**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÀI BÁO CÁO CUỐI KÌ**

**Đề tài : DỰ ĐOÁN GIÁ CỔ PHIẾU BẰNG PHƯƠNG PHÁP HỌC MÁY**

*Người thực hiện*: **- MSSV:**

Lớp **:**

Khoá  **:**

# **MỤC LỤC**

[**MỤC LỤC 2**](#_gjdgxs)

[**DANH MỤC BẢNG 4**](#_30j0zll)

[**DANH MỤC HÌNH ẢNH 5**](#_1fob9te)

[**CHƯƠNG 1: TỔNG QUAN NGHIÊN CỨU 7**](#_2xcytpi)

[1.1 Lý do chọn đề tài 7](#_1ci93xb)

[1.2 Mục tiêu nghiên cứu 8](#_3whwml4)

[1.2.1 Mục tiêu chính 8](#_2bn6wsx)

[1.2.2 Mục tiêu cụ thể 8](#_qsh70q)

[1.3 Câu hỏi nghiên cứu 8](#_3as4poj)

[1.4 Ý nghĩa của nghiên cứu 9](#_uxktutxonraf)

[1.5 Đối tượng và phạm vi nghiên cứu 9](#_6qa8dbswpln6)

[1.5.1 Đối tượng nghiên cứu 9](#_mwah6jy3o0ry)

[1.5.2 Phạm vi nghiên cứu 9](#_tc43p24hro0n)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 10**](#_apxxsijntch2)

[2.1 Cơ sở lý thuyết và mô hình nghiên cứu 10](#_12n0f2gbld4n)

[2.1.1 Tổng quan về Dự báo Giá Chứng khoán 10](#_5wfkx07mikiy)

[2.1.2 Khái niệm về Học máy (Machine Learning) trong Tài chính và mô hình nghiên cứu 10](#_rwazwen099ip)

[2.2 Các chỉ báo có trong dữ liệu 10](#_k7j5lg9m1wds)

[2.2.1 SMA (Simple Moving Average - Đường trung bình động đơn giản) 10](#_9ymhefajkaum)

[2.2.2 EMA (Exponential Moving Average - Đường trung bình động hàm mũ) 11](#_pthfypd5gnw6)

[2.2.3 RSI (Relative Strength Index - Chỉ báo sức mạnh tương đối) 11](#_xrk1frtdk0pm)

[2.2.4 MACD (Moving Average Convergence Divergence) 11](#_z350hcuh0y2o)

[2.2.5 Bollinger Bands (Dải Bollinger) 12](#_t3m1f1ea8hog)

[2.2.6 ATR (Average True Range - Chỉ báo Biên độ Thực Trung bình) 12](#_pciug5wmfudr)

[2.2.7 Stochastic Oscillator (Dao động Stochastic) 12](#_bh3isnd7svez)

[2.3 Các chỉ số đánh giá 14](#_jvyr5s9hk4m)

[2.3.1. Mean Squared Error (MSE) 14](#_9zg3qeo2jpsd)

[2.3.2 Mean Absolute Error (MAE) 15](#_x1ogts4sz8g0)

[2.3.3 R² Score (Hệ số xác định) 15](#_1mu2ul4p4m2a)

[**CHƯƠNG 3 : PHƯƠNG PHÁP NGHIÊN CỨU 16**](#)

[3.1 Thu thập và mô tả dữ liệu 16](#_waz5cr2aa9ky)

[3.2 Xử lý số liệu 17](#_8i9rwe6ies61)

[3.2.1 Loại bỏ những cột không cần thiết 17](#_4zwkurvfuetx)

[3.2.2 Thêm các chỉ báo vào bộ dữ liệu 17](#_6rl6sax6wzh0)

[3.2.3 Kiểm tra các giá trị bị thiếu 18](#_oi22i68j40hq)

[3.2.4 Thông tin chung về bộ dữ liệu 19](#_jrrr30s4kkw1)

[3.2.5 Thống kê mô tả tổng quát bộ dữ liệu 20](#_4jl3jcd06itm)

[3.3 Trực quan hoá một số yếu tố có trong bộ dữ liệu 20](#_za5ayvg8c45w)

[3.4 Mô hình nghiên cứu 24](#_6moarw8a1az8)

[3.4.1 Decision Tree (Cây Quyết Định) 24](#_tulx61hh82zu)

[3.4.2 Random Forest (Rừng Ngẫu Nhiên) 25](#_bvkmdyl2lv4e)

[3.4.3 AdaBoost (Adaptive Boosting) 25](#_va7rgxn2lqfi)

[3.4.4 XGBoost (Extreme Gradient Boosting) 26](#_9l8rqq7hb2x4)

[3.4.5 Support Vector Classifier (SVC) 27](#_a4oo9hoz2ir1)

[**CHƯƠNG 4 : KẾT QUẢ NGHIÊN CỨU 28**](#_n4qtwfml22lw)

[4.1 Thiết lập mô hình Desicion Tree 28](#_afwufsld8ahe)

[4.2 Thiết lập mô hình Random Forest Regressor 31](#_p49itanpjya0)

[4.3 Thiết lập mô hình AdaBoost 35](#_scnwuof6hvr8)

[4.4 Thiết lập mô hình XGBoost (Extreme Gradient Boosting) 39](#_cj4zx39opd83)

[4.5 Thiết lập mô hình SVR 43](#_9tnvq4e9biie)

[**CHƯƠNG 5 : KẾT LUẬN NGHIÊN CỨU 48**](#_mzi1byhz906i)

[5.1 Kết luận nghiên cứu 48](#_t759h0iv9f0g)

[5.2 Hạn chế nghiên cứu 48](#_mrtdcbhbnirt)

[5.3 Đề xuất 48](#_8lotnagg3vtl)

[5.4 Hướng phát triển trong tương lai 49](#_5uyy7jniivyc)

[**TÀI LIỆU THAM KHẢO 49**](#_xynpy5e3ci2h)

# 

# 

# 

# **DANH MỤC BẢNG**

*Bảng 4.1 Kết quả mô hình DecisionTreeRegressor*

*Bảng 4.2 Kết quả mô hình RandomForestRegressor*

*Bảng 4.3 Các độ đo trong mô hình AdaBoost Regressor*

*Bảng 4.4 Kết quả đánh giá hiệu suất mô hình XGBoost Regressor*

*Bảng 4.5 Kết quả mô hình SVR*

# 

# **DANH MỤC HÌNH ẢNH**

*Hình 1.1 Bộ dữ liệu lịch sử của cổ phiếu Vietcombank (VCB) trong 30 dòng đầu tiên*

*Hình 3.2.1 Bộ dữ liệu sau khi xoá các cột không cần thiết*

*Hình 3.2.2 Bộ dữ liệu sau khi thêm các chỉ báo*

*Hình 3.3.1 Biểu đồ trực quan hoá giá mở cửa , giá đóng cửa có trong dữ liệu*

*Hình 3.3.2 Biểu đồ trực quan hoá giá cao nhất , giá thấp nhất trong dữ liệu*

*Hình 3.3.3 Biểu đồ khối lượng giao dịch trong bộ dữ liệu*

*Hình 3.3.1 Phân phối về các loại giá*

*Hình 3.3.2 Mối quan hệ giữa giá đóng cửa và các đường trung bình động*

*Hình 3.3.3 Mối quan hệ giữa giá đóng cửa và các đường trung bình động*

*Hình 4.1.1 Xử lý dữ liệu đầu vào*

*Hình 4.1.2 Xác định đặc trưng và nhãn mục tiêu*

*Hình 4.1.3 Chia tập dữ liệu*

*Hình 4.1.3 Khởi tạo và huấn luyện mô hình*

*Hình 4.1.4 Dự đoán và đánh giá mô hình*

*Hình 4.1.5 Biểu đồ trực quan hoá giá trị dự đoán và giá trị thực*

*Hình 4.2.1 Chuẩn bị dữ liệu cho mô hình Random Forest Regressor*

*Hình 4.2.2 Xử lý dữ liệu*

*Hình 4.2.3 Huấn luyện mô hình*

*Hình 4.2.4 Biểu đồ trực quan hoá giá trị dự đoán và giá trị thực của mô hình RandomForestRegressor*

*Hình 4.2.5 Biểu đồ phân phối sai số dự đoán*

*Hình 4.3.1 Chuẩn bị dữ liệu đầu vào cho mô hình AdaBoost Regressor*

*Hình 4.3.2 Phân chia tập dữ liệu*

*Hình 4.3.3 Tạo và huấn luyện mô hình AdaBoost Regressor*

*Hình 4.3.4 Kết quả dự đoán từ mô hình với 5 mẫu đầu tiên*

*Hình 4.3.5 So sánh giá trị thựuc tế và dự đoán của mô hình AdaBoost*

*Hình 4.4.1 Khởi tạo mô hình XGBoost Regressor*

*Hình 4.4.2 Huấn luyện mô hình XGBoost Regressor*

*Hình 4.4.4 Các chỉ số đánh giá mô hình*

*Hình 4.5.1 Chuẩn bị và xử lý dữ liệu*

*Hình 4.5.2 Chuẩn hoá dữ liệu*

*Hình 4.5.3 Xây dựng mô hình SVR*

*Hình 4.5.4 Dự đoán và đánh giá mô hình*

*Hình 4.5.5 So sánh giá trị thực tế và dự đoán*

*Hình 4.5.6 Biểu đồ phân phối sai lệch*

# 

# **CHƯƠNG 1: TỔNG QUAN NGHIÊN CỨU**

## **1.1 Lý do chọn đề tài**

Thị trường chứng khoán Việt Nam trong những năm gần đây đã chứng kiến sự tăng trưởng mạnh mẽ cả về quy mô và số lượng nhà đầu tư. Theo báo cáo của Ủy ban Chứng khoán Nhà nước, số lượng tài khoản nhà đầu tư tại Việt Nam tính đến năm 2021 đã vượt qua mốc 3 triệu tài khoản, với hàng triệu giao dịch được thực hiện mỗi ngày. Trong đó, Vietcombank (VCB) là một trong những cổ phiếu chủ chốt thuộc nhóm VN30, đại diện cho các doanh nghiệp hàng đầu trên thị trường chứng khoán Việt Nam. Việc dự báo chính xác giá cổ phiếu VCB không chỉ giúp các nhà đầu tư cá nhân có thêm công cụ để đưa ra quyết định đầu tư hiệu quả mà còn hỗ trợ các tổ chức tài chính trong việc quản lý danh mục đầu tư của họ.

Hiện tại, các phương pháp dự đoán truyền thống như phân tích kỹ thuật và phân tích cơ bản đều có những hạn chế nhất định. Chúng thường yêu cầu sự can thiệp của chuyên gia và không có khả năng tự động hóa, đồng thời không thể xử lý dữ liệu lớn và không tận dụng được các mô hình phức tạp để phát hiện ra các mẫu biến động trong dữ liệu. Trong bối cảnh này, việc ứng dụng các mô hình học máy (Machine Learning) trong dự báo giá cổ phiếu mang lại tiềm năng rất lớn. Học máy có thể tự động học hỏi từ dữ liệu, xử lý các mối quan hệ phi tuyến tính và phân tích dữ liệu lớn một cách hiệu quả.

