ĐẠI HỌC QUỐC GIA TP. HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH

LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

Xây dựng mô hình dự đoán xu hướng giá ngắn hạn các đồng tiền mật mã bằng kĩ thuật học máy

HỘI ĐỒNG: Khoa Học Máy Tính
GVHD: TS. NGUYỄN AN KHƯƠNG
GVPB: TS. LÊ THÀNH SÁCH

o0o

SVTH: VŨ QUANG NAM (1512107)

TP. HỒ CHÍ MINH, Tháng 12 năm 2019

Lời cam đoan

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi dưới sự hướng dẫn của thầy Nguyễn An Khương và anh Nguyễn Lê Thành. Nội dung nghiên cứu và các kết quả đều là trung thực và chưa từng được công bố trước đây. Các số liệu được sử dụng cho quá trình phân tích, nhận xét được chính tôi thu thập từ nhiều nguồn khác nhau và sẽ được ghi rõ trong phần tài liệu tham khảo. Ngoài ra, tôi cũng có sử dụng một số nhận xét, đánh giá và số liệu của các tác giả khác, cơ quan tổ chức khác. Tất cả đều có trích dẫn và chú thích nguồn gốc. Nếu phát hiện có bất kì sự gian lận nào, tôi xin hoàn toàn chịu trách nhiệm về nội dung thực tập tốt nghiệp của mình. Trường đại học Bách Khoa thành phố Hồ Chí Minh không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện.

Lời cảm ơn

Lời đầu tiên, tôi xin gửi lời cảm ơn chân thành nhất đến TS. Nguyễn An Khương và anh Nguyễn Lê Thành đã tận tình hướng dẫn trong quá trình chuẩn bị kiến thức để làm luận văn. Tôi cũng xin cảm ơn các bạn trong nhóm Datavisian đã rất nhiệt tình giúp đỡ và góp ý trong quá trình thực hiện các mô hình, các bạn có những phẩm chất của một nhà khoa học dữ liệu mà tôi nên học hỏi.

Trong khoảng thời gian trên, tôi được anh Văn Duy Vinh chỉ bảo những khóa học trực tuyến về học máy, các lý thuyết cơ bản về xác suất và hơn hết là ý tưởng chung của mô hình đồ thị xác suất. Với một nền tảng lý thuyết còn non nớt, tôi đã dần được tiếp cận lý thuyết thông qua những bài tập trên các khóa học và phần nào hiểu hơn khi đọc sách in "Pattern Recognition and Machine Learning" do anh cung cấp.

Sau khi hiện thực mô hình từ ý tưởng gốc của mô hình đồ thị xác suất, anh Nguyễn Xuân Mão và anh Hoàng Phạm Thanh Tài có bổ sung thêm những điểm đáng lưu ý mà tôi chưa nhận ra của mô hình từ đó mở ra cho tôi một bức tranh tổng quát hơn về các mô hình học máy.

Khi giai đoạn hiện thực kết thúc, công việc trở lên phức tạp hơn khi trình bày và viết luận văn. Bản thân tôi cảm thấy đây là giai đoạn khó khăn nhất trong quá trình học tập trước đây, việc hoàn thành được luận văn không thể không kể đến anh Võ Trọng Thư, bạn Nguyễn Quốc Bảo đã đóng góp ý kiến. Một lần nữa xin cảm ơn anh Thư và bạn Quốc Bảo dù bận rộn nhưng đã bỏ thời gian để giúp bài luận văn được tốt hơn.

Và từ đáy lòng mình, tôi xin cảm ơn chị gái và mẹ. Hai người là điểm tựa tinh thần vững chắc cho tôi trong những lúc khó khăn nhất, yêu thương tôi vô điều kiện và ủng hộ những quyết định của tôi.

Nhìn lại quá trình làm luận văn, tôi cảm thấy thật may mắn khi bản thân đang là một sinh viên đại học trên giảng đường được tiếp cận các nguồn lực tốt đến vậy để hoàn thành việc nghiên cứu và củng cố thêm nền tảng cho tương lai.

Tóm tắt nội dung

Hiện nay, tiền mã hóa được biết đến và sử dụng rộng rãi. Việc đầu tư vào thị trường đầy tiềm năng này ẩn chứa nhiều rủi ro. Với mục tiêu tạo ra được lợi nhuận từ việc đầu tư và giảm thiểu rủi ro, cần có các độ đo, đánh giá từ dữ liệu trước đó. Một nhu cầu của nhà đầu tư ngắn hạn là cần dự đoán tình trạng giá của các đồng trong thời gian tiếp theo. Trong đề tài này, chúng tôi sẽ

Mục lục

Danh sách bảng

Danh sách hình vẽ

Chương 1

Giới thiệu

Giới thiệu đề tài nghiên cứu

Hiện nay, tiền mã hóa đã dần trở nên phổ biến và được sử dụng trên nhiều sàn giao dịch. Từ đó có thể dễ dàng mua bán, trao đổi trực tuyến, nhanh chóng. Với số lượng giao dịch ngày càng tăng, tỷ giá giữa các đồng liên tục thay đổi. Trong việc đầu tư, nhu cầu kiểm soát các rủi ro cũng như tính toán lợi nhuận được đặt lên hàng đầu. Khi xét trên phương diện thời gian, chiến lược đầu tư được chia thành hai loại chính là đầu tư dài hạn và đầu tư ngắn hạn. Để kiểm soát được các rủi ro trong việc đầu tư ngắn hạn, cần các công cụ dự đoán giá, xu hướng giá trong các phiên giao dịch tiếp theo. Từ dữ liệu cụ thể là tổng hợp của các giao dịch trên các sàn trực tuyến việc tìm ra một giải thuật có thể dự đoán xu hướng giá của các giao dịch tiếp theo với nguyên tắc đề cao khách quan so với kinh nghiệm bản thân là một vấn đề mới mẻ. Vậy nên chúng tôi quyết định chọn đề tài **Dự đoán xu hướng giá ngắn hạn các đồng tiền mật mã bằng kĩ thuật học máy**.

Mục tiêu và phạm vi đề tài

Đối tượng nghiên cứu

Đề tài gồm ba đối tượng nghiên cứu như sau:

- Đồng tiền nghiên cứu.
- Chiến lược ngắn hạn.
- Các mô hình học máy cho dữ liệu thời gian.

Phạm vi nghiên cứu

Về đối tượng sàn mã hóa: Hiện nay, trên thị trường hiện nay có nhiều sàn giao dịch khác nhau. Để giới hạn phạm vi, đối tượng sàn để nghiên cứu cần phải bao gồm các tiêu chí như tính thanh khoản cao với số lượng giao dịch nhiều, minh bạch về lịch sử giao dịch, ngoài ra sàn phải cung cấp lịch sử giá giữa các cặp đồng với nhau. Thông qua tìm hiểu, sàn *Binance* được thành lập vào tháng 7/2017 là một trong những sàn uy tín với tính thanh khoản cao (số lượng giao dịch luôn nằm trong top 5 những sàn giao dịch trên thế giới), sàn cung cấp api để tra cứu giá giao dịch, lịch sử giao dịch. Ngoài ra sàn *Binance* còn cung cấp cho một tài khoản có thể tạo tối đa 200 tài khoản phụ (tính năng chỉ áp dụng cho tài khoản đặc biệt theo chính sách của sàn), điều này giúp việc so sánh các chiến lược giao dịch với nhau dựa trở nên dễ dàng hơn khi giao dịch thực tế. Chính vì những lí do trên chúng tôi lựa chọn sàn *Binance* để nghiên cứu và triển khai các chiến lược sau này.

Về đối tượng đồng mã hóa: Hiện nay có hơn 4939 loại đồng mã hóa [1] nên việc lựa chọn các cặp đồng mã hóa được chúng tôi xếp theo lượng giao dịch trong ngày. *Bitcoin* và *Ethereum* là hai đồng có sức mua, bán cao nhất trên sàn Binance tính theo hai cặp tương ứng *BTC/USDT* và *ETH/USDT*. Ngoài ra sàn còn cung cấp đồng Binance (BNB) nằm trong top 10 khi xét về khối lượng giao dịch, giao dịch các cặp khi có đồng này phí giao dịch sẽ được giảm xuống 25%. Việc lựa chọn 2 đồng cơ bản là BTC, ETH ứng với 2 cặp sẽ được đề cập trong phần nghiên cứu. Trong tương lai khi so sánh thực nghiệm theo lợi nhuận đối với mỗi chiến lược sẽ bổ sung đồng BNB.

[1] `https://coinmarketcap.com/all/views/all/` cập nhật vào ngày 2019/ 09/16

Phương pháp nghiên cứu

Trong quá trình nghiên cứu, nhóm có ba công việc chính cần giải quyết:

- Thu thập dữ liệu từ sàn.
- Thống kê dữ liệu đã thu thập.
- Nghiên cứu những mô hình cho việc dự đoán trên dữ liệu thời gian. Thí nghiệm mô hình trên dữ liệu đã được thu thập.

Bố cục luận văn

Bố cục luận văn với nội dung tác giả trình bày được chia thành các phần sau đây:

- **Chương 1** Giới thiệu đề tài: Khái quát về vấn đề liên quan đến các chiến lược giao dịch và sự cần thiết của một hệ thống học máy trong mô hình dự đoán.
- **Chương 2** Các công trình liên quan: Đưa ra một số các công trình dự đoán xu hướng tăng giảm về giá của các đồng tiền mã hóa đã tham khảo.
- **Chương 3, 4** Các phương pháp nghiên cứu, quá trình hiện thực: Trình bày sơ lược về lý do sử dụng; ưu, nhược điểm của các mô hình học máy có sử dụng trong đề tài. Các khái niệm cần thiết sẽ được trình bày để làm rõ thêm các mô hình sử dụng.
- **Chương 6** Tổng kết: Kết quả đạt được, những hạn chế của các chiến lược mô hình đã được sử dụng và hướng phát triển hệ thống trong sau này.

Chương 2

Các công trình liên quan

Trong chương này, chúng tôi sẽ trình bày các công trình liên quan tới việc thu thập dữ liệu về đồng tiền mã hóa cũng như các kết luận rút ra sau khi thí nghiệm các mô hình học máy trên dữ liệu.

