

TRƯỜNG ĐẠI HỌC QUỐC TẾ HỒNG BÀNG
BỘ MÔN CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN

MÔN HỌC
KHAI THÁC DỮ LIỆU LỚN

Giảng viên hướng dẫn: Lê Văn Hạnh

Giảng viên hướng dẫn: Nguyễn Minh Vũ

Mã số sinh viên: 2211110063

TP. Hồ Chí Minh, 2025

LỜI CẢM ƠN

Để hoàn thành đề tài này, em đã nhận được sự hướng dẫn, giúp đỡ và góp ý nhiệt tình của quý thầy cô trường Đại Học Quốc Tế Hồng Bàng và thầy Lê Văn Hạnh.

Em xin gửi lời biết ơn sâu sắc đến thầy Lê Văn Hạnh đã dành nhiều thời gian và tâm huyết hướng dẫn nghiên cứu và giúp em hoàn thành môn học.

Em cũng xin chân thành cảm ơn đến quý thầy cô trường Đại học Quốc Tế Hồng Bàng, đặc biệt là những thầy cô đã tận tình dạy bảo cho em suốt thời gian học tập tại trường.

Em đã có nhiều cố gắng hoàn thiện dự án bằng tất cả năng lực của mình, tuy nhiên không thể tránh khỏi nhiều thiếu sót, rất mong nhận được những đóng góp quý báu của quý thầy cô và các bạn.

TP.HCM, ... Tháng ... Năm 2025

Người thực hiện

Nguyễn Minh Vũ

TRANG CAM KẾT

Tôi xin cam kết báo cáo thường kỳ này được hoàn thành dựa trên các kết quả thực hiện bài thực hành của tôi và các mã nguồn và kết quả này chưa được dùng cho bất cứ báo cáo của sinh viên nào khác.

TP.HCM, ngày tháng năm ...2025..

Người thực hiện

Nguyễn Minh Vũ

NHẬN XÉT CỦA GIẢNG VIÊN

TP.HCM, Ngày ... Tháng ... Năm 2025

Chữ ký giảng viên

DANH MỤC BIỂU ĐỒ HÌNH VẼ

Hình 4.1-1. Hình Boxplot Chart về lứa tuổi và cân nặng cơ thể	30
Hình 4.1-2. QQ Plot về sức khỏe thường ngày và giờ giấc ngủ.....	30
Hình 4.1-3. Biểu đồ Histogram giữa BMI và tỷ lệ người bị đau tim	31
Hình 4.1-4. Biểu đồ Scatter Plot giữ chiều cao, cân nặng và chỉ số BMI.....	32
Hình 4.1-5. Boxplot chart giữa chỉ số BMI và khu vực quốc gia	32
Hình 4.1-6. Histogram plot về sự phân bố trong từng lứa tuổi	33
Hình 4.2-1. Hình xử lý dữ liệu đầu vào.....	35
Hình 4.2-2. Các cột dữ liệu trước khi chuyển đổi và phân tách.....	36
Hình 4.2-3. Dữ liệu sau khi chuyển đổi và phân tách	37
Hình 4.2-4. Các cột dữ liệu được điền bằng mean, mode, median	37
Hình 4.2-5. Bộ dữ liệu trước khi điền null	38
Hình 4.2-6. Mảng đo lường sự tương đồng (kết quả của mục 4.4)	39
Hình 4.2-7. Dữ liệu sau khi đã điền null	39
Hình 4.3-1. Stacked Barplot thể hiện tỷ lệ bệnh nhồi máu cơ tim trong giới tính	41
Hình 4.3-2. Pie chart giữa sự phân bố sức khỏe tổng quát.....	41
Hình 4.3-3. Doughnut chart thể hiện sự phân bố mốc thời gian kiểm tra sức khỏe.....	42
Hình 4.3-4. Line chart thể hiện nhóm tuổi có xu hướng sử dụng Ecigarette	42
Hình 4.3-5. Violin chart thể hiện tần xuất hút thuốc giữa các giới tính.....	43
Hình 4.3-6. Desnsity chart về sự ảnh hưởng của hút thuốc đến sức khỏe	43
Hình 4.3-7. Ridgeline plot về sự phân bố sức khỏe giữa nhóm tuổi và tình trạng.....	44
Hình 4.3-8. Correlation plot về mối quan hệ tương quan giữa cân nặng, chiều cao và BMI.....	45
Hình 4.3-9. Circular barplot về bệnh nhồi máu cơ tim giữa các bang	46
Hình 4.3-10. Heatmap thể hiện số lượng bệnh nhân bị nhồi máu cơ tim theo bang và tình trạng chung	47
Hình 4.3-11. Treemap về mối tương quan giữa các thông tin người bệnh theo từng bang	47
Hình 4.3-12. Bubble chart về mức độ hút thuốc làm tăng nguy cơ COPD	48
Hình 4.4-1. Bộ dữ liệu thực hiện đo lường sự tương đồng	49

Hình 4.4-2. Mảng numpy trả về kết quả đo lường của các fields danh nghĩa.....	49
Hình 4.4-3. Mảng numpy trả về kết quả đo lường của các fields thuộc thứ tự	50
Hình 4.4-4. Mảng numpy trả về kết quả đo lường của các fields thuộc tính số.....	50
Hình 4.4-5. Mảng numpy trả về kết quả đo lường của các fields thuộc tính nhị phân .	50
Hình 4.4-6. Mảng numpy trả về kết quả tổng kết của tất cả các fields	51
Hình 4.4-7. Bộ dữ liệu đầu vào để xét độ đo Cosin	51
Hình 4.4-8. Mảng numpy trả về kết quả thu được của độ đo Cosin	51
Hình 4.5-1. Bộ dữ liệu chuẩn bị khai thác dữ liệu	52
Hình 4.5-2. Kết quả tập luật của thuật toán FP-Growth.....	55
Hình 5.2-1. Cấu hình mạng máy ảo.....	58
Hình 5.3-1. Khởi chạy máy ảo với VirtualBox	62
Hình 5.3-2. Kết nối máy thật với máy ảo qua PowerShell.....	63
Hình 5.3-3. Demo quá trình khởi chạy hệ thống 1	63
Hình 5.3-4. Demo quá trình khởi chạy hệ thống 2	64
Hình 5.3-5. Điều kiện cần của các cổng khi chạy server	64
Hình 5.3-6. Giao diện hệ thống sau khi khởi tạo thành công	66
Hình 5.3-7. Kiểm tra các file cần thiết	68
Hình 5.3-8. Theo dõi quá trình hệ thống hoạt động với PowerShell	69
Hình 5.3-9. Theo dõi quá trình hệ thống hoạt động với hdfs 8088	70
Hình 5.3-10. Output chương trình chạy bằng hadoop cluster	71
Hình 6.1-1. Kết quả phương pháp đánh giá Lift của Apriori và FP-Growth	72
Hình 6.2-1. Kết quả phương pháp đánh giá Chi Square của Apriori và FP-Growth	73

DANH MỤC BIỂU ĐỒ BẢNG

Bảng 3.2-1. Bảng thuộc tính danh nghĩa	20
Bảng 3.2-2. Bảng thuộc tính thứ tự	21
Bảng 3.2-3. Bảng thuộc tính kiểu số	22
Bảng 3.2-4. Bảng thuộc tính nhị phân.....	29
Bảng 4.5-1. Bảng kết quả lặp lần 2 của Apriori	53
Bảng 4.5-2. Bảng kết quả lặp lần 3 của Apriori	53
Bảng 4.5-3. Bảng kết quả lặp lần 4 của Apriori (cuối cùng).....	53
Bảng 4.5-4. Kết quả thu được qua các lần lặp của thuật toán Apriori	54
Bảng 4.5-5. Kết quả tập luật về độ tin cậy thuật toán Apriori.....	54

MỤC LỤC

LỜI CẢM ƠN.....	i
TRANG CAM KẾT	ii
NHẬN XÉT CỦA GIÁNG VIÊN	iii
DANH MỤC BIỂU ĐỒ HÌNH VẼ	iv
DANH MỤC BIỂU ĐỒ BẢNG	vi
MỤC LỤC.....	7
CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI.....	12
1.1. Tổng quan.....	12
1.2. Lý do chọn đề tài	12
1.3. Mục tiêu đề tài.....	12
1.4. Ý nghĩa đề tài	13
1.5. Định hướng giải pháp.....	13
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	14
2.1. Tổng quan về dữ liệu lớn (Big Data).....	14
2.1.1. Khái niệm.....	14
2.1.2. Đặc trưng của dữ liệu lớn – mô hình 5V	14
2.2. Khai thác dữ liệu (Data Mining)	14
2.2.1. Khái niệm.....	14
2.2.2. Các bước cơ bản trong quá trình khai thác dữ liệu.....	14
2.3. Giới thiệu về Hadoop	15
2.3.1. Khái niệm.....	15
2.3.2. Các thành phần chính	15
2.3.3. Kiến trúc hoạt động của Hadoop	15
2.3.4. Cơ chế xử lý dữ liệu trong Hadoop – MapReduce.....	15
2.3.5. Ưu điểm và hạn chế của Hadoop	15
2.3.6. Các công cụ mở rộng của hệ sinh thái Hadoop	16
2.4. Giới thiệu về Spark & PySpark	16
2.4.1. Khái niệm Apache Spark.....	16

2.4.2. Kiến trúc Spark	16
2.4.3. Khái niệm về PySpark – Spark cho Python.....	17
2.4.4. Ưu điểm của PySpark trong khai thác dữ liệu lớn.....	17
CHƯƠNG 3: GIỚI THIỆU VỀ BỘ DỮ LIỆU	18
3.1. Tổng quan về cơ sở dữ liệu	18
3.1.1. Giới thiệu về tập dữ liệu	18
3.1.2. Nguồn thu thập	18
3.1.3. Tổng quan về bảng dữ liệu	18
3.2. Giới thiệu thuộc tính.....	19
3.2.1. Thuộc tính danh nghĩa	19
3.2.2. Thuộc tính thứ tự	20
3.2.3. Thuộc tính kiểu số	21
3.2.4. Thuộc tính nhị phân	22
CHƯƠNG 4: PHÂN TÍCH THỐNG KÊ TRÊN CSDL	30
4.1. Tìm hiểu dữ liệu	30
4.1.1. Thực hiện vẽ các đồ thị dựa trên thuộc tính tùy chọn	30
4.1.1.1. Boxplot Chart.....	30
4.1.1.2. QQ Plot	30
4.1.1.3. Histogram Plot	31
4.1.1.4. Scatter Plot.....	32
4.1.2. Vẽ đồ thị với các thuộc tính danh nghĩa	32
4.1.2.1. Boxplot Chart.....	32
4.1.2.2. Histogram Plot	33
4.2. Tiền xử lý dữ liệu	33
4.2.1. Clone dataset.....	33
4.2.2. Import thư viện và model cần sử dụng	34
4.2.3. Xử lý dữ liệu đầu vào	35
4.2.4. Chuyển đổi và phân tách dữ liệu	35
4.2.4.1. Yêu cầu	35

4.2.4.2. Triển khai	35
4.2.5. Điều giá trị còn thiếu	37
4.2.5.1. Đối với cột dữ liệu kiểu số.....	37
4.2.5.2. Đối với các kiểu dữ liệu còn lại	37
4.3. Trực quan hóa dữ liệu.....	39
4.3.1. Barplot chart	41
4.3.2. Pie chart	41
4.3.3. Doughnut chart	42
4.3.4. Line chart.....	42
4.3.5. Violin chart	43
4.3.6. Density chart.....	43
4.3.7. Ridgeline plot.....	44
4.3.8. Correlation plot.....	45
4.3.9. Circular barplot.....	46
4.3.10. Heatmap.....	47
4.3.11. Tree Map.....	47
4.3.12. Bubble chart.....	48
4.4. Đo lường sự tương đồng và khác biệt của dữ liệu	48
4.4.1. Ma trận tương quan.....	48
4.4.1.1. Đối với các fields danh nghĩa	49
4.4.1.2. Đối với các fields thứ tự	49
4.4.1.3. Đối với các fields số	50
4.4.1.4. Đối với các fields nhị phân	50
4.4.1.5. Kết quả của ma trận tương quan.....	51
4.4.2. Độ đo Cosin	51
4.5. Khai thác dữ liệu	52
4.5.1. Tập dữ liệu thực hiện tính toán.....	52
4.5.2. Phương pháp Apriori	52
4.5.2.1. Khái niệm.....	52

4.5.2.2. Diễn giải khái quát thuật toán	52
4.5.2.3. Kết quả qua các lần lặp	53
4.5.2.4. Kết quả của thuật toán Apriori	54
4.5.3. Phương pháp FP-Growth.....	54
4.5.3.1. Khái niệm.....	54
4.5.3.2. Diễn giải khái quát thuật toán	54
4.5.3.3. Kết quả của thuật toán FP-Growth	55
CHƯƠNG 5: GIỚI THIỆU VỀ HADOOP	56
5.1. Giới thiệu	56
5.1.1. Thành phần chính của Hadoop	56
5.1.1.1. Hadoop Distributed File System (HDFS).....	56
5.1.1.2. MapReduce	56
5.1.2. Các thành phần mở rộng trong hệ sinh thái	56
5.1.3. Ưu điểm của Hadoop	57
5.1.4. Ứng dụng của Hadoop	57
5.2. Cài đặt.....	57
5.2.1. Chuẩn bị môi trường	57
5.2.2. Tạo máy ảo trên VirtualBox	57
5.2.3. Cấu hình mạng	58
5.2.4. Cài đặt ubutu.....	58
5.2.5. Cài đặt OpenSSH.....	58
5.2.6. Cài đặt Java.....	58
5.2.7. Tải và cài đặt Hadoop	59
5.2.8. Cấu hình hệ thống	59
5.2.9. Cấu hình các file Hadoop	59
5.2.9.1. Core-site.xml	59
5.2.9.2. hdfs-site.xml	60
5.2.9.3. mapred-site.xml	60
5.2.9.4. yarn-site.xml	61

5.3. Mô phỏng.....	61
5.3.1. Bước 1.....	61
5.3.2. Bước 2.....	62
5.3.3. Bước 3.....	63
5.3.4. Bước 4.....	65
5.3.5. Bước 5.....	66
5.3.5.1. Powershell output	68
5.3.5.2. Giao diện UI Cluster Node	69
CHƯƠNG 6: ĐÁNH GIÁ MẪU THU ĐƯỢC.....	72
6.1. Đánh giá mẫu bằng thang đô tương quan Lift.....	72
6.1.1. Khái niệm.....	72
6.1.2. Triển khai.....	72
6.1.3. Kết quả thu được.....	72
6.2. Đánh giá mẫu bằng thang đo χ^2	73
6.2.1. Khái niệm.....	73
6.2.2. Triển khai.....	73
6.2.3. Kết quả thu được.....	73
CHƯƠNG 7: HƯỚNG PHÁT TRIỂN.....	74
7.1. Mở rộng quy mô dữ liệu và hiệu năng hệ thống	74
7.2. Ứng dụng các thuật toán học máy nâng cao.....	74
7.3. Mở rộng hệ thống phân tích bằng PySpark và các công cụ BI	74
7.4. Nâng cao khả năng tương tác và triển khai thực tế	75
7.5. Ứng Dụng Phân Tích Dữ Liệu Trong Các Lĩnh Vực Đặc Thù	75
TÀI LIỆU THAM KHẢO	76
KẾT LUẬN.....	77

CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

1.1. Tổng quan

Trong bối cảnh chuyển đổi số và sự bùng nổ của công nghệ thông tin, lượng dữ liệu được tạo ra mỗi ngày trên toàn cầu đang tăng trưởng với tốc độ vượt bậc. Dữ liệu không chỉ đến từ các tổ chức, doanh nghiệp mà còn từ người dùng cá nhân, mạng xã hội, thiết bị IoT, cảm biến, và các hệ thống giao dịch trực tuyến. Khối lượng dữ liệu khổng lồ này được gọi là dữ liệu lớn (Big Data) - đặc trưng bởi 5V:

- Volume (khối lượng lớn)
- Velocity (tốc độ cao)
- Variety (đa dạng định dạng)
- Veracity (tính xác thực)
- Value (giá trị tiềm năng)

Để xử lý và khai thác giá trị tiềm ẩn trong Big Data, các hệ thống truyền thống như cơ sở dữ liệu quan hệ (RDBMS) không còn đáp ứng đủ về hiệu năng và khả năng mở rộng. Từ đó, các nền tảng xử lý dữ liệu phân tán như Hadoop ra đời, cho phép lưu trữ và xử lý dữ liệu trên nhiều nút máy song song, đảm bảo khả năng mở rộng linh hoạt, độ tin cậy cao và hiệu quả chi phí tốt.

Hệ sinh thái Hadoop (bao gồm HDFS, YARN, MapReduce, Hive, HBase, Spark, v.v.) hiện là một trong những nền tảng cốt lõi cho việc xây dựng các hệ thống khai thác dữ liệu lớn hiện nay.

1.2. Lý do chọn đề tài

Dữ liệu đang trở thành tài sản quan trọng nhất của doanh nghiệp và tổ chức trong thời đại số. Việc khai thác hiệu quả dữ liệu lớn giúp tạo ra các giá trị thực tế như phân tích hành vi khách hàng, tối ưu vận hành, dự báo xu hướng và ra quyết định chính xác hơn. Tuy nhiên, việc xử lý lượng dữ liệu khổng lồ vượt ngoài khả năng của hệ thống đơn lẻ, đòi hỏi một giải pháp tính toán song song, phân tán và đáng tin cậy.

Hadoop là một trong những công nghệ mã nguồn mở phổ biến nhất, được nhiều công ty công nghệ hàng đầu như Google, Facebook và Amazon. Việc nghiên cứu và triển khai Hadoop trong thực tế giúp sinh viên, nhà nghiên cứu nắm bắt được các kỹ thuật cốt lõi của xử lý dữ liệu lớn, chuẩn bị cho các hướng phát triển chuyên sâu như Data Engineering, Machine Learning, AI, và Data Analytics.

1.3. Mục tiêu đề tài

- Tìm hiểu cơ sở lý thuyết về dữ liệu lớn (Big Data) và nền tảng Hadoop.

- Cài đặt và cấu hình hệ thống Hadoop trên môi trường thực nghiệm (cluster hoặc máy ảo).
- Xây dựng mô hình xử lý dữ liệu lớn trên Hadoop, bao gồm các bước thu thập, lưu trữ, và phân tích dữ liệu.
- Thực hiện các bài toán khai thác dữ liệu (ví dụ: phân tích log, xử lý tập dữ liệu lớn, tính toán thống kê, hoặc trích xuất thông tin hữu ích).
- Đánh giá hiệu năng và khả năng mở rộng của Hadoop so với phương pháp xử lý truyền thống.

1.4. Ý nghĩa đề tài

Về mặt học thuật: Đề tài giúp người nghiên cứu hiểu rõ các nguyên lý cốt lõi của hệ thống xử lý dữ liệu phân tán, cơ chế hoạt động của Hadoop, và các thành phần trong hệ sinh thái Big Data.

Về mặt thực tiễn: Giúp ứng dụng Hadoop để giải quyết các vấn đề thực tế trong doanh nghiệp - ví dụ như khai thác dữ liệu khách hàng, phân tích hành vi người dùng, tối ưu quy trình vận hành hoặc dự báo xu hướng.

Về mặt phát triển cá nhân: Cung cấp kỹ năng làm việc với dữ liệu lớn, quản lý cluster, và phát triển pipeline xử lý dữ liệu - những năng lực rất được yêu cầu trong lĩnh vực Data Engineering và AI hiện nay.

1.5. Định hướng giải pháp

- Cài đặt hệ thống Hadoop trên môi trường ảo dùng VirtualBox.
- Thiết lập cluster gồm NameNode và DataNodes.
- Dữ liệu được lưu trữ trong HDFS (Hadoop Distributed File System).
- Sử dụng MapReduce hoặc PySpark để thực hiện phép xử lý và khai thác dữ liệu.
- Kết hợp với Hive hoặc Spark SQL để truy vấn dữ liệu.
- Có thể dùng công cụ trực quan như Scipy, Matplotlib, Seaborn và Plotly để biểu diễn kết quả.
- So sánh hiệu năng xử lý khi sử dụng Hadoop với phương pháp xử lý truyền thống.
- Đề xuất hướng tối ưu và mở rộng hệ thống trong tương lai.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về dữ liệu lớn (Big Data)

2.1.1. Khái niệm

Dữ liệu lớn (Big Data) là thuật ngữ dùng để mô tả khối lượng dữ liệu khổng lồ, đa dạng và được tạo ra với tốc độ cao đến mức các hệ thống xử lý dữ liệu truyền thống không thể lưu trữ, quản lý và phân tích hiệu quả.

2.1.2. Đặc trưng của dữ liệu lớn – mô hình 5V

- Volume (Khối lượng) – Lượng dữ liệu rất lớn (từ terabyte đến petabyte).
- Velocity (Tốc độ) – Dữ liệu được sinh ra và cần xử lý nhanh (real-time hoặc near real-time).
- Variety (Đa dạng) – Bao gồm dữ liệu có cấu trúc (structured), bán cấu trúc (semi-structured) và phi cấu trúc (unstructured).
- Veracity (Độ tin cậy) – Dữ liệu có thể chứa sai lệch, nhiễu, cần làm sạch trước khi phân tích.
- Value (Giá trị) – Mục tiêu cuối cùng là trích xuất được thông tin có giá trị phục vụ ra quyết định.

2.2. Khai thác dữ liệu (Data Mining)

2.2.1. Khái niệm

Khai thác dữ liệu (Data Mining) là quá trình tìm ra mẫu, mối quan hệ, xu hướng hoặc tri thức tiềm ẩn trong các tập dữ liệu lớn, sử dụng các kỹ thuật thống kê, học máy, và cơ sở dữ liệu.

2.2.2. Các bước cơ bản trong quá trình khai thác dữ liệu

1. Thu thập dữ liệu (Data Collection). Có thể xây dựng phiếu khảo sát để thu thập dữ liệu.
2. Tiền xử lý dữ liệu (Data Preprocessing) – làm sạch, tích hợp, chọn lọc dữ liệu, giảm thiểu dữ liệu và phân tách dữ liệu.
3. Tổng hợp dữ liệu (Data Integration) – tổng hợp các tập dữ liệu từ nhiều nguồn thành một tập duy nhất.
4. Chuyển đổi dữ liệu (Transformation) – chuẩn hóa hoặc rút trích đặc trưng.
5. Khai thác mẫu (Pattern Mining) – sử dụng các thuật toán như Apriori, K-Means, Decision Tree, v.v.
6. Đánh giá và diễn giải kết quả (Evaluation & Interpretation)
7. Trực quan hóa dữ liệu (Data Visualization) – quá trình biểu diễn dữ liệu dưới dạng biểu đồ, đồ thị.

2.3. Giới thiệu về Hadoop

2.3.1. Khái niệm

Apache Hadoop là một nền tảng mã nguồn mở cho phép lưu trữ và xử lý dữ liệu lớn theo mô hình phân tán trên nhiều máy tính (clusters) được phát triển dựa trên ý tưởng MapReduce của Google.

Công nghệ Hadoop là nền tảng chủ chốt cho việc lưu trữ, quản lý và khai thác dữ liệu lớn, hỗ trợ mở rộng quy mô linh hoạt, xử lý song song và đảm bảo độ tin cậy cao. Việc hiểu rõ cấu trúc và cơ chế hoạt động của Hadoop là nền tảng để thực hiện hiệu quả các bài toán khai thác tri thức từ dữ liệu lớn (Big Data Mining).

2.3.2. Các thành phần chính

- HDFS (Hadoop Distributed File System): Hệ thống tệp phân tán, cho phép lưu trữ dữ liệu trên nhiều node. Gồm 2 loại node:
 - NameNode: Quản lý metadata (thông tin về file, vị trí block).
 - DataNode: Lưu trữ dữ liệu thực tế
- MapReduce: Mô hình lập trình giúp xử lý dữ liệu lớn song song. Gồm hai pha:
 - Map: Xử lý và ánh xạ dữ liệu thành cặp key–value.
 - Reduce: Tổng hợp, nhóm và tính toán kết quả cuối.
- YARN (Yet Another Resource Negotiator): Quản lý tài nguyên và lập lịch thực thi các tác vụ (job scheduling).
- Hadoop Common: Bộ thư viện và tiện ích hỗ trợ các module khác hoạt động.