**Đầu vào của mô hình**: Dữ liệu đầu vào của mô hình bao gồm các yếu tố như:Giá mở cửa, giá cao nhất, giá thấp nhất, giá đóng cửa và khối lượng giao dịch của cổ phiếu VCB trong khoảng thời gian từ 20/05/2019 đến 28/06/2019, lấy từ Yahoo Finance.Các yếu tố này sẽ được chuẩn hóa và xử lý để trở thành các đặc trưng (features) đầu vào cho mô hình học máy.

**Đầu ra của mô hình**: Mô hình dự đoán sẽ cung cấp dự đoán giá đóng cửa của cổ phiếu VCB cho các phiên giao dịch tiếp theo, giúp các nhà đầu tư có cơ sở tham khảo để ra quyết định mua, bán hoặc nắm giữ cổ phiếu.Các kết quả dự đoán sẽ được đánh giá dựa trên độ chính xác và sai số, từ đó giúp nhà đầu tư có cái nhìn khách quan và giảm thiểu rủi ro trong quyết định đầu tư.

## **1.2 Mục tiêu nghiên cứu**

### **1.2.1 Mục tiêu chính**

Xây dựng một mô hình học máy dự đoán giá đóng cửa của cổ phiếu VCB dựa trên dữ liệu lịch sử, với mục tiêu đạt được độ chính xác cao nhất có thể.

### **1.2.2 Mục tiêu cụ thể**

Tìm hiểu các mô hình học máy phù hợp cho bài toán dự đoán chuỗi thời gian (time-series forecasting) trong lĩnh vực tài chính, bao gồm các mô hình như Hồi quy tuyến tính, Mạng nơ-ron nhân tạo (Artificial Neural Networks - ANN) và LSTM (Long Short-Term Memory).

Phân tích và xử lý dữ liệu đầu vào từ Yahoo Finance để đảm bảo tính liên tục và đầy đủ của dữ liệu trước khi đưa vào mô hình.

So sánh hiệu quả của các mô hình khác nhau dựa trên các chỉ số đánh giá như Mean Absolute Error (MAE) và Root Mean Squared Error (RMSE), từ đó chọn ra mô hình dự báo phù hợp nhất.

## **1.3 Câu hỏi nghiên cứu**

Để đạt được mục tiêu nghiên cứu, đề tài sẽ tập trung trả lời các câu hỏi sau:

1. *Mô hình học máy nào phù hợp nhất cho bài toán dự báo giá cổ phiếu VCB?*

Các mô hình nào có khả năng dự báo chính xác nhất khi áp dụng cho dữ liệu lịch sử của cổ phiếu VCB?

1. *Mô hình học máy có thể cải thiện độ chính xác dự báo so với các phương pháp truyền thống không?*

So sánh hiệu quả của mô hình học máy với các phương pháp truyền thống để kiểm chứng khả năng ứng dụng của mô hình trong thực tế.

1. *Các yếu tố nào trong dữ liệu lịch sử có tác động lớn nhất đến khả năng dự đoán giá đóng cửa của cổ phiếu VCB?*

Phân tích các biến số đầu vào như giá mở cửa, giá cao nhất, giá thấp nhất và khối lượng giao dịch để xác định yếu tố nào có sức ảnh hưởng mạnh nhất đến giá đóng cửa.

## **1.4 Ý nghĩa của nghiên cứu**

**Đối với nhà đầu tư cá nhân và tổ chức:** Mô hình dự báo giá cổ phiếu sẽ giúp các nhà đầu tư có thêm công cụ hỗ trợ trong việc ra quyết định, từ đó nâng cao khả năng sinh lời và giảm thiểu rủi ro đầu tư. Việc cung cấp một dự báo giá có độ chính xác cao sẽ giúp họ xác định được xu hướng thị trường ngắn hạn của cổ phiếu VCB, từ đó tối ưu hóa lợi nhuận và có chiến lược đầu tư hiệu quả.

**Đối với học thuật và nghiên cứu khoa học:** Đề tài đóng góp vào việc nghiên cứu ứng dụng học máy trong lĩnh vực tài chính, đặc biệt là dự báo thị trường chứng khoán. Nó tạo ra một nền tảng cho các nghiên cứu sâu hơn về ứng dụng của các mô hình AI và Machine Learning trong lĩnh vực tài chính tại Việt Nam.

**Đối với thị trường chứng khoán Việt Nam:** Nếu mô hình đạt hiệu quả cao, nghiên cứu có thể được mở rộng và áp dụng cho các cổ phiếu khác ngoài VCB, góp phần vào sự phát triển bền vững của thị trường chứng khoán và tăng cường khả năng dự báo của các tổ chức tài chính.

## **1.5 Đối tượng và phạm vi nghiên cứu**

### **1.5.1 Đối tượng nghiên cứu**

Cổ phiếu VCB của ngân hàng Vietcombank (Ngân hàng TMCP Ngoại thương Việt Nam) trên thị trường chứng khoán Việt Nam. Đây là một cổ phiếu có tính thanh khoản cao, được giao dịch thường xuyên và có sự quan tâm lớn từ nhà đầu tư.

### **1.5.2 Phạm vi nghiên cứu**

**Dữ liệu đầu vào:** Sử dụng dữ liệu lịch sử của cổ phiếu VCB từ Yahoo Finance, bao gồm các yếu tố giá mở cửa, giá cao nhất, giá thấp nhất, giá đóng cửa và khối lượng giao dịch trong khoảng thời gian từ 20/05/2019 đến 28/06/2019. Đây là những biến số quan trọng, có tác động trực tiếp đến biến động giá của cổ phiếu.

**Phạm vi thời gian:** Dữ liệu thu thập từ ngày 20/05/2019 đến 28/06/2019 sẽ được dùng để huấn luyện và kiểm tra mô hình. Mặc dù phạm vi dữ liệu ngắn, nó giúp nghiên cứu có thể tập trung vào các biến động ngắn hạn, phù hợp với việc dự báo trong thị trường biến động nhanh.

**Phạm vi kỹ thuật:** Đề tài sẽ thử nghiệm các mô hình học máy khác nhau như Hồi quy tuyến tính, Mạng nơ-ron nhân tạo (ANN), và LSTM để dự đoán giá cổ phiếu. Các mô hình sẽ được đánh giá và so sánh dựa trên các chỉ số hiệu suất để tìm ra mô hình có hiệu quả tốt nhất cho bài toán dự báo ngắn hạn.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1 Cơ sở lý thuyết và mô hình nghiên cứu**

Phần này sẽ trình bày lý thuyết nền tảng và mô hình nghiên cứu được sử dụng trong đồ án, bao gồm các khái niệm chính liên quan đến dự đoán giá chứng khoán và các mô hình học máy áp dụng trong lĩnh vực này.

### **2.1.1 Tổng quan về Dự báo Giá Chứng khoán**

Dự báo giá chứng khoán là một trong những nhiệm vụ khó khăn trong tài chính do giá chứng khoán thường biến động mạnh và chịu tác động của nhiều yếu tố như kinh tế vĩ mô, tâm lý thị trường và các sự kiện bất ngờ. Mục tiêu của dự báo giá chứng khoán là cung cấp thông tin hữu ích giúp các nhà đầu tư đưa ra quyết định giao dịch hiệu quả. Với sự phát triển của công nghệ và dữ liệu lớn, các mô hình học máy ngày càng được áp dụng rộng rãi để hỗ trợ phân tích và dự báo trong lĩnh vực này.

### **2.1.2 Khái niệm về Học máy (Machine Learning) trong Tài chính và mô hình nghiên cứu**

Học máy là một nhánh của trí tuệ nhân tạo (AI), cho phép máy tính học từ dữ liệu và cải thiện hiệu suất mà không cần lập trình rõ ràng từng bước. Trong tài chính, học máy được sử dụng để phân tích các xu hướng thị trường, phân loại tín hiệu giao dịch, và dự đoán giá trị của các chỉ số chứng khoán. Các mô hình học máy được sử dụng trong nghiên cứu bao gồm: Desicion Tree, Random Forest, AdaBoost, XGBoost,Support Vector Machine (SVM)

## **2.2 Các chỉ báo có trong dữ liệu**

### **2.2.1 SMA (Simple Moving Average - Đường trung bình động đơn giản)**

SMA là chỉ báo kỹ thuật cơ bản nhất, dùng để làm mượt biến động giá và xác định xu hướng. SMA được tính bằng cách lấy trung bình cộng của giá trong một khoảng thời gian nhất định.

Công thức:

Trong đó:

* n: Số kỳ (period) muốn tính trung bình, thường dùng 10, 20, 50, 200 ngày
* P: Giá đóng cửa của mỗi kỳ
* P₁: Giá của kỳ hiện tại
* Pₙ: Giá của kỳ cũ nhất trong tính toán

Ví dụ: SMA(10) sẽ tính trung bình giá của 10 ngày gần nhất

### **2.2.2 EMA (Exponential Moving Average - Đường trung bình động hàm mũ)**

EMA là phiên bản cải tiến của SMA, đặt trọng số cao hơn cho dữ liệu gần đây hơn. Điều này giúp EMA phản ứng nhanh hơn với biến động giá gần đây.

Công thức:

Trong đó:

* **Price(t):** Giá hiện tại
* **EMA(y):** EMA của ngày hôm trước
* **k**: Hệ số làm mượt (smoothing factor)
* **n**: Số kỳ được chọn

### **2.2.3 RSI (Relative Strength Index - Chỉ báo sức mạnh tương đối)**

RSI là chỉ báo động lượng đo lường tốc độ và sự thay đổi của biến động giá. RSI dao động từ 0 đến 100, với mức trên 70 được coi là quá mua và dưới 30 là quá bán.

Công thức:

Trong đó:

* **RS (Relative Strength):** Tỷ lệ giữa trung bình tăng và trung bình giảm
* **Trung bình tăng:** Trung bình của các giá trị tăng trong n kỳ
* **Trung bình giảm**: Trung bình của các giá trị giảm trong n kỳ

### **2.2.4 MACD (Moving Average Convergence Divergence)**

MACD là chỉ báo động lượng theo xu hướng cho thấy mối quan hệ giữa hai đường trung bình động của giá. MACD giúp xác định xu hướng và điểm đảo chiều.

Công thức:

* **EMA(12):** EMA 12 kỳ của giá đóng cửa
* **EMA(26):** EMA 26 kỳ của giá đóng cửa
* **Signal Line:** Đường tín hiệu (EMA 9 kỳ của MACD Line)
* **Histogram:** Biểu đồ cột thể hiện khoảng cách giữa MACD và Signal Line

### **2.2.5 Bollinger Bands (Dải Bollinger)**

Bollinger Bands là công cụ đo lường độ biến động, bao gồm 3 đường: trung tâm (SMA) và 2 dải trên/dưới cách đều SMA một khoảng 2 độ lệch chuẩn.

Công thức:

Trong đó:

* SMA(n): Đường trung bình động đơn giản n kỳ (thường là 20)
* σ: Độ lệch chuẩn của giá trong n kỳ
* Upper Band: Dải trên
* Lower Band: Dải dưới
* Khoảng cách 2σ: Bao quát khoảng 95% biến động giá

### **2.2.6 ATR (Average True Range - Chỉ báo Biên độ Thực Trung bình)**

ATR đo lường độ biến động của thị trường mà không quan tâm đến hướng giá. Chỉ báo này đặc biệt hữu ích trong việc đánh giá rủi ro và thiết lập mức stop-loss.