Công trình dự đoán giá Bitcoin dựa trên giải thuật học máy

Isaac Madan, Shaurya Saluja và Aojia Zhao [1] đã ứng dụng các mô hình học máy để dự đoán giá của đồng Bitcoin với kết quả có độ chính xác vào khoảng 50-55% trong việc dự đoán giá sau 10 phút tăng hoặc giảm. Nhóm tác giả đã hiện thực việc thu thập thông qua api của sàn Coinbase và sàn OKCoin và cho ra dữ liệu gồm 25 đặc trưng liên quan tới đồng Bitcoin trong vòng liên tục 5 năm. Nhón tác giả đã hiện thực các mô hình như SVM, rừng quyết định, và Binomial GLM. Thông qua lần lượt ba mô hình, nhóm tác giả đã thí nghiệm khi tăng khoảng thời gian dự đoán từ 10 giây lên 10 phút, độ chính xác của mô hình GLM và rừng quyết định tăng lên, với mô hình SVM cho kết quả giảm xuống. Rừng ngẫu nhiên có tính chính xác (precision) thấp hơn so với mô hình GLM. Dựa trên kết quả trên, nhóm tác giả đã đưa ra hai kết luận sau:

- Mô hình rừng ngẫu nhiên có hiện tượng thừa cây quyết định (decision tree), tuy nhiên trên dữ liệu kiểm thử kết quả đạt 57.4% chứng tỏ dữ liệu kiểm thử không

quá khác biệt so với tập huấn luyện. Thêm vào đó rừng ngẫu nhiên cũng đưa ra chỉ số chính xác (precision) thấp, mô hình dự đoán rơi nhiều vào giá tăng gây hiện tượng khi dự đoán, xu hướng tăng thường lấn át xu hướng giảm.

- Khi kết hợp mô hình rừng ngẫu nhiên và GLM theo hàm tuyến tính với các trọng số tỉ lệ thuận với độ chính xác của hai mô hình, chỉ số nhạy (sensitivity) đối với dữ liệu phiên 10 phút cao hơn so với dữ liệu phiên 10 giây.

Công trình dự đoán giá Bitcoin dựa trên mạng lưới giao dịch

Alex Greaves và Benjamin Au [2] phân tích các mạng giao dịch (transaction graph) để dự đoán xu hướng đồng Bitcoin cho ra kết quả vào khoảng 55%. Dữ liệu được thu thập trên mạng lưới giao dịch (transaction graph) trong vòng 1 năm tính từ 2012/01/01 đến 2013/01/01 gồm một vài đặc trưng như: giá đồng Bitcoin hiện tại, trung bình số nút vào (in-node) và nút ra (out-node), số lượng Bitcoin đã được "đào". Các mô hình phân loại được sử dụng trong bài báo trên gồm hồi quy Logistic, SVM, mạng nơ-ron với 2 lớp. Một kết luận quan trọng đưa ra trong bài báo trên về hành vi mua bán: khi số lệnh giao dịch biến động (tăng lên hoặc giảm xuống nhanh), người tham gia có xu hướng tích lũy và ít giao dịch lại.

Các mô hình học máy nổi bật

Trong các công trình nghiên cứu trên, xét về mô hình học máy có các mô hình chung như:

- Rừng ngẫu nhiên
- Support vector machine
- Hồi quy logistic.

Với mỗi mô hình học máy kể trên, chúng tôi sẽ đưa ra các lý do sử dụng, cơ chế chính và hạn chế riêng trên tập dữ liệu nghiên cứu

Rừng ngẫu nhiên

Trong trường hợp giá có xu hướng tăng liên tục trong vòng 5 giờ, giá phiên cuối sẽ tăng so với đường trung bình động hay Moving Average (MA) giá 4 giờ trước. Với cây quyết định, quy luật trên là có thể biểu diễn được khi dữ liệu được thêm thuộc tính mới như biên độ của giá hiện tại và MA 4 giờ trước. Để có thể biểu diễn được các quy luật trên, mô hình được cải tiến thành Rừng ngẫu nhiên khi lựa chọn ngẫu nhiên các thuộc tính của dữ liệu gốc để hình thành các cây quyết định riêng và tổng hợp lại.

Ưu điểm của rừng ngẫu nhiên (Random Forest) là từ từng cây, ta có thể mô tả được tập quy luật tương ứng.

Nhược điểm của cây quyết định dễ nhìn thấy khi phân nhánh cây được chia dựa trên một thuộc tính, do đó việc chọn thuộc tính trong phần xử lý dữ liệu trước khi đưa vào mô hình yêu cầu kiến thức về dữ liệu chuỗi thời gian cũng như kinh nghiệm giao dịch; thêm nữa các thuộc tính thường không đơn giản khi kết hợp số lượng lớn thuộc tính cụ thể như việc biểu diễn mối quan hệ của 3 thuộc tính từ dữ liệu gốc: giá đóng phiên; số lượng người tham gia; số lượng đồng giao dịch thành thuộc tính mới.

Máy vectơ hỗ trợ

Khi biểu diễn dữ liệu dạng 2 chiều như hình nhãn tăng, giảm phiên giao dịch rất khó phân biệt, tuy nhiên khi trên không gian lớn hơn như 36 chiều, việc có thể tìm được đường biên để chia là khả thi khi sử dụng mô hình Máy vectơ hỗ trợ (Support Vector Machine-SVM). Tuy nhiên, vì dữ liệu nhiều chiều như vậy, việc hình dung không gian có số chiều lớn dựa trên hình chiếu 2 chiều là không thể, mô hình lúc này trở thành hộp kín (black box learning algorithm). Do đó việc xử lý dữ liệu, lựa chọn thuộc tính trước khi đưa vào SVM đóng vai trò quan trọng.

Hồi quy logistic

Mô hình hồi quy logistic (logistic regression) được coi như một mô hình phân loại. Mô hình đạt kết quả tốt trong bài toán với dữ liệu có thể dễ dàng phân tách trên đường phân cách tuyến tính. Một điểm hạn chế của mô hình khi trên các dữ liệu bị trùng và ảnh hưởng lẫn nhau như: phiên giao dịch của tháng trước và phiên hiện tại có các thông số như nhau, tuy nhiên sau khi kết thúc phiên, giá của phiên sau sẽ tăng trong khi phiên tháng trước giá giảm.

Chương 3

Phương pháp nghiên cứu

Định nghĩa bài toán

Kiến trúc hệ thống

Thu thập dữ liệu

Hiện nay đa số các sàn giao dịch lớn như Binance, Huobipro, OKCoin đều cung cấp api hỗ trợ cho phép xem lại các OrderBook, tỷ giá giao dịch các phiên trước đó, đặt lệnh giao dịch. Như đã đề cập ở chương 1, trong phạm vi sàn nghiên cứu của đề tài, chúng tôi chọn sàn Binance và các đồng cơ bản là BTC, ETH, USDT, BNB để nghiên cứu; Thông qua tìm hiểu, có hai thư viện hỗ trợ thu thập thông qua api trên sàn biannce là python-binance do sàn viết và ccxt (đã được Binance chứng nhận) đều được viết bằng ngôn ngữ python. Ccxt với khả năng hỗ trợ hơn 120 sàn khác nhau, hỗ trợ với nhiều ngôn ngữ lập trình, chính vì vậy ccxt được chọn làm thư viện chính để thu thập dữ liệu và tạo các lệnh giao dịch để có thể mở rộng đề tài trong tương lai.
Để dễ hình dung về chức năng của api, api được ccxt chia thành hai loại là:

- Public api: hỗ trợ lấy tickers, OrderBook; tỉ giá cặp tại thời điểm trước; các giao dịch trong khoảng thời gian trước đó.

- Private api: hỗ trợ tạo, hủy lệnh giao dịch; lấy các lệnh giao dịch trước đây; xem số đồng trong các ví của tài khoản sở hữu api này.

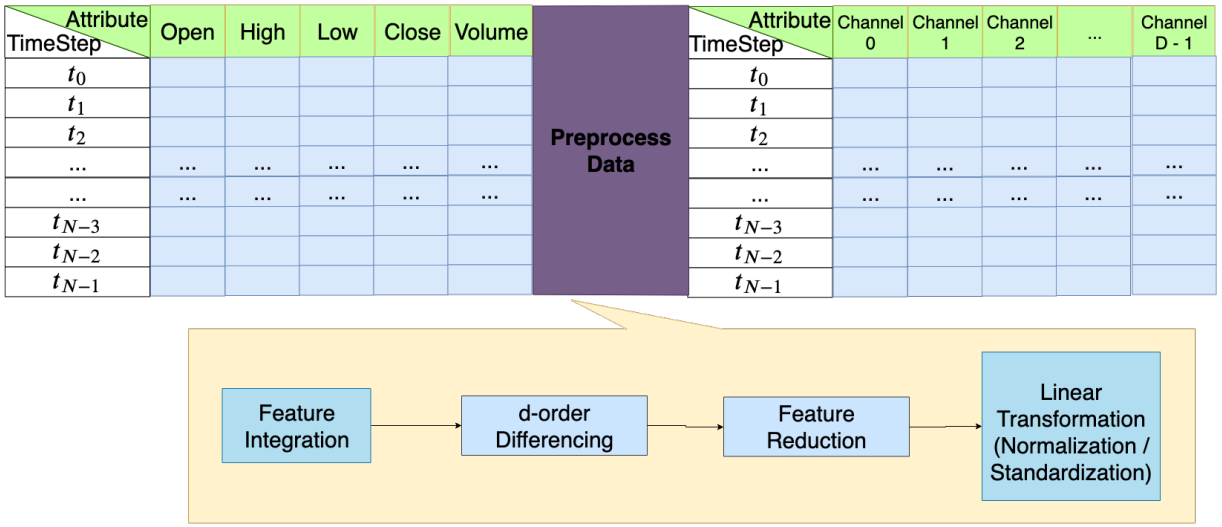
Quá trình chuẩn bị dữ liệu được thực hiện thông qua public api.
Khác với sàn giao dịch truyền thống (sàn chứng khoán), sàn giao dịch tiền mã hóa hoạt động liên tục vì vậy việc chia phiên giao dịch được sàn định nghĩa theo các khoảng thời gian cụ thể theo mặc định như 1 phút, 1 giờ, 1 ngày,... Điều này giúp cho người tham gia có thể biết tỷ giá trong các phiên giao dịch trước. Tuy nhiên trong phiên giao dịch có thể gồm nhiều các giao dịch với giá, số lượng đồng khác nhau. Nhằm dễ thống kê, public api cung cấp để lấy giá OHLCV(Open, High, Low, Close, Volume) tương ứng với giá mở sàn; giá cao nhất, thấp nhất trong phiên; giá đóng sàn và lượng đồng trao đổi. Đây là các đặc trưng cơ bản cho một phiên giao dịch. Dữ liệu sẽ được lưu dạng bảng với mỗi dòng tương ứng với một phiên giao dịch gồm thời gian mở phiên theo dạng Unix time, và 5 đặc trưng nêu trên, các đặc trưng khác sẽ được trình bày tóm tắt như sau:

- Tổng số giao dịch mua; bán.
- Số lượng đồng mua; bán trung bình
- Độ lệch chuẩn số lượng đồng mua; bán.
- Giá trung bình của các giao dịch mua; bán và độ lệch chuẩn tương ứng.

