**TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH  
KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CƠ SỞ**

**TÌM HIỂU VÀ SỬ DỤNG CÔNG NGHỆ AI TRONG DỰ ĐOÁN THỊ TRƯỜNG CHỨNG KHOÁN**

**Giảng viên hướng dẫn: PHẠM ĐÌNH TÀI**

**Sinh viên thực hiện: NGÔ PHƯƠNG NAM**

**MSSV: 2100011233**

**Khoá: 21**

**Ngành/ chuyên ngành: TRÍ TUỆ NHÂN TẠO**

**Tp.HCM, Tháng 09 Năm 2023**

# Lời cảm ơn

Lời nói đầu tiên nhóm chúng em xin gửi lời cám ơn chân thành đến giảng viên bộ môn đồ án cơ sở trí tuệ nhân tạo, là thầy Phạm Đình Tài, đã hướng dẫn cho chúng em những buổi học thực hành qua nhiều hình thức khác nhau, thầy đã nhiệt tình hướng dẫn chúng em trong suốt một học kỳ kéo dài hơn 2 tháng vừa qua. Trong quá trình thực hiện đồ án, chúng em đã được tiếp thu, cả thiện và nâng cao thêm rất nhiều kiến thức quan trọng của môn học này, và bên cạnh đó thầy cũng đã hướng dẫn cho chúng em những kỹ năng, những kinh nghiệm cần thiết hoàn thành dự án một cách tốt nhất có thể.

Dù vẫn chưa có nhiều kinh nghiệm và sản phẩm của chúng em vẫn có nhiều thiếu sót trong quá trình thực hiện đồ án, nên sẽ không tránh được những sai sót và những điểm chưa hợp lí trong đồ án mà chúng em đã triển khai. Chúng em mong qua đồ án của mình sẽ nhận được sự đóng góp ý kiến, nhận xét từ phía thầy để qua đó giúp em bổ sung kiến thức, kỹ năng để có thêm kinh nghiệm trong việc cải thiện cho nguồn vốn kiến thức cũng như trang bị một hành trang cho chúng em sau này.

Em xin chân thành cảm ơn thầy!

Nhóm sinh viên thực hiện

Ngô Phương Nam

Trấn Văn Thắng

# Lời mở đầu

Hiện nay nền kinh tế ngày càng phát triển vfa hầu hết các lĩnh vực đề dùng đến công nghệ, ngành công nghệ thông tin đã trở thành cánh tay đắc lực cho tất các ngành như phương tiện truyền thông, xã hội , van hóa, giải trí khoa học kỹ thuật, kinh doanh, y tế, giáo dục,… đâu đâu cũng cần đến công nghệ để hỗ trợ nhằm để thúc đẩy và phát triển vững mạnh hơn.

Giờ đây, nhiều người đầu tư vào cổ phiếu và họ cần một công cụ dự đoán cổ phiếu để biết khi nào nên mua cổ phiếu. Bây giờ, không thể dự đoán điều gì sẽ xảy ra trong tương lai, nhưng chúng ta có thể ước tính và dự báo dựa trên dữ liệu chúng ta có trong hiện tại và quá khứ về cổ phiếu. Điều này được gọi là Phân tích Kỹ thuật, được sử dụng để dự đoán hướng giá của cổ phiếu, giá trị của cổ phiếu sẽ tăng hay giảm sau một thời gian cụ thể.

Python là ngôn ngữ lập trình phổ biến nhất hiện nay, cấu trúc rõ ràng, thuận tiện cho người ới bắt đầu đầu học lập trình. Với Python là ngôn ngữ lật trình đa năng, có tính hướng đối tượng cao, xử lý các dạng file hiệu quả nhanh và được sử dụng nhiều cho các thuật toán xử lý dữ liệu lớn. chính vì những ưu điểm này nên nhóm chúng em đã chọn ngôn ngữ lập trình Python để thực hiện đề tài “Tìm hiểu và sử dụng công nghệ AI trong dự đoán thị trường chứng khoán”.

# 

BM-ChT-11

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH  **TRUNG TÂM KHẢO THÍ** | **KỲ THI KẾT THÚC HỌC PHẦN**  **HỌC KỲ 6 – NĂM HỌC 2022 - 2023** |

# Phiếu chấm đồ án cơ sở

Môn thi: **Đồ án cơ sở trí tuệ nhân tạo** Lớp học phần: **21DTH1B**

Nhóm sinh viên thực hiện :

1/ Ngô Phương Nam\_ 2100011233.

2/ Trấn Văn Thắng \_ 2100010618.

Ngày thi: 28/09/2023 Phòng thi: L.511

Đề tài tiểu luận/báo cáo của sinh viên : Tìm hiểu ứng dụng cộng nghệ AI trong dự đoán thị trường chứng khoán

Phần đánh giá của giảng viên (căn cứ trên thang rubrics của môn học):

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí (theo CĐR HP)** | **Đánh giá của GV** | **Điểm tối đa** | **Điểm đạt được** |
| Cấu trúc của báo cáo |  | 2 |  |
| Nội dung |  |  |  |
| * Các nội dung thành phần |  | 2 |  |
| * Lập luận |  | 4 |  |
| * Kết luận |  | 1 |  |
| Trình bày |  | 1 |  |
| **TỔNG ĐIỂM** |  | 10 |  |

**Giảng viên chấm thi** *(ký, ghi rõ họ tên)*

Phạm Đình Tài

**MỤC LỤC**

[Lời cảm ơn i](#_Toc146579963)

[Lời mở đầu ii](#_Toc146579964)

[Phiếu chấm đồ án cơ sở iii](#_Toc146579965)

[Danh mục hình ảnh vi](#_Toc146579966)

[Chương 1: Ngôn ngữ lập trình python 1](#_Toc146579967)

[1. Giới thiệu: 1](#_Toc146579968)

[2. Các khái niệm cơ bản trong lật trình python 10](#_Toc146579969)

[3 Cấu trúc dữ liệu là gì, các kiểu cấu trúc dữ liệu trong python. 11](#_Toc146579970)

[4. Cấu trúc điều khiển trong python 12](#_Toc146579971)

[5. Cài đặt các gói thư viện cơ bản trong python 15](#_Toc146579972)

[6. Sử Dụng Hàm Trong Python 18](#_Toc146579973)

[7. Đọc file CSV trong python 20](#_Toc146579974)

[8. Sử dụng thư viện pandas trong Python 22](#_Toc146579975)

[Chương 2 : Bài toán dự báo 25](#_Toc146579976)

[1. Bài toán dự đoán 25](#_Toc146579977)

[2. Tiến trình của một bài toán dự báo 31](#_Toc146579978)

[3. Tiêu chuẩn dự báo 31](#_Toc146579979)

[Chương 3 : Bài toán Tìm hiểu ứng dụng cộng nghệ AI trong dự đoán thị trường chứng khoán 33](#_Toc146579980)

[1. Giới thiệu bài toán dự báo giá cổ phiếu 33](#_Toc146579981)

[2. Phát biểu bài toán 35](#_Toc146579982)

[3. Một số mô hình dự báo thồn kê 36](#_Toc146579983)

[Chương 4 : Ứng dụng công nghệ AI trong thị trường chứng khoán 38](#_Toc146579984)

[1. Các thư viện 38](#_Toc146579985)

[2. Dữ liệu 39](#_Toc146579986)

[3. Thống kê 39](#_Toc146579987)

[4. Lịch sử giá đóng của của các công ty : 40](#_Toc146579988)

[5. Biểu đồ 41](#_Toc146579989)

[6. Huấn luyện 43](#_Toc146579990)

[7. Mô hình huấn luyện 43](#_Toc146579991)

[Kết luận 45](#_Toc146579992)

[Danh mục tài liệu kham thảo 46](#_Toc146579993)

# Danh mục hình ảnh

[Hình 1: Giao diện Google Colab 4](#_Toc146203592)

[Hình 2: Tạo mới, mở hoặc upload 1 Notebook 5](#_Toc146203593)

[Hình 3: Lưu Notebook 6](#_Toc146203594)

[Hình 4: Download Notebook 7](#_Toc146203595)

[Hình 5: Upload Dữ liệu 7](#_Toc146203596)

[Hình 6: Copy File Path 8](#_Toc146203597)

[Hình 7: Comment trong python 9](#_Toc146203598)

[Hình 8: Lệnh if 13](#_Toc146203599)

[Hình 9: Kết quả lệnh if 13](#_Toc146203600)

[Hình 10: Lệnh for 13](#_Toc146203601)

[Hình 11: Kết quả lệnh for 14](#_Toc146203602)

[Hình 12: Lệnh while 14](#_Toc146203603)

[Hình 13: Minh họa cài đặt thư việc python 15](#_Toc146203604)

[Hình 14: Minh họa về Hàm 17](#_Toc146203605)

[Hình 15: Minh họa Hàm có tham số 18](#_Toc146203606)

[Hình 16: Minh họa đọc file 21](#_Toc146203607)

[Hình 17: Minh họa đọc file csv 23](#_Toc146203608)

[Hình 18: Minh họa cho bảng dữ liệu 26](#_Toc146203609)

[Hình 19: Đồ thị xt theo t 27](#_Toc146203610)

[Hình 20: Đồ thị của xt / xt-1 \* 100 theo t 27](#_Toc146203611)

[Hình 21: Đồ thị của Xt – Xt-1 ~ t 28](#_Toc146203612)

[Hình 22: Một số định dạng dữ liệu 29](#_Toc146203613)

[Hình 23: Tiến trình chung của một bài toán adujw báo 30](#_Toc146203614)

[Hình 24: Các thư viện 37](#_Toc146203615)

[Hình 25: Minh họa dữ liệu 38](#_Toc146203616)

[Hình 26: Minh họa thống kê dữ liệu 39](#_Toc146203617)

[Hình 27: Lịch sử giá đóng cửa của các công ty 39](#_Toc146203618)

[Hình 28: Kết quả giao dịch 40](#_Toc146203619)

[Hình 29: Biểu đồ hoàn vốn mỗi ngày 41](#_Toc146203620)

[Hình 30: Khung giá đóng cửa 41](#_Toc146203621)

[Hình 31: Huấn luyện dữ liệu 42](#_Toc146203622)

[Hình 32: Biểu đồ mô hình huấn luyện 43](#_Toc146203623)

# Chương 1: Ngôn ngữ lập trình python

## 1. Giới thiệu:

### 1.1. Lịch sử hình thành:

Python là ngôn ngữ lập trình tập trung vào sự đơn giản và dễ sử dụng hướng đối tượng, cấp cao, mạnh mẽ. Ngày nay, Pthon được sử dụng rộng rãi trong nhiều lĩnh vực và là một trong những ngôn ngữ lập trình được sử dụng nhiều nhất trên thế giới.

Python đã được hình thành vào cuối những năm 1980 và được bắt đầu thực hiện vào tháng 12/1989 bởi Guido Van Rossum tại CWI Hà Lan như là người kế thừa của ngôn ngữ ABC (tự lấy cảm hứng từ SETL) có khả năng xử ngoại lệ và giao tiếp với hệ điều hành Amoeba. Van Rossum là tác giả chính của Python, và vai trò trung tâm của ông tiếp tục trong việc quyết định hướng phát triển của Python được phản ánh trong tiêu đề mà cộng đồng Python dành cho ông “Độc tài nhân từ cho cuộc sống” (benevolent dictator for life) (BDFL)

Python 2.0 được phát hành vào ngày 16/10/2000, với nhiều tính năng chính mới bao gồm một bộ dọn rác đầy đủ hỗ trợ Unicode[[1]](#footnote-1). Với phiên bản này, quá trình phát triển đã được phát triển đã được thay đổi và trở thành minh bạch hơn và được cộng đồng ủng hộ.

Python 3.0 (còn được gọi là Python 3000 hoặc Py3k), một bản phát hành lớn, không tương thích ngược, được phát hành vào ngày 03/12/2008 sau một thời gian dài thử nghiệm. nhiều trong số các tính năng của nó đã được điều chỉnh để tương thích ngược với Python 2.6 và 2.7. Các tính năng và triết lý phát triển Python là một ngôn ngữ lập trình đa hình: lập trình hướng đối tượng và hướng cấu trúc được hỗ trợ đầy đủ, và có 1 số tính năng hỗ trợ ngôn ngữ lập trình theo chức năng và lập trình hướng khía cạnh (Aspect-oriented programming). Nhiều môn hình khác được hỗ trợ bằng việc sử dụng các phần mở rộng, bao gồm thiết kế theo hợp đồng (design by contract) và lập trình luận lý.

Các trang như Mozilla, Reddit, Instagram và PBS đều được viết bằng Python.

Ngôn ngữ lập trình Python được dùng vào các mục đích:

* Phát triển Web (trên máy chủ)
* Phát triển phần mềm
* Tính toán một cách khoa học
* Lên kịch bản cho hệ thông.

Tại sao nên học lật trình Python?

* Python hỗ trợ nhiều nền khác nhau (Windown, Mac, Linux, RaspberryPi, etc)
* Python có cú pháp đơn giản, để đọc hiểu và rất gần gũi với tiếng Anh.
* Cú pháp của Python giúp lập trình viên sử dụng ít dòng code để lập trình cùng một thuật toán hơn so với các ngôn ngữ lập trình khác.
* Python sử dụng trình thông dịch để thực thi các dòng code. Do đó, những dòng code có thể thực thi ngay lập tức mà không cần biên dịch toàn bộ chương trình. Như vậy giúp chúng ta được code nhanh hơn.

