giao diện terminal .....	46
Hình 5.1.3.6 Giao diện terminal .....	46

## **DANH MỤC BẢNG BIỂU**

Bảng 3.1.1 : bảng phân loại BMI .....	27
Bảng 3.2.1 : bảng chuyển đổi độ tuổi.....	29
Bảng 3.3.1 : bảng chuyển đổi thời gian ngủ.....	31
Bảng 4.1.1 Ngôn ngữ và các thư viện sử dụng. ....	33
Bảng 4.4.4.1 Bảng so sánh độ chính xác thuật toán.....	40

## **DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ**

## MỞ ĐẦU

Bệnh tim mạch (BTM) là một căn bệnh nguy hiểm rất phổ biến trên toàn cầu . Theo WHO - Tổ chức Y tế thế giới , mỗi năm có khoảng 17,5 triệu người tử vong do mắc các bệnh lý về tim và mạch máu . Cứ mỗi 2 giây sẽ có một người chết vì bệnh tim mạch , cứ 5 giây sẽ có 1 người bị nhồi máu cơ tim .

Báo cáo của WHO cũng cho biết tỷ lệ bệnh tim mạch đang ngày càng tăng cao ở các nước đang phát triển trong đó có Việt Nam. Bên cạnh đó , chi phí cho khám và chữa bệnh tim mạch cũng là gánh nặng kinh tế với chi phí hàng trăm tỷ mỗi năm .

Đầu tháng 10 năm 2018, Hội Tim mạch Việt Nam đã tổ chức Hội nghị tim mạch với sự tham gia của hơn 2000 đại biểu trong nước và quốc tế. Trao đổi trong hội nghị, các chuyên gia cho biết Việt Nam hiện có khoảng 25% dân số đang mắc bệnh tim mạch và 46% mắc tăng huyết áp. Hơn nữa, tỷ lệ bệnh tim mạch ở Việt Nam ngày càng trẻ hóa.[1] Vì những dấu hiệu của bệnh tim mạch thường xuất hiện không rõ ràng , thoáng qua làm cho chúng ta thường chủ quan không đề tâm tới, các dấu hiệu chuyển biến nặng thì đã quá muộn. Cùng với đó là gánh nặng về kinh phí vì khi mà bệnh đã chuyển nặng thì cần những phương pháp, thủ thuật phẫu thuật cực kỳ tốn kém.

Bệnh tim mạch nếu được phát hiện ra sớm thì việc điều trị và kiểm soát sẽ rất có hiệu quả và hạn chế được nhiều các biến chứng nguy hiểm ,giảm thiểu nguy cơ tử vong đồng thời giảm gánh nặng bệnh tật cho chính mình,cho người thân trong gia đình nói riêng và toàn xã hội nói chung. Nhiều nghiên cứu cũng đã chứng minh một số nguy cơ từ hành vi ,lối sống thiếu lành mạnh có thể dẫn đến bệnh tim mạch (như không thường xuyên tập luyện thể chất, sử dụng chất kích thích như rượu , thuốc lá,... ăn uống không hợp lý...).

Hiện nay ở nước ta các nghiên cứu về bệnh tim mạch chủ yếu tập trung vào điều trị cho đối tượng bị mắc bệnh. Nghiên cứu về mô hình hồi quy trong việc hỗ trợ chẩn đoán bệnh tim mạch còn chưa được chú trọng. Nhận thức từ ý nghĩa thực tiễn, em quyết định chọn đề tài : ***“Ứng dụng mô hình hồi quy trong hỗ trợ chẩn đoán bệnh tim”***.

Do thời gian hạn chế trong thời gian thực hiện đồ án, đầu tiên đồ án tập trung tìm hiểu,nghiên cứu thuật toán trong học máy và các phương pháp học sâu, song song với việc nghiên cứu thuật toán thực hiện cài đặt mô hình hồi quy về dự báo,chẩn đoán trong học sâu với bộ dữ liệu thực tế được sử dụng làm dữ liệu cho hệ thống dự báo chẩn đoán bệnh tim. Dựa vào mục tiêu cụ thể nêu trên, đồ án được tổ chức nhiều phần với nội dung cụ thể như sau:

**Chương 1:** Tổng quan cơ sở lý thuyết.

**Chương 2:** Phân tích dữ liệu tiền xử lý.

**Chương 3:** Tiền xử lý dữ liệu.

**Chương 4:** Xây dựng hệ thống dự đoán và kết quả thực nghiệm.



**Chương 5:** Triển khai mô hình sản phẩm lên web demo.

**Kết luận:** Cuối cùng, phần kết luận sẽ tổng kết các nội dung đã trình bày trong đề án, từ đó đề xuất các hướng nghiên cứu tiếp theo để cải thiện chất lượng hệ thống.

# CHƯƠNG 1 TỔNG QUAN CƠ SỞ LÝ THUYẾT

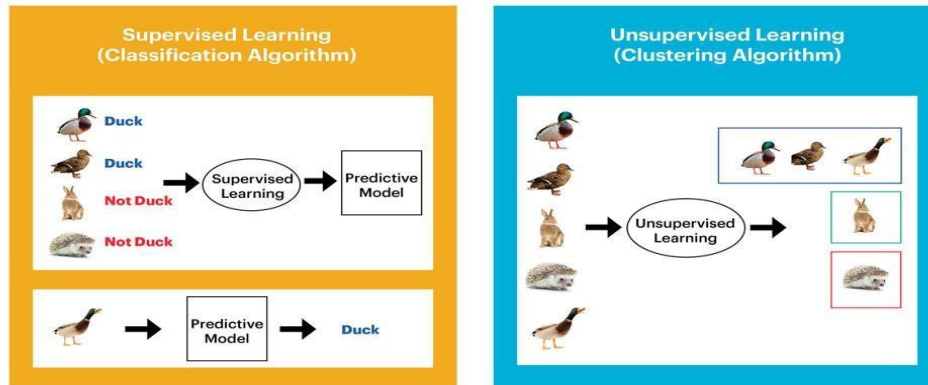
## 1.1 Học máy (Machine learning)

### 1.1.1 Tổng quan về học máy

Machine learning gây nên cơn sốt công nghệ trên toàn thế giới trong vài năm nay. Trong giới học thuật, mỗi năm có hàng ngàn bài báo khoa học về đề tài này. Trong giới công nghiệp, từ các công ty lớn như Google, Facebook, Microsoft đến các công ty khởi nghiệp đều đầu tư vào machine learning. Hàng loạt các ứng dụng sử dụng machine learning ra đời trên mọi lĩnh vực của cuộc sống, từ khoa học máy tính đến những ngành ít liên quan hơn như vật lý, hóa học, y học, chính trị. Xe tự hành của Google và Tesla, hệ thống tự tag khuôn mặt trong ảnh của Facebook, trợ lý ảo Siri của Apple, hệ thống gợi ý sản phẩm của Amazon, hệ thống gợi ý phim của Netflix, máy chơi cờ vây AlphaGo của Google DeepMind, .... Alphago, cỗ máy đánh cờ vây với khả năng tính toán trong một không gian có số lượng phần tử còn nhiều hơn số lượng hạt trong vũ trụ, tối ưu hơn bất kì đại kì thủ nào, là một trong rất nhiều ví dụ hùng hồn cho sự vượt trội của machine learning so với các phương pháp cổ điển.

Machine Learning là một tập con của AI. Theo định nghĩa của Wikipedia, *Machine learning is the subfield of computer science that “gives computers the ability to learn without being explicitly programmed”*. Nói đơn giản, Machine Learning là một lĩnh vực nhỏ của Khoa Học Máy Tính, nó có khả năng tự học hỏi dựa trên dữ liệu đưa vào mà không cần phải được lập trình cụ thể. Với mục tiêu làm cho máy tính có những khả năng nhận thức cơ bản của con người như nghe, nhìn, hiểu được ngôn ngữ, giải toán, lập trình, ... và hỗ trợ con người trong việc xử lý một khối lượng thông tin khổng lồ mà chúng ta phải đối mặt hàng ngày, hay còn gọi là Big Data.

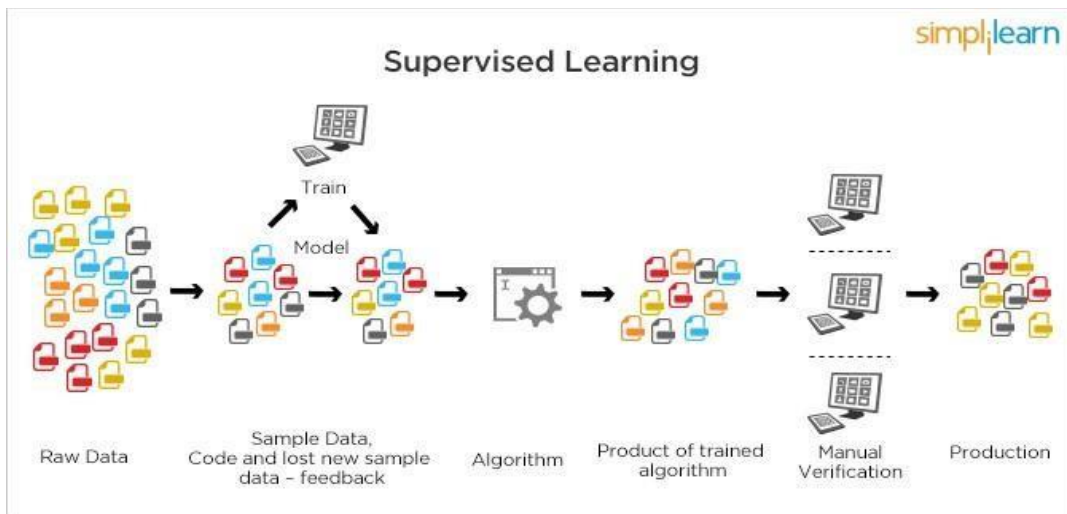
Theo phương thức học, các thuật toán Machine Learning thường được chia làm 2 nhóm chính: Supervised learning, Unsupervised learning.



Hình 1. 1: Supervised learning và Unsupervised learning

### 1.1.2 Học có giám sát (Supervised learning)

Supervised learning là thuật toán dự đoán đầu ra (*outcome*) của một dữ liệu mới (*new input*) dựa trên các cặp (*input, outcome*) đã biết từ trước. Cặp dữ liệu này còn được gọi là (*data, label*), tức (*dữ liệu, nhãn*). Supervised learning là nhóm phổ biến nhất trong các thuật toán Machine Learning.



Hình 1. 2: Mô hình học có giám sát – Supervised Learning

Một cách toán học, Supervised learning là khi chúng ta có một tập hợp biến đầu vào  $X = \{x_1, x_2, \dots, x_N\}$  và một tập hợp nhãn tương ứng  $Y = \{y_1, y_2, \dots, y_N\}$ , trong đó  $x_i, y_i$  là các vector. Các cặp dữ liệu biết trước  $(x_i, y_i) \in X \times Y$  được gọi là tập *training data* (dữ liệu huấn luyện). Từ tập training data này, chúng ta cần tạo ra một hàm số ánh xạ mỗi phần tử từ tập  $X$  sang một phần tử (xấp xỉ) tương ứng của tập  $Y$ :

$$y_i \approx f(x_i), \forall i = 1, 2, \dots, N$$

Mục đích là xấp xỉ hàm số  $f$  thật tốt để khi có một dữ liệu  $x$  mới, chúng ta có thể tính được nhãn tương ứng của nó  $y = f(x)$ .

Thuật toán supervised learning còn được tiếp tục chia nhỏ ra thành hai loại chính:

### 1.1.2.1 Classification (Phân lớp)

Bài toán phân lớp là bài toán xếp đối tượng, dữ liệu vào một trong các lớp đã được xác định sẵn. Một bài toán được gọi là *classification* nếu các *label* của *input data* được chia thành một số hữu hạn nhóm. Ví dụ: Gmail xác định xem một email có phải là spam hay không; các hãng tín dụng xác định xem một khách hàng có khả năng thanh toán nợ hay không. Ba ví dụ phía trên được chia vào loại này.

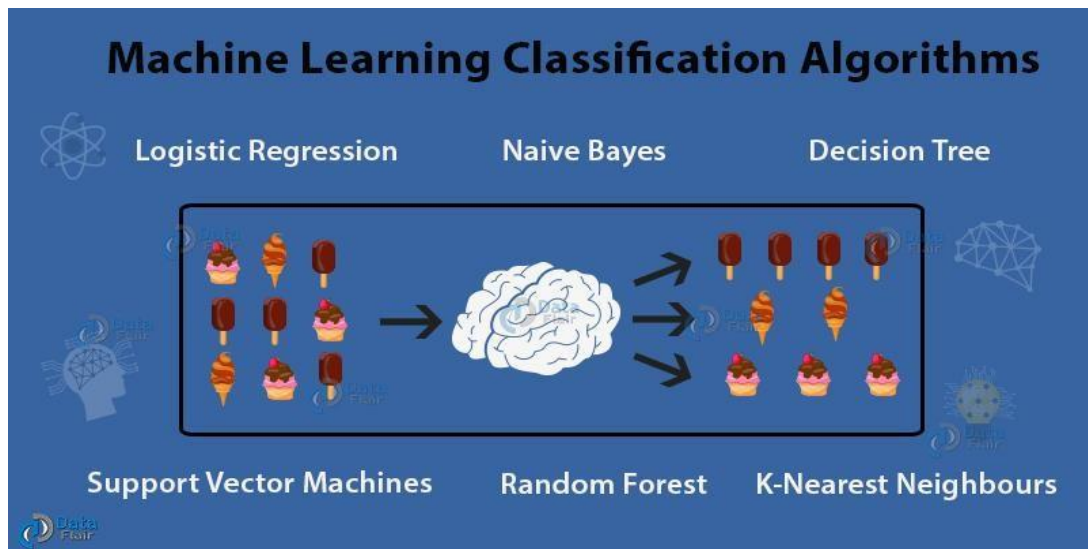
Phân lớp được thực hiện thông qua 2 bước:

Bước 1: Xây dựng mô hình

- Mô tả tập các lớp được xác định trước: Tập học/Tập huấn luyện gồm các mẫu dành cho xây dựng mô hình. Mỗi mẫu thuộc về một lớp đã định nghĩa trước.
- Tìm luật phân lớp, cây quyết định, hoặc công thức mô tả lớp.