Trong trường hợp một phiên giao dịch không có giao dịch mua nào bán hoặc không có giao dịch bán, tổng giao dịch sẽ bằng 1 và các trường trung bình, độ lệch chuẩn sẽ là 0. Trong thực nghiệm, hiếm khi xảy ra trường hợp này, cụ thể với khoảng thời gian từ 2017/08/17 đến 2019/09/01 có 16 phiên giao dịch như trên trong tổng số 17875 phiên giao dịch với khoảng thời gian phiên là 1 giờ cho cặp BTC/USDT.

Tiền xử lý dữ liệu

Sau công đoạn thu thập dữ liệu thô từ sàn, các phiên giao dịch được vectơ hóa và sắp xếp theo thời gian thành bảng được lưu dạng csv. Bước tiền xử lý dùng dữ liệu trên với các bước được tóm tắt theo hình sau:



Hình 3.1: Tiền xử lý dữ liệu

Thêm đặc trưng

Ngoài các đặc trưng được lấy trực tiếp từ sàn, Trong một phiên giao dịch với các giá High, Low chênh lệch nhau không rõ rệt khi thời gian phiên ngắn như 1 phút, 1 giờ; cần chọn thêm các đặc trưng như giá chênh lệch giữa High-Low và giá chênh lệch giữa Close-Open sẽ hiệu quả, đặc biệt khi chuẩn hóa z-score, min-max; hệ số chuẩn hóa được tăng lên giúp tăng độ lệch chuẩn đối với đặc trưng mới này khi so sánh các phiên giao dịch với nhau.

Sai phân bậc d

Khi sử dụng sai phân bậc d, Sử dụng 1d order difference, thể hiện sự chênh lệch giá hiện tại với giá trước.

Lựa chọn đặc trưng

Dữ liệu sau khi được thêm các đặc trưng mới, các đặc trưng có các mức tương đồng khác nhau, việc lựa chọn đặc trưng là điều cần thiết để loại bớt các đặc trưng tương quan với nhau.

Chuẩn hóa dữ liệu

Với dữ liệu dạng bảng mỗi dòng tương ứng một phiên giao dịch, các dòng được sắp xếp với thời gian tăng dần. Tập dữ liệu được chia thành tập huấn luyện và tập kiểm thử. Khi chuẩn hóa dữ liệu tập huấn luyện tạo ra hệ số chuẩn hóa, hệ số này sẽ chuẩn hóa tập dữ liệu kiểm thử với giả thiết khi có tập huấn luyện, một giao dịch mới sẽ được chuẩn hóa theo hệ số trước đây. Điều này có hạn chế khi chuẩn hóa theo min-max với khoảng $[0, 1]$ hoặc $[-1, 1]$ giá trị của các giao dịch trong tập kiểm thử có thể vượt ngoài 1, để tránh trường hợp này có thể xóa các dữ liệu bất thường này hoặc dùng phép chuẩn hóa khác như z-score:

$$Z_{scale} = \frac{Z - \mu}{\sigma}, \tag{3.1}$$

trong đó:

- Z là giá trị trước khi chuẩn hóa.
- μ, σ lần lượt là giá trị trung bình, độ lệch chuẩn trước khi hiệu chỉnh.
- Z_{scale} là giá trị sau khi chuẩn hóa.

Đánh nhãn dữ liệu

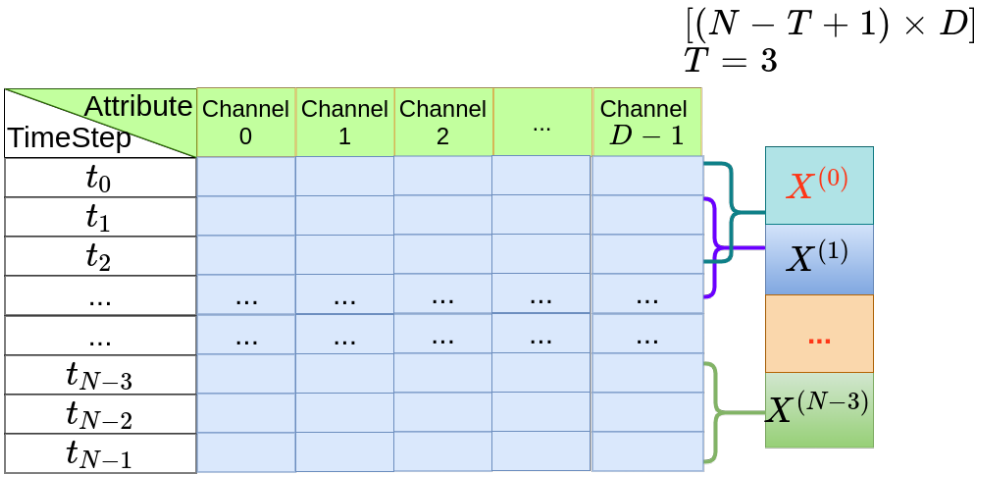
Nhãn được chia thành hai loại là xu hướng tăng và xu hướng giảm của giá đóng phiên thời điểm hiện tại. Thống kê với dữ liệu BTC/USDT thời gian 1 giờ có 9051 nhãn xu hướng tăng và 8452 nhãn xu hướng giảm, trường hợp giá không đổi tại phiên giao dịch sau là không có. Công việc đánh nhãn được thực hiện sau khi gộp các phiên giao dịch thành các điểm dữ liệu như hình sau:

Với các tham số:

- D: Số thuộc tính của mỗi phiên giao dịch
- T: Số phiên giao dịch liên tiếp (Sliding window)

Các điểm hạn chế của ba mô hình nêu trên

Trong ba mô hình kể trên có điểm chung là mỗi điểm dữ liệu được đưa vào trong quá trình huấn luyện là thông tin của một phiên giao dịch. Tuy nhiên với dữ liệu theo



Hình 3.2: Đánh nhãn dữ liệu

dạng thời gian thực, mỗi điểm dữ liệu cần chứa các thông tin phiên giao dịch hiện tại cũng như các phiên giao dịch trước đó. Để biểu diễn mối quan hệ giữa các phiên giao dịch liên tục nhau trên điểm dữ liệu, ta có thể thêm các thuộc tính mới như giá trị trung bình của 5 phiên giao dịch trước, độ chênh lệch của giá mở, giá đóng phiên. Tuy nhiên bước xử lý dữ liệu trước khi đưa vào chứa các tham số cố định. Một hướng làm khác khi gộp nhiều phiên giao dịch liên tục nhau thành một điểm dữ liệu dẫn tới bùng nổ số chiều dữ liệu. Vấn đề này dẫn chúng tôi tới ý tưởng thu giảm số chiều và 'học' tự động dữ liệu thông qua các bộ lọc (filters) thay cho bước tiền xử lý dữ liệu. Thông qua tìm hiểu, nhóm đã tham khảo thêm mô hình variational autoencoder. Chi tiết mô hình sẽ được trình bày tại phần tiếp theo.

Mô hình Variational autoencoder

Để giải thích mô hình Variational Auto Encoder, chúng tôi nêu ý tưởng của mô hình Auto Encoder, sau đó đi đến khái niệm mô hình Variational Auto Encoder và cách ứng dụng để làm công đoạn phân loại.

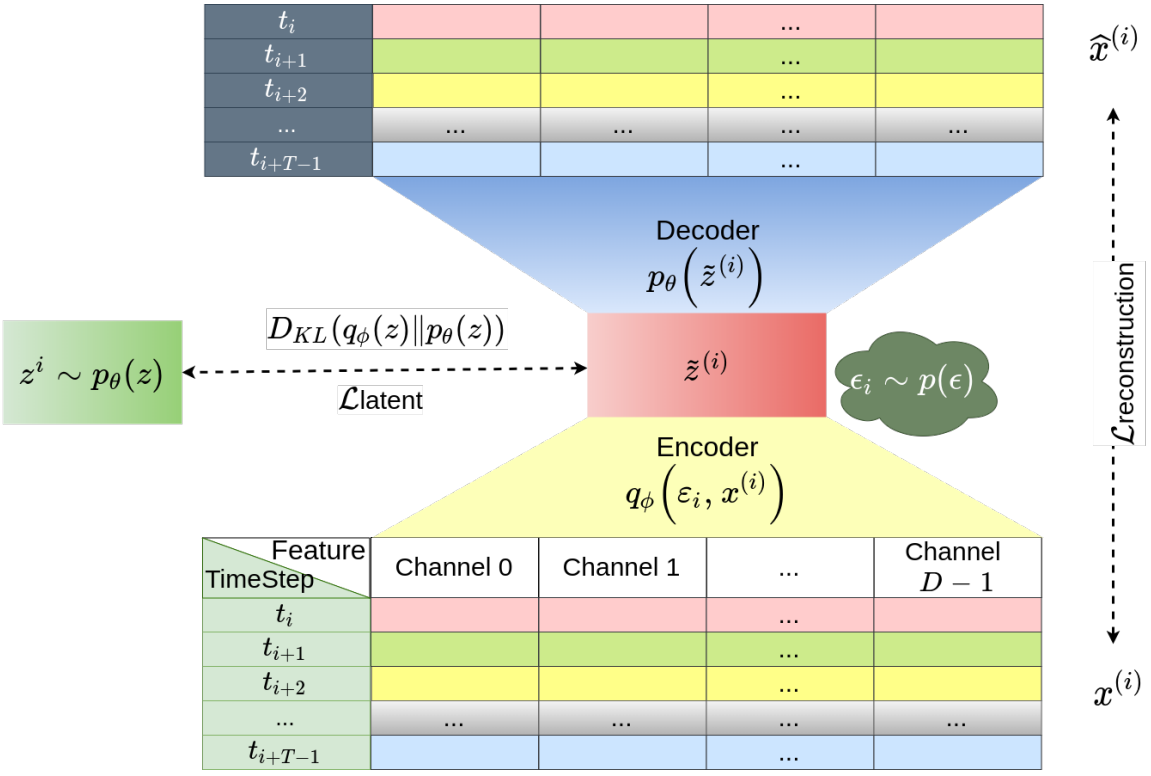
Mô hình Autoencoder

Kiến trúc của mô hình Auto Encoder gồm 2 mô-đun chính là Encoder, Decoder. Với ý tưởng chính là 'nén' xuống chiều không gian nhỏ hơn. Trên chiều không gian mới này (latent space), dữ liệu được thu giảm từ không gian phức tạp thành các phần đơn giản hơn (simple components). Để đảm bảo hơn cho quá trình nén ít bị mất thông tin (nói

cách khác trên chiều không gian nhỏ có thể khôi phục lại dữ liệu gốc) ta cần lớp giải nén (Decoder) với nhiệm vụ giải nén về dữ liệu ban đầu. Khi dữ liệu đi ra từ Encoder được giải nén qua Decoder trùng khớp với dữ liệu ban đầu, ta có thể hiểu rằng số bậc tự do (degrees of freedom) nhỏ hơn so với số chiều của dữ liệu gốc. Việc phân cụm dữ liệu trở nên rõ ràng hơn trên chiều mới và điều này được thí nghiệm trên nhiều loại dữ liệu ảnh[3].