Python cũng hỗ trợ hàm, thủ tục, hay kể cả lập trình hướng đối tượng.

Để viết nguồn mã Pyhton, ta có thể sử bất kỳ một trình soạn thảo nào, kể cả những trình soạn thảo đơn giản nhất như NotePad[[2]](#footnote-2). Tuy nhiên, để phát triển các ứng dụng một cách hiệu quả hơn, ta sử dụng một IDE[[3]](#footnote-3), để có thể tiết kiệm thời gian và công sức viết code.

Trong phần tìm hiểu này, đồ án sử dụng một trong những IDE thông dụng nhất để lạp trình ứng dụng Python, đó là Gogole Colab

### 1.2. Cài đặt Google Colab

Google Colab là một dạng Jupyter Notebook [[4]](#footnote-4)tùy biến cho phép thực thi Python trên nền tảng đám mây, được cung cấp bởi Google. Sử dụng Google Colab có những lợi ích ưu việt như: sẵn sàng chạy Python ở bất kỳ thiết bị nào có kết nối internet mà không cần cài đặt, chia sẻ và làm việc nhóm dễ dàng, sử dụng miễn phí GPU[[5]](#footnote-5) cho các dự án về AI[[6]](#footnote-6).

Ngoài những lợi ích được tóm tắt như trên, Google Colab còn cung cấp cho bạn trải nghiệm lập trình Python tuyệt vời với những nâng cấp cực kỳ hữu ích không có trong Jupyter Notebook, JupyterLab đơn thuần. Những tính năng tôi đánh giá cao ở Google Colab phải kể đến như:

**Tạo mục lục dựa trên các Heading viết bằng ngôn ngữ markdown** giúp bạn dễ dàng cấu trúc Notebook làm việc của mình. Bạn cũng có thể Thu gọn (Collapse) hay Mở rộng (Expand) các phần nội dung khi soạn thảo cực kỳ tiện lợi. Tôi làm việc với Visual Studio Code[[7]](#footnote-7), Jupyter Lab hay Jupyter Notebook đều cầu mong phải chi chúng có tính năng tương tự Google Colab. Tôi cũng đã sử dụng Google Colab để soạn thảo một giáo trình hoàn chỉnh có khả năng tương tác cao cho khoá học Python.

**Thêm hình ảnh, biểu mẫu dễ dàng với markdown** giúp bạn trình bày báo cáo hoặc làm dashboard cực tiện lợi. Thậm chí bạn có thể ẩn các dòng code để trông Notebook gọn gàng hơn với tính năng biểu mẫu.

**Kết nối dễ dàng với Google Drive, Google Sheets** để bắt tay vào phân tích dữ liệu “trên mây” hoàn toàn.

**Chạy Python trên Cloud hay Local Runtime (Python trên máy tính cá nhân) của bạn đều cho trải nghiệm tốt**. Bạn vẫn tận dụng được tính năng tuyệt vời của Google Colab khi chạy với Python trên Local Runtime trong khi không bị Google tự động xóa dữ liệu khi kết thúc phiên làm việc như khi chạy trên Cloud.

**Tự động lưu lịch sử chỉnh sửa thành các phiên bản giúp bạn dễ dàng khôi phục lại phiên bản gần nhất nếu cần khi bạn gặp lỗi**. Tính năng này tương tự như trên Google Sheets hay Google Docs, bạn thậm chí không cần đến Github[[8]](#footnote-8) để lưu trữ các phiên bản chỉnh sửa này.

**Cho phép tìm kiếm và chèn các đoạn mã được soạn thảo sẵn trong các Template (bởi bạn) vào Notebook**. Tính năng này rất hay bởi bạn không cần phải mở thêm nhiều file lưu trữ để tìm lại các đoạn code mẫu mình đã biết khi cần. Workflow lập trình Python trở nên đơn giản và hiệu quả hơn rất nhiều.

**Tạo dashboard viết bằng Python và chia sẻ với team dễ dàng** nếu cần tương tự như Google DataStudio nhưng linh hoạt và mạnh mẽ hơn rất nhiều.

Tuy nhiên Google Colab có 1 nhược điểm gây khó chịu không ít đó là dữ liệu (bộ nhớ tạm) của phiên làm việc sẽ bị xóa sau khi bạn không active trong 1 thời gian nhất định để Colab đảm bảo có thể cung cấp tài nguyên miễn phí cho nhiều người. Do đó mỗi khi mở Google Colab, nếu bạn cần sử dụng các thư viện của bên thứ 3 thì bạn cần install và import lại từ đầu để có thể sử dụng. Phiên bản Colab Pro giúp khắc phục điều này nhưng hiện tại không áp dụng cho thị trường Việt Nam.

### 1.3. Giao diện Google Colab



Hình 1: Giao diện Google Colab

### 1.4. Thiết lập Ngôn ngữ hiển thị

Google Colab mặc định hiển thị ngôn ngữ tiếng Việt cho tôi nhưng tôi luôn lựa chọn làm việc với phiên bản tiếng Anh cho tiện giao tiếp và tìm kiếm hỗ trợ dễ dàng khi cần. Nếu bạn cũng cần thay đổi ngôn ngữ, hãy tìm lựa chọn này ở menu trợ giúp xem bằng tiếng Anh

https://colab.research.google.com/drive/1WdRrOmnTI1s-KppcGGkBjBA9ZU\_X2LiK?authuser=1&hl=vi  sang https://colab.research.google.com/drive/1WdRrOmnTI1s-KppcGGkBjBA9ZU\_X2LiK?authuser=1&hl=en.

### 1.5. Thao tác với File & Folder

#### a. Tạo mới, mở hoặc upload 1 Notebook sẵn có

Để mở File với Google Colab, bạn có thể sử dụng tổ hợp phím tắt Ctrl + O (hoặc Command + O trên Macbook). Bạn có 5 tùy chọn chính trong đó có :

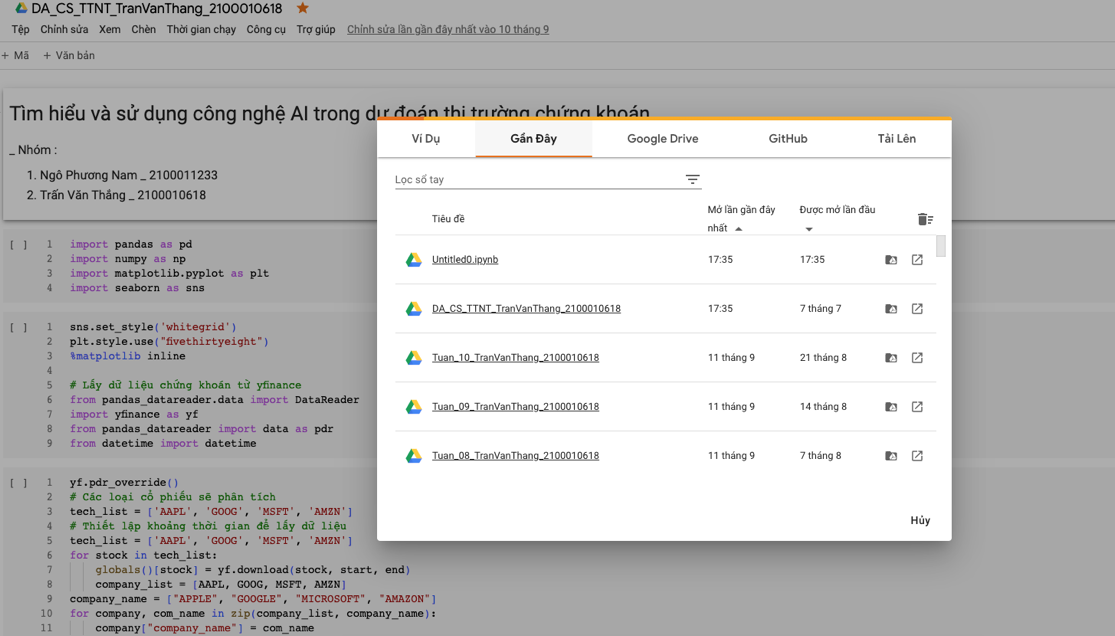
Examples cho phép mở các dataset[[9]](#footnote-9) ví dụ

Recent: Mở notebook được chạy gần đây

Google Drive: mở Jupyter Notebook từ Drive (file định dạng .ipynb)

Github: Cho phép bạn kết nối với Github và clone các project của mình cũng như mở bằng Colab

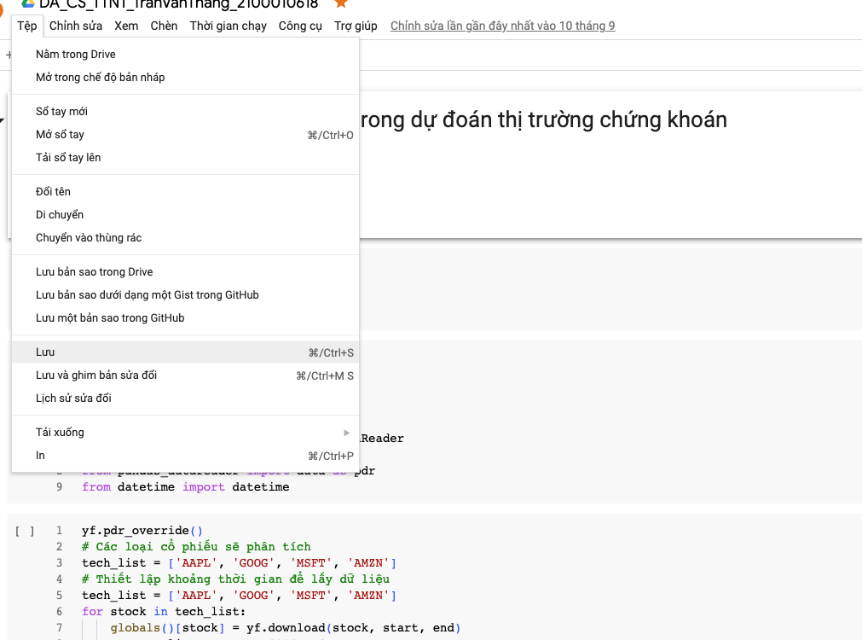
Upload cho phép bạn duyệt file trên máy tính cá nhân và tải lên file notebook có định dạng .ipynb



Hình 2: Tạo mới, mở hoặc upload 1 Notebook

#### b. Lưu Notebook đang làm việc

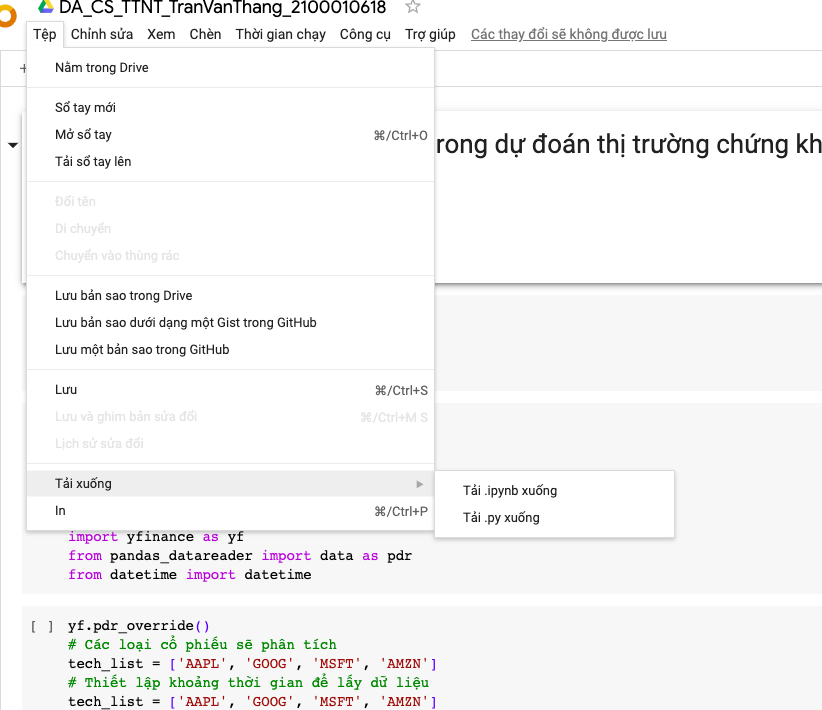
Google Colab thực hiện lưu dữ liệu làm việc của bạn hoàn toàn tự động tuy nhiên trong trường hợp bạn chưa yên tâm hoặc muốn có nhiều tùy chọn lưu trữ hơn thì có thể tìm đến **Menu >> File** và chọn lưu file trên Google Drive hay Github tùy mục đích của mình.



Hình 3: Lưu Notebook

#### c. Download Notebook

Bạn có thể tải Notebook về máy dưới dạng file Jupyter Notebook có định dạng .ipynb hoặc file Python có định dạng .py bằng cách mở menu **File >> Download** sau đó chọn định dạng file tương ứng.



Hình 4: Download Notebook

#### d. Upload Dữ liệu

Để upload dữ liệu vào bộ nhớ tạm của session làm việc trong Colab, bạn có thể chọn mục Folder như trong hình sau đó chọn icon Upload dữ liệu như hình minh họa dưới đây. Trong trường hợp bạn sử dụng dữ liệu từ Google Drive để làm việc, hãy theo dõi hướng dẫn ở phần tiếp theo của bài viết.