Công thức:

Trong đó:

* TR (True Range): Giá trị lớn nhất trong 3 khoảng:

+ Khoảng 1: Giá cao nhất - Giá thấp nhất hiện tại

+ Khoảng 2: |Giá cao nhất - Giá đóng cửa trước|

+ Khoảng 3: |Giá thấp nhất - Giá đóng cửa trước|

* n: Số kỳ (thường là 14)
* close\_prev: Giá đóng cửa phiên trước

### **2.2.7 Stochastic Oscillator (Dao động Stochastic)**

Stochastic là chỉ báo động lượng so sánh giá đóng cửa với phạm vi giá trong một khoảng thời gian. Giúp xác định điều kiện quá mua/quá bán và điểm đảo chiều tiềm năng.

Công thức:

Trong đó:

* C: Giá đóng cửa hiện tại
* L₁₄: Giá thấp nhất trong 14 kỳ
* H₁₄: Giá cao nhất trong 14 kỳ
* %K: Đường nhanh (fast stochastic)
* %D: Đường chậm (slow stochastic)
* Dao động từ 0 đến 100

+ Trên 80: Vùng quá mua

+ Dưới 20: Vùng quá bán

**2.2.8 ADX (Average Directional Index - Chỉ báo xu hướng trung bình)**

ADX đo lường sức mạnh của xu hướng, bất kể là xu hướng tăng hay giảm. Chỉ báo này giúp xác định xem thị trường đang trong xu hướng mạnh hay yếu.

Công thức:

Trong đó:

* +DI: Chỉ báo hướng dương
* -DI: Chỉ báo hướng âm
* +DM: Chuyển động hướng dương
* -DM: Chuyển động hướng âm
* ATR: Average True Range

**2.2.9 CCI (Commodity Channel Index - Chỉ báo Kênh Hàng hóa)**

CCI đo lường sự khác biệt giữa giá hiện tại và giá trung bình lịch sử. Giúp xác định chu kỳ quá mua/quá bán và điểm đảo chiều tiềm năng.

Công thức:

Trong đó:

* TP (Typical Price): Giá điển hình
* SMA(TP): Trung bình động của giá điển hình
* MD (Mean Deviation): Độ lệch trung bình
* 0.015: Hằng số để chuẩn hóa

**2.2.10 Volume Moving Average (Trung bình động Khối lượng)**

Chỉ báo này giúp phân tích xu hướng khối lượng giao dịch, xác nhận sức mạnh của xu hướng giá.

Công thức:

Trong đó:

## **2.3 Các chỉ số đánh giá**

Để đánh giá hiệu quả của các mô hình dự báo, một số chỉ số đánh giá phổ biến được sử dụng trong đề tài bao gồm:

### **2.3.1. Mean Squared Error (MSE)**

MSE (Sai số trung bình bình phương) đo lường độ chênh lệch trung bình giữa giá trị dự đoán và giá trị thực tế. Độ đo này nhấn mạnh vào các sai số lớn do sai số được bình phương. Một giá trị MSE nhỏ hơn cho thấy mô hình có độ chính xác cao hơn.Công thức**:**

Trong đó:

* n: Số lượng mẫu trong tập kiểm tra.
* ​: Giá trị thực tế của mẫu thứ i.
* : Giá trị dự đoán của mẫu thứ i.

### **2.3.2 Mean Absolute Error (MAE)**

MAE (Sai số trung bình tuyệt đối) là giá trị trung bình của các sai số tuyệt đối giữa giá trị dự đoán và giá trị thực tế. Không giống như MSE, MAE không nhấn mạnh vào các sai số lớn mà chỉ đơn thuần đo lường độ chênh lệch tuyệt đối. Công thức:

Trong đó:

* : Số lượng mẫu trong tập kiểm tra.
* : Giá trị thực tế của mẫu thứ i.
* Giá trị dự đoán của mẫu thứ i.

### **2.3.3 R² Score (Hệ số xác định)**

R² (Hệ số xác định) đo lường mức độ giải thích của mô hình đối với phương sai của dữ liệu thực tế. Giá trị R² nằm trong khoảng từ -∞ đến 1:

* R²=1: Mô hình dự đoán hoàn hảo.
* R²=0: Mô hình không giải thích được gì thêm ngoài giá trị trung bình của dữ liệu.
* R²<0: Mô hình dự đoán kém hơn so với việc chỉ dùng giá trị trung bình.

Công thức:

Trong đó:

* : Tổng bình phương sai số của mô hình.
* : Tổng bình phương sai số so với trung bình của dữ liệu.
* : Giá trị trung bình của tập dữ liệu thực tế.

# **CHƯƠNG 3 : PHƯƠNG PHÁP NGHIÊN CỨU**

## **3.1 Thu thập và mô tả dữ liệu**

Để xây dựng một mô hình dự đoán giá cổ phiếu Vietcombank (VCB), dữ liệu lịch sử của cổ phiếu này đã được thu thập từ nguồn Yahoo Finance, bao gồm các chỉ số:

* **Giá mở cửa (Open):** Giá cổ phiếu tại thời điểm mở cửa thị trường**.**
* **Giá cao nhất (High):** Giá cao nhất đạt được trong phiên giao dịch.
* **Giá thấp nhất (Low):** Giá thấp nhất đạt được trong phiên giao dịch.
* **Giá đóng cửa (Close):** Giá cổ phiếu tại thời điểm đóng cửa thị trường.
* **Khối lượng giao dịch (Volume):** Tổng số lượng cổ phiếu giao dịch trong phiên.

Dữ liệu được thu thập từ ngày 20/05/2019 đến 28/06/2019, bao gồm 30 phiên giao dịch, mỗi phiên cung cấp thông tin chi tiết về biến động giá và khối lượng giao dịch của cổ phiếu VCB. Những chỉ số này sẽ được sử dụng làm các biến đầu vào (features) trong mô hình học máy để dự đoán giá đóng cửa (Close) của cổ phiếu cho các phiên tiếp theo.

A screenshot of a table

Description automatically generated

*Hình 3.1 Bộ dữ liệu lịch sử của cổ phiếu Vietcombank (VCB) trong 30 dòng đầu tiên*

## **3.2 Xử lý số liệu**

### **3.2.1 Loại bỏ những cột không cần thiết**

Dữ liệu ban đầu bao gồm các cột như *Unnamed: 0* và *Ticker*, trong đó *Unnamed: 0* là cột chỉ mục tự động được tạo ra khi lưu trữ dữ liệu, không cung cấp thông tin dự đoán cho mô hình. Tương tự, cột *Ticker* chứa mã chứng khoán "VCB", nhưng vì toàn bộ dữ liệu chỉ liên quan đến một mã duy nhất nên cột này cũng không mang lại thông tin hữu ích cho việc dự đoán. Do đó, trong quá trình xử lý dữ liệu, chúng em đã loại bỏ các cột *Unnamed: 0* và *Ticker* để giữ lại các cột có ý nghĩa hơn cho mô hình, bao gồm *time, open, high, low, close, và volume*. Việc loại bỏ các cột không cần thiết giúp mô hình tập trung vào các thông tin quan trọng, tăng cường độ chính xác và hiệu quả của quá trình huấn luyện, đồng thời giảm thiểu sự phức tạp không cần thiết trong dữ liệu.

**A screenshot of a computer

Description automatically generated**

*Hình 3.2.1 Bộ dữ liệu sau khi xoá các cột không cần thiết*

### **3.2.2 Thêm các chỉ báo vào bộ dữ liệu**

Để tăng cường chất lượng dữ liệu cho mô hình dự đoán, chúng em đã bổ sung các chỉ báo kỹ thuật vào tập dữ liệu. Các chỉ báo này bao gồm **SMA (Simple Moving Average)** - trung bình động đơn giản, **EMA (Exponential Moving Average)** - trung bình động hàm mũ, **RSI (Relative Strength Index)** - chỉ số sức mạnh tương đối, **MACD (Moving Average Convergence Divergence)** - chỉ báo phân kỳ hội tụ trung bình động, và các dải **Bollinger Bands.** Ngoài ra, chúng em cũng thêm **ATR (Average True Range)** - chỉ báo biên độ dao động trung bình, Stochastic Oscillator - chỉ báo ngẫu nhiên, **ADX (Average Directional Index)** - chỉ báo xu hướng trung bình, **Volume SMA** - trung bình động khối lượng, và **CCI (Commodity Channel Index)** - chỉ báo kênh hàng hóa. Các chỉ báo này giúp mô hình có thêm thông tin về xu hướng giá, biến động, sức mạnh của xu hướng và động lượng thị trường, từ đó cải thiện khả năng dự đoán của mô hình. Việc thêm các chỉ báo kỹ thuật được thực hiện bằng cách tính toán dựa trên các cột giá trị close, open, high, low, và volume, sau đó bổ sung các chỉ báo tương ứng vào tập dữ liệu để cung cấp các đặc trưng chi tiết hơn cho mô hình

A screenshot of a computer

Description automatically generated

*Hình 3.2.2 Bộ dữ liệu sau khi thêm các chỉ báo*

### **3.2.3 Kiểm tra các giá trị bị thiếu**

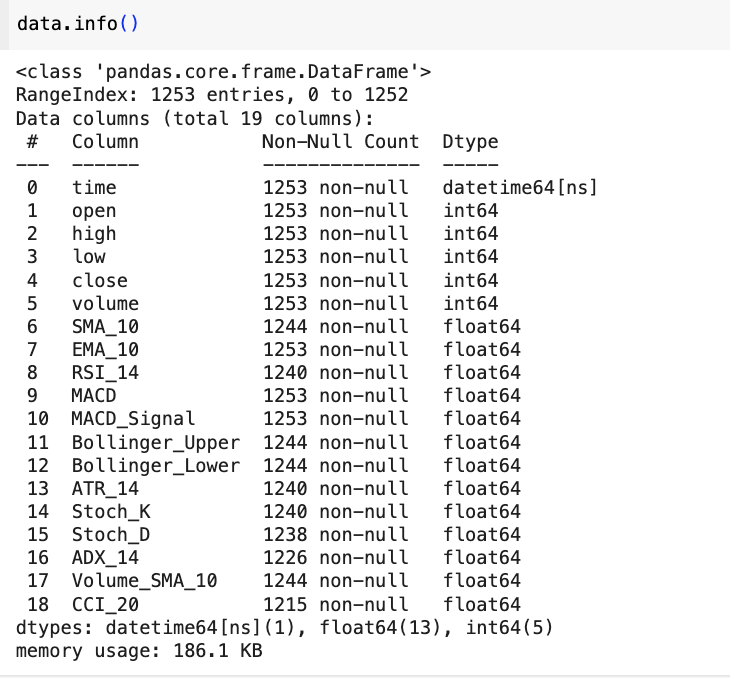
**A screenshot of a computer

Description automatically generated**

*Hình 3.2.3 Các giá trị bị thiếu trong bộ dữ liệu*

Các cột bị trống dữ liệu trong bảng biểu trên, chẳng hạn như các cột SMA\_10, RSI\_14, MACD, Bollinger\_Upper, ATR\_14, và các cột chỉ báo kỹ thuật khác, chủ yếu là do các chỉ báo này cần một khoảng thời gian nhất định để tính toán. Ví dụ, chỉ báo SMA\_10 (Simple Moving Average) yêu cầu có ít nhất 10 giá trị dữ liệu để tính toán trung bình động, vì vậy trong 10 dòng đầu tiên của dữ liệu, cột này sẽ không có giá trị vì chưa đủ dữ liệu để tính toán. Tương tự, các chỉ báo như RSI\_14, MACD, và Bollinger Bands cũng yêu cầu một số lượng dữ liệu lịch sử để tính toán, dẫn đến việc dữ liệu cho các chỉ báo này sẽ bị thiếu trong những dòng đầu tiên. Khi số lượng dữ liệu tăng lên, các cột này sẽ dần có giá trị khi có đủ thông tin để tính toán các chỉ số kỹ thuật đó

