Mục Lục

[Lời cảm ơn i](#_Toc122652924)

[Lời mở đầu ii](#_Toc122652925)

[Mục Lục v](#_Toc122652926)

[Danh Mục Hình viii](#_Toc122652927)

[Danh Mục Bảng ix](#_Toc122652928)

[CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI 1](#_Toc122652929)

[1.1 Lý do chọn đề tài 1](#_Toc122652930)

[1.2 Mục tiêu đề tài 2](#_Toc122652931)

[1.3 Đối tượng nghiên cứu 2](#_Toc122652932)

[1.4 Phạm vi nghiên cứu 2](#_Toc122652933)

[1.5 Các thư viện được sử dụng 2](#_Toc122652934)

[1.5.1 Thư viện Pandas 2](#_Toc122652935)

[1.5.2 Thư viện Scikit-learn 3](#_Toc122652936)

[1.5.3 Thư viện Numpy 3](#_Toc122652937)

[1.5.4 Thư viện Matplotlib 3](#_Toc122652938)

[CHƯƠNG 2. CÁC NGHIÊN CỨU LIÊN QUAN 4](#_Toc122652939)

[2.1 Nghiên cứu trong nước 4](#_Toc122652940)

[2.1.1 Nghiên cứu của Trường Đại học Bách khoa Hà Nội (Trường Đại học Bách khoa Hà Nội, 2018) 4](#_Toc122652941)

[2.1.2 Nghiên cứu của Trường Đại học Thương mại Hà Nội (Trường Đại học Thương mại Hà Nội, 2021) 5](#_Toc122652942)

[2.2 Nghiên cứu ngoài nước 6](#_Toc122652943)

[2.2.1 Nghiên cứu của Đại học Bách Khoa Tây Bắc (Tây An – Trung Quốc) (Đại học Bách Khoa Tây Bắc - Trung Quốc, 2018) 6](#_Toc122652944)

[2.2.2 Nghiên cứu của Đại học Thành Quân Quán (Sungkyunkwan – Seoul – Hàn quốc) (Đại học Thành Quân Quán - Hàn Quốc, 2021) 7](#_Toc122652945)

[CHƯƠNG 3. QUY TRÌNH VÀ CÁC THUẬT TOÁN SỬ DỤNG 8](#_Toc122652946)

[3.1 Xây dựng quy trình cho mô hình học máy 8](#_Toc122652947)

[3.2 Thuật toán Cây Quyết Định 9](#_Toc122652948)

[3.2.1 Cây Quyết Định Là Gì? 9](#_Toc122652949)

[3.2.2. Phân loại cây quyết định 10](#_Toc122652950)

[3.3. Độ đo Entropy 11](#_Toc122652951)

[3.4. Các độ đo trong cây quyết định 11](#_Toc122652952)

[3.5 Thuật toán Hồi Quy Logistic 12](#_Toc122652953)

[3.5.1 Hồi quy logistic là gì? 12](#_Toc122652954)

[3.5.2 Tầm quan trọng của hồi quy logistic 12](#_Toc122652955)

[3.5.3 Ứng dụng của hồi quy logistic 13](#_Toc122652956)

[3.5.4 Mô hình hồi quy logistic hoạt động 14](#_Toc122652957)

[3.6.5 Phân tích hồi quy logistic với nhiều biến độc lập 17](#_Toc122652958)

[3.6.6 Log của tỷ số odds 17](#_Toc122652959)

[3.6 Phân loại hồi quy logistic 18](#_Toc122652960)

[3.7 Mô tả mục tiêu và các tiêu chí so sánh 18](#_Toc122652961)

[CHƯƠNG 4. XÂY DỰNG MÔ HÌNH 19](#_Toc122652962)

[4.1 Giới thiệu về tập dữ liệu (Dataset, 2022) 19](#_Toc122652963)

[4.2 Cấu hình máy để thực nghiệm 19](#_Toc122652964)

[4.3 Tiền xử lý dữ liệu 20](#_Toc122652965)

[4.3.1 Các trường của dataset 20](#_Toc122652966)

[4.3.2 Kiểm tra các giá trị rỗng trong dataset 21](#_Toc122652967)

[4.3.3 Trực quan hoá dữ liệu 22](#_Toc122652968)

[4.3.4 Trích chọn các trường dữ liệu 23](#_Toc122652969)

[4.4 Thực hiện mô hình 25](#_Toc122652970)

[4.4.1 Thuật toán Cây quyết định 25](#_Toc122652971)

[4.4.2 Thuật toán Hồi quy Logistic 26](#_Toc122652972)

[4.5 So sánh kết quả sau học máy 27](#_Toc122652973)

[CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 29](#_Toc122652974)

[5.1 Kết luận 29](#_Toc122652975)

[5.2 Hướng phát triển trong tương lai 29](#_Toc122652976)

[TÀI LIỆU THAM KHẢO 30](#_Toc122652977)

Danh Mục Hình

[Hình 1 Quy trình xây dựng mô hình học máy 8](#_Toc122650554)

[Hình 2 Minh hoạ về cây quyết định 10](#_Toc122650555)

[Hình 3 Đồ thị phương trình Y= 2\*x 15](#_Toc122650556)

[Hình 4 Đồ thị hàm số Sigmoid 16](#_Toc122650557)

[Hình 5 Kết quả in ra màn hình kết quả phân tán của dữ liệu 23](#_Toc122650558)

[Hình 6 Kết quả biểu đồ nhiệt mối tương quan giữa các trường dữ liệu 24](#_Toc122650559)

Danh Mục Bảng

[Bảng 1 Kết quả nghiên cứu của Đại học Thương Mại Hà Nội 5](#_Toc122894705)

[Bảng 2 Kết quả nghiên cứu của Đại học Bách Khoa Tây Bắc - Trung Quốc 7](#_Toc122894706)

[Bảng 3 Câu lệnh in ra toàn bộ dataset 20](#_Toc122894707)

[Bảng 4 Kết quả in ra màn hình toàn bộ dataset 21](#_Toc122894708)

[Bảng 5 Câu lệnh khái quát về dataset bằng hàm info 21](#_Toc122894709)

[Bảng 6 Câu lệnh Kiểm tra giá trị Null trong dataset 22](#_Toc122894710)

[Bảng 7 Câu lệnh Kiểm tra giá trị NaN trong dataset 22](#_Toc122894711)

[Bảng 8 Câu lệnh vẽ biểu đồ phân tán giá trị các cột trong dataset 23](#_Toc122894712)

[Bảng 9 Câu lệnh vẽ biểu đồ nhiệt để thể hiện mối tương quan giữa các trường dữ liệu 24](#_Toc122894713)

[Bảng 10 Câu lệnh phân chia dataset theo phương pháp Hold-Out tỷ lệ 8/2 25](#_Toc122894714)

[Bảng 11 Câu lệnh triển khai cây quyết định với thư viện Sklearn 26](#_Toc122894715)

[Bảng 12 Câu lệnh triển khai thuật toán Hồi Quy Logistic bằng thư viện Sklearn 26](#_Toc122894716)

[Bảng 13 Câu lệnh triển khai thuật toán Hồi Quy Logistic với bộ dữ liệu train có thêm cột ‘KhoiLuongGD’ 27](#_Toc122894717)

[Bảng 14 Bảng so sánh các kết quả sau quá trình học máy 28](#_Toc122894718)

[Bảng 15 Biểu đồ cột so sánh kết quả độ chính xác của các thuật toán 28](#_Toc122894719)

CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI

1.1 Lý do chọn đề tài

Công nghệ toàn cầu hiện nay vẫn luôn là điểm nóng, là chủ đề được bàn tán và nghiên cứu trong nhiều lĩnh vực. Đặc biệt là ứng dụng Trí Tuệ Nhân Tạo (AI) vào doanh nghiệp, đem lại nhiều kết quả ngoài mong đợi cho lợi nhuận. Nổi bật hơn hết là mô hình Hồi quy logistic và Cây quyết định. Chúng có chức năng phân lớp và gán nhãn, có thể giúp các tổ chức thu được thông tin chuyên sâu hữu ích từ dữ liệu kinh doanh của mình. Họ có thể sử dụng những thông tin này để phân tích dự đoán nhằm giảm chi phí hoạt động, tăng độ hiệu quả và điều chỉnh quy mô, nhất là đối với tập đoàn nghìn tỷ như Apple.