Hình 5: Upload Dữ liệu

#### e. Copy File Path

Để có thể copy đường dẫn file hoặc thư mục khi bạn cần thao tác mở hoặc lưu trữ file, bạn có thể nhấp chuộc phải vào file hoặc folder trong cây thư mục và chọn Copy path.



Hình 6: Copy File Path

### 1.6. Edit – Soạn thảo Notebook trong Google Colab

Ngoài việc copy text thông thường, Google Colab cho phép bạn copy/paste Code cell và Text cell khá tiện lợi. Tính năng này khá hay nhưng tôi không thấy trên Jupyter Notebook, Jupyter Lab và Visual Studio Code. Để copy / paste các cell này, bạn sử dụng Ctrl và Click chọn các cell cần copy/paste sau đó sử dụng tổ hợp phím tắt Ctrl + C và Ctrl + V quen thuộc.

Bạn có thể kích hoạt tính năng Find and replace bằng tổ hợp phím Ctrl + H hoặc chọn icon kính lúp ở thanh công cụ bên cạnh trái màn hình.

### 1.7. View – Tuỳ chọn hiển thị trong Google Colab

Chọn Menu >> View:

Executed Code History: Xem lịch sử các dòng lệnh đã được thực thi trong Notebook

Collapse sections: cho phép thu gọn các nội dung phân cấp nhỏ hơn ở mỗi 1 cấp độ heading (ví dụ Collapse tại cell chứa heading 1 thì các nội dung thuộc heading từ 2 trở đi và text thông thường sẽ được thu gọn lại). Bạn có thể chọn 1 cell bất kỳ trong Notebook và sử dụng tổ hợp phím tắt Ctrl + ] để Collapse.

Expand sections: Tác dụng ngược lại so với Collapse sections – cho phép mở rộng các mục nội dung dưới 1 cấp heading bất kỳ. Tổ hợp phím tắt Ctrl + [ cho kết quả tương tự.

### 1.8. Insert – Thêm nội dung vào Notebook trong Google Colab

Để thêm 1 Code cell (thực thi lệnh) hoặc Text cell (văn bản), bạn có thể di chuột vào 1 cell sẵn có và chọn đối tượng cần thêm tương ứng. Nếu là Notebook mới hoàn toàn bạn có thể thêm cell bằng 2 nút ở góc trái màn hình như khoanh đỏ ở hình minh họa sau. Bạn cũng có thể sử dụng tổ hợp phím tắt mặc định Ctrl + MB để thêm 1 Code cell.

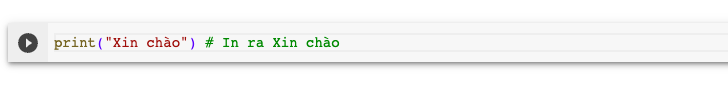
Để thêm 1 Section header bạn có thể dùng tùy chọn tương ứng tron mục Menu >> Insert hoặc thêm 1 Text cell sau đó tạo Heading bằng định dạng markdown. Ví dụ để tạo Heading 1 là mục giới thiệu như trong hình, bạn cần viết **# I. GIỚI THIỆU** vào Text cell. Thêm 1 dấu # trước đầu mục cho mỗi cấp độ Heading nhỏ hơn.

Scratch Code Cell: Bằng cách kích hoạt tùy chọn này, bạn mở ra 1 Code cell sử dụng như bản nháp thực thi code bên cạnh việc soạn thảo chính trong Notebook.

## 2. Các khái niệm cơ bản trong lật trình python

### 2.1. Comment trong python

Trong Python bất kỳ văn bản nào ở bên phải biểu tượng # thì sẽ được trình biên dịch hiểu là một comment và không biên dịch phần đó.



Hình 7: Comment trong python

Sử dụng càng nhiều comment hữu ích trong chương trình của bạn sẽ làm cho công việc lập trình của bạn dễ dàng hơn:

Giải thích các giả định.

Giải thích các quyết định quan trọng.

Giải thích chi tiết quan trọng.

Giải thích vấn đề bạn đang cố gắng giải quyết.

Giải thích các vấn đề đang cố gắng khắc phục trong chương trình của mình, v.v. .

Hằng số (Literal Constants): Ví dụ về một hằng số theo nghĩa đen là một số như 5, 1. 23 hoặc một chuỗi như ‘python’hay “It’s a string!”. Nó được gọi là nghĩa đen bởi vì sử dụng giá trị của nó theo nghĩa đen. Số 2 luôn luôn đại diện cho chính nó và không có gì khác và nó là một hằng số vì giá trị của nó không thể thay đổi. Do đó, tất cả những giá trị này được gọi là hằng số.

Số (Numbers): Số chủ yếu có hai loại – số nguyên (integer) và số thực (float).

Chuỗi (String): Một chuỗi là một dãy các ký tự. Chuỗi về cơ bản chỉ là một loạt các từ.

Biến: Biến chính xác như tên gọi của nó, tức là giá trị của nó có thể thay đổi. Các biến có thể giúp lưu trữ bất cứ cái gì nếu có thể định nghĩa được nó. Các biến chỉ là một phần của bộ nhớ máy tính nơi lưu trữ một số thông tin.

Đối tượng (Object): Python đề cập đến bất cứ điều gì được sử dụng trong một chương trình như là một đối tượng.

## 3 Cấu trúc dữ liệu là gì, các kiểu cấu trúc dữ liệu trong python.

Việc tổ chức, quản lý và lưu trữ dữ liệu rất quan trọng vì nó cho phép truy cập dễ dàng hơn và sửa đổi hiệu quả. Cấu trúc dữ liệu (Data Structure) cho phép bạn sắp xếp dữ liệu của mình theo cách cho phép bạn lưu trữ các bộ dữ liệu được thu thập, liên quan đến chúng và theo đó mà thực hiện các thao tác trên chúng.

Python có hỗ trợ ngầm cho Cấu trúc dữ liệu cho phép lưu trữ và truy cập dữ liệu. Các cấu trúc này được gọi là List, Dictionary, Tuple và Set.

Python cho phép người dùng tạo Cấu trúc dữ liệu của riêng họ, cho phép toàn quyền kiểm soát chức năng. Các cấu trúc dữ liệu nổi bật nhất là Stack, Queue, Tree, Linked List, v.v. đồng thời cũng có sẵn trong các ngôn ngữ lập trình khác. Cấu trúc dữ liệu tích hợp (Built-in Data Structures)

Về cấu trúc dữ liệu trong Python, các Cấu trúc dữ liệu này được tích hợp sẵn với Python giúp lập trình dễ dàng hơn và giúp các lập trình viên sử dụng chúng để có được các giải pháp nhanh hơn. Và có các kiểu cấu trúc dữ liệu là:

List: Được sử dụng để lưu trữ dữ liệu của các loại dữ liệu khác nhau một cách tuần tự. Có các địa chỉ được gán cho mọi thành phần của danh sách, được gọi là Index. Giá trị chỉ mục bắt đầu từ 0 và tiếp tục cho đến khi phần tử cuối cùng được gọi là chỉ số dương.

Dictionary: Được sử dụng để lưu trữ các cặp key-value. Để hiểu rõ hơn, hãy nghĩ đến một thư mục điện thoại nơi hàng trăm và hàng ngàn tên và số tương ứng của chúng đã được thêm vào. Bây giờ các giá trị không đổi ở đây là Tên và Số điện thoại được gọi là các phím. Và các tên và số điện thoại khác nhau là các giá trị đã

được đưa vào các phím. Nếu truy cập các giá trị của các phím, sẽ nhận được tất cả tên và số điện thoại. Vì vậy, đó là những gì một cặp key-value. Và trong Python, cấu trúc này được lưu trữ bằng Dictionary

Set: Là một tập hợp các yếu tố không có thứ tự là duy nhất. Có nghĩa là ngay cả khi dữ liệu được lặp lại nhiều lần, nó sẽ chỉ được nhập vào tập hợp một lần.

Tuple (các bộ dữ liệu): Tuples giống như các list với ngoại lệ là dữ liệu một khi được nhập vào bộ dữ liệu không thể thay đổi bất kể điều gì. Ngoại lệ duy nhất là khi dữ liệu bên trong Tuple có thể thay đổi, chỉ sau đó dữ liệu Tuple có thể được thay đổi.

|  |  |
| --- | --- |
| **LIST** | **TUPLE** |
| Được sử dụng cho các loại dữ liệu đồng nhất | Thường được sử dụng cho các loại dữ liệu không đồng nhất |
| Có thể thay đổi trong môi trường | Bất biến trong môi trường giúp lặp lại nhanh hơn |
| Không có yếu tố bất | Các yếu tố bất biến có thể được sử dụng được làm cho key cho từ điển |
| Khong đảm bảo rằng dữ liệu được bảo vệ chống ghi | Việc thực hiện một số dữ liệu không thay đổi đảm bảo rằng nó được bảo vệ chống ghi |

## 4. Cấu trúc điều khiển trong python

Python luôn chạy một loạt các câu lệnh theo thứ tự từ trên xuống một cách chính xác. Câu lệnh điều khiển là loại câu lệnh được dùng để điều khiển luồng chạy của các câu lệnh khác trong chương trình.

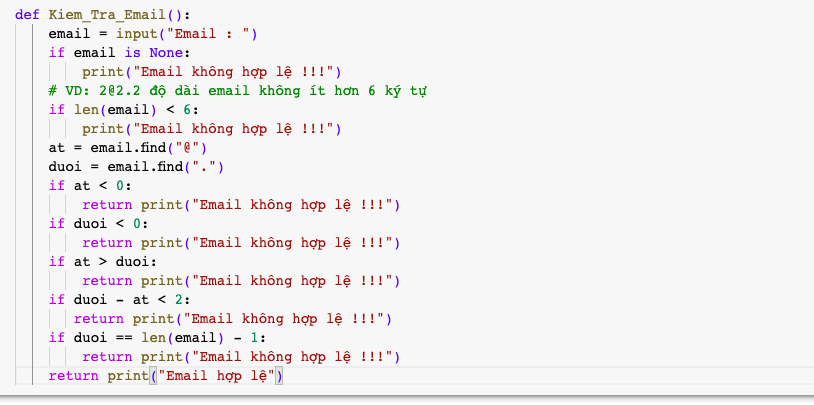
Điều này đạt được bằng cách sử dụng các câu lệnh điều khiển. Có 3 lệnh kiếm soát cấu trúc điều khiển của Python là: if, for và while.

### 4.1. Lệnh IF

Được sử dụng để kiểm tra một điều kiện: nếu điều kiện là đúng sẽ chạy một khối các câu lệnh (được gọi là if-block), nếu sai chương trình sẽ xử lý một khối các câu lệnh khác (được gọi là else-block ).

Ví dụ:

Cho một số nguyên cho trước, nhập vào một số nguyên khác và báo về các kết quả. Nếu thỏa mãn các điều kiện thì in ra các kết quả khác nhau



Hình 8: Lệnh if

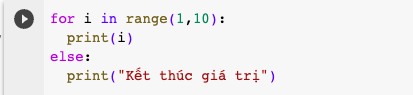
Kết quả cho thấy khi chúng ta nhập một số nguyên bất kì. Nếu thỏa mãn điều kiện bằng với số cho trước thì in ra “chính xác”. Và kết quả khác khi điền vào số lớn hoặc nhỏ hơn số cho trước.



Hình 9: Kết quả lệnh if

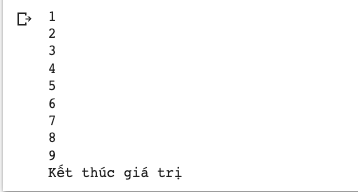
### 4.2 Lệnh for

Câu lệnh for … in là một câu lệnh lặp khác, nó lặp đi lặp lại qua 1 chuỗi (sequences) các đối tượng tức là đi qua từng mục trong chuỗi.



Hình 10: Lệnh for

Kết quả Vòng lặp for trong Python hoàn toàn khác với vòng lặp trong C / C+ vòng lặp for trong Python tương tự như vòng lặp foreach trong C#. Trong C / C ++, nếu bạn muốn viết for(int i = 0; i <5; i ++), thì trong Python bạn chỉ cần viết for i in range(0,5). Vòng lặp for đơn giản hơn, biểu cảm hơn và ít bị lỗi hơn trong Python.



Hình 11: Kết quả lệnh for

### 4.3. Lệnh While

Trong Python: Câu lệnh while cho phép liên tục thực thi một khối các câu lệnh miễn là điều kiện là đúng. Một câu lệnh while là một ví dụ về cái được gọi là câu lệnh lặp. Một câu lệnh while có thể có một mệnh đề khác tùy chọn.

Ví dụ nhập vào một số nguyên sao cho thỏa mãn các điều kiện cho trước, sau khi thỏa mãn một khối các lệnh thì in ra kết quả “Hoàn thành”.



Hình 12: Lệnh while

## 5. Cài đặt các gói thư viện cơ bản trong python

### 5.1. Cài đặt pip

Pip là công cụ quản lý các gói thư viện python (Python package manager). Với việc sử dụng pip, ta sẽ dễ dàng cài đặt các gói thư viện cần thiết trong quá trình làm việc với Python.

Đề cập nhật pip hiện thời lên phiên bản mới nhất, ta mở cmd và dung câu lệnh

*python -m pip install --upgrade pip*

thư viện Matplotlib là một thư viện xây dựng các khối cần thiết để tạo các biểu đồ chất lượng từ mảng và trực quan và tương tác với chúng.