### **3.2.4 Thông tin chung về bộ dữ liệu**



*Hình 3.2.4 Thông tin về bộ dữ liệu*

Dữ liệu của bộ dữ liệu có tổng cộng 1253 mẫu với 19 cột, trong đó có các cột chính như time, open, high, low, close, và volume. Cột time chứa dữ liệu kiểu datetime64, trong khi các cột khác đều có kiểu dữ liệu số học (int64 hoặc float64). Các cột SMA\_10, RSI\_14, Bollinger\_Upper, Bollinger\_Lower, ATR\_14, Stoch\_K, Stoch\_D, ADX\_14, Volume\_SMA\_10, và CCI\_20 là các chỉ số kỹ thuật được tính toán từ các giá trị giá cổ phiếu. Tuy nhiên, một số cột có giá trị thiếu (missing values): SMA\_10, RSI\_14, Bollinger\_Upper, Bollinger\_Lower, ATR\_14, Stoch\_K, Stoch\_D, ADX\_14, Volume\_SMA\_10, và CCI\_20 có một vài giá trị thiếu. Số lượng các giá trị thiếu là không đồng đều, dao động từ 1 đến 38 giá trị bị thiếu trong các cột này

### 

### **3.2.5 Thống kê mô tả tổng quát bộ dữ liệu**

A table of numbers and a few digits

Description automatically generated with medium confidence

*Hình 3.2.5 Các thống kê cơ bản về bộ dữ liệu*

Các thông số thống kê này bao gồm số lượng quan sát (count), giá trị trung bình (mean), giá trị tối thiểu (min), giá trị tối đa (max), các phân vị (25%, 50%, 75%), và độ lệch chuẩn (std). Cụ thể, các cột như open, high, low, và close đều có số lượng quan sát là 1253, với các giá trị trung bình lần lượt là 66974.54, 67729.00, 66229.07 và 67016.93, trong khi các giá trị tối thiểu và tối đa của chúng dao động trong khoảng từ 37210 đến 98500. Các chỉ số kỹ thuật như SMA\_10, EMA\_10, RSI\_14, MACD và các chỉ số khác cũng được tính toán, với các giá trị trung bình và độ lệch chuẩn phản ánh tính biến động của thị trường

## **3.3 Trực quan hoá một số yếu tố có trong bộ dữ liệu**

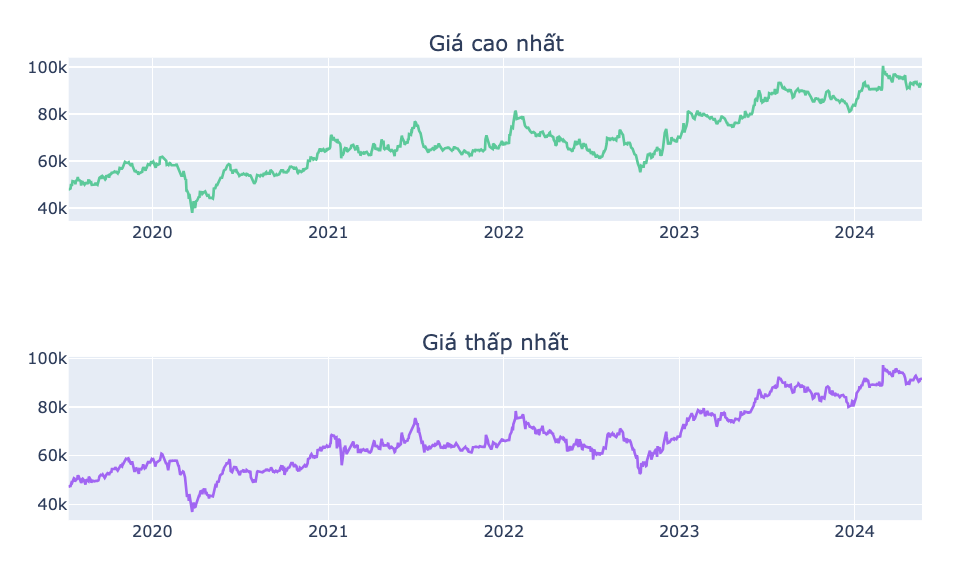
**3.3.1 Giá mở cửa và giá đóng cửa**

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

Description automatically generated

*Hình 3.3.1 Biểu đồ trực quan hoá giá mở cửa , giá đóng cửa có trong dữ liệu*

**3.3.2 Giá cao nhất , giá thấp nhất**



*Hình 3.3.2 Biểu đồ trực quan hoá giá cao nhất , giá thấp nhất trong dữ liệu*

**3.3.3 Khối lượng giao dịch**

**A graph showing the growth of the company's data

Description automatically generated with medium confidence**

*Hình 3.3.3 Biểu đồ khối lượng giao dịch trong bộ dữ liệu*

**3.3.4 Phân phối về giá mở cửa , đóng cửa , thấp nhất, cao nhất**

A graph of different colored lines

Description automatically generated with medium confidence

*Hình 3.3.1 Phân phối về các loại giá*

**3.3.5 Mối quan hệ giữa giá đóng cửa và các đường trung bình động**

A graph of a graph with a variety of blue lines

Description automatically generated with medium confidence

*Hình 3.3.2 Mối quan hệ giữa giá đóng cửa và các đường trung bình động*

**3.3.6 Mối quan hệ giữa giá đóng cửa và các đường trung bình động**

A graph with red and blue lines

Description automatically generated

*Hình 3.3.3 Mối quan hệ giữa giá đóng cửa và các đường trung bình động*

## **3.4 Mô hình nghiên cứu**

### **3.4.1 Decision Tree (Cây Quyết Định)**

Decision Tree là một trong những thuật toán phổ biến trong học máy, đặc biệt trong các bài toán phân loại và hồi quy. Cây quyết định xây dựng mô hình bằng cách chia nhỏ không gian dữ liệu dựa trên các đặc trưng của dữ liệu. Công thức của cây quyết định không phải là một công thức toán học duy nhất, nhưng quá trình xây dựng cây có thể được mô tả qua một số bước và tiêu chí đánh giá. Hai tiêu chí phổ biến nhất là **Gini Impurity** và **Entropy**.

Entropy là một tiêu chí khác để đánh giá mức độ không chắc chắn (hỗn loạn) trong một nút. Entropy càng thấp, sự phân chia càng tốt vì tập dữ liệu trong nút càng đồng nhất.Công thức tính Entropy cho một nút:

Trong đó:

* D là tập dữ liệu tại nút.
* k là số lớp trong bài toán phân loại.
* là tỷ lệ mẫu của lớp trong tập .

Gini Impurity đo lường mức độ "không thuần khiết" của một tập hợp các mẫu. Một nút có Gini Impurity thấp có nghĩa là nó chủ yếu chứa các mẫu thuộc cùng một lớp. Công thức tính Gini Impurity cho một nút:

Trong đó:

* là tập dữ liệu tại nút.
* là số lớp trong bài toán phân loại.
* là tỷ lệ mẫu của lớp iii trong tập .

### **3.4.2 Random Forest (Rừng Ngẫu Nhiên)**

Random Forest là một tập hợp của nhiều Decision Tree hoạt động độc lập, mỗi cây được huấn luyện trên một tập con ngẫu nhiên của dữ liệu. Kết quả cuối cùng của mô hình là giá trị trung bình (hoặc kết quả đa số phiếu) từ các cây quyết định. Random Forest khắc phục được vấn đề overfitting của Decision Tree bằng cách tổng hợp kết quả từ nhiều cây độc lập, nhờ đó giúp mô hình tổng quát hóa tốt hơn trên dữ liệu chưa từng thấy.

Công thức tính dự đoán của Random Forest cho một mẫu là:

trong đó là số lượng cây trong rừng, là kết quả dự đoán của cây thứ , và là giá trị trung bình của các kết quả từ tất cả các cây. Random Forest sử dụng phương pháp *Bagging* (Bootstrap Aggregating) để tạo ra các mẫu ngẫu nhiên từ dữ liệu gốc, từ đó giảm thiểu overfitting. Phương pháp này giúp cải thiện độ ổn định và độ chính xác của mô hình so với các cây quyết định đơn lẻ.

### **3.4.3 AdaBoost (Adaptive Boosting)**

AdaBoost là một thuật toán thuộc nhóm Boosting, được thiết kế nhằm cải thiện độ chính xác của mô hình bằng cách tập trung vào các mẫu khó dự đoán. AdaBoost sử dụng các mô hình yếu (weak learners), thường là các cây quyết định đơn giản, và huấn luyện chúng tuần tự. Trong mỗi lần huấn luyện, các mẫu dữ liệu mà mô hình trước đó dự đoán sai sẽ được tăng trọng số, nhằm "chú ý" hơn vào các mẫu khó. Điều này tạo điều kiện để mô hình tiếp theo tập trung vào các điểm dữ liệu dễ sai lệch, qua đó cải thiện độ chính xác tổng thể của hệ thống.

Trọng số của các mẫu dữ liệu sau mỗi lần huấn luyện được cập nhật theo công thức:

trong đó :

* wi(t+1): Trọng số mới của mẫu i
* wi(t): Trọng số hiện tại của mẫu i
* αt: Hệ số học của mô hình thứ t
* I(yi ≠ ft(xi)): Hàm chỉ thị, bằng 1 nếu dự đoán sai, bằng 0 nếu dự đoán đúngKết quả cuối cùng của AdaBoost là tổng hợp có trọng số từ các mô hình yếu:

AdaBoost có ưu điểm là tăng độ chính xác của các mô hình yếu và đặc biệt hữu ích trong các bài toán mà dữ liệu không đồng nhất.

### **3.4.4 XGBoost (Extreme Gradient Boosting)**

XGBoost là một phiên bản nâng cao của Boosting, tối ưu hóa cả về hiệu suất lẫn độ chính xác. XGBoost sử dụng Gradient Boosting, trong đó các mô hình yếu mới được thêm vào nhằm giảm thiểu sai số còn lại từ mô hình trước đó. Điều này giúp cho mô hình tối ưu hóa được sai số qua từng lần lặp và trở nên chính xác hơn.

Trong XGBoost, hàm mất mát phổ biến thường là Mean Squared Error (MSE), được tính bằng:

XGBoost hoạt động bằng cách liên tục bổ sung các mô hình để giảm thiểu sai số của các mô hình trước đó, với công thức dự đoán cập nhật tại mỗi bước lặp như sau:

trong đó :

* Giá trị dự đoán sau lần lặp thứ .
* : Giá trị dự đoán tại lần lặp thứ t.
* :Hệ số học tại bước t.
* :Mô hình yếu được bổ sung tại bước t.

