**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC KINH TẾ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN KINH DOANH**



**ĐỒ ÁN MÔN HỌC**  
LẬP TRÌNH PHÂN TÍCH DỮ LIỆU

***Đề tài:***

***Phân tích dữ liệu kinh doanh của cửa hàng bán lẻ trực tuyến****.*

**Thành viên**:Nguyễn Phúc Hải

Lê Đức Dũng

Nguyễn Văn Hoàng Dũng

Võ Ngọc Dung

Nguyễn Quỳnh Khánh Hà

**Giảng viên**: TS. Nguyễn An Tế

*Thành phố Hồ Chí Minh , ngày 3 tháng 10 năm 2022*

Mục Lục

[LỜI CẢM ƠN 3](#_Toc121006539)

[Chương I: Tổng Quan Đề Tài 1](#_Toc121006540)

[1.     Giới thiệu đề tài 1](#_Toc121006541)

[2.     Mục tiêu nghiên cứu 1](#_Toc121006542)

[3.     Phương pháp nghiên cứu: 1](#_Toc121006543)

[4.     Tài nguyên sử dụng: 2](#_Toc121006544)

[Chương II: Tổng Quan Bộ Dữ Liệu 2](#_Toc121006545)

[1.     Tổng quan bộ dữ liệu thu thập 2](#_Toc121006546)

[2.     Các thuộc tính của bộ dữ liệu 2](#_Toc121006547)

[Chương III: Tiền xử lý dữ liệu 3](#_Toc121006548)

[1.     Exploratory Data Analysis (EDA) 3](#_Toc121006549)

[a.     Tổng quan bộ dữ liệu nguyên bản 3](#_Toc121006550)

[b. Biểu diễn dữ liệu nguyên bản: 6](#_Toc121006551)

[2. Chỉnh dạng dữ liệu 9](#_Toc121006552)

[a.     Thêm cột 9](#_Toc121006553)

[b.     Xoá cột: 10](#_Toc121006554)

[3.     Xử lý dữ liệu bị thiếu (Missing Values) 10](#_Toc121006555)

[4.  Xử lý outliers của các biến định lượng 12](#_Toc121006556)

[a.     Biến Totalsale 12](#_Toc121006557)

[b.   Biến Quantity 15](#_Toc121006558)

[5.   Xử lý biến Description 18](#_Toc121006559)

[Chương IV: Phân Tích Dữ Liệu 20](#_Toc121006560)

[1. Time Series: 20](#_Toc121006561)

[2. FP-Growth: 24](#_Toc121006562)

[Chương V: Machine Learning 35](#_Toc121006563)

[1. Cluster Category by Product Name (Description) - K-means: 35](#_Toc121006564)

[2. Customer Segmentation by Country and Behavior - K-Prototypes 43](#_Toc121006565)

[CHƯƠNG VI: KẾT LUẬN 48](#_Toc121006566)

[Tài Liệu Tham Khảo 49](#_Toc121006567)

[Bảng phân công: 49](#_Toc121006568)

**LỜI CẢM ƠN**

Hiện nay, trong thời đại 4.0, mặc dù đã có sự xuất hiện của trí tuệ nhân tạo giúp hỗ trợ cho công việc phân tích kết quả kinh doanh nhằm dự báo, đưa ra các quyết định kinh doanh giúp tối ưu hóa doanh thu cho doanh nghiệp nhưng sự cần thiết của những người làm công việc phân tích dữ liệu vẫn là rất cần thiết đối với các doanh nghiệp. Chính vì vậy, nhóm đã quyết định chọn bộ dữ liệu “Online Retails Sale Dataset” làm bộ dữ liệu để làm báo cáo cho môn học “Lập trình phân tích dữ liệu”. Mục đích của việc phân tích bộ dữ liệu này là để tìm ra các mặt hàng được bán chạy theo combo, thời gian; tìm ra các khách hàng tiềm năng,… để từ đó có thể khai thác thêm các dữ liệu tiềm năng cho việc đưa ra các quyết định kinh doanh.

Trong quá trình làm đồ án môn học vẫn còn các hạn chế, sai sót, chưa tối ưu hóa về mặt kiến thức, kĩ thuật. Nhóm chúng em mong sẽ nhận được sự phản hồi, nhận xét của thầy cô giảng viên hướng dẫn để cải thiện các điểm này.

Đặc biệt nhóm xin được gởi lời cảm ơn đến thầy Nguyễn An Tế, giảng viên hướng dẫn của học phần “Lập trình phân tích dữ liệu” này. Thầy đã giúp đỡ, hướng dẫn, cung cấp các tài liệu, kiến thức, kĩ năng cần thiết để nhóm em có thể hoàn thành báo cáo Đồ án kết thúc môn học này.

Chúng em xin chân thành cảm ơn thầy.