Bước 2: Vận hành mô hình phân lớp các đối tượng chưa biết

- Xác định độ chính xác của mô hình, sử dụng tập dữ liệu kiểm tra độc lập.
- Nếu độ chính xác ở mức chấp nhận được thì áp dụng mô hình để phân lớp các mẫu chưa xác định được nhãn lớp.



Hình 1. 3: Bài toán Classification

### 1.1.2.2 Regression (Hồi quy)

Theo R. D. Snee [1]: “Hồi quy là kỹ thuật thống kê trong lĩnh vực phân tích dữ liệu và xây dựng các mô hình từ thực nghiệm, cho phép mô hình hồi quy vừa được khám phá được dùng cho mục đích dự báo (Prediction), điều khiển (Control) hay học (Learn) cơ chế đã tạo ra dữ liệu.

Mô hình hồi quy (Regression Model) là mô hình mô tả mối liên kết (relationship) giữa một tập các biến dự báo/độc lập (predictor/independent variables) và một hay nhiều biến đáp ứng/phụ thuộc (responses/ dependent variables).

Bài toán hồi quy được phân loại như sau:

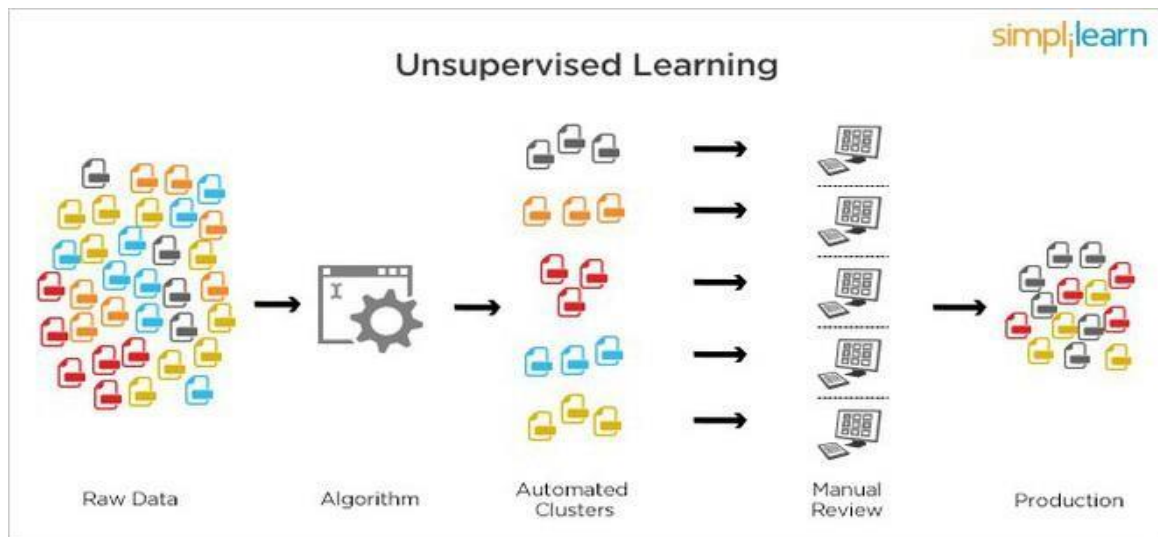
- Hồi quy tuyến tính (linear) và phi tuyến (nonlinear).
- Hồi quy đơn biến (single) và đa biến (multiple).
- Hồi quy có thông số (parametric), phi thông số (nonparametric) và thông số kết hợp (semiparametric).
- Hồi quy đối xứng (symmetric) và bất đối xứng (asymmetric).

Một bài toán được gọi là bài toán hồi quy (Regression) nếu label không được chia thành các nhóm mà là một giá trị thực cụ thể. Ví dụ: một căn nhà rộng  $x$  m<sup>2</sup>, có  $y$  phòng ngủ và cách trung tâm thành phố  $z$  km sẽ có giá là bao nhiêu?

Gần đây Microsoft có một ứng dụng dự đoán giới tính và tuổi dựa trên khuôn mặt. Phần dự đoán giới tính có thể coi là thuật toán Classification, phần dự đoán tuổi có thể coi là thuật toán Regression. Chú ý rằng phần dự đoán tuổi cũng có thể được coi là Classification nếu ta coi tuổi là một số nguyên dương không lớn hơn 150, chúng ta sẽ có 150 class (lớp) khác nhau.

### 1.1.3 Học không giám sát (Unsupervised learning)

Trong thuật toán này, chúng ta không biết được *outcome* (hay *nhãn*) mà chỉ có dữ liệu đầu vào. Thuật toán unsupervised learning sẽ dựa vào cấu trúc của dữ liệu để thực hiện một công việc nào đó, ví dụ như phân nhóm (clustering) hoặc giảm số chiều của dữ liệu (dimension reduction) để thuận tiện trong việc lưu trữ và tính toán.



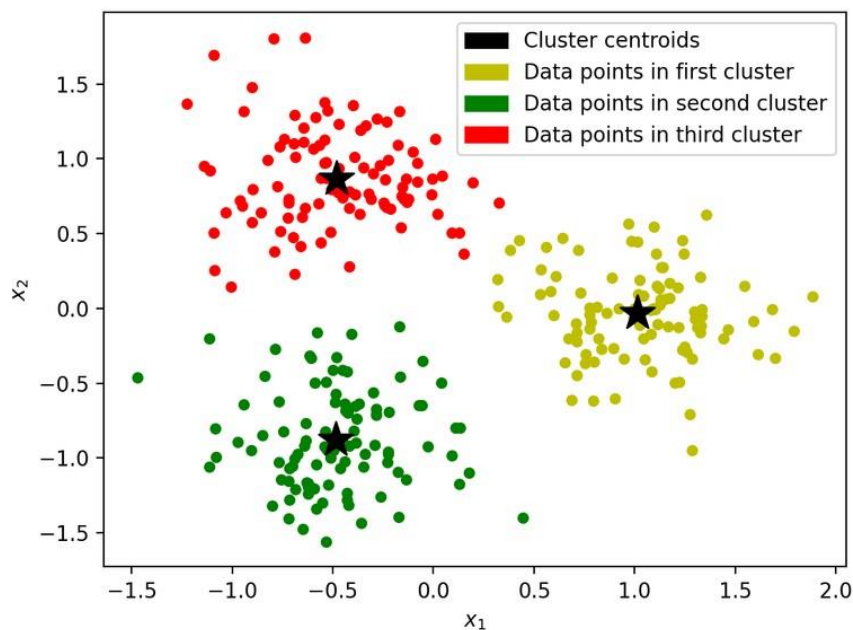
Hình 1. 4: Mô hình học không giám sát – Unsupervised Learning

Một cách toán học, Unsupervised learning là khi chúng ta chỉ có dữ liệu vào  $X$  mà không biết *nhãn*  $Y$  tương ứng. Những thuật toán loại này được gọi là Unsupervised learning vì không giống như Supervised learning, chúng ta không biết câu trả lời chính xác cho mỗi dữ liệu đầu vào. Giống như khi ta học, không có thầy cô giáo nào chỉ cho ta biết đó là chữ A hay chữ B. Cụm *không giám sát* được đặt tên theo nghĩa này. Các bài toán Unsupervised learning được tiếp tục chia nhỏ thành hai loại:

- Clustering (Phân cụm)
- Association

### 1.1.3.1 Clustering (phân cụm)

Một bài toán phân nhóm toàn bộ dữ liệu  $X$  thành các nhóm nhỏ dựa trên sự liên quan giữa các dữ liệu trong mỗi nhóm. Ví dụ: xác định và phân loại các nhóm khách hàng (khách hàng tiềm năng, khách hàng giá trị, ...) dựa trên thói quen sử dụng để dự đoán nhu cầu sử dụng sản phẩm hay dịch vụ của công ty để giúp công ty có chiến lược kinh doanh hiệu quả hơn. Điều này cũng giống như việc ta đưa cho một đứa trẻ rất nhiều mảnh ghép với các hình thù và màu sắc khác nhau, ví dụ tam giác, vuông, tròn với màu xanh và đỏ, sau đó yêu cầu trẻ phân chúng thành từng nhóm. Mặc dù không cho trẻ biết mảnh nào tương ứng với hình nào hoặc màu nào, nhiều khả năng chúng vẫn có thể phân loại các mảnh ghép theo màu hoặc hình dạng.



Hình 1. 5: Mô hình học phân nhóm - Clustering

### 1.1.3.2 Association

Là bài toán khi chúng ta muốn khám phá ra một quy luật dựa trên nhiều dữ liệu cho trước. Ví dụ: những khách hàng mua máy tính laptop thường có xu hướng mua thêm chuột hoặc tai nghe; những khán giả xem phim Iron Man thường có xu hướng xem thêm phim Iron Man 2, Iron Man 3,... dựa vào đó tạo ra một hệ thống gợi ý khách hàng (Recommendation System), thúc đẩy nhu cầu mua sắm.

## 1.2 Tổng quan lý thuyết về thuật toán áp dụng

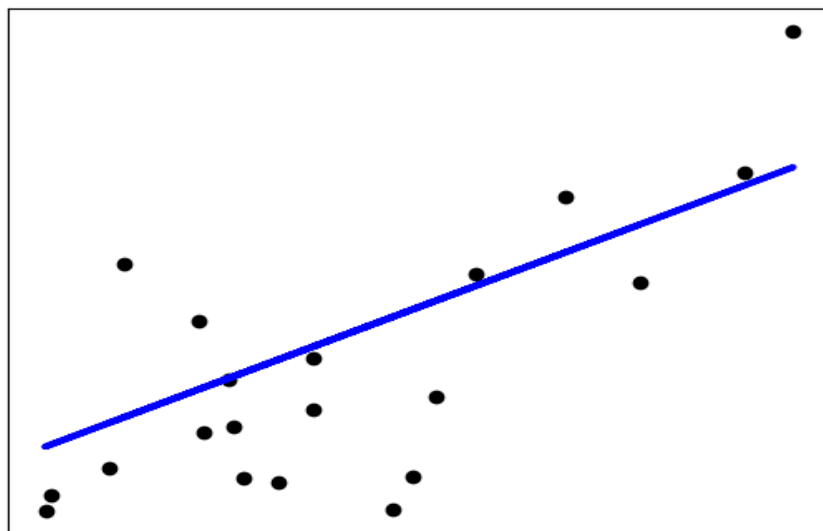
### 1.2.1 Mô hình hồi quy tuyến tính

"Hồi quy tuyến tính" là một phương pháp thống kê để hồi quy dữ liệu với biến phụ thuộc có giá trị liên tục trong khi các biến độc lập có thể có một trong hai giá trị liên tục hoặc là giá trị phân loại. Nói cách khác "Hồi quy tuyến tính" là một phương pháp để dự đoán biến phụ thuộc (Y) dựa trên giá trị của biến độc lập (X). Nó có thể được sử dụng cho các trường hợp chúng ta muốn dự đoán một số lượng liên tục. Ví dụ, dự đoán giao thông ở một cửa hàng bán lẻ, dự đoán thời gian người dùng dừng lại một trang nào đó hoặc số trang đã truy cập vào một website nào đó v.v...

Mô hình Logistic Regression có dạng

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$$

Mô tả cho mô hình hồi quy tuyến tính trong hình dưới đây:



Hình 1.2.1.1 Biểu diễn mô hình hồi quy tuyến tính

### 1.2.2 Mô hình hồi quy logic

Mô hình có tên tiếng Anh là Logistic Regression:

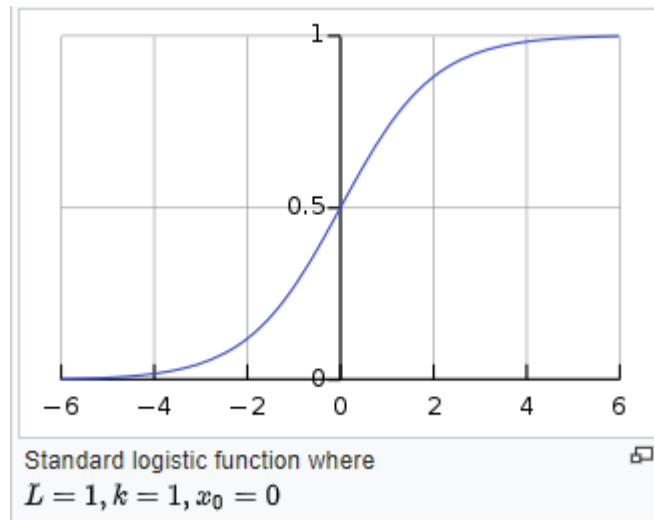
Đầu ra có thể được thể hiện dưới dạng xác suất (probability). Ví dụ: xác suất thi đỗ nếu biết thời gian ôn thi, xác suất ngày mai có mưa dựa trên những thông tin đo được trong ngày hôm nay,...

Mô hình Logistic Regression có dạng

$$f(\mathbf{x}) = \theta(\mathbf{w}^T \mathbf{x})$$

Mô tả cho mô hình hồi quy logic trong hình dưới đây:





Hình 1.2.2.1 Biểu diễn mô hình hồi quy logic

### 1.2.3 Mô hình cây quyết định

Decision tree là một mô hình supervised learning, có thể được áp dụng vào cả hai bài toán classification và regression. Việc xây dựng một decision tree trên dữ liệu huấn luyện cho trước là việc đi xác định các câu hỏi và thứ tự của chúng. Một điểm đáng lưu ý của decision tree là:

- có thể làm việc với các đặc trưng dạng categorical, thường là rời rạc và không có thứ tự. Ví dụ, mưa, nắng hay xanh, đỏ, v.v.
- làm việc với dữ liệu có vector đặc trưng bao gồm cả thuộc tính dạng categorical và liên tục (numeric).
- ít yêu cầu việc chuẩn hoá dữ liệu.

