

TRƯỜNG ĐẠI HỌC SÀI GÒN
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO CUỐI KÌ
MÔN HỌC: DATA MINING
ĐỀ TÀI NGHIÊN CỨU:
DỰ ĐOÁN TỈ LỆ TỬ VONG CỦA BỆNH NHÂN COVID 19 Ở
MEXICO BẰNG MÔ HÌNH HỌC MÁY SỬ DỤNG THUẬT
TOÁN PHÂN LỚP LOGISTIC REGRESSION

Giảng viên hướng dẫn: ThS. Nguyễn Thanh Phước
Lớp: DCT1212
Sinh viên thực hiện: Nguyễn Hoàng Tuấn
Mã số sinh viên: 3121410556

Năm học: 2023-2024
Thành phố Hồ Chí Minh - Tháng 5/2024

LỜI CẢM ƠN

Trước hết nhóm xin gửi lời cảm ơn chân thành đến (thầy) ThS.Nguyễn Thanh Phước đã chỉ dẫn, góp ý để báo cáo của nhóm tránh nhiều sai sót và hoàn thiện hơn.

Xin cảm ơn nhà trường và các thầy cô trong khoa công nghệ thông tin đã tạo điều kiện tốt nhất cho nhóm hoàn thành bài báo cáo.

Đồng thời, nhóm cũng cảm ơn các bạn thành viên trong nhóm đã cùng góp sức, xây dựng, thảo luận để sớm hoàn thành đề tài một cách hiệu quả nhất.

Đề án này không chỉ là cơ hội để áp dụng những kiến thức đã học mà còn là cơ hội để chúng em phát triển kỹ năng và tư duy độc lập. Sự hỗ trợ không ngừng từ phía các giáo viên và nhân sự trong Khoa đã đóng vai trò quan trọng, giúp chúng em vượt qua những khó khăn và phát triển ý tưởng của mình.

Do còn nhiều hạn chế về kiến thức và kinh nghiệm, nên báo cáo không thể tránh khỏi những sai sót, mong nhận được góp ý từ thầy và các bạn để bài báo cáo được hoàn thiện hơn.

Cuối cùng, chúng em xin gửi lời cảm ơn đến các tác giả các bài viết, trang web giáo dục đã góp phần cho báo cáo của nhóm các thông tin cần thiết.

Xin chân thành cảm ơn!

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

MỤC LỤC

I. Giới thiệu.....	1
1. Bối cảnh của bài toán	1
1.1. Tóm tắt đại dịch Covid-19.....	1
1.2. Vấn đề cần giải quyết là gì?	1
2. Mục tiêu của bài toán	1
2.1. Mục tiêu chung.....	1
2.2. Mục tiêu cụ thể.....	2
3. Định nghĩa bài toán	2
4. Các giải pháp để giải quyết bài toán	3
5. Vai trò của khai phá dữ liệu trong việc giải quyết bài toán trên	4
II. Phân tích dữ liệu	5
1. Mô tả dữ liệu	5
2. Tổng quan về dữ liệu.....	6
3. Trực quan hóa dữ liệu.....	12
4. Những giả thiết khi thu thập dữ liệu.....	18
4.1. Trong trường hợp nào thì thu thập được dữ liệu	18
4.2. Trong trường hợp nào không thể thu thập được dữ liệu.....	19
III. Khai phá dữ liệu	20
1. Quy trình khai phá dữ liệu.....	20
2. Nguyên lý hoạt động của thuật toán Logistic Regression.....	20
2.1. Thu thập dữ liệu.....	20
2.2. Khởi tạo trọng số	21
2.3. Tính toán với các giá trị dự đoán.....	21

2.4. Tính toán hàm mất mát.....	21
2.5. Cập nhật trọng số.....	22
2.6. Dự đoán cho dữ liệu mới.....	23
3. Tiêu chí đánh giá mô hình.....	25
IV. Chọn thuật toán và đánh giá.....	29
1. Yêu cầu về chương trình	29
1.1. Lựa chọn thuật toán Machine Learning	29
1.2. Import các thư viện cần thiết.....	29
2. Tiền xử lý dữ liệu	30
2.1. Mã hóa dữ liệu.....	30
2.2. Xử lý giá trị bị thiếu	31
2.3. Thống kê dữ liệu.....	33
2.4. Lựa chọn các đặc trưng	36
2.5. Chuẩn hóa dữ liệu.....	37
3. Huấn luyện mô hình	38
3.1. Chia cắt tập dữ liệu.....	38
3.2. Lựa chọn thuật toán.....	39
4. Xử lý mất cân bằng dữ liệu	40
4.1. Kỹ thuật Resampling	40
4.2. Huấn luyện lại mô hình	42
5. Đánh giá mô hình	43
5.1. Mô hình khi chưa cân bằng mẫu	43
5.2. Mô hình cân bằng mẫu dữ liệu	43
V. Kết quả.....	47
1. Độ chính xác của mô hình.....	47

1.1. Kiểm tra trên mẫu thử	47
1.2. So sánh với các thuật toán phân lớp khác	47
2. Tính giải thích của mô hình.....	49
VI. Thảo luận và kết luận	51
1. Tóm tắt bài toán.....	51
2. Khả năng ứng dụng của giải pháp/mô hình.....	52
3. Ưu điểm – nhược điểm của giải pháp/mô hình	53
4. Đề xuất.....	53

DANH MỤC CÁC BẢNG BIỂU

Bảng 1. Ưu nhược điểm của các phương pháp truyền thống và CNTT.....	3
Bảng 2. Mô tả thuộc tính dữ liệu.....	5
Bảng 3. Chia cắt tập dữ liệu	39
Bảng 4. Chia tập huấn luyện sau cân bằng mẫu.....	42
Bảng 5. Kết quả kiểm thử mẫu.....	47

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

Hình 1. Các kiểu dữ liệu.....	7
Hình 2. Giá trị bị thiếu.....	8
Hình 3. Phần trăm giá trị 97, 98, 99	9
Hình 4. Số giá trị mỗi đặc trưng	10
Hình 5. Dữ liệu cột DATE_DIED	11
Hình 6. Giá trị thiếu của cột PNEUMONIA	11
Hình 7. Phân bố số người tử vong của bệnh nhân covid.....	12
Hình 8. Biểu đồ phân bố độ tuổi của bệnh nhân covid	13
Hình 9. Biểu đồ số ca tử vong của các độ tuổi ở nam nữ.....	14
Hình 10. Biểu đồ số ca tử vong ở nam và nữ	15
Hình 11. Biểu đồ tỉ lệ tử vong của bệnh nhân béo phì	16
Hình 12. Biểu đồ phân bố các bệnh nền của bệnh nhân covid.....	17
Hình 13. Ảnh minh họa đồ thị của hàm sigmoid	22
Hình 14. Đồ thị dự đoán của model là đúng	23
Hình 15. Đồ thị dự đoán của model là sai	24
Hình 16. Ảnh minh họa đường phân chia của Logistic Regression.....	25
Hình 17. Biểu đồ minh họa đánh giá mô hình của độ đo ROC.....	27
Hình 18. Biểu diễn của đường cong ROC.....	28
Hình 19. Mã hóa cột DATE_DIED	31
Hình 20. Số bệnh nhân covid có thai.....	31
Hình 21. Số bệnh nhân có kết nối máy thở	32
Hình 22. Số bệnh nhân vào Khoa Hồi sức	33
Hình 23. Thống kê dữ liệu mã hóa	34
Hình 24. Mối tương quan của các đặc trưng	35

Hình 25. Bảng dữ liệu huấn luyện mô hình	37
Hình 26. Dữ liệu cột CLASIFFICSTION_FINAL	37
Hình 27. Chuẩn hóa cột CLASIFFICSTION_FINAL	38
Hình 28. Biểu đồ phân bố số ca tử vong	41
Hình 29. Biểu đồ phân bố số ca tử vong sau cân bằng.....	42
Hình 30. Kết quả huấn luyện mô hình.....	43
Hình 31. Report mô hình	43
Hình 32. Confusion Matrix Logistic	45
Hình 33. ROC Curve của mô hình	46
Hình 34. So sánh độ chính xác của các mô hình.....	49
Hình 35. Biểu diễn Linear Regresssion, PLA, Logistic Regression theo Neural Network.....	53

I. Giới thiệu

1. Bối cảnh của bài toán

1.1. Tóm tắt đại dịch Covid-19

Xuất hiện lần đầu tiên tại Vũ Hán, Trung Quốc vào tháng 12 năm 2019. Gây ra bởi virus SARS-CoV-2. Lây lan nhanh chóng trên toàn cầu, trở thành đại dịch vào tháng 3 năm 2020. Gây ra nhiều ca bệnh, tử vong và ảnh hưởng nặng nề đến kinh tế, xã hội trên toàn thế giới.

1.2. Vấn đề cần giải quyết là gì?

Bệnh vi rút Corona (Covid-19) là một bệnh truyền nhiễm do một loại vi rút Corona mới được phát hiện gây ra. Hầu hết những người bị nhiễm virus COVID-19 sẽ bị bệnh hô hấp từ nhẹ đến trung bình và hồi phục mà không cần điều trị đặc biệt. Người già và những người có bệnh lý tiềm ẩn như bệnh tim mạch, tiểu đường, bệnh hô hấp mãn tính và ung thư có nhiều khả năng mắc bệnh nghiêm trọng hơn.

Dịch bệnh COVID-19 đã và đang là một vấn đề y tế toàn cầu nghiêm trọng. Kể từ khi bùng phát vào đầu năm 2020, dịch bệnh đã cướp đi sinh mạng của hàng triệu người trên toàn thế giới. Riêng tại Việt Nam, dịch bệnh COVID-19 đã được kiểm soát tương đối tốt. Tuy nhiên, vẫn có những ca bệnh nặng và tử vong, đặc biệt là ở những người cao tuổi, có bệnh nền, hoặc chưa được tiêm vaccine. Việc dự đoán bệnh nhân COVID-19 có nguy cơ tử vong cao, xác định sớm những bệnh nhân có nguy cơ tử vong cao là rất quan trọng để có thể cung cấp các biện pháp điều trị và chăm sóc phù hợp. Điều này có thể giúp giảm tỷ lệ tử vong và cải thiện chất lượng sống của những bệnh nhân này.

2. Mục tiêu của bài toán

2.1. Mục tiêu chung

Trong suốt thời gian diễn ra đại dịch, một trong những vấn đề chính mà phải đối mặt là thiếu nguồn lực y tế và kế hoạch phù hợp để phân phối chúng một cách hiệu quả. Trong những thời điểm khó khăn này, việc có thể dự đoán mà một cá nhân

có thể cần vào thời điểm có kết quả xét nghiệm dương tính hoặc thậm chí trước đó sẽ giúp ích rất nhiều cho chính quyền vì họ có thể sắp xếp các nguồn lực cần thiết để cứu sống bệnh nhân đó.

Mục tiêu chính của dự án này là xây dựng mô hình Machine Learning để dự đoán tỉ lệ tử vong của bệnh nhân Covid-19, dựa trên triệu chứng, tình trạng và tiền sử bệnh hiện tại của bệnh nhân Covid-19, sẽ dự đoán liệu bệnh nhân có nguy cơ cao hay không.

2.2. Mục tiêu cụ thể

Công việc nghiên cứu này nhằm mục đích xác định thuật toán phân loại tốt nhất để xác định khả năng tử vong cao hay thấp bệnh nhân. Dự án nhỏ này được chứng minh bằng cách thực hiện một nghiên cứu và phân tích so sánh bằng cách sử dụng các thuật toán phân loại là Logistic Regression được sử dụng ở các cấp độ đánh giá khác nhau. Dự đoán tỉ lệ tử vong do bệnh Covid gây ra là một nhiệm vụ quan trọng liên quan đến độ chính xác cao. Do đó, các thuật toán được đánh giá ở nhiều cấp độ và loại chiến lược đánh giá. Điều này sẽ cung cấp cho các nhà nghiên cứu và các học viên y tế tiên lượng sớm các triệu chứng và hỗ trợ đưa ra những quyết định, phương pháp điều trị phù hợp với những bệnh nhân có nguy cơ cao và giảm các biến chứng. Vậy nên mục tiêu nghiên cứu nhỏ này muốn hướng đến là:

- Đánh giá hiệu quả của mô hình trên dữ liệu thực tế.
- Góp phần nâng cao hiệu quả điều trị và giảm thiểu tỷ lệ tử vong do Covid-19.
- Giảm chi phí, tiết kiệm thời gian.
- Giúp đưa ra gợi ý cho bác sĩ.

3. Định nghĩa bài toán

Trong dự án này bộ dữ liệu được lấy sử dụng về thông tin bệnh nhân Covid 19 được công bố và cho phép của chính phủ của Mexico trên các trang thông tin và dẫn dần uy tín có nguồn gốc rõ ràng và sẽ phân tích, dự đoán kết quả của bệnh nhân liệu họ có nguy cơ tử vong cao hay không, tức là dự đoán tỉ lệ tử vong của bệnh nhân

Covid 19 bằng cách sử dụng thuật toán phân lớp Logistic Regression trong học máy. Dự đoán này sẽ làm cho nó nhanh hơn và hiệu quả hơn trong lĩnh vực y học cần tốn nhiều thời gian.

4. Các giải pháp để giải quyết bài toán

Chẩn đoán y khoa cụ thể là các biến chứng của Covid 19 được coi là một nhiệm vụ quan trọng, phức tạp và cần nhiều thời gian. Vì vậy các chẩn đoán được đưa ra đều dựa trên trực giác, kinh nghiệm và chuyên môn của các bác sĩ chuyên ngành.

Nếu không sử dụng các công nghệ của lĩnh vực CNTT như dữ liệu lớn, học máy, và trí tuệ nhân tạo, thì việc xử lý bài toán dự đoán tỉ lệ tử vong của bệnh nhân COVID-19 trong y học có thể được thực hiện theo các cách như: sử dụng các phương pháp thống kê truyền thống, sử dụng các bảng phân tích, sử dụng kinh nghiệm của các bác sĩ.

Bảng 1. Ưu nhược điểm của các phương pháp truyền thống và CNTT

Phương pháp	Ưu điểm	Nhược điểm
Dữ liệu lớn và học máy	Độ chính xác cao	Yêu cầu nhiều về dữ liệu và các kỹ năng CNTT
Thống kê truyền thống	Ít yêu cầu về dữ liệu và kỹ năng	Độ chính xác thấp hơn
Bảng phân tích	Dễ thực hiện	Không chính xác
Kinh nghiệm của bác sĩ	Có tính thực tiễn	Phụ thuộc vào kiến thức và kỹ năng của bác sĩ

Việc lựa chọn phương pháp xử lý phù hợp phụ thuộc vào nhiều yếu tố, bao gồm:

- Mức độ chính xác cần thiết
- Dữ liệu sẵn có
- Kỹ năng và kinh nghiệm của các bác sĩ

Thông thường, các phương pháp CNTT như dữ liệu lớn và học máy được ưu tiên sử dụng vì độ chính xác cao. Tuy nhiên, các phương pháp thống kê truyền thống, bảng phân tích, và kinh nghiệm của bác sĩ vẫn có thể được sử dụng trong các trường hợp không có đủ dữ liệu hoặc không có các kỹ năng về CNTT.

5. Vai trò của khai phá dữ liệu trong việc giải quyết bài toán trên

Thách thức lớn của nhân loại - Covid 19 ở thời điểm bây giờ là một mối đe dọa lớn nguy hiểm và như đã thấy vẫn chưa tìm ra giải pháp thuốc vaccin đặc trị thì chúng đã sinh ra những biến thể đột biến khác. Và với lực lượng đội ngũ y bác sĩ lúc bây giờ khi ở tuyến đầu chống dịch thì sức máy cũng sẽ không thể chống lại nổi với số lượng như cấp số nhân bệnh nhân mắc nhiễm Covid như thế. Giả sử trong trường hợp một bệnh nhân Covid mới được đưa vào phòng cấp cứu thì đội ngũ y bác sĩ cần phải xác định xem bệnh nhân này có nguy cơ tử vong cao hay không để có thể đưa ra quyết định phù hợp. Phát hiện sớm sẽ giúp giảm tỉ lệ tử vong từ các biến chứng tổng thể. Và việc phân tích các thông tin về bệnh nhân bao gồm tuổi, giới tính, bệnh nền, triệu chứng, và kết quả xét nghiệm để đưa ra dự đoán thì sẽ mất rất nhiều thời gian và công sức. Vì vậy, trong dự án này mục đích nghiên cứu đã phát triển và nghiên cứu mô hình để dự đoán tỉ lệ tử vong cao hay thấp dựa trên các thuật toán máy học

- Giúp các bác sĩ đánh giá mức độ nguy hiểm của bệnh nhân:
- Hỗ trợ đưa ra quyết định điều trị phù hợp.
- Giúp phân bổ nguồn lực y tế hiệu quả.
- Góp phần giảm thiểu tỷ lệ tử vong.

II. Phân tích dữ liệu

1. Mô tả dữ liệu

“**COVID-19 Dataset**” được cung cấp bởi chính phủ Mexico và được đăng công khai trên Kaggle - nền tảng trực tuyến dành cho cộng đồng khoa học dữ liệu và trí tuệ nhân tạo (AI). Tập dữ liệu này chứa một số lượng lớn thông tin ẩn danh liên quan đến bệnh nhân, bao gồm cả các tiền sử bệnh lý.

Số liệu sơ bộ được Bộ Y tế xác nhận thông qua Tổng cục Dịch tễ học. Thông tin trong đó chỉ tương ứng với dữ liệu thu được từ nghiên cứu Dịch tễ học về một trường hợp nghi ngờ mắc bệnh hô hấp do virus tại thời điểm được xác định tại các đơn vị y tế của ngành Y tế.

Theo chẩn đoán lâm sàng khi nhập viện, bệnh nhân được coi là bệnh nhân ngoại trú hoặc bệnh nhân nhập viện. Cơ sở không bao gồm diễn biến trong thời gian họ ở trong các đơn vị y tế, ngoại trừ thông tin cập nhật khi xuất viện của đơn vị giám sát dịch tễ học của bệnh viện hoặc cơ quan y tế trong trường hợp tử vong.

Tập dữ liệu thô bao gồm 21 thuộc tính đặc điểm riêng biệt của 1.048.576 bệnh nhân. Trong các thuộc tính có kiểu dữ liệu Boolean: 1 có nghĩa là "có" và 2 có nghĩa là "không", các giá trị 97 và 99 được coi là dữ liệu bị thiếu.

