**Biểu đồ đường và chuỗi thời gian**

Nội dung

[1.](#_heading=h.gjdgxs) Giới thiệu 1

[2.](#_heading=h.30j0zll) Đồ thị 2

[3.](#_heading=h.1fob9te) Biểu đồ Đường 6

[4.](#_heading=h.3znysh7) Matplotlib · 10

[5.](#_heading=h.2et92p0) Tùy chỉnh biểu đồ 13

[6.](#_heading=h.tyjcwt) Dữ liệu chuỗi thời gian của WHO 15

[7.](#_heading=h.3dy6vkm) Các loại tăng trưởng 17

[8.](#_heading=h.1t3h5sf) Các loại thay đổi 21

[9.](#_heading=h.4d34og8) So sánh biểu đồ đường 23

# Giới thiệu

Trọng tâm của bất kỳ quy trình làm việc khoa học dữ liệu nào là khám phá dữ liệu. Thông thường nhất, chúng ta khám phá dữ liệu bằng cách sử dụng các mục sau:

* Phương pháp thống kê (đo lường trung bình, đo lường sự thay đổi, v.v.)
* **Trực quan hóa dữ liệu** (chuyển đổi dữ liệu thành một biểu mẫu trực quan)

Điều này chỉ ra rằng một trong những nhiệm vụ trọng tâm của trực quan hóa dữ liệu là giúp chúng ta khám phá dữ liệu.

Nhiệm vụ trung tâm khác là giúp chúng tôi giao tiếp và giải thích kết quả mà chúng tôi đã tìm thấy thông qua việc khám phá dữ liệu. Điều đó đang được nói, chúng ta có hai loại trực quan hóa dữ liệu:

* Trực quan hóa dữ liệu khám phá: chúng ta xây dựng biểu đồ cho *chính mình* để khám phá dữ liệu và tìm các mẫu.
* Trực quan hóa dữ liệu giải thích: chúng ta xây dựng biểu đồ để *những người khác* giao tiếp và giải thích các mẫu mà chúng ta đã tìm thấy thông qua việc khám phá dữ liệu.

Graphical user interface, application

Description automatically generated

Chúng ta sẽ tìm hiểu những điều sau:

* Cách trực quan hóa dữ liệu chuỗi thời gian với các biểu đồ dòng.
* Mối tương quan là gì và làm thế nào để hình dung chúng với các ô phân tán.
* Làm thế nào để trực quan hóa phân phối tần số với các biểu đồ thanh và biểu đồ.
* Cách tăng tốc quy trình trực quan hóa dữ liệu khám phá của chúng tôi với thư viện gấu trúc.
* Cách hình dung nhiều biến bằng cách sử dụng các biểu đồ quan hệ của Seaborn

# Đồ thị

Trước khi chúng ta vào Matplotlib và bắt đầu khám phá một tập dữ liệu, chúng ta sẽ trải qua phần giới thiệu ngắn gọn về đồ thị - chúng là gì và cách xây dựng chúng bằng toán học.

Chúng ta có thể tạo một biểu đồ bằng cách vẽ hai đường thẳng ở các góc vuông với nhau. Mỗi dòng được gọi là một **trục** - đường ngang ở phía dưới là **trục x** và đường thẳng đứng ở bên trái là **trục y**. Điểm mà hai đường giao nhau được gọi là **nguồn gốc.**

**Biểu đồ

Mô tả được tạo tự động**

Mỗi trục có chiều dài - bên dưới, chúng ta thấy cả hai trục được đánh dấu bằng số, đại diện cho độ dài đơn vị.

**Biểu đồ, biểu đồ tần suất

Mô tả được tạo tự động**

Độ dài của các trục giúp chúng ta xác định chính xác bất kỳ điểm nào được vẽ trên biểu đồ. Ví dụ: điểm A trên biểu đồ bên dưới cách trục y bảy đơn vị có độ dài và cách trục x hai đơn vị

**Biểu đồ, biểu đồ tần suất

Mô tả được tạo tự động**

Hai số đại diện cho khoảng cách của một điểm từ trục x và trục y được gọi là **tọa độ**. Điểm *A* ở trên có hai tọa độ: bảy và hai. Bảy là tọa độ x và hai là tọa độ y.