Bên cạnh đó, thị trường chứng khoán ngày càng mở rộng đầu tư. Số liệu thông tin về các phiên giao dịch được lưu trữ và công khai trên Internet để có thể dễ dàng thu thập. Đồng thời, đầu tư cổ phiếu được xem là xu hướng đầu tư sinh lợi thụ động phổ biến nhất thời đại kỹ thuật số. Nhưng thực tế, không phải ai “chơi” cổ phiếu cũng kiếm được số dư thuận lợi, thậm chí không ít trường hợp thua lỗ vì thiếu kiến thức, không hiểu quy tắc giao dịch. Từ đó, công nghệ AI được ứng dụng vào giúp thu thập hàng triệu dữ liệu trong thời gian thực tế và phân tích tự động, dự đoán biến động giá của từng cổ phiếu nói riêng, thị trường nói chung. Chúng sẽ hoạt động dựa trên dữ liệu về thời gian giao dịch, khối lượng giao dịch, giá mở cửa, giá đóng cửa và giá trần, giá sàn.

Chính những luận điểm trên, nhóm quyết định chọn đề tài “Ứng Dụng Mô Hình Hồi Quy Logistic Và Cây Quyết Định Vào Dự Đoán Phân Lớp Lợi Nhuận Apple” nhằm kết hợp cùng các thông tin chính xác và mới nhất về lịch sử giao cổ phiếu qua từng phiên, đưa ra các khuyến nghị mua hoặc bán cổ phiếu mang lại ít rủi ro nhất.

1.2 Mục tiêu đề tài

Khi hoàn thành môn học và chọn đề tài thực hiện, nhóm đặt ra mục tiêu rằng sẽ dự đoán được kết quả trong giao dịch chứng khoán của tập đoàn Apple có lợi nhuận hay không.

Nhóm sử dụng dataaset về lịch sử giao dịch chứng khoán của Apple Inc do tác giả MEERA SHAREEF đã thu thập được và đăng tải công khai trên website Kaggle.com. Nhóm dựa vào đấy và tiến hành các bước tiền xử lý dữ liệu kết hợp trực quan hóa dữ liệu, trích xuất các đặc trưng của bộ dữ liệu thu được và sẽ cho ra một bảng dataset hoàn chỉnh, gọi nôm na là dữ liệu sạch. Tiếp đến sẽ áp dụng 2 thuật toán phân lớp là Cây Quyết Định và Hồi Quy Logistic để thực hiện mục tiêu chung là dự đoán phân lớp Yes|No cũng như vận dụng các thang đo độ chính xác như: F1 core, Accuracy Core… nhằm tính được tỷ lệ chính xác đạt mức như thế nào. Khi hoàn thiện mô hình dùng đem chúng so sánh với nhau và đưa ra kết quả so sánh với mục đích chỉ rõ sự giống và khác của 2 phương pháp trên.

1.3 Đối tượng nghiên cứu

Dữ liệu lịch sử giao dịch Tập đoàn Apple trên sàn chứng khoán NASDAQ.

1.4 Phạm vi nghiên cứu

Không gian: Sàn chứng khoán NASDAQ.

Thời gian của bộ dữ liệu: Bắt đầu: 12-12-1980 cho đến: 28-10-2022.

1.5 Các thư viện được sử dụng

1.5.1 Thư viện Pandas

Là một thư viện Python cung cấp các cấu trúc dữ liệu nhanh, mạnh mẽ, linh hoạt và mang hàm ý. Tên thư viện được bắt nguồn từ panel data (bảng dữ liệu). Pandas được thiết kế để làm việc dễ dàng và trực quan với dữ liệu có cấu trúc (dạng bảng, đa chiều, có tiềm năng không đồng nhất) và dữ liệu chuỗi thời gian.

1.5.2 Thư viện Scikit-learn

Sklearn được thiết kế để xử lý các thư viện số và khoa học của Python như NumPy và SciPy. Các tính năng chính của thư viện Scikit-learning bao gồm thuật toán phân loại, hồi quy và phân cụm (hỗ trợ máy vectơ, rừng ngẫu nhiên, tăng độ dốc, k-means và DBSCAN).

1.5.3 Thư viện Numpy

Là một thư viện dành cho ngôn ngữ lập trình Python, hỗ trợ thêm cho mảng và ma trận lớn, nhiều chiều, cùng với một tập hợp lớn các hàm toán học cấp cao để hoạt động trên các mảng này. Tiền thân của NumPy, Numeric, ban đầu được tạo bởi Jim Hugunin với sự đóng góp của một số nhà phát triển khác. Năm 2005, Travis Oliphant đã tạo NumPy bằng cách kết hợp các tính năng của Numarray cạnh tranh vào Numeric, với nhiều sửa đổi. NumPy là phần mềm mã nguồn mở và có nhiều người đóng góp. NumPy là một dự án được tài trợ bởi NumFOCUS .

1.5.4 Thư viện Matplotlib

Matplotlib, một thư viện vẽ đồ thị cho Python vào năm 2003. Matplotlib là một thư viện vẽ đồ thị cấp thấp và là một trong những thư viện vẽ đồ thị được sử dụng rộng rãi nhất. Đây là một trong những lựa chọn đầu tiên để vẽ đồ thị để hiển thị nhanh một số dữ liệu.

Sử dụng Matplotlib, chúng ta có thể vẽ nhiều biểu đồ thú vị theo dữ liệu của mình như Biểu đồ thanh, Biểu đồ phân tán, Biểu đồ, Biểu đồ đường viền, Biểu đồ hộp, Biểu đồ hình tròn…. Chúng ta cũng có thể tùy chỉnh nhãn, màu sắc, độ dày của chi tiết biểu đồ theo chúng ta cần. Hình ảnh trên được vẽ chỉ bằng Matplotlib.

CHƯƠNG 2. CÁC NGHIÊN CỨU LIÊN QUAN

2.1 Nghiên cứu trong nước

2.1.1 [Nghiên cứu của Trường Đại học Bách khoa Hà Nội](https://www.researchgate.net/publication/330743514_Vietnam_Stock_Index_Trend_Prediction_using_Gaussian_Process_Regression_and_Autoregressive_Moving_Average_Model) (Trường Đại học Bách khoa Hà Nội, 2018)

Đề tài thực hiện: DỰ ĐOÁN XU THẾ CHỈ SỐ CHỨNG KHOÁN VIỆT NAM SỬ DỤNG PHÂN TÍCH HỒI QUY QUÁ TRÌNH GAUSS VÀ MÔ HÌNH TỰ HỒI QUY TRUNG BÌNH ĐỘNG [6]

Được xuất bản ngày 01/11/2018 trên tạp chí khoa học của Trường Đại học Bách khoa Hà Nội

Do 3 tác giả Huỳnh Quyết Thắng, Phùng Đình Vũ, Tống Văn Vinh thuộc Trường Đại học Bách khoa Hà Nội thực hiện

Tác giả áp dụng mô hình tự hồi quy trung bình động (ARMA: Autoregressive moving average) để dự đoán thành phần thời gian ngẫu nhiên ở một bước kế tiếp, phân tích hồi quy quá trình Gauss (GPR: Gaussian process regression) để dự đoán thành phần thời gian xu thế. Cuối cùng, kết quả dự đoán các thành phần riêng lẻ được tổng hợp lại để đưa ra kết quả dự đoán cuối cùng cho phương pháp kết hợp GPR-ARMA.

Thuật toán được sử dụng trong đề tài: Đường trung bình động hồi quy tự động (ARMA).

Với bộ dữ liệu chỉ số chứng khoán Việt Nam (VN-Index) được công khai trên các sàn giao dịch chứng khoán.

Cho ra độ chính xác của mô hình là 61,73%

2.1.2 [Nghiên cứu của Trường Đại học Thương mại Hà Nội](https://tuanvanle.files.wordpress.com/2021/01/ung-dung-mot-so-mo-hinh-hoc-may-trong-du-bao-chieu-bien-dong-cua-thi-truong-chung-khoan-viet-nam-1.pdf) (Trường Đại học Thương mại Hà Nội, 2021)

Đề tài thực hiện: ỨNG DỤNG MỘT SỐ MÔ HÌNH HỌC MÁY TRONG DỰ BÁO CHIỀU BIẾN ĐỘNG CỦA THỊ TRƯỜNG CHỨNG KHOÁN VIỆT NAM [7]

Xuất bản ngày 13/01/2021 trên tạp chí khoa học Trường Đại học Thương mại Hà Nội.

Do nhóm tác giả gồm: Lê Văn Tuấn, Nguyễn Thu Thủy, Lê Thị Thu Giang của Bộ môn Toán thực hiện.