2.3.3. Kiến trúc hoạt động của Hadoop

- Hadoop sử dụng mô hình Master–Slave:
 - NameNode (Master): Quản lý metadata và điều phối.
 - DataNodes (Slaves): Lưu trữ và xử lý dữ liệu thực tế.
- Dữ liệu được chia thành nhiều block (mặc định 128 MB), lưu trữ trên nhiều node để tăng tính sẵn sàng (availability) và khả năng chịu lỗi (fault tolerance).

2.3.4. Cơ chế xử lý dữ liệu trong Hadoop – MapReduce

- Pha Map: Chia dữ liệu đầu vào thành các phần nhỏ, xử lý song song trên các node.
- Pha Shuffle & Sort: Các kết quả trung gian được gom nhóm (group by key) và sắp xếp trước khi đưa vào Reduce.
- Pha Reduce: Tổng hợp kết quả cuối cùng từ các cặp key–value có cùng key.

2.3.5. Ưu điểm và hạn chế của Hadoop

- Ưu điểm
 - Xử lý dữ liệu lớn phân tán, song song.
 - Khả năng mở rộng (scalability) cao.

- Chịu lỗi tốt.
- Chi phí thấp (dùng phần cứng phổ thông).
- Hạn chế
 - Không phù hợp cho xử lý thời gian thực.
 - Việc lập trình MapReduce khá phức tạp.
 - Yêu cầu kiến thức quản trị cluster.
 - Khó khăn trong việc tiếp cận

2.3.6. Các công cụ mở rộng của hệ sinh thái Hadoop

1. Hive: Truy vấn dữ liệu lớn bằng ngôn ngữ tương tự SQL (HiveQL).
2. Pig: Xử lý dữ liệu bằng script (Pig Latin).
3. HBase: Cơ sở dữ liệu NoSQL chạy trên HDFS.
4. Sqoop: Kết nối và chuyển dữ liệu giữa Hadoop và cơ sở dữ liệu quan hệ (RDBMS).
5. Flume: Thu thập dữ liệu log từ nhiều nguồn.
6. Spark: Xử lý dữ liệu trong bộ nhớ (in-memory), nhanh hơn MapReduce.

2.4. Giới thiệu về Spark & PySpark

2.4.1. Khái niệm Apache Spark

Apache Spark là một nền tảng xử lý dữ liệu lớn trong bộ nhớ (in-memory computing), được phát triển để giải quyết các hạn chế về tốc độ của MapReduce trong Hadoop. Spark hỗ trợ xử lý song song và phân tán, có khả năng mở rộng linh hoạt trên cluster, thích hợp cho các ứng dụng streaming, batch processing, machine learning, graph processing.

2.4.2. Kiến trúc Spark

- Driver Program: Chịu trách nhiệm quản lý ứng dụng Spark, lập kế hoạch, phân phối tác vụ đến các executor.
- Cluster Manager: Quản lý tài nguyên của cluster, ví dụ: YARN, Mesos hoặc Spark Standalone.
- Executor: Thực thi các tác vụ trên các node của cluster và lưu trữ dữ liệu trong bộ nhớ (Memory) hoặc đĩa (Disk).
- RDD (Resilient Distributed Dataset): Là khối dữ liệu phân tán cơ bản trong Spark, có khả năng chịu lỗi, lưu trữ song song trên cluster. Hỗ trợ các phép biến đổi (transformations) và hành động (actions).
- DataFrame & Dataset API: Bảng dữ liệu phân tán, tương tự Pandas DataFrame, hỗ trợ tối ưu hóa query là phiên bản kiểu dữ liệu an toàn cho ngôn ngữ Scala/Java.

2.4.3. Khái niệm về PySpark – Spark cho Python

- PySpark là API của Spark cho Python, cho phép viết chương trình xử lý dữ liệu lớn bằng Python mà vẫn tận dụng sức mạnh phân tán của Spark.
- PySpark hỗ trợ các module quan trọng:
 - pyspark.sql: làm việc với DataFrame, SQL queries, và Hive.
 - pyspark.ml: các thuật toán machine learning trên dữ liệu lớn.
 - pyspark.streaming: xử lý dữ liệu streaming real-time.
 - pyspark.rdd: thao tác trực tiếp với RDD.

2.4.4. Ưu điểm của PySpark trong khai thác dữ liệu lớn

- Xử lý dữ liệu phân tán với tốc độ cao, giảm thời gian tính toán so với MapReduce truyền thống.
- Dễ dàng tích hợp với Python ecosystem (NumPy, Pandas, Matplotlib, Scikit-learn).
- Hỗ trợ DataFrame và SQL giúp thao tác dữ liệu gần giống với Pandas, quen thuộc với lập trình viên.
- Khả năng xây dựng pipeline ML và phân tích dữ liệu streaming trên cùng một framework.
- Giảm thiểu việc đọc/ghi đĩa nhiều lần nhờ lưu dữ liệu trung gian trong bộ nhớ.

CHƯƠNG 3: GIỚI THIỆU VỀ BỘ DỮ LIỆU

Indicators of Heart Disease 2022

3.1. Tổng quan về cơ sở dữ liệu

3.1.1. Giới thiệu về tập dữ liệu

Đây là bộ dữ liệu thể hiện các chỉ số chính của bệnh tim - Dữ liệu khảo sát CDC hàng năm năm 2022 của hơn 400.000 người lớn liên quan đến tình trạng sức khỏe của họ.

Theo CDC, bệnh tim là nguyên nhân tử vong hàng đầu đối với những người thuộc hầu hết các chủng tộc tại Hoa Kỳ (người Mỹ gốc Phi, người Mỹ bản địa và người bản địa Alaska, và người da trắng). Khoảng một nửa số người Mỹ (47%) có ít nhất 1 trong 3 yếu tố nguy cơ chính gây bệnh tim: huyết áp cao, cholesterol cao và hút thuốc. Các chỉ số quan trọng khác bao gồm tình trạng tiểu đường, béo phì (BMI cao), không vận động đủ hoặc uống quá nhiều rượu. Việc xác định và ngăn ngừa các yếu tố có tác động lớn nhất đến bệnh tim là rất quan trọng trong chăm sóc sức khỏe. Đổi lại, sự phát triển trong điện toán cho phép ứng dụng các phương pháp học máy để phát hiện "các mẫu" trong dữ liệu có thể dự đoán tình trạng của bệnh nhân.

3.1.2. Nguồn thu thập

Bộ dữ liệu ban đầu đến từ CDC và là một phần chính của Hệ thống giám sát yếu tố nguy cơ hành vi (BRFSS), thực hiện các cuộc khảo sát qua điện thoại hàng năm để thu thập dữ liệu về tình trạng sức khỏe của cư dân Hoa Kỳ. Theo mô tả của CDC: "Được thành lập vào năm 1984 với 15 tiểu bang, BRFSS hiện thu thập dữ liệu ở tất cả 50 tiểu bang, Quận Columbia và ba vùng lãnh thổ của Hoa Kỳ. BRFSS hoàn thành hơn 400.000 cuộc phỏng vấn người lớn mỗi năm, khiến đây trở thành hệ thống khảo sát sức khỏe liên tục lớn nhất trên thế giới. Bộ dữ liệu gần đây nhất bao gồm dữ liệu từ năm 2023. Trong bộ dữ liệu này, tôi nhận thấy nhiều yếu tố (câu hỏi) ảnh hưởng trực tiếp hoặc gián tiếp đến bệnh tim, vì vậy tôi quyết định chọn các biến có liên quan nhất từ đó. Tôi cũng quyết định chia sẻ với bạn hai phiên bản của bộ dữ liệu gần đây nhất: có NaN và không có NaN.

3.1.3. Tổng quan về bảng dữ liệu

Như đã mô tả ở trên, tập dữ liệu gốc gồm gần 300 cột đã được giảm xuống còn 40 cột và hơn 400000 dòng. Ngoài EDA cổ điển, tập dữ liệu này có thể được sử dụng để áp dụng một số phương pháp học máy, đặc biệt là các mô hình phân loại (hồi quy logistic, SVM, rùng ngẫu nhiên, v.v.).

Bộ dữ liệu được lấy từ Kaggle: **Indicators of Heart Disease 2022**

<https://www.kaggle.com/datasets/indicators-of-heart-disease>

3.2. Giới thiệu thuộc tính

3.2.1. Thuộc tính danh nghĩa

State (Bang, tiểu Bang)			
Washington (6%) New York (4%) Other (90%)	Null	Unique	Mode
	0	54	Washington
Sex (Giới tính)			
Female (53%) Male (47%)	Null	Unique	Mode
	0	2	Female
LastCheckupTime (Khoảng thời gian kể từ lần kiểm tra sức khỏe tổng quát gần nhất của người tham gia)			
Within past year (anytime less than 12 months ago) (79%) Within past 2 years (1 year but less than 2 years ago) (9%) Other (12%)	Null	Unique	Mode
	8308	4	Within past year (anytime less than 12 months ago)
SmokerStatus (Cho biết tình trạng hút thuốc của người tham gia)			
Never smoked (55%) Former smoker (26%) Other (19%)	Null	Unique	Mode
	35.5k	4	Never smoked

<u>ECigaretteUsage</u> (Cho biết người tham gia có sử dụng thuốc lá điện tử (e-cigarettes) hay không)			
Never used e-cigarettes in my entire life (70%)	Null	Unique	Mode
	35.7k	4	Never used e-cigarettes in my entire life
	Other (13%)		
<u>RaceEthnicityCategory</u> (Cho biết nhóm chủng tộc và dân tộc của người tham gia)			
White only, Non-Hispanic (72%)	Null	Unique	Mode
	14.1k	5	White only, Non-Hispanic
	Other (18%)		
<u>TetanusLast10Tdap</u> (Cho biết liệu người tham gia có tiêm vắc-xin uốn ván (Tetanus) hoặc vắc-xin Tdap (vắc-xin phòng uốn ván, bạch hầu, ho gà) trong vòng 10 năm qua hay không)			
No, did not receive any tetanus shot in the past 10 years (27%)	Null	Unique	Mode
	82.5k	4	No, did not receive any tetanus shot in the past 10 years
	Other (47%)		

Bảng 3.2-1. Bảng thuộc tính danh nghĩa

3.2.2. Thuộc tính thứ tự

<u>GeneralHealth</u> (Tình trạng sức khỏe)			
Very good (33%)	Null	Unique	Mode
	1198	5	Very good
	Good (32%)		
Other (34%)			

<u>AgeCategory</u> (Cho biết nhóm độ tuổi của người tham gia)			
Age 65 to 69 (11%)	Null	Unique	Mode
Age 60 to 64 (10%)			
Other (79%)	9079	13	Age 65 to 69

Bảng 3.2-2. Bảng thuộc tính thứ tự

3.2.3. Thuộc tính kiểu số

<u>PhysicalHealthDays</u> (Số ngày trong 30 ngày gần nhất mà người tham gia khảo sát cảm thấy sức khỏe thể chất của họ không tốt)							
Null	Mean	Std.Dev	Min	25%	50%	75%	Max
10.9k	4.35	8.69	0	0	0	3	30
<u>MentalHealthDays</u> (Số ngày trong 30 ngày gần nhất mà người tham gia khảo sát cảm thấy bị đau tim)							
Null	Mean	Std.Dev	Min	25%	50%	75%	Max
9067	4.38	8.39	0	0	0	5	30
<u>SleepHours</u> (Thời gian ngủ)							
Null	Mean	Std.Dev	Min	25%	50%	75%	Max
5453	7.02	1.5	1	6	7	8	24
<u>HeightInMeters</u> (Cho biết chiều cao của người tham gia tính bằng đơn vị mét)							
Null	Mean	Std.Dev	Min	25%	50%	75%	Max
28.7k	1.7	0.11	0.91	1.63	1.7	1.78	2.41

<u>WeightInKilograms</u> (Cho biết cân nặng của người tham gia tính bằng đơn vị kilogram)							
Null	Mean	Std.Dev	Min	25%	50%	75%	Max
42.1k	83.1	21.4	22.7	68	80.7	95.3	293
<u>BMI</u> (Chỉ số khối cơ thể (Body Mass Index), là một chỉ số dùng để đánh giá mức độ béo phì của người tham gia dựa trên cân nặng và chiều cao)							
Null	Mean	Std.Dev	Min	25%	50%	75%	Max
48.8k	28.5	6.55	12	24.1	27.4	31.8	99.6

Bảng 3.2-3. Bảng thuộc tính kiểu số

3.2.4. Thuộc tính nhị phân

<u>PhysicalActivities</u> (Cho biết người tham gia có hoạt động thể chất ngoài công việc thường ngày hay không)		
Tên giá trị	Số lượng	Tỷ lệ (%)
True	338k	76
False	106k	24
Null	1093	0
<u>HadHeartAttack</u> (Cho biết người tham gia có từng bị đau tim (heart attack) hay chưa)		
Tên giá trị	Số lượng	Tỷ lệ (%)
True	25.1k	6
False	417k	94
Null	3065	0

HadAngina (Cho biết người tham gia có từng bị đau thắt ngực (angina) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	26.6k	6
False	414k	93
Null	4405	1

HadStroke (Cho biết người tham gia có từng bị đột quy (stroke) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	19.2k	4
False	424k	95
Null	1557	1

HadAsthma (Cho biết người tham gia có từng được chẩn đoán mắc bệnh hen suyễn (asthma) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	66.7k	15
False	377k	85
Null	1773	0

HadSkinCancer (Cho biết người tham gia có từng được chẩn đoán mắc ung thư da (skin cancer) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	35.5k	8
False	407k	91
Null	3143	1

HadCOPD (Cho biết người tham gia có từng được chẩn đoán mắc bệnh phổi tắc nghẽn mạn tính (COPD) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	35.7k	8
False	407k	91
Null	2219	1

HadDepressiveDisorder (Cho biết người tham gia có từng được chẩn đoán mắc rối loạn trầm cảm (depressive disorder) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	91.4k	21
False	351k	79
Null	2812	0

HadKidneyDisease (Cho biết người tham gia có từng được chẩn đoán mắc bệnh thận hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	20.3k	5
False	423k	95
Null	1926	0

HadArthritis (Cho biết người tham gia có từng được chẩn đoán mắc bệnh viêm khớp (arthritis) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	151k	34
False	291k	65
Null	2633	1

HadDiabetes (Cho biết người tham gia có từng được chẩn đoán mắc bệnh tiểu đường (diabetes) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	151k	34
False	291k	65
Null	2633	3

DifficultyWalking (Cho biết người tham gia có gặp khó khăn trong việc đi lại hoặc di chuyển hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	68.1k	15
False	353k	79
Null	24k	5

DifficultyDressingBathing (Cho biết người tham gia có gặp khó khăn trong việc mặc quần áo hoặc tắm rửa hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	16.8k	4
False	404k	91
Null	23.9k	5

DifficultyErrands (Cho biết người tham gia có gặp khó khăn trong việc thực hiện các công việc vặt hoặc công việc hàng ngày như đi chợ, mua sắm, hoặc các hoạt động sinh hoạt khác hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	32.4k	7
False	387k	87
Null	25.7k	6

ChestScan (Cho biết người tham gia có thực hiện chụp X-quang ngực (chụp ngực) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	166k	37
False	223k	50
Null	56k	13

AlcoholDrinkers (Cho biết người tham gia có phải là người uống rượu hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	211k	47
False	188k	42
Null	46.6k	11

HIVTesting (Cho biết người tham gia có thực hiện xét nghiệm HIV hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	129k	29
False	250k	56
Null	66.1k	15

FluVaxLast12 (Cho biết liệu người tham gia có tiêm vắc-xin cúm trong vòng 12 tháng qua hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	209k	47
False	189k	42
Null	47.1k	11

PneumoVaxEver (Cho biết liệu người tham gia có bao giờ tiêm vắc-xin phòng viêm phổi (Pneumococcal vaccine) hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	152k	34
False	216k	48
Null	77k	18

DeafOrHardOfHearing (Cho biết người tham gia có bị điếc hoặc nghe kém hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	38.9k	9
False	386k	87
Null	20.6k	4

BlindOrVisionDifficulty (Cho biết người tham gia có bị mù hoặc gặp khó khăn về thị lực hay không)

Tên giá trị	Số lượng	Tỷ lệ (%)
True	23.7k	5
False	400k	90
Null	21.6k	5

<u>DifficultyConcentrating</u> (Cho biết người tham gia có gặp khó khăn trong việc tập trung hay không)		
Tên giá trị	Số lượng	Tỷ lệ (%)
True	50.1k	11
False	371k	83
Null	24.2k	5

<u>HighRiskLastYear</u> (Cho biết liệu người tham gia có có nguy cơ cao đối với bệnh tim mạch trong năm trước đó hay không)		
Tên giá trị	Số lượng	Tỷ lệ (%)
True	17.2k	4
False	377k	85
Null	77k	18

<u>CovidPos</u> (Cho biết liệu người tham gia có dương tính với COVID-19 hay không)		
Tên giá trị	Số lượng	Tỷ lệ (%)
True	17.2k	25
False	377k	61
Null	77k	18

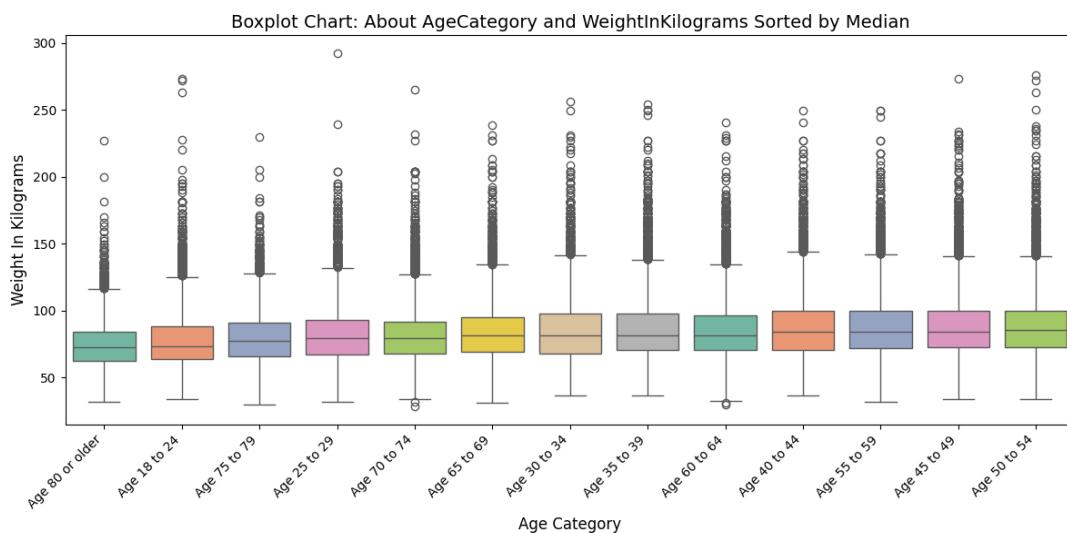
Bảng 3.2-4. Bảng thuộc tính nhị phân

CHƯƠNG 4: PHÂN TÍCH THỐNG KÊ TRÊN CSDL

4.1. Tìm hiểu dữ liệu

4.1.1. Thực hiện vẽ các đồ thị dựa trên thuộc tính tùy chọn

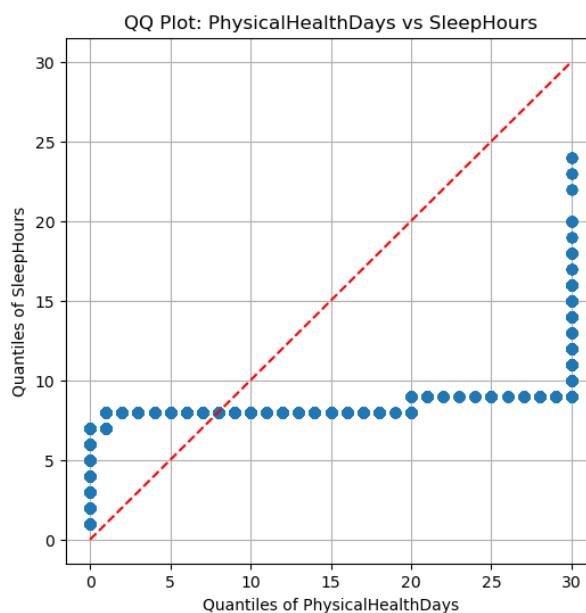
4.1.1.1. Boxplot Chart



Hình 4.1-1. Hình Boxplot Chart về lứa tuổi và cân nặng cơ thể

Đồ thị boxplot này cho thấy xu hướng biến thiên của cân nặng theo từng nhóm tuổi. Khi tuổi tăng lên, cân nặng trung vị có xu hướng tăng cho đến tuổi trung niên, sau đó giảm nhẹ ở tuổi cao hơn. Sự phân tán lớn hơn trong nhóm tuổi trung niên phản ánh sự đa dạng về lối sống và sức khỏe trong độ tuổi này - yếu tố có liên hệ trực tiếp đến nguy cơ mắc bệnh tim mạch.

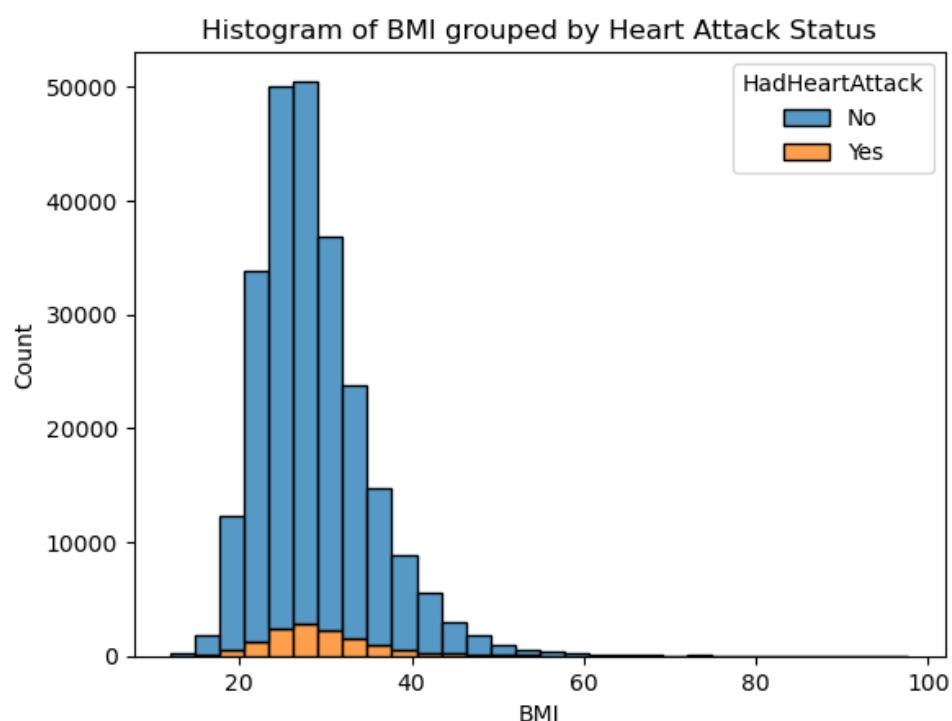
4.1.1.2. QQ Plot



Hình 4.1-2. QQ Plot về sức khỏe thường ngày và giờ giấc ngủ

Đồ thị QQ Plot cho thấy phân phối của số ngày sức khỏe thể chất kém (PhysicalHealthDays) và số giờ ngủ trung bình (SleepHours). Ở các phân vị thấp, hai biến có xu hướng tương đồng, nhưng ở các phân vị cao, số ngày sức khỏe thể chất kém tăng mạnh trong khi số giờ ngủ không tăng tương ứng. Điều này gợi ý rằng người có vấn đề sức khỏe thể chất kéo dài thường có xu hướng ngủ ít hơn, phản ánh mối quan hệ tiêu cực giữa chất lượng sức khỏe thể chất và giấc ngủ - yếu tố quan trọng trong nghiên cứu về bệnh tim mạch và sức khỏe tổng thể.