Các tọa độ thường xuất hiện ở dạng (x, y), với tọa độ x đầu tiên. Vì vậy , tọa độ của *A* là (7, 2). Vì vậy, đây là những gì chúng ta cần biết về tọa độ:

* Tọa độ x hiển thị khoảng cách theo độ dài đơn vị so với trục y.
* Tọa độ y hiển thị khoảng cách theo độ dài đơn vị so với trục x.

Độ dài đơn vị của trục x và trục y không nhất thiết phải giống nhau. Bên dưới, chúng ta thấy đơn vị độ dài trên trục x là 10, trong khi trên trục y là 1.000 (lưu ý rằng chúng ta cũng có thể ẩn một số con số để làm cho biểu đồ trông đẹp hơn).

Biểu đồ

Mô tả được tạo tự động

Kiểm tra biểu đồ dưới đây và trả lời các câu hỏi sau:

Biểu đồ, biểu đồ tán xạ

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

Text

Description automatically generated

# Biểu đồ Đường

Trước đây, chúng ta đã trải qua phần giới thiệu nhanh về đồ thị. Trên màn hình này, chúng ta sẽ tạo một biểu đồ bằng cách sử dụng một tập dữ liệu nhỏ.

Dưới đây, chúng ta thấy một bảng cho thấy số ca nhiễm COVID-19 mới được báo cáo trên toàn thế giới trong bảy tháng đầu năm 2020:

Bàn

Mô tả được tạo tự động

Nguồn: <https://covid19.who.int/>

Mỗi hàng hiển thị một cặp hai điểm dữ liệu được kết nối

* Số tháng (trong đó một có nghĩa là tháng một, hai có nghĩa là tháng hai, v.v.)
* Số trường hợp được báo cáo trong tháng đó

Khi chúng ta có một cặp hai số, chúng ta có thể ánh xạ nó trên một biểu đồ bằng cách sử dụng hai số làm tọa độ. Dưới đây, chúng ta thêm một điểm tương ứng với tọa độ (5, 2835147) - tương ứng với tháng XNUMX. Đằng sau bức màn, chúng tôi đã tạo biểu đồ bằng Matplotlib, mà chúng tôi sẽ giới thiệu trên màn hình tiếp theo

Biểu đồ, biểu đồ tán xạ

Mô tả được tạo tự động

Bây giờ chúng ta hãy đặt tất cả dữ liệu vào bảng trên biểu đồ theo cùng một phương pháp:

Biểu đồ, biểu đồ tán xạ

Mô tả được tạo tự động

Khi chúng tôi lập biểu đồ về cách một cái gì đó thay đổi theo thời gian, chúng tôi kết nối tất cả các điểm bằng một đường - ở trên, chúng tôi đã lập biểu đồ số ca nhiễm COVID-19 mới đã thay đổi như thế nào theo tháng.

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Bởi vì chúng tôi sử dụng các đường để kết nối các điểm, biểu đồ trên là một **biểu đồ đường** (còn được gọi là biểu đồ đường hoặc biểu đồ đường; sự khác biệt giữa đồ thị, biểu đồ và biểu đồ là mơ hồ trong thế giới trực quan hóa dữ liệu, với các tác giả khác nhau gán ý nghĩa hơi khác nhau cho các thuật ngữ này - trong khóa học này, chúng tôi sử dụng cả ba từ đồng nghĩa).

Khi chúng ta tạo biểu đồ đường, người ta thường ẩn các điểm trên đường:

Bằng cách nhìn vào biểu đồ đường mà chúng ta đã xây dựng cho bảng ở trên, chúng ta có thể thấy một vài mẫu.

Nhìn chung, dòng này cho thấy một hướng đi lên, cho thấy số lượng các trường hợp mới được báo cáo đã tăng lên hàng tháng và chưa bao giờ giảm hoặc ổn định. Đây chủ yếu là kết quả của việc virus lây lan. Các quốc gia cũng bắt đầu xét nghiệm nhiều người hơn, điều này làm tăng số ca nhiễm mới được báo cáo.