Mô hình Variational autoencoder

Mô hình Variational autoencoder (VAE) sử dụng trong đề tài là một biến thể khác của mô hình Autoencoder với ràng buộc rằng biến ẩn được sinh theo một hàm phối tiên nghiệm (prior distribution). Việc dự đoán có đầu vào là biến ẩn được lấy mẫu (sampling) từ đầu ra của mạng Encoder. Kiến trúc của mô hình sử dụng được mô tả theo Hình 3.3.



Hình 3.3: Kiến trúc mô hình dựa trên Variational Auto Encoder

Để chi tiết hơn mô hình, chúng tôi sử dụng các kí hiệu cho các giả thiết sau:

- $X = \{x^{(i)}\}_{i=1}^{N}$ gồm N các điểm dữ liệu có phân phối đồng nhất độc lập (iid).

- Giá trị $z^{(i)}$ được sinh ra từ phân phối tiên nghiệm $p_{\theta^*}(z)$.
- Giá trị $x^{(i)}$ được sinh ra từ phân phối đồng thời $p_{\theta^*}(x \mid z)$.

Mục tiêu của bài toán là việc dự đoán cần phải chính xác đồng thời mô hình tìm được phân phối của dữ liệu đã cho theo phân phối tiên nghiệm. Trong phần thực nghiệm, chúng tôi có sử dụng mạng nơ-ron cho từng mô-đun vì lý do dễ dàng cập nhật trọng số khi có được hàm lỗi.

Hàm lỗi cho việc phân loại: $\mathscr{L}_C$ dựa được tính theo lỗi của hàm softmax. **Hàm lỗi cho việc tối đa likelihood theo phân phối tiên nghiệm:**

$$\mathscr{L}_{MLE} = -\frac{1}{N}\sum_{i=1}^{N} log(p_\theta(x^{(i)})) \tag{3.2}$$

Với mô-đun Encoder, Decoder và Classifier là hàm tất định, tuy nhiên vì lý do z được sinh mẫu ngẫu nhiên nên không thể xây dựng hàm lỗi một cách trực tiếp vì lý do khi khai triển $p_\theta(x)$:

$$p_\theta(x) = \int p_\theta(z)p_\theta(x \mid z)dz \tag{3.3}$$

với :

- $p_\theta(z)$ là phân phối tiên nghiệm có thể tính được.
- $p_\theta(x \mid z)$ là phân phối có thể tính được thông qua mô-đun Decoder.

Tuy nhiên không thể tính được (intractable) cho toàn bộ z trên miền liên tục. Khi khai triển phân phối hậu nghiệm: $p_\theta(z|x)$ đưa về dạng không thể phân giải được do $p_\theta(x)$:

$$p_\theta(z \mid x) = \frac{p_\theta(x \mid z)p_\theta(z)}{p_\theta(x)}. \tag{3.4}$$

Một cách giải quyết vấn đề trên là xấp xỉ hàm lỗi theo phương pháp suy luận biến phân được trình bày kĩ hơn tại Tiểu mục 4.7.1. Theo phương pháp này khi khai triển $\log p_\theta(x)$:

$$\log p_\theta(x^{(i)}) \geq \mathscr{L}(x^{(i)}, \theta, \phi) \tag{3.5}$$

Với:

$$\mathscr{L}(x^{(i)}, \theta, \phi) = \mathbb{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - \mathbb{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z)}\right] \tag{3.6}$$

Việc tối ưu $p_\theta(x^{(i)})$ thay bằng việc tối ưu cận dưới $\mathscr{L}(x^{(i)}, \theta, \phi)$. Lúc này hàm lỗi mới thay cho hàm $\mathscr{L}_{MLE}$ được tính như sau:

$$\mathscr{L}_{ELBO} = -\frac{1}{N}\sum_{i=1}^{N}\left(\mathbb{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - \mathbb{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z)}\right]\right) \tag{3.7}$$

$$= \mathscr{L}_{reconstruction} + \mathscr{L}_{latent} \tag{3.8}$$

Trở lại với kiến trúc VAE, phương trình (3.6) có thể mô tả như sau:

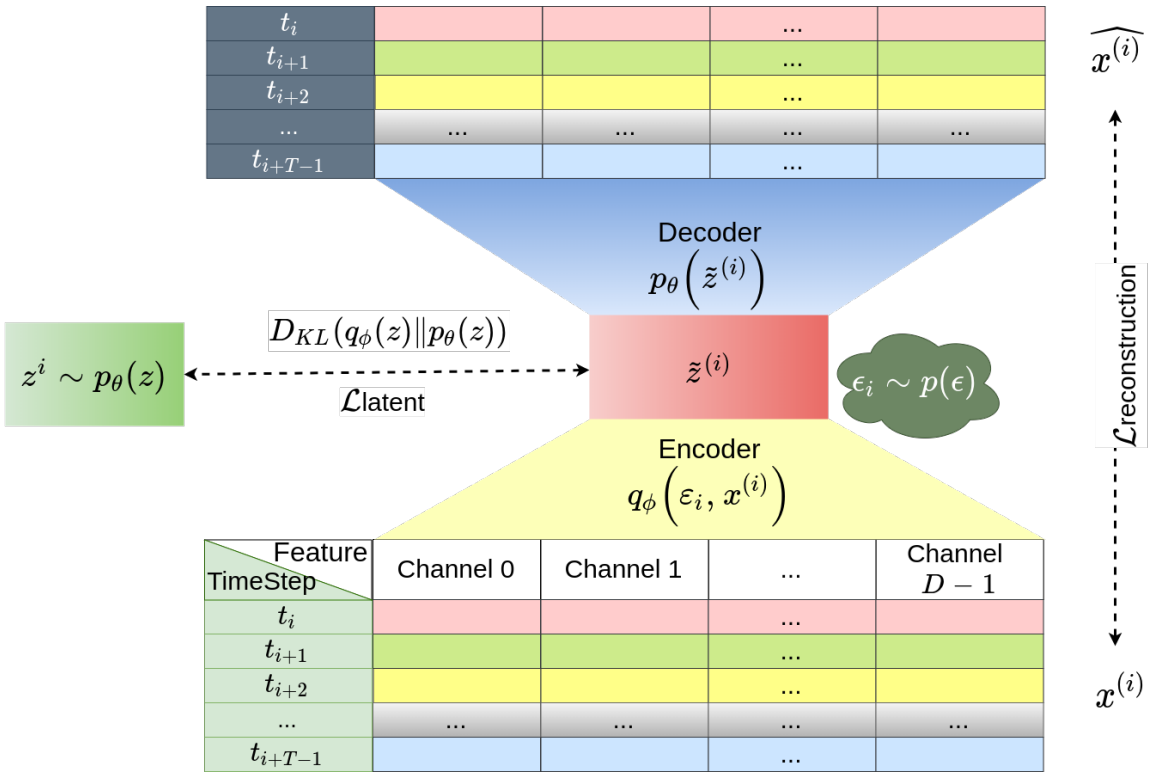
- $\mathbb{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right]$: được tính theo mô-đun Decoder khi tái tạo lại x từ z.
- $D_{KL}(q_\phi(z \mid x^{(i)}) \parallel p_\theta(z))$: Kullback–Leibler divergence giữa hai phân phối hậu nghiệm và tiên nghiệm.

$$\frac{\mathrm{d}b_1}{\mathrm{d}z} - \beta_1 b_1 = C_{12} b_2, \tag{3.9a}$$

$$\frac{\mathrm{d}b_2}{\mathrm{d}z} - \beta_2 b_2 = C_{21} b_1. \tag{3.9b}$$

(3.9)

Hai hàm lỗi được trực quan theo Hình 3.4 dưới đây:



Hình 3.4: Mô tả hàm lỗi trong mô hình VAE

Mục tiêu của bài toán là giảm được giá trị của hàm $\mathscr{L}_{ELBO}$. Trong trường hợp hội tụ cần đảm bảo phân phối hậu nghiệm $q_\phi(z \mid x^{(i)})$ có các tham số từ mô-đun Encoder được tiến về phân phối tiên nghiệm cho trước: $p_\theta(z)$, đồng thời với phân phối hậu nghiệm có thể tái tạo lại dữ liệu cho trước. Nói cách khác, từ quá trình huấn luyện, mô hình có thể 'học' được phân phối dữ liệu ban đầu từ phân phối tiên nghiệm đơn giản hơn được cho trước.