Tác giả sử dụng một số mô hình/thuật toán học máy để dự báo xu hướng biến động (tăng/giảm) của chỉ số thị trường chứng khoán của Việt Nam. Kết quả cho thấy, trong các mô hình hồi quy Logistic, mô hình phân tích phân biệt tuyến tính (LDA), phân tích phân biệt toàn phương (QDA) và mô hình K – lân cận (KNN): mô hình KNN(10) có độ chính xác dự báo tốt nhất.

Bộ dữ liệu: VN-Index (chỉ số đại diện cho TTCK Việt Nam), download từ trang web của Công ty Cổ phần Chứng khoán VNDIRECT.

Cho ra độ chính xác lần lượt là:

|  |  |
| --- | --- |
| **Mô hình/thuật toán** | **Độ chính xác dự báo** |
| Hồi quy Logistic | 50.39% |
| LDA | 50% |
| QDA | 49.60% |
| KNN < 0.54 | < 54% |

Bảng 1 Kết quả nghiên cứu của Đại học Thương Mại Hà Nội

2.2 Nghiên cứu ngoài nước

2.2.1 [Nghiên cứu của Đại học Bách Khoa Tây Bắc (Tây An – Trung Quốc)](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8419702) (Đại học Bách Khoa Tây Bắc - Trung Quốc, 2018)

Đề tài thực hiện: THUẬT TOÁN TRÍ TUỆ NHÂN TẠO NÀO DỰ ĐOÁN TỐT HƠN THỊ TRƯỜNG CHỨNG KHOÁN TRUNG QUỐC [2]

Xuất bản ngày: 25/07/2018 trên tạp chí khoa học Đại Học Bách Khoa Tây Bắc (Trung Quốc)

Do nhóm tác giả: Lin Chen, Zhilin Qiao, Minggang Wang, Chao Wang, Ruijin Du, Harry Eugene Stanley thực hiện.

Nhóm tác giả nghiên cứu thị trường tài chính, giờ đây tác giả có thể trích xuất các tính năng từ môi trường dữ liệu lớn mà không cần thông tin dự đoán trước. Ở đây, tác giả đề xuất cải thiện hơn nữa hiệu suất dự đoán này bằng cách sử dụng kết hợp mô hình dự đoán hợp đồng tương lai chỉ số chứng khoán dựa trên học sâu, bộ mã hóa tự động và máy Boltzmann bị hạn chế. tác giả sử dụng dữ liệu tần suất cao để kiểm tra hiệu suất dự đoán của học sâu và tác giả so sánh ba mạng thần kinh nhân tạo truyền thống: 1) mạng thần kinh lan truyền ngược; 2) máy học cực đoan; và 3) mạng thần kinh chức năng cơ sở xuyên tâm. tác giả sử dụng tất cả dữ liệu giao dịch tần suất cao trong 1 phút của hợp đồng tương lai CSI 300 (IF1704) trong phân tích thực nghiệm của mình và tác giả thử nghiệm ba nhóm mẫu khối lượng khác nhau để xác thực các quan sát của mình. tác giả nhận thấy rằng phương pháp học sâu để dự đoán hợp đồng tương lai chỉ số chứng khoán vượt trội so với phương pháp lan truyền ngược, máy học cực đoan và mạng thần kinh chức năng cơ sở xuyên tâm ở mức độ phù hợp và độ chính xác dự đoán theo hướng. tác giả cũng thấy rằng việc tăng lượng dữ liệu sẽ làm tăng hiệu suất dự đoán. Điều này chỉ ra rằng học sâu nắm bắt các tính năng phi tuyến tính của dữ liệu giao dịch và có thể đóng vai trò là công cụ dự đoán hợp đồng tương lai chỉ số chứng khoán mạnh mẽ cho các nhà đầu tư trên thị trường tài chính.

Thuật toán được sử dụng trong đề tài: BP(Lan truyền ngược), ELM(Mạng nơ-ron truyền thẳng một lớp ẩn), RBF(Thuật toán nội suy)

Bộ dữ liệu về chỉ số thị trường chứng khoán trung quốc CSI 300 download từ website Sở giao dịch chứng khoán Thượng Hải

Độ đo RMSE của 3 thuật toán cho ra:

|  |  |
| --- | --- |
| Thuật toán | Độ đo RMSE |
| BP | 5,1203 |
| ELM | 3,2245 |
| RBF | 3,9305 |

Bảng 2 Kết quả nghiên cứu của Đại học Bách Khoa Tây Bắc - Trung Quốc

2.2.2 [Nghiên cứu của Đại học Thành Quân Quán (Sungkyunkwan – Seoul – Hàn quốc)](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9424620) (Đại học Thành Quân Quán - Hàn Quốc, 2021)

Đề tài: HỆ THỐNG CẢNH BÁO THỊ TRƯỜNG DỰA TRÊN MÁY HỌC CHO THỊ TRƯỜNG CHỨNG KHOÁN HÀN QUỐC [3]

Xuất bản: 06/05/2021 đăng tải Tạp chí khoa học Đại học Thành Quân Quán (Sungkyunkwan)

Do 2 nghiên cứu sinh Tiến sĩ: Daehyeon Park, Doojin Ryu thực hiện

Đề tài này xây dựng một hệ thống cảnh báo sớm (EWS) dựa trên máy học cho thị trường chứng khoán bằng cách sử dụng mạng thần kinh bộ nhớ ngắn hạn dài (LSTM).

Thuật toán được sử dụng: LSTM

Bộ dữ liệu: KOSDAQ(chỉ số đại diện cho TTCK Hàn Quốc)

Độ chính xác mô hình: Thuật toán LSTM: 99.15%

CHƯƠNG 3. QUY TRÌNH VÀ CÁC THUẬT TOÁN SỬ DỤNG

3.1 Xây dựng quy trình cho mô hình học máy

Khâu chuẩn bị mô hình vô cùng quan trọng do nó sẽ khái quát những việc cần làm nhằm mục đích khai phá được tri thức từ bộ dữ liệu và tiến dành dự đoán với độ chính xác cao nhất.

Hình 1 Quy trình xây dựng mô hình học máy

Quy trình gồm có 6 bước:

Bước 1: Sử dụng dataset về dữ liệu giá cổ phiếu tập đoàn Apple

Bước 2: Tiền xử lý dữ liệu bao gồm các việc: kiểm tra và làm sạch, gán nhãn các ô còn trống, loại bỏ các trường giá trị không liên quan,...

Bước 3: Chia bộ dữ liệu: dùng phương pháp Hold-Out với tỷ lệ 8/2

Bước 4: Xây dựng thuật toán: Cây quyết định, Hồi quy logistic

Bước 5: Tiến hành dự đoán phân lớp của mô hình dựa trên dataset

Bước 6: Dùng dộ đo accuracy\_core để đánh giá độ chính xác

3.2 Thuật toán Cây Quyết Định

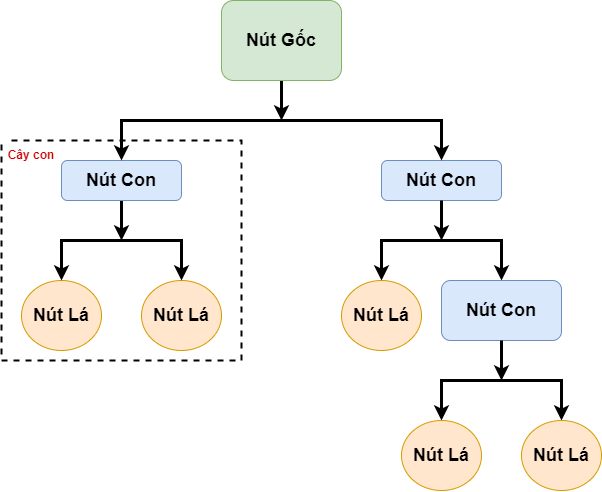
3.2.1 Cây Quyết Định Là Gì?

Cây quyết định là một phương tiện có tính mô tả dành cho việc tính toán các xác suất có điều kiện.

Thường được dùng để biểu diễn tri thức cho mục đích đưa ra quyết định hoặc kết luận.

Bắt đầu từ nút gốc, từ đó, người dùng tách thành các nút con một cách đệ quy theo luật toán học cây quyết định.

Kết quả cuối cùng là một cây quyết định trong đó mỗi nhánh biểu diễn một kịch bản có thể của quyết định và kết quả của nó.