Đường kết nối từ tháng 1 đến tháng 3 có độ dốc tăng vừa phải (đường từ tháng 1 đến tháng 2 gần như nằm ngang), điều này cho thấy sự gia tăng vừa phải về số lượng các trường hợp mới được báo cáo. Trong thời kỳ đó, virus chỉ mới bắt đầu lây lan khắp thế giới và nhiều quốc gia chỉ xét nghiệm cho mọi người khi họ đến bệnh viện.

Đường từ tháng XNUMX đến tháng XNUMX rất dốc, cho thấy sự gia tăng các trường hợp mới được báo cáo. Đường từ tháng 4 đến tháng 5 cho thấy độ dốc nhẹ, vì vậy số ca mắc mới vẫn ở mức cao (khoảng ba triệu). Tuy nhiên, con số này không tăng quá nhiều so với tháng XNUMX - điều này rất có thể là do các lệnh phong tỏa trên toàn thế giới

Đường may-7 rất dốc, cho thấy một sự gia tăng khác về số ca mắc bệnh (từ khoảng ba triệu lên khoảng bảy triệu). Điều này rất có thể là do chấm dứt các biện pháp phong tỏa, điều này tạo điều kiện cho sự lây lan của vi rút nhiều hơn.

Học cách diễn giải đồ thị cũng quan trọng như biết cách xây dựng chúng. Trong bài tập dưới đây, chúng ta sẽ xem xét một biểu đồ đường khác và diễn giải nó. Trên màn hình tiếp theo, chúng ta sẽ học cách xây dựng biểu đồ đường bằng Matplotlib

Dưới đây, chúng ta thấy một biểu đồ đường cho thấy số ca tử vong mới được báo cáo đã phát triển như thế nào theo tháng trong khoảng thời gian từ tháng XNUMX đến tháng XNUMX. Kiểm tra biểu đồ và sau đó đánh giá giá trị sự thật của các câu sau:

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

* 1. Số người chết lớn nhất được ghi lại vào tháng 4, nếu đúng thì gán giá trị Boolean True vào sentence\_1, không thì False
  2. Số người chết ít nhất được ghi lại vào tháng 6, nếu đúng thì gán giá trị Boolean True vào sentence\_2, không thì False
  3. Giai đoạn tháng 4-6 có sự sụt giảm về số người chết so với đỉnh tại tháng 4, nếu đúng thì gán giá trị Boolean True vào sentence\_3, không thì False
  4. Số người chết được ghi lại vào tháng 5 ít hơn tháng 7, nếu đúng thì gán giá trị Boolean True vào sentence\_4, không thì False

Graphical user interface, text, application

Description automatically generated

# Matplotlib ·

Trên màn hình trước, chúng ta đã tìm hiểu về biểu đồ đường, nhưng chúng ta vẫn chưa thảo luận về cách **tạo một biểu đồ bằng code**. Hãy nhớ lại rằng chúng tôi đã kiểm tra biểu đồ đường cho thấy sự tiến triển của các trường hợp mới được báo cáo trong bảy tháng đầu năm 2020.

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Chúng ta có thể tự xây dựng biểu đồ đường này bằng cách sử dụng Matplotlib, một thư viện Python được thiết kế đặc biệt để tạo trực quan hóa. Hãy bắt đầu bằng cách nhập Matplotlib.



Một điều kỳ quặc của Matplotlib là chúng ta thường nhập khẩu mô thức con pyplot thay vì toàn bộ mô-đun: nhập matplotlib.pyplot thay vì nhập matplotlib.

Khi chúng ta nhập matplotlib.pyplot, chúng ta cần sử dụng bí danh plt, theo quy ước (import matplotlib.pyplot as plt).

Mô thức con pyplot là một tập hợp các hàm cấp cao mà chúng ta có thể sử dụng để tạo đồ thị rất nhanh. Để tạo biểu đồ đường ở trên, chúng ta cần:

* Thêm dữ liệu vào  [hàm](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.plot.html#matplotlib-pyplot-plot) [plt.plot().](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.plot.html#matplotlib-pyplot-plot)
* Hiển thị biểu đồ bằng cách sử dụng hàm [plt.show().](https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.show.html)

Graphical user interface, text, application

Description automatically generated

Chart, line chart

Description automatically generated

Chúng ta thấy một dấu hiệu "1e6" khá kỳ quặc ở phần trên cùng bên trái của biểu đồ. Đây là ký hiệu khoa học và nó cho chúng ta biết rằng các giá trị trên trục y được nhân với 106. Điều này có nghĩa là bảy trên trục y có nghĩa là 7 nhân với 106, tức là bảy triệu - chúng ta sẽ quay lại điều này trên màn hình tiếp theo.