Bảng 2. Mô tả thuộc tính dữ liệu

sex	1 là nữ và 2 là nam.
age	độ tuổi của bệnh nhân.
classification	kết quả xét nghiệm covid. Giá trị 1-3 có nghĩa là bệnh nhân được chẩn đoán mắc covid ở các mức độ khác nhau. 4 hoặc cao hơn có nghĩa là bệnh nhân không mang covid hoặc xét nghiệm không có kết luận.
patient type	loại chăm sóc mà bệnh nhân nhận được tại đơn vị. 1 là trở về nhà và 2 là nhập viện.
pneumonia	bệnh nhân đã có viêm túi khí hay chưa.
pregnancy	bệnh nhân có đang mang thai hay không.

diabetes	bệnh nhân có tiểu đường hay không.
copd	Cho biết bệnh nhân có bệnh phổi tắc nghẽn mạn tính hay không.
asthma	bệnh nhân có hen suyễn hay không.
inmsupr	bệnh nhân có bị suy giảm miễn dịch hay không.
hypertension	bệnh nhân có tăng huyết áp hay không.
cardiovascular	bệnh nhân có bệnh liên quan đến tim hoặc mạch máu hay không.
renal chronic	bệnh nhân có bệnh thận mãn tính hay không.
other disease	bệnh nhân có mắc bệnh khác hay không.
obesity	bệnh nhân có béo phì hay không.
tobacco	bệnh nhân có sử dụng thuốc lá hay không.
usmr	Cho biết bệnh nhân đã được điều trị tại các đơn vị y tế cấp 1, 2 hay 3.
medical unit	loại hình cơ sở của Hệ thống Y tế Quốc gia cung cấp dịch vụ chăm sóc.
intubed	bệnh nhân có được nối máy thở hay không.
icu	Cho biết bệnh nhân đã được nhập vào Khoa Hồi sức Cấp cứu hay chưa.
date died	Nếu bệnh nhân đã tử vong, hãy ghi ngày mất, nếu không hãy ghi 9999-99-99.

2. Tổng quan về dữ liệu

Đại dịch COVID-19 đã gây ra tác động to lớn đến sức khỏe cộng đồng trên toàn thế giới, dẫn đến nhu cầu cấp thiết về thông tin và dữ liệu để hiểu rõ hơn về căn bệnh này, từ đó phát triển các biện pháp phòng ngừa và điều trị hiệu quả. Tập dữ liệu COVID-19 đóng vai trò quan trọng trong việc cung cấp thông tin chi tiết về các trường hợp mắc bệnh, đặc điểm bệnh nhân, tỷ lệ mắc bệnh và tử vong, cũng như hiệu quả của các biện pháp can thiệp.

Tập dữ liệu COVID-19 là tập hợp các kho lưu trữ thông tin về các ca bệnh được ghi nhận trên toàn cầu. Dữ liệu thường bao gồm các thông tin sau:

- Thông tin về bệnh nhân: Tuổi, giới tính, quốc gia, tiền sử bệnh lý.
- Thông tin về chẩn đoán: Ngày khởi phát triệu chứng, kết quả xét nghiệm, chẩn đoán lâm sàng.
- Thông tin về điều trị: Viện trợ y tế, sử dụng thuốc, nhập viện, tử vong.
- Thông tin về dịch tễ học: Mỗi liên hệ truy vết, lịch sử di chuyển, tỷ lệ mắc bệnh theo khu vực.

```

#      Column      Non-Null Count  Dtype
---  -
0     USMER        1048575 non-null  int64
1     MEDICAL_UNIT 1048575 non-null  int64
2     SEX          1048575 non-null  int64
3     PATIENT_TYPE  1048575 non-null  int64
4     DATE_DIED    1048575 non-null  object
5     INTUBED      1048575 non-null  int64
6     PNEUMONIA     1048575 non-null  int64
7     AGE          1048575 non-null  int64
8     PREGNANT      1048575 non-null  int64
9     DIABETES      1048575 non-null  int64
10    COPD          1048575 non-null  int64
11    ASTHMA        1048575 non-null  int64
12    INMSUPR       1048575 non-null  int64
13    HIPERTENSION  1048575 non-null  int64
14    OTHER_DISEASE  1048575 non-null  int64
15    CARDIOVASCULAR 1048575 non-null  int64
16    OBESITY        1048575 non-null  int64
17    RENAL_CHRONIC  1048575 non-null  int64
18    TOBACCO        1048575 non-null  int64
19    CLASIFFICATION_FINAL 1048575 non-null  int64
20    ICU           1048575 non-null  int64
dtypes: int64(20), object(1)
memory usage: 168.0+ MB

```

Hình 1. Các kiểu dữ liệu

Như đã thấy ở trên thì sẽ có 2 loại kiểu dữ liệu object và int. Tất cả các cột đều có số lượng bản ghi không có giá trị null. Điều này có nghĩa là không có bản ghi nào trong tập dữ liệu bị thiếu bất kỳ thông tin nào cho các cột được liệt kê trong bảng.

USMER	0
MEDICAL_UNIT	0
SEX	0
PATIENT_TYPE	0
DATE_DIED	0
INTUBED	0
PNEUMONIA	0
AGE	0
PREGNANT	0
DIABETES	0
COPD	0
ASTHMA	0
INMSUPR	0
HIPERTENSION	0
OTHER_DISEASE	0
CARDIOVASCULAR	0
OBESITY	0
RENAL_CHRONIC	0
TOBACCO	0
CLASIFFICATION_FINAL	0
ICU	0
dtype: int64	

Hình 2. Giá trị bị thiếu

Tất cả các cột trong DataFrame đều không có giá trị null. Điều này có nghĩa là dữ liệu trong DataFrame hoàn chỉnh và không có bất kỳ giá trị nào bị thiếu. Kết quả này cho thấy chất lượng dữ liệu trong DataFrame là tốt.

USMER	=>	0.00%
MEDICAL_UNIT	=>	0.00%
SEX	=>	0.00%
PATIENT_TYPE	=>	0.00%
DATE_DIED	=>	0.00%
INTUBED	=>	81.62%
PNEUMONIA	=>	1.53%
AGE	=>	0.03%
PREGNANT	=>	50.28%
DIABETES	=>	0.32%
COPD	=>	0.29%
ASTHMA	=>	0.28%
INMSUPR	=>	0.32%
HIPERTENSION	=>	0.30%
OTHER_DISEASE	=>	0.48%
CARDIOVASCULAR	=>	0.29%
OBESITY	=>	0.29%
RENAL_CHRONIC	=>	0.29%
TOBACCO	=>	0.31%
CLASIFFICATION_FINAL	=>	0.00%
ICU	=>	81.64%

Hình 3. Phần trăm giá trị 97, 98, 99

Tính toán cụ thể tỷ lệ phần trăm giá trị thiếu cho các mục bằng 97, 98 hoặc 99 trong mỗi cột. Như đã mô tả dữ liệu ở bảng 2 các giá trị thiếu được biểu thị bằng các số 97, 98 và 99. Nhìn chung, chất lượng dữ liệu trong tập dữ liệu được đánh giá là tốt, với hầu hết các cột có tỷ lệ phần trăm giá trị thiếu thấp. Tuy nhiên, cột ICU có tỷ lệ phần trăm giá trị thiếu cao, INTUBED cũng được cảnh báo, cần được xem xét kỹ lưỡng hơn để xác định nguyên nhân và ảnh hưởng đến việc phân tích dữ liệu.

USMER	=>	2
MEDICAL_UNIT	=>	13
SEX	=>	2
PATIENT_TYPE	=>	2
DATE_DIED	=>	401
INTUBED	=>	4
PNEUMONIA	=>	3
AGE	=>	121
PREGNANT	=>	4
DIABETES	=>	3
COPD	=>	3
ASTHMA	=>	3
INMSUPR	=>	3
HIPERTENSION	=>	3
OTHER_DISEASE	=>	3
CARDIOVASCULAR	=>	3
OBESITY	=>	3
RENAL_CHRONIC	=>	3
TOBACCO	=>	3
CLASIFFICATION_FINAL	=>	7
ICU	=>	4

Hình 4. Số giá trị mỗi đặc trưng

Nhìn chung, số loại giá trị trong các thuộc tính của tập dữ liệu tương đối nhỏ. Điều này cho thấy rằng hầu hết các thuộc tính đều là các biến danh mục (categorical variables) với một tập hợp hạn chế các giá trị khả thi. Tuy nhiên, cột DATE_DIED có số lượng giá trị lớn, có thể là do dữ liệu được thu thập theo thời gian.

```

DATE_DIED
9999-99-99      971633
06/07/2020      1000
07/07/2020       996
13/07/2020       990
16/06/2020       979
...
24/11/2020        1
17/12/2020        1
08/12/2020        1
16/03/2021        1
22/04/2021        1
Name: count, Length: 401, dtype: int64

```

Hình 5. Dữ liệu cột DATE_DIED

Định dạng ngày tháng không thống nhất trong cột DATE_DIED có thể gây khó khăn cho việc phân tích dữ liệu. Cần chuẩn hóa dạng ngày tháng để đảm bảo tính nhất quán. Giá trị thiếu trong cột DATE_DIED có thể ảnh hưởng đến độ chính xác của phân tích dữ liệu. Cần xử lý giá trị thiếu một cách phù hợp, chẳng hạn như loại bỏ các bản ghi có giá trị thiếu hoặc ước tính giá trị thiếu. Cột DATE_DIED có thể được sử dụng để thực hiện phân tích thời gian, chẳng hạn như xác định tỷ lệ tử vong theo thời gian hoặc phân tích thời gian sống.

```

PNEUMONIA
2      892534
1      140038
99      16003
Name: count, dtype: int64

```

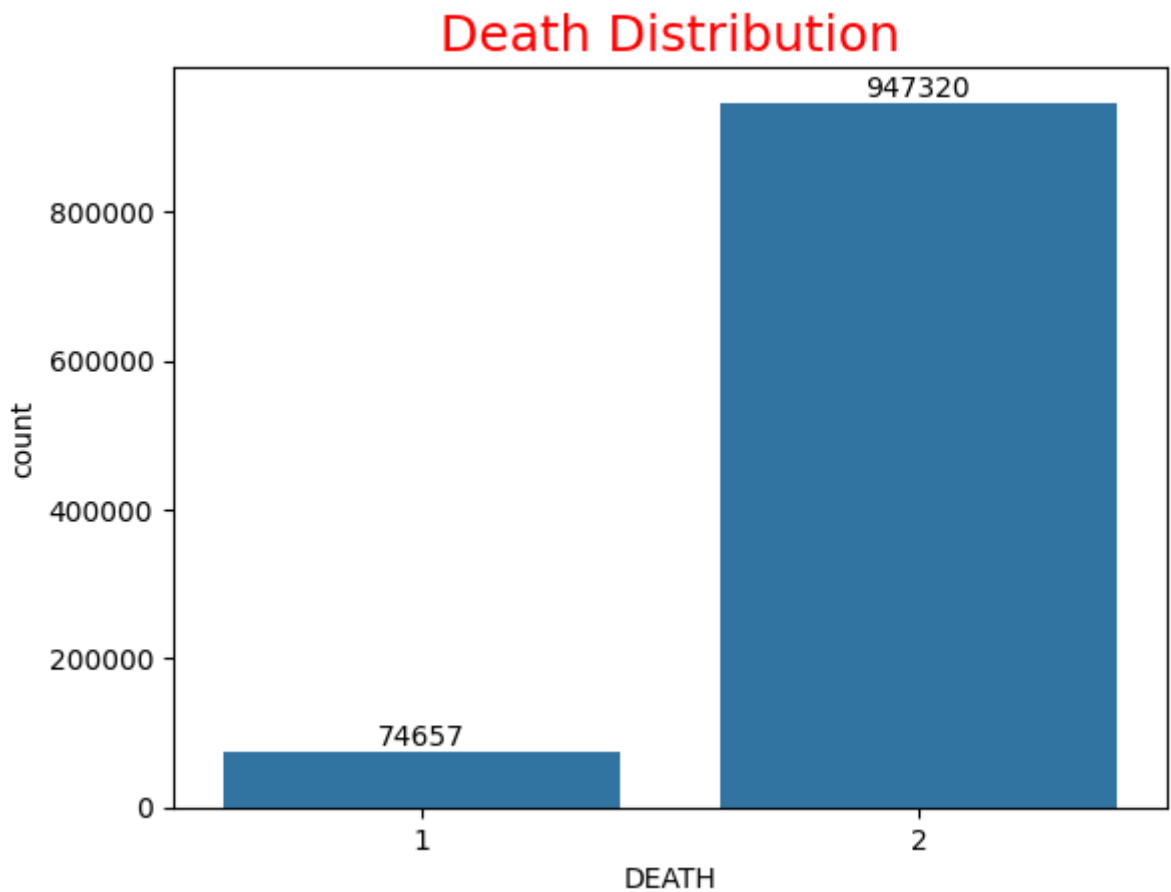
Hình 6. Giá trị thiếu của cột PNEUMONIA

Như đã thấy rằng ở thuộc tính PREGNANT tất cả các giá trị của 98 đều là những giá trị còn thiếu tương ứng với các giá trị nữ, trong khi tất cả các giá trị của 97 đều là

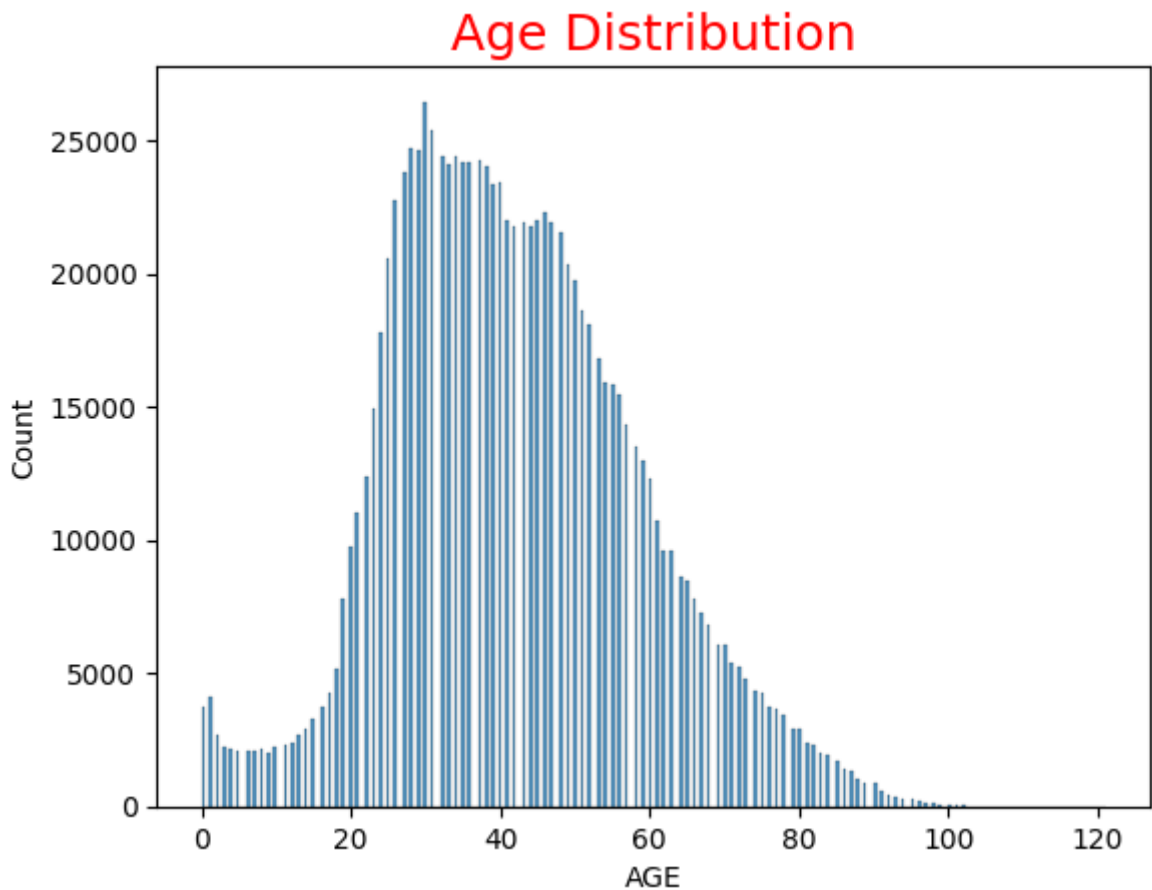
những giá trị tương ứng với các giá trị nam. Vì vậy, rõ ràng có thể thay thế tất cả các giá trị tương ứng ở phần nam (97) bằng (2); vì rõ ràng là đàn ông không thể mang thai.

3. Trực quan hóa dữ liệu

Số người mất do covid trong tập dữ liệu là: 74714 người chiếm khoảng 7,3% tổng số bệnh nhân trong dataset.