XGBoost nổi bật nhờ khả năng xử lý dữ liệu lớn, tự động xử lý giá trị thiếu và hỗ trợ tối ưu hóa hiệu suất thông qua các kỹ thuật như *shrinkage* (giảm hệ số học sau mỗi lần cập nhật) và *column subsampling* (chọn ngẫu nhiên một số lượng cột khi xây dựng mô hình).

XGBoost có ưu điểm là hiệu quả cao, có khả năng chính xác cao trên các bộ dữ liệu phức tạp, và đã được chứng minh là đạt hiệu suất tốt trong các bài toán dự đoán giá trị liên tục như dự đoán giá cổ phiếu.

### **3.4.5 Support Vector Classifier (SVC)**

SVC (Support Vector Classifier) là một thuật toán phân loại phổ biến trong học máy, đặc biệt hiệu quả khi xử lý các bài toán phân loại có số lượng lớn đặc trưng. Mặc dù SVC thường được áp dụng trong phân loại, nó cũng có thể được mở rộng để áp dụng cho hồi quy với phương pháp **Support Vector Regression (SVR)**, có khả năng dự đoán chính xác các giá trị liên tục.

SVC tìm kiếm một siêu phẳng phân cách tối ưu giữa các lớp dữ liệu bằng cách tối đa hóa khoảng cách giữa siêu phẳng và các điểm dữ liệu gần nhất (các điểm này được gọi là vector hỗ trợ). Trong trường hợp dữ liệu không thể phân chia tuyến tính, SVC sử dụng hàm *kernel* để biến đổi dữ liệu sang không gian có chiều cao hơn, nơi các lớp có thể được phân tách tuyến tính.

Công thức của siêu phẳng phân cách trong không gian đặc trưng là:

trong đó

* w: Vector trọng số.
* x: Dữ liệu đầu vào.
* b: Sai lệch (bias).

. Trong trường hợp sử dụng kernel, chúng ta có:

Trong đó:

* : Hệ số Lagrange.
* : Nhãn của điểm dữ liệu thứ i.
* : Hàm kernel áp dụng cho dữ liệu xix\_ixi​ và xxx.
* b : Sai lệch (bias).

. Các kernel phổ biến gồm *linear*, *polynomial*, *radial basis function (RBF)* và *sigmoid*, giúp biến đổi dữ liệu để đạt được phân tách tốt hơn.

# **CHƯƠNG 4 : KẾT QUẢ NGHIÊN CỨU**

## **4.1 Thiết lập mô hình Desicion Tree**

**4.1.1 Quy trình thực hiện**

Trong nghiên cứu này, mô hình Decision Tree được thiết lập để dự đoán sự tăng hoặc giảm giá của tài sản dựa trên các đặc trưng kỹ thuật. Quy trình thực hiện được trình bày như sau:

**Bước 1: Chuẩn bị dữ liệu đầu vào:**

Dữ liệu được sao chép từ tập dữ liệu ban đầu nhằm đảm bảo tính toàn vẹn và sau đó được làm sạch bằng cách loại bỏ các giá trị bị thiếu thông qua phương pháp dropna(). Biến mục tiêu (Target) được xây dựng dựa trên giá trị giá đóng cửa (close) của ngày tiếp theo bằng cách dịch chuyển cột close xuống một hàng với phương pháp shift(-1). Các đặc trưng đầu vào (features) được lựa chọn bao gồm giá mở cửa (open), giá cao nhất (high), giá thấp nhất (low), giá đóng cửa (close), khối lượng giao dịch (volume), và các chỉ số kỹ thuật như SMA\_10, EMA\_10, RSI\_14, MACD, MACD\_Signal, dải Bollinger (Bollinger\_Upper, Bollinger\_Lower), và các chỉ số khác.

A computer code with red text

Description automatically generated

*Hình 4.1.1 Xử lý dữ liệu đầu vào*

**Bước 2: Chia tập dữ liệu thành tập huấn luyện và kiểm tra**

Dữ liệu được chia thành hai phần:

* **Tập huấn luyện (80%)**: Dùng để huấn luyện mô hình.
* **Tập kiểm tra (20%)**: Dùng để đánh giá mô hình.

Quá trình chia dữ liệu được thực hiện bằng hàm train\_test\_split() với tham số random\_state=42 nhằm đảm bảo tính tái lập kết quả.



*Hình 4.1.2 Chia tập dữ liệu*

**Bước 3: Khởi tạo và huấn luyện mô hình Decision Tree**

Mô hình **Decision Tree** được chọn để dự đoán giá đóng cửa ngày tiếp theo. Mô hình được khởi tạo với random\_state=42 nhằm đảm bảo tính tái lập, và sau đó được huấn luyện trên tập dữ liệu huấn luyện bằng phương pháp fit():

A black text with black text

Description automatically generated

*Hình 4.1.3 Khởi tạo và huấn luyện mô hình*

**Bước 4: Dự đoán và đánh giá mô hình**

Mô hình dự đoán xu hướng giá cho tập kiểm tra bằng cách sử dụng hàm predict():

A screenshot of a computer program

Description automatically generated

*Hình 4.1.4 Dự đoán và đánh giá mô hình*

Hiệu quả của mô hình được đánh giá trên tập dữ liệu kiểm tra thông qua các chỉ số:

* **Mean Squared Error (MSE):** Sai số bình phương trung bình, phản ánh độ lệch tổng quát của các dự đoán.
* **Mean Absolute Error (MAE):** Sai số trung bình tuyệt đối, dễ diễn giải và trực quan hơn.
* **R² Score:** Hệ số xác định, đo lường mức độ phù hợp của mô hình với dữ liệu.

**4.1.2 Kết quả của mô hình**

Sau khi hoàn thành quá trình huấn luyện và kiểm tra mô hình Decision Tree, các kết quả đánh giá hiệu suất được trình bày ở bảng sau (kết quả được làm tròn hai chữ số sau dấu phẩy):

| **Model** | **Mean Squared Error (MSE)** | **Mean Absolute Error (MAE)** | **R² Score** |
| --- | --- | --- | --- |
| DecisionTreeRegressor | 2221031.5021 | 1048.0453 | 0.9866 |

*Bảng 4.1 Kết quả mô hình DecisionTreeRegressor*

Kết quả từ mô hình DecisionTreeRegressor cho thấy rằng mô hình này hoạt động rất tốt trong việc dự đoán. Cụ thể, giá trị Mean Squared Error (MSE) là 2,221,031.5, cho thấy độ lệch trung bình bình phương giữa giá trị dự đoán và giá trị thực tế khá thấp, mặc dù con số này không phải là nhỏ. Mean Absolute Error (MAE) ở mức 1,048.04 cho thấy độ sai lệch tuyệt đối trung bình giữa giá trị dự đoán và giá trị thực tế không quá cao, điều này cũng phản ánh khả năng dự đoán khá chính xác của mô hình. Đặc biệt, R² score đạt 0.9866, chứng tỏ mô hình có khả năng giải thích 98.66% phương sai của dữ liệu, cho thấy sự phù hợp cao giữa các dự đoán và dữ liệu thực tế.

Trực quan hoá kết quả thu được :

A graph with red and blue dots

Description automatically generated

*Hình 4.1.5 Biểu đồ trực quan hoá giá trị dự đoán và giá trị thực*

A graph with a line

Description automatically generated

*Hình 4.1.6 Phân phối sai số dự đoán*

## **4.2 Thiết lập mô hình Random Forest Regressor**

**4.2.1 Quy trình thực hiện**

Trong quá trình nghiên cứu dự đoán giá cổ phiếu, mô hình Random Forest Regressor được lựa chọn do khả năng học và tổng hợp thông tin từ nhiều cây quyết định, giúp tăng cường độ chính xác và giảm thiểu nguy cơ overfitting. Các bước thực hiện triển khai mô hình Random Forest Regressor bao gồm:

**Bước 1: Chuẩn bị dữ liệu**

Tạo một bản sao của bộ dữ liệu ban đầu và loại bỏ các giá trị khuyết (NaN) để đảm bảo dữ liệu sạch sẽ và không bị thiếu.

Thêm một cột mới, "Target," bằng cách dịch chuyển cột "close" xuống một hàng (shift(-1)). Cột "Target" này đại diện cho giá đóng cửa của ngày tiếp theo và là biến mục tiêu trong bài toán dự đoán.

Xác định các đặc trưng (features) sẽ được sử dụng để huấn luyện mô hình, bao gồm các chỉ báo kỹ thuật phổ biến như SMA (Simple Moving Average), EMA (Exponential Moving Average), RSI (Relative Strength Index), MACD (Moving Average Convergence Divergence), dải Bollinger, ATR (Average True Range), Stochastic Oscillator, ADX (Average Directional Index), CCI (Commodity Channel Index) và các chỉ báo về khối lượng.

A computer code with red text

Description automatically generated

*Hình 4.2.1 Chuẩn bị dữ liệu cho mô hình Random Forest Regressor*

**Bước 2 :** **Xử lý dữ liệu**

Để tránh các giá trị NaN phát sinh từ việc dịch chuyển cột "Target", loại bỏ các hàng cuối cùng của X và y sau khi dịch chuyển.

Phân chia dữ liệu thành tập huấn luyện (train) và tập kiểm thử (test) với tỷ lệ 80% cho tập huấn luyện và 20% cho tập kiểm thử. Điều này giúp đảm bảo rằng mô hình được huấn luyện và kiểm tra trên các tập dữ liệu riêng biệt, từ đó giảm nguy cơ overfitting.

A computer code with text

Description automatically generated with medium confidence

*Hình 4.2.2 Xử lý dữ liệu*

**Bước 3:** **Huấn luyện mô hình**

Khởi tạo mô hình RandomForestRegressor với n\_estimators=100, tức là sử dụng 100 cây quyết định trong rừng ngẫu nhiên. Số lượng cây được chọn dựa trên tính cân bằng giữa độ chính xác và thời gian tính toán của mô hình.

Tiến hành huấn luyện mô hình trên tập huấn luyện bằng cách sử dụng phương pháp fit của mô hình, giúp Random Forest Regressor xây dựng các cây quyết định dựa trên các đặc trưng và biến mục tiêu đã chọn.



*Hình 4.2.3 Huấn luyện mô hình*

**Bước 4:** **Dự đoán**

Sử dụng mô hình đã huấn luyện để dự đoán giá trị biến mục tiêu trên tập kiểm thử. Dự đoán này được lưu trữ trong biến y\_pred, đại diện cho giá đóng cửa dự đoán của cổ phiếu trong ngày tiếp theo.

Tạo một DataFrame kết quả để so sánh giữa giá trị thực tế (y\_test) và giá trị dự đoán (y\_pred) nhằm đánh giá hiệu suất của mô hình.

**Bước 5: Đánh giá mô hình**

Sử dụng các chỉ số đánh giá độ chính xác của mô hình, bao gồm:

* **Mean Squared Error (MSE):** Trung bình bình phương sai số giữa giá trị thực tế và giá trị dự đoán, giúp đánh giá mức độ sai lệch tổng thể của mô hình.
* **Mean Absolute Error (MAE):** Sai số tuyệt đối trung bình, đánh giá mức độ sai lệch trung bình của mô hình.
* **Root Mean Squared Error (RMSE):** Căn bậc hai của MSE, giúp dễ dàng so sánh với giá trị thực tế nhờ vào việc trả về đơn vị tương tự như biến mục tiêu.
* **R² Score:** Hệ số xác định, đo lường mức độ mô hình giải thích được biến động của biến mục tiêu. Giá trị gần 1 cho thấy mô hình có khả năng dự đoán tốt.