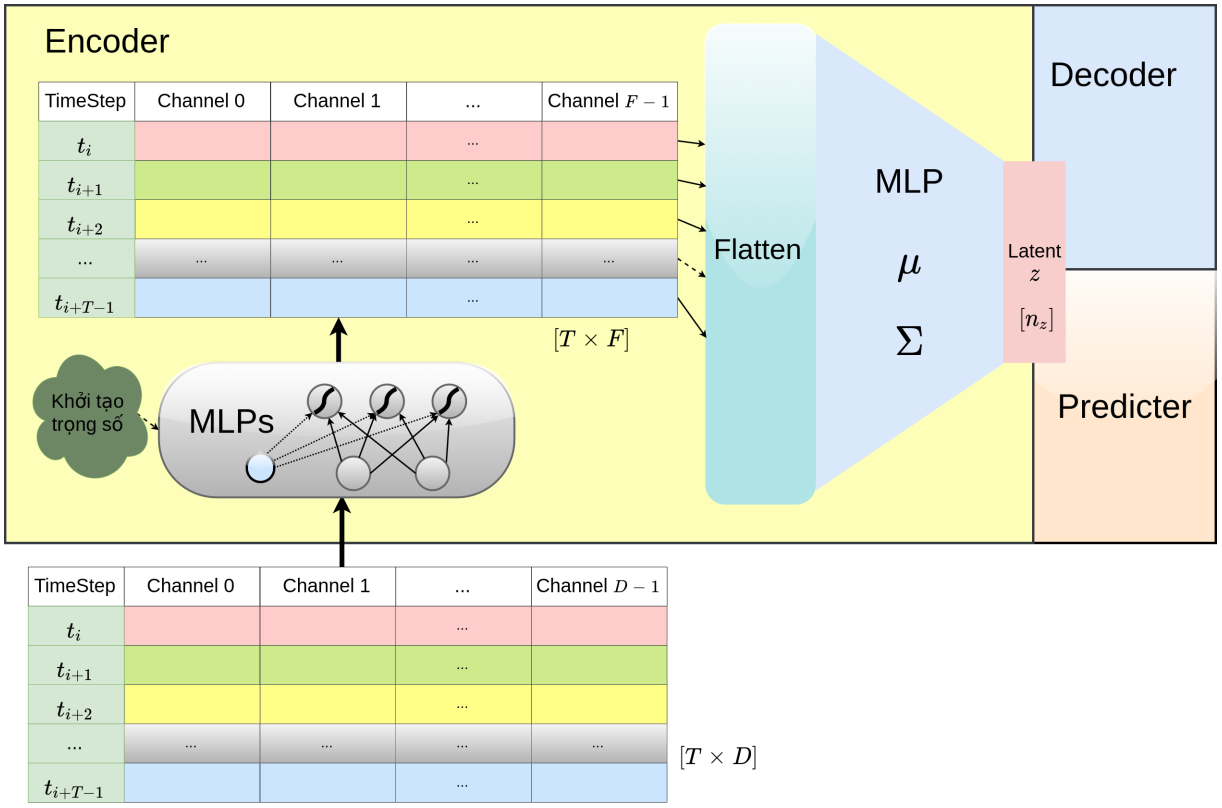
Trong phần hiện thực, chúng tôi có giả định biến ẩn (latent variable) có số chiều là n_z thuộc phân phối multivariate normal distribution với vector trung bình là $\mu = 0 \in \mathbb{R}^{n_z}$, và hiệp phương sai là $\Sigma = I \in \mathbb{S}^{n_z}_{++}$. Theo đó, $\mathscr{L}_{latent}$ được tính như sau:

$$\mathscr{L}_{latent} = D_{KL}(q_\phi(z \mid x^{(i)}) \parallel \mathcal{N}(0, I)) \tag{3.10}$$

$$= \frac{1}{2}\sum_{i=1}^{n_z}(\sigma_i^2 + \mu_i^2 - \log(\sigma_i^2) - 1) \geq 0 \tag{3.11}$$

Với dữ liệu thời gian, việc tính lỗi $\mathscr{L}_{reconstruction}$ không thể chuyển (scale) dữ liệu về khoảng $[0-1]$ vì trong phiên tiếp theo, các biến có giá trị vượt ngoài phạm vi $[0-1]$. Cụ thể, với 1 tháng trước tỷ giá trong phạm vi từ 2000 đến 5000, khi đưa dữ liệu tháng sau có tỷ giá là 5500 vượt ngoài khoảng trên. Vì lý do trên, chúng tôi lựa chọn hàm log mean square error để thể hiện lỗi khi tái tạo lại dữ liệu:

$$\mathscr{L}_{reconstruction} = \frac{1}{N}\sum_{i=1}^{N}\log((x^{(i)} - \hat{x}^{(i)})^2) \geq 0 \tag{3.12}$$



Hình 3.5: Kiến trúc mô-đun Encoder

Chương 4

Cơ sở lý thuyết

Tiền mã hóa

Khái niệm về tiền mã hóa

Tiền mã hóa là một dạng tiền tệ kỹ thuật số được tạo ra như một phương thức để trao đổi, sử dụng các phương pháp mã hóa để bảo vệ, xác minh các giao dịch cũng việc quản lý việc tạo ra các đơn vị tiền mã hóa trong hệ thống.

Tiền mã hóa sử dụng một hệ thống phân tán để quản lý thay vì một hệ thống xác thực trung tâm như các cách thức quản lý trước đây. Hệ thống phân tán này như một cuốn sổ cái phân tán, được gọi là Blockchain, đảm nhận vai trò lưu trữ các giao dịch một cách công khai đến những người tham gia. Quá trình xác minh giao dịch dựa trên sự đồng thuận phân tán. Quá trình này còn gọi là đào (mining).

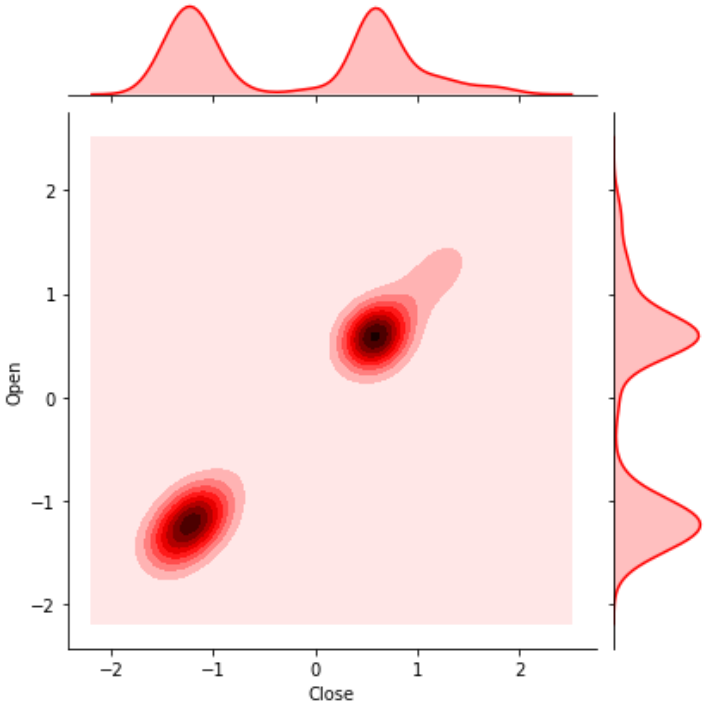
Nhiễu dữ liệu

Trong tài chính khái niệm nhiễu (noise) có quan hệ đối lập với khái niệm thông tin (information), với dữ liệu cung cấp đầy đủ thông tin, việc dự đoán dễ dàng và ngược lại với dữ liệu có nhiễu cao do bị ảnh hưởng bởi các yếu tố khác như đã đề cập tại Tiểu mục 6.3.1 do các lệnh mua bán trong tương lai không tuân theo các quy luật từ

giao dịch trước đó. Việc khử nhiễu dữ liệu là loại bỏ các yếu tố trên, nhằm cho việc tìm luật chung được dễ dàng hơn.

Hàm mật độ xác suất

Với các phiên giao dịch có các thành phần như giá mở, giá đóng, số lượng đồng giao dịch,... ta có thể coi như các biến ngẫu nhiên liên tục tương ứng. Khái niệm hàm mật độ xác suất (probability density function) trong văn cảnh trên được hiểu như một hàm gồm các tham số thể hiện được mật độ phân bố của các biến ngẫu nhiên.



Hình 4.1: Phân phối biên giá mở/đóng dữ liệu đã xử lý

Hình 4.1 thể hiện mật độ của phân phối đồng thời giữa giá đóng và giá mở của các khối nến được biểu diễn dưới dạng $p_{data}(Open, Close)$.

Hàm phân phối biên

Với dữ liệu liên tục như trên, hàm phân phối biên (marginal distribution) đối với giá mở được biểu diễn dưới dạng:

$$p_{data}(Open) = \int_y p_{data}(Open, Close = y)dy = \int_y p_{data}(Open \mid Close = y)p_{data}(Close = y)dy \tag{4.1}$$

Một cách trực quan, hàm phân phối trên được biểu diễn bởi đường biên bên trái Hình 4.1

Nhiễu trắng

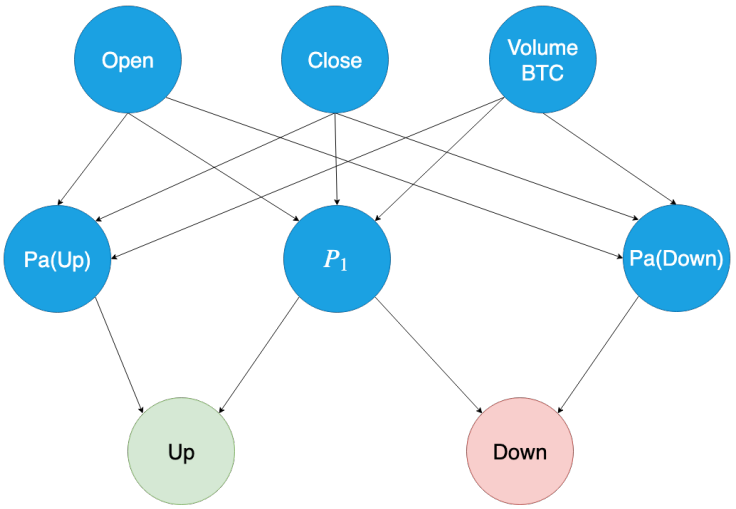
Khái niệm nhiễu có thể được giảm thiểu bằng cách tìm hàm phân phối của nhiễu bằng thống kê, nếu phân phối của nhiễu có dạng phân phối chuẩn với trung bình là 0, nhiễu này được gọi là nhiễu trắng Gauss (white Gaussian noise). Việc giảm thiểu nhiễu trong dữ liệu làm mô hình trở nên dễ tìm được mẫu đặc trưng (pattern) hơn.

Biến ẩn

Biến ẩn (Latent variable) được hiểu theo cách trừu tượng là biến không thể quan sát trực tiếp[4, trang 264] mà được suy luận từ biến quan sát được trong dữ liệu. Cụ thể hơn, với dữ liệu là giá của 10 ngày đầu một mô hình có khả năng tìm được quan hệ giữa giá ngày thứ 5 phụ thuộc nhiều vào giá ngày thứ 4 hơn so với ngày thứ 9, mô hình này được gọi là mô hình biến ẩn (latent variable model) với quan hệ được biểu diễn bằng phép toán có giá trị được lưu trong các biến ẩn.