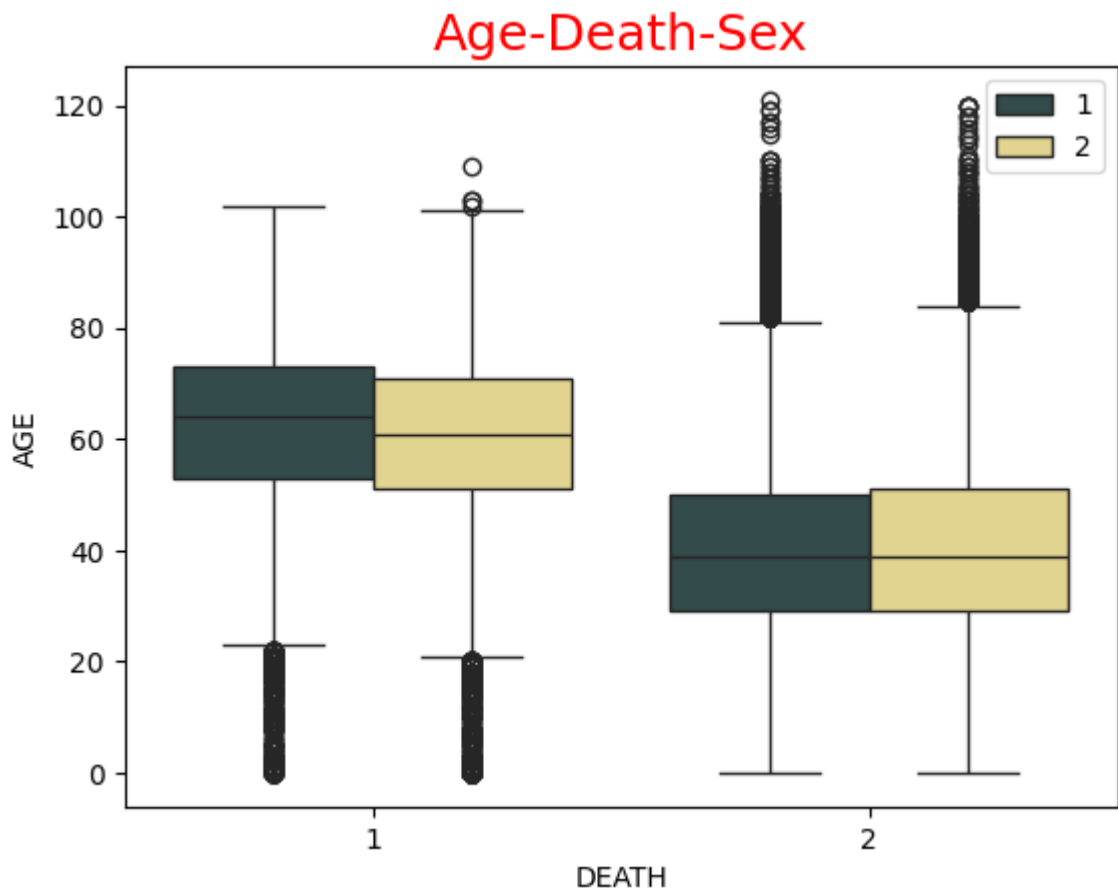
Hình 7. Phân bố số người tử vong của bệnh nhân covid



Hình 8. Biểu đồ phân bố độ tuổi của bệnh nhân covid

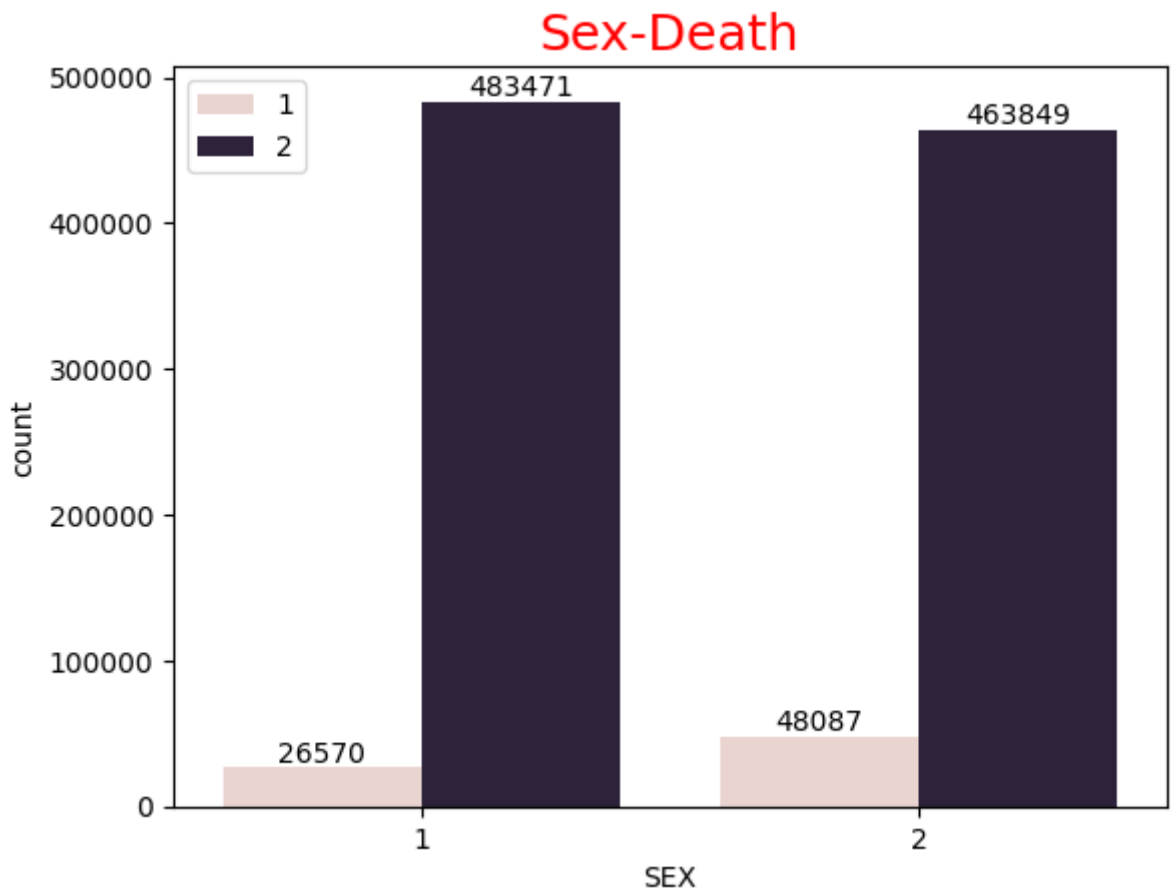
Nhóm tuổi 40-49 có số lượng bệnh nhân cao nhất. Theo biểu đồ, nhóm tuổi 40-49 có số lượng bệnh nhân Covid nhiều hơn các nhóm tuổi khác.

Số lượng bệnh nhân giảm dần theo độ tuổi. Điều này có thể là do trẻ em và thanh thiếu niên có hệ miễn dịch mạnh hơn so với người lớn, giúp họ chống lại virus hiệu quả hơn. Người cao tuổi có nhiều khả năng được tiêm chủng COVID-19 hơn so với các nhóm tuổi khác, giúp họ giảm nguy cơ mắc bệnh.



Hình 9. Biểu đồ số ca tử vong của các độ tuổi ở nam nữ

Nhìn chung, biểu đồ boxplot cho thấy xu hướng tuổi tử vong thấp hơn ở nữ giới so với nam giới. Tuổi trung vị tử vong ở nữ giới là khoảng 60, trong khi ở nam giới là gần 70. Phạm vi dữ liệu, được biểu thị bằng râu của biểu đồ boxplot, lớn hơn ở nam giới so với nữ giới. Điều này cho thấy có tính biến đổi cao hơn trong tuổi tử vong ở nam giới. Có một số ngoại lệ, các điểm dữ liệu nằm ngoài râu, trong cả phân phối nam và nữ. Các ngoại lệ này có thể đại diện cho những cá nhân đã chết ở độ tuổi trẻ hơn hoặc lớn hơn nhiều so với mô hình điển hình. Về các phân đoạn tuổi cụ thể, ở nữ giới, phần lớn các ca tử vong dường như tập trung trong khoảng từ 50 đến 70 tuổi. Có ít ca tử vong trước 50 và sau 70 tuổi. Ở nam giới, phân phối tử vong phân tán hơn, với các ca tử vong xảy ra trong khoảng từ 40 đến 80 tuổi. Có một sự giảm nhẹ về số ca tử vong ở độ tuổi 60.

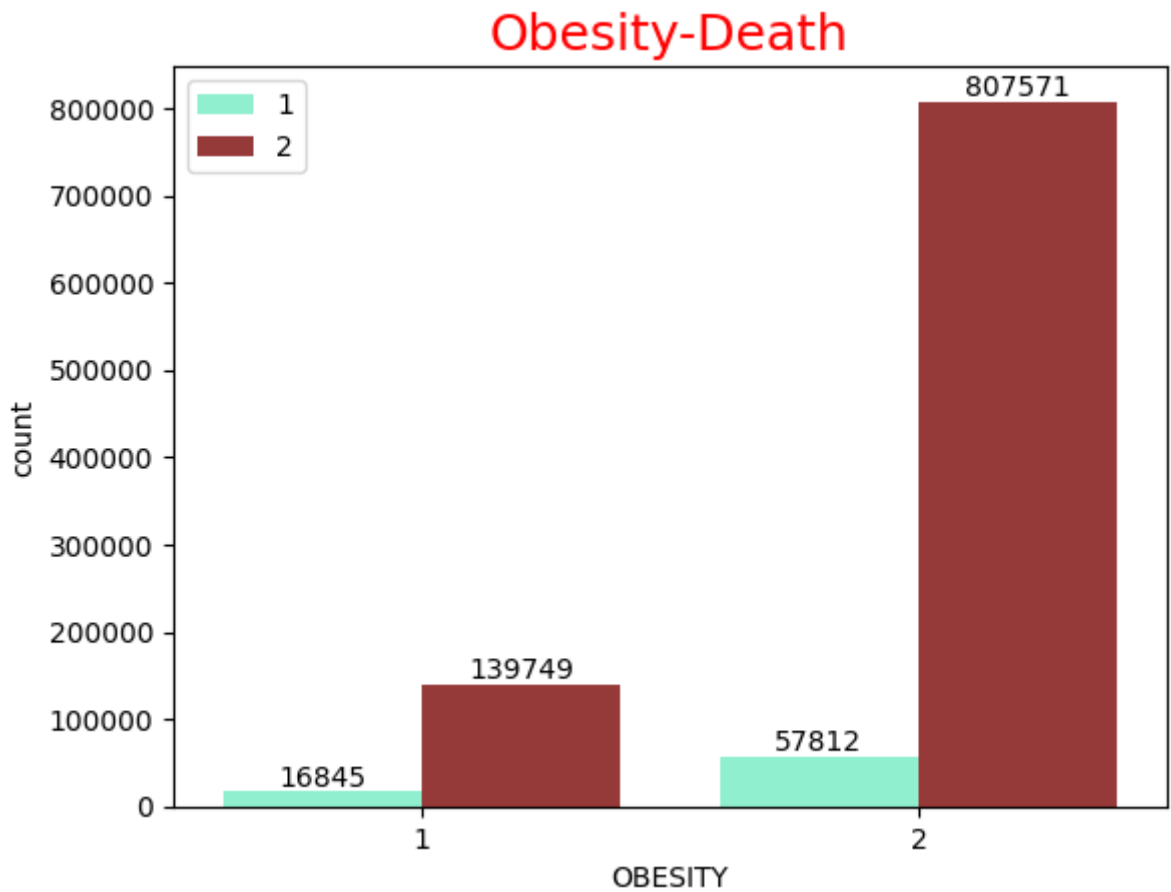


Hình 10. Biểu đồ số ca tử vong ở nam và nữ

Số ca tử vong ở nam giới cao hơn đáng kể so với nữ giới. Theo biểu đồ, số ca tử vong ở nam giới cao gấp đôi so với nữ giới. Có sự chênh lệch lớn về số ca tử vong giữa hai giới tính. Chênh lệch này có thể được giải thích bởi một số yếu tố, bao gồm:

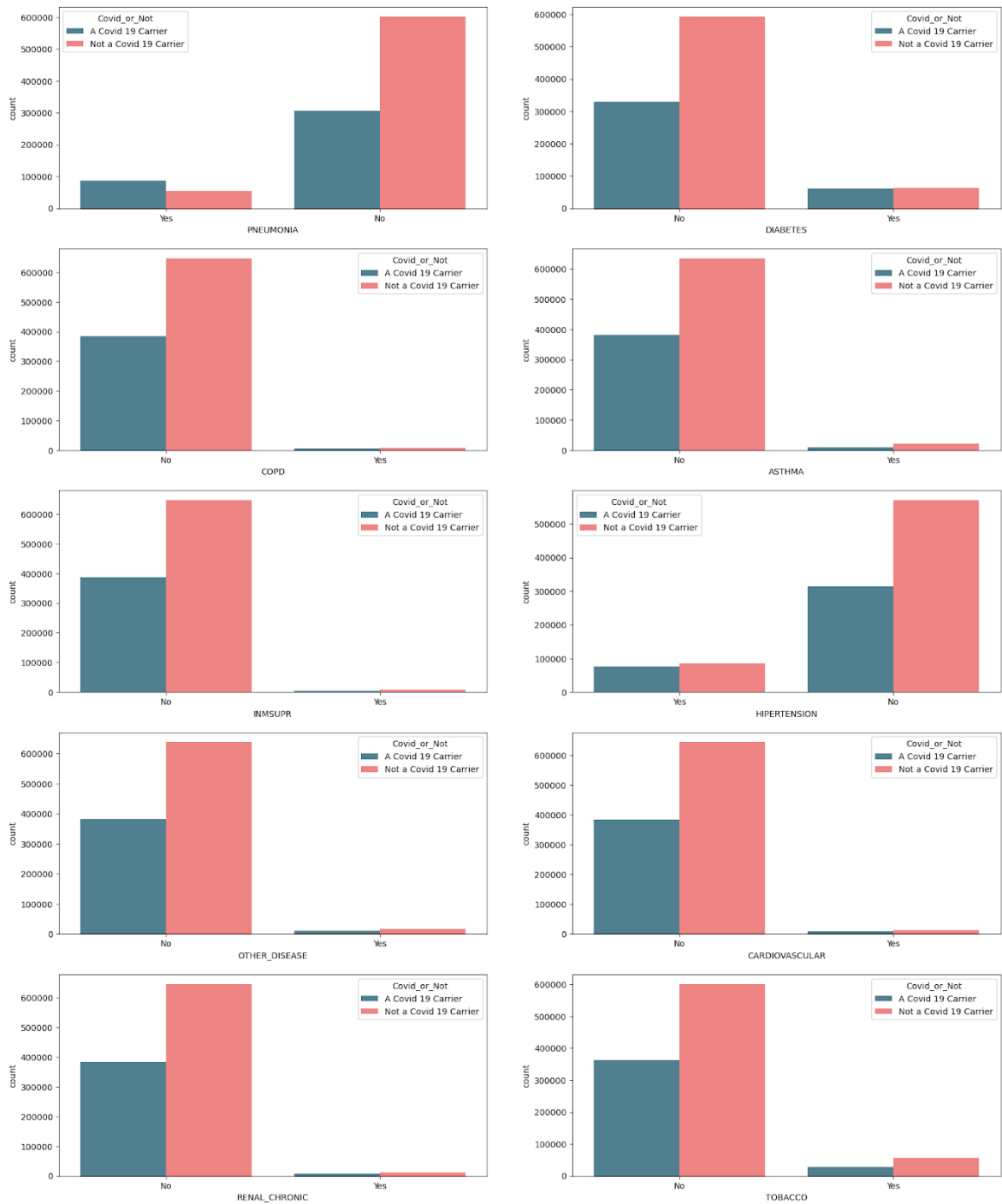
- Sự khác biệt về sinh học: Nam giới có nguy cơ cao hơn bị biến chứng nặng do COVID-19 so với nữ giới.
- Hành vi: Nam giới có nhiều khả năng tham gia vào các hoạt động có nguy cơ cao lây nhiễm COVID-19, chẳng hạn như hút thuốc lá và uống rượu bia.
- Điều trị: Nam giới có thể ít được tiếp cận với dịch vụ chăm sóc sức khỏe hơn nữ giới.

Số ca tử vong ở cả hai giới tính đều có xu hướng giảm dần theo thời gian. Điều này có thể là do các yếu tố sau



Hình 11. Biểu đồ tỉ lệ tử vong của bệnh nhân béo phì

Có thể thấy rằng đối với những bệnh nhân béo phì, tỷ lệ này rất gần nhau mặc dù họ được coi là thiểu số; nhưng điều đó cho chúng ta thấy rằng tỷ lệ bệnh nhân mắc bệnh trong số họ là khá cao, trong khi đối với những bệnh nhân không bị béo phì thì tỷ lệ này là khoảng 1:2, nghĩa là cứ 100 bệnh nhân không bị béo phì chỉ có một nửa trong số họ có cơ hội mang Covid. Vì vậy, theo phân tích của chúng tôi, những người đang bị béo phì có nhiều khả năng mang Covid hơn.



Hình 12. Biểu đồ phân bố các bệnh nền của bệnh nhân covid

Qua trên thấy những bệnh và thói quen sau có tác động lớn nhất: Viêm phổi, Tăng huyết áp, Bệnh tiểu đường, Sử dụng thuốc lá ('Pneumonia', 'Hypertension', 'Diabetes', 'Tobacco usage'). Cũng nhận thấy rằng bệnh nhân Viêm phổi có nhiều khả năng mang Covid hơn với tỷ lệ cao hơn. Và điều này hoàn toàn hợp lý vì ngay từ đầu đây là bệnh về phổi.

4. Những giả thiết khi thu thập dữ liệu

4.1. Trong trường hợp nào thì thu thập được dữ liệu

Bệnh nhân còn sống trong lúc làm nghiên cứu, đầy đủ các chỉ số khi lấy mẫu thử, bộ test phải được bảo quản đúng nơi quy định. Để thu thập dữ liệu từ bài toán dự đoán tỉ lệ tử vong của bệnh nhân COVID-19, cần có sự phối hợp của nhiều bên liên quan.

Các bệnh viện và cơ sở y tế là nơi thu thập dữ liệu trực tiếp từ bệnh nhân COVID-19. Các dữ liệu này bao gồm:

- Thông tin cá nhân: tuổi, giới tính, chủng tộc,...
- Tiền sử bệnh: các bệnh lý nền,...
- Triệu chứng: sốt, ho, khó thở,...
- Kết quả xét nghiệm: xét nghiệm PCR, xét nghiệm kháng thể,...
- Dấu hiệu sinh tồn: nhịp tim, huyết áp,...
- Hình ảnh X-quang phổi, CT-scan phổi,...

Các cơ quan chức năng như Bộ Y tế, Sở Y tế có thể thu thập dữ liệu từ các nguồn sau: số liệu thống kê về dịch bệnh COVID-19, báo cáo của các bệnh viện và cơ sở y tế, kết quả nghiên cứu khoa học.

Các tổ chức nghiên cứu: có thể thu thập dữ liệu từ nguồn các cuộc khảo sát trực tuyến hoặc trực tiếp, các ứng dụng di động, app hoặc website khai báo.

Các dữ liệu này cần được thu thập một cách đầy đủ, chính xác và kịp thời để đảm bảo độ chính xác của các mô hình dự đoán. Cần xác định rõ các thông tin cần thu thập: các thông tin cần thu thập phải liên quan đến nguy cơ tử vong của bệnh nhân COVID-19. Cần sử dụng các phương pháp thu thập dữ liệu phù hợp đảm bảo tính chính xác và đầy đủ của dữ liệu. Cần bảo mật dữ liệu để tránh bị sử dụng cho các mục đích trái phép.

Việc thu thập dữ liệu cho bài toán dự đoán tỉ lệ tử vong của bệnh nhân COVID-19 là một công việc phức tạp và cần có sự phối hợp của nhiều bên liên quan. Tuy nhiên, việc thu thập dữ liệu đầy đủ và chính xác sẽ giúp xây dựng các mô hình dự

đoán có độ chính xác cao, từ đó giúp các bác sĩ đưa ra các quyết định điều trị kịp thời và hiệu quả, cải thiện khả năng sống sót của bệnh nhân.

4.2. Trong trường hợp nào không thể thu thập được dữ liệu

Bệnh nhân không còn sống trong lúc làm nghiên cứu, thiếu các chỉ số khi lấy mẫu thử, bộ test không được bảo quản đúng nơi quy định. Việc xác định các trường hợp không thể thu thập dữ liệu là cần thiết để đảm bảo độ chính xác của các mô hình dự đoán. Không thể thu thập dữ liệu cho bài toán dự đoán tỉ lệ tử vong của bệnh nhân COVID-19 trong các trường hợp sau:

- Bệnh nhân không được nhập viện hoặc không được điều trị y tế: trong trường hợp này, bệnh nhân sẽ không có dữ liệu về tình trạng sức khỏe, kết quả xét nghiệm, và quá trình điều trị.
- Bệnh nhân tử vong ngay sau khi nhiễm bệnh: trong trường hợp này, bệnh nhân sẽ không có dữ liệu về tình trạng sức khỏe và quá trình điều trị.
- Bệnh nhân tử vong do các nguyên nhân khác: trong trường hợp này, dữ liệu về tình trạng sức khỏe và quá trình điều trị của bệnh nhân không liên quan đến nguy cơ tử vong do COVID-19.
- Dữ liệu không được thu thập một cách đầy đủ, chính xác và kịp thời: trong trường hợp này, các mô hình dự đoán sẽ không có độ chính xác cao
- Vấn đề về quyền riêng tư: một số bệnh nhân có thể không đồng ý cung cấp dữ liệu cá nhân của họ.
- Vấn đề về chi phí: việc thu thập dữ liệu có thể tốn kém, đặc biệt là trong trường hợp cần thu thập dữ liệu từ nhiều bệnh nhân.
- Vấn đề về kỹ thuật: việc thu thập dữ liệu có thể gặp khó khăn trong các trường hợp bệnh nhân ở vùng sâu, vùng xa hoặc trong các điều kiện đặc biệt.