Hàm plt.plot() tạo ra một đồ thị đường theo mặc định. Tất cả những gì nó cần là hai **mảng dữ liệu có cùng độ** dài - đây có thể là danh sách Python, series pandas, mảng NumPy, v.v. Ở trên, chúng ta đã sử dụng hai danh sách Python.

Lưu ý thứ tự các đối số trong plt.plot(month\_number, new\_cases): month\_number đến trước, tiếp theo là new\_cases. Mảng đi trước cho tọa độ x và mảng thứ hai cho tọa độ y.

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Hai mảng phải có độ dài bằng nhau, nếu không một số tọa độ sẽ không được ghép đôi và Matplotlib sẽ gây ra lỗi.

Hãy tạo một biểu đồ đường mới trong bài tập dưới đây. Trên màn hình tiếp theo, chúng ta sẽ tìm hiểu thêm về cách tùy chỉnh biểu đồ: thêm tiêu đề, nhãn trục và xóa ký hiệu "1e6".

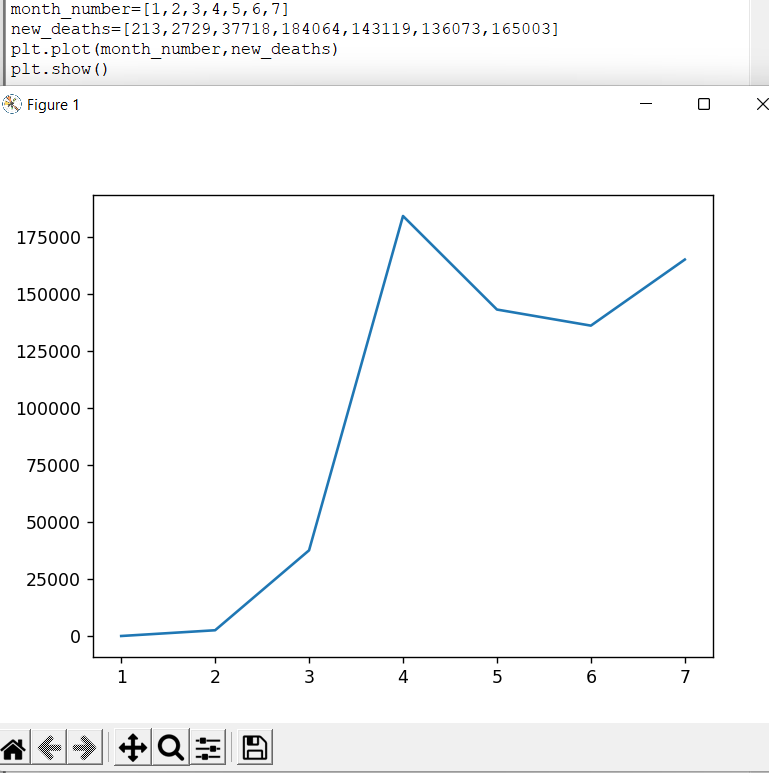
Giao diện người dùng đồ họa, văn bản, ứng dụng

Mô tả được tạo tự độngBiến month\_number chứa các tháng đã được tạo sẵn, tạo thêm biến new\_deaths chứa số người chết

1. Thêm module pyplot lưu dưới dạng plt

2. Tạo một biểu đồ đường sử dụng plt.plot(), với month\_number là tọa độ trục x, new\_deaths là tọa độ trục y

3. Biểu diễn biểu đồ sử dụng plt.show()



# Tùy chỉnh biểu đồ

Trên màn hình trước đó, chúng tôi đã xây dựng một biểu đồ đường cho thấy sự tiến triển của các trường hợp mới theo tháng:

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Ở phía trên cùng bên trái của biểu đồ, chúng ta thấy một dấu hiệu "1e6" - đây là ký hiệu khoa học. Matplotlib thay đổi thành ký hiệu khoa học nếu một giá trị trên trục cần phải là một triệu hoặc lớn hơn. Nếu chúng ta muốn loại bỏ ký hiệu khoa học, chúng ta có thể sử dụng [chức năng plt.ticklabel\_format (trục, kiểu](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.ticklabel_format.html#matplotlib-pyplot-ticklabel-format)).