Mô hình đồ thị có hướng

Trong mô hình đồ thị có hướng (directed graphical model) hay mạng Bayes (Bayes network) việc suy diễn từ các trạng thái trước sang các trạng thái sau. Cụ thể với mô



Hình 4.2: Mô hình mạng Bayes

hình được được trực quan theo như Hình 4.2, một giao dịch BTC/USD vào 6 giờ sáng 2017/7/31/ có giá mở là 2439.97$, giá đóng là 2415.19$, lượng giao dịch là 138.82 đồng BTC với xu hướng giao dịch tiếp theo có xác suất được kí hiệu là:

$$P(Up \mid Open = 2727.26, Close = 2740.01, VolumeBTC = 385.41)$$

với xác suất đồng thời của giao dịch và được tính:

$$\begin{aligned} & p(Up, Open = 2727.26, Close = 2740.01, VolumeBTC = 385.41) \\ = & \, p(Open = 2727.26) \cdot p(Close = 2740.01) \cdot p(VolumeBTC = 385.41) \\ \cdot & p(Pa(Up) \mid Open = 2727.26, Close = 2740.01, VolumeBTC = 385.41) \\ \cdot & p(P_1 \mid Open = 2727.26, Close = 2740.01, VolumeBTC = 385.41) \\ \cdot & p(Up \mid Pa(Up), P_1) \end{aligned}$$

Một cách tổng quát xác suất đồng thời của giao dịch và xu hướng tăng giảm về giá của giao dịch tiếp theo được biểu diễn dưới dạng:

$$p_\theta(x_1, x_2, \ldots, x_M) = \prod_{i=1}^{M} p_\theta(x_j, Pa(x_j))$$

với $Pa(x_j)$ giá trị của nút mạng trước đó (parent variable) của x_j.

Phương pháp cận dưới biến phân

Để giải quyết hàm bất trị (intractable function) trong Tiểu mục 3.3, ta có thể dùng phương pháp cận dưới biến phân [5, trang 198] (Variational lower bound). Chi tiết

phương pháp đi từ việc xấp xỉ likelihood:

$$\log p_\theta(x^{(i)}) = \mathbb{E}_z\left[\log p_\theta(x^{(i)}\right] \tag{4.2}$$

$$= \mathbb{E}_z\left[\log \frac{p_\theta(x^{(i)} \mid z)p_\theta(z)}{p_\theta(z \mid x^{(i)})}\right] \tag{4.3}$$

$$= \mathbb{E}_z\left[\log \frac{p_\theta(x^{(i)} \mid z)p_\theta(z)}{p_\theta(z \mid x^{(i)})}\frac{q_\phi(z \mid x^{(i)})}{q_\phi(z \mid x^{(i)})}\right] \tag{4.4}$$

$$= \mathbb{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - \mathbb{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z)}\right] + \mathbb{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\phi(z \mid x^{(i)})}\right] \tag{4.5}$$

$$= \mathscr{L}(x^{(i)}, \theta, \phi) + D_{KL}(q_\phi(z \mid x^{(i)}) \parallel p_\phi(z \mid x^{(i)})) \tag{4.6}$$

Với $D_{KL}(q_\phi(z|x^{(i)}) \parallel p_\phi(z|x^{(i)}))$ còn được gọi là độ đo bất đồng Kullback–Leibler nhằm thể hiện sự khác nhau giữa hai phân phối $q_\phi(z|x^{(i)})$ và $p_\phi(z|x^{(i)})$. Một cách tổng quát độ đo bất đồng Kullback–Leibler giữa phân phối Q so với phân phối P mang giá trị không âm và được tính như sau:

$$D_{KL}(Q \parallel P) \triangleq \mathbb{E}_{Q(Z)}\left[\log \frac{Q(Z)}{P(Z \mid X)}\right] \geq 0 \tag{4.7}$$

Sử dụng luật Bayes:

$$D_{KL}(P \parallel Q) = \mathbb{E}_{Q(Z)}\left[\log \frac{Q(z)}{P(z,x)} + \log P(x)\right] \tag{4.8}$$

hay:

$$\log(P(x)) = D_{KL}(P \parallel Q) - \mathbb{E}_{Q(z)}\left[\log \frac{Q(z)}{P(z,x)}\right] = D_{KL}(P \parallel Q) + \mathscr{L}(Q) \tag{4.9}$$

Chương 5

Hiện thực hệ thống

Với lộ trình nghiên cứu đã đề ra trong chương 3, tiếp theo đây, chúng tôi hiện thực việc thu thập dữ liệu. Sau đó, chúng tôi xử lý dữ liệu trước khi đưa vào mô hình học máy.

Thu thập dữ liệu

Để chi tiết hơn cho phần 3.3, chúng tôi sẽ mô tả việc thu thập bằng công cụ CCXT (CryptoCurrency eXchange Trading Library) trên nền python. Một giao dịch có giá trị cụ thể như sau:

```
{
 "timestamp": 1569758400471,
 "datetime": "2019-09-29T12:00:00.471Z",
 "symbol": "BTC/USDT",
 "id": 166647503,
 "order": None,
 "type": None,
 "takerOrMaker": None,
 "side": "sell",
 "price": 8073.86,
 "amount": 0.028108,
```

```
 "cost": 226.94005688,
 "fee": None
}
```

Với các giáel trị:

- timestamp, datetime: thời gian dạng Unix và thời gian thực với GMT +0
- symbol: tên của loại cặp đồng.
- id: mã số định danh cho phiên giao dịch.
- side: "sell" (bên bán), trường hợp còn lại là "buy" (bên mua).
- price: giá đặt bán với tỷ giá *BTC/USDT* là 8073.86.
- amount: lượng đồng BTC bán ra.
- cost: số đồng USDT nhận được.
- fee: không có khoản phí phải trả cho giao dịch.

Trong khoảng thời gian được định sẵn là một giờ, ta có thể thống kê lại được số lượng đồng mua, bán theo đoạn mã sau:

```
t = exchange.fetch_trades(symbol=symbol, since=from_timestamp)
buy_amount_list = []
sell_amount_list = []
price_list = []
cost_list = []
for trade in t:
    if trade.get('side') == 'buy':
        buy_amount_list.append(trade.get('amount'))

    elif trade.get('side') == 'sell':
        sell_amount_list.append(trade.get('amount'))

    price_list.append(trade.get('price'))
    cost_list.append(trade.get('cost'))
```

Tiền xử lý dữ liệu

Sau quá trình thu thập dữ liệu, dữ liệu thô được lưu dạng csv tiếp đến được xử lý bằng cách thêm cột và chuẩn hóa.

Đánh nhãn dữ liệu

Hiện thực các mô hình đã tham khảo

Các thư viện sử dụng trong mô hình

Chúng tôi sử dụng ngôn ngữ lập trình Python phiên bản 3.6 để tiến hành thí nghiệm, các thư viện sử dụng được viết trên ngôn ngữ này, đồng thời là mã nguồn mở được sử dụng rộng rãi như: Scikit-learn, NumPy, Pandas, TensorFlow và một số thư viện khác.

***Scikit-learn* 0.19.1**: là một thư viện mã nguồn mở được viết bằng ngôn ngữ lập trình Python. Thư viện này hiện thực hầu hết các mô hình học máy hiện tại bao gồm cả học có giám sát và học không có giám sát. Thư viện cũng cung cấp các công cụ cho quá trình đọc dữ liệu, tiền xử lý, trích xuất đặc trưng và có sẵn nhiều bộ dữ liệu mẫu cho các ví dụ. Đây là một thư viện dễ sử dụng, hiệu năng tốt cho làm việc nghiên cứu.

***Numpy* 1.15.3**: là một gói cơ bản cho các tính toán khoa học sử dụng Python. Thư viện này cung cấp các công cụ toán các hàm liên quan đến đại số tuyến tính,... Các tính toán trong Numpy đều đã được tối ưu để xử lý song song, tăng hiệu năng tính toán và tích hợp với cả các ngôn ngữ và hệ cơ sở dữ liệu khác.

***Pandas* 0.21.0**: để xử lý các tập tin dữ liệu dưới dạng dataframe (gồm tập huấn luyện và tập kiểm tra)

***TensorFlow* 1.12.1**: là một thư viện mã nguồn mở cung cấp khả năng xử lí tính toán số học dựa trên biểu đồ mô tả sự thay đổi của dữ liệu, trong đó các nút (node) là các phép tính toán học còn các cạnh biểu thị luồng dữ liệu. Ngoài ra, trong mô hình sử dụng thư viện Tensorpack 0.9.5 dựa trên nền thư viện TensorFlow nhằm tối ưu tốc độ xử lý.

Trong ba mô hình đầu gồm Rừng ngẫu nhiên; SVM; Hồi quy Logistic được thí nghiệm

bằng thư viện Scikit-learn. Mô hình cuối dựa trên VAE sẽ được hiện thực bằng thư viện Tensorpack nhằm tối ưu hai luồng chạy: xáo trộn dữ liệu (shuffle) và huấn luyện mạng nơ-ron.

Chương 6

Thí nghiệm và đánh giá

Trong hai chương trước có đề cập tới các mô hình tham khảo và cách thu thập, xử lý dữ liệu để chuẩn bị cho quá trình thí nghiệm. Tiếp sau đây, trong chương này sẽ trình bày thư viện sử dụng, mô tả tham số trong mô hình. Dựa trên kết quả các mô hình từ đó rút ra các nhận xét về mô hình sử dụng và các điểm hạn chế khi hiện thực đồng thời đưa ra các hướng phát triển của luận văn.