Hình 13: Minh họa cài đặt thư việc python

### 5.2. Một số gói thư viện thông dụng cho Python

#### 5.2.1. NumPy

Được tạo bởi Travis Oliphant, NumPy là một “ngựa kéo” phân tích thực sự của Python. Nó cung cấp cho người dùng cách làm việc với các mảng nhiều chiều, cùng một số lượng lớn các hàm để xử lý trên các toán tử toán học nhiều chiều trên các mảng đó. Mảng là các khối dữ liệu được sắp xếp theo nhiều chiều dựa trên các véc tơ và ma trận trong toán học. Mảng thường hữu ích không chỉ trong việc lưu dữ liệu mà cả việc tính toán nhanh các ma trận, điều không thể thiếu khi giải quyết các vấn đề liên quan đến khoa học dữ liệu.

*Cài đặt: pip install numpy*

Trang chủ: http://www. numpy. org

#### 5.2.2. SciPy

Là một dự án gốc bởi Travis Oliphant, Pearu Peterson, and Eric Jones, SciPy hoàn thiện các tính năng của NumPy, nhằm cung cấp các thuật toán cho đại số tuyến tính, không gian ma trận, xử lý tín hiệu và xử lý ảnh, tối ưu, biến đổi Fourier, ...

*Cài đặt: pip install sclipy*

- Trang chủ: http://www. scipy. org

#### 5.2.3. Pandas

Pandas là thư viện thực hiện mọi thứ mà NymPy và SciPy không thể làm. Nó làm việc với các đối tượng cấu trúc dữ liệu, DataFrames và Chuỗi (Series). pandas cho phép bạn có thể xử lý các bảng dữ liệu phức tạp của nhiều loại khác nhau (điều mà các mảng của NumPy thông thể làm được) và chuỗi thời gian. Bạn sẽ dễ dàng tải dữ liệu từ nhiều nguồn khác nhau, sau đó slide, dice, xử lý các thành phần còn thiếu, thêm, đổi tên, tổng hợp (aggregate), reshape và cuối cùng là trực quan dữ liệu theo ý của bạn.

*Cài đặt: pip install pandas*

- Trang chủ: http://pandas. pydata. org

#### 5.2.4. Scikit-learn

Bắt đầu như một phần của SciKits, Scikit-learn là lõi hoạt động của khoa học dữ liệu trên Python. Nó cung cấp tất cả những gì bạn cần để tiền xử lý dữ liệu, học

giám sát và không giám sát, lựa chọn mô hình, validate và error metrics.

*Cài đặt: pip install scikit-learn*

Trang chủ: http://scikit-learn. org/stable

#### 5.2.5. IPython

Một cách tiếp cận khoa học yêu cầu thử nghiệm nhanh các giả thuyết khác nhau trong một khoảng thời gian. IPython được tạo bởi Fernando Perez để giải quyết việc cần thiết một lệnh Shell Python (dựa trên shell, trình duyệt web, và giao diện ứng dụng) với đồ họa tích hợp, các lệnh có thể tùy chỉnh, lịch sử phong phú (dưới định dạng JSON) và khả năng tính toán song song để cải thiện hiểu năng tính toán.

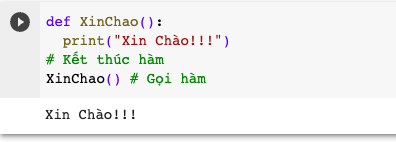
*Cài đặt: pip install “ipython[ notebook”*

- Trang chủ: http://ipython.Org

## 6. Sử Dụng Hàm Trong Python

### 6.1 Hàm (Function)

Là những phần tái sử dụng của chương trình. Chúng cho phép đặt tên cho một khối các câu lệnh, cho phép chạy khối đó bằng cách sử dụng tên được chỉ định ở bất kỳ đâu trong chương trình với số lần không hạn chế. Điều này được gọi là gọi hàm. Các Hàm được xác định bằng cách sử dụng từ khóa “def. Sau khi từ khóa này xuất hiện một tên định danh cho hàm, theo sau là một cặp dấu ngoặc đơn có thể kèm theo một số tên của các biến và bởi dấu hai chấm cuối cùng kết thúc dòng. Tiếp theo sau là khối các câu lệnh của Hàm.



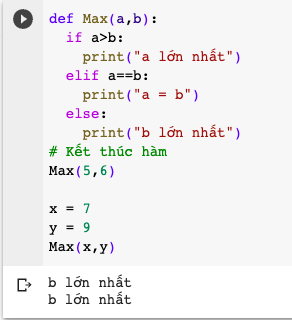
Hình 14: Minh họa về Hàm

### 6.2 Các thông số của Hàm (Function Parameters)

Một hàm có thể lấy tham số, là các giá trị bạn cung cấp cho hàm để hàm có thể sử dụng các giá trị đó cho một mục đích cụ thể. Các tham số này giống như các biến ngoại, trừ các giá trị của các biến này được xác định khi chúng ta gọi hàm và đã được gán các giá trị khi hàm chạy.

Các tham số được chỉ định trong cặp dấu ngoặc trong khai báo hàm, được phân tách bằng dấu phẩy. Khi chúng ta gọi hàm, chúng ta cung cấp các giá trị theo cùng một cách. Lưu ý thuật ngữ được sử dụng – các tên được đưa ra trong định nghĩa

hàm được gọi là tham số trong khi các giá trị bạn cung cấp trong lệnh gọi hàm được gọi là đối số.



Hình 15: Minh họa Hàm có tham số

### 6.3 Câu lệnh return trong Python

Lệnh return thường được dùng để thoát hàm và trở về nơi mà tại đó hàm được gọi. Lệnh này có thể chứa biểu thức được tính toán và giá trị trả về. Nếu không có biểu thức nào trong câu lệnh hoặc không có lệnh return trong hàm thì hàm sẽ trả về None. Lệnh return dùng để trả về một giá trị (hoặc một biểu thức), hoặc đơn giản là trả về "không gì cả". Khi lệnh return được thực thi, hàm sẽ kết thúc. return là

lệnh không bắt buộc phải có trong thân hàm.

### 6.4 DocStrings trong Python

Chuỗi đầu tiên ngay sau tiêu đề hàm được gọi là docstring (documentation string), nó được dùng để giải thích chức năng cho hàm. Mặc dù docstring là không bắt buộc, nhưng việc giải thích ngắn gọn về chức năng của hàm sẽ giúp người dùng sau khi gọi hàm có thể hiểu ngay hàm sẽ làm gì mà không cần phải tìm lại định nghĩa hàm để xem xét.

Việc thêm tài liệu cho code giúp sau khi quay trở lại có thể nhớ được chi tiết, rõ ràng đoạn code đã viết trước đó mà không có sai sót gì.

*Python có một tính năng tiện lợi gọi là chuỗi tài liệu, thường được gọi bằng tên ngắn hơn của nó docstrings. DocStrings là một công cụ quan trọng mà bạn nên sử dụng vì nó giúp ghi lại chương trình tốt hơn và dễ hiểu hơn. Chúng ta thậm chí có thể in hoặc sử dụng các chuỗi này trong các hàm.*

## 7. Đọc file CSV trong python

### 7.1 Tập CSV là gì ?

Một tệp csv là viết tắt của (comma separated values) được định nghĩa là định dạng tệp đơn giản sử dụng cấu trúc cụ thể để sắp xếp dữ liệu dạng bảng.

Tệp CSV là một loại tệp văn bản thuần túy sử dụng cấu trúc cụ thể để sắp xếp dữ liệu dạng bảng. CSV là một định dạng phổ biến để trao đổi dữ liệu vì sự nhỏ gọn, đơn giản và tính khái quát. Nhiều dịch vụ trực tuyến cho phép người dùng xuất dữ liệu dạng bảng từ trang web thành tệp CSV. Các tệp của CSV sẽ mở bằng Excel và gần như tất cả các cơ sở dữ liệu đều có công cụ cho phép nhập từ tệp CSV. Định dạng chuẩn được xác định bởi dữ liệu hàng và cột. Hơn nữa, mỗi hàng được kết thúc bởi một dòng mới để bắt đầu hàng tiếp theo. Cũng trong hàng, mỗi cột được phân tách bằng dấu phẩy.

### 7.2 Các hàm của module CSV trong Python

Công việc mô-đun CSV được sử dụng để xử lý các tệp CSV để đọc ghi và nhận dữ

liệu từ các cột được chỉ định. Có nhiều loại hàm CSV khác nhau, như sau:

*csv. field\_size\_limit - Nó trả về kích thước max của trường hiện tại.*

*csv. get\_dialect - Trả về dialect liên kết đến tên.*

*csv. list\_dialects - Trả về tên của tất cả các dialect đã đăng ký.*

*csv. reader - Đọc dữ liệu từ tệp csv.*

*csv. register\_dialect - Nó liên kết dialect với một tên. Tên phải là một chuỗi hoặc một đối tượng Unicode.*

*csv. writer - Ghi dữ liệu vào tệp csv.*

*csv. unregister\_dialect - Nó xóa dialect được liên kết với tên từ sổ đăng ký dialect. Nếu tên không phải là tên dialect đã đăng ký, thì lỗi sẽ được đưa ra.*

csv. QUOTE\_ALL-Nó hướng dẫn đối tượng writer trích dẫn tất cả các trường. csv. QUOTE\_MINIMAL - Nó hướng dẫn đối tượng writer chỉ trích dẫn những trường có chứa các ký tự đặc biệt như dấu ngoặc kép, dấu phân cách, v.v.

*Csv. QUOTE\_NONNUMERIC - Nó hướng dẫn đối tượng writer trích dẫn tất cả các trường không phải là số.*

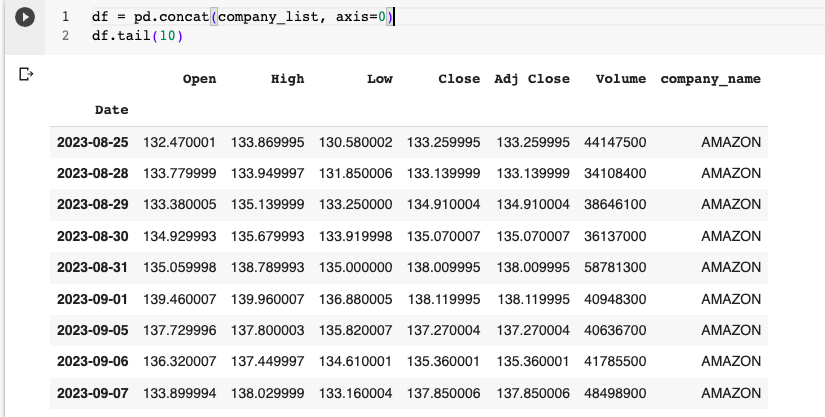
*csv. QUOTE\_NONE - Nó hướng dẫn đối tượng writer không bao giờ trích dẫn các trường.*

### 7.3 Cách đọc tập CSV

Để đọc dữ liệu từ tệp CSV, cần phải sử dụng hàm đọc để tạo đối tượng đọc.

Hàm đọc được viết để lấy từng hàng của tệp và tạo danh sách tất cả các cột. Sau đó, chọn cột bạn muốn trích xuất dữ liệu.

Vi dụ đọc và hiển thị file “1. csv”



Hình 16: Minh họa đọc file

## 8. Sử dụng thư viện pandas trong Python

### 8.1 Thư viện pandas python là gì?

Thư viện pandas trong python là một thư viện mã nguồn mở, hỗ trợ đắc lực trong thao tác dữ liệu. Đây cũng là bộ công cụ phân tích và xử lý dữ liệu mạnh mẽ của ngôn ngữ lập trình python. Thư viện này được sử dụng rộng rãi trong cả nghiên cứu lẫn phát triển các ứng dụng về khoa học dữ liệu. Thư viện này sử dụng một cấu trúc dữ liệu riêng là Dataframe. Pandas cung cấp rất nhiều chức năng xử lý và làm việc trên cấu trúc dữ liệu này. Chính sự linh hoạt và hiệu quả đã khiến cho pandas được sử dụng rộng rãi.

### 8.2 Tại sao lại dùng thư viện pandas?

DataFrame đem lại sự linh hoạt và hiệu quả trong thao tác dữ liệu và lập chỉ mục;

Là một công cụ cho phép đọc/ghi dữ liệu giữa bộ nhớ và nhiều định dạng file: csv, text, excel, sql database, hdf5;

Liên kết dữ liệu thông minh, xử lý được trường hợp dữ liệu bị thiếu. Tự động đưa dữ liệu lộn xộn về dạng có cấu trúc;

Dễ dàng thay đổi bố cục của dữ liệu;

Tích hợp cơ chế trượt, lập chỉ mục, lấy ra tập con từ tập dữ liệu lớn.

Có thể thêm, xóa các cột dữ liệu;

Tập hợp hoặc thay đổi dữ liệu với group by cho phép bạn thực hiện các toán tử trên tập dữ liệu;

Hiệu quả cao trong trộn và kết hợp các tập dữ liệu;

Lập chỉ mục theo các chiều của dữ liệu giúp thao tác giữa dữ liệu cao chiều và dữ liệu thấp chiều;

Tối ưu về hiệu năng;

Pandas được sử dụng rộng rãi trong cả học thuật và thương mại. Bao gồm thống kê, thương mại, phân tích, quảng cáo, v.v.