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Tham số trục xác định trục nào cần cấu hình — các đối số của nó là các chuỗi 'x', 'y' và 'both'.

Tham số kiểu điều khiển kiểu ký hiệu (đơn giản hoặc khoa học). Lập luận của nó là 'khoa học viễn tưởng', 'khoa học' và 'đơn giản'.

Điều tiếp theo chúng ta sẽ làm là sử dụng  [hàm](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.title.html#matplotlib-pyplot-title) [plt.title()](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.title.html#matplotlib-pyplot-title) để thêm tiêu đề vào đồ thị đường của chúng ta.

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Trục x hiển thị số tháng và trục y hiển thị số trường hợp mới được báo cáo. Chúng tôi có thể hiển thị điều này trên biểu đồ của mình bằng cách thêm **nhãn** vào mỗi trục - nhãn y và nhãn x. Để thêm các nhãn trục, chúng ta sử dụng [plt.xlabel()](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.xlabel.html#matplotlib-pyplot-xlabel) và [plt.ylabel()](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.ylabel.html#matplotlib-pyplot-ylabel)

Biểu đồ

Mô tả được tạo tự động

Thêm nhãn tiêu đề và trục luôn là một điều tốt - ngay cả khi chúng ta chỉ đang khám phá dữ liệu cho chính mình và sẽ không ai khác nhìn thấy công việc của chúng ta.

Chúng tôi tạo ra nhiều biểu đồ khi khám phá dữ liệu và chúng tôi thường mất dấu những gì mỗi biểu đồ mô tả. Nếu chúng ta vẽ một biểu đồ ngay bây giờ và sau đó kiểm tra lại nó bốn mươi phút sau đó, tiêu đề và nhãn trục sẽ giúp chúng ta xác định ngay biểu đồ đó là gì.

Hãy tùy chỉnh một biểu đồ trong bài tập tiếp theo.

Giao diện người dùng đồ họa, văn bản, ứng dụng

Mô tả được tạo tự động

Giao diện người dùng đồ họa, văn bản, ứng dụng

Mô tả được tạo tự động

# Dữ liệu chuỗi thời gian của WHO

Trên màn hình trước, chúng tôi đã lưu trữ dữ liệu của mình trong một vài danh sách Python và sử dụng chúng để tạo biểu đồ đường. Tiếp theo, chúng ta sẽ sử dụng một tập dữ liệu lớn hơn mà chúng ta đã thu thập được từ [Tổ chức Y tế Thế giới](https://covid19.who.int/).

Hãy đọc trong tập dữ liệu bằng cách sử dụng thư viện gấu trúc:

Bàn

Mô tả được tạo tự động

Tập dữ liệu chứa dữ liệu từ ngày 4 tháng 31 đến ngày XNUMX tháng XNUMX. Mỗi hàng mô tả báo cáo COVID-19 trong một ngày ở một quốc gia cụ thể (một vài hàng đầu tiên chỉ hiển thị Trung Quốc vì vi rút chỉ có ở Trung Quốc vào thời điểm đó)

Nhắn tin

Mô tả được tạo tự động

Các hàng trong tập dữ liệu của chúng tôi được liệt kê theo thứ tự thời gian, bắt đầu từ ngày 4 tháng 1 và kết thúc bằng ngày 31 tháng 7. Chúng tôi gọi một loạt các điểm dữ liệu được liệt kê theo thứ tự thời gian là một **chuỗi thời gian.**

Thông thường , chúng tôi trực quan hóa chuỗi thời gian với biểu đồ đường. Các giá trị thời gian luôn được vẽ, theo quy ước, trên trục x.

Hãy đọc trong tập dữ liệu của chúng tôi trong bài tập sau.