Hình Minh hoạ về cây quyết định

Ví dụ ở **Hình 2** đã mô tả được cây quyết định là một cây trong đó:

* Nút gốc gọi là root node.
* Những nút được tách ra từ nút gốc gọi là nút con.
* Mỗi nút nhánh gọi là sub tree, nhằm biểu diễn một lựa chọn giữa một số khả năng.
* Mỗi nút lá có tên là leaf node để biểu diễn một quyết định.

3.2.2. Phân loại cây quyết định

* **Cây quyết định ID3**

ID3 biểu diễn các khái niệm ở dạng các cây quyết định. Hoạt động trên biến có giá trị rời rạc.

Biểu diễn này cho phép chúng ta xác định phân loại của một đối tượng bằng cách kiểm tra các giá trị của nó trên một số thuộc tính.

ID3 sử dụng độ đo Information Gain chọn thuộc tính phân chia cây.

* **Cây quyết định C4.5**

C4.5 là thuật toán cải tiến so với ID3.

Xử lý tốt cả hai biến: rời rạc, liên tục.

Bằng cách xác định phạm vi hoặc các ngưỡng cho dữ liệu liên tục nhờ vậy dữ liệu liên tục được chuyển sang dạng rời rạc.

Cây C4.5 Sử dụng Gain Ratio thay vì Information Gain.

3.3. Độ đo Entropy

Entropy là khái niệm trong lý thuyết thông tin cho phép lượng hóa thông tin.

Dùng đo lượng thông tin trong một thuộc tính của một bộ giá trị thu thập được của tập mẫu.

Đo mức nghi ngờ, mức ngẫu nhiên của kết luận, và nó biến thiên trong đoạn [0,1].

Cho tập dữ liệu S và xét thuộc tính kết luận C trên S có n giá trị, entropy của C được tính là:

Ví dụ về Entropy: Xét lớp có 35 sinh viên, trong đó có 25 sinh viên đi học, 10 sinh viên vắng. Tính entropy của lớp học trong trường học này.

E(LopHoc) = – (25/35)log2 (25/35) – (10/35)log2 (10/35)

= – (25/35)\*(–0.485) – (10/35)\*(–1.807)

= 0.863

3.4. Các độ đo trong cây quyết định

* **Độ lợi thông tin Information Gain**

Độ lợi thông tin (Information Gain) là độ đo được sử dụng trong thuật toán ID3

Công thức Information Gain:

=> Phân hoạch lại cây quyết định với Information Gain (IG càng lớn càng tốt).

Nhược điểm: của độ đo này là có xu hướng lựa chọn thuộc tính có nhiều giá trị

* **Độ đo lựa chọn thuộc tính Gain ratio**

Sử dụng trong thuật toán C4.5

Gain ratio khắc phục nhược điểm của độ đo Information gain.

Một thuộc tính A được chọn nếu nó có giá trị Gain Ratio lớn nhất.

Công thức:

3.5 Thuật toán Hồi Quy Logistic

3.5.1 Hồi quy logistic là gì?

Hồi quy logistic là một kỹ thuật phân tích dữ liệu sử dụng toán học để tìm ra mối quan hệ giữa hai yếu tố dữ liệu. Sau đó, kỹ thuật này sử dụng mối quan hệ đã tìm được để dự đoán giá trị của những yếu tố đó dựa trên yếu tố còn lại. Dự đoán thường cho ra một số kết quả hữu hạn, như có hoặc không.

3.5.2 Tầm quan trọng của hồi quy logistic

Hồi quy logistic là một kỹ thuật quan trọng trong lĩnh vực trí tuệ nhân tạo và máy học (AI/ML). Mô hình ML là các chương trình phần mềm có thể được đào tạo để thực hiện các tác vụ xử lý dữ liệu phức tạp mà không cần sự can thiệp của con người. Mô hình ML được xây dựng bằng hồi quy logistic có thể giúp các tổ chức thu được thông tin chuyên sâu hữu ích từ dữ liệu kinh doanh của mình. Họ có thể sử dụng những thông tin chuyên sâu này để phân tích dự đoán nhằm giảm chi phí hoạt động, tăng độ hiệu quả và đổi chỉnh quy mô nhanh hơn.

Dưới đây là một số lợi ích của việc sử dụng hồi quy logistic so với các ký thuật Ml khác:

* **Tính đơn giản**

Các mô hình hồi quy logistic ít phức tạp về mặt toán học hơn các phương pháp ML khác. Do đó, chúng ta có thể triển khai chúng ngay cả khi đội ngũ của chúng ta không ai có chuyên môn sâu về ML.

* **Tốc độ**

Các mô hình hồi quy logistic có thể xử lý khối lượng lớn dữ liệu ở tốc độ cao bởi chúng cần ít khả năng điện toán hơn, chẳng hạn như bộ nhớ và sức mạnh xử lý. Điều này khiến các mô hình hồi quy logistic trở nên lý tưởng đối với những tổ chức đang bắt đầu với các dự án ML để đạt được một số thành tựu nhanh chóng.

* **Sự linh hoạt**

Chúng ta có thể sử dụng hồi quy logistic để tìm đáp án cho các câu hỏi có hai hoặc nhiều kết quả hữu hạn. Chúng ta cũng có thể sử dụng phương pháp này để xử lý trước dữ liệu.

* **Khả năng hiển thị**

Phân tích hồi quy logistic cung cấp cho nhà phát triển khả năng nhìn nhận các quy trình phần mềm nội bộ rõ hơn so với các kỹ thuật phân tích dữ liệu khác. Khắc phục sự cố và sửa lỗi cũng trở nên dễ dàng hơn do các phép toán ít phức tạp hơn.

3.5.3 Ứng dụng của hồi quy logistic

Hồi quy logistic có một số ứng dụng thực tế trong nhiều ngành công nghiệp khác nhau.

* **Sản xuất**

Các công ty sản xuất áp dụng phân tích hồi quy logistic để ước tính xác suất xảy ra sự cố ở bộ phận trong máy móc. Sau đó, họ sẽ lên lịch bảo trì dựa trên xác suất đã ước tính này để giảm thiểu sự cố trong tương lai.

* **Chăm sóc sức khỏe**

Các nhà nghiên cứu y khoa lên kế hoạch điều trị và chăm sóc dự phòng bằng cách dự đoán khả năng mắc bệnh ở bệnh nhân. Họ sử dụng các mô hình hồi quy logistic để so sánh tác động của tiền sử gia đình hoặc của bộ gen lên bệnh tật.

* **Tài chính**

Các công ty tài chính phải phân tích các giao dịch tài chính để đề phòng gian lận, xem xét các đơn xin vay và đơn bảo hiểm để đề phòng rủi ro. Những vấn đề này phù hợp với mô hình hồi quy logistic bởi chúng có kết quả cụ thể, chẳng hạn như rủi ro cao hoặc rủi ro thấp và gian lận hoặc không gian lận.