Thay mặt nhóm sinh viên thực hiện đồ án,

Hải,

Nguyễn Phúc Hải

# Chương I: Tổng Quan Đề Tài

## 1.     Giới thiệu đề tài

Ngày nay, mạng Internet là một trong những công cụ cần thiết trong cuộc sống hiện đại, là nền tảng cho sự truyền tải và trao đổi thông tin trên toàn cầu. Cùng với sự phát triển nhảy vọt của công nghệ thông tin, Internet đang dần chiếm giữ vai trò quan trọng trong mọi mặt của đời sống, giúp con người làm việc với độ chính xác cao, quản lý và tổ chức công việc hiệu quả, cũng như nhanh chóng cập nhật thông tin một cách chính xác.

Vì vậy, thương mại điện tử ngày càng phát triển và đem lại bước đột phá mới cho công tác quản lý bán hàng. Nó giúp doanh nghiệp dễ dàng nắm bắt thông tin, dữ liệu về người tiêu dùng, hàng hóa và các đơn đặt hàng một cách nhanh chóng. Cùng với đó là sự phát triển của việc phân tích dữ liệu khách hàng, giúp cho quản lý doanh nghiệp có cái nhìn tổng quát về tình hình bán hàng của doanh nghiệp, khai thác được những thông tin hữu ích từ bộ dữ liệu mả họ đang có để từ đó đưa ra những chính sách bán hàng hiệu quả.

## 2.     Mục tiêu nghiên cứu

-       Tìm ra những kết hợp sản phẩm có khả năng thường xuyên được bán cùng nhau nhất để từ đó biết được mối liên hệ giữa các sản phẩm và có thể đưa ra các chiến lược thúc đẩy khả năng bán chéo giữa các sản phẩm đó.

-       Phân tích chuỗi thời gian bán hàng tiềm năng để phát hiện ra được tính xu hướng và chu kỳ của tập dữ liệu khách hàng, từ đó đưa ra các chiến dịch quảng bá sản phẩm phù hợp theo từng mùa.

- Phân khúc khách hàng theo vị trí địa lý và hành vi mua hàng, xem xét những gì khách hàng mua, mức giá, tần suất và số lượng họ mua hàng để từ đó có thể đưa ra các chiến dịch tiếp thị phù hợp với từng nhóm khách hàng.

## 3.     Phương pháp nghiên cứu:

             EDA : Sử dụng các biểu đồ vẽ nhằm tương quan cũng như làm rõ mục đích nghiên cứu đề tài, sự liên kết với nhau giữa các biến.

FP-Growth: Sử dụng FP-Growth để tìm ra các mặt hàng thường được bán chung với nhau (trong cùng 1 hóa đơn) và khoảng thời gian mà các mặt hàng được bán chạy nhất giúp tối ưu hóa lượng hàng được bán ra. Vì giới hạn về mặt phần cứng (nếu áp dụng FP-Growth trên dataset Online Retail đã qua tiền xử lý sẽ phát sinh một số lượng tập kết hợp quá lớn (lên đến gần 33000 tập kết hợp) khiến việc tính toán dựa vào các thiết bị thông thường là gần như không thể. Vì thế ở đây, ta bắt đầu với việc lấy mẫu ngẫu nhiên 350000 dòng dữ liệu (sau khi thử nghiệm thì 350000 dòng dữ liệu là số dòng dữ liệu được lấy từ bộ dữ liệu gốc có thể thực hiện thuật toán FP-Growth mà không bỏ quá nhiều dữ liệu, chỉ bỏ khoảng 5% dữ liệu)  từ bộ dữ liệu gốc sau tiền xử lý.

## 4.     Tài nguyên sử dụng:

-       Ngôn ngữ lập trình: Python.

-       Bộ dữ liệu “Online Retails Sale Data” được lấy từ Kaggle.

# Chương II: Tổng Quan Bộ Dữ Liệu

## 1.     Tổng quan bộ dữ liệu thu thập

-       Bộ dữ liệu “Online Retails Sale Data”  chứa các giao dịch từ ngày 12/01/2010 đến 12/09/2011  của một công ty bán lẻ trực tuyến có trụ sở tại UK.