Giao diện người dùng đồ họa, văn bản

Mô tả được tạo tự động

Nhắn tin

Mô tả được tạo tự động

# Các loại tăng trưởng

Trên màn hình trước, chúng tôi đọc trong tập dữ liệu của mình và lưu trữ nó trong một biến có tên who\_time\_series. Hãy nhanh chóng nhắc nhở bản thân về cấu trúc của tập dữ liệu:

Bảng, lịch

Mô tả được tạo tự động

Ý là tâm chấn thứ hai của đại dịch sau Trung Quốc. Hãy xem tổng số trường hợp tích lũy (nhớ lại điều này khác với số trường hợp mới) đã phát triển như thế nào trong bảy tháng đầu năm 2020. Trong đoạn mã bên dưới, chúng tôi bắt đầu bằng cách cô lập dữ liệu cho Ý, và sau đó chúng tôi tạo cốt truyện.

Nhắn tin

Mô tả được tạo tự động

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Cho đến tháng Ba, số ca tích lũy vẫn rất thấp. Nhưng sau đó, con số bắt đầu tăng rất nhanh (đường trên biểu đồ đi lên rất nhanh vào tháng Ba) và nó duy trì tốc độ nhanh đó cho đến tháng Năm. Sự tăng trưởng sau đó bắt đầu ổn định và trên biểu đồ, chúng ta thấy một đường gần như nằm ngang.

Nói chung, một số lượng tăng rất nhanh ngay từ đầu - và sau đó nó ngày càng chậm lại theo thời gian - có **sự tăng trưởng logarit.**

Trong giai đoạn từ tháng Ba đến tháng Bảy (do đó không bao gồm tháng Một và tháng Hai), Ý đã có sự tăng trưởng logarit về số ca tích lũy vì có nhiều trường hợp mới trong giai đoạn tháng Ba-tháng Tư, nhưng sau đó số ca mắc mới bắt đầu giảm. Đường trên biểu đồ sẽ trở nên nằm ngang hoàn hảo khi sẽ không còn trường hợp mới nào nữa.

Nếu chúng ta nhìn vào Ấn Độ, chúng ta có thể thấy một loại tăng trưởng khác:

Nhắn tin

Mô tả được tạo tự động

Ảnh có chứa sơ đồ

Mô tả được tạo tự động

Số lượng các trường hợp tích lũy tăng rất chậm trong giai đoạn tháng Hai-tháng Năm (dòng gần như nằm ngang). Nhưng sau đó sự tăng trưởng trở nên nhanh chóng (đường nhanh chóng chuyển hướng lên trên), và nó ngày càng nhanh hơn theo thời gian, mà không có bất kỳ dấu hiệu chậm lại nào.

Nói chung, một số lượng tăng chậm ngay từ đầu - nhưng sau đó bắt đầu tăng nhanh hơn và nhanh hơn theo thời gian - có **sự tăng trưởng theo cấp số nhân.**

Ấn Độ cho thấy sự tăng trưởng theo cấp số nhân đối với dữ liệu chúng tôi có, nhưng khi số ca mắc mới sẽ giảm, sự tăng trưởng (của các trường hợp tích lũy) sẽ trở thành logarit.

Nếu chúng ta nhìn vào Ý một lần nữa, chúng ta thực sự cũng có thể thấy sự tăng trưởng theo cấp số nhân nếu chúng ta chỉ cô lập giai đoạn tháng Hai-tháng Năm. Nhìn chung, Ý có tốc độ tăng trưởng chậm trong thời gian đầu, tiếp theo là sự tăng trưởng nhanh trong giai đoạn từ tháng XNUMX đến tháng XNUMX, và sau đó tăng trưởng lại chậm lại. Chuỗi tốc độ tăng trưởng này thường được mô tả là tăng trưởng hậu cần.

Bây giờ, hãy vẽ một biểu đồ đường cho Ba Lan để xem một loại tăng trưởng khác:

Nhắn tin

Mô tả được tạo tự động

Biểu đồ, biểu đồ đường

Mô tả được tạo tự động

Nếu chúng ta nhìn vào giai đoạn tháng Tư-tháng Bảy, chúng ta có thể thấy một đường thẳng xấp xỉ. Có một vài biến thể ở đây và ở đó, nhưng không có đường cong rõ ràng như chúng ta thấy đối với Ý hoặc Ấn Độ. Tuy nhiên, số lượng trường hợp vẫn tăng, nhưng nó tăng với tốc độ không đổi.