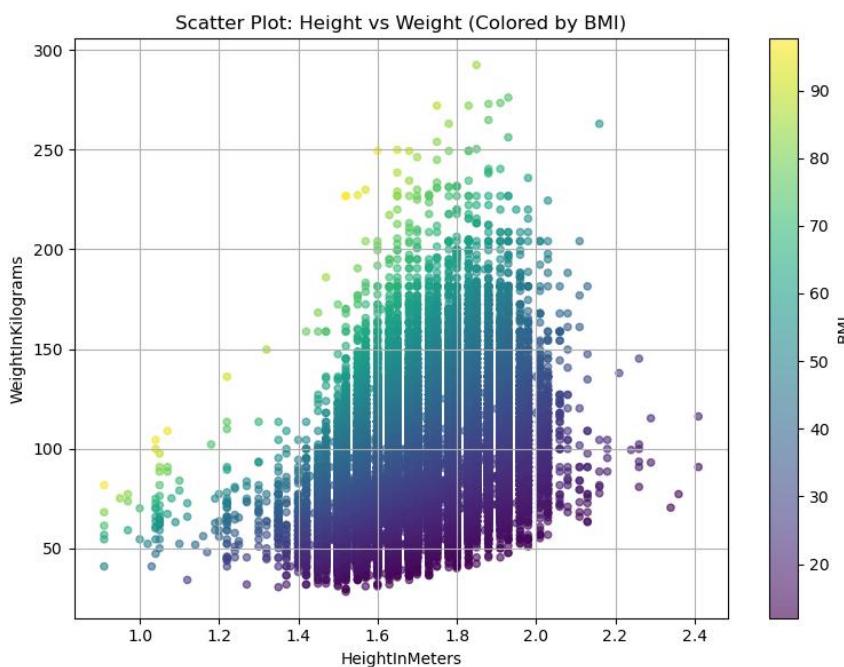
4.1.1.3. Histogram Plot



Hình 4.1-3. Biểu đồ Histogram giữa BMI và tỷ lệ người bị đau tim

Biểu đồ histogram cho thấy mối tương quan giữa chỉ số BMI và tình trạng đau tim. Tỷ lệ người từng bị đau tim có xu hướng tăng ở nhóm có BMI cao, gợi ý rằng béo phì có thể là một yếu tố nguy cơ quan trọng đối với bệnh tim mạch.

4.1.1.4. Scatter Plot

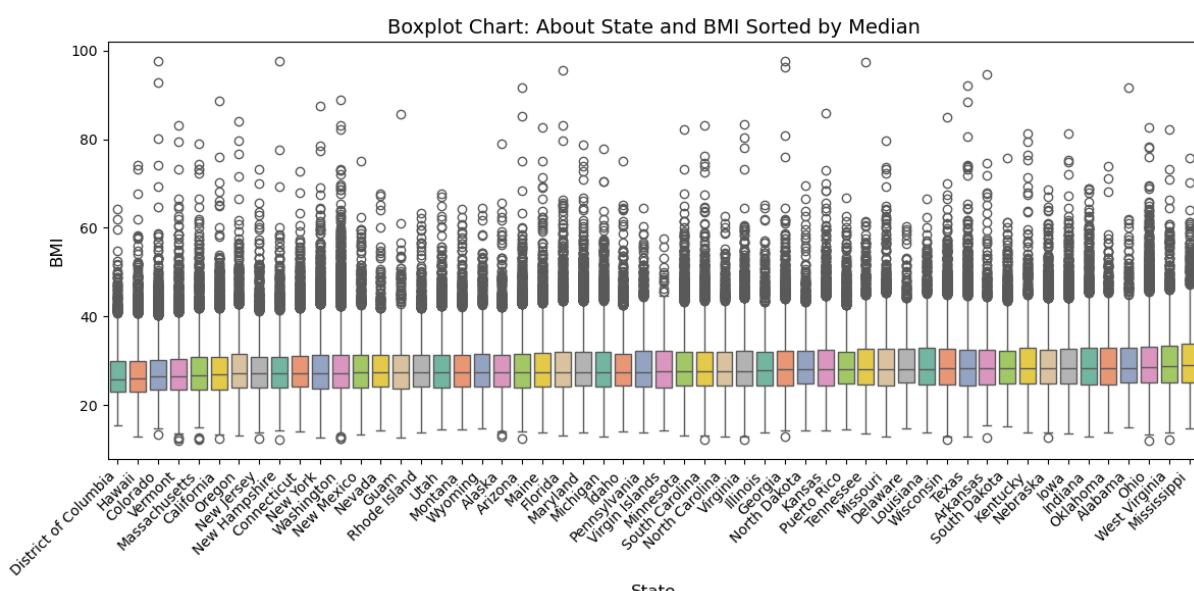


Hình 4.1-4. Biểu đồ Scatter Plot giữa chiều cao, cân nặng và chỉ số BMI

Biểu đồ phân tán giữa chiều cao và cân nặng, với màu sắc biểu thị chỉ số BMI, cho thấy mối quan hệ tuyến tính dương giữa hai biến. Người có cân nặng cao nhưng chiều cao thấp thường có BMI cao, thể hiện nguy cơ béo phì cao hơn. Trong khi đó, những người có chiều cao cao và cân nặng tương ứng có xu hướng duy trì BMI ở mức bình thường. Biểu đồ này giúp trực quan hóa mối liên hệ giữa thể hình và chỉ số sức khỏe cơ bản (BMI), hỗ trợ nhận định về yếu tố nguy cơ tim mạch.

4.1.2. Vẽ đồ thị với các thuộc tính danh nghĩa

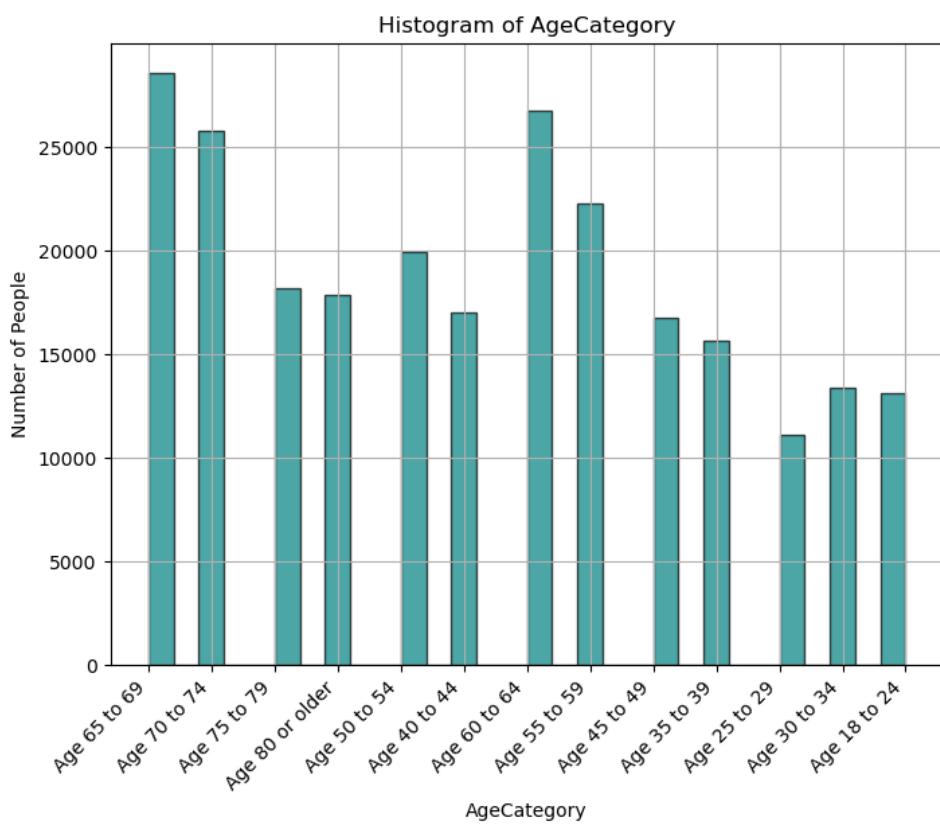
4.1.2.1. Boxplot Chart



Hình 4.1-5. Boxplot chart giữa chỉ số BMI và khu vực quốc gia

Biểu đồ hộp cho thấy sự khác biệt về chỉ số khối cơ thể (BMI) giữa các tiểu bang ở Hoa Kỳ. Một số bang có xu hướng dân số có BMI thấp hơn, trong khi ở một số bang khác, mức trung vị BMI cao hơn, phản ánh tình trạng thừa cân phổ biến hơn. Mức độ phân tán BMI cũng thay đổi tùy bang, có thể liên quan đến sự khác biệt trong chế độ ăn uống, thu nhập, hoặc mức độ hoạt động thể chất của cư dân.

4.1.2.2. Histogram Plot



Hình 4.1-6. Histogram plot về sự phân bố trong từng lứa tuổi

Biểu đồ histogram thể hiện phân bố số người tham gia theo độ tuổi. Có thể thấy rằng phần lớn người trong bộ dữ liệu nằm trong các nhóm tuổi trung niên và cao tuổi (từ 50 tuổi trở lên). Điều này hợp lý vì bệnh tim thường gặp hơn ở các độ tuổi này. Số người ở nhóm tuổi trẻ ít hơn, phản ánh sự phân bố không đều trong khảo sát và cũng phù hợp với thực tế về tỷ lệ mắc bệnh tim theo tuổi.

4.2. Tiền xử lý dữ liệu

4.2.1. Clone dataset

Em thu thập và lấy dữ liệu về từ trang web [kaggle.com](https://www.kaggle.com). Tên của bộ dữ liệu là: *Indicators of Heart Disease (2022 UPDATE)*. Em sử dụng thư viện Kaggle để clone dữ liệu về máy tính cá nhân.

```

import kagglehub
# Download latest version
path = kagglehub.dataset_download("kamilpytlak/personal-key-
indicators-of-heart-disease")
print("Path to dataset files:", path)

```

4.2.2. Import thư viện và model cần sử dụng

1. json: Thư viện chuẩn để làm việc với dữ liệu JSON (JavaScript Object Notation), thường dùng để đọc/ghi dữ liệu có cấu trúc dạng từ khóa-giá trị.
2. pandas (pd): Thư viện mạnh mẽ để xử lý và phân tích dữ liệu, đặc biệt là dữ liệu dạng bảng (DataFrame).
3. numpy (np): Thư viện xử lý tính toán số học hiệu năng cao, hỗ trợ mảng n chiều và các hàm toán học.
4. csv: Thư viện chuẩn dùng để đọc và ghi các tệp CSV (Comma-Separated Values).
5. random: Thư viện chuẩn để tạo ra các số ngẫu nhiên và thực hiện thao tác ngẫu nhiên như chọn mẫu, trộn danh sách,...
6. matplotlib.pyplot (plt): Thư viện vẽ biểu đồ cơ bản trong Python, hỗ trợ biểu đồ đường, cột, scatter, histogram,...
7. scipy.stats: Mô-đun thuộc SciPy để thực hiện các phép thống kê như kiểm định giả thuyết, tính phân phối xác suất,...
8. seaborn (sns): Thư viện vẽ biểu đồ dựa trên matplotlib, cung cấp các biểu đồ thống kê đẹp và dễ dùng.
9. joypy.joyplot: Dùng để vẽ biểu đồ “joyplot” (biểu đồ density xếp chồng), thường dùng để hiển thị phân phối của nhiều nhóm.
10. matplotlib.cm.viridis: Bảng màu (colormap) “Viridis” từ matplotlib, thường dùng để tô màu theo cường độ trong các biểu đồ heatmap, scatter,...
11. plotly.express (px): Thư viện vẽ biểu đồ tương tác, hỗ trợ nhanh chóng tạo biểu đồ động và đẹp mắt.
12. squarify: Thư viện để vẽ biểu đồ treemap – hiển thị dữ liệu phân cấp dưới dạng hình chữ nhật lồng nhau.
13. numpy.float64: Kiểu dữ liệu số thực dấu chấm động 64-bit trong NumPy, thường dùng để đảm bảo độ chính xác cao trong tính toán.
14. pyspark.sql: Thư viện cần thiết để chạy các chương trình được viết bằng PySpark.

4.2.3. Xử lý dữ liệu đầu vào

Vì dữ liệu không có trường id nên ta chủ động tạo để dễ dàng so sánh các phần tử với nhau. Đầu tiên khởi tạo cột “id” với các giá trị tăng dần từ 0. Sau đó tạo cột mới “PersonID” với giá trị là ký tự “Person” + giá trị của từng hàng trong cột “id” vừa tạo. Sau đó thực hiện xóa đi cột “id” tạm.

PersonID	State	Sex	...	HighRiskLastYear	CovidPos	id
Person0	Alabama	Female	...	No	No	0
Person1	Alabama	Female	...	No	No	1
Person2	Alabama	Female	...	No	Yes	2
Person3	Alabama	Female	...	No	No	3
Person4	Alabama	Female	...	No	No	4
Person5	Alabama	Male	...	No	No	5

Hình 4.2-1. Hình xử lý dữ liệu đầu vào

4.2.4. Chuyển đổi và phân tách dữ liệu

4.2.4.1. Yêu cầu

- Kiểu danh mục (ví dụ như các mùa: Xuân, Hạ, Thu, Đông): chuyển thành kiểu số nguyên.
- Kiểu thứ tự (ví dụ như: Giới, khá, trung bình, yếu): chuyển thành kiểu số nguyên.
- Kiểu boolean: khi dữ liệu có từ 5 field kiểu này trở lên, chuyển 5 field thành 1 field kiểu số.
- Kiểu chuỗi: trong field kiểu chuỗi lại chứa danh sách các chuỗi con có ý nghĩa (ví dụ Tên hàng Đã mua trong đó gồm tên các sản phẩm đã mua trong đơn hàng như: bột giặt, khăn tắm, chén, khăn tắm): cần tách tên các sản phẩm ra khỏi field này để có nhiều field mới. Sau đó, do có thể có rất nhiều field mới nên cần chuyển đổi tất cả các field mới này trở lại thành 1 field duy nhất có kiểu là int.
- Trong quá trình chuyển đổi cần tạo ra các table làm trung gian, sau đó tìm cách gộp các table vừa phát sinh có cùng cấu trúc, ý nghĩa để giảm bớt số lượng table thực tế sẽ dùng.

4.2.4.2. Triển khai

- Bộ dữ liệu của em có các cột thuộc tính danh nghĩa, thứ tự hay thuộc tính nhị phân. Các cột có khả năng chuyển đổi thành kiểu số nguyên.
- Các cột danh nghĩa và thứ tự:

```
[ "State", "GeneralHealth", "LastCheckupTime", "RemovedTeeth",
  "SmokerStatus", "ECigaretteUsage", "RaceEthnicityCategory",
  "AgeCategory", "TetanusLast10Tdap"]
```

- Các cột thuộc tính nhị phân:

```
[ 'PhysicalActivities', 'HadHeartAttack', 'HadAngina',
'HadStroke', 'HadAsthma', 'HadSkinCancer', 'HadCOPD',
'HadDepressiveDisorder', 'HadKidneyDisease', 'HadArthritis',
'HadDiabetes', 'DeafOrHardOfHearing',
'BlindOrVisionDifficulty', 'DifficultyConcentrating',
'DifficultyWalking', 'DifficultyDressingBathing',
'DifficultyErrands', 'ChestScan', 'AlcoholDrinkers',
'HIVTesting', 'FluVaxLast12', 'PneumoVaxEver',
'HighRiskLastYear', 'CovidPos' ]
```

- Em thực hiện tạo hai vòng for cho hai thuộc tính để duyệt từng cột. Sau đó em tạo dictionary mapping, thực hiện lấy dữ liệu trong cột khi đã loại bỏ trùng lặp, loại bỏ null, sắp xếp tăng dần và cuối cùng là đánh số từ 0.
- Sau đó em thực hiện tạo cột với <“name col”_mapped> để thực hiện map lại các giá trị.
- Việc chuyển đổi dữ liệu giúp chúng ta giảm thiểu được dữ liệu, việc xử lý dữ liệu dạng số sẽ dễ dàng hơn rất nhiều so với các kiểu dữ liệu khác. Việc áp dụng thuật toán phát triển sau này cũng sẽ yêu cầu các col dạng số.
- Dữ liệu trước khi chuyển đổi:

WeightInKilograms	BMI	AlcoholDrinkers	PneumoVaxEver	TetanusLast10Tdap	HighRiskLastYear	CovidPos
NULL NULL	No	No Yes, received tet...	No	No	No	No
68.04 26.57	No	No No, did not recei...	No	No	No	No
63.5 25.61	No	No	NULL	No	Yes	
63.5 23.3	No	Yes No, did not recei...	No	No	No	No
53.98 21.77	Yes	Yes No, did not recei...	No	No	No	No
84.82 26.08	No	Yes No, did not recei...	No	No	No	No
62.6 22.96	Yes	No No, did not recei...	No	No	No	No
73.48 27.81	No	Yes Yes, received tet...	No	No	No	No
NULL NULL	No	No Yes, received tet...	No	No	No	No
81.65 29.05	Yes	Yes No, did not recei...	No	No	No	No
74.84 29.23	No	Yes Yes, received tet...	No	No	No	No
59.42 23.21	No	Yes Yes, received tet...	No	No	No	No
85.28 28.59	Yes	No No, did not recei...	No	No	No	No
106.59 32.78	Yes	NULL	NULL	No	No	No
71.21 25.34	Yes	Yes Yes, received tet...	No	No	No	No
64.41 25.97	No	No Yes, received Tdap	No	No	No	No
61.23 24.69	No	Yes Yes, received Tdap	No	No	No	No
90.72 32.28	No	Yes No, did not recei...	No	Yes	No	No
65.77 24.89	No	Yes Yes, received tet...	No	No	No	No
66.22 22.87	Yes	NULL	NULL	NULL	NULL	NULL

Hình 4.2-2. Các cột dữ liệu trước khi chuyển đổi và phân tách

- Dữ liệu sau khi chuyển đổi

WeightInKilograms	BMI	TetanusLast10Tdap_mapped	AlcoholDrinkers_mapped	PneumoVaxEver_mapped	HighRiskLastYear_mapped	CovidPos_mapped
90.72 28.529842	3	0	0	0	0	0
68.04 26.57	1	0	0	0	0	0
63.5 25.61	NULL	0	0	0	0	1
63.5 23.3	1	0	1	1	0	0
53.98 21.77	1	1	1	1	0	0
84.82 26.08	1	0	1	0	0	0
62.6 22.96	1	1	0	0	0	0
73.48 27.81	3	0	1	0	0	0
90.72 28.529842	3	0	0	0	0	0
81.65 29.05	1	1	1	0	0	0
74.84 29.23	3	0	1	0	0	0
59.42 23.21	3	0	1	0	0	0
85.28 28.59	1	1	0	0	0	0
106.59 32.78	NULL	1	NULL	0	0	0
71.21 25.34	3	1	1	0	0	0
64.41 25.97	2	0	0	0	0	0
61.23 24.69	2	0	1	0	0	0
90.72 32.28	1	0	1	1	0	1
65.77 24.89	3	0	1	0	0	0
66.22 22.87	NULL	1	NULL	NULL	NULL	NULL

Hình 4.2-3. Dữ liệu sau khi chuyển đổi và phân tách

4.2.5. Điền giá trị còn thiếu

4.2.5.1. Đối với cột dữ liệu kiểu số

Điền giá trị NULL bằng giá trị lớn nhất của mode, median, mean nếu có nhiều mode thì chọn mode có giá trị cao nhất hoặc đối với kiểu số thực, có thể thực hiện làm tròn (đến phần nguyên hoặc phần ngàn, ...) trước khi tính mean/mode/median.

PersonID	PhysicalHealthDays	MentalHealthDays	SleepHours	HeightInMeters	WeightInKilograms	BMI
Person0	0.0	0.0	8.0	1.7026906	90.72 28.529842	
Person1	0.0	0.0	6.0	1.6	68.04 26.57	
Person2	2.0	3.0	5.0	1.57	63.5 25.61	
Person3	0.0	0.0	7.0	1.65	63.5 23.3	
Person4	2.0	0.0	9.0	1.57	53.98 21.77	
Person5	1.0	0.0	7.0	1.8	84.82 26.08	
Person6	0.0	0.0	7.0	1.65	62.6 22.96	
Person7	0.0	0.0	8.0	1.63	73.48 27.81	
Person8	0.0	0.0	6.0	1.7	90.72 28.529842	
Person9	1.0	0.0	7.0	1.68	81.65 29.05	
Person10	8.0	9.0	8.0	1.6	74.84 29.23	
Person11	0.0	0.0	6.0	1.6	59.42 23.21	
Person12	5.0	0.0	6.0	1.73	85.28 28.59	
Person13	0.0	0.0	8.0	1.8	106.59 32.78	
Person14	30.0	5.0	8.0	1.68	71.21 25.34	
Person15	0.0	0.0	8.0	1.57	64.41 25.97	
Person16	0.0	0.0	6.0	1.57	61.23 24.69	
Person17	0.0	15.0	6.0	1.68	90.72 32.28	
Person18	0.0	0.0	4.0	1.63	65.77 24.89	
Person19	0.0	0.0	6.0	1.7	66.22 22.87	

Hình 4.2-4. Các cột dữ liệu được điền bằng mean, mode, median

4.2.5.2. Đối với các kiểu dữ liệu còn lại

Đối với bộ dữ liệu này, kết quả thống kê cho thấy dữ liệu có nhiều null. Việc điền null vào những cột dữ liệu kiểu nhị phân hay thứ tự, danh nghĩa trở nên mất tính minh bạch và độ tin cậy cao.

Để kết quả từ những thuật toán phát triển sắp tới, hay sự phát triển và tích hợp dữ liệu trong tương lai. Em muốn đề xuất cách điền giá trị null như sau.

Với mảng kết quả thu được từ ma trận tương quan đo lường sự tương đồng hoặc mảng kết quả độ đo Cosin. Bản chất của chúng thể hiện khoảng cách liên kết giữa các

phần tử với nhau. Vậy nên việc lấy giá trị từ đối tượng có khoảng cách liên kết gần nhất để điền cho phần tử đang xét có giá trị null trở nên khả thi và vẫn giữ được tối ưu độ tin cậy của bộ dữ liệu. Điều đó tránh được các giá trị ngoại lệ, tuy nhiên quá trình cài đặt và triển khai hệ thống sẽ gặp nhiều bất cập và khó khăn.

Các bước chi tiết và cài đặt được nêu ở mục 2.4.

“Báo cáo chủ yếu demo trên số lượng ít hàng để có thể debug và sửa lỗi nhanh chóng nhằm tối ưu và nâng cao thuật toán. Khi chương trình đã ổn định sẽ tiến hành triển khai trên toàn bộ tập dữ liệu hơn 400 nghìn dòng.”