Decision tree trained on all the iris features

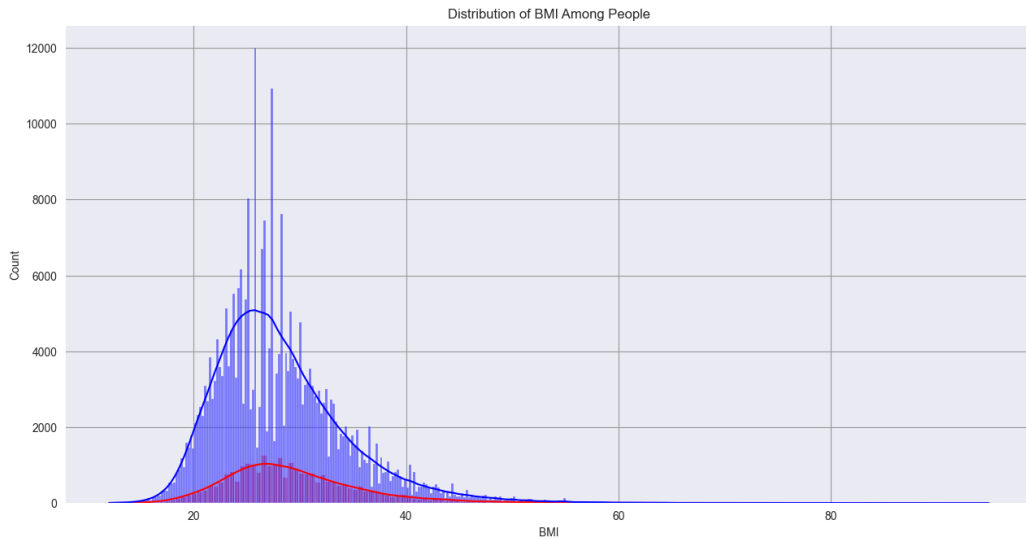


Hình 1.2.3.1 Biểu diễn mô hình cây quyết định

## CHƯƠNG 2 PHÂN TÍCH DỮ LIỆU TIỀN XỬ LÝ.

### 2.1 Dữ liệu về BMI

Những người bị bệnh tim thường có chỉ số BMI cao hơn mức trung bình.



Hình 2.1.1 Dữ liệu về BMI

### 2.2 Dữ liệu về người hút thuốc

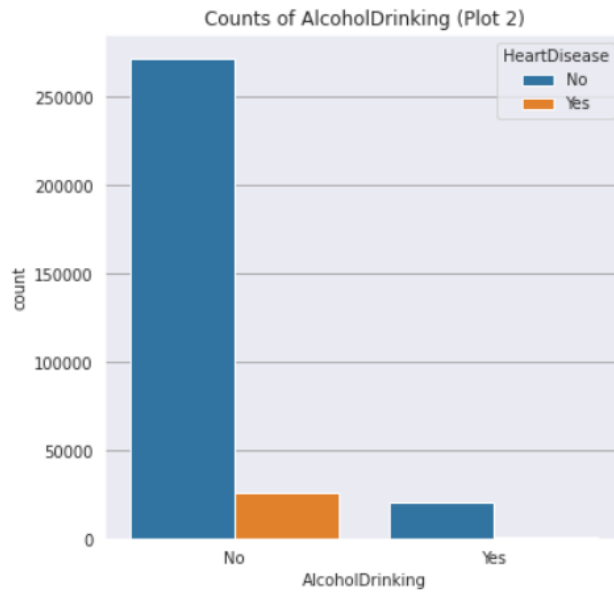


Hình 2.2.1: Dữ liệu về người hút thuốc

Có 131908 người có hút thuốc : trong đó có 16037 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 12,16% .

Có 187887 người không hút thuốc : trong đó có 11336 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 6,03% .

## 2.3 Dữ liệu về người uống rượu

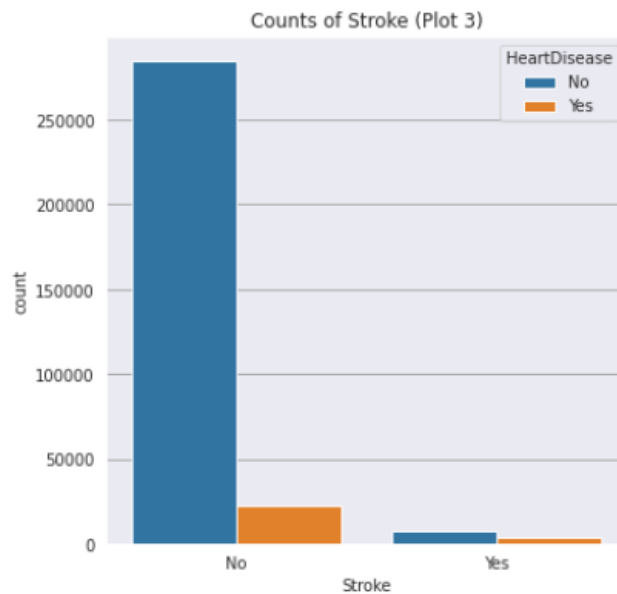


Hình 2.3.1 Dữ liệu về người uống rượu

Có 21777 người có uống rượu: trong đó có 1141 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 5,24% .

Có 298018 người không uống rượu : trong đó có 26232 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 8,8% .

## 2.4 Dữ liệu về người bị đột quỵ



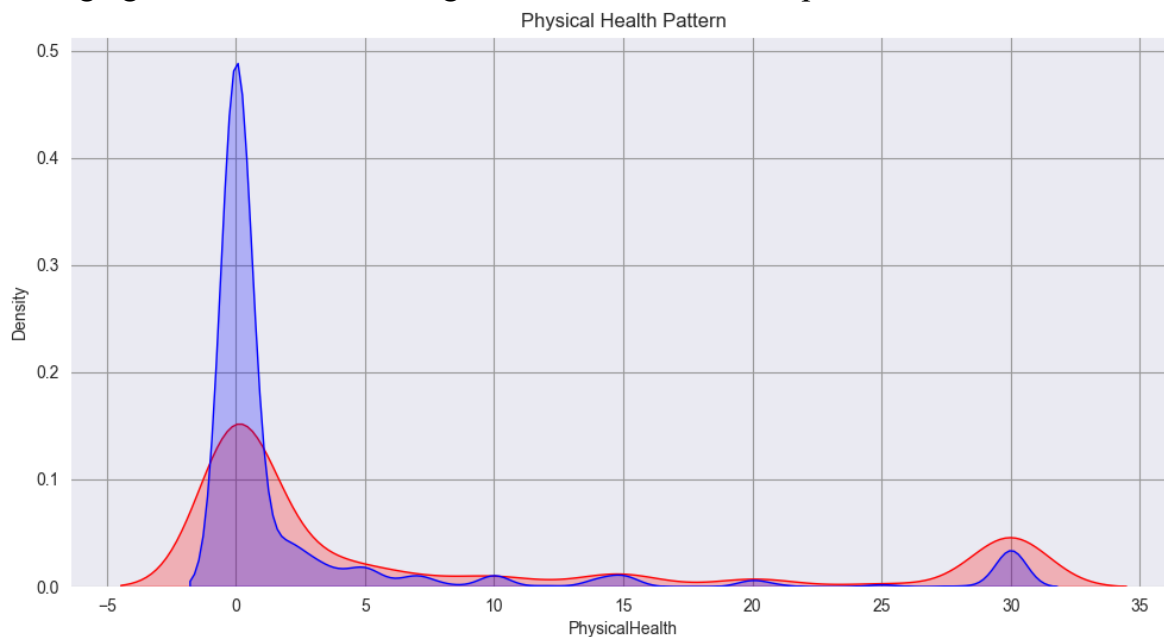
Hình 2.4.1 Dữ liệu về người bị đột quỵ

Có 12069 người bị đột quỵ: trong đó có 4389 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 36,37% .

Có 307726 người không bị đột quỵ : trong đó có 22984 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 7,47% .

## 2.5 Dữ liệu về sức khoẻ thể chất

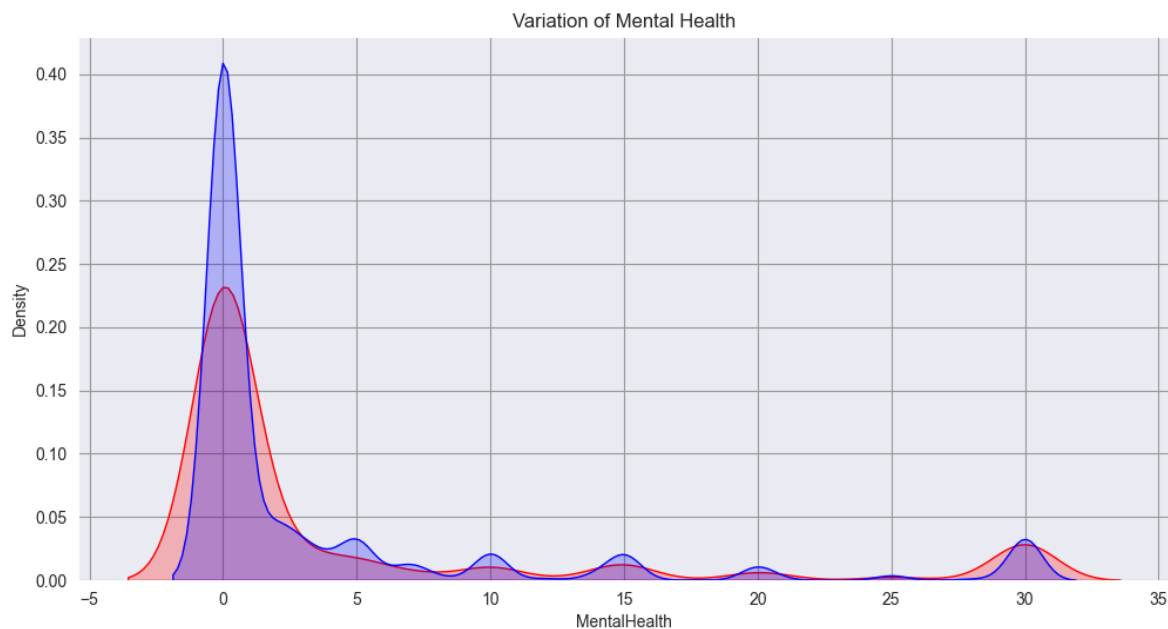
Những người bị bệnh tim thường có sức khoẻ thể chất thấp hơn.



Hình 2.5.1 Dữ liệu về sức khoẻ thể chất

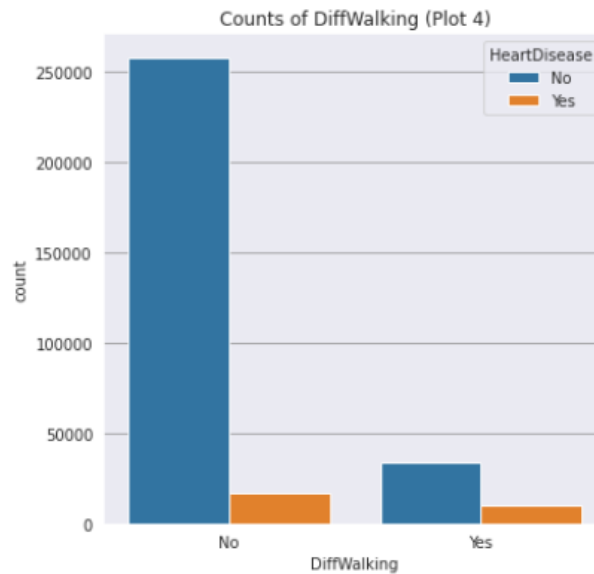
## 2.6 Dữ liệu về sức khoẻ tinh thần

Những người bị bệnh tim thường có sức khoẻ tinh thần thấp hơn.



Hình 2.6.1 Dữ liệu về sức khoẻ tinh thần

## 2.7 Dữ liệu về những người đi bộ gặp khó khăn

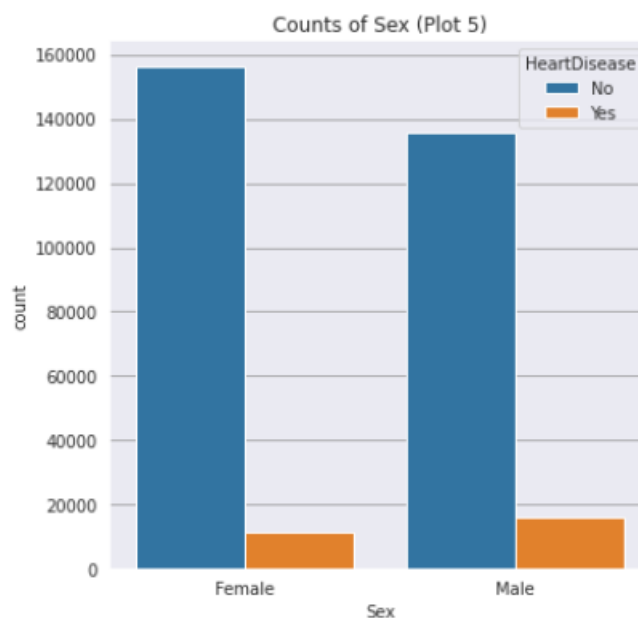


Hình 2.7.1 Dữ liệu về những người đi bộ gặp khó khăn

Có 44410 người đi bộ gặp khó khăn: trong đó có 10028 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 22,58 % .

Có 275385 người đi bộ không gặp khó khăn: trong đó có 17345 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 6,3 % .