Nói chung, một số lượng tăng liên tục theo thời gian có **sự tăng trưởng tuyến tính.**

Tóm lại, đây là ba loại tăng trưởng mà chúng ta đã học được trong màn hình này:

Biểu đồ

Mô tả được tạo tự động

Chúng tôi sẽ tiếp tục thảo luận về các loại tăng trưởng trên màn hình tiếp theo. Bây giờ chúng ta hãy xem xét một bài tập.

Ảnh có chứa văn bản

Mô tả được tạo tự động

Giao diện người dùng đồ họa, văn bản

Mô tả được tạo tự động

# Các loại thay đổi

Trên màn hình trước, chúng ta đã học được ba loại tăng trưởng phổ biến: tuyến tính, hàm mũ và logarit. Như một lời cảnh báo, việc dán nhãn một loại tăng trưởng chỉ bằng cách nhìn vào biểu đồ còn lâu mới chính xác. Những loại tăng trưởng này được mô tả tốt nhất bằng các hàm toán học chính xác và được xác định rõ ràng. Tuy nhiên, những xấp xỉ trực quan này có thể đóng vai trò là công cụ tâm trí hữu ích mà chúng ta có thể sử dụng để giải thích cách dữ liệu chuỗi thời gian thay đổi.

Biểu đồ

Mô tả được tạo tự động với độ tin cậy trung bình

Thay đổi không chỉ là về tăng trưởng. Một đại lượng cũng có thể giảm theo một mô hình tuyến tính, hàm mũ hoặc logarit.

Ảnh có chứa biểu đồ

Mô tả được tạo tự động

Tuy nhiên, dữ liệu hiếm khi phù hợp hoàn hảo với bất kỳ mẫu nào trong số này. Thông thường, đồ thị đường của chúng tôi chỉ *xấp xỉ* tuyến tính, xấp xỉ hàm mũ hoặc xấp xỉ logarit. Hơn nữa, một phần của biểu đồ đường đơn lẻ có thể hiển thị thay đổi theo cấp số nhân, một phần khác của cùng một biểu đồ có thể hiển thị thay đổi tuyến tính, trong khi một phần khác có thể hiển thị thay đổi bất thường không giống với bất kỳ mẫu thông thường nào.

Trong thực tế, hầu hết các biểu đồ đường mà chúng tôi vẽ không hiển thị bất kỳ mẫu rõ ràng nào. Chúng ta cần chú ý đến những gì chúng ta thấy và cố gắng trích xuất ý nghĩa mà không buộc dữ liệu thành một số mẫu mà chúng ta đã biết.

Ví dụ, nếu chúng ta nhìn vào sự tiến triển của các trường hợp mới ở Belarus, chúng ta thấy nhiều bất thường trên biểu đồ đường:

Giao diện người dùng đồ họa, văn bản, ứng dụng

Mô tả được tạo tự động

Biểu đồ, biểu đồ tần suất

Mô tả được tạo tự động

Trong giai đoạn từ tháng XNUMX đến tháng XNUMX, chúng ta thấy một số mức tăng đột biến trên biểu đồ đi lên hoặc xuống. Trong một số ngày, số ca mắc mới lên tới gần 2,000 (mức tăng đột biến), trong khi đối với những người khác là XNUMX (mức tăng đột biến xuống). Những biến thể lớn này cho thấy rằng các báo cáo đã không đến hàng ngày - có thể là không ai gửi báo cáo vào cuối tuần hoặc vào các ngày lễ quốc gia. Số ca mắc mới tiếp tục tăng cho đến báo cáo tiếp theo, và sau đó chúng ta thấy một trong những mức tăng đột biến đó

Khi chúng ta thấy những bất thường trên biểu đồ đường, điều này không có nghĩa là chúng ta không thể trích xuất bất kỳ ý nghĩa nào. Bằng cách phân tích những bất thường, đôi khi chúng ta có thể khám phá ra những chi tiết thú vị.