❖ Bộ dữ liệu trước khi điền null

PersonID	PhysicalHealthDays	MentalHealthDays	SleepHours	HeightInMeters	WeightInKilograms	BMI	State_mapped	GeneralHealth_mapped
Person0	0.0	0.0	8.0	1.7026906	90.72	28.529842	1	5
Person1	0.0	0.0	6.0	1.6	68.04	26.57	1	1
Person2	2.0	3.0	5.0	1.57	63.5	25.61	1	5
Person3	0.0	0.0	7.0	1.65	63.5	23.31	1	1
Person4	2.0	0.0	9.0	1.57	53.98	21.77	1	2

LastCheckupTime_mapped	SmokerStatus_mapped	ECigaretteUsage_mapped	RaceEthnicityCategory_mapped	AgeCategory_mapped
4	4	2	5	13
NULL	4	1	5	13
4	4	1	5	8
4	2	1	5	NULL
4	4	1	5	5

TetanusLast10Tdap_mapped	PhysicalActivities_mapped	HadHeartAttack_mapped	HadAngina_mapped	HadStroke_mapped	HadAsthma_mapped
3	0	0	0	0	0
1	0	0	0	0	0
NULL	1	0	0	0	0
1	1	0	0	0	1
1	1	0	0	0	0

HadSkinCancer_mapped	HadCOPD_mapped	HadDepressiveDisorder_mapped	HadKidneyDisease_mapped	HadArthritis_mapped	HadDiabetes_mapped
0	0	0	0	0	1
1	0	0	0	0	0
1	0	0	0	0	0
0	0	0	0	1	0
0	0	0	0	0	0

DeafOrHardOfHearing_mapped	BlindOrVisionDifficulty_mapped	DifficultyConcentrating_mapped	DifficultyWalking_mapped
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

DifficultyDressingBathing_mapped	DifficultyErrands_mapped	ChestScan_mapped	AlcoholDrinkers_mapped	HIVTesting_mapped	FluVaxLast12_mapped
0	0	0	0	0	1
0	0	0	0	0	0
0	0	0	0	0	0
0	0	1	0	0	1
0	0	1	1	1	0

PneumoVaxEver_mapped	HighRiskLastYear_mapped	CovidPos_mapped	Sex_mapped
0	0	0	0
0	0	0	0
0	0	1	0
1	0	0	0
1	0	0	0

Hình 4.2-5. Bộ dữ liệu trước khi điền null

❖ Mảng đo lường sự tương đồng thu được

```
[ [0.          0.          0.          0.          0.        ]
[0.49770485 0.          0.          0.          0.        ]
[0.44742737 0.49972253 0.          0.          0.        ]
[0.52205465 0.29113755 0.548321   0.          0.        ]
[0.675       0.46062849 0.44423929 0.26977869 0.        ]]
```

Hình 4.2-6. Mảng đo lường sự tương đồng (kết quả của mục 4.4)

❖ Sau khi điền null

PersonID	PhysicalHealthDays	MentalHealthDays	SleepHours	HeightInMeters	WeightInKilograms	BMI	State_mapped	GeneralHealth_mapped		
Person0	0.0	0.0	8.0	1.7026906	90.72	28.529842	1	5		
Person1	0.0	0.0	6.0	1.6	68.04	26.57	1	1		
Person2	2.0	3.0	5.0	1.57	63.5	25.61	1	5		
Person3	0.0	0.0	7.0	1.65	63.5	23.31	1	1		
Person4	2.0	0.0	9.0	1.57	53.98	21.77	1	2		
LastCheckupTime_mapped	SmokerStatus_mapped	ECigaretteUsage_mapped	RaceEthnicityCategory_mapped	AgeCategory_mapped	TetanusLast10Tdap_mapped	PhysicalActivities_mapped	HadHeartAttack_mapped	HadAngina_mapped	HadStroke_mapped	HadAsthma_mapped
4	4	2	5	13						
4	4	1	5	13						
4	4	1	5	8						
4	2	1	5	13						
4	4	1	5	5						

Hình 4.2-7. Dữ liệu sau khi đã điền null

4.3. Trực quan hóa dữ liệu

1. Pandas

- Mục đích: Pandas là một thư viện mạnh mẽ dùng để xử lý và phân tích dữ liệu. Pandas giúp bạn dễ dàng thao tác với dữ liệu dạng bảng (DataFrame) và thực hiện các thao tác như lọc, nhóm, tóm tắt, và tính toán.
- Ứng dụng trong trực quan hóa:
 - Chuẩn bị, làm sạch dữ liệu (ví dụ: xóa dữ liệu thiếu, thay đổi kiểu dữ liệu).
 - Tạo DataFrame từ các nguồn dữ liệu khác nhau (CSV, Excel, SQL, JSON).

2. Numpy

- Mục đích: Numpy là thư viện toán học mạnh mẽ, chủ yếu cung cấp các cấu trúc dữ liệu đa chiều và các phép toán toán học. Nó hỗ trợ nhanh chóng các tính toán khoa học và kỹ thuật.
- Ứng dụng trong trực quan hóa:
 - Tạo dữ liệu giả lập (ngẫu nhiên).
 - Thực hiện các phép toán mảng để xử lý dữ liệu trước khi vẽ đồ thị.

3. Matplotlib

- Mục đích: Matplotlib là thư viện chính để tạo các đồ thị cơ bản trong Python. Nó cung cấp các công cụ để vẽ biểu đồ 2D như line charts, bar charts, scatter plots, pie charts, và nhiều loại biểu đồ khác.
- Ứng dụng trong trực quan hóa:
 - Vẽ các biểu đồ cơ bản như line chart, scatter plot, bar chart, và histogram.
 - Tùy chỉnh các tham số trực quan như màu sắc, tiêu đề, nhãn trực.

4. Seaborn

- Mục đích: Seaborn được xây dựng dựa trên Matplotlib và cung cấp các biểu đồ đẹp mắt, dễ sử dụng hơn, đặc biệt là cho các loại dữ liệu thống kê như heatmaps, violin plots, và boxplots.
- Ứng dụng trong trực quan hóa:
 - Tạo các biểu đồ thống kê cao cấp như heatmap, pairplot, violin plot.
 - Làm việc tốt với dữ liệu Pandas và hỗ trợ các chủ đề màu sắc đẹp mắt.

5. Joyplot

- Mục đích: Joyplot là một thư viện cho phép tạo các Joy plots (hay còn gọi là Ridgeline plots), giúp hình dung phân bố của một biến trên các nhóm khác nhau.
- Ứng dụng trong trực quan hóa:
 - Trực quan hóa phân bố dữ liệu của nhiều nhóm trong cùng một biểu đồ.
 - Thường dùng để vẽ các biểu đồ phân bố nhiều chiều cho dữ liệu liên quan đến phân phối.

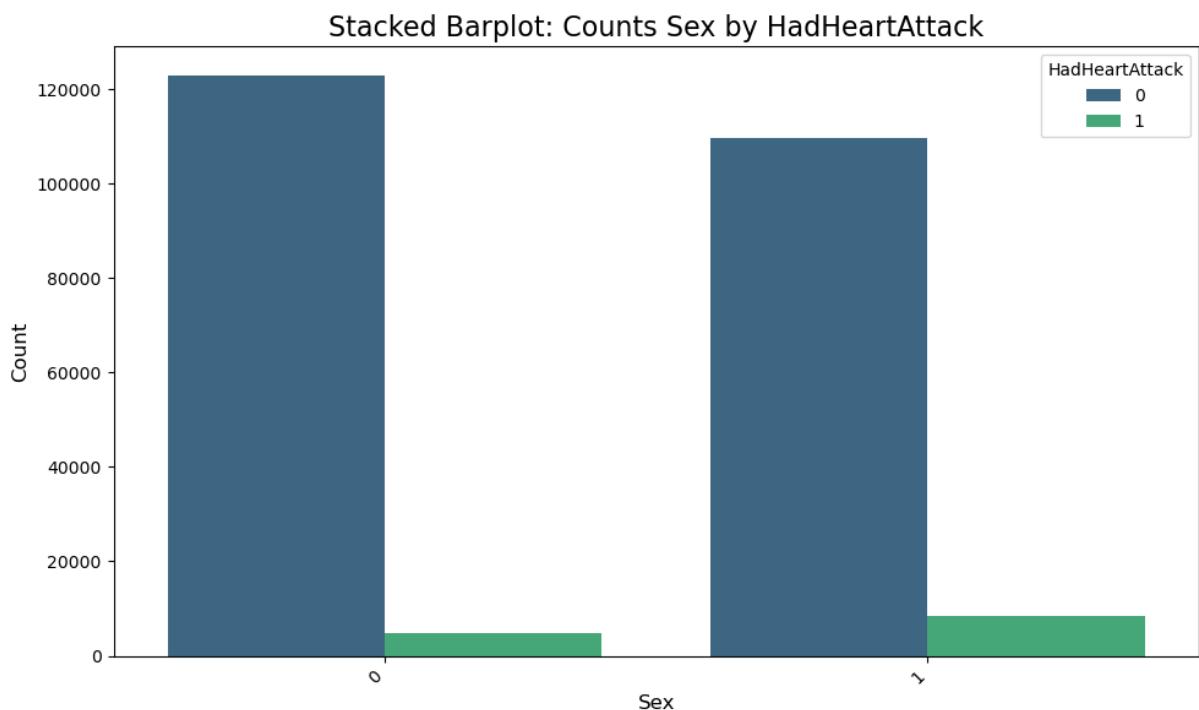
6. Matplotlib

- Mục đích: Thư viện cm của Matplotlib chứa các bảng màu. viridis là một trong các bảng màu phổ biến và được chọn vì sự dễ đọc và rõ ràng của nó.
- Ứng dụng trong trực quan hóa, sử dụng bảng màu để trực quan hóa các giá trị liên tục trong các biểu đồ.

7. Plotly

- Mục đích: Plotly là một thư viện mạnh mẽ để tạo các biểu đồ tương tác, đặc biệt là cho các biểu đồ 3D và trên bản đồ.
- Ứng dụng trong trực quan hóa:
 - Tạo các biểu đồ tương tác đẹp mắt với khả năng zoom, hover và click.
 - Phù hợp cho các ứng dụng web hoặc trực quan hóa dữ liệu trên nền tảng trực tuyến.

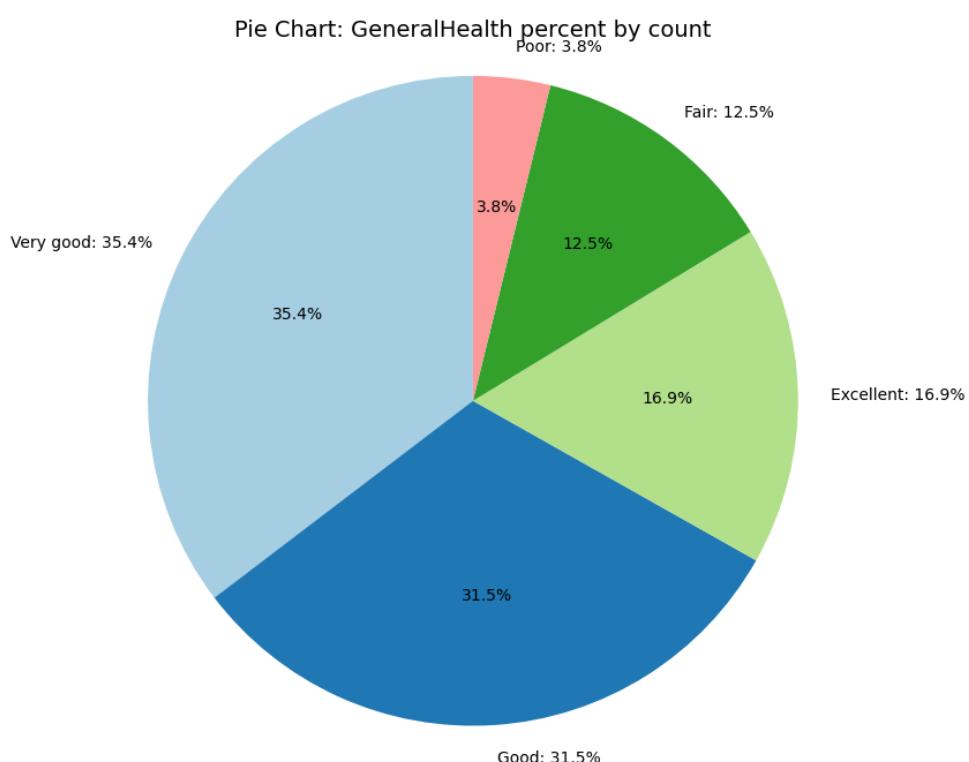
4.3.1. Barplot chart



Hình 4.3-1. Stacked Barplot thể hiện tỷ lệ bệnh nhồi máu cơ tim trong giới tính

Cột mỗi giới tính thể hiện tổng số người. Màu bên trong cho thấy tỷ lệ nam/nữ đã từng bị nhồi máu cơ tim. Để nhận ra giới tính nào có tỷ lệ mắc bệnh cao hơn hoặc thấp hơn

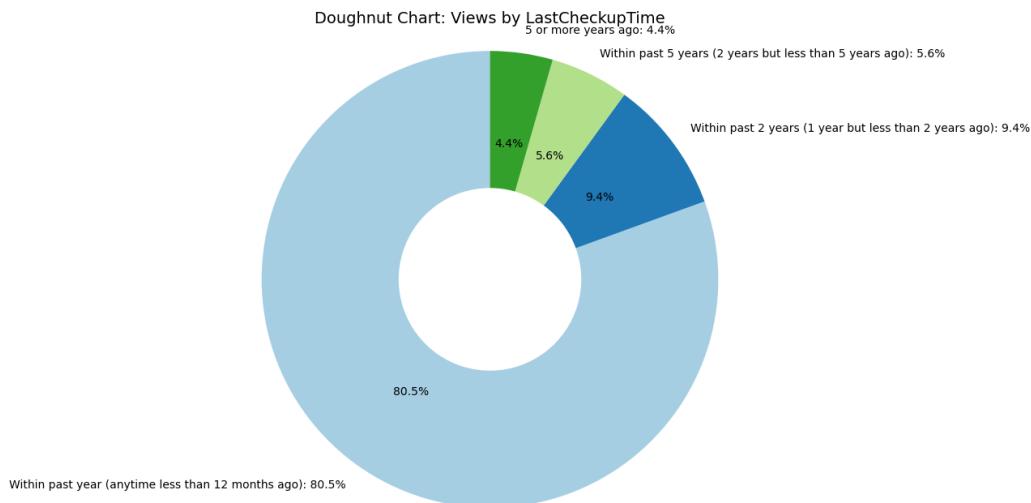
4.3.2. Pie chart



Hình 4.3-2. Pie chart giữa sự phân bố sức khỏe tổng quát

Biểu đồ tròn cho thấy phân bố tỷ lệ sức khỏe tổng quát của người tham gia khảo sát. Đa số người được khảo sát đánh giá sức khỏe ở mức “Good” và “Very Good”, chiếm phần lớn tổng mẫu. Trong khi đó, tỷ lệ người tự đánh giá sức khỏe “Fair” hoặc “Poor” nhỏ hơn đáng kể. Kết quả này phản ánh rằng phần lớn mẫu khảo sát có tình trạng sức khỏe ổn định, tuy nhiên vẫn có một nhóm nhỏ cần quan tâm hơn về sức khỏe cá nhân.

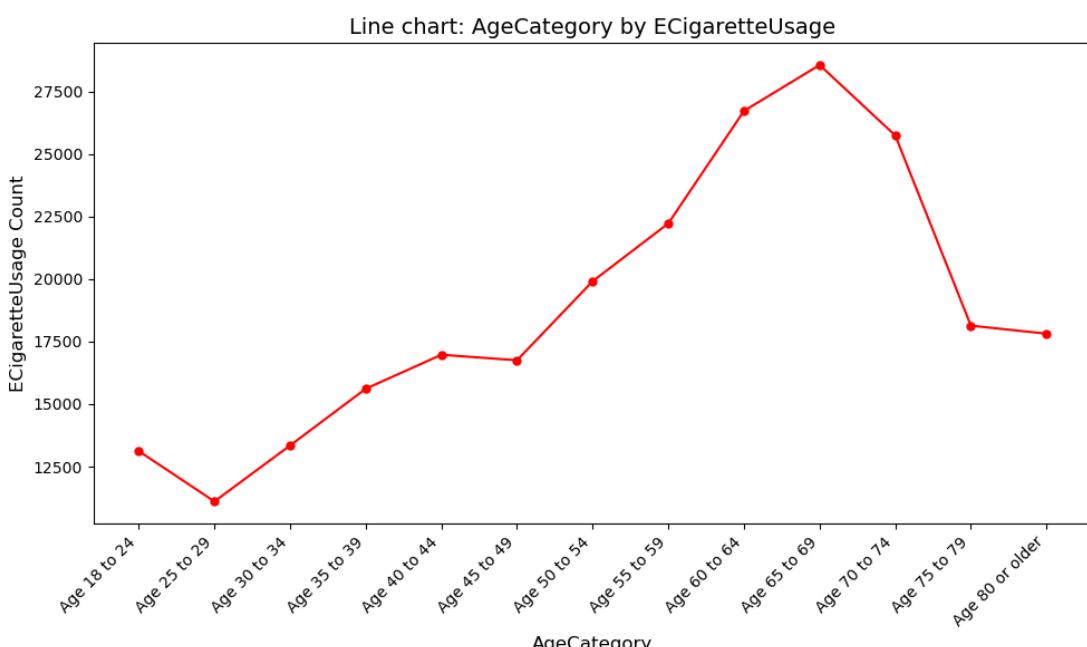
4.3.3. Doughnut chart



Hình 4.3-3. Doughnut chart thể hiện sự phân bố mốc thời gian kiểm tra sức khỏe

Biểu đồ doughnut cho thấy phân bố các mốc thời gian kiểm tra sức khỏe gần nhất của người tham gia khảo sát. Hầu hết người khảo sát đi kiểm tra sức khỏe trong vòng 1 năm gần đây, phản ánh ý thức chăm sóc sức khỏe tốt. Tuy nhiên, vẫn tồn tại một số người kiểm tra lâu, cần nhấn mạnh tầm quan trọng của việc kiểm tra sức khỏe định kỳ.

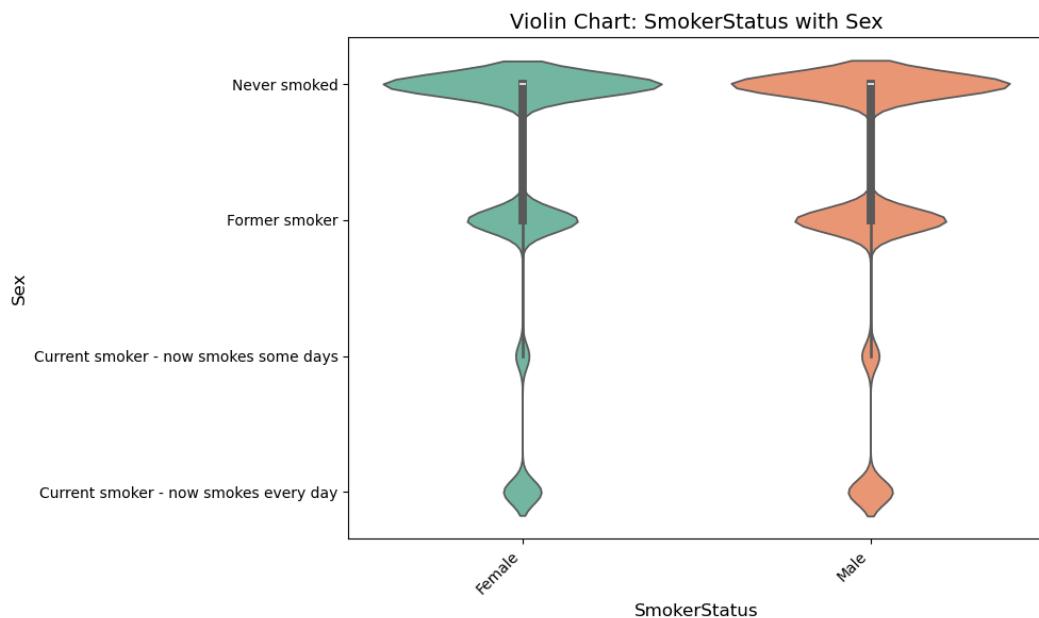
4.3.4. Line chart



Hình 4.3-4. Line chart thể hiện nhóm tuổi có xu hướng sử dụng Ecigarette

Nhóm tuổi trung niên hoặc thanh thiếu niên có thể sử dụng ECigarette nhiều hơn, biểu thị xu hướng hoặc thói quen mới. Nhóm tuổi cao tuổi thường có số lượng sử dụng thấp hơn, phản ánh sự khác biệt về hành vi hoặc nhận thức về sức khỏe.

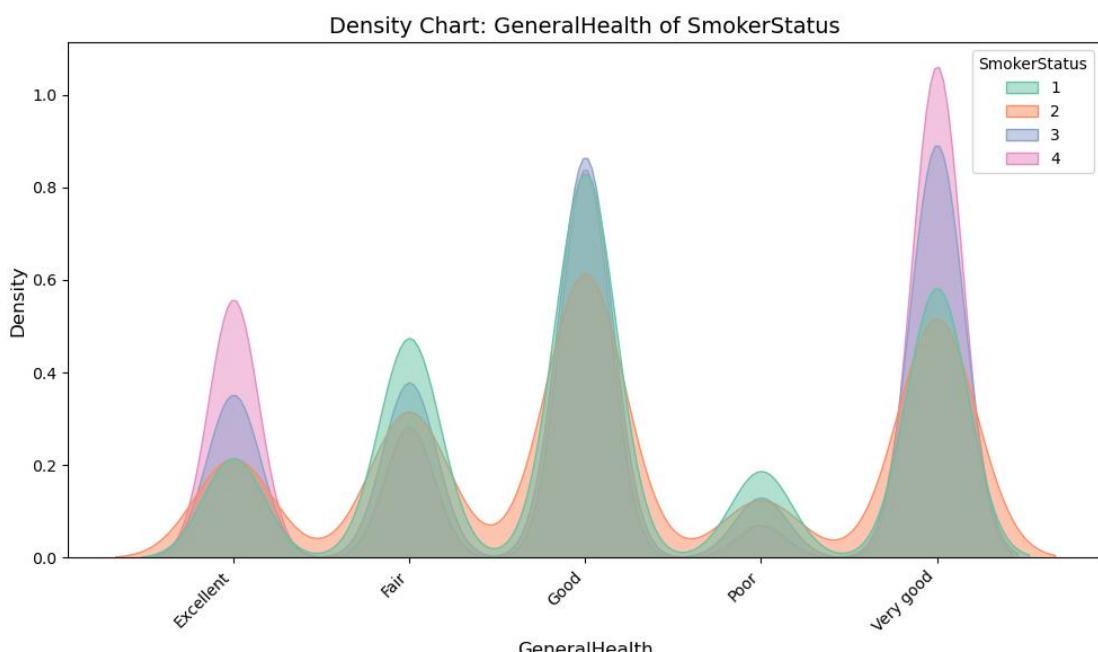
4.3.5. Violin chart



Hình 4.3-5. Violin chart thể hiện tần xuất hút thuốc giữa các giới tính

Có thể quan sát giới tính nam/nữ có tần suất hút thuốc khác nhau. Nếu violin của một giới tính mỏng hoặc hẹp ở các trạng thái nhất định, chứng tỏ ít người thuộc trạng thái đó. Biểu đồ này giúp so sánh trực quan giữa các nhóm giới tính về thói quen hút thuốc.

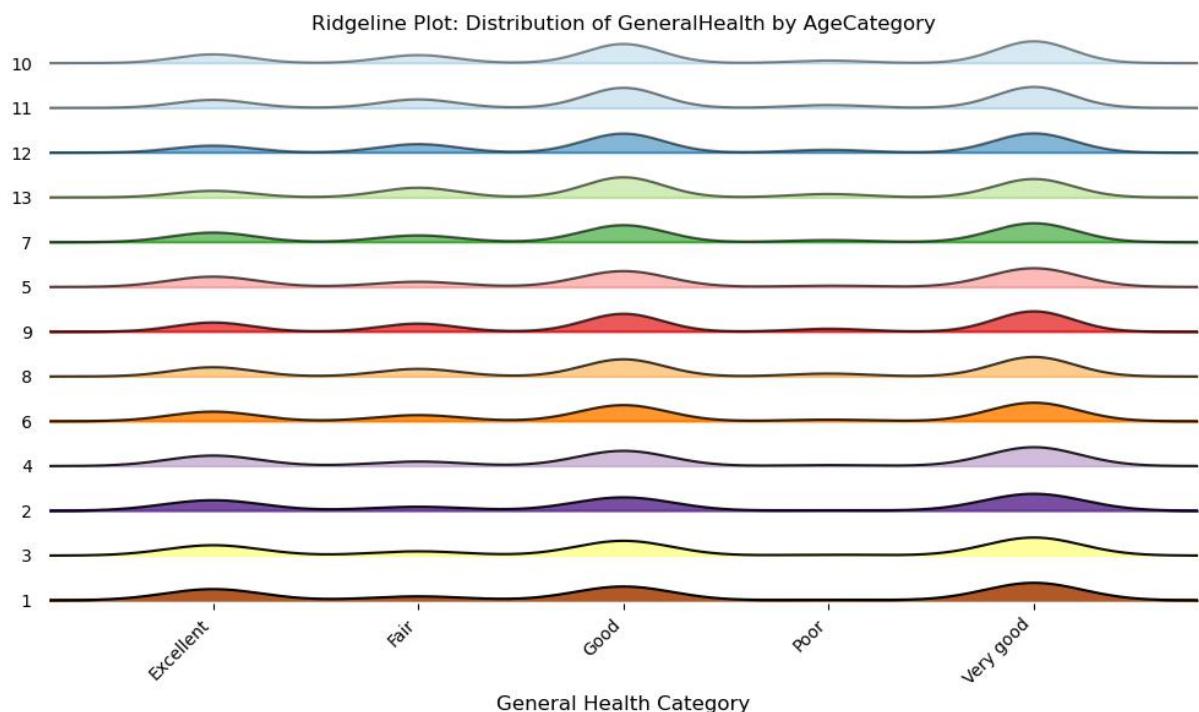
4.3.6. Density chart



Hình 4.3-6. Desnsity chart về sự ảnh hưởng của hút thuốc đến sức khỏe

Đỉnh của đường mật độ cao → nhiều người thuộc nhóm đó. Nếu một màu chiếm ưu thế ở một giá trị GeneralHealth nhất định, nghĩa là nhóm người hút thuốc hoặc không hút thuốc có sức khỏe tập trung ở mức đó. Cho phép nhận xét trực quan về ảnh hưởng hút thuốc đến sức khỏe tổng quát, ví dụ nhóm hút thuốc có xu hướng tập trung ở mức sức khỏe thấp hơn hoặc cao hơn so với nhóm không hút thuốc.