## 2.8 Dữ liệu về giới tính



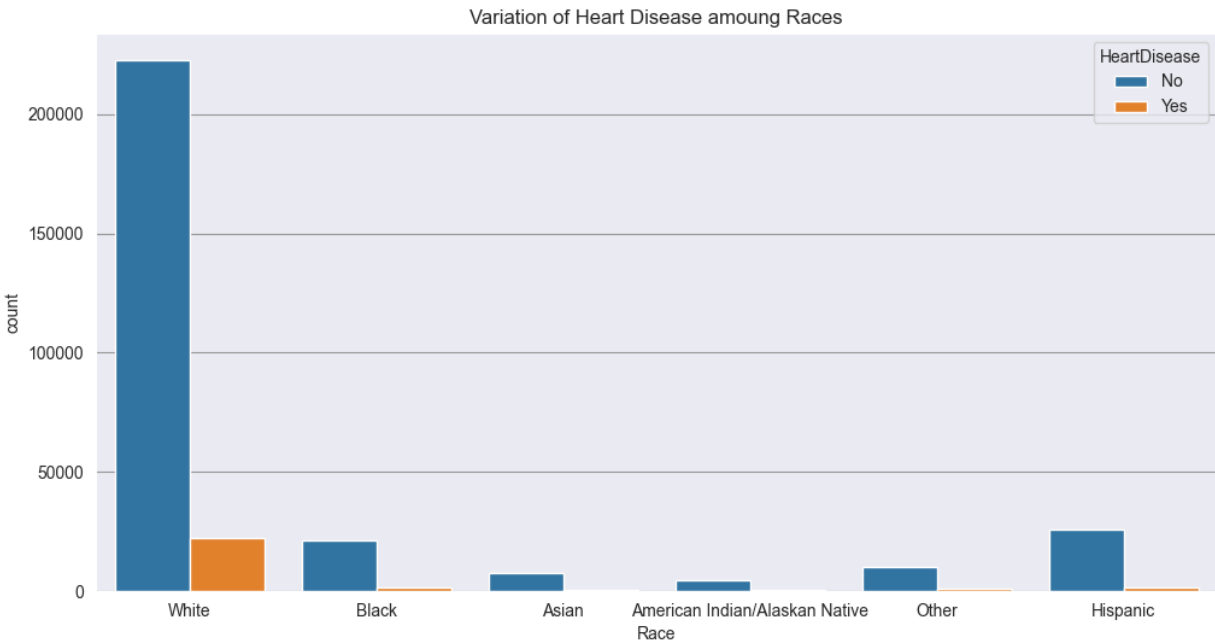
Hình 2.8.1 Dữ liệu về giới tính

Có 167805 người giới tính nữ: trong đó có 11234 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 6,7% .

Có 151990 người giới tính nam: trong đó có 16139 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 10,62% .

## 2.9 Dữ liệu về sắc tộc

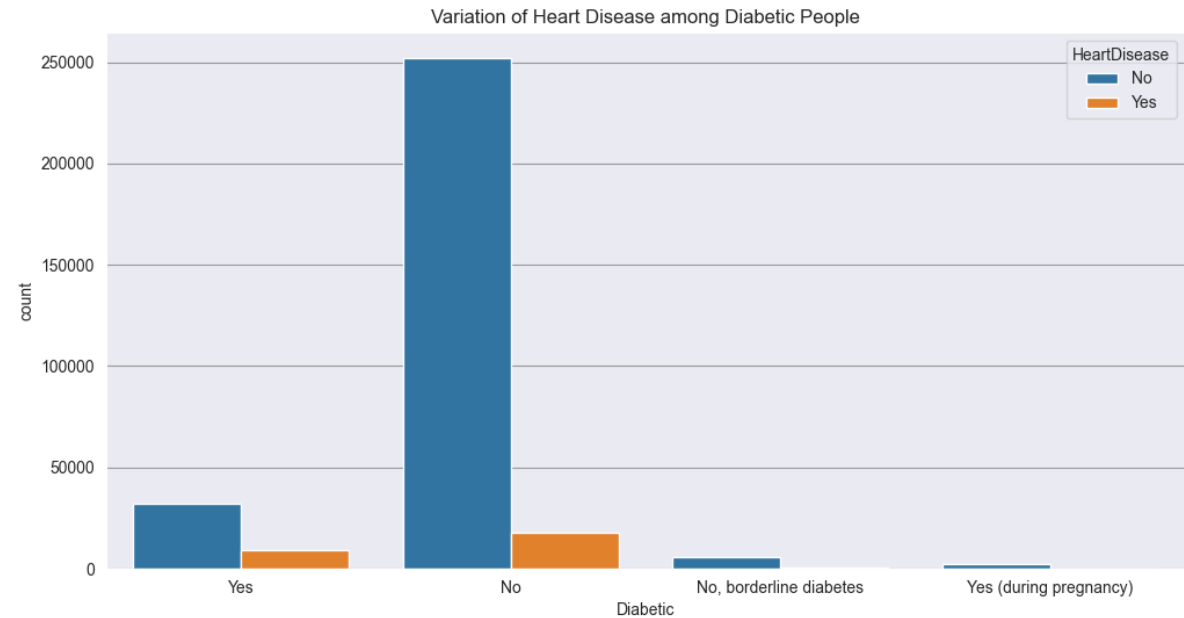
Người da trắng có tỉ lệ mắc bệnh tim cao nhất.



Hình 2.9.1 Dữ liệu về sắc tộc

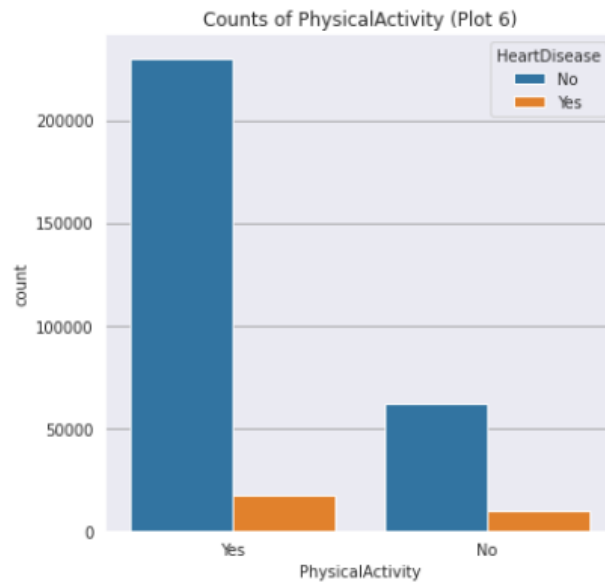
## 2.10 Dữ liệu về người bị tiểu đường

Những người không mắc bệnh tiểu đường có tỉ lệ mắc bệnh tim cao hơn.



Hình 2.10.1 Dữ liệu về người bị tiểu đường

## 2.11 Dữ liệu về người có tập thể dục

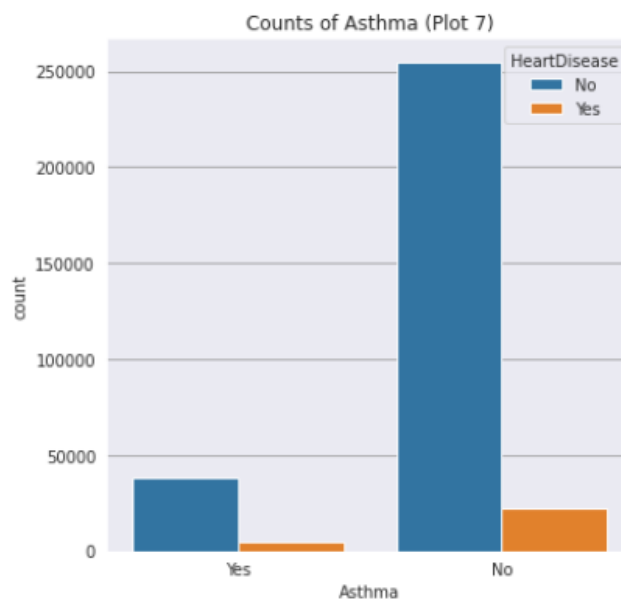


Hình 2.11.1 Dữ liệu về người có tập thể dục

Có 247975 người có tập thể dục: trong đó có 17489 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 7,05 % .

Có 71838 người không tập thể dục: trong đó có 9884 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 13,76 % .

## 2.12 Dữ liệu về người mắc bệnh hen suyễn

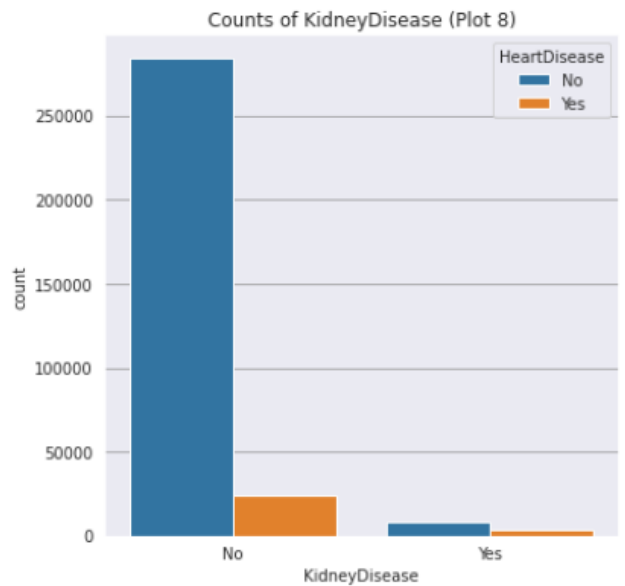


Hình 2.12.1 Dữ liệu về người mắc bệnh hen suyễn

Có 42872 người mắc hen suyễn: trong đó có 4933 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 11,51 % .

Có 276923 người không mắc hen suyễn: trong đó có 22440 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 8,1 % .

## 2.13 Dữ liệu về người bị mắc bệnh về thận

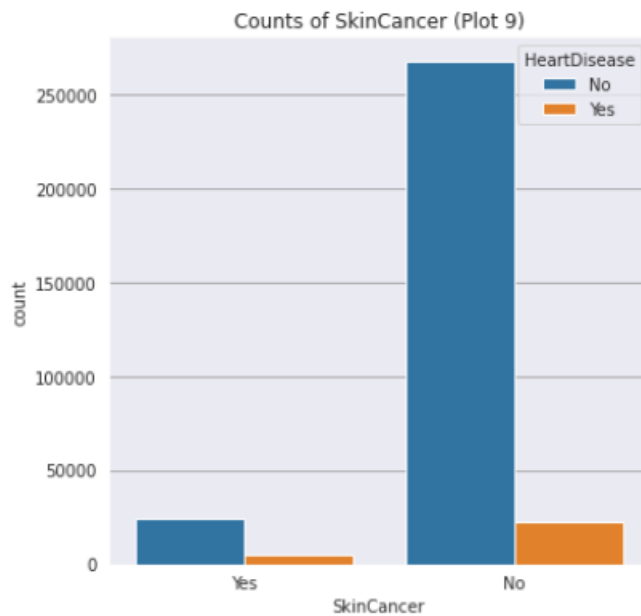


Hình 2.13.1 Dữ liệu về người bị mắc bệnh về thận

Có 11779 người mắc bệnh về thận: trong đó có 3455 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 29,33 % .

Có 308016 người không mắc bệnh về thận: trong đó có 23918 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 7,77 % .

## 2.14 Dữ liệu về người bị ung thư da



Hình 2.14.1 Dữ liệu về người bị ung thư da

Có 29819 người mắc ung thư da: trong đó có 4980 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 16,7 % .

Có 289976 người không mắc bệnh về thận: trong đó có 22393 người mắc bệnh tim chiếm tỉ lệ xấp xỉ 7,72 % .



## CHƯƠNG 3 TIỀN XỬ LÝ DỮ LIỆU

### 3.1 Chuyển đổi BMI

Tổ chức Y tế thế giới khuyến dùng “chỉ số khối cơ thể” (Body Mass Index – BMI), WHO 1995) để đánh giá tình trạng dinh dưỡng của người trưởng thành. Theo WHO thì tình trạng dinh dưỡng ở người trưởng thành được đánh giá là “Bình thường” khi BMI trong ngưỡng 18,50-24,99; “Gầy” khi chỉ số BMI <18,50; “Thừa cân” khi BMI >25,0; “Béo phì” khi BMI >30,0.

Thừa cân là tình trạng vượt quá cân nặng nên có so với chiều cao còn béo phì là tình trạng tích lũy mỡ thái quá không bình thường một cách cục bộ hay toàn thể của lipid trong các tổ chức mỡ tới mức ảnh hưởng xấu đến sức khỏe. Có nhiều chỉ số có thể dùng để đánh giá tình trạng thừa cân – béo phì. Trên cộng đồng, để đánh giá mức độ thừa cân – béo phì, người ta thường dùng chỉ số khối cơ thể  $BMI = W \text{ (kg)} / (H^2 \text{ (m)})$  và dựa vào bảng phân loại sau[4] :

Bảng 3.1.1 : bảng phân loại BMI

Tình trạng dinh dưỡng	Chỉ số BMI	Định nghĩa dữ liệu
Gầy độ 3	<16,00	1/8
Gầy độ 2	16,00 – 16,99	2/8
Gầy độ 1	17,00 – 18,49	3/8
Bình thường	18,50 – 24,99	4/8
Tiền béo phì	25.00 – 29.99	5/8
Béo phì độ 1	30.00 – 34.99	6/8
Béo phì độ 2	35.00 – 39.99	7/8
Béo phì độ 3	$\geq 40.00$	8/8

```
for i in range(len(data)):
    if data.iloc[i,1] < 16: # gầy độ 3
        data.iloc[i,1] = 1/8
    elif data.iloc[i,1] >= 16 and data.iloc[i,1] < 17: # gầy độ 2
        data.iloc[i,1] = 2/8
    elif data.iloc[i,1] >= 17 and data.iloc[i,1] < 18.5: # gầy độ 1
        data.iloc[i,1] = 3/8
    elif data.iloc[i,1] >= 18.5 and data.iloc[i,1] < 25: # bình thường
        data.iloc[i,1] = 4/8
    elif data.iloc[i,1] >= 25 and data.iloc[i,1] < 30: # tiền béo phì
        data.iloc[i,1] = 5/8
    elif data.iloc[i,1] >= 30 and data.iloc[i,1] < 35: # béo phì độ 1
        data.iloc[i,1] = 6/8
    elif data.iloc[i,1] >= 35 and data.iloc[i,1] < 40: # béo phì độ 2
        data.iloc[i,1] = 7/8
    elif data.iloc[i,1] >= 40: # béo phì độ 3
        data.iloc[i,1] = 8/8
```

### 3.2 Chuyển đổi độ tuổi thành các nhóm tuổi

Độ tuổi từ 18 đến dưới 40 tuổi được chuyển đổi thành nhóm thanh niên ở độ tuổi này :  
Là một thời kỳ quan trọng để thành lập bản sắc và tính độc lập ,sức khỏe phát triển đến độ ổn định cao .

Độ tuổi từ 40 đến dưới 60 tuổi được chuyển đổi thành nhóm trung niên ở độ tuổi này :  
Sức khỏe bắt đầu suy giảm và các dấu hiệu của tuổi già . Những khủng hoảng tuổi trung niên: về thể chất , về gia đình, nghề nghiệp không đạt đến mức độ phát triển mong muốn...