### 8.3 Đọc file csv sử dụng thư viện pandas

Đọc csv file vào dataframe

Bạn có thể dễ dàng đọc vào một file. csv bằng cách sử dụng hàm read\_csv và được trả về 1 dataframe. Mặc định, hàm này sẽ phân biệt các trường của file csv theo dấu phẩy. Cách đọc như sau:

peoples\_df = pd. read\_csv('./people. csv')

Để in ra n bản ghi đầu tiên của dataframe sử dụng hàm head. Ngược lại của hàm head là hàm tail:

peoples\_df. head (5)

Một số tham số của hàm read\_csv:

encoding: chỉ định encoding của file đọc vào. Mặc định là utf-8.

sep: thay đổi dấu ngăn cách giữa các cột. Mặc định là dấu phẩy (, )

header: chỉ định file đọc vào có header(tiêu đề của các cột) hay không. Mặc định là infer.

index\_col: chỉ định chỉ số cột nào là cột chỉ số(số thứ tự). Mặc định là None. nrows: chỉ định số bản ghi sẽ đọc vào. Mặc định là None – đọc toàn bộ.

import pandas as pd

import matplotlib. pyplot as plt

import random

peoples\_df = pd.read\_csv('./people. csv', encoding='utf-8', header=None, sep=',')

peoples\_df. head (5)

Khi chỉ định không có header, dòng header của chúng ta đã biến thành 1 bản ghi dữ liệu:



Hình 17: Minh họa đọc file csv

# Chương 2 : Bài toán dự báo

## 1. Bài toán dự đoán

Dự đoán là quá trình tạo ra nhận định về các hiện tượng mà thông thường các đầu ra của chúng còn chưa quan sát được.

Dự đoán là một trong những yếu tố quan trọng nhất trong việc ra quyết định quản lý bởi vì ảnh hưởng sau cùng của một quyết định thường phụ thuộc vào sự tác động của các nhân tố không thể nhìn thấy tại thời điểm ra quyết định. Vai trò của dự đoán là nhậy cảm trong các lĩnh vực như tài chính, nghiên cứu thị trường, lập kế hoạch sản xuất, hành chính, điều khiển quá trình sản xuất hay nghiên cứu,…

Trong giới doanh nhân, các câu thường xuyên được đưa ra là:

Lượng hàng bán trong thời gian tới là bao nhiêu?

Tháng này nên đặt bao nhiêu hàng?

Nên giữ bao nhiêu cổ phiếu?

Nên mua bao nhiêu nguyên liệu?

Mục tiêu bán hàng sắp tới là gì?

Có nên tăng nhân công không?

### 1.1 Dự đoán hỗ trợ quá trình ra quyết định trông các tình huống

Điêu tiết nguồn tài nguyên sẵn có : Dự báo nhu cầu cho sản phẩm, nguyên liệu, nhân công, tài chính hay dịch vụ như là một đầu vào thiết yếu để điều tiết kế hoạch sản xuất, vận tải, tiền vốn và nhân lực.

Yêu cầu thêm tài nguyên : dự báo giúp xác định tài nguyên cần có trong tương lai (như nhân lực, máy móc thiết bị, vốn …)

Thiết kế, lập quy hoạch : Dự báo các hiện tượng thiên nhiên như lũ lụt, hạn hán để thiết kế các công trình như đê, đập, hồ chứa và quy hoạch vùng sản xuất. Nhược điểm của dự báo là không thể tránh khỏi sai số. trên quan điểm thực tiễn cần hiều rõ các mặt mạnh lẫn mặt hạn chế của các phương pháp dự báo và tính đến chúng trong khi sử dụng dự báo.

### 1.2 Chuỗi thời gian

Chuỗi thời gian là một dãy dữ liệu được quan sát ở các thời điểm thiết kế tiếp nhau với cùng một đơn vị đo mẫu,

Trong chuỗi thời gian, trình tự thời gian đóng một vai trò thực sự quan trọng, vì vậy các tính toán thông kê thông thường như trung bình mẫu, độ lệch quân phương mẫu, khoảng tin cậy, kiểm định các giải quyết, … không còn thích hợp.

Một chuỗi thời gian bao gồm những thành phần nào sau đây:

#### Thành phần xu thế

Xu hướng thể hiện sự tăng trưởng hoặc giảm sút của một biến số theo thời gian đủ dài. Một số biến cố kinh tế có xu hướng tăng giảm dài hạn như:

Tốc độ tăng dân số của Việt Nam có xu hướng giảm.

Tỷ trọng nông nghiệp trong GDP[[10]](#footnote-10) của Việt Nam có xu hướng giảm.

Mức giá có xu hướng tăng.

#### Thành phần mùa (thời vụ).

Biến động thời vụ của biến số kinh tế là sự thay đổi lặp đi lặp lại từ năm này sang năm năm khác theo mùa vụ. Biến động thời vụ xảy ra do khí hậu, ngày lễ, phong tục tập quán… biến động thời vụ có tính ngắn hạn với chu kỳ lặp lại thương là một năm.

#### Thành phần ngẫu nhiên.

Những dao động không thuộc ba loại trên được xếp vào dao động ngẫu nhiên. Các nguyên nhân gây ra biến động ngãu nhiên có thể thời tiết bất thường, chiến tranh, khủng hoảng năng lượng, biến động chính trị…

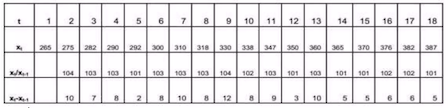
#### hành phần chu kỳ.

Các số liệu kinh tế vĩ mô thường có sự tăng giảm có quy luật theo chu kỳ kinh tế. Sau một thời kỳ suy thoái kinh tế là thời kỳ phục hồi vào bùng nổ kinh tế, kế tiếp tăng trưởng kinh tế sẽ chựng lại và khỏi đầu cho một cuộc suy thoái mới. Tùy theo nền kinh tế mà chu kỳ kinh tế có thời hạn là 5 năm, 7 năm hay 10 năm.

### 1.3 Các phương pháp hiển thị chuỗi thời gian

Phân tích chuỗi thời gian bao gồm việc nghiên cứu dạng dữ liệu trong quá khứ và giải thích các đặc điểm chính của nó. Một trong những phương pháp đơn giản và hiệu quả nhất là hiển thị trực quan chuỗi đó. Các đặc điểm không thấy trong bảng dữ liệu thường nổi lên qua các minh họa đồ thị.

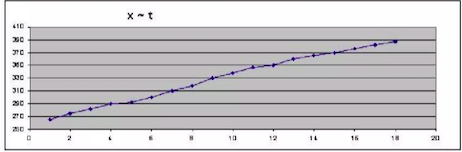
Ví dụ cho bẳng dữ liệu sau:



Hình 18: Minh họa cho bảng dữ liệu

#### 1.3.1 Đồ thị của xt theo t:

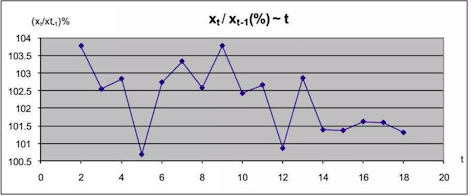
Cung cấp lịch sử dữ liệu gốc chưa bị chuyển đổi qua bất cứ phép biến đổi nào, giúp cho việc nghiên cứu xu thế và nhận dạng



Hình 19: Đồ thị xt theo t

#### 1.3.2 Đồ thị của xt / xt-1 \* 100 theo t :

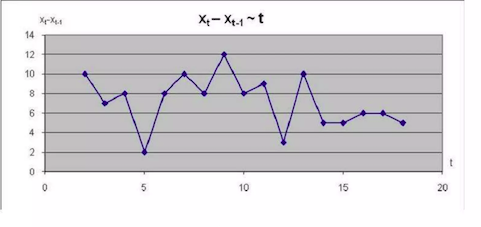
Mỗi điểm trên đồ thị này cho biết giá trị hiện thời của chuỗi tăng hay giảm so với giá trị trước đó.



Hình 20: Đồ thị của xt / xt-1 \* 100 theo t

#### 1.3.3 Đồ thị của Xt – Xt-1 ~ t :

Đồ thị này biểu diện sự thay đổi giữa các bước thời gian kế tiếp nhau. Nhìn vào đồ thị ta thấy được khoản các giá trị biết đổi giữa các bước kề nhau.



Hình 21: Đồ thị của Xt – Xt-1 ~ t

#### 1.3.4 Các định dạng dữ liệu :

Trước khi áp dụng bấc cứ một phương pháp dự báo khoa học cho một tình huống nào, cần phải ghép nối các thông tin (dữ liệu có liên quan) về tình huống đó càng nhiều càng tốt. Những dữ liệu đó được phân thành 2 loại:

Các dữ liệu bên trong, ví dụ số liệu sản phẩm bán ra trong quá khứ, …

Các dữ liệu bên ngoài, ví dụ như các thống kê của ngân hàng về tình tài chính của công ty (phản ánh thông tin bên trong).

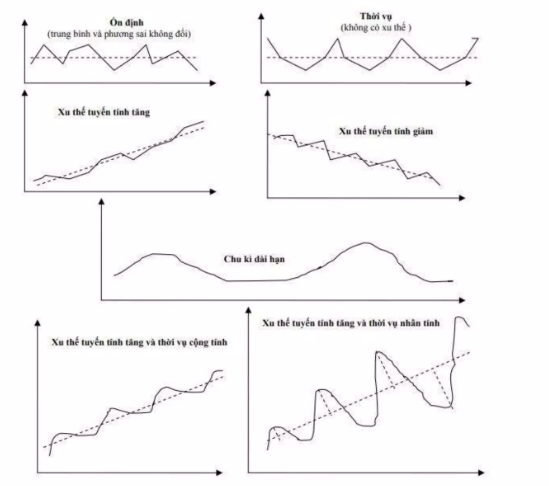
Từ các thông tin này, người làm dự báo phải chọn ra thông tin liên quan nhiều nhất đến tình huống cần dự báo. Chẳng hạn, trong dự báo bán hàng, báo cáo bán hàng được trong quá khứ của công ty sẽ cung cấp những thông tin tối thiểu cho việc dự báo.

Thông tin tối thiểu cần thỏa các yêu cầu:

Tính liên quan : Nó có phải là thông tin liên quan trực tiếp hay không?

Độ tin cậy : Dữ liệu được thu nhập như thế nào? Có đáng tin cậy hay không?

Tính thời sự : Liệu các thông tin mới nhất đã được cập nhật chưa? Chúng có sẵn khi cân hay không?

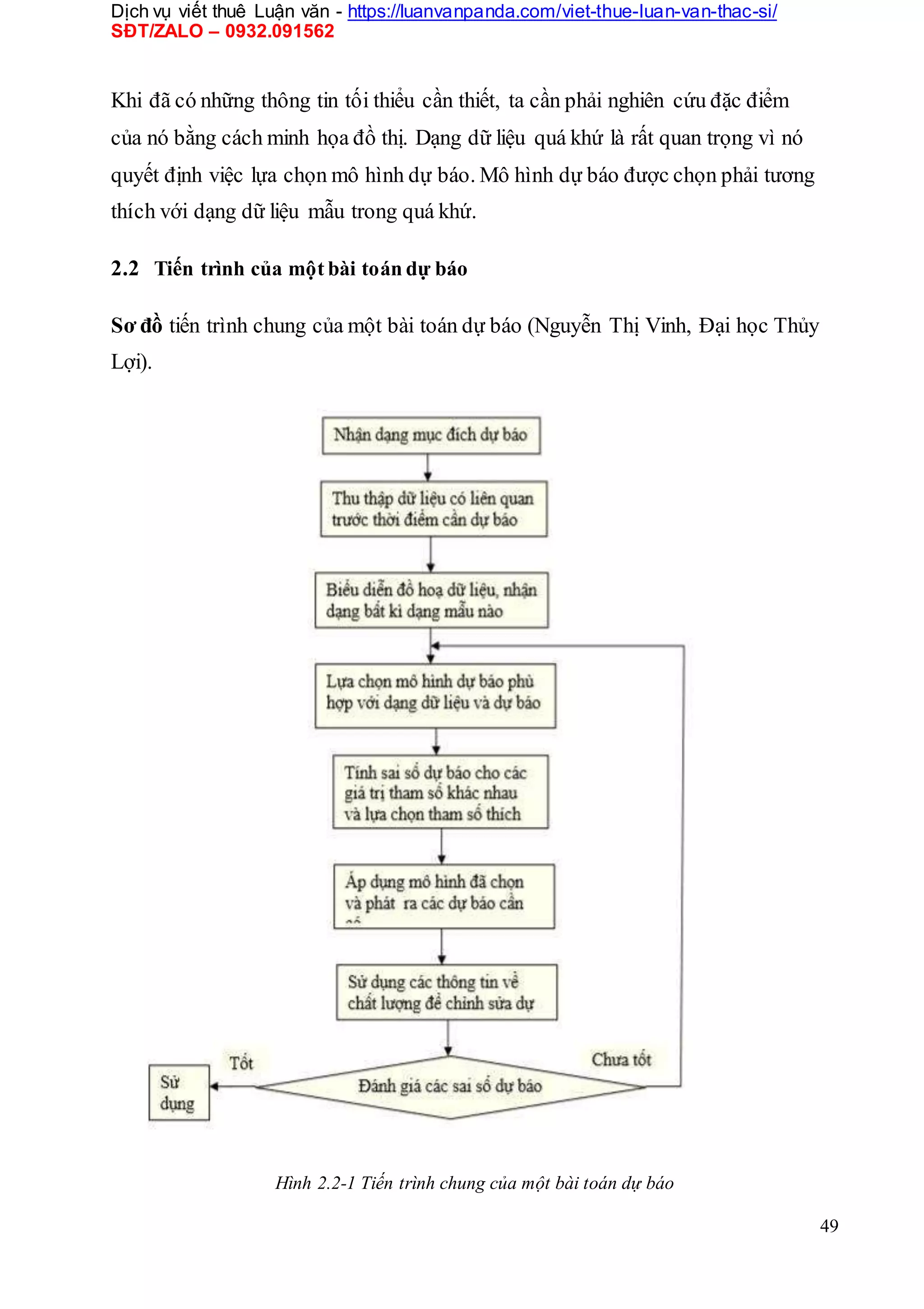


Hình 22: Một số định dạng dữ liệu

Khi đã có những htoong tin tối thiểu cần thiết, ta cần phải nghiên cứu đặc điểm của nó bằng cách minh họa đồ thị. Dạng dữ liệu quá khứ rất quan trọng vì nó quyết định việc lựa chọn mô hình dự báo. Mô hình dự báo được chọn phải tương thích với dangj dữ mẫu trong quá khứ.