Các độ đo được sử dụng

Trong phần này, chúng tôi trình bày các độ đo được sử dụng để đánh giá các mô hình. Trước hết, chúng tôi trình bày ma trận nhầm lẫn (confusion matrix) cho các nhãn dự đoán như trong Bảng

		Kết quả dự đoán	
		Giá tăng	**Giá giảm**
Nhãn thực tế	**Giá tăng**	True Positive (TP)	False Positive (FP)
	Giá giảm	False Negative (FN)	True Negative (TN)

Bảng 6.1: Ma trận nhầm lẫn cho các nhãn dữ liệu

Kết quả

So sánh kết quả các mô hình đề xuất

Chúng tôi so sánh kết quả dự đoán của bốn mô hình được sử dụng trong luận văn trên tập dữ liệu giao dịch từ ngày 2017/08/17 gồm hai khung thời gian phiên là 1 giờ và 5 phút cho ra các kết quả trong bảng:

Mô hình	**Độ chính xác**	**f1 score**
Rừng ngẫu nhiên	**rf**	**rf**
SVM	**svm**	**svm**
Hồi quy Logistic	**lr**	**lr**
VAE	**vae**	**vae**

Bảng 6.2: Kết quả dự đoán trên 1 giờ (đơn vị: %)

Mô hình	**Độ chính xác**	**f1 score**
Rừng ngẫu nhiên	61.88	**rf**
SVM	60.67	svm
Hồi quy Logistic	64.24	**lr**
VAE	**vae**	**vae**

Bảng 6.3: Kết quả dự đoán trên 5 phút (đơn vị: %)

Phụ lục

Một số thuật ngữ được sử dụng

Block: là một cấu trúc dữ liệu chứa dữ liệu của giao dịch.

Blockchain: một cuốn sổ cái công khai chứa thông tin của các giao dịch đã được thực hiện trên hệ thống. Nó bao gồm một chuỗi các block theo trình tự thời gian. Các block bao gồm các giao dịch và thông tin từ các khối trước đó. Một đường dẫn duy nhất từ block đầu tiên đến khối hiện tại được sử dụng và mỗi block sẽ bao gồm mã băm của block trước đó.

Mining: bước xác thực cần phải có cho mỗi giao dịch tiền mã hóa và để thêm các bản ghi vào Blockchain.

Mã băm: một hàm một chiều lấy dữ liệu đầu vào có kích thước bất kỳ và tạo ra kết quả có độ dài cố định. Việc tính toán mã băm phải nhanh và dễ dàng, việc đảo ngược mã phải khó và tốn nhiều thời gian, chi phí đồng thời phải tránh tối đa đầu vào khác nhau cho ra kết quả giống nhau (tránh đụng độ).

Peer-to-peer: kiểu thiết kế hệ thống kết nối trực tiếp người dùng với người dùng mà không thông qua một hệ thống quản lý trung gian nào. Đây là một hệ thống có kiến trúc phân tán.

Public key và Private key: là một cách mã hóa sử dụng một cặp khóa. Mỗi người dùng có một cặp mã khóa Public key và Private key này. Public key là khóa công khai mà mọi người ai cũng đều biết đến, xem như một cách để mọi người xác định một

người cụ thể nào đó. Private key là khóa bí mật, chỉ chủ sở hữu mới biết mã bí mật của mình và không được tiết lộ cho người khác. Cơ chế hoạt động của cách mã hóa này là giao dịch sẽ được gửi và mã hóa với khóa công khai của người nhận. Chỉ người nhận có khóa bí mật tương ứng với khóa công khai này có thể giải mã được giao dịch trên.

Double-spending: lỗi phát sinh khi thực hiện các giao dịch bằng hệ thống điện tử. Lỗi này xuất hiện khi một khoản tiền được dùng để chi tiêu cho hai hay nhiều việc cùng lúc, khi đó hệ thống chưa giải quyết xong giao dịch trước, chưa trừ đi số tiền giao dịch thì đã phát sinh giao dịch sau.

Những yếu tố tác động đến giá trị đồng tiền mã hóa

Cung, cầu và nhiễu trong thị trường

Trong nguyên tắc chính của kinh tế nếu nhu cầu mua đối với một đồng tiền tăng, giá trị của đồng tiền sẽ tăng và ngược lại khi nhu cầu bán tăng, giá sẽ giảm. Cụ thể khi thị trường tuân theo một quy luật chung (pattern) giá 3 ngày của cặp đồng A/B liên tục tăng và đến ngày thứ tư giảm theo chu kì. Khi một nhà đầu tư H phát hiện được tính chất này, đến ngày thứ tư trong chu kì sẽ thực hiện bán đồng A để kiếm lợi khiến giá thị trường bị điều chỉnh lại và có xu hướng tăng tại ngày thứ tư, nói cách khác thị trường sẽ thay đổi quy luật (pattern) trên bởi lệnh bán của người H, hay lệnh bán này gây ra nhiễu cho thị trường đồng A/B.

Tin tức trên các phương tiện thông tin đại chúng

Các sự kiện chính trị và kinh tế trên toàn thế giới ảnh hưởng đến cách mà con người phản ứng với các dự đoán giá, tin tức cảnh báo về rủi ro tác động chính lên cung-cầu.

Quy định của chính phủ

Có 4 cấp độ quản lý tiền ảo hiện nay đang được các nước áp dụng, cụ thể như sau:

- Cấm trên diện rộng:
- Cấm trong lĩnh vực tài chính ngân hàng (trong đó có Trung Quốc, Nga).

Chính sách của các tổ chức

Facebook, Google và Twitter đã ngăn chặn khách hàng và người dùng sử dụng dịch vụ cryptocurrency.

Các vấn đề kỹ thuật

Khi đồng tiền mật mã bị khai thác bằng các lỗ hổng từ Rủi ro khi sử dụng đồng tiền mã hóa Khi tài khoản mã hóa bị tấn công bằng kĩ thuật

Tính thanh khoản

Khái niệm về tính thanh khoản (Liquidity) dùng để chỉ mức độ mà một tài sản có thể được mua hoặc bán trên thị trường mà không làm ảnh hưởng nhiều đến giá thị trường. Khái niệm tính thanh khoản được chia thành 2 loại: tính thanh khoản thị trường (liquid market) và tính thanh khoản về tài sản (liquid asset). Thị trường có tính thanh khoản cao đồng nghĩa với việc trong thị trường thường xuyên có các nhà đầu tư sẵn sàng giao dịch. Một tài sản có tính thanh khoản cao mang nghĩa rằng tài sản đó có thể chuyển đổi sang tiền mặt một cách dễ dàng. Đối với thị trường tiền mã hóa, để so sánh tính thanh khoản giữa các sàn trong cùng một thời điểm hoặc tính thanh khoản của một sàn tại những thời điểm khác nhau có 3 yếu tố quan trọng:

- Lượng đồng giao dịch trong ngày.

- Số lượng lệnh mua/bán dựa trên danh sách lệnh (order book) được công khai dựa theo các sàn như Coinbase Pro, Binance, Bittrex [1], ...
- Lượng chênh lệch giữa giá yêu cầu của bên bán và giá đặt của bên mua (bid/ask spread).

Nhu cầu sử dụng tiền mã hoá của mỗi hệ sinh thái

- Số thành viên tham gia vào hệ sinh thái (Số người đến khu vui chơi mua vé tham gia các trò chơi trong đó bằng tiền A).
- Số lượng dịch vụ trong hệ sinh thái (Khu vui chơi có càng nhiều trò chơi thì nhu cầu sử dụng tiền A càng tăng); Và các nền tảng như Ethereum luôn mở cho các đối tác tạo các dịch vụ gia tăng trên đó giống như khu vui chơi cho phép đối tác bên ngoài vào tổ chức trò chơi ở trong.
- Số người đầu cơ: Những người nhận thấy nhu cầu tiền mã hoá của một hệ sinh thái tăng dần sẽ mua để nắm giữ chờ tăng giá thì bán ra. (Giống như phe vé bóng đá ngày trước mua vé chờ sát trận nhu cầu tăng vọt thì bán ra. Khu vui chơi thì ít có nhóm này vì lượng vé không bị giới hạn).
- Số người bán bên ngoài chấp nhận tiền mã hoá: Một số người bán nhận thấy tính thanh khoản của tiền mã hoá và giá trị tăng dần của nó nên đã chấp nhận khách hàng thanh toán các hàng hoá dịch vụ của mình bằng loại tiền này (Nhà hàng bên cạnh khu vui chơi có thể chấp nhận khách hàng thanh toán bằng tiền A).

Thị trường luôn bị biến động do ảnh hưởng của các yếu tố. Trong chương này, chúng tôi sẽ đề cập tới những yếu tố cơ bản tác động lên đồng tiền mã hóa. Tiếp theo sau đó chúng tôi sẽ trình bày các chiến lược kèm theo ưu, nhược điểm khi trao đổi ngắn hạn trên đồng mã hóa.

[1] `https://www.cryptometer.io/data/coinbase_pro/btc/usd` truy cập vào ngày 2019/04/18

Giao dịch tiền mã hóa

Các sàn tiền mã hóa cung cấp các lệnh giao dịch cơ bản: mua, bán với những cặp đồng (pair) với nhau, ngoài ra còn có thêm phương thức mua, bán tiền ảo bằng tiền mặt thông qua sàn đóng vai trò như bên thứ ba.

Để đảm bảo các đồng mã hóa có giá trị, cần một đồng có giá trị ổn định (stable coin) có giá trị cố định, với 1 USDT có giá trị ngang với 1 USD. Một đồng ổn định kể trên cần có những hai tính chất: Đáng tin cậy (có một tập đoàn có tài sản tương ứng đứng ra đảm bảo số đồng không bị lạm phát) và không bị thao túng (có giải thuật để kiểm soát số lượng đồng dựa trên tài sản thế chấp hoặc các đồng tiền mã hóa có liên quan). Các đồng ổn định hiện nay gồm: TrueUSD (TUSD), USD Tether (USDT), USD Coin (USDC), Digix Gold Tokens (DGX), trong đó đồng USDT có tổng giá trị lưu thông đạt tới 4.4 tỉ USD[2], cao nhất trong các đồng ổn định.

Các mô hình và chiến lược trong luận văn được đánh giá dựa trên tổng giá trị của các đồng mã hóa tại một thời điểm được quy về USDT.

Các chiến lược giao dịch ngắn hạn

Hai chiến lược cơ bản được nghiên cứu và thực hiện trong đề tài như sau:

- Giao dịch cùng một loại cặp với nhau tại hai thời điểm khác nhau nhằm tăng số lượng đồng ban đầu.
- Giao dịch nhiều cặp với nhau theo một vòng dựa theo giá tại cùng một thời điểm. Hai chiến lược trên sẽ được trình bày chi tiết hơn trong phần tiếp theo. Tiếp sau đó, phần 6.7 sẽ trình bày rủi ro và tiềm năng của thị trường, từ đó làm nỗi bật các hạn chế, ưu điểm của từng chiến lược tương ứng.