* **Bộ phận tiếp thị**

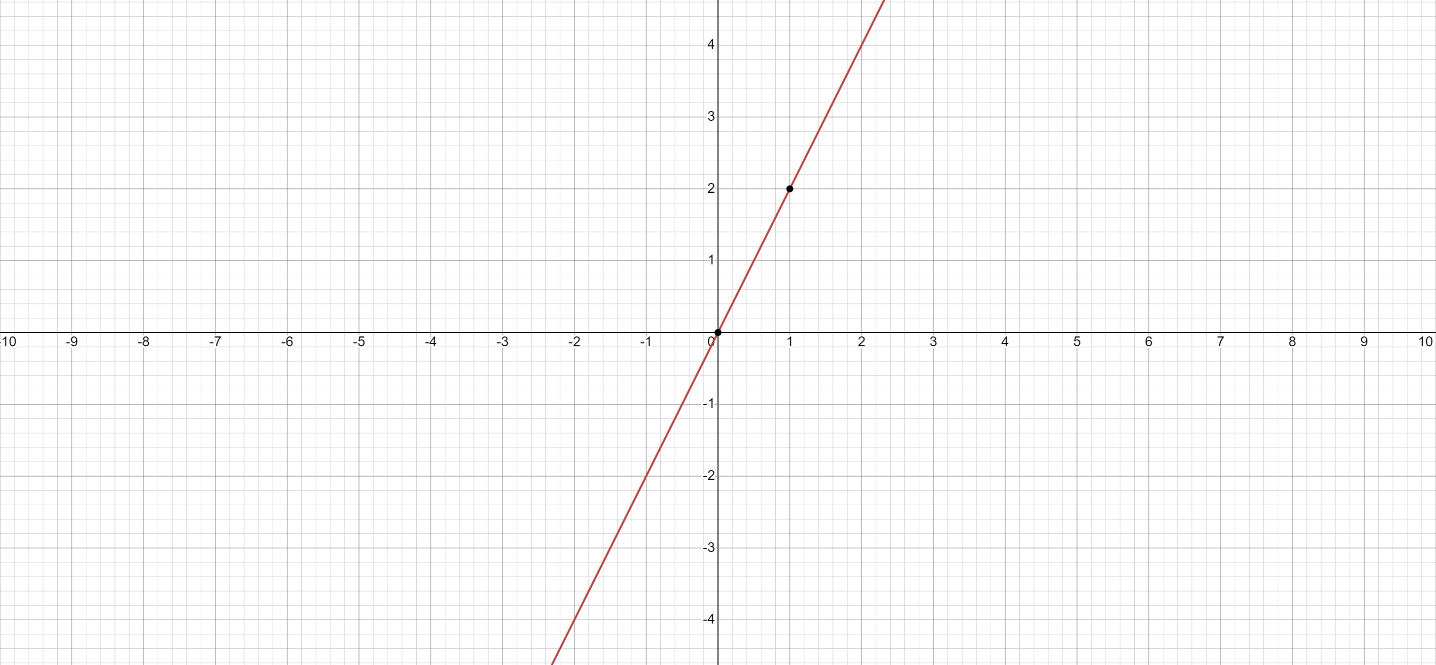
Các công cụ quảng cáo trực tuyến sử dụng mô hình hồi quy logistic để dự đoán xem người dùng sẽ nhấp vào một quảng cáo hay không. Kết quả là, các nhà tiếp thị có thể phân tích phản ứng của người dùng đối với những từ ngữ và hình ảnh khác nhau, tạo ra các quảng cáo hiệu suất cao có khả năng thu hút khách hàng.

3.5.4 Mô hình hồi quy logistic hoạt động

**Phương trình**

Trong toán học, phương trình cho ta mối quan hệ giữa hai biến: *x* và *y*. Chúng ta có thể sử dụng các phương trình hoặc hàm này để vẽ đồ thị theo trục x và trục y bằng cách nhập các giá trị khác nhau của *x* và *y*.

Ví dụ: nếu chúng ta vẽ đồ thị cho hàm *y* = 2\**x*, chúng ta sẽ có một đường thẳng như hình dưới đây. Do đó hàm này còn được gọi là hàm tuyến tính.



Hình Đồ thị phương trình Y= 2\*x

**Biến hồi quy logistic**

Trong thống kê, biến là các yếu tố dữ liệu hoặc thuộc tính có giá trị khác nhau. Bất kỳ phân tích nào cũng có một số biến nhất định là biến độc lập hoặc biếngiải thích.Những thuộc tính này là nguyên nhân của một kết quả. Các biến khác là biến phụ thuộc hoặc biến đáp ứng; giá trị của chúng phụ thuộc vào các biến độc lập.

Nhìn chung, hồi quy logistic khám phá cách các biến độc lập ảnh hưởng đến một biến phụ thuộc bằng cách xem xét các giá trị dữ liệu lịch sử của cả hai biến.

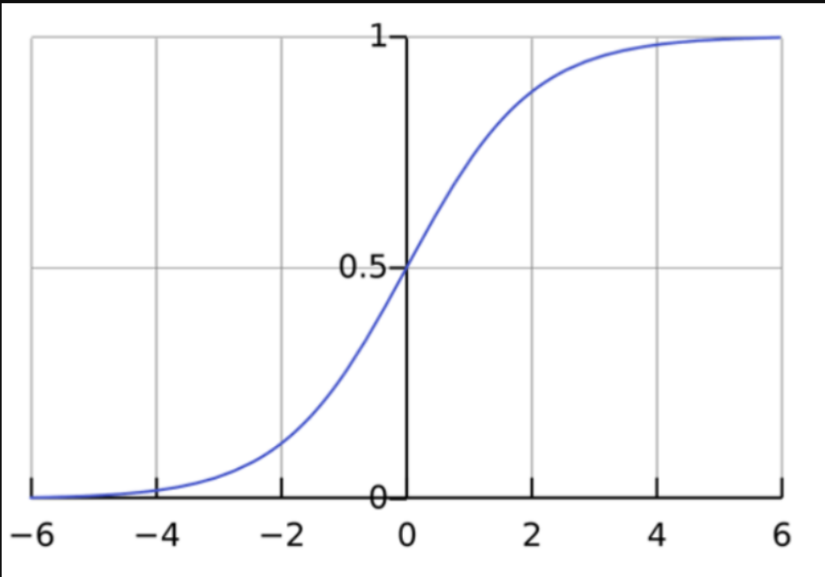
**Hàm hồi quy logistic**

Hồi quy logistic là một mô hình thống kê sử dụng hàm logistic, hay hàm logit trong toán học làm phương trình giữa *x* và *y*. Hàm logit ánh xạ *y* làm hàm sigmoid của *x*.

**Sigmoid Function là gì?**

Hàm số Sigmoid

Sigmoid Function (Hàm Sigmoid) còn được gọi là đường cong Sigmoid. Đây là một hàm toán học có đặc trưng là đường cong hình chữ S. Nó thể hiện cho sự biến đổi các giá trị giữa phạm vi 0 và 1. Nó là một trong những hàm kích hoạt (activation function) phi tuyến tính được sử dụng rộng rãi nhất.



Hình Đồ thị hàm số Sigmoid

**Sigmoid Function dùng để làm gì?**

Tất cả các hàm Sigmoid đều có một đặc điểm chung. Chúng có thể chuyển những con số đầu vào thành một phạm vi nhỏ nhất định. Cụ thể, các con số đầu vào sẽ chuyển thành từ 0 đến 1 hoặc -1 và 1. Nghĩa là, Hàm Sigmoid dùng để chuyển một giá trị thực thành một giá trị kiểu xác suất.

Hàm Sigmoid sẽ nhận đầu vào (input) và thực hiện những công việc sau:

* Nếu biến đầu vào âm, hàm Sigmoid sẽ chuyển gần như tất cả thành một số gần với 0.
* Với gần như tất cả đầu vào dương, hàm Sigmoid sẽ biến đầu vào thành một số gần với 1.
* Trường hợp đầu vào tương đối gần 0, hàm Sigmoid sẽ chúng thành số bất kỳ từ 0 đến 1.

3.6.5 Phân tích hồi quy logistic với nhiều biến độc lập

Trong nhiều trường hợp, nhiều biến giải thích ảnh hưởng đến giá trị của biến phụ thuộc. Để lập mô hình các tập dữ liệu đầu vào như vậy, công thức hồi quy logistic phải giả định mối quan hệ tuyến tính giữa các biến độc lập khác nhau. Chúng ta có thể sửa đổi hàm sigmoid và tính toán biến đầu ra cuối cùng như sau.

* Ký hiệu β đại diện cho hệ số hồi quy.
* Mô hình logit có thể đảo ngược tính toán các giá trị hệ số này khi chúng ta cho nó một tập dữ liệu thực nghiệm đủ lớn có các giá trị đã xác định của cả hai biến phụ thuộc và biến độc lập.

3.6.6 Log của tỷ số odds

Mô hình logit cũng có thể xác định tỷ số thành công trên thất bại hay log của tỷ số odds.

Ví dụ: nếu đang chơi poker với bạn bè và thắng bốn ván trên mười ván, tỷ số chiến thắng của bạn là bốn phần sáu, hoặc 4/6 và đó là tỷ số thành công trên thất bại của bạn. Mặt khác, xác suất thắng là 4/10.

Về mặt toán học, tỷ số odds về mặt xác suất là p/(1 - p) và log của tỷ số odds là log (p/(1 - p)). Có thể biểu diễn hàm logistic bằng log của tỷ số odds

Phương trình Log Odds Logit:

3.6 Phân loại hồi quy logistic

Có ba cách tiếp cận phân tích hồi quy logistic dựa trên kết quả của biến phụ thuộc.

* **Hồi quy logistic nhị phân**

Hồi quy logistic nhị phân phù hợp với các vấn đề phân lớp nhị phân chỉ có hai kết quả có thể xảy ra. Biến phụ thuộc chỉ có thể có hai giá trị, chẳng hạn như có và không hoặc 0 và 1.

* **Hồi quy logistic đa thức**

Hồi quy đa thức có thể phân tích các vấn đề có một số kết quả có thể xảy ra, miễn là số kết quả hữu hạn.