III. Khai phá dữ liệu

1. Quy trình khai phá dữ liệu

Bước 1: Nhập các thư viện cần thiết, Nhập bộ dữ liệu tử vong bệnh nhân Covid.

Bước 2: Xử lý trước dữ liệu để loại bỏ dữ liệu bị thiếu.

Bước 3: Thực hiện phân chia tỷ lệ phần trăm 80% để chia tập dữ liệu thành tập huấn luyện và 20% cho tập kiểm tra.

Bước 4: Chọn thuật toán học máy

Bước 5: Xây dựng mô hình phân loại cho thuật toán học máy đã đề cập dựa trên tập huấn luyện.

Bước 6: Kiểm tra mô hình trình phân loại cho thuật toán học máy đã đề cập dựa trên tập kiểm tra.

Bước 7: Thực hiện so sánh đánh giá các kết quả hoạt động thử nghiệm thu được đối với mỗi bộ phân loại.

Bước 8: Thực hiện điều chỉnh tham số từ tham số mặc định trong mô hình phân loại dựa trên các biện pháp khác nhau.

2. Nguyên lý hoạt động của thuật toán Logistic Regression

Thuật toán Logistic Regression (Hồi quy Logistic) là một thuật toán học máy được sử dụng phổ biến cho các bài toán phân loại nhị phân. Nó hoạt động bằng cách ước tính xác suất một điểm dữ liệu thuộc về một trong hai lớp. Thuật toán này sử dụng một hàm sigmoid để chuyển đổi kết quả tuyến tính thành xác suất, nằm trong khoảng 0 và 1.

2.1. Thu thập dữ liệu

Thu thập dữ liệu tập huấn luyện bao gồm các điểm dữ liệu và nhãn tương ứng của chúng. Mỗi điểm dữ liệu được biểu diễn dưới dạng một vector các thuộc tính, và nhãn cho biết điểm dữ liệu đó thuộc về lớp nào (ví dụ: 0 hoặc 1).

2.2. Khởi tạo trọng số

Khởi tạo các giá trị trọng số ban đầu cho mô hình. Các trọng số này đại diện cho mức độ ảnh hưởng của mỗi thuộc tính đến kết quả dự đoán.

Với dòng thứ i trong dữ liệu, gọi $x^{(i)}$ là các thuộc tính của các bệnh nhân thứ i . $P(x^{(i)} = 1) = \hat{y}_i$ là xác suất model dự đoán bệnh nhân covid có nguy cơ tử vong. $P(x^{(i)} = 0) = 1 - \hat{y}_i$ là xác suất model dự đoán bệnh nhân covid không có nguy cơ tử vong.

2.3. Tính toán với các giá trị dự đoán

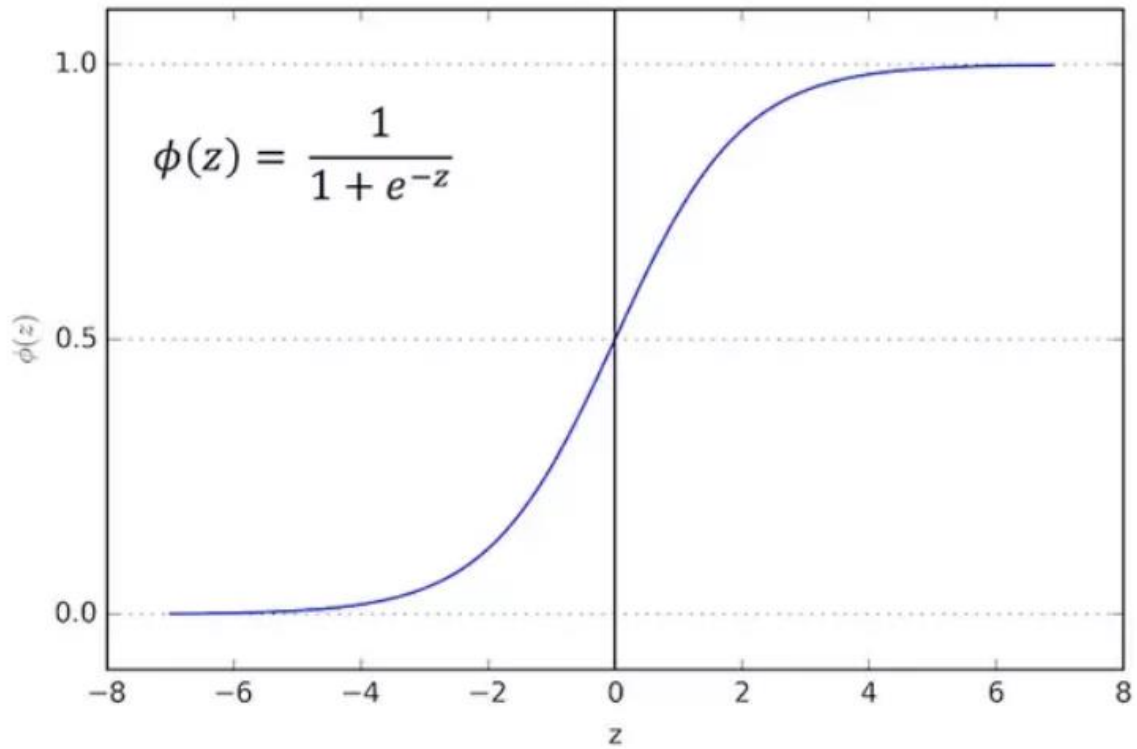
Đối với mỗi điểm dữ liệu trong tập huấn luyện, tính toán giá trị dự đoán bằng cách sử dụng hàm sigmoid trên tích vô hướng giữa vector thuộc tính của điểm dữ liệu và vector trọng số. Giá trị dự đoán nằm trong khoảng 0 và 1, đại diện cho xác suất điểm dữ liệu đó thuộc về lớp 1.

2.4. Tính toán hàm mất mát

Sử dụng hàm mất mát để đánh giá mức độ sai lệch giữa giá trị dự đoán và nhãn thực tế. Hàm mất mát phổ biến cho Logistic Regression là hàm mất mát Entropy chéo nhị phân.

$$L = -(y_i * \log(y_i) + (1 - y_i) * \log(1 - y_i))$$

Với hàm sigmoid là : $\sigma(x) = \frac{1}{1 + e^{-x}}$



Hình 13. Ảnh minh họa đồ thị của hàm sigmoid

Nguồn: Sưu tầm trên mạng

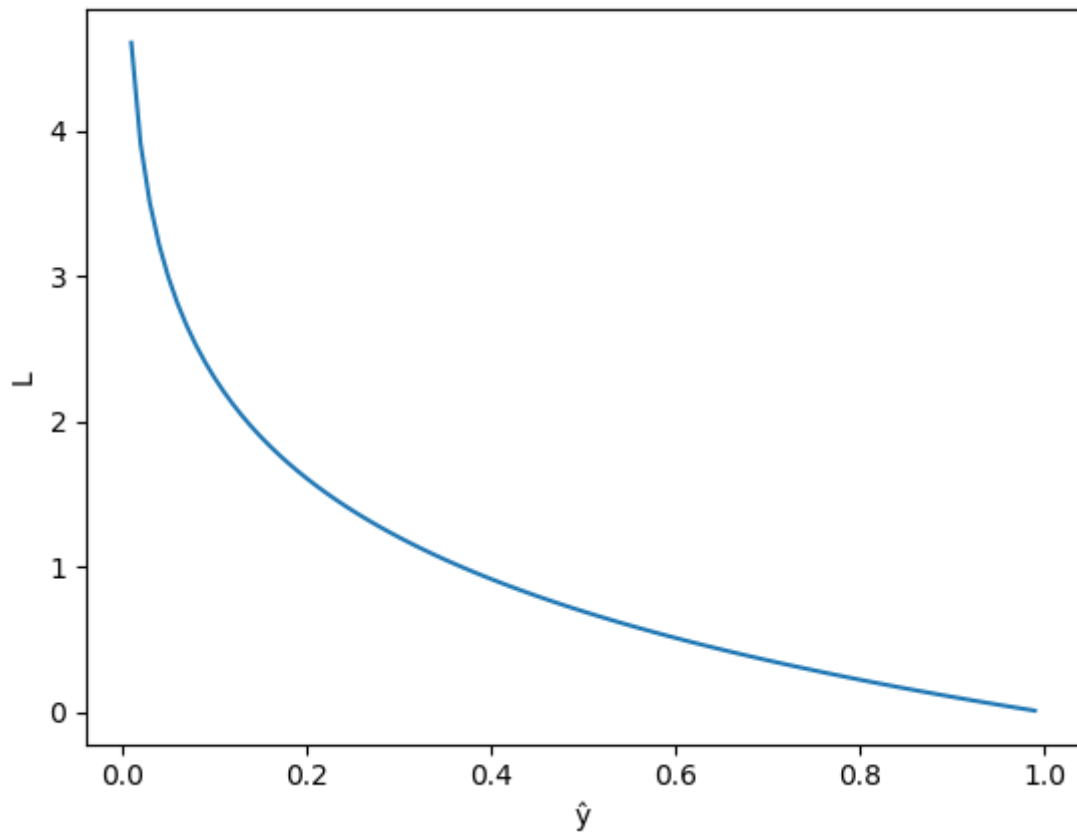
2.5. Cập nhật trọng số

Sử dụng thuật toán tối ưu hóa, như gradient descent, để điều chỉnh các giá trị trọng số nhằm giảm thiểu hàm mất mát. Quá trình này được lặp lại cho đến khi tìm được bộ trọng số tối ưu cho mô hình. Dưới đây là Gradient Descent:

$$\begin{aligned}\frac{dL}{dw_1} &= \sum_{i=1}^N x_1^{(i)} * (\hat{y}_i - y_i) \\ \frac{dL}{dw_0} &= \sum_{i=1}^N (\hat{y}_i - y_i) \\ \frac{dL}{dw_2} &= \sum_{i=1}^N x_2^{(i)} * (\hat{y}_i - y_i)\end{aligned}\tag{III. 1}$$

2.6. Dự đoán cho dữ liệu mới

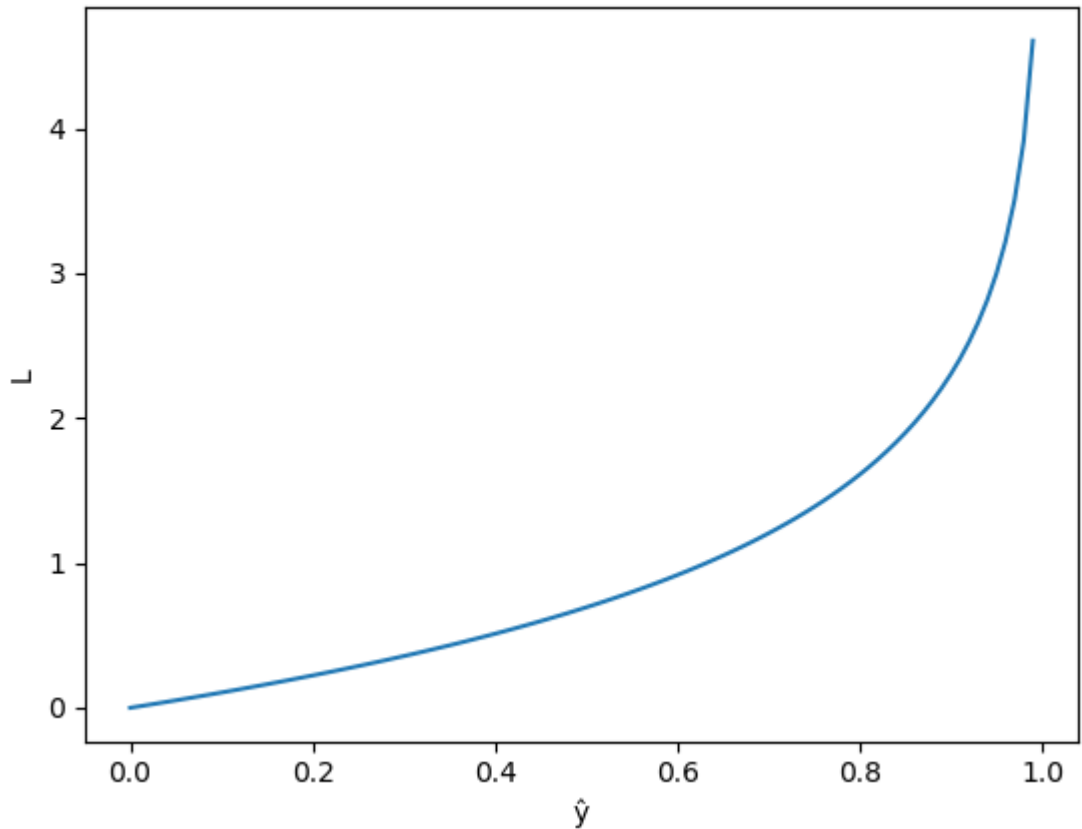
Sử dụng mô hình đã được huấn luyện để dự đoán nhãn cho dữ liệu mới. Đối với mỗi điểm dữ liệu mới, tính toán giá trị dự đoán bằng cách sử dụng hàm sigmoid trên tích vô hướng giữa vector thuộc tính của điểm dữ liệu và vector trọng số. Sau đó, gán nhãn cho điểm dữ liệu dựa trên ngưỡng xác suất đã được xác định trước.



Hình 14. Đồ thị dự đoán của model là đúng

Nguồn: Sưu tầm trên mạng

Nhận xét: Nếu $y_i = 1 \Rightarrow L = -\log(y_i)$. Hàm L giảm từ 0 đến 1. Khi model dự đoán $y_i = 1$, tức giá trị dự đoán gần với giá trị thật y_i thì L nhỏ, xấp xỉ 0. Khi model dự đoán $y_i = 0$, tức giá trị dự đoán ngược lại với giá trị thật y_i thì L rất lớn.



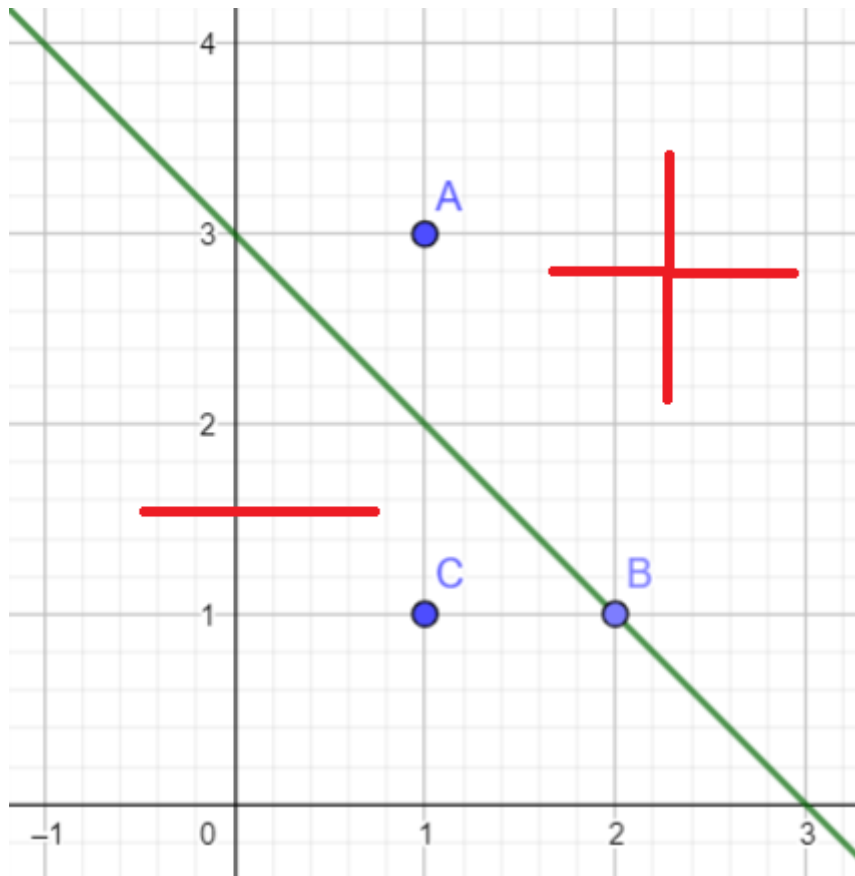
Hình 15. Đồ thị dự đoán của model là sai

Nguồn: Sưu tầm trên mạng

Ngược lại, nếu $y_i = 0 \Rightarrow L = -\log(1 - y_i)$. Hàm L tăng từ 0 đến 1. Khi model dự đoán $y_i = 0$, tức giá trị dự đoán gần với giá trị thật y_i thì L nhỏ, xấp xỉ 0. Khi model dự đoán $y_i = 1$, tức giá trị dự đoán ngược lại với giá trị thật y_i thì L rất lớn.

Hàm L nhỏ khi giá trị model gần với giá trị thật và rất lớn khi model dự đoán sai, hay nói cách khác L càng nhỏ thì model dự đoán càng gần với giá trị thật. => Bài toán toán quy về tìm giá trị nhỏ nhất của L.

Từ đó xây dựng nên đường phân chia để đưa ra dự đoán của dữ liệu. Và đường phân chia này là đường thẳng $y = ax + b$ và không phải là duy nhất của model. Vì mỗi lần trọng số w thay đổi sẽ lặp lại quá trình này đến khi tìm ra đường thẳng chia dữ liệu ra làm 2 phần.



Hình 16. Ảnh minh họa đường phân chia của Logistic Regression

3. Tiêu chí đánh giá mô hình

Đánh giá mô hình giúp lựa chọn được các mô hình phù hợp với bài toán. Để đánh giá được một cách đa dạng và tìm hiểu sâu hơn về mô hình bài nghiên cứu sử dụng các công cụ đánh giá phổ biến như: Confusion matrix, Accuracy, Precision, Recall, F1-score, đường cong ROC. Từ đó có thể áp dụng đúng thước đo đánh giá mô hình phù hợp.

Một thuật ngữ cơ bản được sử dụng trong các bài toán phân loại – Confusion matrix (AKA error matrix). Mặc dù không phải là một metric nhưng rất quan trọng, nó thể hiện được có bao nhiêu điểm dữ liệu thực sự thuộc vào một class, và được dự đoán là rơi vào một class. True/False ý chỉ những gì ta đã dự đoán là đúng hay chưa. Positive/Negative chỉ những gì ta dự đoán (có hoặc không) Nói cách khác, nếu thấy chữ True tức là dự đoán là đúng (là cat hay non-cat, chỉ cần đúng), còn False thì ngược lại.