Độ tuổi từ 60 trở đi được chuyển đổi thành nhóm người cao tuổi ở độ tuổi này :  
Sự suy giảm của sức khỏe tăng dần , là thời kỳ có nhiều mất mát(có ảnh hưởng tới sức khỏe tinh thần).[5]

Bảng 3.2.1 : bảng chuyển đổi độ tuổi

Độ tuổi trong file dữ liệu	Nhóm tuổi chuyển đổi	Chuyển đổi dữ liệu
18 - 24	Thanh niên	1/3
25 - 29		
30 - 34		
35 - 39		
40 - 44	Trung niên	2/3
45 - 49		
50 - 54		
55 - 59		
60 - 64	Người cao tuổi	3/3
65 - 69		
70 - 74		
75 - 79		
80 - older		

```
data.iloc[:,9].replace("18-24",1/3,inplace=True)# thanh niên
data.iloc[:,9].replace("25-29",1/3,inplace=True)# thanh niên
data.iloc[:,9].replace("30-34",1/3,inplace=True)# thanh niên
data.iloc[:,9].replace("35-39",1/3,inplace=True)# thanh niên
data.iloc[:,9].replace("40-44",2/3,inplace=True)# trung niên
data.iloc[:,9].replace("45-49",2/3,inplace=True)# trung niên
data.iloc[:,9].replace("50-54",2/3,inplace=True)# trung niên
data.iloc[:,9].replace("55-59",2/3,inplace=True)# trung niên
data.iloc[:,9].replace("60-64",3/3,inplace=True)# người già
data.iloc[:,9].replace("65-69",3/3,inplace=True)# người già
data.iloc[:,9].replace("70-74",3/3,inplace=True)# người già
data.iloc[:,9].replace("75-79",3/3,inplace=True)# người già
data.iloc[:,9].replace("80 or older",3/3,inplace=True)# người già
```

### 3.3 Chuyển đổi thời gian ngủ

Thời lượng giấc ngủ của một người phụ thuộc vào nhiều yếu tố, bao gồm cả độ tuổi theo đó :

Từ 7 đến 9 giờ là khoảng thời gian ngủ cần thiết mà hầu hết người trưởng thành cần, mặc dù một số người có thể cần ngủ ít hơn 6 giờ hoặc lên đến 10 giờ mỗi ngày.

Người cao tuổi (từ 65 tuổi trở lên) cần ngủ 7 - 8 giờ mỗi ngày.[6]

Bảng 3.3.1 : bảng chuyển đổi thời gian ngủ

Độ tuổi	Thời gian ngủ	Chuyển đổi	Chuyển đổi dữ liệu
Tất cả độ tuổi	< 7h	Thiếu ngủ	0
18 – 64 tuổi	$7h \leq \text{Time} < 9h$	Ngủ đủ	0.5
	$\geq 9h$	Ngủ nhiều	1
65 tuổi trở lên	$7h \leq \text{Time} < 8h$	Ngủ đủ	0.5
	$\geq 8h$	Ngủ nhiều	1

```
for i in range(len(data)):
    if data.iloc[i,9] in ["18-24","25-29","30-34","35-39","40-44","45-49","50-54","55-59","60-64"]:
        if data.iloc[i,14] < 7:
            data.iloc[i,14] = 0 # ngủ thiếu
        elif data.iloc[i,14] == 7 or data.iloc[i,14] == 8:
            data.iloc[i,14] = 0.5 # ngủ đủ
        elif data.iloc[i,14] > 8:
            data.iloc[i,14] = 1 # ngủ quá
    elif data.iloc[i,9] in ["65-69","70-74","75-79","80 or older"]:
        if data.iloc[i,14] < 7:
            data.iloc[i,14] = 0 # ngủ thiếu
        elif data.iloc[i,14] == 7 or data.iloc[i,14] == 8:
            data.iloc[i,14] = 0.5 # ngủ đủ
        elif data.iloc[i,14] > 8:
            data.iloc[i,14] = 1 # ngủ quá
```

### 3.4 Chuyển đổi các giá trị dạng chữ thành dạng số.

Do các thuật toán chỉ hoạt động trên dữ liệu dạng số vì vậy cần chuyển đổi tất cả các dữ liệu dạng chữ về dạng số .

```
# chuyển đổi giới tính sang số
data.iloc[:,8].replace("Female",1,inplace=True)
data.iloc[:,8].replace("Male",0,inplace=True)
```

```
# chuyển đổi nhóm sắc tộc sang số
data.iloc[:,10].replace("White",1,inplace=True)
data.iloc[:,10].replace("Black",4/5,inplace=True)
data.iloc[:,10].replace("Asian",3/5,inplace=True)
data.iloc[:,10].replace("American Indian/Alaskan Native",2/5,inplace=True)
data.iloc[:,10].replace("Other",1/5,inplace=True)
data.iloc[:,10].replace("Hispanic",0,inplace=True)
```

```
# chuyển đổi bệnh tiểu đường sang số
data.iloc[:,11].replace("Yes",1,inplace=True)
data.iloc[:,11].replace("Yes (during pregnancy)",2/3,inplace=True)
data.iloc[:,11].replace("No, borderline diabetes",1/3,inplace=True)
data.iloc[:,11].replace("No",0,inplace=True)
```

```
# Categorize the Health of the person into integers values
data.iloc[:,13].replace("Excellent",1,inplace=True)
data.iloc[:,13].replace("Very good",3/4,inplace=True)
data.iloc[:,13].replace("Good",2/4,inplace=True)
data.iloc[:,13].replace("Fair",1/4,inplace=True)
data.iloc[:,13].replace("Poor",0,inplace=True)
```

```
#chuyển các giá trị y/n thành số
data.replace("Yes",1,inplace=True)
data.replace("No",0,inplace=True)
```

```
# chuyển về vùng xử lí 0 - 1
for i in range(len(data)):
    data.iloc[i,5] = data.iloc[i,5]/100
    data.iloc[i,6] = data.iloc[i,6]/100
```

## CHƯƠNG 4 XÂY DỰNG HỆ THỐNG DỰ ĐOÁN VÀ KẾT QUẢ THỰC NGHIỆM

### 4.1 Ngôn ngữ và thư viện sử dụng

Python là một ngôn ngữ lập trình thông dịch (interpreted), hướng đối tượng (object-oriented), và là một ngôn ngữ bậc cao (high-level) ngữ nghĩa động (dynamic semantics). Python hỗ trợ các module và gói (packages), khuyến khích chương trình module hóa và tái sử dụng mã. Trình thông dịch Python và thư viện chuẩn mở rộng có sẵn dưới dạng mã nguồn hoặc dạng nhị phân miễn phí cho tất cả các nền tảng chính và có thể được phân phối tự do. Python có các ưu điểm dễ dàng kết nối với các thành phần khác, chạy trên nhiều nền tảng, là ngôn ngữ mã nguồn mở, ...[7]

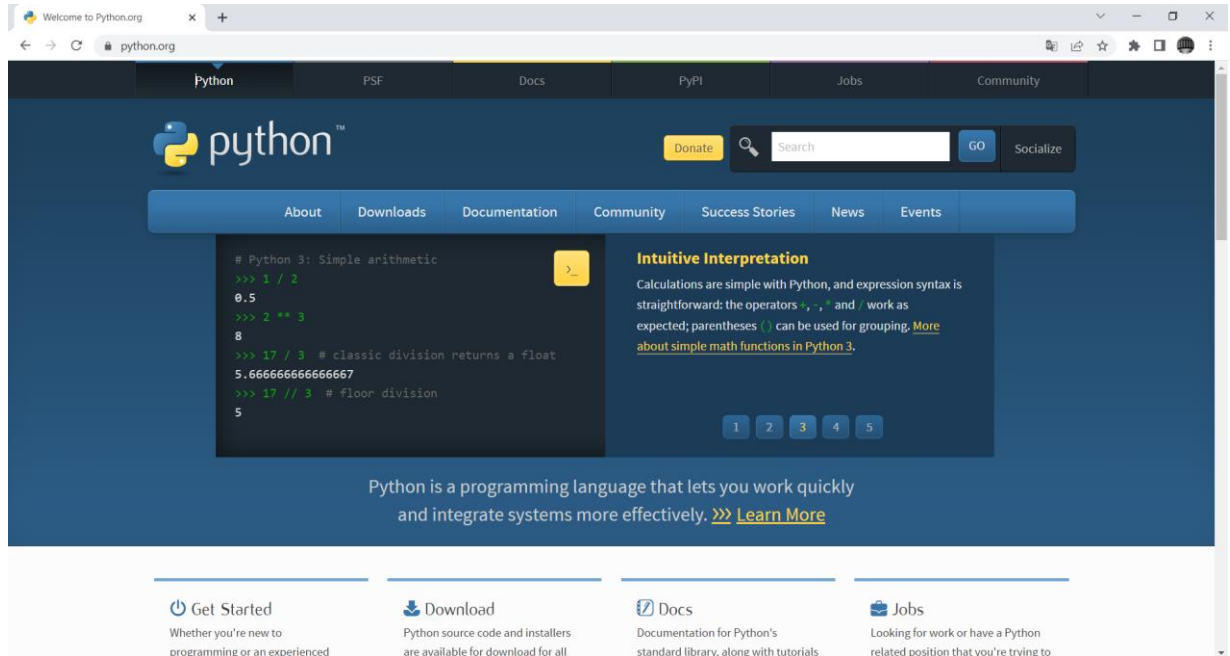
Bảng 4.1.1 Ngôn ngữ và các thư viện sử dụng.

Python 3.6	Matplotlib
	Pandas
	Seaborn
	Numpy
	Sklearn
	Request
	Json
	Urllib

## 4.2 Cài đặt ngôn ngữ và thư viện cần thiết

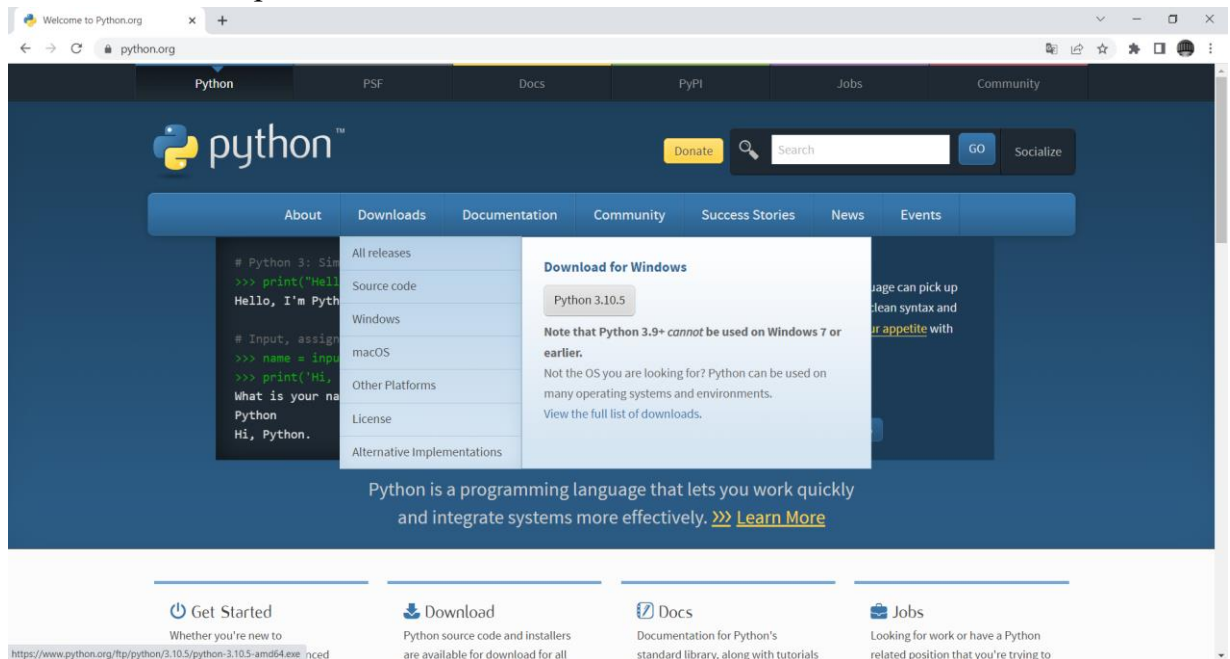
### 4.2.1 Cài đặt python

Bước 1 : Vào trang web <https://www.python.org/>



Hình 4.2.1.1 Giao diện trang web

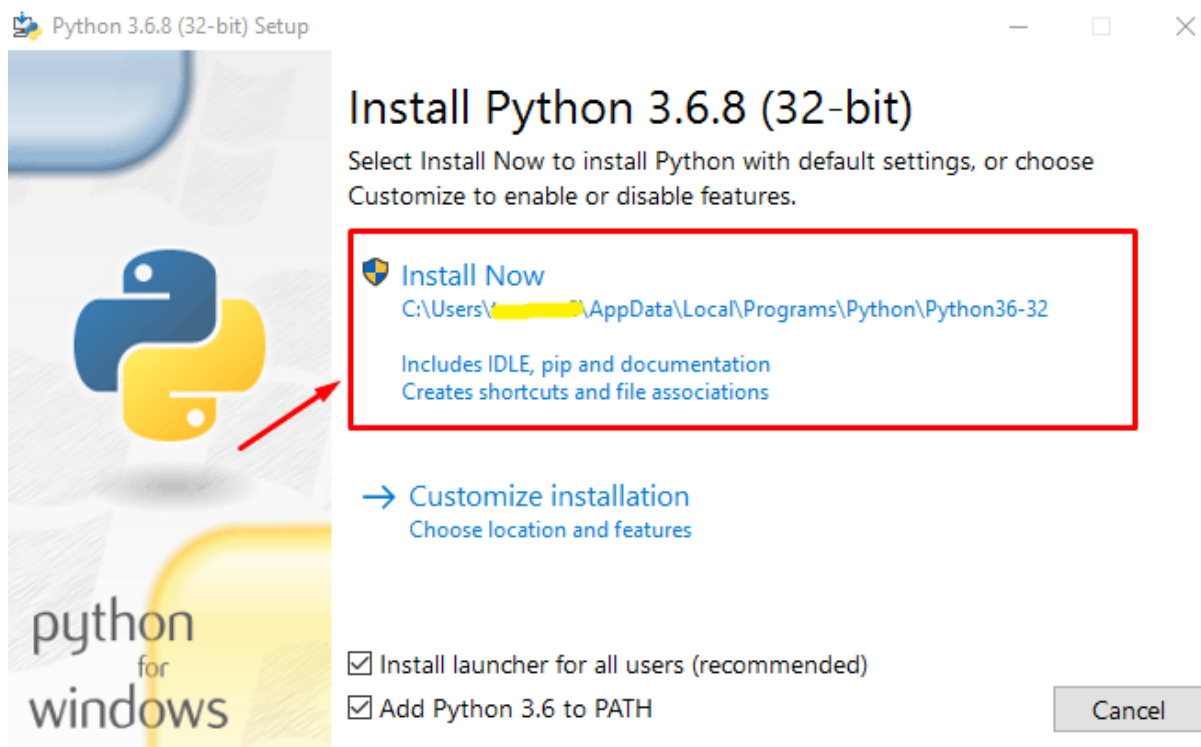
Bước 2 : Bấm tải phiên bản mới nhất .