In kết quả đánh giá để có cái nhìn tổng quan về hiệu suất của mô hình Random Forest Regressor trong dự đoán giá cổ phiếu.

**4.2.2 Kết quả của mô hình**

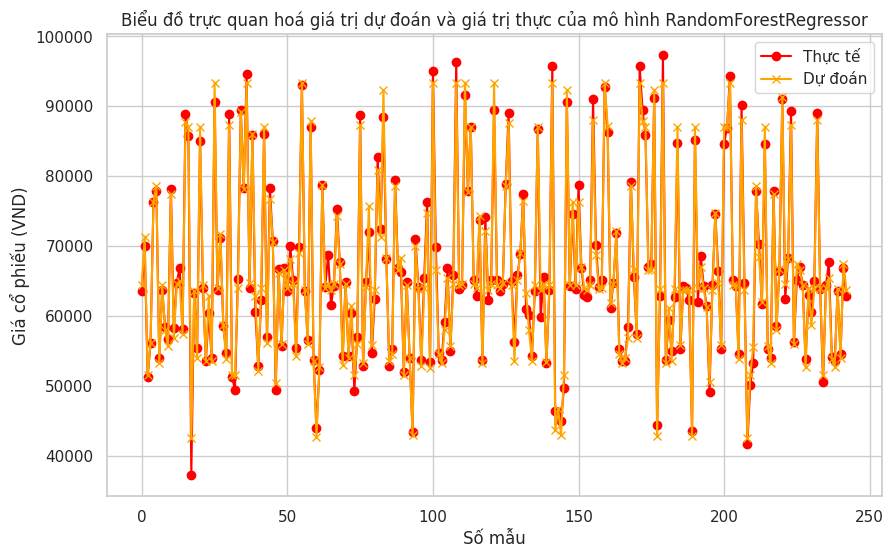
Sau khi hoàn thành quá trình huấn luyện và kiểm tra mô hình RandomForestRegressor, các kết quả đánh giá hiệu suất được trình bày ở bảng sau (kết quả được làm tròn hai chữ số sau dấu phẩy):

| **Model** | **Mean Squared Error (MSE)** | **Mean Absolute Error (MAE)** | **R² Score** |
| --- | --- | --- | --- |
| RandomForestRegressor | 2221031.5021 | 1048.0453 | 0.9866 |

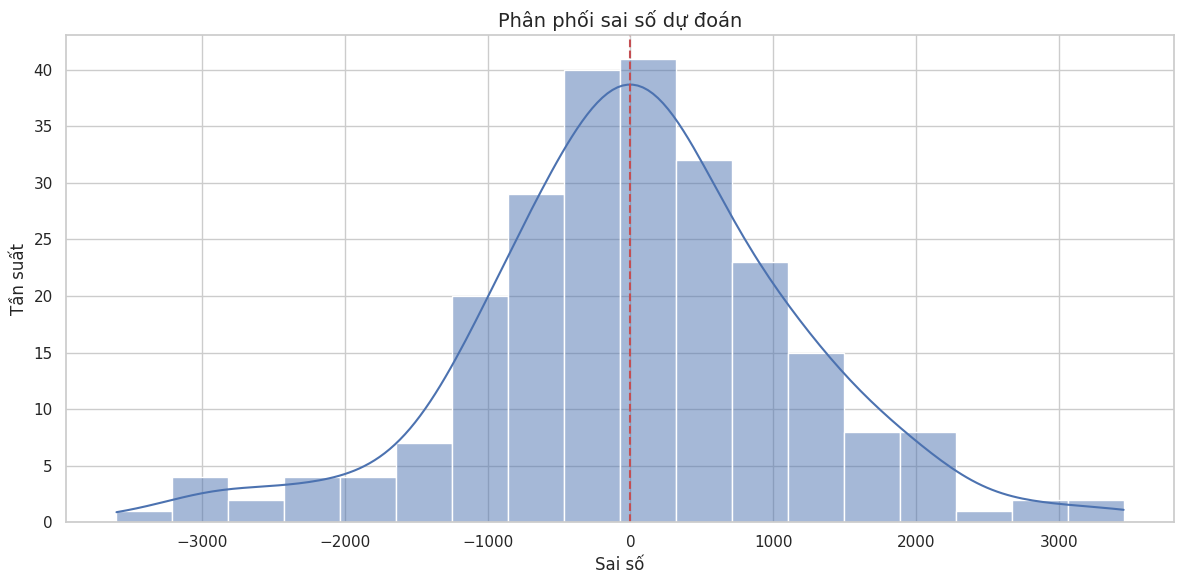
*Bảng 4.2 Kết quả mô hình RandomForestRegressor*

Kết quả từ mô hình RandomForestRegressor cho thấy hiệu suất dự đoán cũng rất tốt. Với giá trị Mean Squared Error (MSE) là 2,221,031.5, mô hình này có độ lệch trung bình bình phương tương tự như DecisionTreeRegressor, cho thấy sai số dự đoán ở mức chấp nhận được. Mean Absolute Error (MAE) là 1,048.04, chỉ ra rằng sai lệch tuyệt đối trung bình giữa các dự đoán và giá trị thực tế không lớn, phản ánh khả năng dự đoán chính xác của mô hình. Đặc biệt, R² score đạt 0.9866, giống như trong trường hợp của DecisionTreeRegressor, cho thấy mô hình RandomForestRegressor có khả năng giải thích tới 98.66% phương sai của dữ liệu, chỉ ra sự phù hợp rất cao với dữ liệu thực tế.

Trực quan hoá kết quả thu được :



*Hình 4.2.4 Biểu đồ trực quan hoá giá trị dự đoán và giá trị thực của mô hình RandomForestRegressor*



*Hình 4.2.5 Biểu đồ phân phối sai số dự đoán*

## **4.3 Thiết lập mô hình AdaBoost**

**4.3.1 Quy trình thực hiện**

**Bước 1. Chuẩn bị dữ liệu**

Đầu tiên, dữ liệu được thu thập và xử lý để sẵn sàng cho việc huấn luyện mô hình. Các đặc trưng được chọn để dự đoán giá trị mục tiêu bao gồm các chỉ báo tài chính và kỹ thuật như giá mở cửa, giá cao nhất, giá thấp nhất, giá đóng cửa, khối lượng giao dịch, các chỉ số SMA (Simple Moving Average), EMA (Exponential Moving Average), RSI (Relative Strength Index), MACD (Moving Average Convergence Divergence), Bollinger Bands, ATR (Average True Range), Stochastic Oscillator, ADX (Average Directional Index), CCI (Commodity Channel Index), và các chỉ báo liên quan đến khối lượng giao dịch.

Sau khi xác định các đặc trưng cần thiết, dữ liệu bị thiếu (NaN) sẽ được loại bỏ bằng cách sử dụng hàm dropna(), đảm bảo rằng chỉ những mẫu dữ liệu đầy đủ mới được sử dụng cho quá trình huấn luyện mô hình.

A screenshot of a computer code

Description automatically generated

*Hình 4.3.1 Chuẩn bị dữ liệu đầu vào cho mô hình AdaBoost Regressor*

**Bước 2. Xây dựng các biến đặc trưng (X) và nhãn mục tiêu (y)**

Sau khi xử lý dữ liệu, các biến đặc trưng X được xác định bằng cách chọn ra các cột cần thiết từ bộ dữ liệu, bao gồm tất cả các chỉ báo tài chính đã được nêu. Biến nhãn mục tiêu y là cột Target, là giá trị cần được dự đoán từ các đặc trưng.

**Bước 3. Chia dữ liệu thành tập huấn luyện và tập kiểm tra**

Tiếp theo, dữ liệu được chia thành hai phần: tập huấn luyện (training set) và tập kiểm tra (test set). Tập huấn luyện chiếm 80% dữ liệu, trong khi tập kiểm tra chiếm 20%. Việc chia dữ liệu này giúp mô hình học từ phần lớn dữ liệu trong khi vẫn có thể đánh giá hiệu quả của mô hình trên tập dữ liệu chưa được thấy trước đó. Quy trình này được thực hiện thông qua hàm train\_test\_split của thư viện sklearn, với một giá trị random\_state cố định là 42 để đảm bảo tính tái lập cho việc chia tập dữ liệu.

A black text on a white background

Description automatically generated

*Hình 4.3.2 Phân chia tập dữ liệu*

**Bước 4. Tạo và huấn luyện mô hình AdaBoost Regressor**

Để xây dựng mô hình dự đoán, thuật toán AdaBoost Regressor được lựa chọn. AdaBoost là một phương pháp học máy thuộc nhóm boosting, giúp tăng cường độ chính xác của mô hình bằng cách trọng số các mẫu dữ liệu dự đoán sai trong mỗi vòng huấn luyện. Mô hình AdaBoost được cấu hình với 100 ước lượng (n\_estimators=100) và tốc độ học (learning\_rate) là 0.1.

A screenshot of a computer code

Description automatically generated

*Hình 4.3.3 Tạo và huấn luyện mô hình AdaBoost Regressor*

Mô hình sau đó được huấn luyện bằng tập huấn luyện X\_train và y\_train thông qua phương thức fit. Quá trình huấn luyện này cho phép mô hình học từ các đặc trưng của dữ liệu và điều chỉnh các tham số để tối ưu hóa khả năng dự đoán.

**5. Dự đoán và đánh giá mô hình**

Sau khi huấn luyện, mô hình được sử dụng để dự đoán giá trị mục tiêu trên tập kiểm tra X\_test bằng phương thức predict. Các giá trị dự đoán sau đó được so sánh với các giá trị thực tế trong tập kiểm tra (y\_test). Kết quả dự đoán và thực tế được trình bày dưới dạng bảng dữ liệu với hai cột: "Thực tế" (giá trị thực tế) và "Dự đoán" (giá trị dự đoán từ mô hình).

A screenshot of a computer code

Description automatically generated

*Hình 4.3.4 Kết quả dự đoán từ mô hình với 5 mẫu đầu tiên*

Để đánh giá hiệu quả của mô hình, các chỉ số như **Mean Squared Error (MSE)**, **Mean Absolute Error (MAE)** và **R² (Coefficient of Determination)** có thể được tính toán. Những chỉ số này giúp đo lường độ chính xác của mô hình trong việc dự đoán giá trị mục tiêu và đưa ra những đánh giá về khả năng tổng quát của mô hình.