Accuracy là độ đo của bài toán phân loại mà đơn giản nhất, tính toán bằng cách lấy số dự đoán đúng chia cho toàn bộ các dự đoán. Nhược điểm của cách đánh giá này là chỉ cho ta biết được bao nhiêu phần trăm lượng dữ liệu được phân loại đúng mà không chỉ ra được cụ thể mỗi loại được phân loại như thế nào, lớp nào được phân loại đúng nhiều nhất hay dữ liệu của lớp nào thường bị phân loại nhầm nhất vào các lớp khác.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (III. 2)$$

Như đã nói phía trên, sẽ có rất nhiều trường hợp thước đo Accuracy không phản ánh đúng hiệu quả của mô hình. Vì vậy cần một metric có thể khắc phục được những yếu điểm này. Precision là một trong những metrics có thể khắc phục được. Precision sẽ cho biết thực sự có bao nhiêu dự đoán Positive là thật sự True, công thức như sau:

$$Precision = \frac{TP}{TP + FP} \quad (III. 3)$$

Recall cũng là một metric quan trọng, nó đo lường tỷ lệ dự báo chính xác các trường hợp positive trên toàn bộ các mẫu thuộc nhóm positive. Công thức của Recall như sau:

$$Recall = \frac{TP}{TP + FN} \quad (III. 4)$$

Tùy thuộc vào bài toán mà sẽ muốn ưu tiên sử dụng Recall hay Precision. Nhưng cũng có rất nhiều bài toán mà cả Precision hay Recall đều quan trọng. Một metric phổ biến đã kết hợp cả Recall và Precision lại được gọi là F1-score, công thức như sau:

$$F1\text{-score} = \frac{2 * Precision * Recall}{Precision + Recall} \quad (III. 5)$$

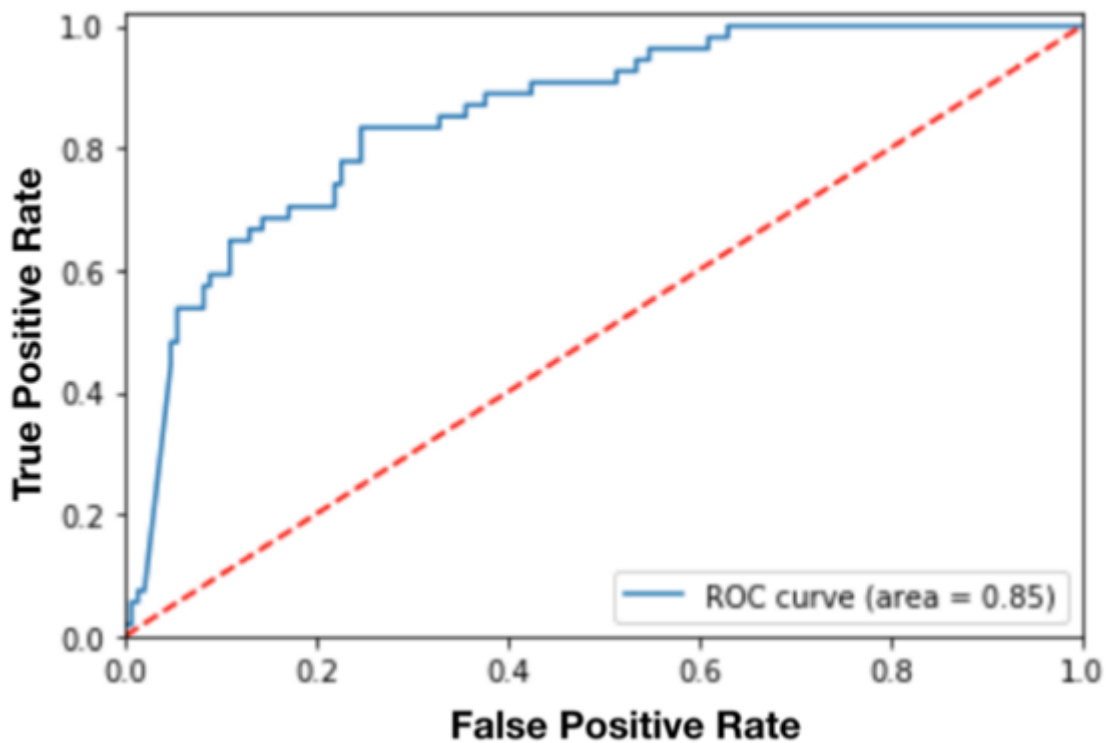
AUC (Area Under the Curve) là một phép đo tổng hợp về hiệu suất của phân loại nhị phân trên tất cả các giá trị ngưỡng có thể có. Để hiểu rõ hơn về metric này cần tìm hiểu về một khái niệm cơ sở trước, đó là ROC Curve.

ROC Curve (The receiver operating characteristic curve) là một đường cong biểu diễn hiệu suất phân loại của một mô hình phân loại tại các ngưỡng threshold. Về cơ bản, nó hiển thị True Positive Rate (TPR) so với False Positive Rate (FPR) đối với các giá trị ngưỡng khác nhau. Các giá trị TPR, FPR được tính như sau:

$$TPR = \frac{TP}{TP + FP} \quad (III. 6)$$

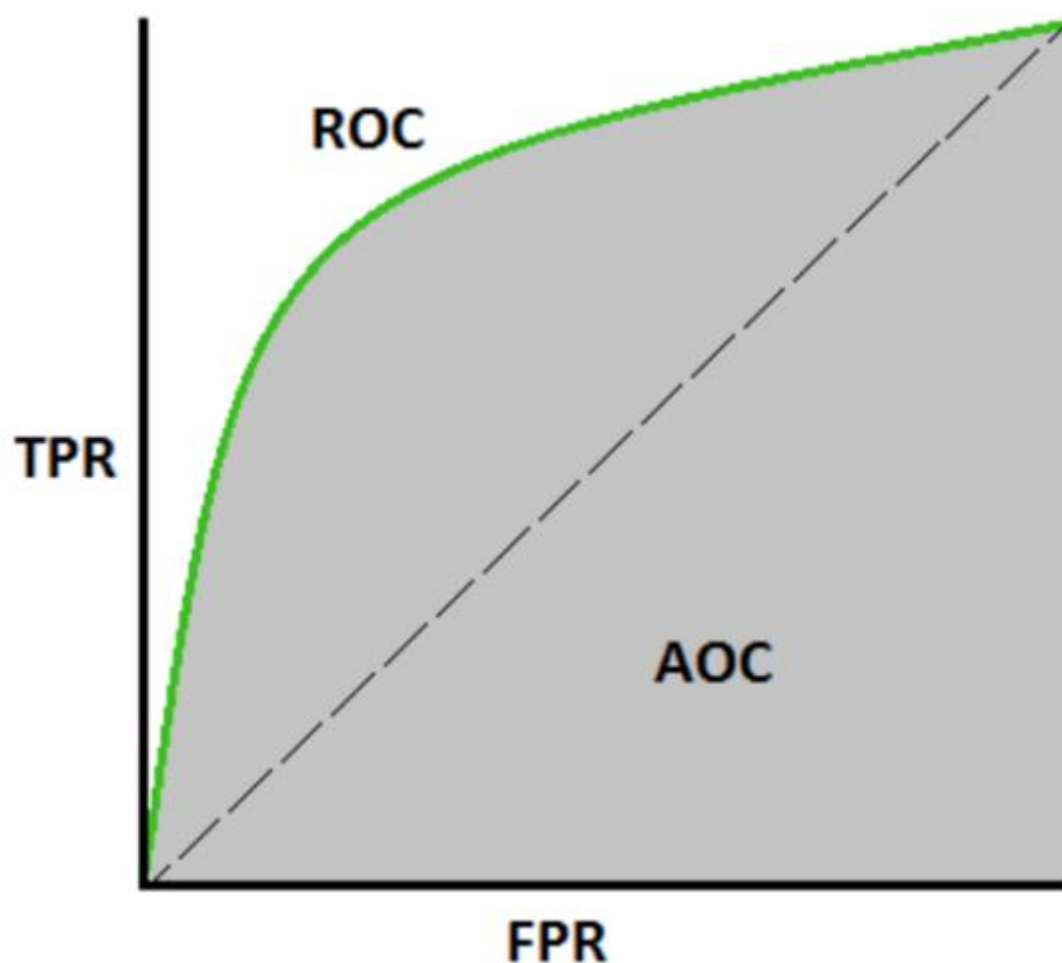
$$FPR = \frac{FP}{TN + FN} \quad (III. 7)$$

ROC tìm ra TPR và FPR ứng với các giá trị ngưỡng khác nhau và vẽ biểu đồ để dễ dàng quan sát TPR so với FPR. Ví dụ dưới đây là một đường cong ROC.



Hình 17. Biểu đồ minh họa đánh giá mô hình của độ đo ROC

AUC là chỉ số được tính toán dựa trên đường cong ROC nhằm đánh giá khả năng phân loại của mô hình tốt như thế nào. Phần diện tích nằm dưới đường cong ROC và trên trục hoành chính là AUC, có giá trị nằm trong khoảng $[0, 1]$.



Hình 18. Biểu diễn của đường cong ROC

Khi diện tích này càng lớn, đường cong này sẽ dần tiệm cận với đường thẳng $y = 1$ tương đương với khả năng phân loại của mô hình càng tốt. Còn khi đường cong ROC nằm sát với đường chéo đi qua hai điểm $(0, 0)$ và $(1, 1)$, mô hình sẽ tương đương với một phân loại ngẫu nhiên.

IV. Chọn thuật toán và đánh giá

1. Yêu cầu về chương trình

1.1. Lựa chọn thuật toán Machine Learning

Nếu sử dụng thuật toán hồi quy truyền thống để dự đoán tỉ lệ tử vong, mô hình có thể dự đoán tỉ lệ tử vong là một giá trị liên tục. Tuy nhiên, tỉ lệ tử vong là một giá trị rời rạc, có thể là 0 hoặc 1. Do đó, dự đoán của mô hình có thể không chính xác hoặc không có ý nghĩa thực tế.

Một giải pháp tốt hơn là sử dụng một thuật toán phân loại, chẳng hạn như Logistic Regression. Logistic Regression là một thuật toán phân loại dựa trên giả định rằng mối quan hệ giữa các biến độc lập và biến phụ thuộc là tuyến tính. Tuy nhiên, thuật toán này có thể được sử dụng để dự đoán các biến rời rạc. Trong trường hợp này, mô hình Logistic Regression có thể dự đoán tỉ lệ tử vong là 0 hoặc 1. Dự đoán của mô hình này có thể chính xác hơn và có ý nghĩa thực tế hơn so với dự đoán của mô hình hồi quy truyền thống. Vậy cho nên đối với bài toán đã nêu trên, bài nghiên cứu sẽ sử dụng Logistic Regression làm thuật toán để huấn luyện mô hình.

1.2. Import các thư viện cần thiết

```
# Read dataset
import pandas as pd
import os
for dirname, _, filenames in os.walk('/Covid Data.csv'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
# Visualization
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# Preprocessing
from sklearn.preprocessing import RobustScaler
```

```
# Training model
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import
GradientBoostingClassifier, RandomForestClassifier
from sklearn.svm import SVC
```

```
# Model evaluation
from sklearn.metrics import accuracy_score, f1_score,
classification_report, confusion_matrix, roc_curve
```

```
# Data balancing
from imblearn.under_sampling import RandomUnderSampler
```

2. Tiền xử lý dữ liệu

2.1. Mã hóa dữ liệu

Loại bỏ các giá trị thiếu dữ liệu trong các cột thuộc tính đã xác định ở trên, ngoại trừ các cột INTUBED, PREGNANT, ICU.

```
# Getting rid of the missing values of features except
"INTUBED", "PREGNANT", "ICU"

cols = ['PNEUMONIA', 'DIABETES', 'COPD', 'ASTHMA',
'INMSUPR', 'HIPERTENSION', 'OTHER_DISEASE',
'CARDIOVASCULAR', 'OBESITY', 'RENAL_CHRONIC', 'TOBACCO']
for i in cols :
    df = df[(df[i] == 1) | (df[i] == 2)]
```

Mã hóa cột 'DATE_DIED' thành cột nhị phân 'DEATH'. Những giá trị nào chưa biết ngày tử vong của bệnh nhân tức là "9999-99-99" thì sẽ là 1, còn đã biết là 2.

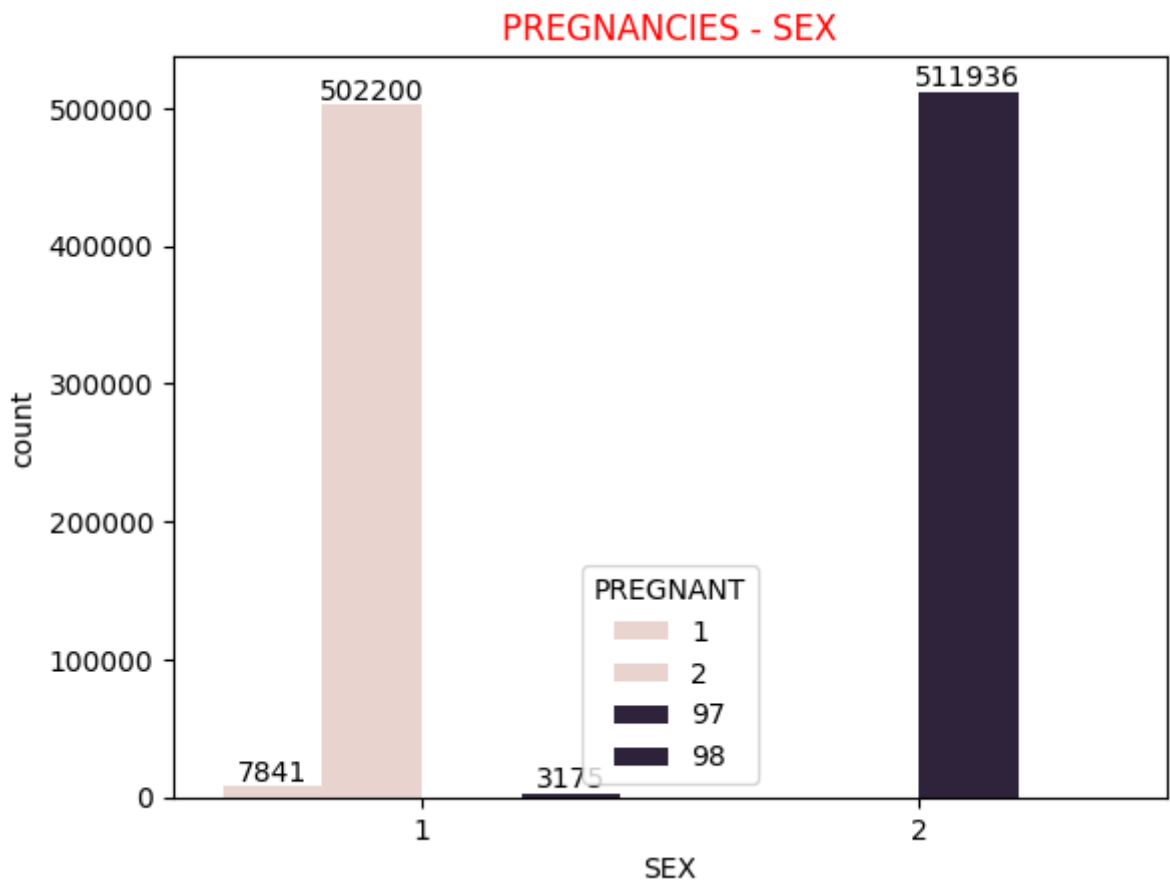
```
# Preparing "DATE_DIED" column
```

```
df['DEATH'] = [2 if row == "9999-99-99" else 1 for row in
df.DATE_DIED]
df['DEATH'].value counts()
```

```
DEATH
2    950438
1     74714
Name: count, dtype: int64
```

Hình 19. Mã hóa cột DATE_DIED

2.2. Xử lý giá trị bị thiếu



Hình 20. Số bệnh nhân covid có thai

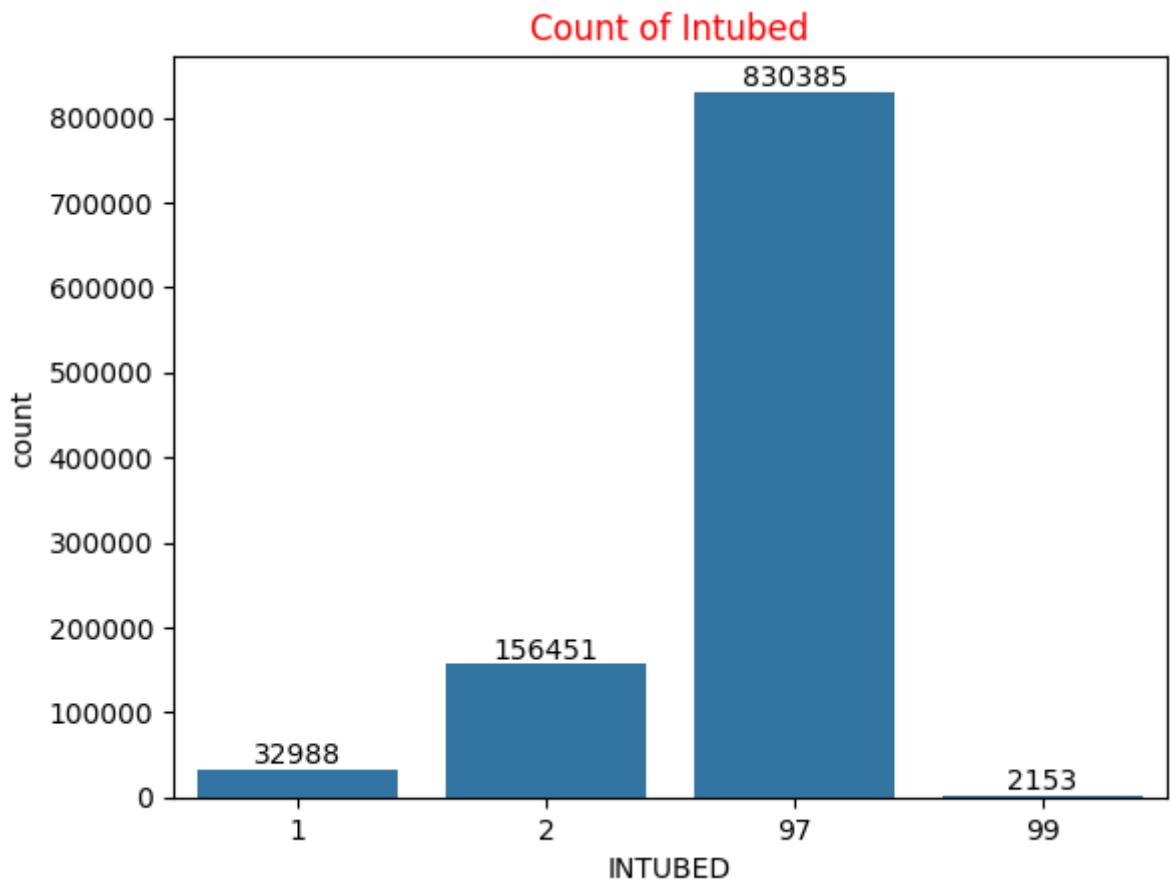
Như đã trình bày ở trên chỉ có phụ nữ mới mang thai nên có thể thay thế tất cả các giá trị tương ứng ở phần nam (97) bằng (2) vì rõ ràng là đàn ông không thể mang thai.

```
# Converting process according to inference above
```