# So sánh biểu đồ đường

Cho đến nay, chúng tôi đã tìm hiểu biểu đồ đường là gì, cách xây dựng biểu đồ đường bằng Matplotlib và một số loại thay đổi phổ biến. Tiếp theo, chúng ta sẽ tập trung vào việc so sánh biểu đồ đường.

Một trong những yếu tố chính của khám phá dữ liệu là so sánh - giá trị này so với giá trị khác như thế nào? Đối với chuỗi thời gian COVID-19 của chúng tôi, chúng tôi có thể hình thành nhiều câu hỏi về mặt so sánh:

* Vương quốc Anh so với Pháp như thế nào liên quan đến sự tiến triển của các trường hợp mới tích lũy?
* Làm thế nào để Mexico so sánh với Hoa Kỳ liên quan đến số ca tử vong đáng kinh ngạc?
* Diễn biến của các trường hợp mới được báo cáo so sánh giữa Ấn Độ, Indonesia và Trung Quốc như thế nào?
* Diễn biến của tổng số ca nhiễm so với châu Âu và châu Á như thế nào? Hay giữa Châu Phi và Nam Mỹ?

Ví dụ, hãy hình dung sự tiến triển của các trường hợp tích lũy cho Pháp và Vương quốc Anh. Matplotlib cho phép chúng ta có hai biểu đồ đường chia sẻ cùng một trục x- và y:

Biểu đồ

Mô tả được tạo tự động với độ tin cậy thấp

Chúng tôi thấy hai dòng màu sắc khác nhau ở trên, nhưng chúng tôi không thể biết cái nào dành cho Pháp và cái nào dành cho Vương quốc Anh. Để giải quyết vấn đề này, chúng tôi sẽ thêm một **chú giải** cho thấy màu nào tương ứng với quốc gia nào. Trong đoạn code dưới đây, trước tiên chúng ta thêm một đối số label vào hàm plt.plot(), sau đó chúng ta sử dụng  [hàm](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.legend.html#matplotlib-pyplot-legend) [plt.legend():](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.legend.html#matplotlib-pyplot-legend)

Biểu đồ

Mô tả được tạo tự động

Khi chúng ta sử dụng plt.plot() lần đầu tiên, Matplotlib sẽ tạo ra một đồ thị đường. Khi chúng ta sử dụng plt.plot() một lần nữa, Matplotlib tạo ra một đồ thị đường khác có cùng trục x và y như đồ thị đầu tiên. Nếu chúng ta muốn Matplotlib vẽ riêng đồ thị đường thứ hai, chúng ta cần đóng đồ thị đầu tiên bằng  [hàm](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.show.html) [plt.show()](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.show.html)

Giao diện người dùng đồ họa

Mô tả được tạo tự động

Biểu đồ

Mô tả được tạo tự động

Nhìn vào hai biểu đồ trên, sự phát triển của các trường hợp tích lũy trông rất giống nhau nếu chúng ta chỉ đánh giá bằng hình dạng của đường thẳng. Tuy nhiên, nếu chúng ta nhìn vào trục y, chúng ta thấy rằng hai biểu đồ có phạm vi khác nhau và các giá trị cho Vương quốc Anh lớn gần gấp đôi. Sẽ dễ dàng hơn nhiều để so sánh hai hình ảnh trực quan này nếu chúng chia sẻ cùng một trục.

Bây giờ chúng ta hãy làm một bài tập và kết thúc bài học này trên màn hình tiếp theo.

Ảnh có chứa văn bản

Mô tả được tạo tự động

Nhắn tin

Mô tả được tạo tự động

Tôin bài học này, chúng tôi đã trải qua một giới thiệu nhanh về đồ thị, và sau đó chúng tôi đã học được cách làm như sau:

* Vẽ và tùy chỉnh biểu đồ đường bằng Matplotlib
* Trực quan hóa chuỗi thời gian bằng biểu đồ đường
* Diễn giải biểu đồ đường kẻ bằng cách xác định các loại thay đổi

Trong bài học tiếp theo, chúng ta sẽ tìm hiểu về tính thời vụ, mối tương quan và các ô phân tán