Hình 4.2.1.2 Giao diện trang download

Bước 3 : Cài đặt

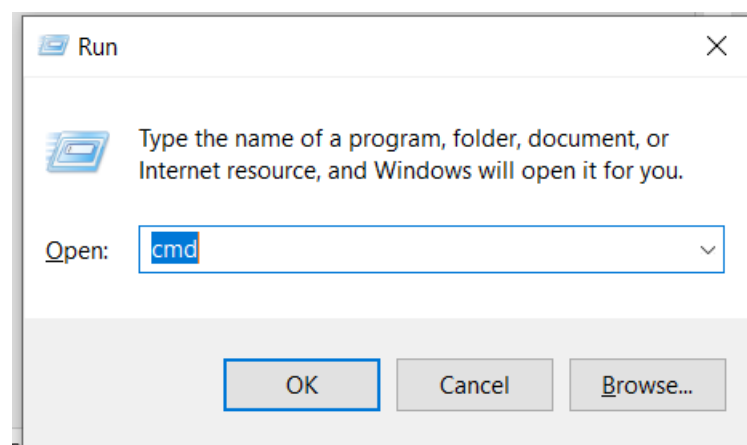




Hình 4.2.1.3 Giao diện cài đặt

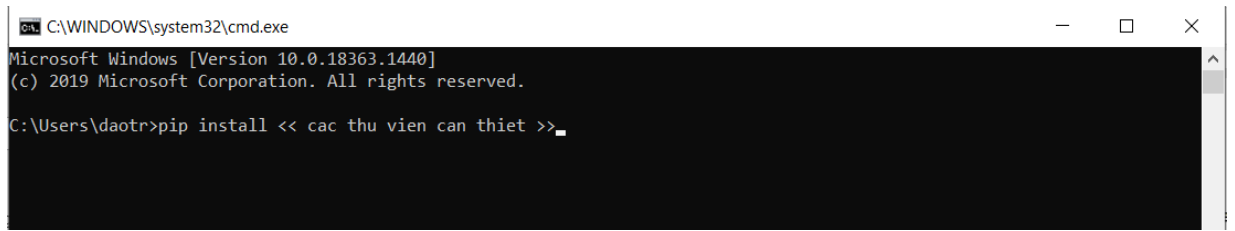
## 4.2.2 Cài đặt các thư viện cần thiết

Bước 1 : Mở terminal(bấm tổ hợp phím window + R gõ cmd rồi bấm Ok hoặc phím Enter)



Hình 4.2.2.1 Giao diện mở hộp thoại Run

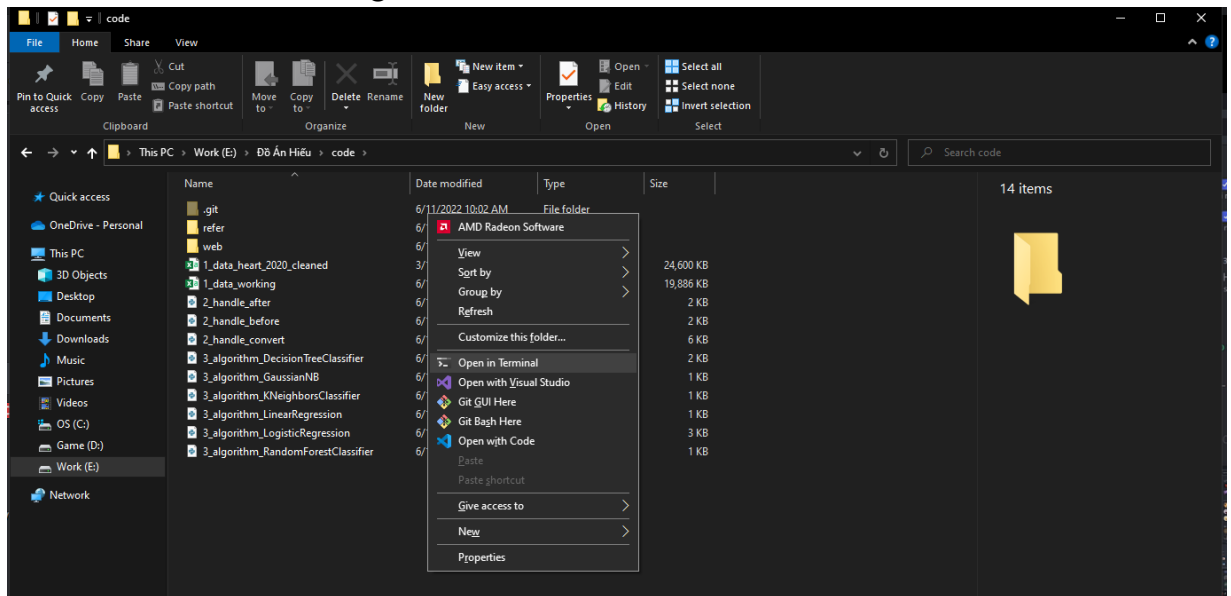
Bước 2 : gõ pip install << các thư viện cần thiết >> (như hình)



Hình 4.2.2.2 Giao diện terminal

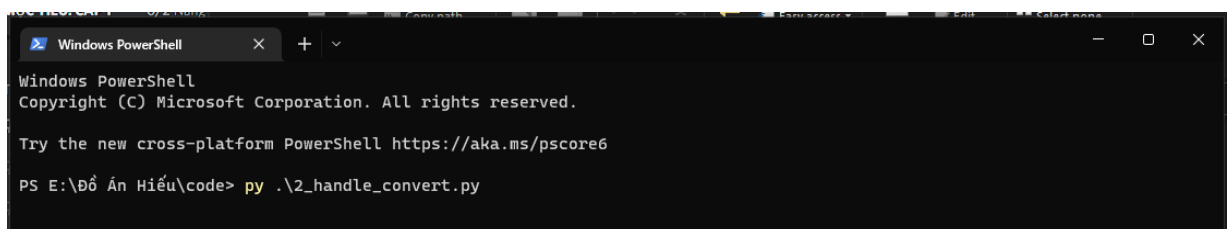
### 4.3 Chạy chương trình tiền xử lý dữ liệu

Bước 1 : mở terminal trong file chứa code.



Hình 4.3.1 Folder chứa code

Bước 2 : Chạy lệnh như trong hình



Hình 4.3.2 Giao diện terminal

## 4.4 Triển khai các thuật toán

### 4.4.1 Thuật toán hồi quy tuyến tính

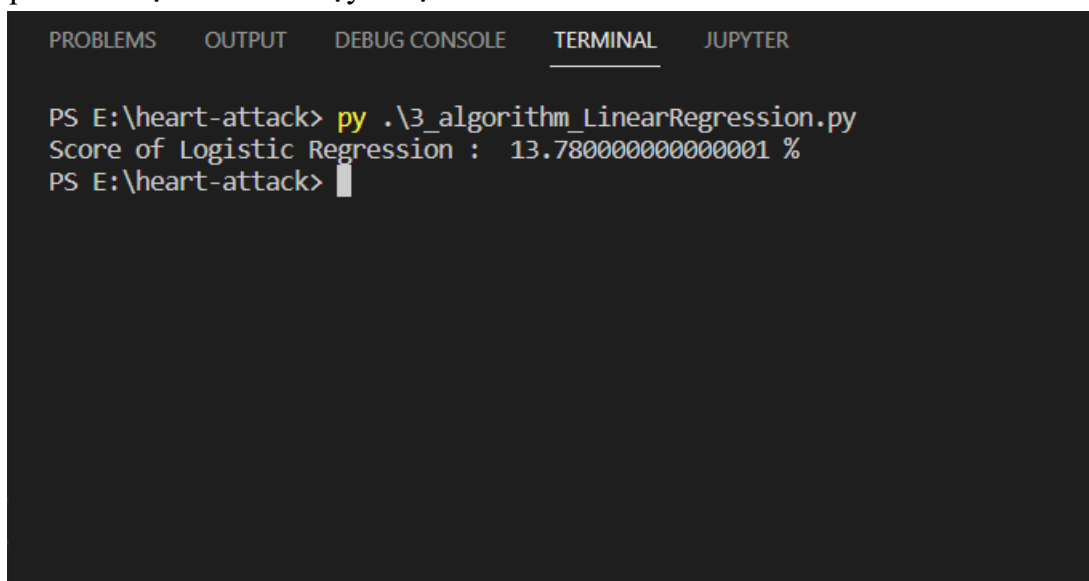
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import requests

#load dataset
data = pd.read_csv('1_data_working.csv',encoding='utf-8',sep=',')

one = np.ones((data.shape[0],1))
data.insert(loc=0,column='A',value=one)
data_X =
data[["A","BMI","Smoking","AlcoholDrinking","Stroke","PhysicalHealth","MentalHe
alth","DiffWalking","Sex","AgeCategory","Race","Diabetic","PhysicalActivity","G
enHealth","SleepTime","Asthma","KidneyDisease","SkinCancer"]]
data_Y = data["HeartDisease"]
Y_train, Y_test, X_train, X_test = train_test_split(data_Y, data_X,
test_size=0.2, random_state=50)

#khớp vào mẫu bằng hồi quy tuyến tính
lr = LinearRegression()
lr.fit(X_train,Y_train)
all_pred = lr.predict(X_test)
score2 = lr.score(X_test,Y_test)
print("Score of Logistic Regression : ",round(score2, 4)*100,"%")
```

Kết quả thu được sau khi chạy thuật toán :



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER

PS E:\heart-attack> py .\3_algorithm_LinearRegression.py
Score of Logistic Regression : 13.780000000000001 %
PS E:\heart-attack>
```

Hình 4.4.1.1 Độ chính xác của thuật toán

#### 4.4.2 Thuật toán hồi quy logic

```
from urllib import response
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
import requests
import json

#load dataset
data = pd.read_csv('1_data_working.csv',encoding='utf-8',sep=',')

one = np.ones((data.shape[0],1))
data.insert(loc=0,column='A',value=one)
data_X =
data[["A","BMI","Smoking","AlcoholDrinking","Stroke","PhysicalHealth","MentalHe
alth","DiffWalking","Sex","AgeCategory","Race","Diabetic","PhysicalActivity","G
enHealth","SleepTime","Asthma","KidneyDisease","SkinCancer"]]
data_Y = data["HeartDisease"]
Y_train, Y_test, X_train, X_test = train_test_split(data_Y, data_X,
test_size=0.2, random_state=50)

#khớp vào mẫu bằng hồi quy logic
lr = LogisticRegression(solver='lbfgs', max_iter=1000)
lr.fit(X_train,Y_train)
all_pred = lr.predict(X_test)
score2 = lr.score(X_test,Y_test)
print("Score of Logistic Regression : ",round(score2, 4)*100,"%")
```

Kết quả thu được sau khi chạy thuật toán :

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER

PS E:\heart-attack> py .\3_algorithm_LogisticRegression.py
Score of Logistic Regression :  91.57 %
PS E:\heart-attack> █
```

Hình 4.4.2.1 Độ chính xác của thuật toán

### 4.4.3 Thuật toán cây quyết định

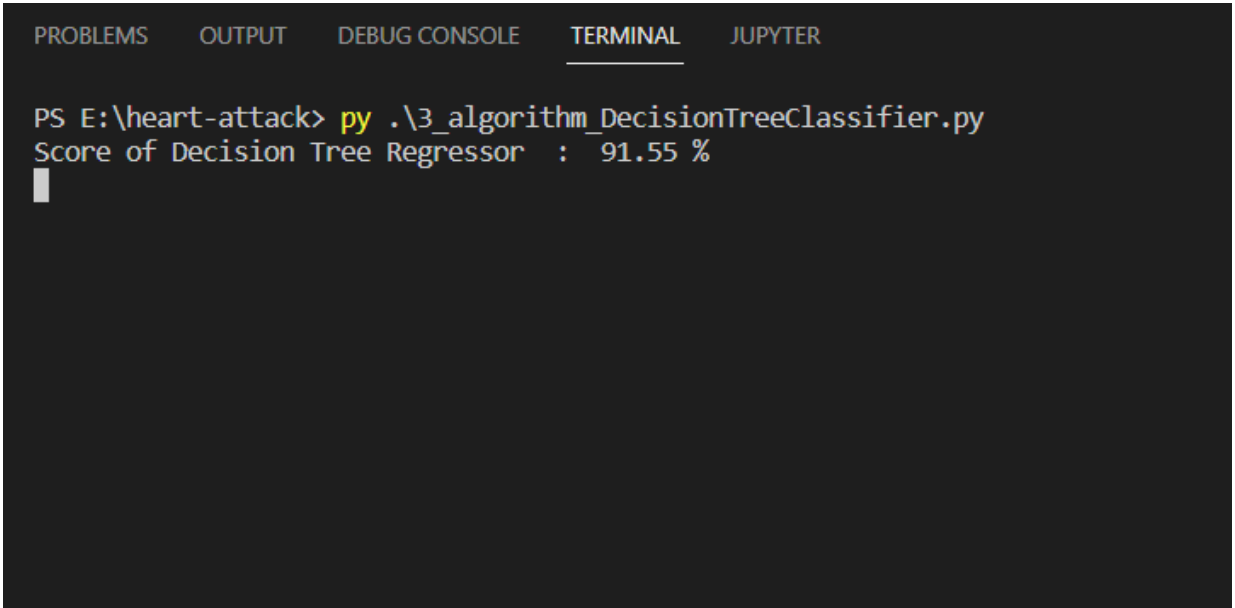
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn import tree

#load dataset
data = pd.read_csv('1_data_working.csv',encoding='utf-8',sep=',')

one = np.ones((data.shape[0],1))
data.insert(loc=0,column='A',value=one)
data_X =
data[["A","BMI","Smoking","AlcoholDrinking","Stroke","PhysicalHealth","MentalHe
alth","DiffWalking","Sex","AgeCategory","Race","Diabetic","PhysicalActivity","G
enHealth","SleepTime","Asthma","KidneyDisease","SkinCancer"]]
data_Y = data["HeartDisease"]
Y_train, Y_test, X_train, X_test = train_test_split(data_Y, data_X,
test_size=0.2, random_state=50)

# khớp vào mẫu bằng cây phân lớp
dtc = DecisionTreeClassifier(max_depth=5)
dtc.fit(X_train, Y_train)
y_pred5 = dtc.predict(X_test)
score3 = dtc.score(X_test,Y_test)
print("Score of Decision Tree Regressor  :",round(score3, 4)*100,"%")
```