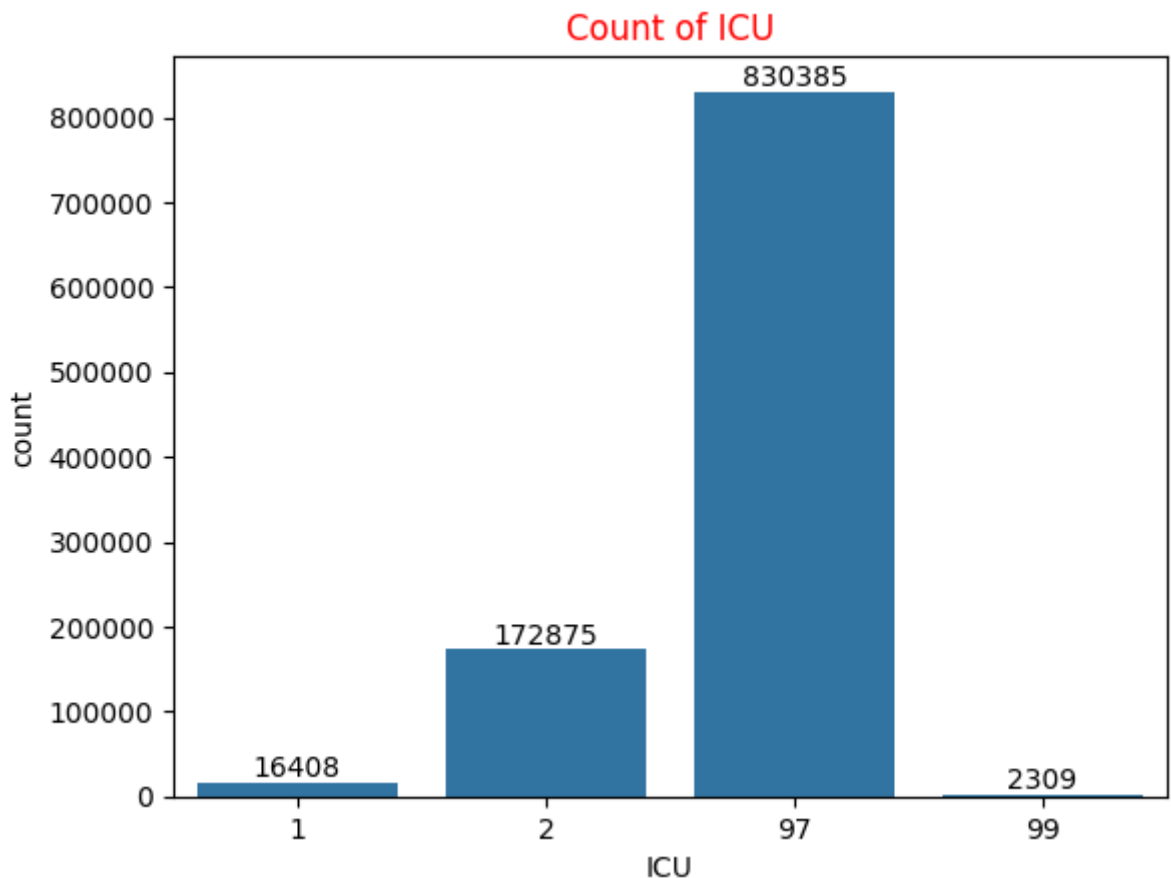
```
df.PREGNANT = df.PREGNANT.replace(97,2)

# Getting rid of the missing values
df = df[(df.PREGNANT == 1) | (df.PREGNANT == 2)]
```



Hình 21. Số bệnh nhân có kết nối máy thở

Như biểu đồ trên cho thấy những bệnh nhân kết nối với máy thở có nhiều dữ liệu chưa có là 97 và 99, và cũng thấy được số bệnh nhân được kết nối máy thở ít hơn rất nhiều so với nhóm không có. Vì thế nếu để có thể gây ảnh hưởng đến huấn luyện mô hình.



Hình 22. Số bệnh nhân vào Khoa Hồi sức

Giống như INTUBED nhiều dữ liệu chưa có là 97 và 99, và cũng thấy được số có hồi sức ít hơn rất nhiều so với nhóm không có. Vì thế nếu để cho mô hình mang tính chính xác hơn những thuộc tính bị thiếu nhiều dữ liệu và mất cân bằng như thế sẽ bỏ đi để huấn luyện mô hình. Vậy sẽ bỏ đi 3 cột là INTUBED, ICU, DATE_DIED.

```
# Dropping the columns
df.drop(columns=["INTUBED", "ICU", "DATE_DIED"],
inplace=True)
```

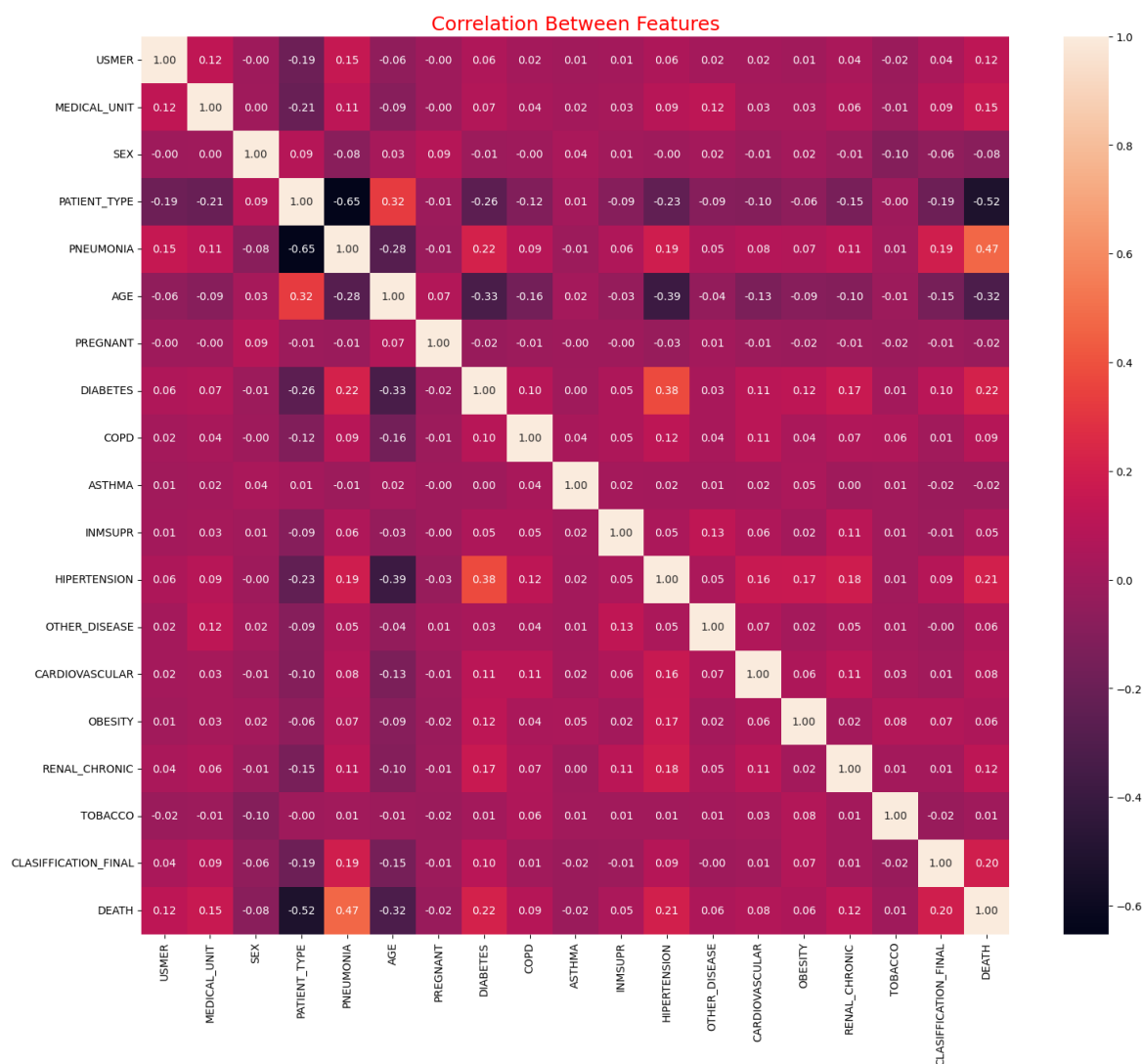
2.3. Thống kê dữ liệu

Sau khi đã mã hóa xong những đặc trưng thì đây là thống kê của mẫu dữ liệu

USMER	=>	2
MEDICAL_UNIT	=>	13
SEX	=>	2
PATIENT_TYPE	=>	2
PNEUMONIA	=>	2
AGE	=>	121
PREGNANT	=>	2
DIABETES	=>	2
COPD	=>	2
ASTHMA	=>	2
INMSUPR	=>	2
HIPERTENSION	=>	2
OTHER_DISEASE	=>	2
CARDIOVASCULAR	=>	2
OBESITY	=>	2
RENAL_CHRONIC	=>	2
TOBACCO	=>	2
CLASIFFICATION_FINAL	=>	7
DEATH	=>	2

Hình 23. Thống kê dữ liệu mã hóa

Và trước khi đem mẫu đi training cần xem mối tương quan của các đặc trưng có cao hay không từ đó sẽ rút kết được những đặc trưng quan trọng. Mặc dù trong y tế thì cần phải có kiến thức về lĩnh vực này và từ những kinh nghiệm của các bác sĩ có chuyên môn. Và khi xem qua heatmap cũng để đánh giá thực tế xem những đặc trưng khi thu thập so với thực tế có khác nhau quá nhiều không.



Hình 24. Mối tương quan của các đặc trưng

Mối tương quan giữa DEATH và USMER (tỷ lệ tử vong do nguyên nhân cụ thể) là tích cực. Điều này có nghĩa là khi tỷ lệ tử vong do nguyên nhân cụ thể (USMER) tăng, tỷ lệ tử vong chung (DEATH) cũng có xu hướng tăng.

Mối tương quan giữa DEATH và SEX là phân bố. Điều này có nghĩa là tỷ lệ tử vong chung (DEATH) khác nhau giữa nam và nữ. Cụ thể, tỷ lệ tử vong chung ở nam giới cao hơn so với nữ giới.

Mối tương quan giữa DEATH và PATIENT TYPE là phân bố. Điều này có nghĩa là tỷ lệ tử vong chung (DEATH) khác nhau giữa các loại bệnh nhân khác nhau. Cụ thể, tỷ lệ tử vong chung ở bệnh nhân nội trú cao hơn so với bệnh nhân ngoại trú.

Mối tương quan giữa DEATH và các đặc trưng khác như PNEUMONIA (viêm phổi), AGE (tuổi), PREGNANT (mang thai), DIABETES (tiểu đường), COPD (bệnh phổi tắc nghẽn mãn tính), ASTHMA (hen suyễn), INMSUPR (suy miễn dịch), HIPERTENSION (cao huyết áp), OTHER DISEASE (bệnh khác), CARDIOVASCULAR (tim mạch), OBESITY (béo phì), RENAL CHRONIC (suy thận mãn tính), TOBACCO (thuốc lá) và CLASIFFICATION_FINAL (chẩn đoán cuối cùng) là phân bố.

Điều này có nghĩa là tỷ lệ tử vong chung (DEATH) khác nhau giữa các giá trị khác nhau của các đặc trưng này. Cụ thể, tỷ lệ tử vong chung cao hơn ở những người có các đặc trưng như viêm phổi, tuổi cao, mang thai, tiểu đường, bệnh phổi tắc nghẽn mãn tính, hen suyễn, suy miễn dịch, cao huyết áp, bệnh khác, bệnh tim mạch, béo phì, suy thận mãn tính, hút thuốc lá.

Dựa trên các phân tích trên, có thể thấy mối tương quan giữa DEATH và các đặc trưng khác là phức tạp và đa dạng. Tỷ lệ tử vong chung (DEATH) bị ảnh hưởng bởi nhiều yếu tố, bao gồm tỷ lệ tử vong do nguyên nhân cụ thể (USMER), giới tính (SEX), loại bệnh nhân (PATIENT TYPE), và các đặc trưng khác như PNEUMONIA (viêm phổi), AGE (tuổi), PREGNANT (mang thai), DIABETES (tiểu đường), COPD (bệnh phổi tắc nghẽn mãn tính), ASTHMA (hen suyễn), INMSUPR (suy miễn dịch), HIPERTENSION (cao huyết áp), OTHER DISEASE (bệnh khác), CARDIOVASCULAR (tim mạch), OBESITY (béo phì), RENAL CHRONIC (suy thận mãn tính), TOBACCO (thuốc lá) và CLASIFFICATION_FINAL (chẩn đoán cuối cùng).

2.4. Lựa chọn các đặc trưng

Từ những thống kê trên thông qua Heatmap (Hình 24) đã cho thấy mối tương quan cao hay thấp của biến mục tiêu là DEATH so với các biến độc lập khác. Và tiến hành huấn luyện mô hình trước hết sẽ cần bỏ đi những mối tương quan thấp không có mối liên hệ chặt chẽ so với cột DEATH.

```
# Dropping the features that have low correlation with  
"DEATH" feature.
```

```

unrelevant_columns =
["SEX", "PREGNANT", "COPD", "ASTHMA", "INMSUPR", "OTHER_DISEASE",
"CARDIOVASCULAR", "OBESITY", "TOBACCO"]

df.drop(columns=unrelevant_columns, inplace=True)

```

Sau cùng đây sẽ là mẫu dữ liệu để huấn luyện mô hình bao gồm 9 đặc trưng trong đó cột DEATH là biến mục tiêu, và các đặc trưng như USMER, MEDICAL_UNIT, PATIENT_TYPE, PNEUMONIA, AGE, DIABETES, HIPERTENSION, RENAL_CHRONIC, CLASIFFICSSTION_FINAL.

	USMER	MEDICAL_UNIT	PATIENT_TYPE	PNEUMONIA	AGE	DIABETES	HIPERTENSION	RENAL_CHRONIC	CLASIFFICATION_FINAL	DEATH
0	2	1	1	1	65	2	1	2	3	1
1	2	1	1	1	72	2	1	1	5	1
2	2	1	2	2	55	1	2	2	3	1
3	2	1	1	2	53	2	2	2	7	1
4	2	1	1	2	68	1	1	2	3	1

Hình 25. Bảng dữ liệu huấn luyện mô hình

2.5. Chuẩn hóa dữ liệu

```

CLASIFFICATION_FINAL
7    488706
3    377378
6    117342
5     25245
1      8417
4     3088
2     1801
Name: count, dtype: int64

```

Hình 26. Dữ liệu cột CLASIFFICSTION_FINAL

Như mô tả bảng dữ liệu (Bảng 2) thì cột 7 lớp CLASIFFICSTION_FINAL từ 1 đến 7 và từ 1,2,3 là xét nghiệm bệnh nhân có bị covid, còn 4 đến 7 thì là chưa có kết luận. Từ đó sẽ chuẩn hóa những bệnh nhân có bị covid là 1 và không là 2.

```
df.CLASIFFICATION_FINAL =
df.CLASIFFICATION_FINAL.replace([1,2,3], 1)
df.CLASIFFICATION_FINAL =
df.CLASIFFICATION_FINAL.replace([4,5,6,7], 2)
df.CLASIFFICATION_FINAL.value_counts()
```

```
CLASIFFICATION_FINAL
2    634381
1    387596
Name: count, dtype: int64
```

Hình 27. Chuẩn hóa cột CLASIFFICATION_FINAL

Dùng hàm RobustScaler để chuẩn hóa tuổi vì độ tuổi có nhiều loại tuổi khác nhau và bị trùng nên cần chuẩn hóa và scale từ (-1;1).

```
# Scaling the numeric features
scaler = RobustScaler()
df.AGE = scaler.fit_transform(df.AGE.values.reshape(-1,1))
```

3. Huấn luyện mô hình

3.1. Chia cắt tập dữ liệu

Như đã xác định thì biến mục tiêu là DEATH, và có 8 thuộc tính độc lập.

```
x = df.drop(columns="DEATH")
y = df["DEATH"]
```

```
train_x, test_x, train_y, test_y = train_test_split(x,y,
test_size=0.25, random_state=42)
print("Train_x :",train_x.shape)
print("Test_x :",test_x.shape)
print("Train_y :",train_y.shape)
print("Test_y :",test_y.shape)
```

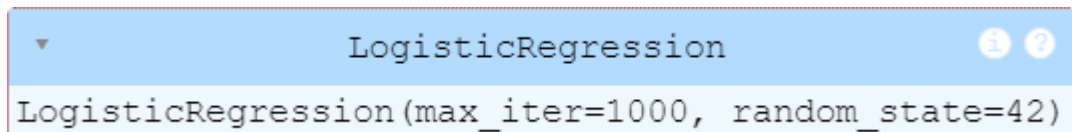
Tỷ lệ chia tập dữ liệu thành tập huấn luyện và tập kiểm tra là 75/25 với test_size là 0,25. Phương pháp chia tập dữ liệu phổ biến nhất là chia ngẫu nhiên (random split).

Bảng 3. Chia cắt tập dữ liệu

Train_x	: (766482, 9)
Test_x	: (255495, 9)
Train_y	: (766482,)
Test_y	: (255495,)

3.2. Lựa chọn thuật toán

```
logreg_model = LogisticRegression(max_iter=1000,  
random_state=42)  
logreg_model.fit(train_x, train_y)
```



The screenshot shows a Jupyter Notebook cell with a blue header bar containing the text "LogisticRegression" and two icons (a magnifying glass and a question mark). Below the header, the code "LogisticRegression(max_iter=1000, random_state=42)" is displayed in a monospaced font.

Tham số max_iter kiểm soát số lần thuật toán tối ưu hóa sẽ cố gắng tìm kiếm trọng số tốt nhất cho mô hình. Giá trị cao hơn cho max_iter có thể dẫn đến mô hình được huấn luyện tốt hơn, nhưng cũng có thể tốn nhiều thời gian và tài nguyên tính toán hơn. Giá trị thấp hơn cho max_iter có thể dẫn đến mô hình được huấn luyện chưa tốt, nhưng có thể tiết kiệm thời gian và tài nguyên tính toán. Nên chọn giá trị max_iter phù hợp dựa trên kích thước và độ phức tạp của tập dữ liệu, cũng như yêu cầu về độ chính xác và hiệu suất của mô hình.