**4.3.2 Kết quả của mô hình**

Sau khi hoàn thành quá trình huấn luyện và kiểm tra mô hình XGBoost Regressor, các kết quả đánh giá hiệu suất được trình bày ở bảng sau (kết quả được làm tròn hai chữ số sau dấu phẩy):

| **Model** | **Mean Squared Error (MSE)** | **Mean Absolute Error (MAE)** | **R² Score** |
| --- | --- | --- | --- |
| AdaBoost Regressor | 1883960.6526 | 1049.4828 | 0.9887 |

*Bảng 4.3 Các độ đo trong mô hình AdaBoost Regressor*

Kết quả từ mô hình AdaBoost Regressor cho thấy giá trị Mean Squared Error (MSE) là 1,883,960.65, thấp hơn so với các mô hình trước, cho thấy độ lệch trung bình bình phương giữa giá trị dự đoán và giá trị thực tế là khá nhỏ. Mean Absolute Error (MAE) đạt 1,049.48, cho thấy sai lệch tuyệt đối trung bình vẫn ở mức hợp lý và không quá cao, chứng tỏ độ chính xác của mô hình trong việc dự đoán giá trị thực tế. Đặc biệt, R² score của AdaBoost Regressor đạt 0.9887, cao hơn một chút so với các mô hình trước, phản ánh rằng mô hình có khả năng giải thích tới 98.87% phương sai của dữ liệu, tức là độ phù hợp của mô hình với dữ liệu thực tế rất cao

Trực quan hoá kết quả thu được :

A graph with blue dots

Description automatically generated

*Hình 4.3.5 So sánh giá trị thực tế và dự đoán của mô hình AdaBoost*

A graph with a line with Ryugyong Hotel in the background

Description automatically generated

*Hình 4.3.6 Phân phối sai số dự đoán*

## **4.4 Thiết lập mô hình XGBoost (Extreme Gradient Boosting)**

**4.4.1 Quy trình thực hiện**

Mô hình **XGBoost (Extreme Gradient Boosting)** là một trong những thuật toán mạnh mẽ trong nhóm học máy, được tối ưu hóa cho cả tốc độ lẫn hiệu suất. Được biết đến với khả năng xử lý các vấn đề phức tạp và mô hình hóa dữ liệu có đặc tính không tuyến tính, XGBoost rất hữu ích cho bài toán dự đoán giá cổ phiếu. Các bước triển khai mô hình XGBoost Regressor trong bài toán này bao gồm:

**Bước 1:Chuẩn bị dữ liệu**

**Dữ liệu sạch:** Giống như với mô hình Random Forest, dữ liệu phải được xử lý sạch sẽ và không chứa các giá trị khuyết (NaN). Các đặc trưng (features) được xác định từ bộ dữ liệu gốc và bao gồm các chỉ báo kỹ thuật như SMA, EMA, RSI, MACD, Bollinger Bands, và các chỉ số khác.

**Cột mục tiêu:** Dịch chuyển cột "close" xuống một hàng để tạo ra biến mục tiêu "Target," với giá trị của ngày tiếp theo. Điều này giúp mô hình dự đoán giá cổ phiếu trong ngày tiếp theo.

**Phân chia dữ liệu:** Sau khi dữ liệu đã được xử lý, tiến hành phân chia thành tập huấn luyện (80%) và tập kiểm thử (20%), giúp đảm bảo mô hình được kiểm nghiệm một cách công bằng trên dữ liệu chưa được huấn luyện.

**Bước 2: Khởi tạo mô hình XGBoost**

Mô hình XGBoost Regressor được khởi tạo với các tham số cơ bản như sau:

* n\_estimators=100: Sử dụng 100 cây quyết định trong mô hình để tăng khả năng tổng hợp và cải thiện độ chính xác.
* earning\_rate=0.1: Tốc độ học thấp giúp mô hình học một cách từ từ và giảm thiểu hiện tượng overfitting.
* max\_depth=3: Độ sâu tối đa của các cây quyết định, giúp điều chỉnh độ phức tạp của mô hình, tránh việc mô hình quá khớp (overfitting).
* random\_state=42: Đảm bảo tính lặp lại và kết quả có thể tái tạo trong các lần chạy khác nhau.

A screenshot of a computer

Description automatically generated

*Hình 4.4.1 Khởi tạo mô hình XGBoost Regressor*

**Bước 3: Huấn luyện mô hình**

Sau khi khởi tạo mô hình, phương thức fit được sử dụng để huấn luyện mô hình trên tập huấn luyện (X\_train và y\_train). Trong quá trình này, mô hình XGBoost học được các mối quan hệ giữa các đặc trưng và biến mục tiêu, từ đó tạo ra một mô hình dự đoán giá trị của cổ phiếu.

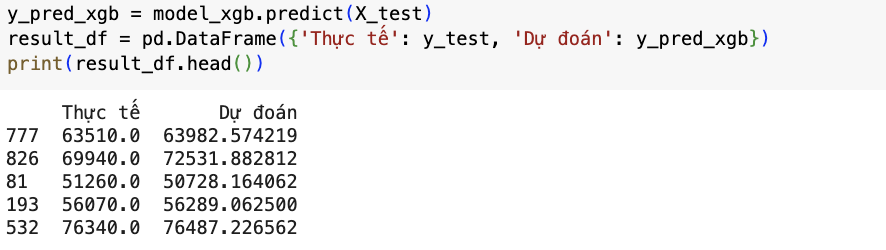
A close up of a word

Description automatically generated

*Hình 4.4.2 Huấn luyện mô hình XGBoost Regressor*

**Bước 4: Dự đoán giá trị**

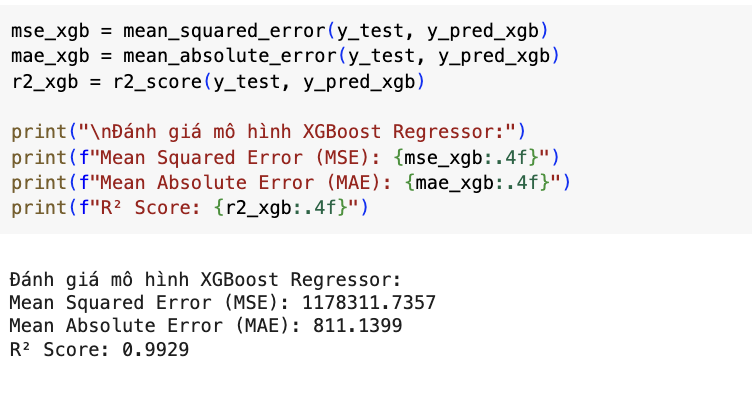
Sau khi huấn luyện, mô hình sử dụng phương thức predict để đưa ra dự đoán về giá trị mục tiêu (y\_pred\_xgb) trên tập kiểm thử (X\_test). Kết quả này cho phép so sánh giữa giá trị thực tế và giá trị dự đoán, giúp đánh giá hiệu quả của mô hình.



*Hình 4.4.3 Kết quả dự đoán 5 mẫu đầu của mô hình*

**Bước 5: Đánh giá mô hình:**

* **Mean Squared Error (MSE):** Tính toán trung bình bình phương sai số giữa giá trị thực tế và giá trị dự đoán. Đây là một chỉ số quan trọng giúp đánh giá độ sai lệch của mô hình.
* **Mean Absolute Error (MAE):** Đo lường sai số tuyệt đối trung bình giữa giá trị thực tế và dự đoán, cung cấp cái nhìn dễ hiểu về mức độ sai lệch trung bình.
* **R² Score:** Đánh giá mức độ giải thích của mô hình đối với biến động của giá trị mục tiêu. Giá trị R² gần với 1 cho thấy mô hình có khả năng dự đoán tốt, còn gần 0 thể hiện mô hình không giải thích được nhiều sự thay đổi trong dữ liệu.



*Hình 4.4.4 Các chỉ số đánh giá mô hình*

**4.4.2 Kết quả của mô hình**

Sau khi hoàn thành quá trình huấn luyện và kiểm tra mô hình XGBoost Regressor, các kết quả đánh giá hiệu suất được trình bày ở bảng sau (kết quả được làm tròn bốn chữ số sau dấu phẩy):

| **Model** | **Mean Squared Error (MSE)** | **Mean Absolute Error (MAE)** | **R² Score** |
| --- | --- | --- | --- |
| XGBoost Regressor | 1178311.7357 | 811.1399 | 0.9929 |

*Bảng 4.4 Kết quả đánh giá hiệu suất mô hình XGBoost Regressor*

Kết quả từ mô hình XGBoost Regressor cho thấy giá trị Mean Squared Error (MSE) là 1,178,311.74, thấp hơn đáng kể so với các mô hình trước, cho thấy độ lệch trung bình bình phương giữa giá trị dự đoán và giá trị thực tế rất nhỏ, phản ánh độ chính xác cao của mô hình. Mean Absolute Error (MAE) đạt 811.14, cho thấy sai lệch tuyệt đối trung bình giữa các dự đoán và giá trị thực tế thấp, củng cố khả năng dự đoán chính xác của mô hình. Đặc biệt, R² score của XGBoost Regressor đạt 0.9929, cao hơn tất cả các mô hình trước, chỉ ra rằng mô hình có thể giải thích tới 99.29% phương sai của dữ liệu, tức là độ phù hợp và sự chính xác của mô hình rất cao

Trực quan hoá các kết quả thu được:



A graph with a line

Description automatically generated

## 

## **4.5 Thiết lập mô hình SVR**

**4.5.1 Quy trình thực hiện**

**Bước 1: Chuẩn bị và xử lý dữ liệu**

Trong bước đầu tiên, chúng ta chuẩn bị dữ liệu bằng cách loại bỏ các giá trị thiếu bằng phương pháp dropna() của Pandas. Dữ liệu bao gồm nhiều đặc trưng kỹ thuật, như giá mở cửa (open), giá cao nhất (high), giá thấp nhất (low), khối lượng giao dịch (volume), cùng với các chỉ báo kỹ thuật như SMA\_10, EMA\_10, RSI\_14, MACD, MACD Signal, Bollinger Bands, ATR\_14, Stochastic Oscillator (Stoch\_K, Stoch\_D), và ADX\_14. Dữ liệu này sẽ là các đặc trưng đầu vào cho mô hình. Biến đầu ra (target) là giá đóng cửa (close), được sử dụng để huấn luyện và kiểm tra mô hình.

A screenshot of a computer program

Description automatically generated

*Hình 4.5.1 Chuẩn bị và xử lý dữ liệu*

Sau khi xác định được các biến đầu vào và đầu ra, tập dữ liệu được chia thành hai phần: 80% dữ liệu được sử dụng để huấn luyện mô hình, và 20% còn lại dùng để kiểm tra độ chính xác của mô hình.

**Bước 2: Chuẩn hóa dữ liệu**

Để đảm bảo rằng các đặc trưng đầu vào có cùng phạm vi giá trị và không làm ảnh hưởng đến quá trình huấn luyện, chúng ta sử dụng phương pháp chuẩn hóa (scaling). Cả dữ liệu đầu vào (X) và đầu ra (y) đều được chuẩn hóa bằng cách sử dụng StandardScaler từ thư viện sklearn.preprocessing. Phương pháp này giúp dữ liệu có giá trị trung bình bằng 0 và phương sai bằng 1, giúp mô hình học nhanh hơn và chính xác hơn.

A screenshot of a computer code

Description automatically generated

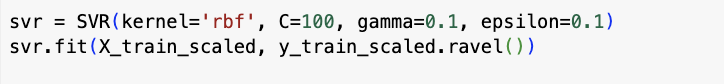
*Hình 4.5.2 Chuẩn hoá dữ liệu*

Dữ liệu huấn luyện và kiểm tra sau khi được chuẩn hóa sẽ được sử dụng cho bước huấn luyện mô hình.