Kết quả thu được sau khi chạy thuật toán :



```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    JUPYTER

PS E:\heart-attack> py .\3_algorithm_DecisionTreeClassifier.py
Score of Decision Tree Regressor  :  91.55 %
```

Hình 4.4.3.1 Độ chính xác của thuật toán

#### 4.4.4 So sánh các kết quả và chọn ra thuật toán tốt nhất

Bảng 4.4.4.1 Bảng so sánh độ chính xác thuật toán

STT	Thuật toán	Độ chính xác thu được
1	Hồi quy tuyến tính	13,78%
2	Hồi quy logic	91,57%
3	Cây quyết định	91,55%

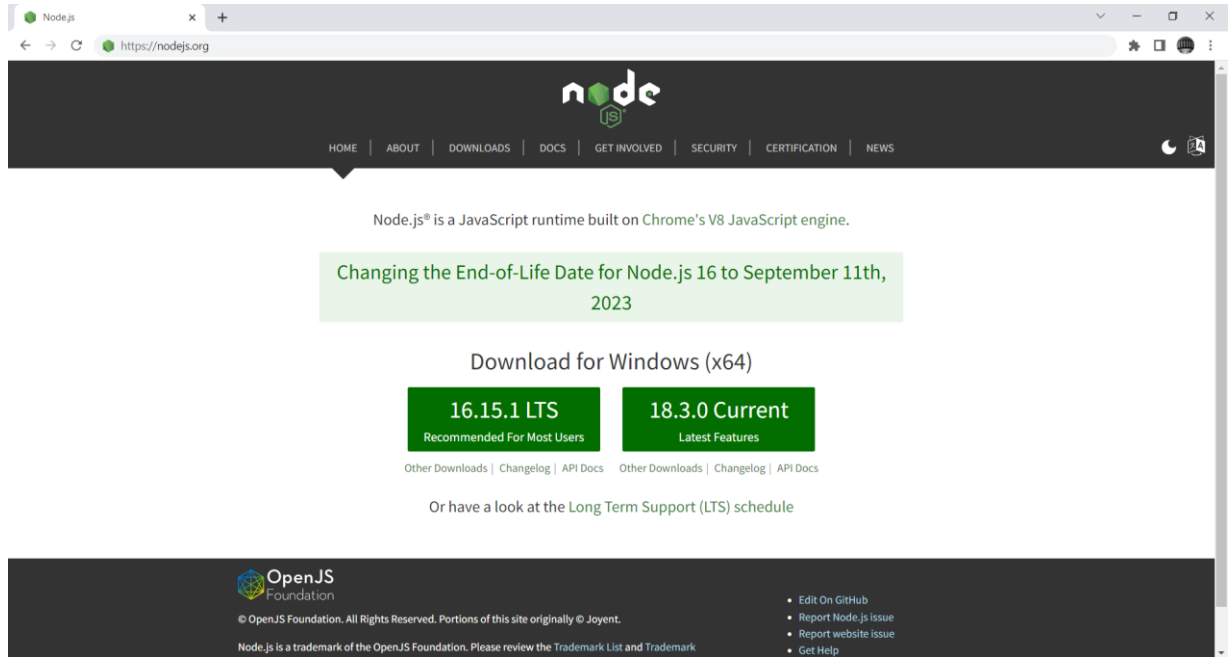
Kết luận : Vì thuật toán hồi quy logic có độ chính xác cao nhất nên được lựa chọn để triển khai vào mô hình sản phẩm thử nghiệm.

## CHƯƠNG 5 TRIỂN KHAI MÔ HÌNH SẢN PHẨM LÊN WEB DEMO

### 5.1 Cài đặt môi trường cần thiết

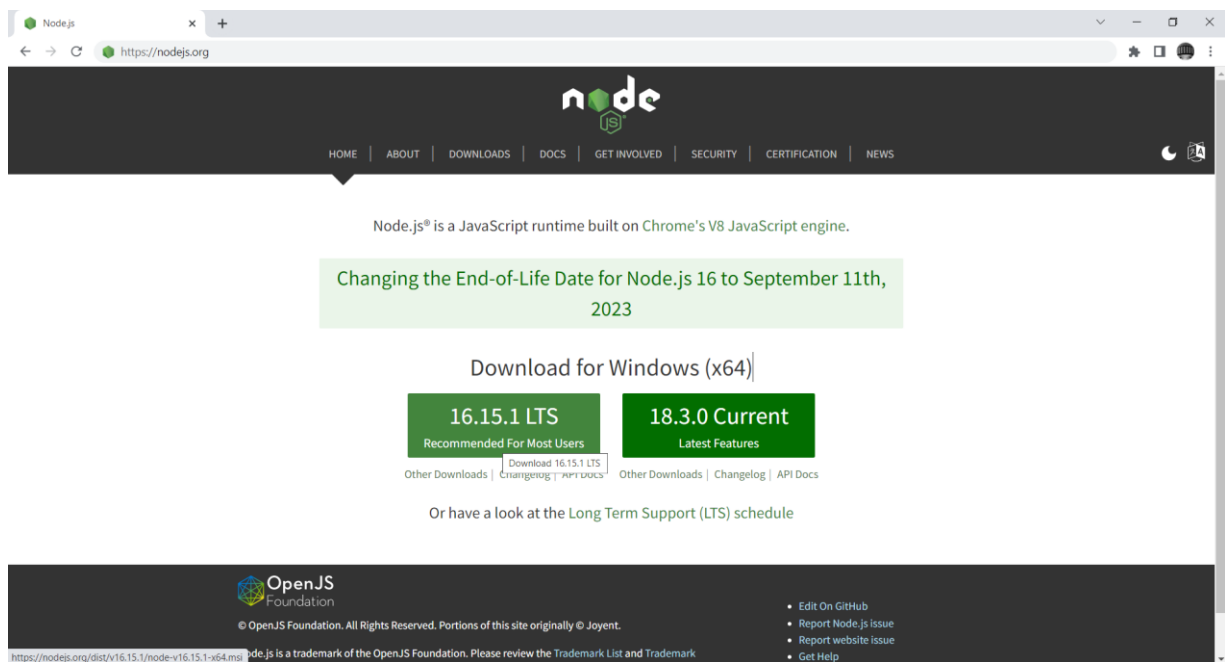
#### 5.1.1 Cài đặt NodeJS

Bước 1 : Cài đặt NodeJS, truy cập trang web <https://nodejs.org/>



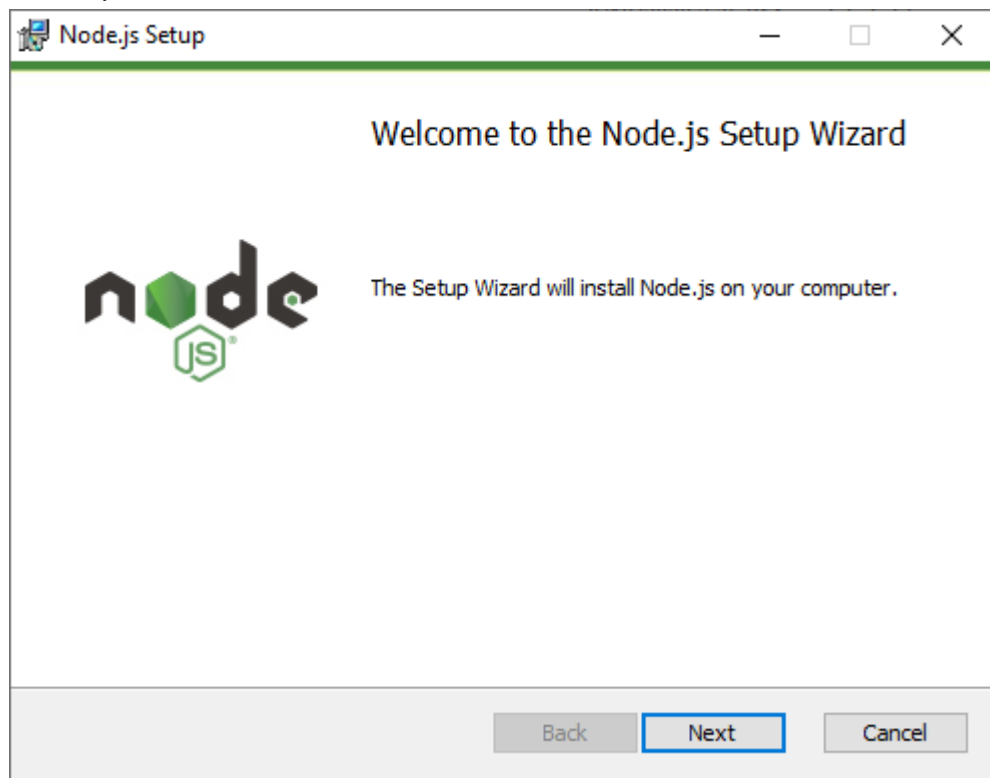
Hình 5.1.1.1 Giao diện trang web

Bước 2 : Tải xuống phiên bản phù hợp.



Hình 5.1.1.2 Giao diện trang download

Bước 3 : Cài đặt



Hình 5.1.1.3 Giao diện cài đặt



## 5.1.2 Cài đặt Xampp

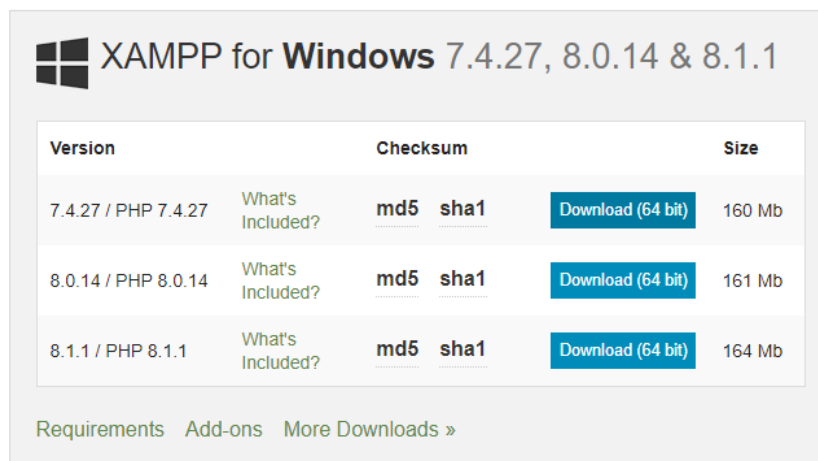
Bước 1 : Truy cập website : <https://www.apachefriends.org>



Hình 5.1.2.1 Giao diện trang web

Bước 2 : Chọn tải phiên bản phù hợp

XAMPP is an easy to install Apache distribution containing MariaDB, PHP, and Perl. Just download and start the installer. It's that easy.



Hình 5.1.2.2 Giao diện trang download

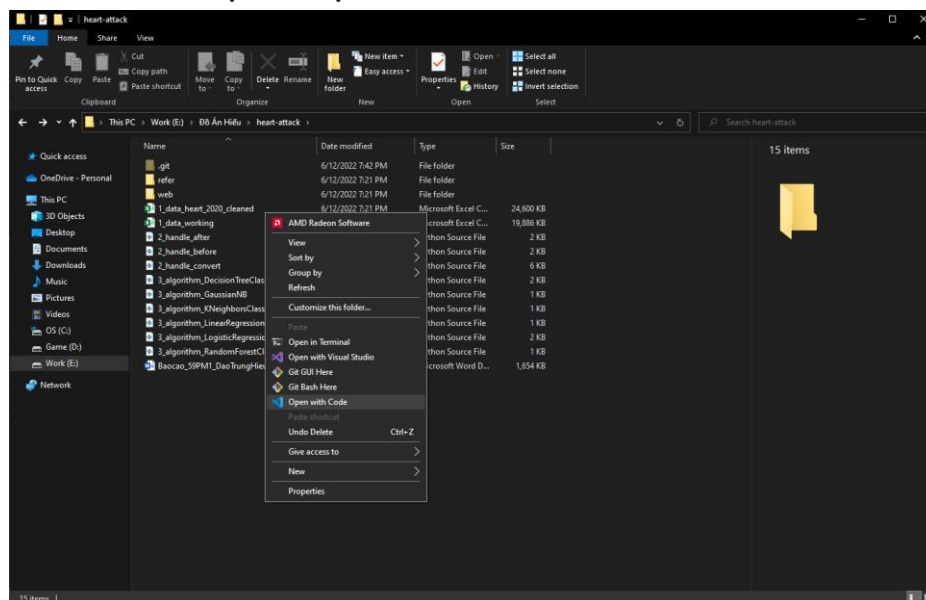
Bước 3 : Cài đặt



Hình 5.1.2.3 Giao diện cài đặt

### 5.1.3 Cài đặt mô hình sản phẩm

Bước 1 : Mở visual code tại file dự án



Hình 5.1.3.1 Giao diện folder chứa code

Bước 2 : Truy cập vào file server



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

PS E:\heart-attack> cd web\server
PS E:\heart-attack\web\server> 
```

Hình 5.1.3.2 Giao diện terminal

Bước 3 : Chạy server của dự án



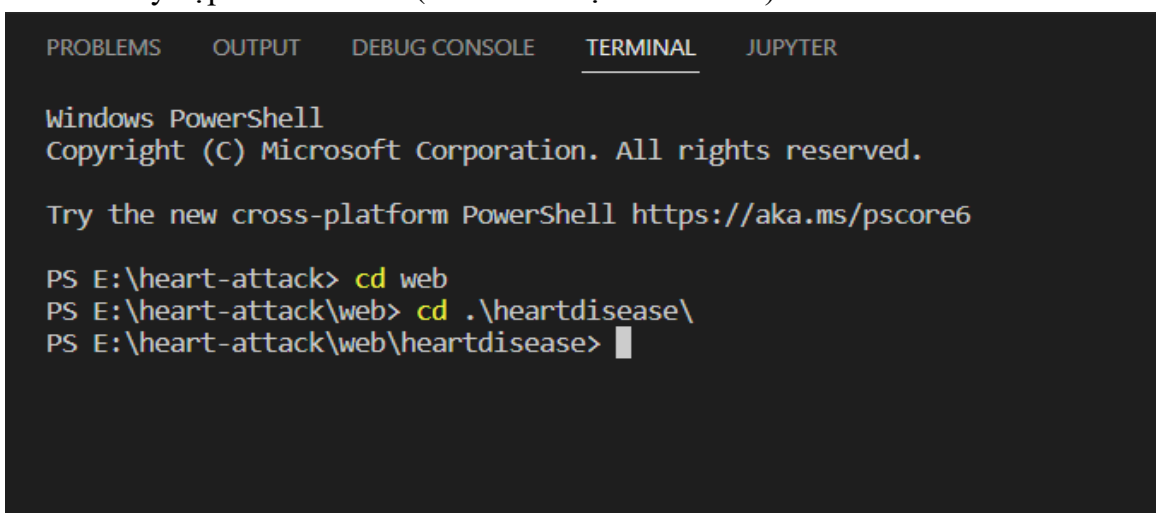
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

PS E:\heart-attack> cd web\server
PS E:\heart-attack\web\server> node .\index.js