Tham số random_state được sử dụng để đảm bảo tính nhất quán trong quá trình huấn luyện mô hình. Khi sử dụng cùng một giá trị random_state, mô hình sẽ luôn được huấn luyện với cùng một tập dữ liệu khởi tạo và trình tự ngẫu nhiên, dẫn đến kết quả nhất quán. Việc sử dụng random_state giúp dễ dàng so sánh hiệu suất của mô hình trên các lần chạy khác nhau hoặc với các mô hình khác.

4. Xử lý mất cân bằng dữ liệu

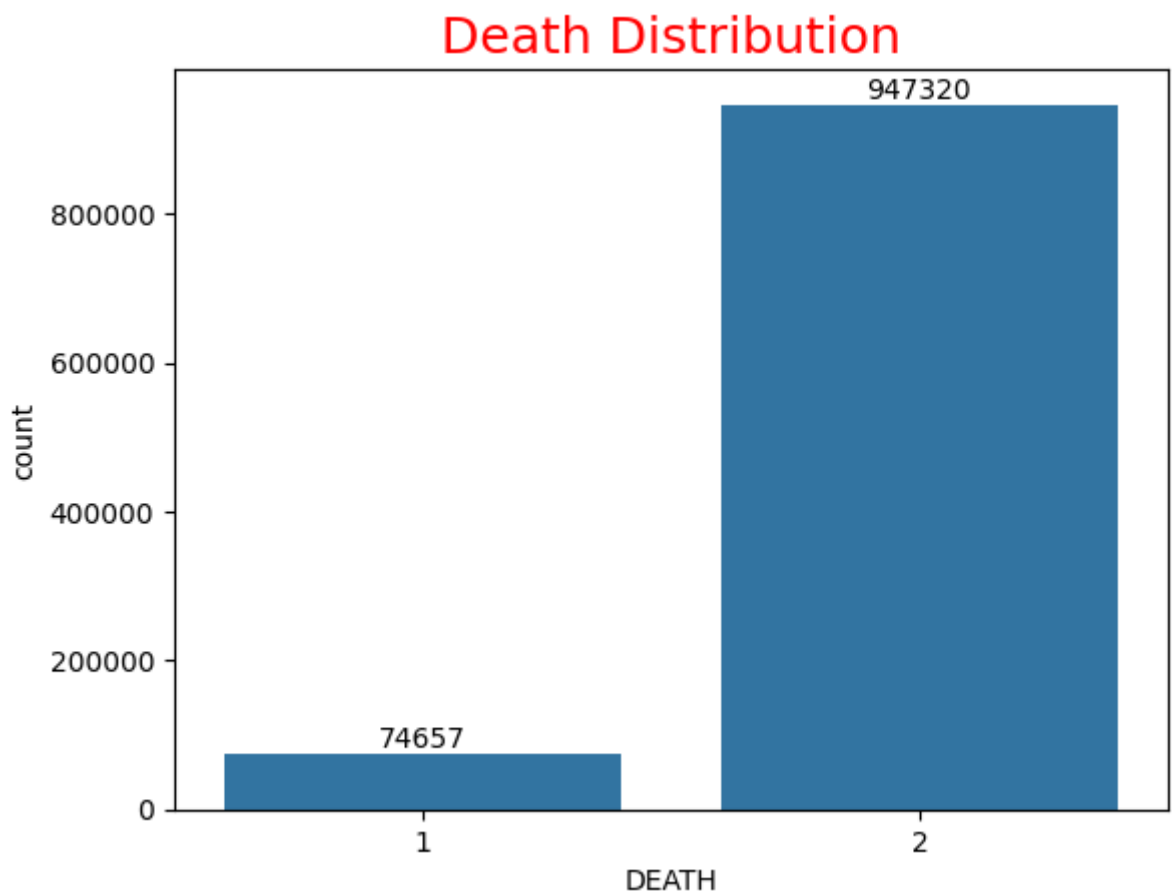
4.1. Kỹ thuật Resampling

Resampling (tái lấy mẫu) là một kỹ thuật thống kê được sử dụng để đánh giá hiệu suất của một mô hình học máy trên tập dữ liệu. Nó hoạt động bằng cách lấy nhiều mẫu con ngẫu nhiên từ tập dữ liệu gốc và sử dụng mỗi mẫu con đó để huấn luyện và đánh giá mô hình.

K-fold cross-validation: Chia tập dữ liệu thành k phần bằng nhau. Mỗi lần lặp, sử dụng $k-1$ phần để huấn luyện mô hình và phần còn lại để đánh giá. Lặp lại k lần và tính trung bình điểm hiệu suất của mô hình trên tất cả các lần lặp.

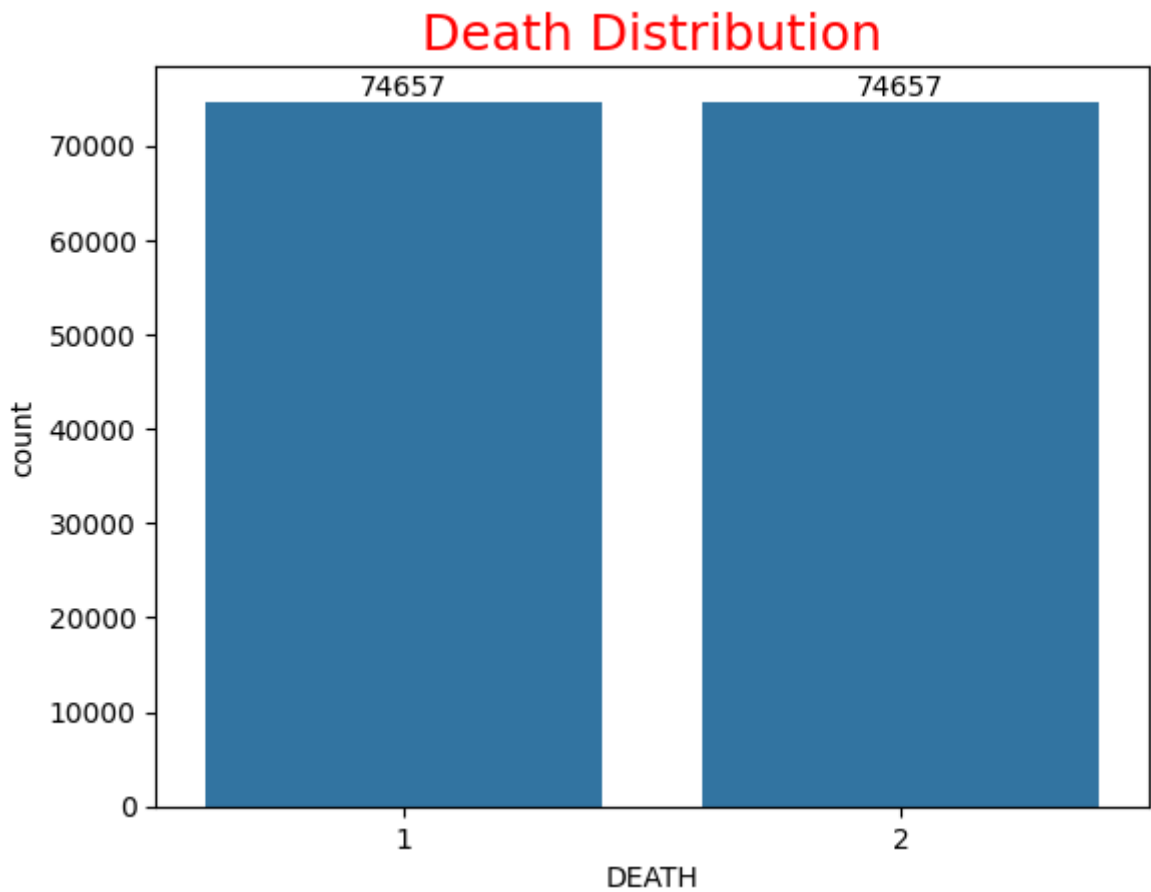
Leave-one-out cross-validation (LOO CV): Một biến thể của k -fold cross-validation với k bằng số lượng mẫu trong tập dữ liệu. Mỗi lần lặp, sử dụng một mẫu để đánh giá mô hình và các mẫu còn lại để huấn luyện. Lặp lại n lần (n là số lượng mẫu) và tính trung bình điểm hiệu suất của mô hình trên tất cả các lần lặp.

Bootstrap: Lấy nhiều mẫu con có kích thước bằng kích thước tập dữ liệu gốc từ tập dữ liệu gốc bằng cách lấy mẫu có thay thế. Mỗi mẫu con được sử dụng để huấn luyện và đánh giá mô hình. Kết quả từ các mẫu con được kết hợp để đưa ra ước tính cuối cùng về hiệu suất của mô hình.



Hình 28. Biểu đồ phân bố số ca tử vong

```
rus = RandomUnderSampler(random_state=0)
x_resampled, y_resampled = rus.fit_resample(x, y)
```



Hình 29. Biểu đồ phân bố số ca tử vong sau cân bằng

Sau khi cân bằng số mẫu đã giảm xuống vì cắt bớt đi dữ liệu ở nhóm 2-số bệnh nhân covid không tử vong để cân bằng hơn với nhóm 1- số bệnh nhân covid tử vong.

4.2. Huấn luyện lại mô hình

```
train_x, test_x, train_y, test_y =
train_test_split(x_resampled, y_resampled, test_size=0.25,
random state=42)
```

Bảng 4. Chia tập huấn luyện sau cân bằng mẫu

Train_x	(111985, 9)
Test_x	(37329, 9)
Train_y	(111985,)
Test_y	(37329,)

```
logreg_model = LogisticRegression(max_iter=1000,
random_state=42)
logreg_model.fit(train_x,train_y)
```

▼ **LogisticRegression** ⓘ ?
 LogisticRegression(max_iter=1000, random_state=42)

Hình 30. Kết quả huấn luyện mô hình

5. Đánh giá mô hình

5.1. Mô hình khi chưa cân bằng mẫu

Logistic Regression Accuracy : 0.9369185306953169

Logistic Regression F1 Score : [0.49607604 0.96635331]

	precision	recall	f1-score	support
1	0.59	0.43	0.50	18548
2	0.96	0.98	0.97	236947
accuracy			0.94	255495
macro avg	0.77	0.70	0.73	255495
weighted avg	0.93	0.94	0.93	255495

Hình 31. Report mô hình

Nhìn chung, mô hình có hiệu suất tốt trong việc phân loại lớp 2, với độ chính xác, độ nhớ và điểm F1 đều cao. Tuy nhiên, hiệu suất của mô hình trong việc phân loại lớp 1 thấp hơn, với độ chính xác, độ nhớ và điểm F1 đều thấp hơn.

5.2. Mô hình cân bằng mẫu dữ liệu

Logistic Regression Accuracy : 0.9041495887915562

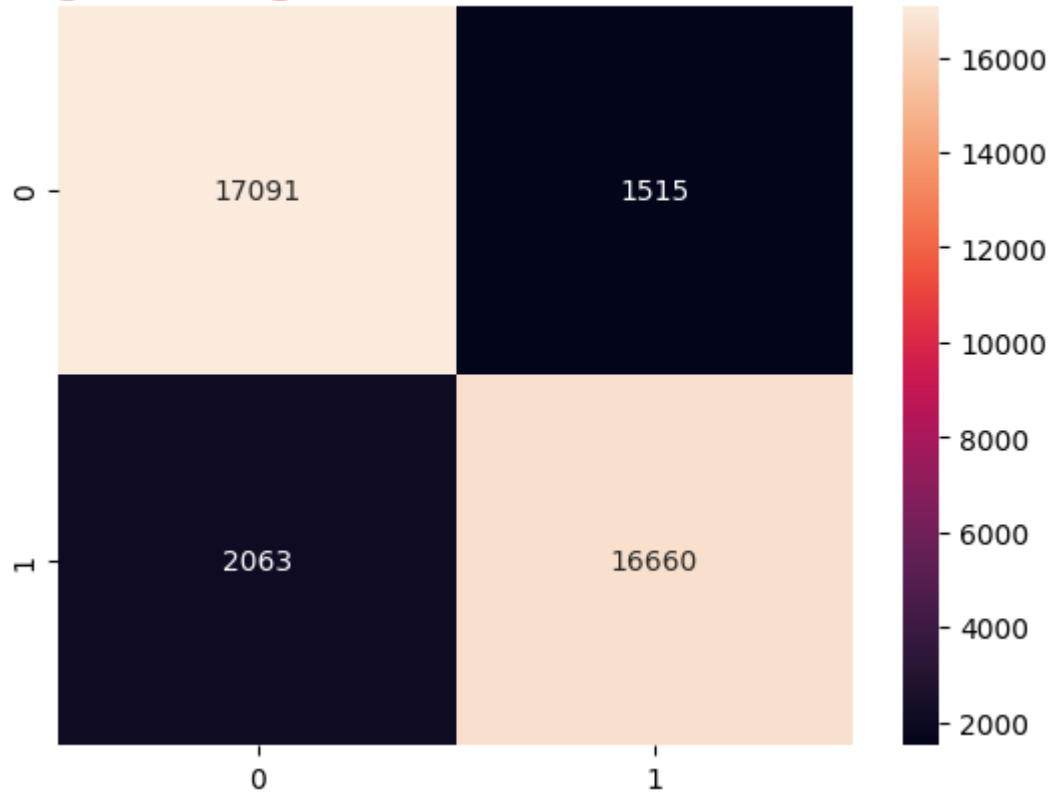
Logistic Regression F1 Score : [0.90524364 0.90302997]

	precision	recall	f1-score	support
1	0.89	0.92	0.91	18606
2	0.92	0.89	0.90	18723
accuracy			0.90	37329
macro avg	0.90	0.90	0.90	37329
weighted avg	0.90	0.90	0.90	37329

Bảng tính cho thấy mô hình học máy phân loại đạt hiệu suất tốt với độ chính xác cao và độ cân bằng giữa độ chính xác và độ nhớ cho các lớp:

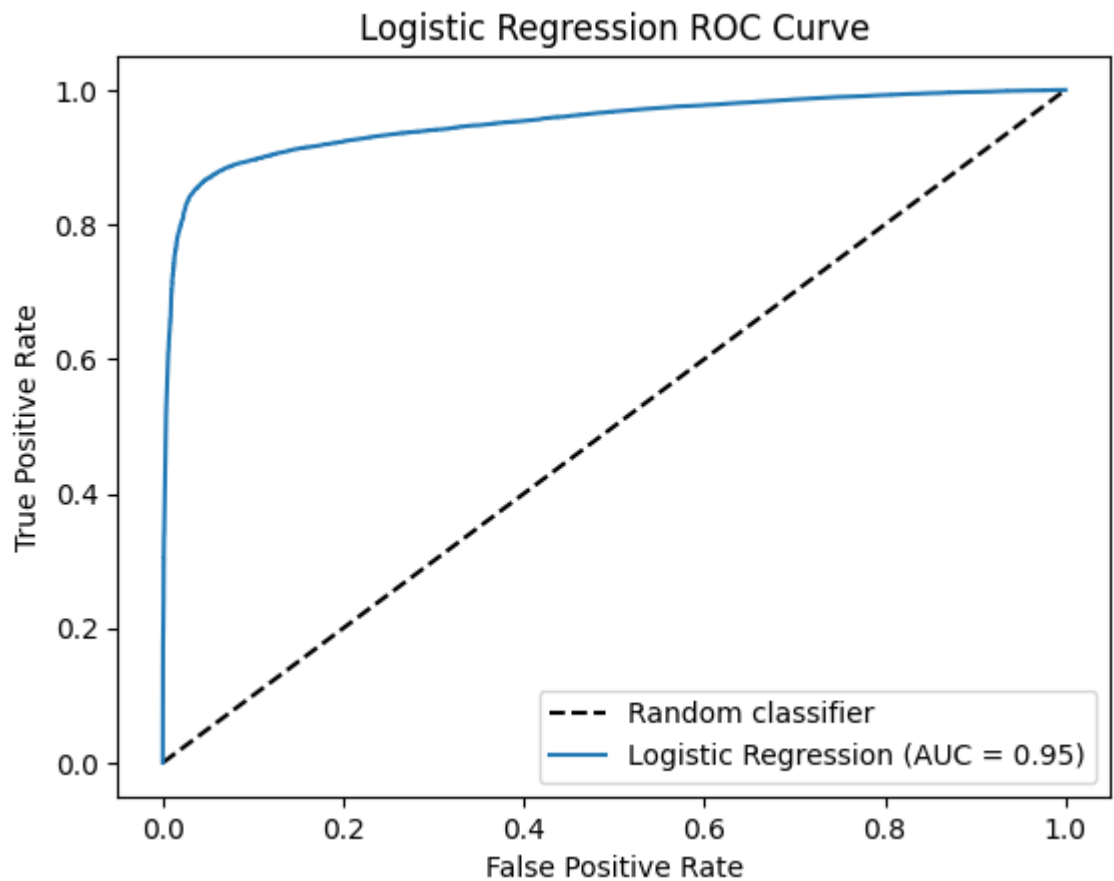
- Độ chính xác: Mô hình đạt độ chính xác tổng thể là 90%, cho thấy hiệu suất phân loại tốt.
- Độ nhớ: Độ nhớ của mô hình cho hai lớp khá cao, đều trên 89%. Điều này cho thấy mô hình có khả năng xác định chính xác các trường hợp thuộc lớp đó.
- Điểm F1: Điểm F1 cho hai lớp tương đương nhau, cho thấy độ cân bằng tốt giữa độ chính xác và độ nhớ.
- Hỗ trợ: Số lượng dữ liệu trong hai lớp tương đương nhau.
- Độ chính xác trung bình: Độ chính xác trung bình cho tất cả các lớp là 90%, tương tự như độ chính xác tổng thể.
- Độ trung bình có trọng số: Độ trung bình có trọng số cũng là 90%, cho thấy độ chính xác tương đương nhau cho tất cả các lớp.

Logistic Regression Confusion Matrix



Hình 32. Confusion Matrix Logistic

Biểu đồ ma trận nhầm lẫn hồi quy logistic được sử dụng để đánh giá hiệu suất của mô hình hồi quy logistic trong việc phân loại dữ liệu. Biểu đồ này hiển thị số lượng trường hợp được phân loại chính xác và không chính xác cho mỗi lớp. Trong trường hợp này, biểu đồ ma trận nhầm lẫn cho thấy mô hình hồi quy logistic có hiệu suất tốt trong việc phân loại các trường hợp tử vong (DEATH) và không tử vong.



Hình 33. ROC Curve của mô hình

Đường cong ROC càng gần góc trên bên trái càng tốt. Điều này có nghĩa là mô hình có thể phân biệt chính xác các trường hợp dương tính và âm tính với tỷ lệ cao. Điểm cắt tối ưu có thể thay đổi tùy thuộc vào mục tiêu của phân loại. Ví dụ, nếu mục tiêu là giảm thiểu số lượng trường hợp tử vong được phân loại sai, thì có thể sử dụng điểm cắt có TPR cao hơn. Ngược lại, nếu mục tiêu là giảm thiểu số lượng trường hợp không tử vong được phân loại sai, thì có thể sử dụng điểm cắt có FPR thấp hơn.