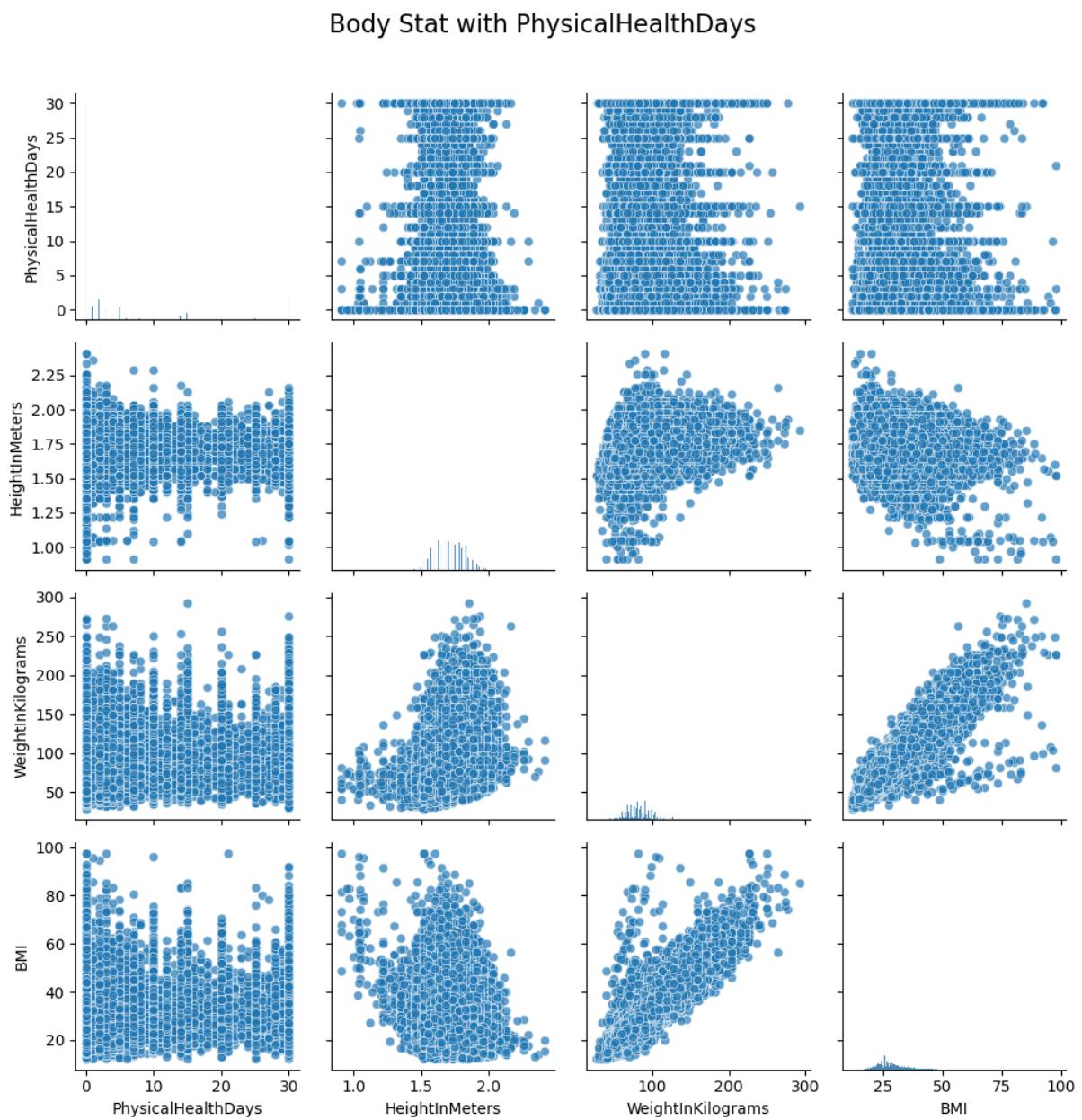
4.3.7. Ridgeline plot



Hình 4.3-7. Ridgeline plot về sự phân bố sức khỏe giữa nhóm tuổi và tình trạng

Đỉnh cao → nhiều người trong nhóm tuổi đó có sức khỏe ở mức đó. Nếu các đường cong chồng lấn mạnh, nghĩa là các nhóm tuổi có phân bố sức khỏe tương tự. Nếu đỉnh của các nhóm tuổi dịch sang trái hoặc phải, nghĩa là mức sức khỏe trung bình thay đổi theo tuổi. Có thể nhận xét trực quan người trẻ tuổi có xu hướng sức khỏe tốt hơn hay kém hơn so với người lớn tuổi dựa vào vị trí các đỉnh.

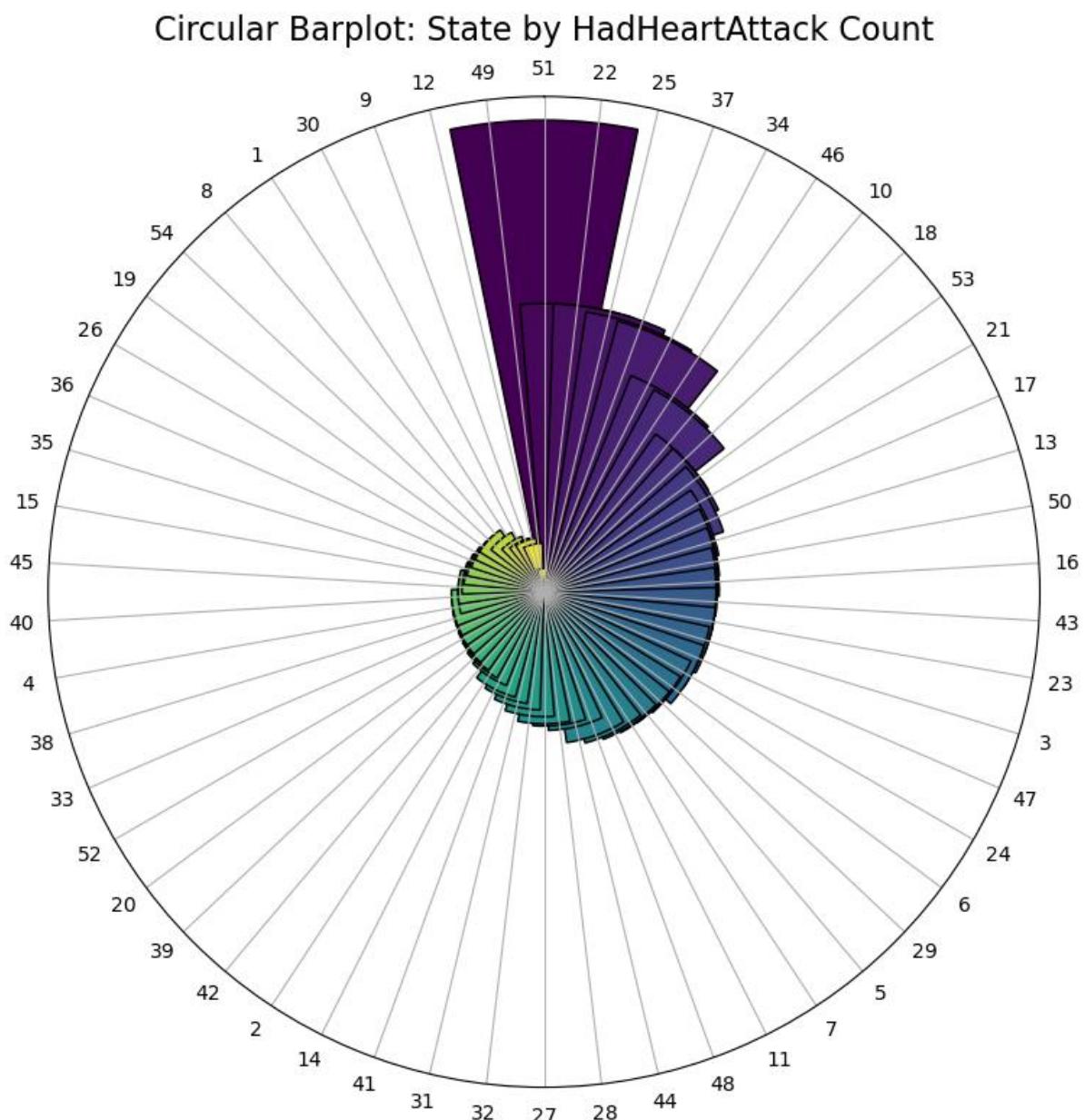
4.3.8. Correlation plot



Hình 4.3-8. Correlation plot về mối quan hệ tương quan giữa cân nặng, chiều cao và BMI

HeightInMeters vs WeightInKilograms: có thể thấy mối quan hệ tương quan dương, điểm càng cao càng nặng. WeightInKilograms với BMI và HeightInMeters với BMI: phản ánh công thức BMI ($BMI = \text{weight} / \text{height}^2$), nên sẽ thấy mối tương quan rõ rệt. PhysicalHealthDays với các chỉ số cơ thể: giúp kiểm tra xem sức khỏe thể chất (số ngày cảm thấy không khỏe) có liên quan đến cân nặng, chiều cao hay BMI không. Histogram trên đường chéo giúp nhận biết dữ liệu có phân bố lệch, tập trung hay trải đều.

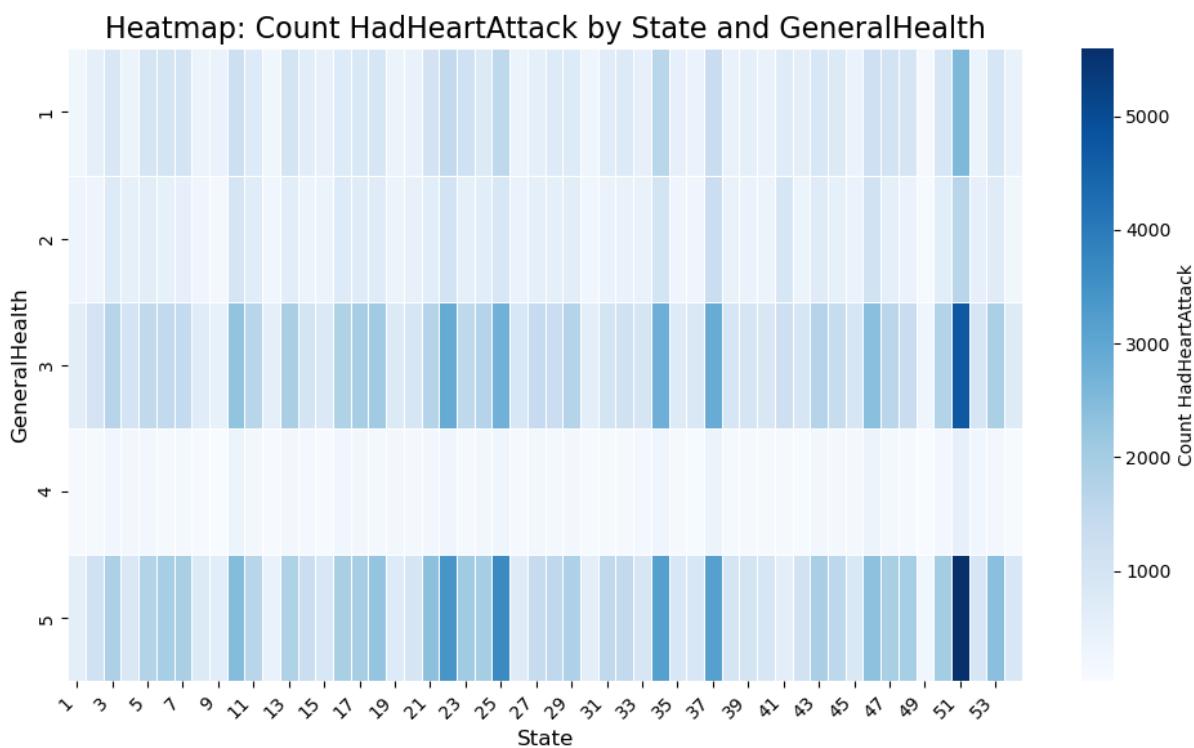
4.3.9. Circular barplot



Hình 4.3-9. Circular barplot về bệnh nhồi máu cơ tim giữa các bang

So sánh số người bị nhồi máu cơ tim giữa các bang. Dễ nhận ra bang nào có số lượng cao nhất và thấp nhất. Thanh dài → bang có nhiều ca nhồi máu cơ tim. Thanh ngắn → bang có ít ca nhồi máu cơ tim. Cho phép nhận biết xu hướng phân bố bệnh theo địa lý.

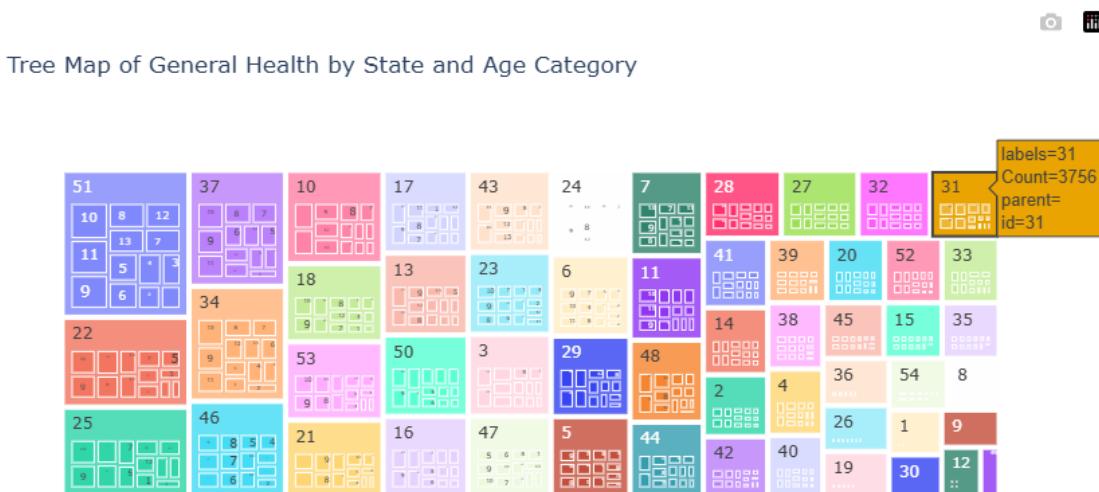
4.3.10. Heatmap



Hình 4.3-10. Heatmap thể hiện số lượng bệnh nhân bị nhồi máu cơ tim theo bang và tình trạng chung

Xem nhanh số lượng nhồi máu cơ tim theo từng bang và tình trạng sức khỏe chung. Dễ dàng nhận ra bang nào có số ca nhồi máu cơ tim cao nhất. Nhóm tình trạng sức khỏe nào dễ bị nhồi máu cơ tim. Có thể phát hiện mối quan hệ giữa sức khỏe chung và nguy cơ nhồi máu: ví dụ, người có sức khỏe “Poor” có khả năng cao bị nhồi máu hơn. Đây là công cụ trực quan hóa tương quan hai biến phân loại với biến định lượng (count).

4.3.11. Tree Map

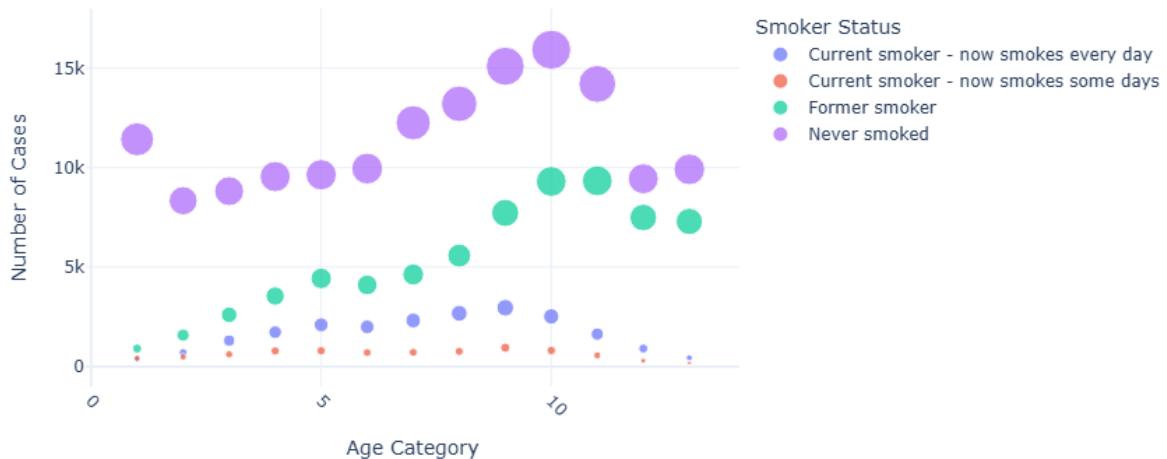


Hình 4.3-11. Treemap về mối tương quan giữa các thông tin người bệnh theo từng bang

Cấu trúc phân cấp: dễ quan sát tỷ lệ người trong từng bang → nhóm tuổi → tình trạng sức khỏe. Kích thước ô: lớn → nhiều người, nhỏ → ít người. Dễ nhận ra Bang nào đông dân hơn (tổng số người). Nhóm tuổi nào chiếm tỷ lệ lớn trong từng bang. Tình trạng sức khỏe nào phổ biến trong từng nhóm tuổi. Nhằm phân tích sức khỏe theo khu vực và tuổi và xác định nhóm cần chú trọng chăm sóc y tế.

4.3.12. Bubble chart

Bubble Chart: COPD Cases by Smoker Status and Age Category



Hình 4.3-12. Bubble chart về mức độ hút thuốc làm tăng nguy cơ COPD

Bong bóng lớn ở nhóm tuổi nào → nhiều ca COPD. Màu sắc cho biết tình trạng hút thuốc → dễ so sánh mức độ nguy cơ giữa nhóm Non-smoker, Former Smoker, Current Smoker. Nhóm tuổi cao và Current Smoker có bong bóng lớn → hút thuốc làm tăng nguy cơ COPD. Non-smoker thường có số ca thấp hơn ở mọi nhóm tuổi.

4.4. Đo lường sự tương đồng và khác biệt của dữ liệu

4.4.1. Ma trận tương quan

“Báo cáo chủ yếu demo trên số lượng ít hàng để có thể debug và sửa lỗi nhanh chóng nhằm tối ưu và nâng cao thuật toán. Khi chương trình đã ổn định sẽ tiến hành triển khai trên toàn bộ tập dữ liệu hơn 400 nghìn dòng.”

Thực hiện demo lấy 10 hàng.

PersonID	PhysicalHealthDays	MentalHealthDays	SleepHours	HeightInMeters	WeightInKilograms	BMI	State_mapped	GeneralHealth_mapped
Person0	0.0	0.0	8.0	1.7026906	90.72 28.529842	1	5	
Person1	0.0	0.0	6.0	1.6	68.04 26.57	1	1	
Person2	2.0	3.0	5.0	1.57	63.5 25.61	1	5	
Person3	0.0	0.0	7.0	1.65	63.5 23.3	1	1	
Person4	2.0	0.0	9.0	1.57	53.98 21.77	1	2	
Person5	1.0	0.0	7.0	1.8	84.82 26.08	1	4	
Person6	0.0	0.0	7.0	1.65	62.6 22.96	1	5	
Person7	0.0	0.0	8.0	1.63	73.48 27.81	1	3	
Person8	0.0	0.0	6.0	1.7	90.72 28.529842	1	3	
Person9	1.0	0.0	7.0	1.68	81.65 29.05	1	3	

SmokerStatus_mapped	ECigaretteUsage_mapped	RaceEthnicityCategory_mapped	AgeCategory_mapped	TetanusLast10Tdap_mapped	PhysicalActivities_mapped						
4	2	5	13	3	0						
4	1	5	13	1	0						
4	1	5	8	NULL	1						
2	1	5	NULL	1	1						
4	1	5	5	1	1						
4	1	5	13	1	0						
3	1	1	13	1	1						
4	1	5	13	3	0						
3	2	5	12	3	1						
4	1	5	11	1	1						

HadHeartAttack_mapped	HadAngina_mapped	HadStroke_mapped	HadAsthma_mapped	HadSkinCancer_mapped	HadCOPD_mapped	HadDepressiveDisorder_mapped					
0	0	0	0	0	0	0					
0	0	0	0	1	0	0					
0	0	0	0	1	0	0					
0	0	0	1	0	0	0					
0	0	0	0	0	0	0					
1	0	1	0	0	0	0					
0	0	0	0	0	0	0					
0	0	0	0	0	0	0					
0	0	0	0	1	0	0					
0	0	0	0	0	0	0					

DifficultyErrands_mapped	ChestScan_mapped	AlcoholDrinkers_mapped	HIVTesting_mapped	FluVaxLast12_mapped	PneumoVaxEver_mapped	HighRiskLastYear_mapped					
0	0	0	0	1	0	0					
0	0	0	0	0	0	0					
0	0	0	0	0	0	0					
0	1	0	0	1	1	0					
0	1	1	0	0	0	1					
0	0	0	0	0	0	1					
0	0	0	1	0	0	0					
0	1	0	0	1	1	0					
0	NULL	0	1	0	0	0					
0	NULL	1	NULL	1	1	1					

Hình 4.4-1. Bộ dữ liệu thực hiện đo lường sự tương đồng

4.4.1.1. Đối với các fields danh nghĩa

Thực hiện tạo hàm: *noun_cal_spark(list, df_spark)*. Đầu vào là danh sách list các cột danh nghĩa và data frame spark của bộ dữ liệu xét.

Kết quả thu được

```
[[0.          0.          0.          0.          0.          0.          0.          0.          0.          0.          0.          ],
 [0.42857  0.          0.          0.          0.          0.          0.          0.          0.          0.          0.          ],
 [0.28571  0.28571  0.          0.          0.          0.          0.          0.          0.          0.          0.          ],
 [0.42857  0.28571  0.28571  0.          0.          0.          0.          0.          0.          0.          0.          ],
 [0.28571  0.14286  0.14286  0.14286  0.          0.          0.          0.          0.          0.          0.          ],
 [0.42857  0.28571  0.28571  0.28571  0.14286  0.          0.          0.          0.          0.          0.          ],
 [0.57143  0.42857  0.42857  0.28571  0.28571  0.42857  0.          0.          0.          0.          0.          ],
 [0.14286  0.28571  0.14286  0.28571  0.14286  0.28571  0.42857  0.          0.          0.          0.          ],
 [0.14286  0.57143  0.42857  0.42857  0.42857  0.57143  0.42857  0.28571  0.          0.          0.          ],
 [0.28571  0.14286  0.14286  0.14286  0.          0.14286  0.28571  0.14286  0.428570.        ]]
```

Hình 4.4-2. Mảng numpy trả về kết quả đo lường của các fields danh nghĩa

4.4.1.2. Đối với các fields thứ tự

Thực hiện tạo hàm: *ordinal_cal_spark(col_name, df_spark, mapping_01)*. Đầu vào là cột thứ tự, data frame spark của bộ dữ liệu xét và dictionary để thực hiện đổi giá trị dựa trên khoảng [0:1]. Thực hiện demo lấy 10 hàng.