## 2. Tiến trình của một bài toán dự báo

Sơ đồ tiến trình chung của một bài toán dự báo



Hình 23: Tiến trình chung của một bài toán adujw báo

## 3. Tiêu chuẩn dự báo

Các tiêu chuẩn chung đánh giá sự thành công của một mô hình dự báo khi áp dự và một tập dữ liệu là:

\_ Trùng càng nhiều với các thay đổi ngẫu nhiên trong tập dữ liệu càng càng tốt

\_ Không vượt quá xa bất kỳ một một đặc tính nào của dữ liệu

Xét về mặt sai số, hai hoặc đặc tính cần quan tâm khi thử nghiệm một công thức dự báo trên dữ liệu là:

Các đặc tính thống kê : Một phương pháp dự báo tốt thường cho sai số trung bình nhỏ. Trong các mô hình dự báo, người ta thương sử dụng các loại sai số như

MAE = (Mean Absolute Error)

MSE = (Mean Square Error)

RMSE = (SquareRoot Mean Square Error)

Các đặc tính định dạng : Trong các mô hình dự báo, sự có mặt của sai số (tính lệch, tính chu kỳ, tính kiên định, …) điều bị xem là dấu hiệu không tốt. Sự xuất hiện của bất cứ xu thế nàotrong sai số cũng nên khử càng nhanh càng tốt. Có thể sai phân hóa chuỗi các giá trị ban đầu để đối phó với các tác động này.

Như vậy có hai tiêu chuẩn dự báo về định lượng và định tính là : sai số nhỏ và không tuân theo một định dạng nào.

# Chương 3 : Bài toán Tìm hiểu ứng dụng cộng nghệ AI trong dự đoán thị trường chứng khoán

## 1. Giới thiệu bài toán dự báo giá cổ phiếu

Đầu tư tài chính là tròng những cách thức kiếm tiền khá phổ biến hiện nay. Bằng việc phân tích các yếu tố cơ bản hay yếu tố định lượng để dự báo sự tăng giả của giá cổ phiếu, các nhà đầu tư sẽ ra quyết định mua hay bán cổ phiếu nhằm tiềm kiếm lợi nhuận. tuy nhiên, thị trường chứng khoán luôn có những yếu tố bất ngờ không theo ý chủ quan của người dùng, do đó dự đoán xu thế của thị trường chứng khoán là bài toán qua trọng đã và đang thu hút được sự quan tâm của nhà đầu tư, các chuyên gia kinh tế học và các nhà khoa học. Bên cạnh đó sự phát triển mạnh mẻ của công nghệ thông tin giúp cho việc trao đổi và lưu trữ dữ liệu về chững khoán dễ dàng hơn bao hết. Người dùng có thể tra cứu dễ dàng lịch sử tăng giảm của các mã cổ phiếu mà học quan tâm. Mặt khác nhiều công cụ và ứng dụng được phát triển nhằm khai thác và xử lý dữ liệu tài chính khổng lồ cho các mô hình dự báo đã làm cho này trở nên dễ dàng, nhanh và chính xác.

Dự báo giá cổ phiếu luôn là bài toán thách thức vì tính không thể đoán trước của nó. Giả thiết thị trường hàng ngày là không thể dự báo giá cổ phiếu và giá cổ phiếu là ngẫu nhiên. Tuy nhiên, bằng cách phân tích kỹ thuật cho thấy hầu hết các giá cổ phiếu được phản ánh dữ liệu trước đó, do đó xu hướng vận động giá là rất quan trọng để dự đoán giá hiệu quả. Hơn nữa, các nhóm cổ phiếu và thị trường chứng khoán bị ảnh hưởng bởi một số yếu tố kinh tế như sự kiện chính trị, điều kiện kinh tế chung, chỉ số giá hàng hóa, kỳ vọng của nhà đầu tư, sự vận động của thị trường chứng khoán khác, tâm lý của nhà đầu tư,… Giá trị của các nhóm chứng khoán được tính toán với vốn hóa thị trường cao. Có các thông số kỹ thuật khác nhau để xác định được dữ liệu thống kê từ giá trị của cổ phiếu. Thông thường, các chỉ số chứng khoán đạt được từ giá của các cổ phiếu có đầu tư thị trường thường rất cao và từ đó đưa ra đánh giá về tình trạng kinh tế ở mỗ quốc gia.

Bản chất của sự vận động giá cổ phiếu là mô hồ và làm cho các nhà đầu tư có tính rủi ro cao. Trên thực tế, giá chứng khoán nói chúng là động, không tham số và phi tuyến tính, do đó chúng thường giảm hiệu năng của các mô hình thông kê và không thể dự đoán giá và sự vận động chính xác.

Học máy là công cụ mạnh nhất bao gồm các thuật toán khác nhau để tăng hiệu năng trong một số trường hớp nhất định. Học máy có khả năng xác định thông tin và phát hiện các mẫu từ bộ dữ liệu. Các mô hình học nhóm dựa trên học máy như học tăng cường (boosting) học bao đóng (bagging), cho kết quả vượt trội so với từng phương pháp khi dự đoán chuỗi thời gian. Gần đây các mô hình dựa trên cây như thuật toán tăng cường độ dốc (Deep Learning) có khả nagw tuyệt vời để trích xuất các thồng tin từ chuỗi thời gian tài chính. Mạng nơ ron hồi quy (RNN) đã đạt được thàng cô đáng kể trong lĩnh vực phân tích tài chính với hiệu năng cao. Có thể thấy, quá trình dự đoán của thị trường chứng khoán không chỉ liên quan đến thông tin mà dữ liệu trước đó đóng vai trò rất quan trọng. Việc huấn luyện mô hình sẽ không đủ nếu chỉ sử dụng dữ liệu vào thời điểm gần nhất. RNN có thể sử dụng mạng để duy trì bộ nhớ của các sự kiện gần đây và xây dựng các kết nối giữa mỗi đơn vị của mạng, do đó, nó hoàn toàn phù hợp với các dự báo kinh tế. Mô hình bộ nhớ dài ngắn (LSTM) là một cải tiến của phương pháp RNN được sử dụng trong lĩnh vực học sâu. LSTM có ba đầu ra khác nhau để loại bỏ các vấn đề trong các nơ ron RNN và cũng có thể xử lý các điểm dữ liệu đơn lẻ hoặc toàn bộ chuỗi dữ liệu. nhiều nghiên cứu sử dụng các phương pháp này đã được sử dụng trong dự báo thị trường.

Đã có nhiều phương pháp dự báo đã và đang được sử dụng cho dự báo su biến động của giá chứng khoán. Các tiếp cận thống kê như: làm mịn theo cấp số nhân (exponential smooothing), trung bình động (moving average) và ARIMA là các mô hình tuyến tính phổ biến trong kinh tế để dự đoán xu thế vận động của giá trong tương lai. Bên cạnh đó, một số nghiên cứu cho bài toán dự báo như Mạng nơ ron nhân tạo (ANN), giải thuật di truyền (GA), logic mờ, một số phương pháp học máy thống kê hiện đại như RNN, LSTM, v.v. đã được nghiên cứu trong thời gian gần đây có khả năng xử lý các biến động của thị trường chứng khoán với độ chính xác dự đoán cao.

## 2. Phát biểu bài toán

Thị trường chứng khoán Việt Nam đã có một chặng đường dài phát triển với hơn 20 năm hoạt động. Trong thời gian đó, tổng sản phẩm quốc nội (GDP) bình quân đầu người tăng khoảng 7 lần và Việt Nam từ một nước nghèo đã trở thành một quốc gia có thu nhập trung bình. Đất nước hình chữ S đã tham gia một cách sâu rộng vào nền kinh tế thế giới với việc bình thường hóa quan hệ thương mại với Mỹ, gia nhập tổ chức thương mại thế giới WTO[[11]](#footnote-11) và ký hàng loạt các hiệp định thương mại tự do với hầu hết các cường quốc trên thế giới, trong đó có cả các hiệp định thế hệ mới như CPTPP, EVFTA, EVIPA, v.v. Do đó, có thể thấy những kết quả đạt được của thị trường chứng khoán khi thể hiện rõ vai trò là kênh dẫn vốn quan trọng của nền kinh tế, theo nhiều góc độ về định lượng như quy mô vốn hóa, cơ sở nhà đầu tư, số lượng sản phẩm, quy mô giao dịch hàng ngày, v.v. hay định tính như tính chuyên nghiệp của các thành viên, v.v.

Sau 20 năm vận hành, quy mô thị trường và đặc biệt các công ty chứng khoán (CTCK) đã có nhiều thay đổi cả về số lượng, quy mô lẫn chất lượng, sản phẩm cung cấp cho nhà đầu tư. Số lượng CTCK tăng vọt, quy mô vốn tăng hàng trăm lần so với thời điểm mới thành lập cùng với sự xuất hiện của nhiều CTCK ngoại. Phí giao dịch ngày càng thấp, nhiều CTCK miễn phí giao dịch. Nhiều sản phẩm mới ra đời, áp dụng công nghệ vào hoạt động hỗ trợ nhà đầu tư giao dịch thay vì viết phiếu lệnh bằng cách thực hiện Online[[12]](#footnote-12). Các báo cáo phân tích được thực hiện chuyên nghiệp, nhiều công ty thực hiện báo cáo bằng tiếng Anh để hỗ trợ cho NĐT nước ngoài. Các CTCK không chỉ cung cấp thuần dịch vụ môi giới mà còn có nhiều nghiệp vụ khác như Ngân hàng đầu tư (IB), tư vấn niêm yết, bảo lãnh phát hành trái phiếu, cung cấp các sản phẩm phái sinh (HĐTL chỉ số VN30, HĐTL trái phiếu chính phủ), v.v. Không những vậy, nhiều CTCK cũng đã áp dụng công nghệ vào hoạt động môi giới cũng như tư vấn đầu tư. Việc có thể dễ dàng tham gia thị trường và tìm kiếm thông tin đã tạo ra sức hút rất lớn đối với nhà đầu tư. Nhiều trang web cho phép người dùng có thể xem và tải các dữ liệu lịch sử về giá cũng như các báo cáo hoạt động của các công ty niêm yết trên thị trường một cách dễ dàng như: cafef. vn, cophieu68. vn, investing. com, tradingview. com, v.v. Giúp các nhà đầu tư có dữ liệu để thực hiện các phân tích kỹ thuật cũng như phân tích định lượng nhằm xác định xu hướng giá cổ phiếu mà mình quan tâm trong các hoạt động đầu tư của mình.

Trong nghiên cứu này, em sẽ thực hiện phân tích dự báo sự tăng giảm của giá cổ phiếu trong thị trường chứng khoán việt nam dựa trên phân tích chuỗi thời gian. Xây dựng mô hình thực hiện yêu cầu sau:

***Input:*** Cho giá của mã chứng khoán ở ngày t

***Output:*** Giá của mã chứng khoán đó trong ngày t+1

## 3. Một số mô hình dự báo thồn kê

### 3.1 Mô hình trượt đơn Moving Average (MA) để phân tích dự báo giá chứng khoán.

Mô hình trung bình trượt đơn (Moving average) là cách tiếp cận ngây thơ nhất đối với mô hình chuỗi thời gian.

Mô hình này thực hiện quan sát tiếp theo là giá trị trung bình của tất cả các quan sát trong quá khứ.

Mô hình trượt đơn có thể sử dụng để xác định các xu hướng quan tâm trong dữ liệu. Ta có thể định nghĩa một cửa sổ trượt để áp dụng mô hình trượt để làm trơn cho các chuỗi thời gian và đánh dấu các xu hướng khác nhau.

= ( + + … + ) / N

= ( + + … + ) / N + / N – / N

Hay = + / N – / N

Dự báo ở thời điểm t +1 chỉ là điều chỉnh của dự báo ở thời điểm t trước đó.

Khi N tăng đủ lớn thì lượng điều chỉnh xt / N – Xt - N / N→0 và trung bình trượt trở thành trung bình mẫu như phương pháp ngây thơ.

Chỉ nên áp dụng phương pháp này khi số giá trị quan sát được là ít và tập dữ liệu có tính ổn định theo thời gian.

Mặt khác, mô hình trượt đơn có thể sử dụng để xác định các xu hướng quan tâm trong dữ liệu. Ta có thể định nghĩa một cửa sổ trượt để áp dụng mô hình trượt để làm trơn cho các chuỗi thời gian và đánh dấu các xu hướng khác nhau.