Ví dụ: kỹ thuật này có thể dự đoán xem giá nhà sẽ tăng 25%, 50%, 75% hay 100% dựa trên dữ liệu dân số, nhưng sẽ không thể dự đoán được giá trị chính xác của một ngôi nhà.

* **Hồi quy logistic thứ tự**

Hồi quy logistic thứ tự, hay mô hình logit có thứ tự, là một loại hồi quy đa thức đặc biệt cho các vấn đề trong đó các số đại diện cho các bậc chứ không phải là giá trị thực tế.

3.7 Mô tả mục tiêu và các tiêu chí so sánh

Sử dụng thư viện Matplotlib và Seaborn để trực quan hoá dữ liệu, vẽ biểu đồ nhiệt so sánh mức độ liên quan giữa các cột, sau đó loại bỏ những cột không liên quan và đưa dataset “sạch” này vào mô hình.

Sử dụng 2 thuật toán phân lớp là Hồi Quy Logistic và Cây quyết định để cho máy học sau đó so sánh độ chính xác của từng mô hình.

Từ 10559 dòng dữ liệu sẽ lấy ngẫu nhiên ra 80% để thực hiện train và 20% còn lại dùng để test sau đó sẽ dùng “F1 Core”, “Accuracy Core” để tính độ chính xác.

CHƯƠNG 4. XÂY DỰNG MÔ HÌNH

4.1 Giới thiệu về tập dữ liệu (Dataset, 2022) [1]

Dataset được thu thập tại website Kaggle (Kaggle.com, 2022) [4]

Lịch sử doanh thu hàng năm/hàng quý của Apple và tốc độ tăng trưởng từ năm 1980 đến năm 2022. Doanh thu có thể được định nghĩa là số tiền mà một công ty nhận được từ khách hàng của mình để đổi lấy việc bán hàng hóa hoặc dịch vụ. Doanh thu là mục hàng đầu trên báo cáo thu nhập mà từ đó tất cả các chi phí và phí tổn được trừ đi để đạt được thu nhập ròng.

* Doanh thu của Apple trong quý kết thúc vào ngày 30 tháng 6 năm 2022 là 82,959 tỷ USD, tăng 1,87% so với cùng kỳ năm trước.
* Doanh thu của Apple trong mười hai tháng kết thúc vào ngày 30 tháng 6 năm 2022 là $387,542 tỷ, tăng 11,63% so với năm trước.
* Doanh thu hàng năm của Apple trong năm 2021 là $365,817 tỷ, tăng 33,26% so với năm 2020.
* Doanh thu hàng năm của Apple trong năm 2020 là $274,515 tỷ, tăng 5,51% so với năm 2019.
* Doanh thu hàng năm của Apple trong năm 2019 là $260,174 tỷ, giảm 2,04% so với năm 2018.

4.2 Cấu hình máy để thực nghiệm

* Môi trường: Ngôn ngữ: Python 3.7
* Sản phẩm được viết và thực thi tại Google Colaboratory [5].
* Chạy trên thiết bị: Asus Tuf A15, CPU AMD Ryzen 5 4600H, Ram 8GB
* Bộ dữ liệu được lưu trữ tại Drive. [1]
* Địa chỉ mã nguồn: Google Colab (Trần Cao Minh, 2022) [5]

4.3 Tiền xử lý dữ liệu

4.3.1 Các trường của dataset

Bộ dữ liệu thu thập từ giá trị Cổ phiếu của tập đoàn Apple trên sàn chứng khoán NASDAQ

Bộ dữ liệu bao gồm 7 cột và có 10.559 dòng:

* 1. Ngày tháng (NgayThang)
  2. Giá mở cửa (Gia\_MoCua)
  3. Giá trần (Gia\_Tran)
  4. Giá sàn (Gia\_San)
  5. Giá đóng cửa (Gia\_DongCua)
  6. Khối lượng giao dịch (KhoiLuongGD)
  7. Lợi nhuận hay không (LoiNhuan\_Hay\_Khong)

|  |
| --- |
| *# Câu lệnh in ra toàn bộ dataset*  *data[['NgayThang','Gia\_MoCua','Gia\_Tran','Gia\_San','Gia\_DongCua','KhoiLuongGD','LoiNhuan\_Hay\_Khong']]* |

Bảng Câu lệnh in ra toàn bộ dataset

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Ngay  Thang | Gia\_MoCua | Gia\_Tran | Gia\_San | Gia\_DongCua | KhoiLuongGD | LoiNhuan\_Hay\_Khong |
| 0 | 12/12/1980 | 0.1003 | 0.1007 | 0.1003 | 0.1003 | 469033600 | 0 |
| 1 | 15/12/1980 | 0.0955 | 0.0955 | 0.0951 | 0.0951 | 175884800 | 1 |
| 2 | 16/12/1980 | 0.0885 | 0.0885 | 0.0881 | 0.0881 | 105728000 | 1 |
| 3 | 17/12/1980 | 0.0902 | 0.0907 | 0.0902 | 0.0902 | 86441600 | 0 |
| 4 | 18/12/1980 | 0.0929 | 0.0933 | 0.0929 | 0.0929 | 73449600 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 10554 | 24/10/2022 | 1.471.850 | 1.502.300 | 1.460.000 | 1.494.500 | 75569074 | 0 |
| 10555 | 25/10/2022 | 1.500.900 | 1.524.900 | 1.493.600 | 1.523.400 | 73583421 | 0 |
| 10556 | 26/10/2022 | 1.509.600 | 1.519.900 | 1.480.400 | 1.493.500 | 87769611 | 1 |
| 10557 | 27/10/2022 | 1.480.700 | 1.490.460 | 1.441.300 | 1.448.000 | 108110875 | 1 |
| 10558 | 28/10/2022 | 1.482.000 | 1.575.000 | 1.478.200 | 1.557.400 | 162857942 | 0 |

Bảng 4 Kết quả in ra màn hình toàn bộ dataset

Sau đó sẽ dùng hàm info() được hỗ trợ bởi thư viện Sklearn để xem khái quát về dataset.

|  |
| --- |
| *# Khái quát về dataset*  *data.info()* |
| *>>Output*  *<class 'pandas.core.frame.DataFrame'>*  *RangeIndex: 10559 entries, 0 to 10558*  *Data columns (total 7 columns):*  *# Column Non-Null Count Dtype*  *--- ------ -------------- -----*  *0 NgayThang 10559 non-null object*  *1 Gia\_MoCua 10559 non-null float64*  *2 Gia\_Tran 10559 non-null float64*  *3 Gia\_San 10559 non-null float64*  *4 Gia\_DongCua 10559 non-null float64*  *5 KhoiLuongGD 10559 non-null int64*  *6 LoiNhuan\_Hay\_Khong 10559 non-null int64*  *dtypes: float64(4), int64(2), object(1)*  *memory usage: 577.6+ KB* |

Bảng 5 Câu lệnh khái quát về dataset bằng hàm info

4.3.2 Kiểm tra các giá trị rỗng trong dataset

|  |
| --- |
| *''' Kiểm tra xem có giá trị nào*  *trong dataset bị Null hay không '''*  *data.isnull().sum()* |
| *>>Output:*  *NgayThang 0*  *Gia\_MoCua 0*  *Gia\_Tran 0*  *Gia\_San 0*  *Gia\_DongCua 0*  *KhoiLuongGD 0*  *LoiNhuan\_Hay\_Khong 0*  *dtype: int64* |

Bảng Câu lệnh Kiểm tra giá trị Null trong dataset

→ Kết quả trả về không có giá trị Null trong dataset

|  |
| --- |
| *''' Kiểm tra xem có giá trị nào*  *trong dataset bị NaN hay không '''*  *data.isna().sum()* |
| *>>Output:*  *NgayThang 0*  *Gia\_MoCua 0*  *Gia\_Tran 0*  *Gia\_San 0*  *Gia\_DongCua 0*  *KhoiLuongGD 0*  *LoiNhuan\_Hay\_Khong 0*  *dtype: int64* |