Kết quả thu được

```
[ [0.      0.      0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.5     0.      0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.20833333 0.70833333 0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.5     0.      0.5     0.      0.      0.      0.      0.      0.      0.      ]
[0.70833333 0.45833333 0.5    0.125   0.      0.      0.      0.      0.      0.      ]
[0.125   0.375   0.33333333 0.375   0.58333333 0.      0.      0.      0.      0.      ]
[0.      0.5     0.20833333 0.5    0.70833333 0.125   0.      0.      0.      0.      ]
[0.25    0.25    0.45833333 0.25   0.45833333 0.125   0.25   0.      0.      0.      ]
[0.29166667 0.29166667 0.41666667 0.25   0.41666667 0.16666667 0.29166667 0.04166667 0.      0.      ]
[0.33333333 0.33333333 0.375   0.25    0.375   0.20833333 0.33333333 0.08333333 0.04166667 0.      ]]
```

Hình 4.4-3. Mảng numpy trả về kết quả đo lường của các fields thuộc thứ tự

4.4.1.3. Đối với các fields số

Thực hiện tạo hàm: `num_cal_spark(lst_col_name, df_spark)`. Đầu vào là danh sách list các cột kiểu số, data frame spark của bộ dữ liệu xét. Thực hiện demo lấy 10 hàng.

Kết quả thu được

```
[ [0.      0.      0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.30550016 0.      0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.74481246 0.4393123 0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.32305947 0.17335637 0.52752231 0.      0.      0.      0.      0.      0.      0.      ]
[0.62591093 0.48707743 0.46443179 0.38618479 0.      0.      0.      0.      0.      0.      ]
[0.27836486 0.35726597 0.60747571 0.35238938 0.57190751 0.      0.      0.      0.      0.      ]
[0.33492611 0.18522301 0.53938894 0.01186664 0.37431815 0.36425602 0.      0.      0.      0.      ]
[0.14736151 0.15813866 0.59745096 0.20468345 0.47854942 0.3392372 0.21655009 0.      0.      0.      ]
[0.08528301 0.22021715 0.65952945 0.32110979 0.70729458 0.28031453 0.33297643 0.22874516 0.      0.      ]
[0.19449584 0.3014877 0.57413333 0.319047 0.53856512 0.16933139 0.33091364 0.22668237 0.19254617 0.      ]]
```

Hình 4.4-4. Mảng numpy trả về kết quả đo lường của các fields thuộc tính số

4.4.1.4. Đối với các fields nhị phân

Thực hiện tạo hàm: `num_cal_spark(lst_col_name, df_spark)`. Đầu vào là danh sách list các cột kiểu số, data frame spark của bộ dữ liệu xét. Thực hiện demo lấy 10 hàng.

Kết quả thu được

```
[ [0.      0.      0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.75    0.      0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.75    0.375   0.      0.      0.      0.      0.      0.      0.      0.      ]
[0.875   1.      0.9175 0.      0.      0.      0.      0.      0.      0.      ]
[1.      0.75    0.625   0.5425 0.      0.      0.      0.      0.      0.      ]
[0.5     0.75    0.75    0.875  0.875  0.      0.      0.      0.      0.      ]
[0.75    0.5     0.375   0.875  0.375  0.75   0.      0.      0.      0.      ]
[0.625   1.      1.      0.25    0.6675 0.875  1.      0.      0.      0.      ]
[1.      0.875   0.75    0.6675 0.875  1.      0.875  0.75   0.      0.      ]
[0.5425  0.75    0.625   0.4575 0.3325 0.5425 0.4175 0.5825 0.875  0.      ]]
```

Hình 4.4-5. Mảng numpy trả về kết quả đo lường của các fields thuộc tính nhị phân

4.4.1.5. Kết quả của ma trận tương quan

Sau khi đã có những ma trận tương quan con đối với từng kiểu dữ liệu. Ta cộng lại lấy trung bình để tính ma trận tương quan cuối cùng.

```
[ [0.          0.          0.          0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.48610003 0.          0.          0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.43229583 0.49619579 0.          0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.51461189 0.28467127 0.53900446 0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.65851552 0.45574882 0.44288636 0.26073696 0.          0.          0.          0.          0.          0.          ]  
[ 0.28067297 0.42145319 0.45482848 0.44547788 0.54771483 0.          0.          0.          0.          0.          ]  
[ 0.31698522 0.4120446 0.34121112 0.42737333 0.48319696 0.3478512 0.          0.          0.          0.          ]  
[ 0.2794723 0.38162773 0.52782352 0.24093669 0.43754322 0.34284744 0.41831002 0.          0.          0.          ]  
[ 0.35872327 0.4357101 0.52357256 0.37272196 0.55812558 0.42272957 0.43326195 0.2624157 0.          0.          ]  
[ 0.3307325 0.36863087 0.41482667 0.2803094 0.32421302 0.25069961 0.33301606 0.22016981 0.3051759 0.          ] ]
```

Hình 4.4-6. Mảng numpy trả về kết quả tổng kết của tất cả các fields

4.4.2. Độ đo Cosin

Độ tương tự cosine (Cosine similarity) là thước đo độ tương tự có thể được sử dụng để so sánh các tài liệu hoặc đưa ra thứ hạng các tài liệu đối với một vectơ từ (words) truy vấn nhất định. Cho x và y là hai vectơ để so sánh. Sử dụng thước đo cosin làm hàm tương tự, ta có:

$$\text{sim}(x, y) = \frac{x \times y}{\|x\| \times \|y\|}$$

Bộ dữ liệu demo gồm có

	SmokerStatus_mapped	HadHeartAttack_mapped	AlcoholDrinkers_mapped	PhysicalActivities_mapped	HeightInMeters
1	4	0	0	0	1.70269061
1	4	0	0	0	1.61
1	4	0	0	1	1.571
1	2	0	0	1	1.651
1	4	0	1	1	1.571
1	4	1	0	0	1.81
1	3	0	1	1	1.651
1	4	0	0	0	1.631
1	3	0	0	1	1.71
1	4	0	1	1	1.681

Hình 4.4-7. Bộ dữ liệu đầu vào để xét độ đo Cosin

Kết quả thu được

```
[ [0.          0.          0.          0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.99975946 0.          0.          0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.97358086 0.9739536 0.          0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.89475271 0.88873787 0.94535979 0.          0.          0.          0.          0.          0.          0.          ]  
[ 0.94949636 0.94985988 0.97526194 0.92197342 0.          0.          0.          0.          0.          0.          ]  
[ 0.97478039 0.97410944 0.9484768 0.87745007 0.92501332 0.          0.          0.          0.          0.          ]  
[ 0.9196039 0.91735086 0.95393182 0.9444574 0.99000649 0.89825558 0.          0.          0.          0.          ]  
[ 0.99988011 0.99997921 0.97389264 0.89055079 0.94980042 0.97435536 0.91805913 0.          0.          0.          ]  
[ 0.95429013 0.95168312 0.98921054 0.98274803 0.96473938 0.93238066 0.96280094 0.95249718 0.          0.          ]  
[ 0.95075163 0.95062328 0.97543184 0.92833752 0.99974441 0.92668299 0.9920827 0.95070849 0.96783603 0.          ] ]
```

Hình 4.4-8. Mảng numpy trả về kết quả thu được của độ đo Cosin

4.5. Khai thác dữ liệu

4.5.1. Tập dữ liệu thực hiện tính toán

“Báo cáo chủ yếu demo trên số lượng ít hàng để có thể debug và sửa lỗi nhanh chóng nhằm tối ưu và nâng cao thuật toán. Khi chương trình đã ổn định sẽ tiến hành triển khai trên toàn bộ tập dữ liệu hơn 400 nghìn dòng.”

Tập dữ liệu demo gồm 50 hàng và 8 cột được thể hiện như sau:

PhysicalActivities_mapped	HadHeartAttack_mapped	HadAngina_mapped	HadStroke_mapped	HadAsthma_mapped	HadSkinCancer_mapped	HadCOPD_mapped	HadDepressiveDisorder_mapped
0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0
1	0	0	0	0	1	0	0
1	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0

Hình 4.5-1. Bộ dữ liệu chuẩn bị khai thác dữ liệu

4.5.2. Phương pháp Apriori

4.5.2.1. Khái niệm

- Apriori là một thuật toán cơ bản được đề xuất bởi R. Agrawal và R. Srikant vào năm 1994 để khai thác các tập phổ biến cho luật kết hợp Boolean.
- Apriori sử dụng một cách tiếp cận lặp đi lặp lại được gọi là tìm kiếm theo cấp độ, trong đó các tập mục k được sử dụng để khám phá các tập mục (k + 1).

4.5.2.2. Diễn giải khái quát thuật toán

- Với phương pháp sau ta có min_sup = 2. Sử dụng vòng lặp tổ hợp. Tập dữ liệu xét ta sẽ có từng phần tử có 2 giá trị. Đếm và xét min_sup. Nếu không thỏa mãn thì loại bỏ.
- Sau khi đã có tập dữ liệu qua lần lặp 1, ta tiếp tục chạy vòng lặp tổ hợp tiếp theo, với mục đích tạo ra bộ dữ liệu gồm 3 phần tử, mà các phần tử khác nhau và phải thỏa mãn min_sup và loại bỏ các phần tử khác 3 số phần tử bên trong.
- Từ đó ta tiếp tục lặp lại các bước trên cho đến khi nào không còn bộ dữ liệu gồm các phần tử < min_sup.
- Hàm `generate_apriori_spark(D_temp, min_sup)` được cài đặt để thực hiện tìm ra các tập thỏa mãn min_sup. Với đầu vào là data frame spark và giới hạn chọn lọc.

4.5.2.3. Kết quả qua các lần lặp

```
{ ('PhysicalActivities_mapped', 'HadHeartAttack_mapped'): 6,
  ('PhysicalActivities_mapped', 'HadAngina_mapped'): 6,
  ('PhysicalActivities_mapped', 'HadStroke_mapped'): 4,
  ('PhysicalActivities_mapped', 'HadAsthma_mapped'): 2,
  ('PhysicalActivities_mapped', 'HadSkinCancer_mapped'): 10,
  ('PhysicalActivities_mapped', 'HadCOPD_mapped'): 2,
  ('PhysicalActivities_mapped', 'HadDepressiveDisorder_mapped'): 4,
  ('HadHeartAttack_mapped', 'HadAngina_mapped'): 5,
  ('HadHeartAttack_mapped', 'HadStroke_mapped'): 3,
  ('HadHeartAttack_mapped', 'HadDepressiveDisorder_mapped'): 4,
  ('HadAngina_mapped', 'HadStroke_mapped'): 2, ('HadAngina_mapped',
  'HadSkinCancer_mapped'): 2, ('HadAngina_mapped', 'HadCOPD_mapped'):
  3, ('HadAngina_mapped', 'HadDepressiveDisorder_mapped'): 3,
  ('HadStroke_mapped', 'HadDepressiveDisorder_mapped'): 2,
  ('HadAsthma_mapped', 'HadCOPD_mapped'): 2, ('HadSkinCancer_mapped',
  'HadCOPD_mapped'): 2, ('HadSkinCancer_mapped',
  'HadDepressiveDisorder_mapped'): 2, ('HadCOPD_mapped',
  'HadDepressiveDisorder_mapped'): 2}
```

Bảng 4.5-1. Bảng kết quả lặp lần 2 của Apriori

```
{ ('PhysicalActivities_mapped', 'HadHeartAttack_mapped',
  'HadAngina_mapped'): 4, ('PhysicalActivities_mapped',
  'HadHeartAttack_mapped', 'HadStroke_mapped'): 2,
  ('PhysicalActivities_mapped', 'HadHeartAttack_mapped',
  'HadDepressiveDisorder_mapped'): 3, ('PhysicalActivities_mapped',
  'HadAngina_mapped', 'HadStroke_mapped'): 2,
  ('PhysicalActivities_mapped', 'HadAngina_mapped',
  'HadSkinCancer_mapped'): 2, ('PhysicalActivities_mapped',
  'HadAngina_mapped', 'HadDepressiveDisorder_mapped'): 2,
  ('PhysicalActivities_mapped', 'HadStroke_mapped',
  'HadDepressiveDisorder_mapped'): 2, ('HadHeartAttack_mapped',
  'HadAngina_mapped', 'HadDepressiveDisorder_mapped'): 3,
  ('HadHeartAttack_mapped', 'HadStroke_mapped',
  'HadDepressiveDisorder_mapped'): 2}
```

Bảng 4.5-2. Bảng kết quả lặp lần 3 của Apriori

```
{ ('PhysicalActivities_mapped', 'HadHeartAttack_mapped',
  'HadAngina_mapped', 'HadDepressiveDisorder_mapped'): 2,
  ('PhysicalActivities_mapped', 'HadHeartAttack_mapped',
  'HadStroke_mapped', 'HadDepressiveDisorder_mapped'): 2}
```

Bảng 4.5-3. Bảng kết quả lặp lần 4 của Apriori (cuối cùng)

4.5.2.4. Kết quả của thuật toán Apriori

- Kết quả thu được từ bước trước như sau:

```
{ ('PhysicalActivities_mapped', 'HadHeartAttack_mapped',
'HadAngina_mapped', 'HadDepressiveDisorder_mapped') : 2,
('PhysicalActivities_mapped', 'HadHeartAttack_mapped',
'HadStroke_mapped', 'HadDepressiveDisorder_mapped') : 2}
```

Bảng 4.5-4. Kết quả thu được qua các lần lặp của thuật toán Apriori

- Kết quả độ tin cậy của tập luật

```
('PhysicalActivities_mapped',) -> ('HadAsthma_mapped',) = 0.15
('HadAsthma_mapped',) -> ('PhysicalActivities_mapped',) = 0.67
('PhysicalActivities_mapped',) -> ('HadSkinCancer_mapped',) = 0.38
('HadSkinCancer_mapped',) -> ('PhysicalActivities_mapped',) = 0.83
```

Bảng 4.5-5. Kết quả tập luật về độ tin cậy thuật toán Apriori

4.5.3. Phương pháp FP-Growth

4.5.3.1. Khái niệm

- Nén bộ dữ liệu biểu diễn các mục phổ biến thành cây mẫu phổ biến hoặc FP-Tree, cây này giữ lại thông tin liên quan đến tập mục.
- Chia bộ dữ liệu nén thành một tập hợp có điều kiện, mỗi tập dữ liệu được liên kết với một mục phổ biến và khai thác từng bộ dữ liệu riêng biệt. Đối với mỗi “đoạn mẫu”, chỉ cần kiểm tra các tập dữ liệu liên quan của nó. Do đó, cách tiếp cận này có thể làm giảm đáng kể kích thước của các tập dữ liệu cần tìm kiếm, cùng với sự “tang trưởng” của các mẫu đang được kiểm tra.

4.5.3.2. Diễn giải khái quát thuật toán

- Tìm ra các tập mục (itemsets) thường xuyên xuất hiện cùng nhau trong dữ liệu (gọi là frequent itemsets), và từ đó xây dựng luật kết hợp dạng: Nếu A xảy ra \rightarrow có khả năng cao B cũng xảy ra.
- Mỗi dòng dữ liệu (một người/bệnh nhân) được chuyển thành một giao dịch gồm các “mục” có giá trị bằng 1 (tức là có đặc điểm đó).
- Sau đó đếm số lần xuất hiện của từng mục đơn lẻ trong tất cả các giao dịch.
- Chỉ giữ lại những mục có độ hỗ trợ $\geq \text{min_sup}$ (ở đây là 2) \rightarrow Kết quả là danh sách các mục “phổ biến” nhất.
- Chương trình ghép các tập mục nhỏ lại để tạo ứng viên Ck (candidate itemsets) có kích thước lớn hơn.
- Sau đó, đếm số lần chúng cùng xuất hiện trong các giao dịch.
- Nếu ứng viên nào xuất hiện $\geq \text{min_sup}$, nó được thêm vào danh sách tập mục phổ biến (frequent itemsets).
- Vòng lặp này lặp cho đến khi không còn tập mục phổ biến mới được tạo ra.

4.5.3.3. Kết quả của thuật toán FP-Growth

Non-empty subsets	Candidate of Association Rules	confidence
[HadSkinCancer_mapped]	[PhysicalActivities_mapped]	0.8333333333333334
[HadAsthma_mapped]	[PhysicalActivities_mapped]	0.6666666666666666
[PhysicalActivities_mapped]	[HadSkinCancer_mapped]	0.38461538461538464
[PhysicalActivities_mapped]	[HadAsthma_mapped]	0.15384615384615385

Hình 4.5-2. Kết quả tập luật của thuật toán FP-Growth

CHƯƠNG 5: GIỚI THIỆU VỀ HADOOP

5.1. Giới thiệu

Hadoop là một nền tảng mã nguồn mở được phát triển bởi Apache Software Foundation nhằm hỗ trợ lưu trữ và xử lý dữ liệu ở quy mô rất lớn trong môi trường phân tán. Hadoop được thiết kế để hoạt động hiệu quả trên cụm máy tính (cluster) gồm nhiều máy giá rẻ (commodity hardware), giúp các tổ chức lưu trữ và khai thác dữ liệu lớn với chi phí thấp nhưng vẫn đảm bảo hiệu năng, độ tin cậy và khả năng mở rộng.

Hadoop ra đời dựa trên nhu cầu xử lý lượng dữ liệu khổng lồ của các công ty Internet như Google, Yahoo,... Tư tưởng ban đầu được lấy từ hai bài báo nổi tiếng của Google về Google File System (GFS) và MapReduce, sau đó được hiện thực hóa thành hai thành phần cốt lõi của Hadoop.

5.1.1. Thành phần chính của Hadoop

5.1.1.1. Hadoop Distributed File System (HDFS)

HDFS là hệ thống tập phân tán được thiết kế để lưu trữ dữ liệu ở kích thước rất lớn (từ GB đến PB). Dữ liệu được chia nhỏ thành các block và phân tán trên nhiều node trong cluster nhằm:

- Tăng dung lượng lưu trữ
- Tăng độ an toàn nhờ cơ chế replication (nhân bản dữ liệu)
- Giảm thiểu rủi ro mất mát do lỗi phần cứng

HDFS gồm:

- NameNode: quản lý metadata và cấu trúc hệ thống tập
- DataNode: lưu trữ data block thực tế

5.1.1.2. MapReduce

MapReduce là mô hình lập trình phân tán giúp xử lý dữ liệu lớn bằng cách chia nhỏ bài toán thành hai giai đoạn chính:

- Map: xử lý độc lập trên từng phần dữ liệu
- Reduce: tổng hợp và kết luận kết quả

Mô hình này cho phép xử lý dữ liệu song song trên hàng ngàn node mà không cần lập trình phân tán phức tạp.

5.1.2. Các thành phần mở rộng trong hệ sinh thái

Ngoài HDFS và MapReduce, Hadoop được phát triển thành một hệ sinh thái (ecosystem) mạnh mẽ gồm nhiều công cụ hỗ trợ:

- YARN: quản lý tài nguyên và lịch trình tác vụ (Resource Manager)
- Hive: hệ thống kho dữ liệu hỗ trợ truy vấn dữ liệu bằng SQL
- HBase: cơ sở dữ liệu NoSQL chạy trên HDFS
- Pig: ngôn ngữ xử lý dữ liệu quy mô lớn
- Sqoop, Flume: nhập dữ liệu vào Hadoop từ các nguồn khác
- Spark: framework xử lý dữ liệu in-memory nhanh hơn MapReduce

Hệ sinh thái này giúp Hadoop trở thành một nền tảng linh hoạt cho phân tích dữ liệu lớn, học máy và xử lý dữ liệu thời gian thực.

5.1.3. Ưu điểm của Hadoop

- Khả năng mở rộng (Scalability): dễ dàng mở rộng bằng cách thêm node vào cluster
- Chịu lỗi (Fault tolerance): dữ liệu an toàn nhờ cơ chế replication
- Chi phí thấp: sử dụng phần cứng phổ thông
- Xử lý song song hiệu quả: tăng tốc độ xử lý dữ liệu lớn
- Hệ sinh thái phong phú: hỗ trợ nhiều công cụ phân tích dữ liệu

5.1.4. Ứng dụng của Hadoop

- Phân tích dữ liệu doanh nghiệp (BI)
- Machine Learning quy mô lớn
- Xử lý log và dữ liệu IoT
- Quản lý dữ liệu phi cấu trúc từ mạng xã hội
- Phân tích dữ liệu y tế, tài chính, thương mại điện tử, viễn thông

5.2. Cài đặt

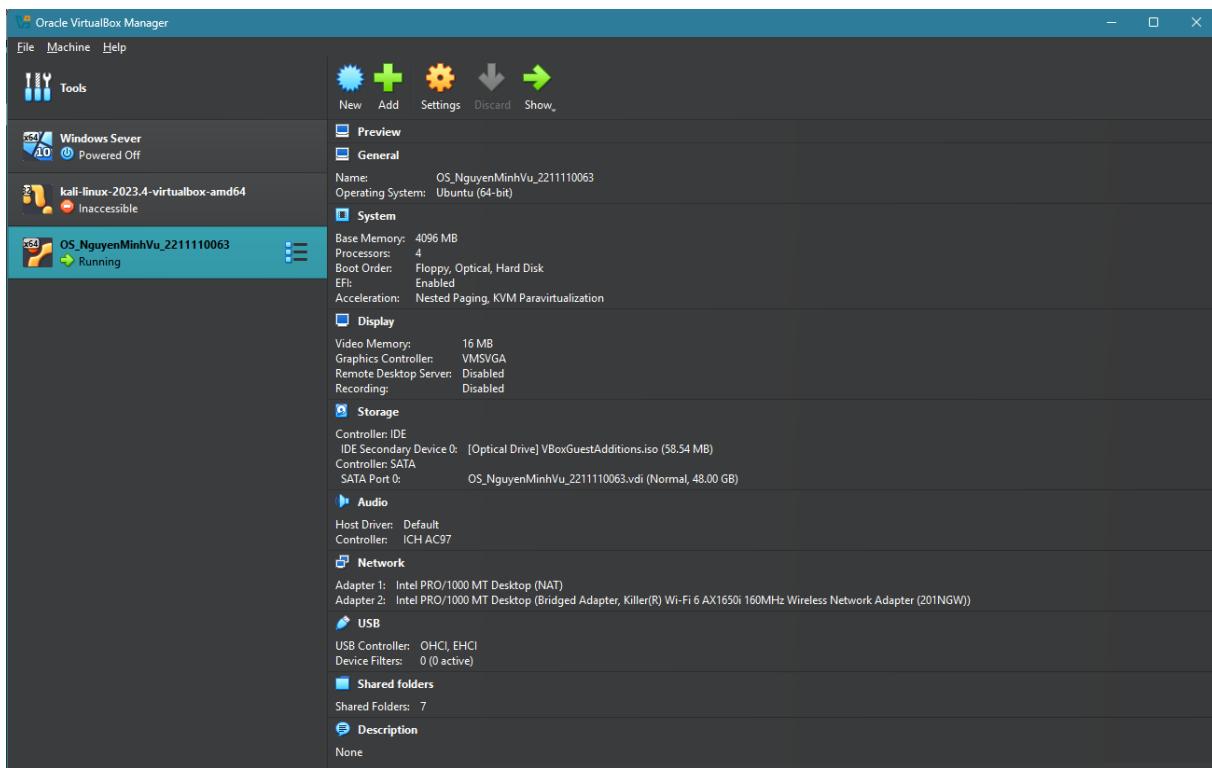
5.2.1. Chuẩn bị môi trường

- VirtualBox: phần mềm tạo máy ảo
- Ubuntu Server / Ubuntu Desktop (khuyến nghị phiên bản 20.04 hoặc 22.04)
- OpenSSH để điều khiển máy ảo từ bên ngoài
- Java JDK 8 hoặc JDK 11 (Hadoop yêu cầu Java cũ, phổ biến nhất là JDK 8)

5.2.2. Tạo máy ảo trên VirtualBox

- Mở VirtualBox → New
- Đặt tên: hadoop-node
- Type: Linux
- Version: Ubuntu (64-bit)
- RAM, CPU, DISK (tùy chọn)

5.2.3. Cấu hình mạng



Hình 5.2-1. Cấu hình mạng máy ảo

5.2.4. Cài đặt ubutu

- Boot ISO Ubuntu
- Cài đặt bình thường
- Tạo user (ví dụ: hadoop)
- Password tùy chọn

```
sudo apt update  
sudo apt upgrade -y
```