### 3.2 Mô hình làm mịn theo cấp số nhân (exponential smoothing).

Mô hình làm mịn theo cấp số nhân “exponential smoothing” là một phương pháp dự báo chuỗi thời gian cho dữ liệu đơn biến.

Mô hình làm mịn theo cấp số nhân thực hiện tương tự như mô hình trung bình động, nhưng trong đó sử dụng trọng số giảm dần khác nhau được gán cho mỗi quan sát. Nói cách khác, các quan sát lâu hơn (về quá khứ) hơn xo với hiện tại được đánh giá là ít quan trọng hơn.

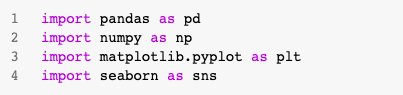
Khi đó, công thức làm trơn theo cấp số nhân được thực hiện như sau:

|  |
| --- |
| y = + (1 – ) , t > 0 |

Trong đó, là hệ số làm mịn nhận các giá trị từ 0 đến 1. Nó quyết định trọng số giảm nhanh như thế nào đối với các quan sát trước đó. Alpha () : Hệ số làm mịn cho cấp độ.

# Chương 4 : Ứng dụng công nghệ AI trong thị trường chứng khoán

## 1. Các thư viện



Hình 24: Các thư viện

Thư viện pandas trong python là một thư viện mã nguồn mở, hỗ trợ đắc lực trong thao tác dữ liệu. Đây cũng là bộ công cụ phân tích và xử lý dữ liệu mạnh mẽ của ngôn ngữ lập trình python. Thư viện này được sử dụng rộng rãi trong cả nghiên cứu lẫn phát triển các ứng dụng về khoa học dữ liệu. Thư viện này sử dụng một cấu trúc dữ liệu riêng là Dataframe[[13]](#footnote-13). Pandas cung cấp rất nhiều chức năng xử lý và làm việc trên cấu trúc dữ liệu này. Chính sự linh hoạt và hiệu quả đã khiến cho pandas được sử dụng rộng rãi.

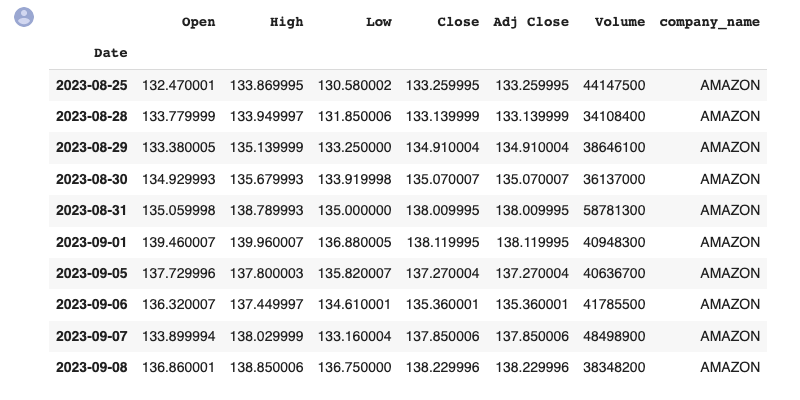
Để thực hiện các suy luận thống kê cần thiết, cần phải trực quan hóa dữ liệu của bạn và Matplotlib là một trong những giải pháp như vậy cho người dùng Python. Nó là một thư viện vẽ đồ thị rất mạnh mẽ hữu ích cho những người làm việc với Python và NumPy. Module được sử dụng nhiều nhất của Matplotib là Pyplot cung cấp giao diện như MATLAB nhưng thay vào đó, nó sử dụng Python và nó là nguồn mở.

thư viện Python Seaborn được sử dụng để giảm bớt nhiệm vụ trực quan hóa dữ liệu đầy thách thức và nó dựa trên Matplotlib. Seaborn cho phép tạo đồ họa thống kê thông qua các chức năng sau:

* Một API[[14]](#footnote-14) dựa trên tập dữ liệu cho phép so sánh giữa nhiều biến
* Hỗ trợ lưới nhiều ô để dễ dàng xây dựng các hình ảnh trực quan phức tạp
* Hình ảnh hóa đơn biến và lưỡng biến có sẵn để so sánh giữa các tập hợp con dữ liệu
* Có sẵn các bảng màu khác nhau để hiển thị các loại mẫu khác nhau
* Dự toán và âm mưu tuyến tính hồi quy tự động

## 2. Dữ liệu

Dữ liệu của 4 công ty lớn là Apple, Google, Facebook, Amazon sử dụng thư viện yfinance



Hình 25: Minh họa dữ liệu

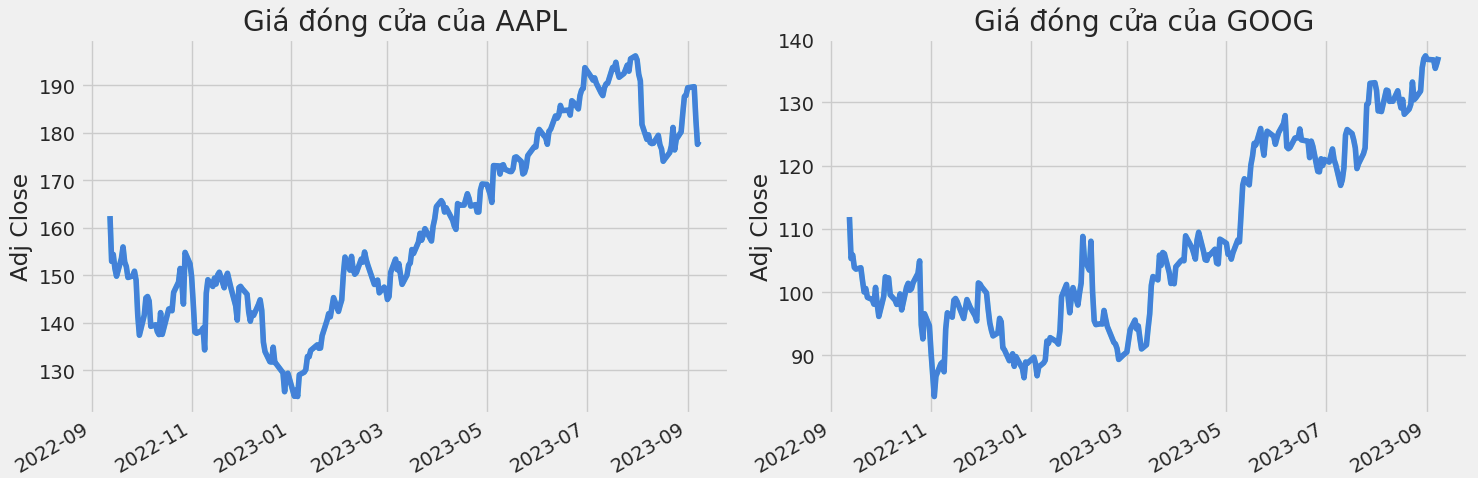
## 3. Thống kê

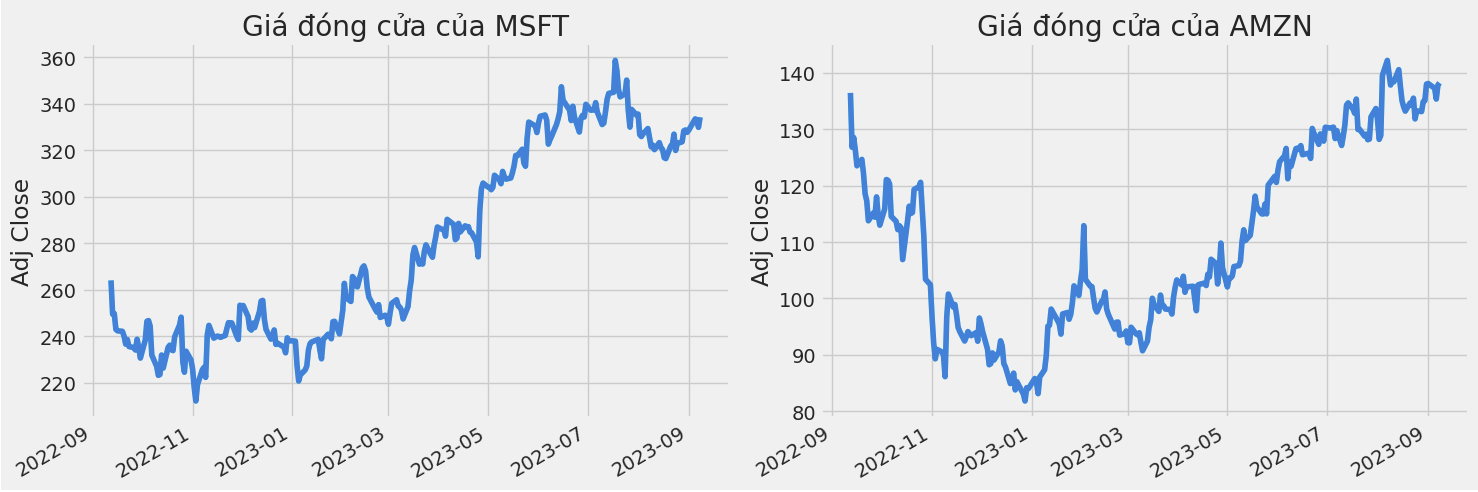
Tóm tắt thống kê của các dữ liệu được lấy về từ thư viện yfinance



Hình 26: Minh họa thống kê dữ liệu

## 4. Lịch sử giá đóng của của các công ty :

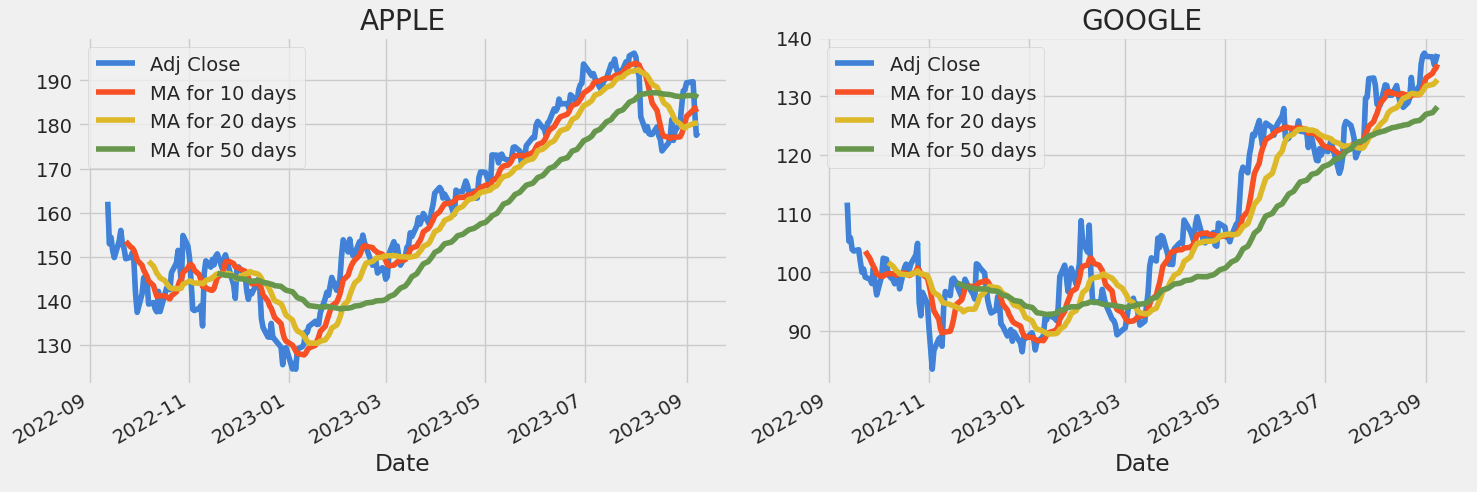


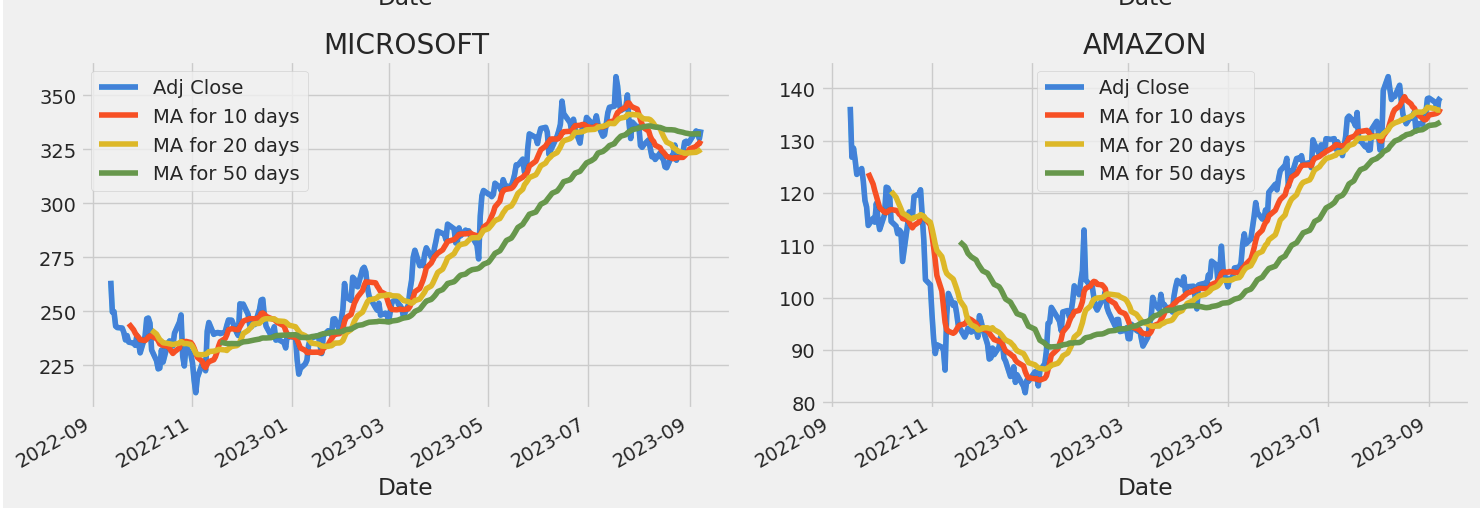


Hình 27: Lịch sử giá đóng cửa của các công ty

## 5. Biểu đồ

Biểu đồ tổng khối lượng cổ phiếu được giao dịch mỗi ngày





Hình 28: Kết quả giao dịch

Chú thích đồ thị :