**Bước 3: Xây dựng mô hình SVR**

Tiếp theo, chúng ta xây dựng mô hình Support Vector Regression (SVR) sử dụng hàm kernel RBF (Radial Basis Function). SVR là một thuật toán học máy mạnh mẽ, đặc biệt hiệu quả trong việc dự đoán các giá trị liên tục. Các tham số chính của mô hình bao gồm:

* **C**: Hệ số điều chỉnh mức phạt đối với các điểm dữ liệu nằm ngoài vùng biên epsilon (C=100C = 100C=100).
* **Gamma**: Được sử dụng để xác định mức độ ảnh hưởng của các điểm dữ liệu trong không gian đặc trưng (γ=0.1\gamma = 0.1γ=0.1).
* **Epsilon**: Xác định khoảng biên mà không có hình phạt đối với sai số (ϵ=0.1\epsilon = 0.1ϵ=0.1).

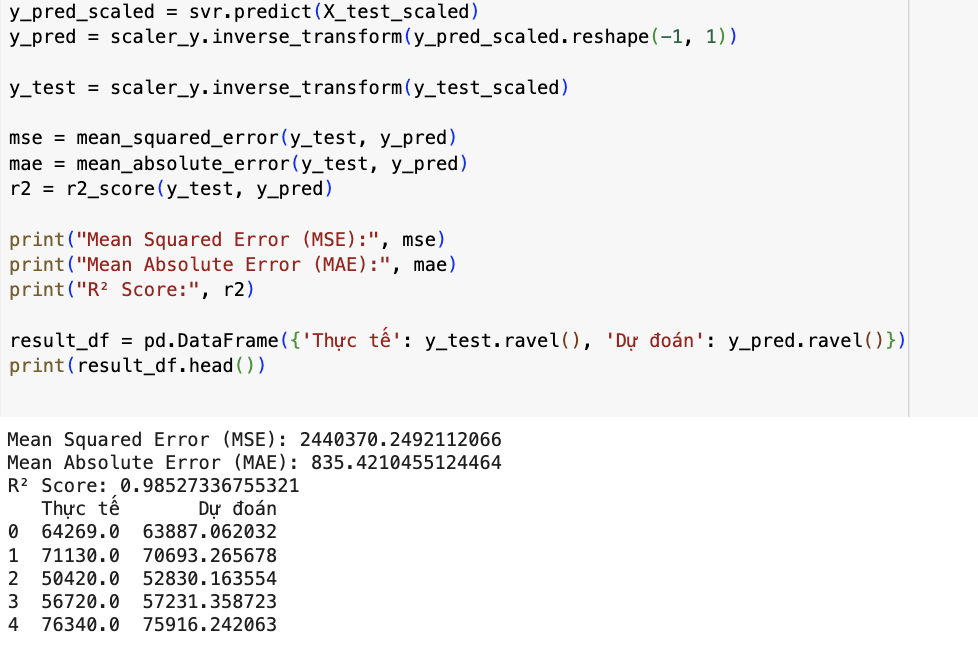


*Hình 4.5.3 Xây dựng mô hình SVR*

Mô hình SVR sẽ được huấn luyện với dữ liệu đã chuẩn hóa từ tập huấn luyện.

**Bước 4: Dự đoán và đánh giá mô hình**

Sau khi mô hình được huấn luyện, chúng ta tiến hành dự đoán giá đóng cửa cho tập kiểm tra. Dự đoán của mô hình ban đầu sẽ ở dạng chuẩn hóa, vì vậy chúng ta sử dụng inverse\_transform để chuyển đổi kết quả dự đoán về giá trị thực tế trong không gian gốc.



*Hình 4.5.4 Dự đoán và đánh giá mô hình*

Để đánh giá hiệu quả của mô hình, chúng ta sử dụng ba chỉ số chính: **Mean Squared Error (MSE)**, **Mean Absolute Error (MAE)** và **R² Score**.

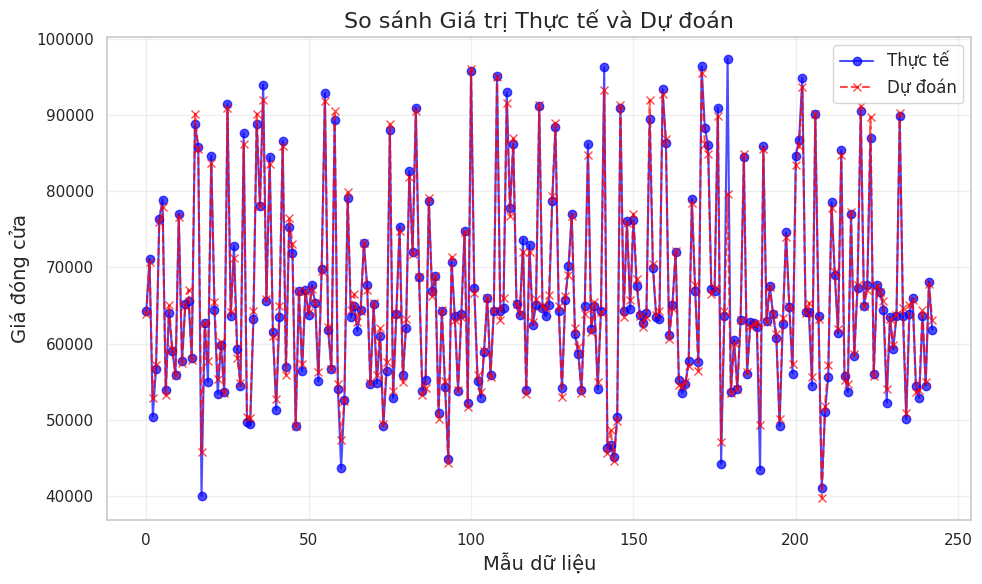
**4.5.2 Kết quả mô hình**

Sau khi đã huấn luyện và dự đoán giá đóng cửa (close price) trên tập kiểm tra Support Vector Regression (SVR) sau khi đã huấn luyện và dự đoán giá đóng cửa (close price) trên tập kiểm tra, ta có bảng sau:

| **Model** | **Mean Squared Error (MSE)** | **Mean Absolute Error (MAE)** | **R² Score** |
| --- | --- | --- | --- |
| SVR | 2440370.2492 | 835.4210 | 0.9852 |

*Bảng 4.5 Kết quả mô hình SVR*

Các kết quả này bao gồm các chỉ số đánh giá mô hình như Mean Squared Error (MSE), Mean Absolute Error (MAE) và R² Score, đồng thời cung cấp cái nhìn về độ chính xác và hiệu suất của mô hình đối với dữ liệu thực tế.



*Hình 4.5.5 So sánh giá trị thực tế và dự đoán*

A graph with red lines

Description automatically generated

*Hình 4.5.6 Biểu đồ phân phối sai lệch*

# **CHƯƠNG 5 : KẾT LUẬN NGHIÊN CỨU**

## **5.1 Kết luận nghiên cứu**

Qua quá trình triển khai và đánh giá các mô hình học máy bao gồm Decision Tree, Random Forest, AdaBoost, XGBoost, và Support Vector Machine (SVM) trong bài toán dự đoán giá cổ phiếu, chúng tôi đã thu được một số kết quả đáng chú ý. Cụ thể, các mô hình Boosting như AdaBoost và XGBoost cho thấy khả năng cải thiện hiệu quả dự đoán so với các mô hình cơ bản. Tuy nhiên, đối với bài toán dữ liệu có nhiễu và biến động như giá cổ phiếu, các mô hình phức tạp hơn như XGBoost và Random Forest có xu hướng hoạt động ổn định và cho độ chính xác cao hơn. Ngược lại, SVM với các tham số cơ bản lại không đạt hiệu quả tốt do khó tối ưu hóa trên dữ liệu không tuyến tính mà không điều chỉnh kernel phù hợp. Kết quả cho thấy tầm quan trọng của việc lựa chọn mô hình và tinh chỉnh các tham số để đạt hiệu suất cao nhất.

## **5.2 Hạn chế nghiên cứu**

Mặc dù đạt được một số kết quả nhất định, nghiên cứu này còn một số hạn chế đáng lưu ý. Đầu tiên, việc sử dụng các tham số mặc định của các thuật toán có thể khiến các mô hình chưa đạt được hiệu quả tối ưu, đặc biệt là với SVM và XGBoost. Thứ hai, bộ dữ liệu được sử dụng có thể chưa bao quát đủ các yếu tố tác động đến giá cổ phiếu như yếu tố kinh tế, xã hội, và tâm lý thị trường. Hơn nữa, mô hình chỉ được huấn luyện và kiểm thử trên một tập dữ liệu nhất định, do đó có thể dẫn đến việc mô hình thiếu khả năng tổng quát hóa khi áp dụng trên các tập dữ liệu khác. Cuối cùng, do hạn chế về thời gian và tài nguyên, quá trình kiểm thử trên dữ liệu thực tế có quy mô lớn và thời gian dài vẫn chưa được thực hiện đầy đủ.

## **5.3 Đề xuất**

Để cải thiện hiệu quả dự đoán, chúng em đề xuất một số hướng tiếp cận như sau:

1. Tối ưu hóa các tham số của từng thuật toán thông qua các phương pháp như Grid Search hoặc Random Search để tìm ra các tham số phù hợp nhất.
2. Thử nghiệm thêm các mô hình phức tạp khác, hoặc kết hợp các mô hình hiện có để tạo ra mô hình ensemble mạnh hơn.
3. Bổ sung thêm các yếu tố kinh tế vĩ mô và dữ liệu mạng xã hội nhằm cải thiện độ chính xác của mô hình.
4. Tăng cường khả năng dự báo dài hạn của mô hình bằng cách sử dụng các kỹ thuật học sâu như LSTM hoặc các mạng neural hồi quy.

## **5.4 Hướng phát triển trong tương lai**

Trong tương lai, chúng em hướng đến việc mở rộng nghiên cứu bằng cách thử nghiệm trên tập dữ liệu có quy mô lớn hơn và mang tính chất toàn cầu, bao gồm các chỉ số kinh tế quốc tế. Ngoài ra, áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) để phân tích tâm lý thị trường từ các bài báo hoặc phương tiện truyền thông cũng là một hướng đi tiềm năng. Việc xây dựng hệ thống dự báo tự động và tích hợp vào các hệ thống giao dịch tài chính cũng là mục tiêu hướng tới, nhằm nâng cao ứng dụng thực tế và tính hiệu quả của nghiên cứu này.

# **TÀI LIỆU THAM KHẢO**

1. Desicion Tree Algorithm . <https://www.geeksforgeeks.org/decision-tree/>

2. AdaBoost . <https://viblo.asia/p/adaboost-buoc-di-dau-cua-boosting-gAm5yrGwKdb>

3.XBoost.<https://ongxuanhong.wordpress.com/2017/12/21/xgboost-thuat-toan-gianh-chien-thang-tai-nhieu-cuoc-thi-kaggle/>

4.XBoost.<https://machinelearningmastery.com/extreme-gradient-boosting-ensemble-in-python/>

5.SVR.<https://towardsdatascience.com/machine-learning-basics-support-vector-regression-660306ac5226>

6.Random Forest algorithm. <https://machinelearningcoban.com/tabml_book/ch_model/random_forest.html>

7.Độ đo R^2 . <https://www.phamlocblog.com/2019/10/y-nghia-r-binh-phuong-hieu-chinh.html>

8.[Bài toán hồi quy và mô hình hồi quy tuyến tính](https://viblo.asia/p/bai-toan-hoi-quy-va-mo-hinh-hoi-quy-tuyen-tinh-GrLZDQgOlk0).

9. [Đánh giá model trong Machine Learing](https://viblo.asia/p/danh-gia-model-trong-machine-learing-RnB5pAq7KPG)