5.2.5. Cài đặt OpenSSH

```
sudo apt install openssh-server -y  
sudo systemctl status ssh
```

5.2.6. Cài đặt Java

```
sudo apt install openjdk-8-jdk -y  
java -version
```

5.2.7. Tải và cài đặt Hadoop

- Tải Hadoop

```
cd /opt
sudo wget https://downloads.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz
sudo tar -xvzf hadoop-3.3.6.tar.gz
sudo mv hadoop-3.3.6 hadoop
```

- Cấp quyền

```
sudo chown -R hadoop:hadoop /opt/hadoop
```

5.2.8. Cấu hình hệ thống

```
nano ~/.bashrc
```

```
export HADOOP_HOME=/opt/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
```

5.2.9. Cấu hình các file Hadoop

5.2.9.1. Core-site.xml

```
<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://localhost:9000</value>
  </property>
</configuration>
```

5.2.9.2. hdfs-site.xml

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/home/hadoop/hadoopdata/namenode</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/home/hadoop/hadoopdata/datanode</value>
  </property>
</configuration>
```

5.2.9.3. mapred-site.xml

```
cp $HADOOP_HOME/etc/hadoop/mapred-site.xml.template \
$HADOOP_HOME/etc/hadoop/mapred-site.xml
```

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
</configuration>
```

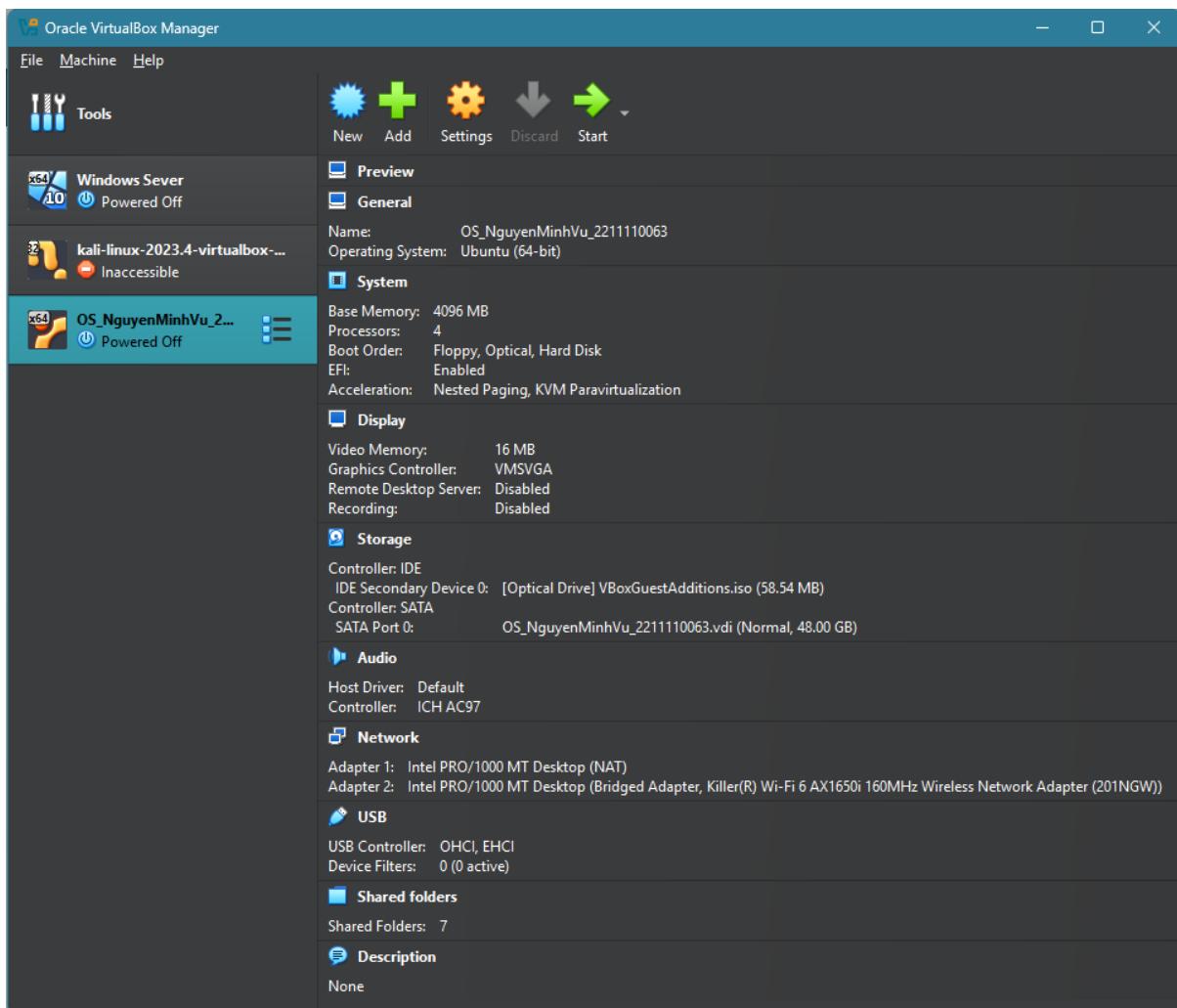
5.2.9.4. yarn-site.xml

```
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
</configuration>
```

5.3. Mô phỏng

5.3.1. Bước 1

- Khởi chạy máy ảo trong Virtual Box



```
OS_NguyenMinhVu_2211110063 [Running] - Oracle VirtualBox
File Machine Input Devices Help
Ubuntu 24.04.3 LTS vunm2211110063 tty1
vunm2211110063 login: biu
Password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-86-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Fri Nov 14 03:08:44 AM UTC 2025

System load: 0.91
Usage of /: 67.3% of 21.95GB
Memory usage: 6%
Swap usage: 0%
Processes: 153
Users logged in: 0
IPv4 address for enp0s3: 10.0.2.15
IPv6 address for enp0s3: fd17:625c:f087:2:a00:27ff:fe9e:25d9

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
Just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

4 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

4 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

biu@vunm2211110063:~$
```

Hình 5.3-1. Khởi chạy máy ảo với VirtualBox

5.3.2. Bước 2

- Khởi chạy Power Shell trên máy thật. Sau đó kết nối đến máy ảo. Nhập mật khẩu để có thể login

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

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\minhv> ssh -p 2222 hdoop@127.0.0.1|
```

```

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

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\minhv\ssh -p 2222 hdoop@127.0.0.1
hdoop@127.0.0.1's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-86-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Fri Nov 14 03:12:11 AM UTC 2025

System load:          0.04
Usage of /:           67.3% of 21.95GB
Memory usage:         6%
Swap usage:          0%
Processes:            158
Users logged in:     1
IPv4 address for enp0s3: 10.0.2.15
IPv6 address for enp0s3: fd17:625c:f037:2:a00:27ff:fe9e:25d9

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.
https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

# updates can be applied immediately.
To see these additional updates run: apt list --upgradable
# additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Mon Nov  3 07:30:02 2025 from 10.0.2.2
--bash: export: `HADOOP_OPTS-Djava.library.path=/home/hdoop/hadoop-3.4.0/lib/native': not a valid identifier
hdoop@vunm2211110063:~$ 

```

Hình 5.3-2. Kết nối máy thật với máy ảo qua PowerShell

5.3.3. Bước 3

- Khởi động server Hadoop bằng cách sau. Sử dụng lần lượt các câu lệnh trỏ file và khởi chạy.

```
cd $HADOOP_HOME/sbin
```

```
hdfs namenode -format
```

```
./start-all.sh
```

- Khởi chạy sẽ có kết quả như sau:

```

Last login: Mon Nov  3 07:30:02 2025 from 10.0.2.2
--bash: export: `HADOOP_OPTS-Djava.library.path=/home/hdoop/hadoop-3.4.0/lib/native': not a valid identifier
hdoop@vunm2211110063:~$ cd $HADOOP_HOME/sbin
hdoop@vunm2211110063:~/hadoop-3.4.0/sbin$ hdfs namenode -format
2025-11-14 03:16:11,795 INFO namenode.NameNode: STARTUP_MSG:
/*****STARTUP_MSG: Starting NameNode*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG: host = vunm2211110063/127.0.1.1
STARTUP_MSG: args = [-format]
STARTUP_MSG: version = 3.4.0

```

Hình 5.3-3. Demo quá trình khởi chạy hệ thống I

- Chọn N khi đã tồn tại dữ liệu trên hdfs:9870

```

2025-11-14 03:16:14,285 INFO snapshot.SnapshotManager: dfs.namenode.snapshot.deletion.ordered = false
2025-11-14 03:16:14,288 INFO snapshot.SnapshotManager: Skiplist is disabled
2025-11-14 03:16:14,293 INFO util.GSet: Computing capacity for map cachedBlocks
2025-11-14 03:16:14,294 INFO util.GSet: VM type      = 64-bit
2025-11-14 03:16:14,294 INFO util.GSet: 0.25% max memory 868 MB = 2.2 MB
2025-11-14 03:16:14,294 INFO util.GSet: capacity     = 2^18 = 262144 entries
2025-11-14 03:16:14,331 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
2025-11-14 03:16:14,331 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2025-11-14 03:16:14,336 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2025-11-14 03:16:14,336 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
2025-11-14 03:16:14,341 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2025-11-14 03:16:14,341 INFO util.GSet: VM type      = 64-bit
2025-11-14 03:16:14,341 INFO util.GSet: 0.02999999932947746% max memory 868 MB = 266.6 KB
2025-11-14 03:16:14,342 INFO util.GSet: capacity     = 2^15 = 32768 entries
Re-format filesystem in Storage Directory root= /home/hadoop/tmpdata/dfs/name; location= null ? (Y or N) N

```

```

Re-format filesystem in Storage Directory root= /home/hadoop/tmpdata/dfs/name; location= null ? (Y or N) N
Format aborted in Storage Directory root= /home/hadoop/tmpdata/dfs/name; location= null
2025-11-14 03:19:27,713 INFO blockmanagement.DatanodeManager: Slow peers collection thread shutdown
2025-11-14 03:19:27,721 INFO namenode.FSNamesystem: Stopping services started for active state
2025-11-14 03:19:27,722 INFO namenode.FSNamesystem: Stopping services started for standby state
2025-11-14 03:19:27,724 INFO util.ExitUtil: Exiting with status 1: ExitException
2025-11-14 03:19:27,728 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****SHUTDOWN_MSG: Shutting down NameNode at vunm2211110063/127.0.1.1*****
*****/
```

Hình 5.3-4. Demo quá trình khởi chạy hệ thống 2

- Khởi chạy server và kiểm thử các cổng đã hoạt động chưa

```

hadoop@vunm2211110063:~/hadoop-3.4.0/sbin$ ./start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [10.0.2.15]
Starting datanodes
Starting secondary namenodes [vunm2211110063]
2025-11-14 03:20:59,981 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using built-in java classes where applicable
WARNING: YARN_CONF_DIR has been replaced by HADOOP_CONF_DIR. Using value of YARN_CONF_DIR.
Starting resourcemanager
WARNING: YARN_CONF_DIR has been replaced by HADOOP_CONF_DIR. Using value of YARN_CONF_DIR.
Starting nodemanagers
WARNING: YARN_CONF_DIR has been replaced by HADOOP_CONF_DIR. Using value of YARN_CONF_DIR.
hadoop@vunm2211110063:~/hadoop-3.4.0/sbin$ jps
2210 SecondaryNameNode
2739 Jps
2403 ResourceManager
2548 NodeManager
2039 DataNode
1913 NameNode
hadoop@vunm2211110063:~/hadoop-3.4.0/sbin$ |

```

Hình 5.3-5. Điều kiện cần của các cổng khi chạy server

- Lưu ý phải có đầy đủ những port sau để có thể chạy được server

```

hadoop@vunm2211110063:~/hadoop-3.4.0/sbin$ jps
2210 SecondaryNameNode
2739 Jps
2403 ResourceManager
2548 NodeManager
2039 DataNode
1913 NameNode

```

5.3.4. Bước 4

- Kiểm thử hệ thống server bằng cách truy cập vào

<http://localhost:8088/cluster>

<http://localhost:9870/dfshealth.html#tab-overview>

- Nếu hệ thống đã chạy ổn định sẽ có giao diện như sau:

The screenshot shows the HDFS Health Overview page at <http://localhost:9870/dfshealth.html#tab-overview>. The top navigation bar includes tabs for Hadoop, Overview, Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, and Utilities. The Overview tab is selected.

Overview '10.0.2.15:9000' (active)

Started:	Fri Nov 14 10:21:02 +0700 2025
Version:	3.4.0.rbd8b77739ef626bb7791783192ee7a5dfaeecc760
Compiled:	Mon Mar 04 13:35:00 +0700 2024 by root from (HEAD detached at release-3.4.0-RC3)
Cluster ID:	CID-a8beccfc9-8cf4-4e35-a9b6-444e48fb6640
Block Pool ID:	BP-1861760268-127.0.1.1-1760771424903

Summary

Security is off.
Safemode is off.

28 files and directories, 8 blocks (8 replicated blocks, 0 erasure coded block groups) = 36 total filesystem object(s).

Heap Memory used 222.68 MB of 305.5 MB Heap Memory. Max Heap Memory is 868 MB.

Non Heap Memory used 53.71 MB of 55.22 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	21.95 GB
Configured Remote Capacity:	0 B
DFS Used:	6.71 MB (0.03%)
Non DFS Used:	14.16 GB
DFS Remaining:	6.65 GB (30.28%)
Block Pool Used:	6.71 MB (0.03%)
DataNodes usages% (Min/Median/Max/stdDev):	0.03% / 0.03% / 0.03% / 0.00%
Live Nodes	1 (Decommissioned: 0, In Maintenance: 0)
Dead Nodes	0 (Decommissioned: 0, In Maintenance: 0)

The screenshot shows the Hadoop Cluster Metrics page at <http://localhost:8088/cluster>. The left sidebar includes links for Cluster, About, Nodes, Node Labels, Applications, Scheduler, and Tools. The Applications section is currently selected.

All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Resources	Reserved Resources
0	0	0	0	0	<memory 0 B, vCores 0>	<memory 21.48 GB, vCores 6>	<memory 0 B, vCores 0>

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes
1	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority	Scheduler Busy %	RH
Capacity Scheduler	[memory_mb (unit=M), vcores]	<memory 1024, vCores 1>	<memory 11000, vCores 3>	0	0	0

Show 20 entries

ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU Vcores	Allocated Memory MB	Allocated GPUs	Reserved CPU Vcores	Reserved Memory MB
No data available in table																	

Showing 0 to 0 of 0 entries

The screenshot shows the Hadoop Cluster Metrics interface. At the top, there's a navigation bar with links like 'About', 'Nodes', 'Node Labels', 'Applications', 'Scheduler', and 'Tools'. The main area is titled 'Nodes of the cluster' and displays a table of active nodes. The table columns include: Node Labels, Rack, Node State, Node Address, Node HTTP Address, Last health-update, Health-report, Containers, Allocation Tags, Mem Used, Mem Avail, Phys Mem Used %, VCores Used, VCores Avail, Phys VCores Used %, and Version. One node is listed with the following details:

/default-rack	RUNNING	vunm221110063.38585	xunm221110063.8042	Fri Nov 14 03:31:05 +0000 2025	0	0 B	21.48 GB	55	0	6	0	3.4.0
---------------	---------	---------------------	--------------------	--------------------------------	---	-----	----------	----	---	---	---	-------

At the bottom of the table, it says 'Showing 1 to 1 of 1 entries'. There are also 'First', 'Previous', 'Next', and 'Last' navigation buttons.

Hình 5.3-6. Giao diện hệ thống sau khi khởi tạo thành công

Khi đã kiểm tra node cluster và server đã chạy ổn định ta bắt đầu tạo ra các chương trình python để thực hiện khai thác dữ liệu.

5.3.5. Bước 5

- Thực hiện đẩy dữ liệu lên hdfs như sau

Tạo file folder “/datalake/mydata” trên hdfs bằng cách

```
hdfs dfs -mkdir /datalake
hdfs dfs -mkdir/datalake/mydata
```

Tạo file python main.py có chức năng đẩy dữ liệu lên hdfs như sau:

```
from pyspark.sql import SparkSession

from pyspark.sql.functions import current_date, year, month,
dayofmonth

spark = SparkSession.builder \
    .appName("Load CSV to HDFS") \
    .getOrCreate()

file_path = 'E:/Khaithacduelieulon/data/heart_2022_with_nans.csv'

df = spark.read.csv(file_path, header=True)

df_with_dates = df.withColumn("created_dateDL", current_date()) \
    .withColumn("updated_dateDL", current_date())
```

```

df_partitioned = df_with_dates.withColumn("year",
year("created_dateDL")) \
    .withColumn("month", month("created_dateDL")) \
    .withColumn("day", dayofmonth("created_dateDL"))

output_path = 'hdfs://localhost:9000/datalake/mydata'

df_partitioned.write.partitionBy("year", "month",
"day").mode("overwrite").parquet(output_path)

spark.stop()

```

- Sau đó đẩy file bằng câu lệnh spark-submit

```

spark-submit --master yarn --deploy-mode cluster --num-executors 6
--executor-cores 2 --executor-memory 4G --conf
"spark.hadoop.fs.defaultFS=hdfs://10.0.2.15:9000" main.py

```

Ví dụ cụ thể luồng hoạt động của hệ thống như sau. Khi tôi đã có dữ liệu trên hdfs. Tôi thực hiện truy xuất dữ liệu như sau:

- Khởi tạo file python myjob.py

```

from pyspark.sql import SparkSession

if __name__ == "__main__":
    spark =
SparkSession.builder.appName("ReadHDFSData").getOrCreate()

    input_path = "/datalake/mydata"

    df = spark.read.parquet(input_path)

    sample_df = df.limit(50)

    sample_df.show(truncate=False)

    spark.stop()

```

Sau đó thực hiện câu lệnh spark-submit trong powershell. Lưu ý trước khi chạy phải đảm bảo rằng trong hệ thống có đã chuyển file myjob.py qua máy ảo trước đó. Và đường dẫn host phải đúng

- Kiểm tra file myjob.py và host như sau:

```
cd
ls
ip a
```

```
hdooop@vnm2211110063:~$ cd
hdooop@vnm2211110063:~$ ls
data dfsdata hadoop-3.4.0 hadoop-3.4.0.tar.gz main.py myjob.py pyspark_env spark-3.5.6-bin-hadoop3 tmpdata
hdooop@vnm2211110063:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host brd :: valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:9e:25:d9 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 metric 100 brd 0.0.2.255 scope global dynamic enp0s3
            valid_lft 84268sec preferred_lft 84268sec
        inet6 fd17:625c:fd37:2:a08:27ff:fe9e:25d9/64 scope global dynamic mngtmpaddr noprefixroute
            valid_lft 86168sec preferred_lft 14168sec
        inet6 fe80::a08:27ff:fe9e:25d9/128 scope link
            valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:a3:24:68 brd ff:ff:ff:ff:ff:ff
hdooop@vnm2211110063:~$ ls
data dfsdata hadoop-3.4.0 hadoop-3.4.0.tar.gz main.py myjob.py pyspark_env spark-3.5.6-bin-hadoop3 tmpdata
hdooop@vnm2211110063:~$ |
```

Hình 5.3-7. Kiểm tra các file cần thiết

- Khi đã hoàn tất, thực hiện submit job lên Hadoop cluster để hệ thống thực hiện chạy chương trình phân tán.

```
spark-submit --master yarn --deploy-mode cluster --num-executors 6
--executor-cores 2 --executor-memory 4G --conf
"spark.hadoop.fs.defaultFS=hdfs://10.0.2.15:9000" myjob.py
```

- Bên cạnh đó kết hợp theo dõi UI hdfs:8088 cluster mode và powershell để theo dõi job.

5.3.5.1. Powershell output

```
hdooop@vnm2211110063:~$ spark-submit --master yarn --deploy-mode cluster --num-executors 6 --executor-cores 2 --executor-memory 4G --conf "spark.hadoop.fs.defaultFS=hdfs://10.0.2.15:9000" myjob.py
25/11/14 03:51:33 WARN Utils: Your hostname, vnm2211110063 resolves to a loopback address: 127.0.1.1; using 10.0.2.15 instead (on interface enp0s3)
25/11/14 03:51:33 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
25/11/14 03:51:34 INFO SecurityManager: Setting spark security library for your platform... using built-in java classes where applicable
25/11/14 03:51:54 INFO Configuration: Connecting to ResourceManager at /10.0.2.15:8882
25/11/14 03:51:56 INFO ResourceUtils: Unable to find 'resource-types.xml' found
25/11/14 03:51:56 INFO Client: Verifying our application has not requested more than the maximum memory capability of the cluster (11000 MB per container)
25/11/14 03:51:56 INFO Client: Will allocate AM container, with 10488 MB memory including 384 MB overhead
25/11/14 03:51:56 INFO Client: Setting up container launch context for our AM
25/11/14 03:51:56 INFO Client: Setting up the launch environment for our AM container
25/11/14 03:51:56 INFO Client: Need to copy yarn jars for spark,yarn.archive is set, falling back to uploading libraries under SPARK_HOME.
25/11/14 03:52:04 INFO Client: Uploading resource file:/tmp/spark-6317bc84-f792-4ab7-b90f-cf25ec759f2a/_spark_libs_7418267026982163986.zip -> hdfs://10.0.2.15:9000/user/hadoop/.sparkStaging/application_1763090535566_0001/_spark_libs_7418267026982163986.zip
25/11/14 03:52:17 INFO Client: Uploading resource file:/home/hadoop/myjob.py -> hdfs://10.0.2.15:9000/user/hadoop/.sparkStaging/application_1763090535566_0001/myjob.py
25/11/14 03:54:17 INFO Client: Uploading resource file:/home/hadoop/spark-3.5.6-bin-hadoop3/python/lib/pyspark.zip -> hdfs://10.0.2.15:9000/user/hadoop/.sparkStaging/application_1763090535566_0001/pyspark.zip
25/11/14 03:54:17 INFO Client: Uploading resource file:/home/hadoop/spark-3.5.6-bin-hadoop3/python/lib/py4j-0.10.9.7-src.zip -> hdfs://10.0.2.15:9000/user/hadoop/.sparkStaging/application_1763090535566_0001/py4j-0.10.9.7-src.zip
25/11/14 03:54:18 INFO Client: Uploading resource file:/tmp/spark-6317bc84-f792-4ab7-b90f-cf25ec759f2a/_spark_conf_6813300621546673435.zip -> hdfs://10.0.2.15:9000/user/hadoop/.sparkStaging/application_1763090535566_0001/_spark_conf_.zip
25/11/14 03:54:19 INFO SecurityManager: Changing view acls to: hdoop
25/11/14 03:54:19 INFO SecurityManager: Changing modify acls to: hdoop
25/11/14 03:54:19 INFO SecurityManager: Changing view acls groups to:
25/11/14 03:54:19 INFO SecurityManager: Changing modify acls groups to:
25/11/14 03:54:19 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view permissions: hdoop; groups with view permissions: EMPTY; users with modify permissions: hdoop
25/11/14 03:54:19 INFO SecurityManager: SecurityManager: ui acls disabled; users with view permissions: EMPTY
25/11/14 03:54:19 INFO Client: Submitting application application_1763090535566_0001 to ResourceManager
25/11/14 03:54:20 INFO ApplicationClientImpl: Submitted application application_1763090535566_0001
25/11/14 03:54:21 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:54:21 INFO Client:
client token: N/A
diagnostics: [Fri Nov 14 03:54:20 +0000 2025] Scheduler has assigned a container for AM, waiting for AM container to be launched
ApplicationMaster host: N/A
ApplicationMaster RPC port: -1
queue: default
start time: 1763092459939
final status: UNDEFINED
tracking URL: http://10.0.2.15:8088/proxy/application_1763090535566_0001/
user: hdoop
```

```

25/11/14 03:54:19 INFO Client: Submitting application application_1763090535566_0001 to ResourceManager
25/11/14 03:54:20 INFO YarnClientImpl: Submitted application application_1763090535566_0001
25/11/14 03:54:21 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:54:21 INFO Client:
    client token: N/A
    diagnostics: [Fri Nov 14 03:54:20 +0000 2025] Scheduler has assigned a container for AM, waiting for AM container to be launched
    ApplicationMaster host: N/A
    ApplicationMaster RPC port: -1
    queue: root.default
    start time: 1763092459939
    final status: UNDEFINED
    tracking URL: http://10.0.2.15:8088/proxy/application_1763090535566_0001/
    user: hdoop
25/11/14 03:54:51 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:55:01 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:55:51 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:56:22 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:56:52 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:57:22 INFO Client: Application report for application_1763090535566_0001 (state: ACCEPTED)
25/11/14 03:57:29 INFO Client: Application report for application_1763090535566_0001 (state: RUNNING)
25/11/14 03:57:29 INFO Client:
    client token: N/A
    diagnostics: N/A
    ApplicationMaster host: 10.0.2.15
    ApplicationMaster RPC port: 46561
    queue: root.default
    start time: 1763092459939
    final status: UNDEFINED
    tracking URL: http://10.0.2.15:8088/proxy/application_1763090535566_0001/
    user: hdoop
25/11/14 03:57:59 INFO Client: Application report for application_1763090535566_0001 (state: RUNNING)
25/11/14 03:58:32 INFO Client: Application report for application_1763090535566_0001 (state: RUNNING)
25/11/14 03:59:03 INFO Client: Application report for application_1763090535566_0001 (state: RUNNING)
25/11/14 03:59:36 INFO Client: Application report for application_1763090535566_0001 (state: RUNNING)
25/11/14 03:59:57 INFO Client: Application report for application_1763090535566_0001 (state: FINISHED)
25/11/14 03:59:57 INFO Client:
    client token: N/A
    diagnostics: N/A
    ApplicationMaster host: 10.0.2.15
    ApplicationMaster RPC port: 46561
    queue: root.default
    start time: 1763092459939
    final status: SUCEEDED
    tracking URL: http://10.0.2.15:8088/proxy/application_1763090535566_0001/
    user: hdoop
25/11/14 04:00:00 INFO ShutdownHookManager: Shutdown hook called
25/11/14 04:00:09 INFO ShutdownHookManager: Deleting directory /tmp/spark-6317bc84-f792-4ab7-b99f-cf25ec759f2a
25/11/14 04:00:10 INFO ShutdownHookManager: Deleting directory /tmp/spark-6de53983-6c30-42bd-991d-6719627504e5

```

Hình 5.3-8. Theo dõi quá trình hệ thống hoạt động với PowerShell

5.3.5.2. Giao diện UI Cluster Node

All Applications												
Cluster Metrics												
About Nodes	Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Resources					
Nodes Labels	0	1	0	1	<memory 2 GB, vCores 1>	<memory 21.48 GB, vCores 6>						
Applications												
NEW												
NEW_SAVING												
SUBMITTED												
ACCEPTED												
RUNNING												
FINISHED												
FAILED												
KILLED												
Scheduler												
Tools												

All Applications												
Cluster Metrics												
About Nodes	Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Resources					
Nodes Labels	1	0	1	0	1	<memory 2 GB, vCores 1>	<memory 21.48 GB, vCores 6>					
Applications												
NEW												
NEW_SAVING												
SUBMITTED												
ACCEPTED												
RUNNING												
FINISHED												
FAILED												
KILLED												
Scheduler												
Tools												

All Applications												
Cluster Metrics												
About Nodes	Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Resources					
Nodes Labels	1	0	1	0	4	<memory 17 GB, vCores 4>	<memory 21.48 GB, vCores 6>					
Applications												
NEW												
NEW_SAVING												
SUBMITTED												
ACCEPTED												
RUNNING												
FINISHED												
FAILED												
KILLED												
Scheduler												
Tools												

Hình 5.3-9. Theo dõi quá trình hệ thống hoạt động với hdfs 8088

Sau đó khi hệ thống đã chạy thành công. Dữ liệu sẽ không in ra ngay trong màn hình powershell hay hdfs 8088. Vì vậy chúng ta cần phải sử dụng câu lệnh đọc log để xuất được dữ liệu output của job trước đó.

