Xin chào mọi người

Cảm ơn mọi người đã tham gia buổi livestream hôm nay của chúng mình , để thuận tiện hơn cho việc thực hành các nội dung hôm nay thì mở đầu mình sẽ bày các bạn cách cài các thư viện cần thiết cho python nhé :v

Đầu tiên là thư viện Numpy, mình sẽ nói sơ qua công dụng của numpy 1 chút nhé , numpy được sử dụng để hỗ trợ việc tính toán các mảng nhiều chiều , có kích thước lớn với các hàm tối ưu sẵn, numpy đặc biệt hữu ích khi thực hiện các hàm liên quan đại số tuyến tính :D , bạn nào đã cài python với pip theo video bọn mình up lên thì việc cài đặt numpy rất là đơn giản nhé, các bạn vào setting ->cmd-> và gõ dòng lệnh (đoạn ni anh làm trên màn hình luôn cho dễ nhìn a nhé ;) )

Pip install numpy, chờ cho nó chạy xong là các bạn có thể sử dụng rồi nhé, để biết được thư viện đã được cài đặt chưa rất dễ dàng nhé :V , các bạn chạy lệnh python trong cmd để chạy python và sau đó gõ import numpy nhé :v ,nếu không có lỗi gì thì các bạn đã cài thành công rồi đấy ;)

Tiếp theo các bạn tiếp tục cài đặt thư viện matplotlib nhé :V , các bạn khoa mình chắc không còn xa lạ gì với matlab rồi gì nữa :V matplotlib tương tự như matlab dành cho python vậy :v , câu lệnh cài đặt cũng gần giống như numpy nhé ,

Pip install matplotlib

Và cuối cùng là sẽ là thư viện chính cho buổi học ngày hôm nay của chúng ta , để nói sơ về pandas thì pandas là 1 thư viện mã nguồn mở (tức là free đấy ☺)) ) , đây là một bộ công cụ phân tích và xử lý dữ liệu mạnh mẽ của python, sự đặc biệt riêng của pandas đấy là cấu trúc dữ liệu dataframe :v, chút nữa chúng ta sẽ tìm hiểu kỹ hơn nhé :D .câu lệnh cài đặt cùng cực kỳ đơn giản

Pip install pandas

Vậy là bước đầu đã xong :D , bây giờ chúng ta vào bài học thôi nào, các bạn có thể dùng bất cứ IDE nào có hỗ trợ python nhé , mình sẽ recommend cho các bạn một nền tảng giúp các bạn học pandas một cách trực quan hơn nhé , đó là jupyter notebook, vậy jupyter khác gì những ide còn lại nhỉ ? jupyter cho phép các bạn thực thi từng câu lệnh 1 , có nghĩa là bạn có thể show nhiều kết quả của code cùng 1 lúc :v rất tiện cho việc đánh giá và so sánh đúng không nào .

Mọi thứ đã sẵn sàng chúng ta bắt đầu vào bài học thôi nào !!!!

Đầu tiên sẽ là khởi tạo 1 dataframe nhé , tính chất của dataframe là 1 dictionary nhé các bạn, ở đây các keys đóng vai trò là các columns, values sẽ là giá trị của các columns đấy, ở hàng đầu tiên chúng ta nhập vào ID = 1 thì tất cả các giá trị của cột ID sẽ là 1, ở hàng tiếp theo chúng t có 1 hàm là datarange, hàm này cho phép chúng ta tạo ra một chuỗi thời gian (time series) ,ở đây các bạn có thể thấy ngày bắt đầu là ngày 2-2-2020 và periods=4 có nghĩa là chu kỳ bằng 4,đại khái nó là số ngày chúng ta muốn timeseries thể hiện đấy,ngoài ra hàm datarange còn hỗ trợ một biến là freq có nghĩa là bạn có thể chọn chu kì theo ngày, tháng , hoặc năm nữa nhé , nếu không khai báo gì thì nó sẽ là chu kỳ theo ngày :D ,ở cuối df chúng ta sử dụng hàm numpy để tạo 1 mảng từ 1-5 , rất đơn giản phải không nào, sau khi chạy đoạn code trên kết quả của chúng ta sẽ là :

Rồi, Đa số các bài toàn ML của chúng ta sẽ sử dụng dữ liệu được tạo sẵn, mình sẽ giới thiệu cho các bạn 1 trang web để tải các dataset và cả các ví dụ sử dụng dữ liệu đó nữa đó là trang kaggle.com , tiếp theo chúng ta sẽ thử tải 1 dataset từ kaggle và sử dụng pandas lên nó nhé :D