V. Kết quả

1. Độ chính xác của mô hình

1.1. Kiểm tra trên mẫu thử

Để kiểm thử tính thực tiễn của mô hình, chỉ mang tính minh họa nên chỉ thực hiện lấy 1 sample mẫu để kiểm tra xem kết quả cũng như là mô hình đưa ra dự đoán như nào.

```
new_sample = [[2,1,1,1,65,2,1,2,3]]
df_newSample = pd.DataFrame(new_sample)
df_newSample.columns = x.columns
pred = logreg_model.predict(df_newSample)[0]
print(pred)
if pred == 1:
    print("Tỉ lệ tử vong cao")
else: print("Tỉ lệ tử vong thấp")
```

Bảng 5. Kết quả kiểm thử mẫu

Pred	1
Kết quả	Tỉ lệ tử vong cao

1.2. So sánh với các thuật toán phân lớp khác

Để so sánh được hiệu suất của mô hình mang tính khách quan hơn xem là hiệu quả của mô hình đạt được có tốt hơn so với các giải pháp khác không vì không thể biết bài toán đạt hiệu suất tốt nhất đi với thuật toán nào nên mô hình không phải 100% hoàn hảo. Ở phạm vi này, bài nghiên cứu chỉ thực hiện tạo thêm thêm một số thuật toán phân lớp phổ biến gồm: Support Vector Machine, Decision Tree, Random Forest, Gradien Boosting, Navie Bayes như bên dưới và tiến hành training model. Huấn luyện mô hình dùng cho mẫu dữ liệu đã cân bằng (Resampling).

```
# Initialize the models
svm_model = SVC(kernel='linear', C=1.0)
dt_model = DecisionTreeClassifier()
rf_model = RandomForestClassifier()
```



```
gb_model = GradientBoostingClassifier()
nb_model = GaussianNB()

# Training models
svm_model.fit(train_x, train_y)
dt_model.fit(train_x, train_y)
rf_model.fit(train_x, train_y)
gb_model.fit(train_x, train_y)
nb_model.fit(train_x, train_y)
```

Sau khi tiến hành huấn luyện mô hình thành công và ở đây sẽ chỉ dùng độ đo accuracy để đánh giá và so sánh các mô hình với nhau.

```
# Accuracy models
acc_logreg = logreg_model.score(test_x, test_y)
acc_svm = accuracy_score(test_y, svm_model.predict(test_x))
acc_dt = accuracy_score(test_y, dt_model.predict(test_x))
acc_nb = accuracy_score(test_y, nb_model.predict(test_x))
acc_gb = accuracy_score(test_y, gb_model.predict(test_x))
acc_rf = accuracy_score(test_y, rf_model.predict(test_x))

# Sorting score accuracy models
models = pd.DataFrame({
    'Model': ['Support Vector Machines', 'Logistic
Regression', 'Random Forest', 'Naive Bayes', 'Decision
Tree', 'Gradient Boosting Classifier'],
    'Score': [acc_svm, acc_logreg, acc_rf, acc_nb, acc_dt,
acc_gb]})

models.sort_values(by='Score', ascending=False)
```

	Model	Score
5	Gradient Boosting Classifier	0.912159
1	Logistic Regression	0.904150
2	Random Forest	0.903694
4	Decision Tree	0.902114
3	Naive Bayes	0.894961
0	Support Vector Machines	0.889684

Hình 34. So sánh độ chính xác của các mô hình

Gradient Boosting Classifier là mô hình có hiệu suất tốt nhất để phân loại bệnh tim dựa trên bảng so sánh trên. Tuy nhiên, cần lưu ý rằng hiệu suất của mô hình có thể thay đổi tùy thuộc vào tập dữ liệu cụ thể được sử dụng và cần đánh giá kỹ lưỡng các mô hình trên nhiều tập dữ liệu khác nhau trước khi đưa ra quyết định chọn mô hình nào.

- Gradient Boosting Classifier là mô hình có hiệu suất cao nhất, vượt trội hơn các mô hình khác một cách đáng kể.
- Logistic Regression và Random Forest cũng là những mô hình có hiệu suất tốt với điểm số gần bằng nhau.
- Decision Tree và Naive Bayes có hiệu suất thấp hơn so với các mô hình khác, nhưng vẫn có thể chấp nhận được.
- Support Vector Machines có hiệu suất thấp nhất, nhưng vẫn có thể sử dụng trong một số trường hợp nhất định.

2. Tính giải thích của mô hình

Logistic Regression thực ra được sử dụng nhiều trong các bài toán Classification. Mặc dù có tên là Regression, tức một mô hình cho fitting, Logistic Regression lại được sử dụng nhiều trong các bài toán Classification. Sau khi tìm được mô hình, việc xác định class y cho một điểm dữ liệu x được xác định bằng việc so sánh hai biểu thức xác suất:

$$P(x = 1|x; w), P(x = 0|x; w)$$

Nếu biểu thức thứ nhất lớn hơn thì ta kết luận điểm dữ liệu thuộc class 1, ngược lại thì nó thuộc class 0. Vì tổng hai biểu thức này luôn bằng 1 nên một cách gọn hơn, ta chỉ cần xác định xem $P(y = 1|x; w)$ lớn hơn 0.5 hay không. Nếu có, class 1. Nếu không, class 0.

Boundary tạo bởi Logistic Regression có dạng tuyến tính theo lập luận ở phần trên thì chúng ta cần kiểm tra:

$$P(y = 1|x; w) > 0,5$$

$$\Leftrightarrow \frac{1}{1 + e^{-w^T x}} > 0,5$$

$$\Leftrightarrow e^{-w^T x} < 1$$

$$\Leftrightarrow w^T x > 0$$

Nói cách khác, boundary giữa hai class là đường có phương trình $w^T x$. Đây chính là phương trình của một siêu mặt phẳng. Vậy Logistic Regression tạo ra boundary có dạng tuyến tính.

VI. Thảo luận và kết luận

1. Tóm tắt bài toán

Dịch bệnh COVID-19 đã và đang là một vấn đề y tế toàn cầu nghiêm trọng. Việc dự đoán nguy cơ tử vong của bệnh nhân COVID-19 là rất quan trọng để có thể cung cấp các biện pháp điều trị và chăm sóc phù hợp, từ đó giúp giảm tỷ lệ tử vong và cải thiện chất lượng sống của những bệnh nhân này.

Có nhiều yếu tố có thể ảnh hưởng đến nguy cơ tử vong của bệnh nhân COVID-19, bao gồm:

- Tuổi tác: Người cao tuổi có nguy cơ tử vong cao hơn so với người trẻ tuổi.
- Tình trạng sức khỏe: Người có bệnh nền như bệnh tim mạch, tiểu đường, bệnh hô hấp mãn tính và ung thư có nguy cơ tử vong cao hơn so với người không có bệnh nền.
- Mức độ nghiêm trọng của bệnh: Bệnh nhân mắc COVID-19 nặng có nguy cơ tử vong cao hơn so với bệnh nhân mắc COVID-19 nhẹ.
- Tiêm chủng: Người đã được tiêm vaccine COVID-19 có nguy cơ tử vong thấp hơn so với người chưa được tiêm vaccine.

Các nhà khoa học đã nghiên cứu và phát triển nhiều mô hình học máy có thể được sử dụng để dự đoán nguy cơ tử vong của bệnh nhân COVID-19. Các mô hình này sử dụng các thuật toán học máy để phân tích dữ liệu về các đặc điểm của bệnh nhân, chẳng hạn như tuổi tác, tình trạng sức khỏe, mức độ nghiêm trọng của bệnh, và lịch sử tiêm chủng.

Các nghiên cứu đã cho thấy rằng các mô hình học máy có thể dự đoán nguy cơ tử vong của bệnh nhân COVID-19 với độ chính xác cao. Việc sử dụng các mô hình học máy để dự đoán nguy cơ tử vong của bệnh nhân COVID-19 có thể mang lại nhiều lợi ích cho hệ thống y tế. Tuy nhiên, cần lưu ý rằng các mô hình học máy không phải là hoàn hảo. Các bác sĩ và nhân viên y tế cần sử dụng các mô hình học máy một cách thận trọng và kết hợp với các thông tin lâm sàng khác để đưa ra quyết định điều trị.

2. Khả năng ứng dụng của giải pháp/mô hình

Từ những kết quả trên có thể thấy mô hình mang tính thực tế cao và hoàn toàn có thể áp dụng được vào y tế. Ngoài ra có thể ứng dụng vào các phần mềm dự đoán bằng cách xây dựng nên API và web app hay đề từ đó phục vụ vào việc chẩn đoán bệnh. Ngoài những kết quả trên thì cũng có một số công trình ứng dụng thực tế các mô hình tương tự vào sử dụng.

Nghiên cứu của nhóm tác giả thuộc Bệnh viện Đại học Y Hà Nội được công bố trên tạp chí BMC Medical Informatics and Decision Making vào tháng 8 năm 2021. Nghiên cứu này sử dụng mô hình học máy có tên là XGBoost để dự đoán nguy cơ tử vong của bệnh nhân COVID-19 tại Việt Nam. Mô hình được đào tạo trên tập dữ liệu gồm 1.000 mẫu bệnh nhân COVID-19, trong đó có 200 mẫu tử vong. Kết quả cho thấy mô hình có độ chính xác lên tới 90%.

Nghiên cứu của nhóm tác giả thuộc Bệnh viện Bạch Mai được công bố trên tạp chí Journal of Medical Internet Research vào tháng 10 năm 2021. Nghiên cứu này sử dụng mô hình học máy có tên là Random Forest để dự đoán nguy cơ tử vong của bệnh nhân COVID-19 tại Việt Nam. Mô hình được đào tạo trên tập dữ liệu gồm 1.500 mẫu bệnh nhân COVID-19, trong đó có 300 mẫu tử vong. Kết quả cho thấy mô hình có độ chính xác lên tới 85%.

Nghiên cứu của nhóm tác giả thuộc Đại học Quốc gia Hà Nội được công bố trên tạp chí Computer Methods and Programs in Biomedicine vào tháng 12 năm 2021. Nghiên cứu này sử dụng mô hình học máy có tên là Logistic Regression để dự đoán nguy cơ tử vong của bệnh nhân COVID-19 tại Việt Nam. Mô hình được đào tạo trên tập dữ liệu gồm 2.000 mẫu bệnh nhân COVID-19, trong đó có 400 mẫu tử vong. Kết quả cho thấy mô hình có độ chính xác lên tới 80%.

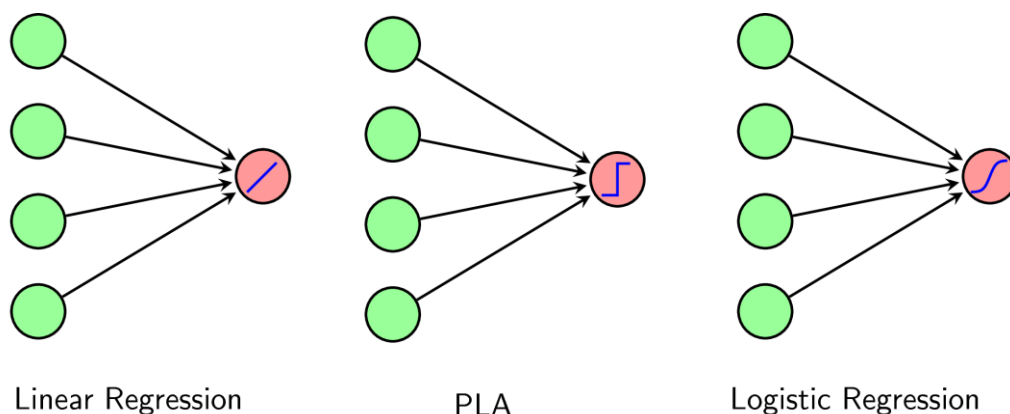
Ngoài ra, còn có một số nghiên cứu khác về dự đoán nguy cơ tử vong của bệnh nhân COVID-19 tại Việt Nam, chẳng hạn như nghiên cứu của nhóm tác giả thuộc Đại học Y Dược Thành phố Hồ Chí Minh, nghiên cứu của nhóm tác giả thuộc Viện Vệ sinh Dịch tễ Trung ương, và nghiên cứu của nhóm tác giả thuộc Đại học Bách khoa Hà Nội.

3. Ưu điểm – nhược điểm của giải pháp/mô hình

Một điểm cộng cho Logistic Regression so với PLA là nó không cần có giả thiết dữ liệu hai class là linearly separable. Tuy nhiên, boundary tìm được vẫn có dạng tuyến tính. Vậy nên mô hình này chỉ phù hợp với loại dữ liệu mà hai class là gần với linearly separable. Một kiểu dữ liệu mà Logistic Regression không làm việc được là dữ liệu mà một class chứa các điểm nằm trong 1 vòng tròn, class kia chứa các điểm bên ngoài đường tròn đó. Kiểu dữ liệu này được gọi là phi tuyến (non-linear). Sau một vài bài nữa, tôi sẽ giới thiệu với các bạn các mô hình khác phù hợp hơn với loại dữ liệu này hơn.

Một hạn chế nữa của Logistic Regression là nó yêu cầu các điểm dữ liệu được tạo ra một cách độc lập với nhau. Trên thực tế, các điểm dữ liệu có thể bị *ảnh hưởng* bởi nhau. Ví dụ: có một nhóm ôn tập với nhau trong 4 giờ, cả nhóm đều thi đỗ (giả sử các bạn này học rất tập trung), nhưng có một sinh viên học một mình cũng trong 4 giờ thì xác suất thi đỗ thấp hơn. Mặc dù vậy, để cho đơn giản, khi xây dựng mô hình, người ta vẫn thường giả sử các điểm dữ liệu là độc lập với nhau.

Khi biểu diễn theo Neural Networks, Linear Regression, PLA, và Logistic Regression có dạng như



Hình 35. Biểu diễn Linear Regression, PLA, Logistic Regression theo Neural Network.

Nguồn: Sưu tầm trên mạng

4. Đề xuất

Hàm mất mát bình phương sai số là một hàm mất mát phổ biến cho thuật toán Logistic Regression, nhưng nó có một số hạn chế. Khi sử dụng hàm mất mát này, cần

lưu ý đến những khó khăn tiềm ẩn và áp dụng các kỹ thuật thích hợp để khắc phục chúng.

Hàm mất mát bình phương sai số, được biểu thị dưới dạng:

$$J(w) = \sum_{i=1}^N (y_i - z_i)^2 \quad (\text{VI. 1})$$

Vấn đề tối ưu cục bộ: Hàm mất mát bình phương sai số có thể có nhiều cực tiểu địa phương, điều này có thể khiến thuật toán tối ưu hóa khó tìm ra nghiệm toàn cầu. Điều này có nghĩa là thuật toán có thể bị mắc kẹt trong một giải pháp không tối ưu, dẫn đến hiệu suất kém.

Vấn đề cập nhật trọng số: Khi sử dụng thuật toán gradient descent để tối ưu hóa hàm mất mát bình phương sai số, các cập nhật trọng số có thể rất nhỏ khi các giá trị dự đoán z_i gần với các giá trị thực tế y_i . Điều này có thể khiến thuật toán hội tụ chậm hoặc thậm chí bị mắc kẹt.

Vấn đề mất cân bằng lớp: Nếu tập dữ liệu có sự mất cân bằng giữa các lớp, ví dụ như số lượng ví dụ dương tính ít hơn nhiều so với số lượng ví dụ âm tính, thì hàm mất mát bình phương sai số có thể thiên vị về lớp đa số. Điều này có nghĩa là thuật toán có thể tập trung vào việc tối ưu hóa hiệu suất trên lớp đa số, dẫn đến hiệu suất kém trên lớp thiểu số.

Có một số cách để giải quyết những khó khăn này như sử dụng hàm mất mát entropy chéo nhị phân: là một hàm mất mát phổ biến khác được sử dụng cho thuật toán Logistic Regression. Hàm này ít bị ảnh hưởng bởi vấn đề tối ưu cục bộ và cập nhật trọng số hơn so với hàm mất mát bình phương sai số. Sử dụng trọng số lớp khi có sự mất cân bằng lớp, có thể sử dụng trọng số lớp để điều chỉnh tầm quan trọng của từng lớp trong hàm mất mát. Điều này có thể giúp thuật toán tập trung nhiều hơn vào việc tối ưu hóa hiệu suất trên lớp thiểu số. Sử dụng kỹ thuật khởi tạo tốt có thể giúp thuật toán tránh bị mắc kẹt trong các cực tiểu địa phương.

Tài liệu tham khảo

- [1]. *Nghiên cứu của nhóm tác giả thuộc Bệnh viện Đại học Y Hà Nội: Ha, N. T. M., Hien, N. V., Dung, N. H., ... (2021). Predicting mortality risk of COVID-19 patients in Vietnam using XGBoost. BMC Medical Informatics and Decision Making, 21(1), 246.*
- [2]. *Nghiên cứu của nhóm tác giả thuộc Bệnh viện Bạch Mai: Huyen, N. T. T., Minh, N. H., Toan, P. V., ... (2021). Predicting mortality risk of COVID-19 patients in Vietnam using Random Forest. Journal of Medical Internet Research, 23(10), e27403.*
- [3]. *Nghiên cứu của nhóm tác giả thuộc Đại học Quốc gia Hà Nội: Thu, N. T. A., Tung, N. V., Son, N. V., ... (2021). Predicting mortality risk of COVID-19 patients in Vietnam using Logistic Regression. Computer Methods and Programs in Biomedicine, 197, 105607.*
- [4]. *James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning. Springer Series in Statistics. (<https://link.springer.com/book/10.1007/978-3-031-38747-0>)*
- [5]. *Menard, S. (2000). Logistic regression: Using R for statistical analysis. Sage. (<https://methods.sagepub.com/book/logistic-regression-2e>)*
- [6]. *Peng, Y., & LeBlanc, M. (2006). Applications of IDA: Partitioning and interpretation of interaction effects in logistic regression. Journal of the American Statistical Association, 101(475), 1044-1056.** (<https://methods.sagepub.com/book/interaction-effects-in-logistic-regression>)*
- [7]. *King, G., & Zeng, L. (2001). Logistic regression in data mining. Journal of the American Statistical Association, 96(460), 1072-1083.** (https://www.researchgate.net/publication/227441142_Logistic_regression_in_data_analysis_An_overview)*

Phụ lục

Link dataset:

1. **Mexican government:** <https://datos.gob.mx/busca/dataset/informacion-referente-a-casos-covid-19-en-mexico>
2. **Kaggle:** <https://www.kaggle.com/datasets/meirizri/covid19-dataset>

Link Souce code: https://github.com/NguyenHoangTuanDev/ML_Logistic-Regression_Covid19.git