```
yarn logs -applicationId application_1763090535566_0001
```

Khai thác dữ liệu lớn

Indiana Female Excellent 0.0	0.0	Within past year (anytime less than 12 months ago)	No	5.0	None of them	No	No	No
No No Never smoked No No	No No Use them every day	No Yes	Black only, Non-Hispanic Age 30 to 34 1.52	83.91	No	No	No	
No No Yes, received Tdap 2.0 0.0	Within past year (anytime less than 12 months ago)	No Yes	2025-10-19 2025-10-19 2025 10 19					
Alabama Female Fair 0.0	0.0	Never used e-cigarettes in my entire life No	No No	White only, Non-Hispanic Age 40 to 44 1.57	53.98	No	No	No
No No Never smoked No No	Never used any tetanus shot in the past 10 years No	White only, Non-Hispanic Age 40 to 44 1.57	53.98	21.77 Yes	No			
No No Yes No, did not receive any tetanus shot in the past 10 years No	Within past 5 years (2 years but less than 5 years ago) No	White only, Non-Hispanic Age 40 to 44 1.57	53.98	21.77 Yes	No			
Indiana Female Good 0.0	20.0	Never used e-cigarettes in my entire life Yes	No Yes	Hispanic Age 40 to 44 1.8	86.18	No	No	No
No No Never smoked No No	Never used e-cigarettes in my entire life No	Hispanic Age 40 to 44 1.8	86.18	26.5 No	No			
No No Yes No, did not receive any tetanus shot in the past 10 years NULL	Within past year (anytime less than 12 months ago)	No No	7.0 NULL					
Alabama Male Poor 1.0 0.0	Never used e-cigarettes in my entire life No	White only, Non-Hispanic Age 80 or older 1.8 84.82	26.08 No	No				
No No Never smoked No No	Never used any tetanus shot in the past 10 years No	White only, Non-Hispanic Age 80 or older 1.8 84.82	26.08 No	No				
Indiana Female Very good 0.0	10.0	Within past year (anytime less than 12 months ago)	Yes No	7.0 1 to 5				
Yes No Yes No No	Not at all (right now)	Yes No	7.0 1 to 5					
No Yes Former smoker No No	Not at all (right now)	Yes No	7.0 1 to 5					
Alabama Female Very good 0.0	0.0	Yes, received tetanus shot but not sure what type No	No Yes	White only, Non-Hispanic Age 80 or older 1.57 67.13	27.07 No	No		
No No No No No	Within past year (anytime less than 12 months ago)	No No	7.0 NULL					
No No Never smoked No No	Within past year (anytime less than 12 months ago)	No No	7.0 NULL					
No No Yes No No	Never used e-cigarettes in my entire life No	Black only, Non-Hispanic Age 80 or older 1.65 62.6	22.96 Yes	No				
No No No No No	Never used any tetanus shot in the past 10 years No	Black only, Non-Hispanic Age 80 or older 1.65 62.6	22.96 Yes	No				
Indiana Male Poor 30.0 0.0	Within past year (anytime less than 12 months ago)	No No	5.0 None of them					
No No No No No	Never used e-cigarettes in my entire life Yes	White only, Non-Hispanic Age 55 to 59 1.78 90.72	28.7 Yes	No				
Yes No Yes No No	Received tetanus shot but not sure what type No	White only, Non-Hispanic Age 55 to 59 1.78 90.72	28.7 Yes	No				
Alabama Female Good 0.0	0.0	Within past year (anytime less than 12 months ago)	No No	8.0 NULL				
No No No No No	Within past year (anytime less than 12 months ago)	No No	8.0 NULL					
No No Never smoked No No	Never used e-cigarettes in my entire life Yes	White only, Non-Hispanic Age 80 or older 1.63 73.48	27.81 No	No				
No No Yes No No	Yes, received tetanus shot but not sure what type No	White only, Non-Hispanic Age 80 or older 1.63 73.48	27.81 No	No				
Indiana Male Excellent 0.0	0.0	Within past year (anytime less than 12 months ago)	Yes No	4.0 6 or more, but not all No	Yes No			
No No No No No	Yes No Yes No No	4.0 6 or more, but not all No	Yes No					
No No Yes No No	Current smoker now smokes every day Use them some days	Yes No	White only, Non-Hispanic Age 65 to 69 1.8 85.28	26.22 Yes	No			
No No No NULL	Within past year (anytime less than 12 months ago)	Yes No	White only, Non-Hispanic Age 65 to 69 1.8 85.28	26.22 Yes	No			
Alabama Female Good 0.0	0.0	Within past year (anytime less than 12 months ago)	Yes No	6.0 NULL				
No Yes No No No	Within past year (anytime less than 12 months ago)	Yes No	6.0 NULL					
No No Former smoker No No	Not at all (right now)	NULL	White only, Non-Hispanic Age 75 to 79 1.7 NULL					
No No Yes No No	Yes, received tetanus shot but not sure what type No	No No	White only, Non-Hispanic Age 75 to 79 1.7 NULL					
Indiana Female Very good 0.0	NULL	Within past year (anytime less than 12 months ago)	Yes No	8.0 None of them				
No No No No No	Never used e-cigarettes in my entire life Yes	White only, Non-Hispanic Age 65 to 69 1.63 73.48	27.81 No	No				
No No Never smoked No No	Never used e-cigarettes in my entire life Yes	White only, Non-Hispanic Age 65 to 69 1.63 73.48	27.81 No	No				
Yes No Yes No No	Yes, received Tdap 0.0	Within past year (anytime less than 12 months ago)	Yes No	7.0 NULL				
No No No No No	Yes No Yes No No	7.0 NULL						
No No Never smoked No No	Never used e-cigarettes in my entire life NULL	White only, Non-Hispanic Age 70 to 74 1.68 81.65	29.05 Yes	NULL				
Yes Yes No No No	No, did not receive any tetanus shot in the past 10 years No	White only, Non-Hispanic Age 70 to 74 1.68 81.65	29.05 Yes	NULL				
Indiana Male Fair 30.0 0.0	Within past year (anytime less than 12 months ago)	Yes No	7.0 NULL					
Yes No Yes No No	Use them every day	Yes No	7.0 NULL					
Yes Yes No No No	Yes, received tetanus shot but not sure what type	No Yes	Age 35 to 39 1.65 74.84	27.46 NULL	No			
No No Never smoked No No	Yes, received tetanus shot but not sure what type	No No	Age 35 to 39 1.65 74.84	27.46 NULL	No			

Hình 5.3-10. Output chương trình chạy bằng hadoop cluster

CHƯƠNG 6: ĐÁNH GIÁ MẪU THU ĐƯỢC

6.1. Đánh giá mẫu bằng thang độ tương quan Lift

6.1.1. Khái niệm

Trong khai thác dữ liệu, đặc biệt là khi áp dụng luật kết hợp (Association Rules Mining) như thuật toán Apriori hoặc FP-Growth, việc đánh giá mức độ tin cậy và giá trị thực tế của các luật tìm được là rất quan trọng.

Bên cạnh các chỉ số phổ biến như Support (độ hỗ trợ) và Confidence (độ tin cậy), Lift là một thước đo tương quan giúp xác định mức độ ảnh hưởng thực sự giữa hai biến (hoặc hai sự kiện) trong dữ liệu.

6.1.2. Triển khai

Cài đặt hàm và lấy ra những phần tử trong tập kết quả của thuật toán Apriori và FP-Growth. Sau đó cài đặt hàm *lift_cal(Data, list_col_rules)* để thực hiện tính thang đo. Với Data là tập dữ liệu đã sau một vài bước làm sạch và danh sách các phần tử.

6.1.3. Kết quả thu được

Kết quả thu thập phần tử từ thuật toán Apriori vào danh sách

```
[PhysicalActivities_mapped', 'HadHeartAttack_mapped', 'HadAngina_mapped',
'HadDepressiveDisorder_mapped', 'HadStroke_mapped']
```

Kết quả thu thập phần tử từ thuật toán FP-Growth vào danh sách

```
[HadSkinCancer_mapped', 'PhysicalActivities_mapped', 'HadAsthma_mapped',
'HadHeartAttack_mapped', 'HadStroke_mapped', 'HadAngina_mapped']
```

Kết quả của phương pháp đánh giá Lift

```
Lift -> Apriori
('PhysicalActivities_mapped -> HadHeartAttack_mapped': 3.2406937947645708e-06, 'PhysicalActivities_mapped -> HadAngina_mapped': 3.3228146296257983e-06,
'PhysicalActivities_mapped -> HadDepressiveDisorder_mapped': 3.644969497159268e-06, 'PhysicalActivities_mapped -> HadStroke_mapped': 3.154549868864931e-06,
'HadHeartAttack_mapped -> HadAngina_mapped': 3.283416739742337e-05, 'HadHeartAttack_mapped -> HadDepressiveDisorder_mapped': 4.8122480792001165e-06,
'HadHeartAttack_mapped -> HadStroke_mapped': 1.802500508326495e-05, 'HadAngina_mapped -> HadDepressiveDisorder_mapped': 4.9518814603594885e-06,
'HadAngina_mapped -> HadStroke_mapped': 1.52806191963548e-05, 'HadDepressiveDisorder_mapped -> HadStroke_mapped': 5.6060531960232236e-06)
Lift -> Fp-Growth
('HadSkinCancer_mapped -> PhysicalActivities_mapped': 4.051856544679438e-06, 'HadSkinCancer_mapped -> HadAsthma_mapped': 3.947151816809064e-06,
'HadSkinCancer_mapped -> HadHeartAttack_mapped': 6.724831522114881e-06, 'HadSkinCancer_mapped -> HadStroke_mapped': 6.602950671609099e-06,
'HadSkinCancer_mapped -> HadAngina_mapped': 7.979091638774234e-06, 'PhysicalActivities_mapped -> HadAsthma_mapped': 3.7677494131254205e-06,
'PhysicalActivities_mapped -> HadHeartAttack_mapped': 3.2406937947645708e-06, 'PhysicalActivities_mapped -> HadStroke_mapped': 3.154549868864931e-06,
'PhysicalActivities_mapped -> HadAngina_mapped': 3.3228146296257983e-06, 'HadAsthma_mapped -> HadHeartAttack_mapped': 5.037919047991019e-06,
'HadAsthma_mapped -> HadStroke_mapped': 5.746080969764132e-06, 'HadAsthma_mapped -> HadAngina_mapped': 5.376389700166156e-06,
'HadHeartAttack_mapped -> HadStroke_mapped': 1.802500508326495e-05, 'HadHeartAttack_mapped -> HadAngina_mapped': 3.283416739742337e-05,
'HadStroke_mapped -> HadAngina_mapped': 1.52806191963548e-05)
```

Hình 6.1-1. Kết quả phương pháp đánh giá Lift của Apriori và FP-Growth

6.2. Đánh giá mẫu bằng thang đo χ^2

6.2.1. Khái niệm

Thang đo χ^2 (Chi-square) là một phương pháp kiểm định thống kê được sử dụng rộng rãi trong khai thác dữ liệu và phân tích mối tương quan giữa các biến rời rạc.

Mục tiêu của phép kiểm định này là đánh giá xem hai biến có độc lập với nhau hay không - tức là mối quan hệ giữa chúng có ý nghĩa thống kê hay chỉ là ngẫu nhiên.

Trong khai thác dữ liệu lớn, đặc biệt khi xử lý bằng Hadoop hoặc Spark, phép kiểm định χ^2 giúp xác định mức độ liên quan giữa các thuộc tính (features), hỗ trợ chọn lọc các đặc trưng có ảnh hưởng thực sự trong mô hình học máy hoặc trong quá trình phân tích dữ liệu.

6.2.2. Triển khai

Cài đặt hàm và lấy ra những phần tử trong tập kết quả của thuật toán Apriori và FP-Growth. Sau đó cài đặt hàm *chi_square(Data, list_col_rules)* để thực hiện tính thang đo. Với Data là tập dữ liệu đã sau một vài bước làm sạch và danh sách các phần tử.

6.2.3. Kết quả thu được

Kết quả thu thập phần tử từ thuật toán Apriori vào danh sách

```
[PhysicalActivities_mapped', 'HadHeartAttack_mapped', 'HadAngina_mapped',
'HadDepressiveDisorder_mapped', 'HadStroke_mapped']
```

Kết quả thu thập phần tử từ thuật toán FP-Growth vào danh sách

```
[HadSkinCancer_mapped', 'PhysicalActivities_mapped', 'HadAsthma_mapped',
'HadHeartAttack_mapped', 'HadStroke_mapped', 'HadAngina_mapped']
```

Kết quả của phương pháp đánh giá Chi Square

```
Chi-Square -> Apriori
{'PhysicalActivities_mapped -> HadHeartAttack_mapped': 336103.6772884498, 'PhysicalActivities_mapped -> HadAngina_mapped': 329202.19270987273,
'PhysicalActivities_mapped -> HadDepressiveDisorder_mapped': 194891.5969261002, 'PhysicalActivities_mapped -> HadStroke_mapped': 350801.8420552311,
'HadHeartAttack_mapped -> HadAngina_mapped': 583919.3235662989, 'HadHeartAttack_mapped -> HadDepressiveDisorder_mapped': 356836.1734900106,
'HadHeartAttack_mapped -> HadStroke_mapped': 587998.4512433796, 'HadAngina_mapped -> HadDepressiveDisorder_mapped': 350545.6449997204,
'HadAngina_mapped -> HadStroke_mapped': 576795.4445731449, 'HadDepressiveDisorder_mapped -> HadStroke_mapped': 374678.11241142964}
Chi-Square -> Fp-Growth
{'HadSkinCancer_mapped -> PhysicalActivities_mapped': 294793.5684717573, 'HadSkinCancer_mapped -> HadAsthma_mapped': 381865.2032464472,
'HadSkinCancer_mapped -> HadHeartAttack_mapped': 512205.5491400589, 'HadSkinCancer_mapped -> HadStroke_mapped': 531662.2085906233,
'HadSkinCancer_mapped -> HadAngina_mapped': 506329.18100700574, 'PhysicalActivities_mapped -> HadAsthma_mapped': 237491.70205219561,
'PhysicalActivities_mapped -> HadHeartAttack_mapped': 336103.6772884498, 'PhysicalActivities_mapped -> HadStroke_mapped': 350801.8420552311,
'PhysicalActivities_mapped -> HadAngina_mapped': 329202.19270987273, 'HadAsthma_mapped -> HadHeartAttack_mapped': 422388.8618262863,
'HadAsthma_mapped -> HadStroke_mapped': 441520.0387432799, 'HadAsthma_mapped -> HadAngina_mapped': 415870.1913678375,
'HadHeartAttack_mapped -> HadStroke_mapped': 587998.4512433796, 'HadHeartAttack_mapped -> HadAngina_mapped': 583919.3235662989,
'HadStroke_mapped -> HadAngina_mapped': 576795.4445731449}
```

Hình 6.2-1. Kết quả phương pháp đánh giá Chi Square của Apriori và FP-Growth

CHƯƠNG 7: HƯỚNG PHÁT TRIỂN

7.1. Mở rộng quy mô dữ liệu và hiệu năng hệ thống

Trong phạm vi đề tài, hệ thống chỉ được thử nghiệm trên một tập dữ liệu có quy mô giới hạn và được xử lý trong môi trường mô phỏng Hadoop – Spark. Trong tương lai, đề tài có thể được phát triển theo hướng:

- Tích hợp trên cụm Hadoop thực tế (multi-node cluster) để kiểm tra khả năng mở rộng (scalability) khi kích thước dữ liệu tăng lên hàng terabyte.
- Tối ưu hiệu năng xử lý bằng cách:
 - Sử dụng Spark Structured Streaming cho dữ liệu thời gian thực.
 - Cấu hình tối ưu YARN ResourceManager và Spark Executor để tận dụng tốt hơn tài nguyên phần cứng.
 - Áp dụng caching, partitioning và broadcast join để tăng tốc quá trình tính toán.

7.2. Ứng dụng các thuật toán học máy nâng cao

Hiện tại, đề tài chủ yếu tập trung vào việc khai thác, phân tích và đánh giá tương quan dữ liệu (bằng Lift, χ^2 ,...). Trong tương lai, hướng phát triển có thể là:

- Xây dựng mô hình dự đoán (predictive modeling) bằng các thuật toán trong Spark MLlib như: Logistic Regression, Decision Tree, Random Forest, Gradient Boosted Tree, KMeans,...
- Tự động lựa chọn đặc trưng (Feature Selection) dựa trên kết quả từ các thang đo thống kê (χ^2 , Information Gain,...).
- Kết hợp Deep Learning bằng cách tích hợp TensorFlowOnSpark hoặc PyTorch Lightning để xử lý dữ liệu phi cấu trúc (ảnh, âm thanh, văn bản).

7.3. Mở rộng hệ thống phân tích bằng PySpark và các công cụ BI

- Phát triển pipeline phân tích tự động trong PySpark, giúp người dùng dễ dàng xử lý dữ liệu mà không cần can thiệp sâu vào lập trình.
- Kết hợp với các công cụ trực quan hóa dữ liệu như Tableau, Power BI hoặc Apache Superset, cho phép người dùng khai thác dữ liệu lớn trực quan và dễ hiểu hơn.
- Tích hợp Apache Airflow để tự động hóa luồng công việc (workflow automation) từ giai đoạn nhập dữ liệu, xử lý, đến xuất kết quả.

7.4. Nâng cao khả năng tương tác và triển khai thực tế

- Kết hợp Flask / FastAPI + React / Streamlit để cho phép người dùng tải dữ liệu, chạy phân tích Hadoop/Spark, và xem kết quả trực tiếp qua trình duyệt.
- Hỗ trợ triển khai mô hình trên nền tảng đám mây (AWS EMR, Google Dataproc, Azure HDInsight) để phục vụ doanh nghiệp hoặc tổ chức cần xử lý dữ liệu lớn.
- Xây dựng API RESTful cho phép các hệ thống khác tích hợp truy vấn phân tích dữ liệu lớn thông qua Spark.

7.5. Ứng Dụng Phân Tích Dữ Liệu Trong Các Lĩnh Vực Đặc Thù

Xu hướng: Các ngành công nghiệp khác nhau đang đẩy mạnh việc áp dụng phân tích dữ liệu để tối ưu hóa hoạt động và nâng cao hiệu suất.

Ví dụ:

- o Y tế: Dự đoán bệnh, tối ưu hóa lịch khám bệnh.
- o Tài chính: Quản lý rủi ro, tối ưu hóa danh mục đầu tư.
- o Marketing: Cá nhân hóa chiến dịch tiếp thị và phân khúc khách hàng.

Ý nghĩa: Phân tích dữ liệu sẽ trở thành trung tâm của các chiến lược kinh doanh và vận hành trong từng ngành nghề.

TÀI LIỆU THAM KHẢO

- [1] C01_DataAnalysis_Introduction.pdf – Lê Văn Hạnh
- [2] C02_DataCollection.pdf – Lê Văn Hạnh
- [3] C03_Getting to Know Your Data.pdf – Lê Văn Hạnh
- [4] C04_Data Preprocessing.pdf – Lê Văn Hạnh
- [5] C05_Pandas.pdf – Lê Văn Hạnh
- [6] C06_Data Loading, Storage & File Formats.pdf – Lê Văn Hạnh
- [7] C07_Data Cleaning & Preparation.pdf – Lê Văn Hạnh
- [8] C08_Data Wrangling_Join_Combine&Reshape.pdf – Lê Văn Hạnh
- [9] C09_Data Aggregation&Group Operations.pdf – Lê Văn Hạnh
- [10] <https://www.w3schools.com/python/pandas/default.asp> - Pandas tutorial
- [11] <https://www.kaggle.com/> - Kaggle
- [12] C01_DataScience_Introduction.pdf – Lê Văn Hạnh
- [13] C02_Getting to Know Your Data.pdf – Lê Văn Hạnh
- [14] C03_Data_Preprocessing.pdf – Lê Văn Hạnh
- [15] C04_MiningFrequentPatterns_Associations&Correlations.pdf – Lê Văn Hạnh
- [16] C05_Classification.pdf – Lê Văn Hạnh
- [17] C06_Cluster Analysis.pdf – Lê Văn Hạnh
- [18] C06_Cluster Analysis backup.pdf – Lê Văn Hạnh
- [19] Numpy_Pandas_ver2.pdf – Lê Văn Hạnh

KẾT LUẬN

Em xin gửi lời biết ơn sâu sắc đến thầy Lê Văn Hạnh đã dành nhiều thời gian và tâm huyết hướng dẫn nghiên cứu và giúp em hoàn thành môn học.

Em đã có nhiều cố gắng hoàn thiện dự án bằng tất cả năng lực của mình, tuy nhiên không thể tránh khỏi nhiều thiếu sót, rất mong nhận được những đóng góp quý báu của quý thầy cô và các bạn.