Mình sẽ để đường link cho các bạn tải dataset về nhé , thì cơ bản pandas hỗ trợ rất nhiều hàm để đọc nhiều loại file khác nhau ví dụ như , csv, excel, table,pickle, json, sql ….. thì hôm nay chúng ta sẽ thử đọc 1 loại file khá phổ biến trong việc lưu trữ các dataset đó là file csv.chúng ta sẽ sử dụng hàm read\_csv nhé mọi người, thì hàm này có khá nhiều tham số các bạn có thể lên trang chủ để tham khảo nhá , để đọc 1 file csv cơ bản thì chúng ta chỉ cần truyền đường dẫn của file csv vào nhá , một lưu ý nho nhỏ là định dạng cơ bản cho dữ liệu trong dataframe là utf-8 nhé nếu bạn đọc một file csv với định dạng khác thì nó sẽ báo lỗi, sau đó để hiện thị dataframe theo thứ tự từ trên xuống thì chúng ta sẽ dùng hàm head tham số truyền vào sẽ là số hàng chúng ta muốn hiển thị, còn muốn ngược lại thì chúng ta sẽ dùng hàm tail nhé mn .

Tiếp theo chúng ta sẽ đến với bước đầu của xử lý dữ liệu đó là lọc giá trị (Selection)

[In5] : đây là cú pháp đơn giản nhất để chúng ta có thể lấy ra được 1 cột dữ liệu của dataframe, nhưng nó vẫn ở dạng pandas series nên sẽ không phù hợp cho các công việc tính toán khác , để chuyển nó thành dạng numpy array chúng ta thực hiện câu lệnh [In6]

[In7] + [In8] : sẽ thực hiện chức năng vẫn là lấy dữ liệu của một cột nhưng theo dạng dataframe, ngoài ra chúng ta còn có 3 hàm để lọc dữ liệu là loc, iloc, và hàm filter vậy chúng khác nhau như thế nào

Loc(): dùng để lấy ra dữ liệu của hàng và cột bằng các **nhãn** hay là **mảng boolean**

Filter() : có chức năng tương tự như Loc() , nhưng có hỗ trợ thêm một vài thao tác lọc nhãn chi tiết hơn hàm Loc() nhưng lại không hỗ trợ **mảng boolean**.

Câu lệnh [In9] và [In8] thực hiện nhiệm vụ như câu lệnh 7 nhưng với hàm loc, và hàm filter(), đơn giản là nó sẽ lọc ra tất cả các hàng của cột star, với hàm filter chúng ta không cần khai báo giá trị của hàng.

Iloc(): dùng để lấy ra dữ liệu dựa trên **vị trí số nguyên** của vị trí đó.

Câu lệnh [In10] cũng thực hiện nhiệm vụ như 3 câu lệnh trên nhưng với hàm iloc() và các bạn đã thấy sự khác nhau giữa 2 hàm chưa nào.

Câu lệnh [In11] sẽ bày chúng ta cách để lọc ra 1 dataframe dựa theo điều kiện của 1 cột , như câu lệnh in11 nó sẽ lọc ra dataframe mà tại cột “State” giá trị sẽ là “Open”

[In12] : câu lệnh này cũng thực hiện chức năng tương tự câu lệnh 11 nhưng lại dùng 1 hàm khác là hàm isin(), ở đây nó lọc ra dataframe mà tại cột “income” có giá tri bằng “20M”

Vậy thì khi chúng ta tạo 1 dataframe ngẫu nhiên thì “Index” của dataframe sẽ là một cột với số thứ tự tăng dần, để thay đổi Index của df thì chúng ta sử dụng câu lệnh set\_index() như ở [In13] , tham số inplace như để xác nhận việc thay đổi ngay lập tức vậy ^^

Tiếp theo chúng ta sẽ đến với các hàm để hỗ trợ tính toán giá trị trong các cột của dataframe nhé

Để tính toán các giá trị trong các cột của df thì chúng ta cần có các giá trị để sử dụng tính toán phải không nào, một vài trường hợp chúng ta cần sử dụng đến các giá trị trong quá khứ để tính toán cho giá trị hiện tại vậy nên pandas đã hỗ trợ cho chúng ta 1 hàm để giải quyết vấn đề này đó là rolling(), tham số n của hàm này là 1 interger cho phép chúng ta duyệt qua n giá trị trong quá khứ để thực hiện việc tính toán , như trong [In14] chúng ta sẽ tạo ra 1 cột “Future” mới mà giá trị tại mỗi hàng của cột này sẽ là tổng của 2 giá trị trước đó của cột “ AveragePrice” , giá trị đầu tiên của cột là giá trị NAN vì trước hàng đầu tiên làm gì có hàng nào mà cộng ☺))))))), ngoài sum() ra chúng ta còn có thể dùng hàm mean() để lấy giá trị trung bình nữa nhá

Vậy thì làm chúng ta giải quyết như thế nào với giá trị Nan ở đầu hàng của cột Future, chúng ta sẽ có 2 lựa chọn, 1 là sẽ loại bỏ luôn hàng mà có giá trị NAN ở bất kì cột nào bằng hàm dropna() , 2 là chúng ta sẽ thay thế giá trị NAN ấy bằng giá trị mà chúng ta mong muốn bằng hàm fillna().