[2] `https://stablecoinindex.com` truy cập vào ngày 2019/10/17

Với dữ liệu được lấy từ sàn, có thể tạo một đánh giá thị trường với giao dịch ngắn hạn có tiềm năng hay không? Từ đó có thể tạo ra công cụ với khả năng dự đoán để tự động giao dịch không? Nhằm trả lời cho hai câu hỏi trên, trong chương này sẽ đề cập tới hai phần chính:

- Các chiến lược cơ bản được đề ra và mô phỏng hai chiến lược trên dữ liệu đã có. Ứng dụng các mô hình học máy để dự đoán xu hướng giá.
- Đánh giá rủi ro của hai chiến lược thông qua giá có trước, từ đó nhận định tiềm năng của thị trường.

Chiến lược giao dịch cùng một loại cặp đồng

Khi giao dịch cùng một loại cặp đồng A/B theo thời gian khác nhau, đặt lệnh mua hay đổi đồng B để mua A khi tỷ giá A/B có xu hướng giảm ngược lại đặt lệnh bán khi tỷ giá có xu hướng tăng. Chiến lược này sẽ không hiệu quả khi giá ở mỗi phiên không chênh lệch nhau nhiều đặc biệt có trường hợp lỗ khi mỗi lần giao dịch sẽ mất tiền phí do bên sàn thu. Vì vậy chiến lược nói trên sẽ được thêm một ràng buộc là ngưỡng phí giao dịch ϵ và các biến:

- W_t^a: số đồng A quy ra B theo giá tại thời điểm t.
- W_t^b: số đồng B quy ra A theo giá tại thời điểm t.
- y_t: tỷ giá đồng A/B tại thời điểm t.
- a, b: số đồng A, số đồng B trong ví tại thời điểm đang xét.

Với t là thời điểm gần nhất giao dịch, xét tại thời điểm τ xảy ra sau đó, Việc đặt lệnh mua phải thỏa yêu cầu sau: $W_\tau^a > W_t^a$ hay:

$$\frac{b}{y_\tau}(1-\epsilon) > \frac{b}{y_t} \tag{6.1}$$

do đó: $y_\tau < y_t(1-\epsilon)$

Tương tự, việc đặt lệnh bán phải thỏa yêu cầu sau: $W_\tau^b > W_t^b$ hay:

$$a(1-\epsilon)y_\tau > ay \tag{6.2}$$

hay $y_\tau(1-\epsilon) > y_t$

Việc mô phỏng chiến lược này cần tuân theo ràng buộc của sàn như sau: đơn vị tối thiểu là 0.000001 BTC và 0.01 USDT ví dụ muốn mua cặp đồng trên khi có 1.234 USDT với số lượng tối đa phí giao dịch sẽ là 0.1% với giá khớp lệnh là 8000 số lượng USDT còn lại trong ví là 0.004 số lượng giao dịch sẽ là 1.23 USDT, số lượng BTC nhận vào ví là $1.23/8000 * (1-0.1/100) = 0.00015359625$.

Chiến lược giao dịch nhiều loại cặp đồng

Với 3 đồng là A, B, C, việc chuyển đồng A chuyển sang đồng B, chuyển đồng B sang đồng C và cuối cùng chuyển lại đồng C sang đồng A tạo thành một vòng lặp, việc đồng A tăng lên hoặc giảm đi có thể xảy ra. Trong cùng một phiên giao dịch, việc tìm vòng lặp như trên sao cho số lượng đồng A được tăng lên so với trước đòi hỏi các lần chuyển đổi giữa các cặp diễn ra liên tục và có thứ tự nói cách khác tất cả các lần giao dịch đều phải được hoàn thành, đây cũng là nhược điểm của chiến lược này vì trong khi biết giá của các giao dịch trước, giá của các cặp sẽ đổi khi thực hiện giao dịch đòi hỏi giao dịch phải diễn ra nhanh. Lấy ví dụ ở thời điểm lúc 8 giờ ngày 04/01/2018 xét 3 cặp đồng là BTC/USDT, ETH/BTC, ETH/USDT có tỉ giá tương ứng là 15172.12, 0.060893, 920.08 với phí giao dịch cho mỗi lần trao đổi mặc định là 0.1% đối với sàn Binance, với 1.0 USDT lần lượt đổi các cặp là USDT sang ETH, ETH sang BTC và BTC sang USDT số đồng USDT thu về trong ví là 1.0011 với giả thiết ở mỗi giao dịch đều đổi hết (bỏ qua ràng buộc về số đồng tối thiểu). Trong ví dụ này với 3 đồng trên ta có thể thấy một cách trực quan rằng số đồng USDT tăng sau một vòng chuyển đổi. Việc tìm các vòng chuyển đổi tại mỗi thời điểm như trên có thể được mô hình hóa bằng bài toán như sau:

Data: T phiên giao dịch; $\frac{N(N-1)}{2}$ cặp tương ứng với N đồng

Result: Số lần xuất hiện vòng có xu hướng làm tăng số lượng đồng ban đầu

Khởi tạo:

- Đồ thị hai phía đầy đủ.
- Tensor M kích thước $TxNxN$ chứa thông số của T phiên giao dịch.
- Phí giao dịch ϵ.
- $t = 0$.

for *t trong T* **do**

Cập nhật trọng số của đồ thị

Khi không có giao dịch giữa hai đồng A/B trọng số cạnh $d(A,B) \leftarrow 1e-20$;

$d(B,A) \leftarrow 1e-20$

for *u, v trong* M_t **do**

$d(u,v) \leftarrow -log(d(u,v)) - log(\epsilon))$ ▷ Chuyển sang giá trị logarit

end

if *Đồ thị có trọng số âm* **then**

$t \leftarrow t+1$

Tìm chu trình nhỏ nhất không lặp.

end

end

Thuật toán 1: Tìm số lượng phiên giao dịch có thể làm tăng số đồng ban đầu.

Thống kê với phí giao dịch mặc định là 0.1% đối với sàn Binance từ 2018-01-01 đến 2019-09-21 gồm 15000 phiên giao dịch với khoảng thời gian mỗi phiên là 1 giờ xét trên 3 đồng BTC, ETH, USDT được thu thập từ 3 cặp: USDT/ETH, ETH/BTC, BTC/USDT đưa ra 75 lần đồ thị tồn tại chu trình âm. Khi thêm đồng BNB đồ thị với 4 loại đồng gồm 6 cặp, con số này đạt 1027 lần.

Rủi ro và tiềm năng của thị trường

Chiến lược giao dịch trên một cặp đồng

Khi thực hiện khảo sát chiến lược trao đổi trên một cặp, với dữ liệu thu được trên sàn Binance với cặp đồng BTC/USDT trong khoảng thời gian từ 2017-08-17 đến 2019-09-01, giả sử số tiền trong ví ban đầu là 1.0 BTC các ngưỡng phí ϵ được thay đổi cho ra kết quả được thống kê trong bảng sau:

Thời gian bắt đầu	Ngưỡng phí (%)	Số lần giao dịch	Tổng USDT đầu	Tổng USDT sau
2017/08/28 13:00:00	0.1	129	4221.04	2193.22
2017/08/28 13:00:00	0.2	129	4221.04	2290.84
2017/08/28 13:00:00	5	17	4221.04	6632.89
2017/08/28 13:00:00	10	7	4221.04	6133.60
2017/12/11 01:00:00	0.1	80	14975.03	9721.09
2017/12/11 01:00:00	0.2	82	14975.03	9884.65
2017/12/11 01:00:00	5	22	14975.03	17373.62
2017/12/11 01:00:00	10	6	14975.03	13452.00

Bảng 6.4: Mô phỏng chiến lược giao dịch một cặp đồng theo thời gian

Hạn chế dễ thấy của chiến lược này là không biết trước giá của phiên giao dịch tiếp theo. Cụ thể với ngày 2017/12/15 giá đạt ngưỡng cao nhất khi đó theo chiến lược, đổi hết đồng BTC sang thành USDT, tiếp theo giá giảm đều và qua ngưỡng phí và tiếp tục giảm, khi này số đồng USDT đã được chuyển sang BTC số lượng đồng BTC so với thời điểm trước khi bán ban đầu là nhiều hơn, khi giá tiếp tục giảm lệnh mua sẽ không được thực hiện do đã hết đồng USDT phải chờ đến khi giá tăng so với lần mua tại ngày 2018/16/03. Điều này dẫn tới việc tính theo giá USDT tổng giá trị BTC là giảm từ 14975.03 USDT xuống 13452 USDT. Để giảm rủi ro này, ta có thể dự đoán

xu hướng giá đóng của phiên giao dịch kế tiếp , nếu giá có xu hướng giảm, lệnh mua sẽ được giữ lại tới khi giá có xu hướng tăng. Đây chính là ý tưởng chính cho việc hình thành bài toán dự đoán xu hướng giá ngắn hạn, các mô hình sẽ được học từ các phiên giao dịch trước và dự đoán xu hướng giá của phiên giao dịch sau. Việc đánh nhãn cho dữ liệu sẽ được trình bày trong phần 3.5.

Chiến lược giao dịch trên nhiều cặp đồng

Với chiến lược đổi trên nhiều đồng, khi tăng số lượng đồng, việc giao dịch theo chiến lược này trở lên khó khăn hơn vì khi duyệt chu trình, tất cả các cạnh đều phải đi qua, nói cách khác các lần đổi đều phải hoàn thành. Tuy nhiên trong quá trình đổi, giá của hai đồng sẽ không giữ nguyên như giá hiện tại, việc khớp giá sẽ khó xảy ra. Do đó, việc đặt lệnh các đồng nên được hiện thực cùng lúc. Ngoài ra, bổ sung mô hình dự đoán xu hướng giá của phiên giao dịch tiếp theo có thể hỗ trợ thêm cho chiến lược này để tính khả năng đồ có chu trình âm có thể được duyệt tại phiên sau.

Tài liệu tham khảo

[1] A. Z. Isaac Madan, Shaurya Salu, "Automated bitcoin trading via machine learning algorithms."

[2] B. A. Alex Greaves, "Using the bitcoin transaction graph to predict the price of bitcoin."

[3] D. D. J. M. H.-L. B. S.-L. D. S. J. A.-I. T. D. H. R. P. A. v. A. A.-G. Rafael Gómez-Bombarelli, Jennifer N. Wei, "Automatic chemical design using a data-driven continuous representation of molecules," *ACS Central ScienceACS Central Science*, 2018.

[4] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006. [Online]. Available: http://research.microsoft.com/en-us/um/people/cmbishop/prml/

[5] T. S. J. L. K. S. Michael I. Jordan, Zoubin Ghahramani, *An Introduction to Variational Methods for Graphical Models*. Springer, 1999.