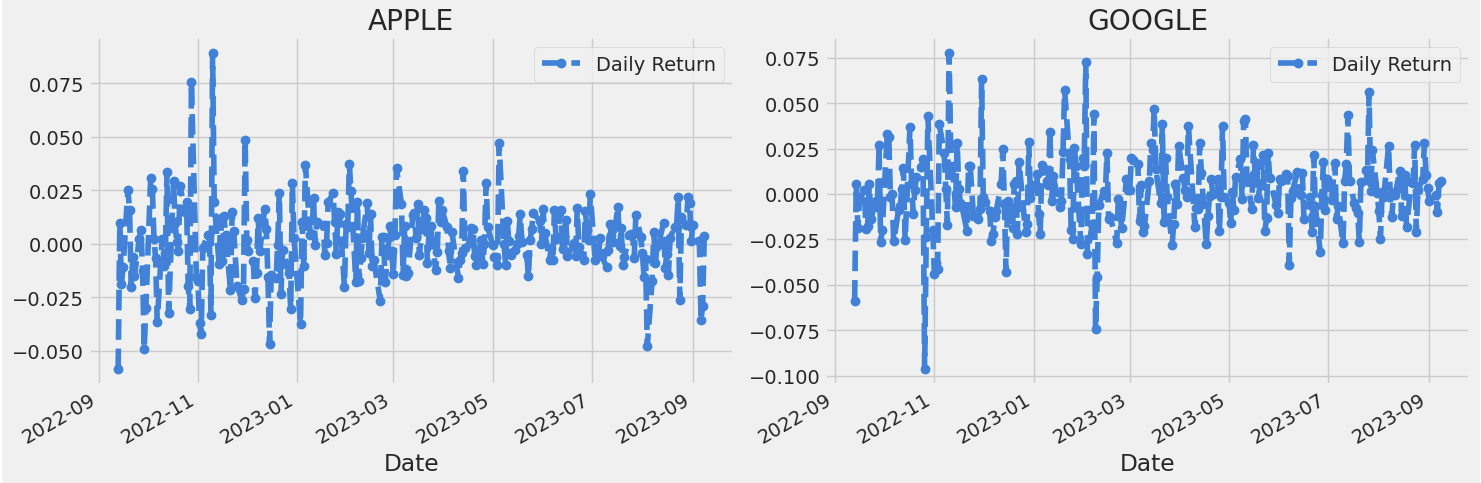
Adj Close : Điều chỉnh đóng

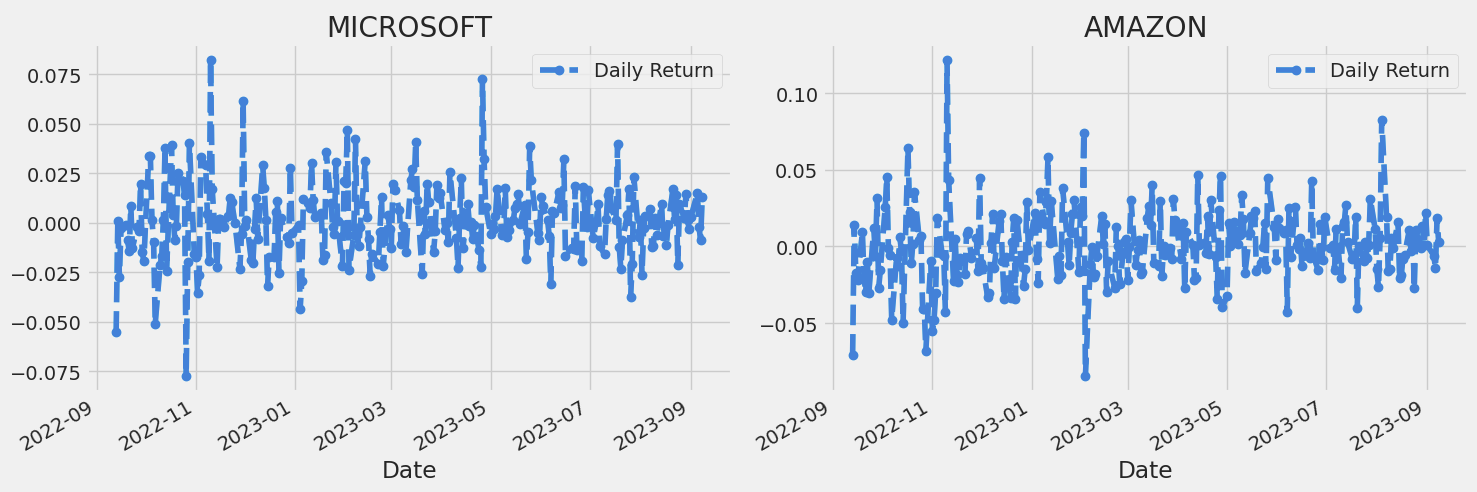
MA for 10 days : Giá biến động trong 10 ngày qua

MA for 20 days : Giá biến động trong 20 ngày qua

MA for 50 days : Giá biến động trong 50 ngày qua

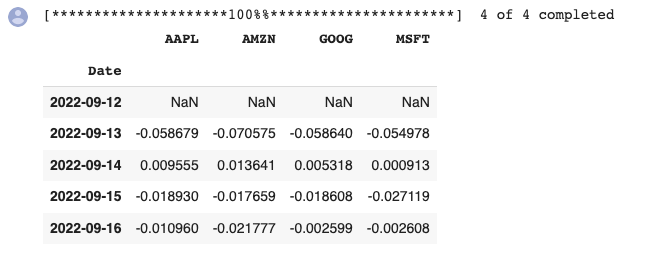
Biểu đồ tỷ lệ hoàn vốn hàng ngày





Hình 29: Biểu đồ hoàn vốn mỗi ngày

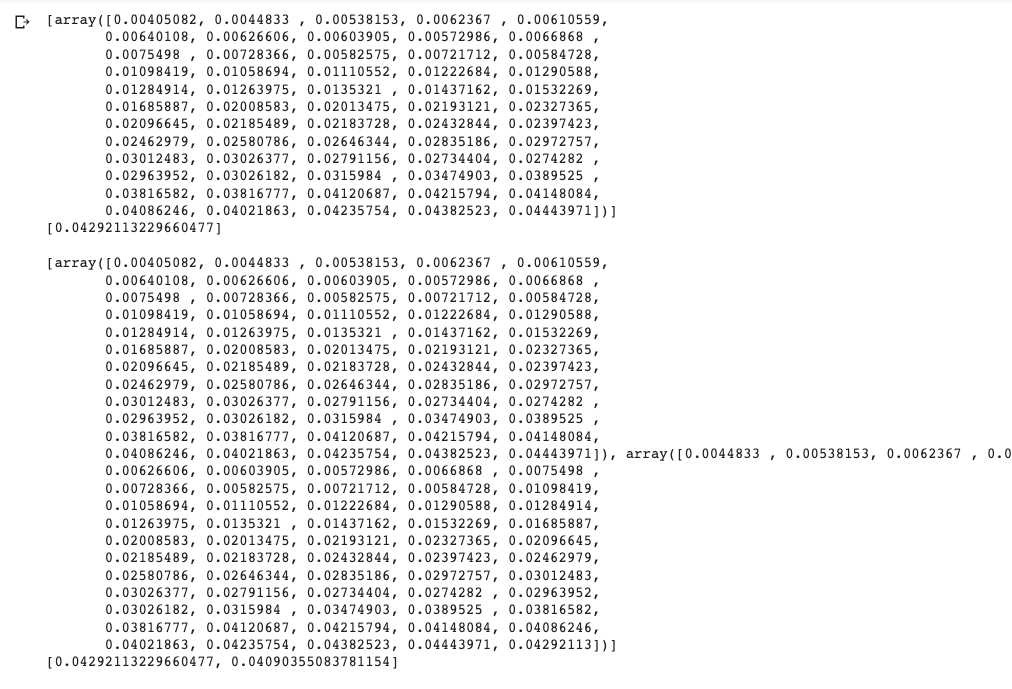
Danh sách giá đóng cửa của các cổ phiếu công nghệ vào khung dữ liệu



Hình 30: Khung giá đóng cửa

## 6. Huấn luyện

Tạo tập dữ dữ liệu huấn luyện, chia tỷ lệ dữ liệu để huấn luyện, sau đó tách dữ liệu để train, và chuyển đổi dữ liệu thành mảng.



Hình 31: Huấn luyện dữ liệu

## 7. Mô hình huấn luyện

Tạo một mảng mới chứa các giá trị được chia tỷ lệ từ chỉ số 1543 đến 2002 để train. Sau đó tạo bộ dữ liệu test rồi chuyển dữ liệu thành mảng có nhiều chiều để dự đoán các giá trị.



Hình 32: Biểu đồ mô hình huấn luyện

Chú thích đồ thị:

Màu xanh : Mô hình huấn luyện

Màu đỏ : Giá thực tế trên thị trường

Màu cam : Giá dự đoán

Nhận xét : như vậy mô hình huấn đã dự đoán gần đúng so với giá thực tế.

# Kết luận

Sau thời gian khoảng 2 tháng tham thảo tìm tòi và dưới sự hướng dẫn của thầy Phạm Đình Tài về môn Đồ án cơ sở trí tuệ nhân tạo. Nhóm chúng em viết chương trình thử nghiệm cho đề tài “Tìm hiểu ứng dụng cộng nghệ AI trong dự đoán thị trường chứng khoán”. Trong thời gian không đủ nên nhóm chúng còn nhiều sai xót hoặc chưa đúng chúng em mong thầy và ban giám thị thông cảm, bên canh đó đồ án nhóm chúng em đã đạt được những kết quả như sau :

Tìm hiểu về ngôn ngữ lập trình python

Tìm hiểu về môi trường lập trình Google Colaratory

Tìm hiểu về bài toán dự báo

Viết chương trình thử nghiệm.

Tuy nhiên trong quá trình thực hiện đồ án nhóm chúng em vẫn chưa hiểu sâu nên còn nhiều điều thiếu xót, chúng em sẽ cãi thiện và trau dồi kỹ năng ngày tốt hơn.

Nhóm chúng em xin chân thành cảm ơn thầy và khoa công nghệ thông tin!

# Danh mục tài liệu kham thảo

1, Dữ liệu để train : finance.yahoo.com

2, <https://www.w3schools.com/>

3, *Cách lấy giá cổ phiếu thời gian thực bằng python*, tạp chí VniTeach – Giáo viên 4.0, Năm 2023.

<https://www.vniteach.com/2023/02/16/cach-lay-gia-co-phieu-thoi-gian-thuc-bang-python/>. [Truy cập ngày 10/07/2023].

4, *Trí tuệ nhân tao: kỷ nguyên mới cho thị trường chứng khoán*, Tạp chí DNTECH.

<https://dntech.vn/tri-tue-nhan-tao--ky-nguyen-moi-cho-thi-truong-chung-khoan-171-26.html>. [Truy cập ngày 13/07/2023].

5, Đài Trung, *Ứng dụng của python trong tài chính*, Tạp chí TaiChinhHub, Năm 2020.

<https://taichinhhub.com/ung-dung-python-trong-tai-chinh/>. [Truy cập ngày 03/08/2023]

1. *Bộ mã chuẩn quốc tế* [↑](#footnote-ref-1)
2. *Trình soạn thảo văn bản* [↑](#footnote-ref-2)
3. *Môi trường phát triển tích hợp* [↑](#footnote-ref-3)
4. *Dự án phát triển phần mềm mã nguồn mỡ* [↑](#footnote-ref-4)
5. *Bộ phận sử lý đồ họa* [↑](#footnote-ref-5)
6. *Trí tuệ nhân tạo* [↑](#footnote-ref-6)
7. *Trì soạn thảo mã nguồn* [↑](#footnote-ref-7)
8. *Kho lưu trữ mã nguồn* [↑](#footnote-ref-8)
9. Bộ dữ liệu [↑](#footnote-ref-9)
10. *Tổng sản phẩm nội địa* [↑](#footnote-ref-10)
11. *Tổ chứ thương mại quốc tế* [↑](#footnote-ref-11)
12. *Trực tuyến* [↑](#footnote-ref-12)
13. *Cấu trúc dữ liệu hai chiều* [↑](#footnote-ref-13)
14. *Giao diện lập trình ứng dụng* [↑](#footnote-ref-14)