```

Hình 5.1.3.3 Giao diện terminal

Bước 4 : Truy cập vào file web (trở về file dự án ban đầu)



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

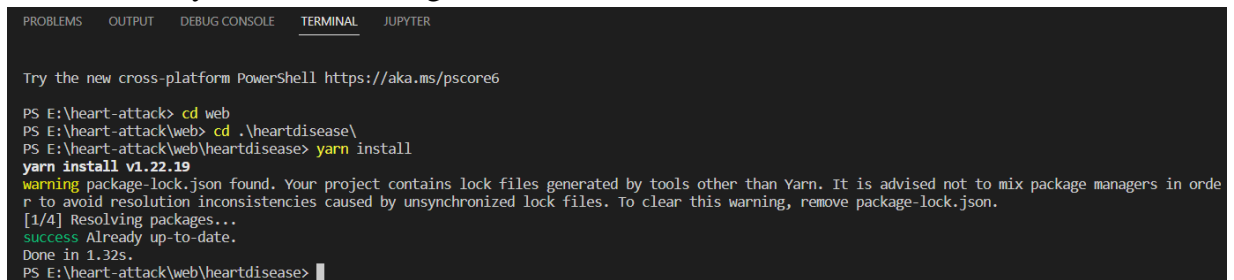
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS E:\heart-attack> cd web
PS E:\heart-attack\web> cd .\heartdisease\
PS E:\heart-attack\web\heartdisease> 
```

Hình 5.1.3.4 Giao diện terminal

## Bước 5 : Chạy file cài đặt trang web



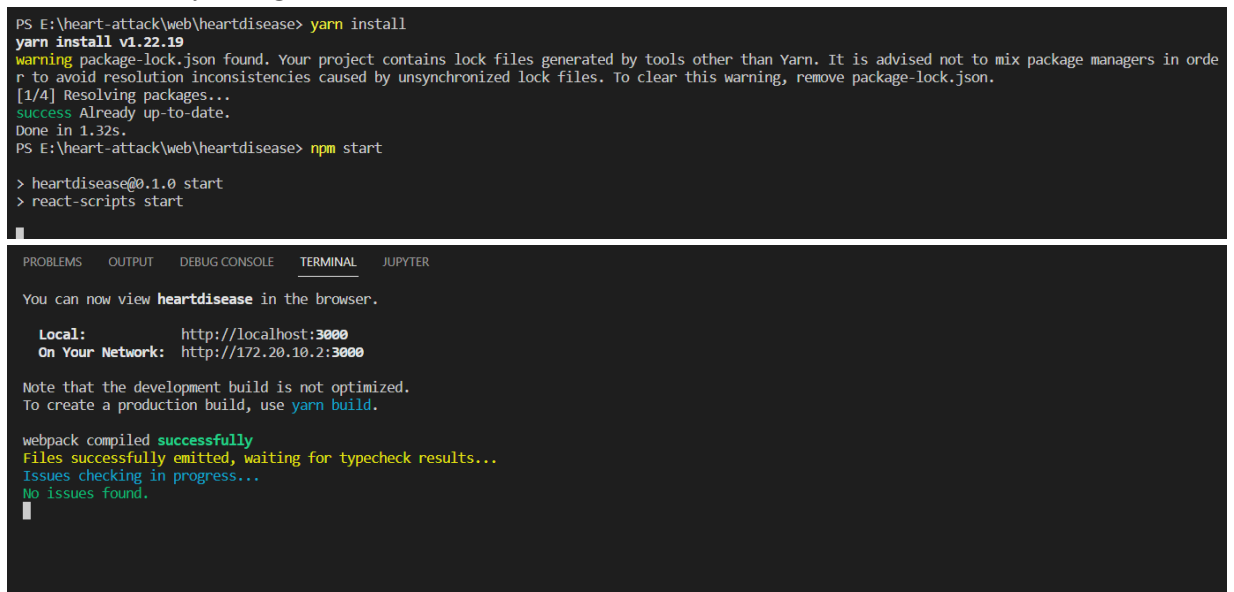
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS E:\heart-attack> cd web
PS E:\heart-attack\web> cd .\heartdisease\
PS E:\heart-attack\web\heartdisease> yarn install
yarn install v1.22.19
warning package-lock.json found. Your project contains lock files generated by tools other than Yarn. It is advised not to mix package managers in order to avoid resolution inconsistencies caused by unsynchronized lock files. To clear this warning, remove package-lock.json.
[1/4] Resolving packages...
success Already up-to-date.
Done in 1.32s.
PS E:\heart-attack\web\heartdisease>
```

Hình 5.1.3.5 Giao diện terminal

## Bước 6 : Chạy trang web



```
PS E:\heart-attack\web\heartdisease> yarn install
yarn install v1.22.19
warning package-lock.json found. Your project contains lock files generated by tools other than Yarn. It is advised not to mix package managers in order to avoid resolution inconsistencies caused by unsynchronized lock files. To clear this warning, remove package-lock.json.
[1/4] Resolving packages...
success Already up-to-date.
Done in 1.32s.
PS E:\heart-attack\web\heartdisease> npm start

> heartdisease@0.1.0 start
> react-scripts start

You can now view heartdisease in the browser.

Local: http://localhost:3000
On Your Network: http://172.20.10.2:3000

Note that the development build is not optimized.
To create a production build, use yarn build.

webpack compiled successfully
files successfully emitted, waiting for typecheck results...
Issues checking in progress...
No issues found.
```

Hình 5.1.3.6 Giao diện terminal

## 5.2 Kết quả cài đặt

### 5.2.1 Hình ảnh trang web

**Heart Disease prediction**

BMI  
0

Chỉ số khối cơ thể

Hút thuốc

☐ Có ☐ Không

Bạn có hút thuốc không ?

Uống rượu

☐ Có ☐ Không

Bạn có uống rượu không ?

Đột quỵ

☐ Có ☐ Không

Bạn đã bao giờ bị đột quỵ chưa ?

Hình 5.2.1.1 Giao diện trang sản phẩm(1)

Sức khoẻ thể chất

0

Bây giờ, hãy nghĩ về sức khỏe thể chất của bạn, bao gồm cả bệnh tật và thương tích, trong bao nhiêu ngày trong suốt 30 ngày qua

Sức khỏe tinh thần

0

Suy nghĩ về sức khỏe tinh thần của bạn, trong 30 ngày qua sức khỏe tinh thần của bạn không tốt là bao nhiêu ngày? (0-30 ngày)

Đi bộ gặp khó khăn

☐ Có ☐ Không

Bạn có gặp khó khăn nghiêm trọng khi đi bộ hoặc leo cầu thang không?

Giới tính

☐ Nữ ☐ Nam

Tuổi

0

Sắc tộc

☐ White ☐ Black ☐ Asian ☐ American Indian/Alaskan Native ☐ Other ☐ Hispanic

Hình 5.2.1.2 Giao diện trang sản phẩm(2)

Bạn có bị tiểu đường không ?

Hoạt động thể chất

☐ Có ☐ Không

Bạn có thường xuyên vận động không ?

Sức khỏe hiện tại

☐ Excellent ☐ Very good ☐ Good ☐ Fair ☐ Poor

Bạn có thể nói rằng nhìn chung sức khỏe của bạn là ...

Thời gian ngủ

0

Hen suyễn

☐ Có ☐ Không

Bạn có bị hen suyễn không ?

Bệnh thận

☐ Có ☐ Không

Hình 5.2.1.3 Giao diện trang sản phẩm(3)

Bạn có bị bệnh thận không ?

Ung thư da

☐ Có ☐ Không

Bạn có bị ung thư da không ?

Kiểm tra

Hình 5.2.1.4 Giao diện trang sản phẩm(4)

## 5.2.2 Kết quả thử nghiệm

Sau khi chạy thử một trường hợp thử nghiệm với thông số ngẫu nhiên nhận được thông báo như sau :

**localhost:3000 says**

Tỉ lệ mắc bệnh tim của bạn là : 7.968697946970191% nguy cơ mắc bệnh của bạn thấp với tỉ lệ chính xác là : 91.57%

OK

Hình 5.2.2.1 Kết quả thử nghiệm

## KẾT LUẬN

Việc áp dụng các mô hình hồi quy học máy trong dự báo chẩn đoán bệnh tim là một chủ đề nghiên cứu đặc biệt quan trọng trong công tác y tế dự phòng và đóng vai trò hết sức quan trọng trong ngành y tế. Các kết quả dự báo bệnh dịch là một đầu vào quan trọng cho việc lập kế hoạch và chuẩn bị các nguồn lực cho công tác phòng chống bệnh tim một cách hiệu quả. Đồ án này tập trung xây dựng các mô hình dự báo chẩn đoán bệnh tim dựa trên bộ dữ liệu có sẵn.

Đối với vấn đề lựa chọn thuật toán phù hợp để ứng dụng trong triển khai xây dựng mô hình hồi quy dự báo bệnh tim trên web demo, dựa vào kết quả trong bảng so sánh độ chính xác thuật toán ở chương 4 và kết quả thực nghiệm đã khẳng định mô hình hồi quy logic cho kết quả dự đoán chính xác nhất. Chính vì vậy đồ án chọn sử dụng thuật toán hồi quy logic(Logistic Regression) có độ chính xác cao nhất trong ba thuật toán sử dụng để triển khai lên web demo.

Tổng hợp những phần chính của đồ án bao gồm :

- Chương 1 : Tổng quan cơ sở lý thuyết. Trong chương này, ta sẽ có cái nhìn tổng quan về các hướng tiếp cận và giải pháp đã được ứng dụng trong hệ thống.
- Chương 2 : Phân tích dữ liệu tiền xử lý. Chương này giới thiệu rõ về từng trường dữ liệu trong bộ dữ liệu gốc.
- Chương 3 : Tiền xử lý dữ liệu. Chương 3 sẽ tập trung tiền xử lý dữ liệu của bộ dữ liệu gốc.
- Chương 4 : Xây dựng hệ thống dự đoán và kết quả thực nghiệm. Chương 4 hướng dẫn cài đặt ngôn ngữ và thư viện cần thiết cho các thuật toán, đồng thời triển khai các thuật toán và so sánh kết quả đưa ra thuật toán tốt nhất.
- Chương 5 : Triển khai mô hình sản phẩm lên web demo. Chương cuối sẽ tập trung trình bày về trang web demo mô hình sản phẩm.

Kết quả đạt được :

- Hiểu được về các mô hình phân lớp, mô hình hồi quy
- Hiểu về các thuật toán: hồi quy tuyến tính, hồi quy logic và cây quyết định.
- Nắm được bài toán chẩn đoán bệnh tim mạch nói chung.
- Hiểu và sử dụng được ngôn ngữ Python trong xây dựng thuật toán hồi quy đối với bài toán chẩn đoán bệnh tim mạch.

Những điểm hạn chế :

- UI trang web demo còn chưa hoàn thiện kỹ càng
- Các thuật toán còn chưa tối ưu
- Chưa có tính ứng dụng thực tiễn cao

Hướng phát triển :

- Phát triển ứng dụng trên một số nền tảng khác
- Phát triển thuật toán tối ưu hơn
- Phát triển UI/UX web demo hoàn thiện hơn



## TÀI LIỆU THAM KHẢO

- [1] <https://www.who.int/vietnam/vi/health-topics/cardiovascular-disease>
- [2] <https://sites.google.com/site/toilamaichovui/machine-learning/khoa-hoc-ml-tren-coudera---andrew-ng/gioi-thieu-ve-machine-learning> , <https://tuyensi.vn/khai-niem-machine-learning/>
- [3] N.T.Tuấn, Sách Deep Learning cơ bản, 2020.  
<https://machinelearningcoban.com/2018/01/14/id3/>
- [4] [Bảng phân loại tình trạng dinh dưỡng - Y Học Công Đồng \(yhoccongdong.com\)](#)
- [5] [PHCN Online - QUÁ TRÌNH PHÁT TRIỂN CON NGƯỜI. ĐAI CƯỜNG. \(phcn-online.com\)](#), [CÁC GIAI ĐOẠN PHÁT TRIỂN CỦA CON NGƯỜI: LÚA TUỔI, ĐẶC ĐIỂM - KHOA HỌC VÀ SỨC KHỎE - 2022 \(encyclopedia-titanica.com\)](#)
- [6] [Thời lượng ngủ theo từng độ tuổi | Vinmec](#)
- [7] <https://viblo.asia/p/tim-hieu-ve-python-co-ban-1-LzD5djkeZjY>