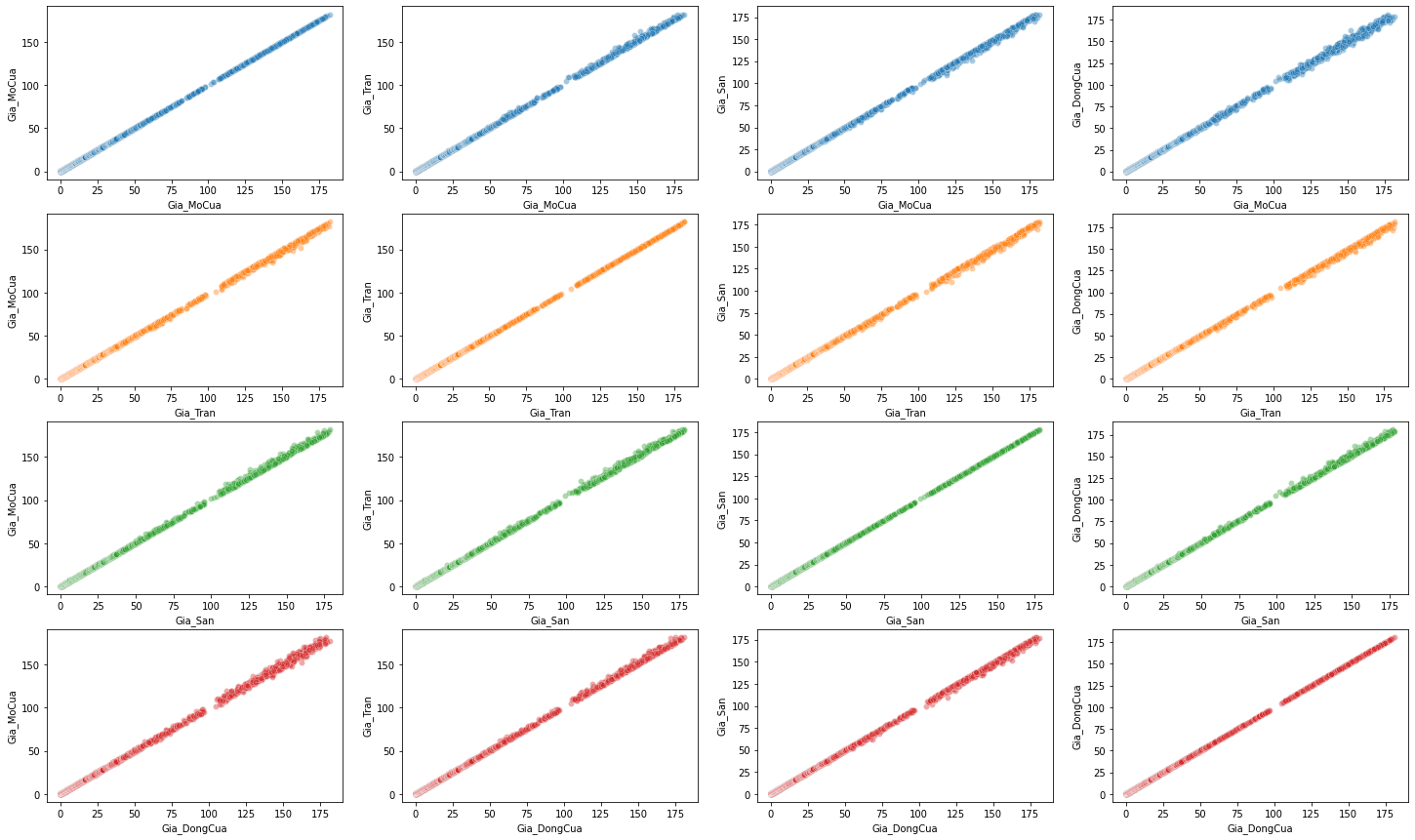
Bảng Câu lệnh Kiểm tra giá trị NaN trong dataset

4.3.3 Trực quan hoá dữ liệu

Dùng thư viện matplotlib để vẽ biểu đồ phân tán các giá trị của từng trường dữ liệu

|  |
| --- |
| *# Vẽ biểu đồ phân tán toàn bộ các giá trị trong dataset*  *plt.subplots(4,4,figsize=(25, 15))*  *n = 0*  *for x, c in zip(['Gia\_MoCua','Gia\_Tran','Gia\_San','Gia\_DongCua'], list(sns.color\_palette())):*  *for y in ['Gia\_MoCua','Gia\_Tran','Gia\_San','Gia\_DongCua']:*  *plt.subplot(4,4,n+1)*  *sns.scatterplot(x = x, y = y, data = data, color = c, alpha = 0.4)*  *n += 1* |

Bảng 8 Câu lệnh vẽ biểu đồ phân tán giá trị các cột trong dataset



Hình Kết quả in ra màn hình kết quả phân tán của dữ liệu

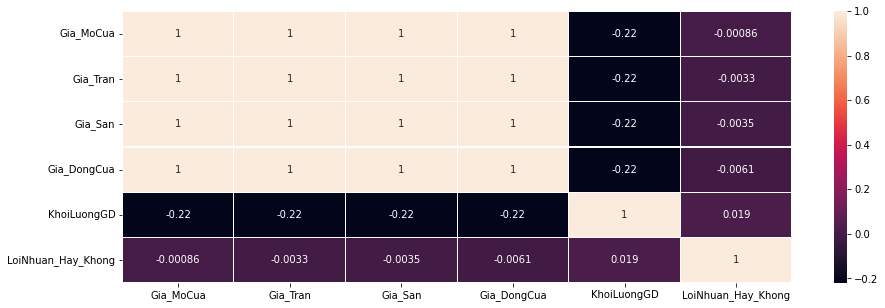
Hình 5 là kết quả của biểu đồ phân tán cho ra thấy rằng các giá trị tăng tuyến tính qua từng mốc thời gian khác nhau. Các giá trị từng cột khi phân tán thì không có giá trị nào bị lệch khỏi đường ‘tuyến tính’ quá xa, cho thấy rằng không có giá trị Outliers tồn tại trong bộ dữ liệu

4.3.4 Trích chọn các trường dữ liệu

Vẽ biểu đồ nhiệt để thể hiện mối tương quan giữa các trường dữ liệu giá trị trong đoạn từ [-1:1]. Với tiêu chí giá trị biểu đồ tỷ lệ thuận với mức độ liên quan giữa các trường

|  |
| --- |
| *plt.subplots(figsize=(15, 5))*  *# Biểu đồ nhiệt mối tương quan giữa các cột*  *sns.heatmap(data.corr(), linewidths=0.1, annot=True)* |

Bảng 9 Câu lệnh vẽ biểu đồ nhiệt để thể hiện mối tương quan giữa các trường dữ liệu



Hình Kết quả biểu đồ nhiệt mối tương quan giữa các trường dữ liệu

Kết quả cho ra cột ‘KhoiLuongGD’ có mức độ liên quan thấp nhất, nên sẽ bị loại khỏi bỏ dataset. Loại thêm cột ‘NgayThang’ do cột này chỉ mang tính chất đánh số thứ tự cho bộ dữ liệu.

4.4 Thực hiện mô hình

Tiến hành phân chia bộ dữ liệu theo phương pháp HOLD OUT với tỷ lệ là 8/2 tương ứng với 80% bộ dữ liệu sẽ được dùng để cho máy học và dùng 20% còn lại kiểm tra.

|  |
| --- |
| *# Phân chia dataset*  *# Lấy 4 Cột làm biến độc lập*  *inputs = data[['Gia\_MoCua','Gia\_Tran','Gia\_San','Gia\_DongCua']]*  *# Lấy 1 Cột làm biến phụ thuộc*  *targets = data['LoiNhuan\_Hay\_Khong']*  *# Lấy ngẫu nhiên 80% dòng để máy học và 20% dòng để kiểm tra*  *X\_train, X\_test, y\_train, y\_test = train\_test\_split(inputs, targets,*  *random\_state=16, test\_size = 0.2)* |

Bảng Câu lệnh phân chia dataset theo phương pháp Hold-Out tỷ lệ 8/2

4.4.1 Thuật toán Cây quyết định

Thư viện Sklearn hỗ trợ thuật toán Cây Quyết Định bằng hàm có tên DecisionTreeClassifier() dựa theo độ đo ‘Entropy’ với độ sâu tối đa là 3.

|  |
| --- |
| *# Cây Quyết Định*  *# Triển khai cây quyết định với độ đo là Entropy và độ sâu mỗi nút là 3*  *model\_CayQuyetDinh = DecisionTreeClassifier(criterion='entropy', max\_depth=3)*  *model\_CayQuyetDinh.fit(X\_train, y\_train)*  *prediction\_CayQuyetDinh = model\_CayQuyetDinh.predict(X\_test)*  *CayQuyetDinh\_Acc = accuracy\_score(y\_test, prediction\_CayQuyetDinh)*  *print(f'Từ bộ dữ liệu có {len(data)} dòng ta lấy ra {len(X\_train)} dòng cho máy học và dùng {len(X\_test)} dòng để kiểm tra')*  *print(f'Cây Quyết Định cho ra kết quả: {CayQuyetDinh\_Acc\*100}% Độ Chính Xác')* |
| >>Output  *Từ bộ dữ liệu có 10559 dòng ta lấy ra 8447 dòng cho máy học và dùng 2112 dòng để kiểm tra*  *Cây Quyết Định cho ra kết quả: 51.5625% Độ Chính Xác* |