[In15] : Đầu tiên chúng ta sẽ copy ra 1 bản của df data để khỏi ảnh hưởng đến df chính nhá , sau đó chúng ta sẽ dùng hàm dropna để loại bỏ tất cả hàng có giá trị NA ở bất kì cột nào trong df… và kết quả như các bạn đã thấy …

[In16] : sẽ diễn tả lựa chọn thứ 2 mà mình đã nói ở trên nhé , bắt đầu chúng ta cũng tạo ra 1 bản copy như [In15] nhé , tiếp theo chúng ta sẽ sử dụng hàm fillna() để thay đổi giá trị NAN thành giá trị trung bình của cột “Average Price” và chúng ta đặt một biến mới đó là data4 và kết quả sau khi thực hiện là …..

Chúng ta vừa tìm hiểu xong cách để loại bỏ một giá trị NA , còn nếu trong trường hợp cần loại bỏ 1 cột hoặc 1 hàng nào đó thì như thế nào ?

Câu lệnh [In18] và [In19] sẽ hướng dẫn các bạn làm nhé, hàm drop() hỗ trợ chúng ta bỏ hàng và cột bằng cả nhãn và chỉ mục của hàng hoặc cột đó, ví dụ ở câu lệnh [In18] sẽ bày các bạn loại bỏ hàng với tham số là 1 nhãn , và câu lệnh [In19] là thực hiện với tham số là 1 chỉ mục,để thực hiện điều đó trước tiên chúng ta phải reset lại index của dataframe data đã nhé, và đương nhiên tham số inplace = True rồi, tiếp theo sẽ là loại bỏ cột nhé để chuyển đổi từ hàng sang cột chúng ta chỉ cần thay đổi tham số axis = 1 nhé ,còn không thì mặc định của nó sẽ là 0 là hàng đấy.Xonggggg

Hmmm vậy là chúng ta đã đi qua được kha khá cách xử lý dự liệu mà pandas giúp chúng ta rồi phải không nào, trong một số trường hợp thì dữ liệu của chúng ta là một chuỗi dữ liệu rất nhiều giá trị thì làm thế nào để chúng ta biết được có bao nhiều loại giá trị trong dataframe và năm được một số thông số cơ bản của nó , thì đương nhiên rồi ☺)))))))))) pandas cũng sẽ hỗ trợ chúng ta cả vấn đề này với 2 hàm cực kì đơn giản là unique() và describe() ở [In21] và [In22] ,hàm unique sẽ giúp chúng ta có bao nhiêu loại giữ liệu trong cột đó, còn hàm describe sẽ giúp chúng ta biết được một số thông số cơ bản của dữ liệu trong cột như min,max,sdt(độ lệch chuẩn),giá trị trung bình(mean)…… và đương nhiên dữ liệu của cột phải là numeric nhá (dạng số) nhá.

Sau đây mình xin làm thử một bài tập đơn giản để vận dụng các kiến thức mà chúng ta đã học từ đầu buổi livestream đến bây giờ nhé ;)

[In24] : chúng ta sẽ tạo ra 1 dataframe df4 là bản copy của df3 , và sau đó lấy ra dataframe mà trong đó cột “type” có giá trị là “organic” , tiếp theo chúng ta sẽ sắp xếp lại cột “Date” bằng hàm sort\_values() nhé, ở đây chúng ta có 1 tham số đặc biệt đó là “ascending” nếu giá trị của nó là True thì cách thức sắp xếp sẽ là tăng dần còn nếu là False thì sẽ là ngược lại nhé ;). Mục đích chúng ta làm như vậy để giảm lọc ra dataframe của sản phẩm organic để thực hiện công đoạn tiếp theo nhé.

[In25] : Đoạn code dưới đây sẽ thực hiện chức năng tính giá trung bình của sản phẩm organic của các khu vực khác nhau nhé và hiển thị trong một dataframe mới trực quang hơn , chúng ta sẽ làm quen một hàm mới trong pandas là hàm join() mục đích thì chắc các bạn đọc tên hàm thì cũng đoán ra rồi nhỉ , đúng vậy nó dùng để ghép các dataframe lại với nhau ^^.

Đoạn code [In25] có vẻ làm các bạn khá là đau đầu phải không nào, mình sẽ up file code lên page câu lạc bộ các bạn có thể về xem lại và có thể inbox cho bọn mình để đặt câu hỏi nhé, vậy là chúng ta cũng đã đi gần hết thời lượng của buổi livestream rồi nhỉ , phần cuối cùng chúng mình sẽ hướng dẫn cho bạn một note nhỏ để xử lý môt lỗi nhỏ trong khi sử dụng pandas mà các bạn có thể gặp phải đó là lỗi decode . Như mình đã nói ở phần đầu thì mặc định định dạng encode của dataframe sẽ là utf-8 , nhưng mà sẽ có 1 số trường hợp file dataset chúng ta tải về thì định dạng encode của chúng sẽ khác, vậy nên mình sẽ hướng dẫn các bạn cách xử lý nhé .