Bảng 11 Câu lệnh triển khai cây quyết định với thư viện Sklearn

Sau khi đưa bộ dữ liệu train và test vào thuật toán Cây Quyết Định và thang đo độ chính xác là ‘Accuracy\_Core’ cho ra kết quả 51.5625% Độ Chính Xác

4.4.2 Thuật toán Hồi quy Logistic

Thư viện Sklearn cũng hỗ trợ thuật toán Hồi Quy Logistic với tên hàm là LogisticRegression()

|  |
| --- |
| *# Hồi Quy Logistic*  *# Triển khai thuật toán hồi quy logistic*  *model\_Logistic = LogisticRegression()*  *model\_Logistic.fit(X\_train, y\_train)*  *prediction\_Logistic = model\_Logistic.predict(X\_test)*  *Logistic\_Acc = accuracy\_score(y\_test, prediction\_Logistic)*  *print(f'Từ bộ dữ liệu có {len(data)} dòng ta lấy ra {len(X\_train)} dòng cho máy học và dùng {len(X\_test)} dòng để kiểm tra')*  *print(f'Hồi Quy Logistic cho ra kết quả: {Logistic\_Acc\*100}% Độ Chính Xác')* |
| *>>Output*  *Từ bộ dữ liệu có 10559 dòng ta lấy ra 8447 dòng cho máy học và dùng 2112 dòng để kiểm tra*  *Hồi Quy Logistic cho ra kết quả: 79.26136363636364% Độ Chính Xác* |

Bảng 12 Câu lệnh triển khai thuật toán Hồi Quy Logistic bằng thư viện Sklearn

Sau khi đưa bộ dữ liệu train và test vào thuật toán Hồi Quy Logistic và thang đo độ chính xác là ‘Accuracy\_Core’ cho ra kết quả 79.2613% Độ Chính Xác

Để đánh giá lại mức độ hợp lý khi loại bỏ cột dữ liệu ‘KhoiLuongGD’ ta tiến hành chạy lại thuật toán Hồi Quy Logistic và sử dụng thêm cột ‘KhoiLuongGD’ vào bộ dữ liệu train.

|  |
| --- |
| *# Hồi quy Logistic sử dụng thêm cột KhoiLuongGD*  *data[['NgayThang','Gia\_MoCua','Gia\_Tran','Gia\_San','Gia\_DongCua','KhoiLuongGD','LoiNhuan\_Hay\_Khong']]*  *X = data[['Gia\_MoCua','Gia\_Tran','Gia\_San','Gia\_DongCua','KhoiLuongGD']]*  *y = data['LoiNhuan\_Hay\_Khong']*  *X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, random\_state=40, test\_size = 0.1)*  *model = LogisticRegression()*  *model.fit(X\_train, y\_train)*  *prediction = model.predict(X\_test)*  *print(f'Hồi Quy Logistic sử dụng thêm cột KhoiLuongGD cho ra kết quả: {(accuracy\_score(y\_test, prediction))\*100}%')* |
| *>>Output*  *Hồi Quy Logistic sử dụng thêm cột KhoiLuongGD cho ra kết quả: 54.166%* |

Bảng 13 Câu lệnh triển khai thuật toán Hồi Quy Logistic với bộ dữ liệu train có thêm cột ‘KhoiLuongGD’

Với độ đo ‘Accuracy\_Core’ chỉ cho ra độ chính xác 54.1666%

4.5 So sánh kết quả sau học máy

Với cùng bộ dữ liệu đã được trích chọn thuộc tính và đưa vào 2 thuật toán khác nhau, sau khi trải qua bước học máy, thì cho ra 2 kết quả tính phần trăm độ chính xác dự đoán được với thang đo độ chính xác là Accuracy\_Core

|  |  |
| --- | --- |
| Độ đo Thuật toán | Accuracy\_Core |
| Cây quyết định | 51,5625% |
| Hồi quy Logistic | 54,1666% |
| Hồi quy Logistic thêm cột KhoiLuongGD | 79,2613% |

Bảng 14 Bảng so sánh các kết quả sau quá trình học máy

Bảng Biểu đồ cột so sánh kết quả độ chính xác của các thuật toán

CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Từ bộ dữ liệu nhóm download từ website Kanggle [1] tiếp đến nhóm tiến hành các bước của quy trình xây dựng mô hình và tiến hành thực nghiệm, đánh giá kết quả của mô hình.

Bộ dữ liệu bao gồm 7 cột và có 10.559 dòng:

1. Ngày tháng (NgayThang)
2. Giá mở cửa (Gia\_MoCua)
3. Giá trần (Gia\_Tran)
4. Giá sàn (Gia\_San)
5. Giá đóng cửa (Gia\_DongCua)
6. Khối lượng giao dịch (KhoiLuongGD)
7. Lợi nhuận hay không (LoiNhuan\_Hay\_Khong)

Kết quả so sánh các thuật toán hồi quy Logistic và cây quyết định cho thấy thuật toán hồi quy Logistic có độ chính xác tốt nhất. Và thậm chí còn tốt hơn nếu không có cột ‘KhoiLuongGD’.

Như vậy, sự biến động cột phân lớp lợi nhuận của bộ dữ liệu không phụ thuộc vào khối lượng giao dịch. Và bộ dữ liệu hoạt động tốt hơn với các thuật toán hồi quy.

5.2 Hướng phát triển trong tương lai

* Thêm giao diện trực quan cho người sử dụng.
* Thu thập và phân tích dữ liệu theo thời gian thực.
* Sử dụng thêm các thuật toán Neural Network như: RNN, ANN,... để tăng độ chính xác cho mô hình Machine Learning
* Dùng thêm PCA để giảm kích thước số chiều của dữ liệu từ đó tăng thêm tốc độ xử lý và độ chính xác của mô hình

TÀI LIỆU THAM KHẢO

1. Dataset. (2022, 12 23). *Địa chỉ file dữ liệu*. From File dữ liệu dưới dạng csv: https://drive.google.com/file/d/1Jjm7Mw4yRZUKcVLMBdk-2opeCG8t35P4/view?usp=sharing
2. Đại học Bách Khoa Tây Bắc - Trung Quốc. (2018, 07 25). *Ieeexplore*. From Which Artificial Intelligence Algorithm Better Predicts the Chinese Stock Market?: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8419702
3. Đại học Thành Quân Quán - Hàn Quốc. (2021, 05 06). *Ieeexplore*. From A Machine Learning-Based Early Warning System for the Housing and Stock Markets: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9424620
4. Kaggle.com. (2022, 10 23). *Kaggle Dataset*. From Kaggle datasets download -d meerashareef/apple-revenue-from-1980-to-2022: https://www.kaggle.com/datasets/meerashareef/apple-revenue-from-1980-to-2022
5. Trần Cao Minh. (2022, 12 23). *Thực thi tại Google Colab*. From Soure Đồ Án Môn Học: https://colab.research.google.com/drive/1ZaYXAas8tWO-Ga2AjWBWV4SkamFpn0Ml?usp=sharing
6. Trường Đại học Bách khoa Hà Nội. (2018, 11 01). *ResearchGate*. From Vietnam Stock Index Trend Prediction using Gaussian Process Regression and Autoregressive Moving Average Model: https://www.researchgate.net/publication/330743514\_Vietnam\_Stock\_Index\_Trend\_Prediction\_using\_Gaussian\_Process\_Regression\_and\_Autoregressive\_Moving\_Average\_Model
7. Trường Đại học Thương mại Hà Nội. (2021, 01 13). *tuanvanle.files.wordpress*. From ỨNG DỤNG MỘT SỐ MÔ HÌNH HỌC MÁY TRONG DỰ BÁO CHIỀU BIẾN ĐỘNG CỦA THỊ TRƯỜNG CHỨNG KHOÁN VIỆT NAM: https://tuanvanle.files.wordpress.com/2021/01/ung-dung-mot-so-mo-hinh-hoc-may-trong-du-bao-chieu-bien-dong-cua-thi-truong-chung-khoan-viet-nam-1.pdf