Bao gồm 10 thuộc tính, số dòng của bộ dữ liệu là 541909 dòng

## 2.     Các thuộc tính của bộ dữ liệu

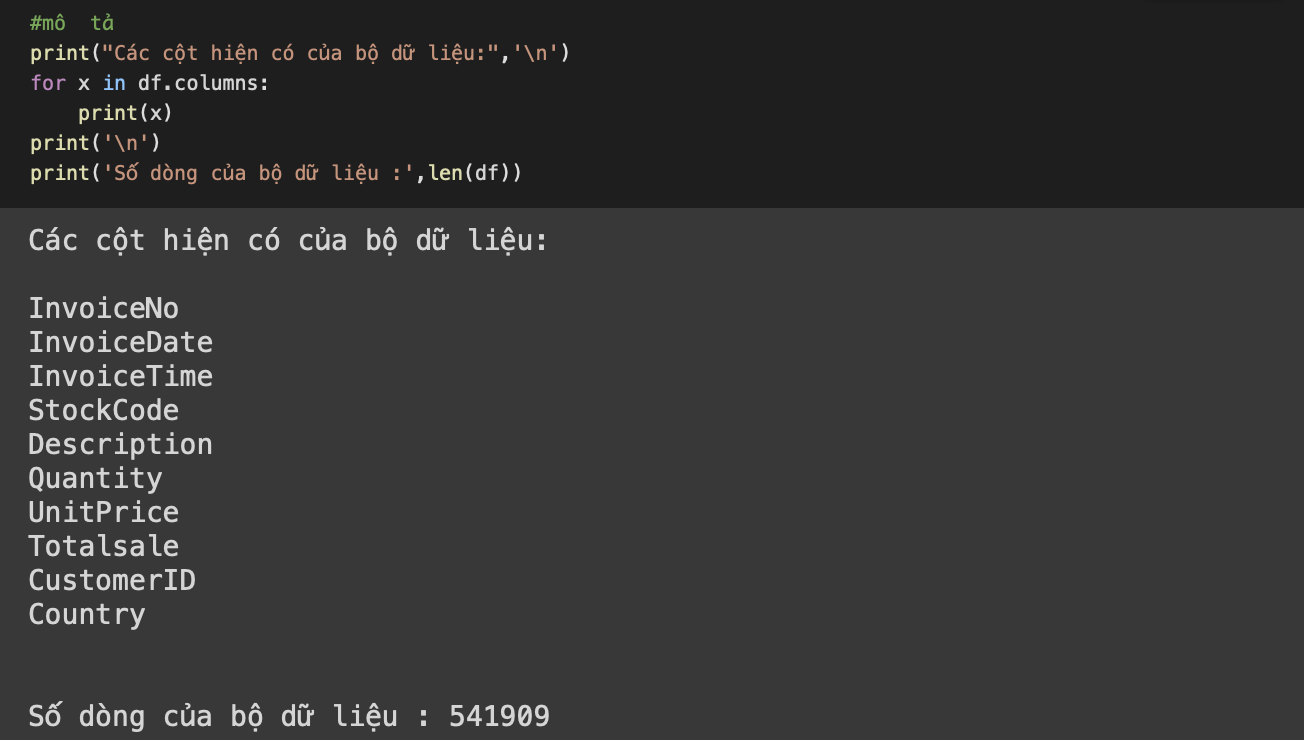
|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Mô tả** | **Ghi chú** |
| InvoiceNo | Mã đơn hàng cho mỗi order thành công | Bao gồm 25900 mã đơn hàng khác nhau |
| InvoiceDate | Ngày đặt hàng | Từ ngày 12 tháng 1 năm 2010 đến ngày 10 tháng 12 năm 2011 |
| InvoiceTime | Thời gian đặt hàng | Phụ thuộc vào vị trí địa lí của từng khách hàng |
| StockCode | Mã sản phẩm | Bao gồm 4070 mã sản phẩm khác nhau |
| Description | Mô tả sản phẩm | Bao gồm 4224 mô tả, trong đó có các sản phẩm bị lỗi, hư hỏng, bị trả về, thất lạc |
| Quantity | Số lượng mỗi sản phẩm | Số lượng của mỗi đơn vị sản phẩm được bán ra trong 1 đơn hàng. |
| UnitPrice | Giá bán mỗi đơn vị sản phẩm | Giá của 1 sản phẩm, mỗi sản phẩm sẽ có mức giá khác nhau. |
| Totalsale | Doanh thu từ mỗi sản phẩm trong 1 đơn hàng. | Totalsale = Quantity \* UnitPrice |
| CustomerID | Mã khách hàng | Mỗi khách hàng sẽ được cấp 1 mã duy nhất. |
| Country | Quốc gia của khách đặt hàng | Căn cứ vào vị trí địa lý của khách hàng. |

# Chương III: Tiền xử lý dữ liệu

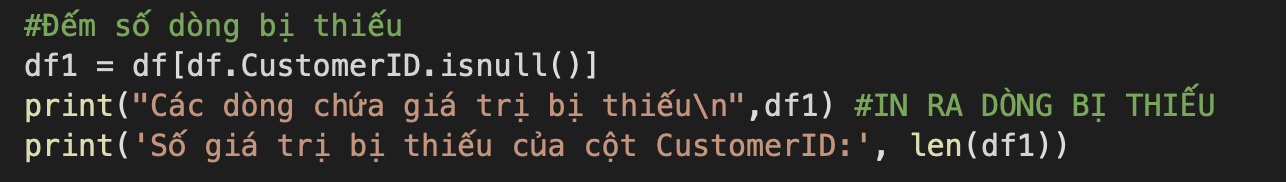
## 1.     Exploratory Data Analysis (EDA)

### a.     Tổng quan bộ dữ liệu nguyên bản

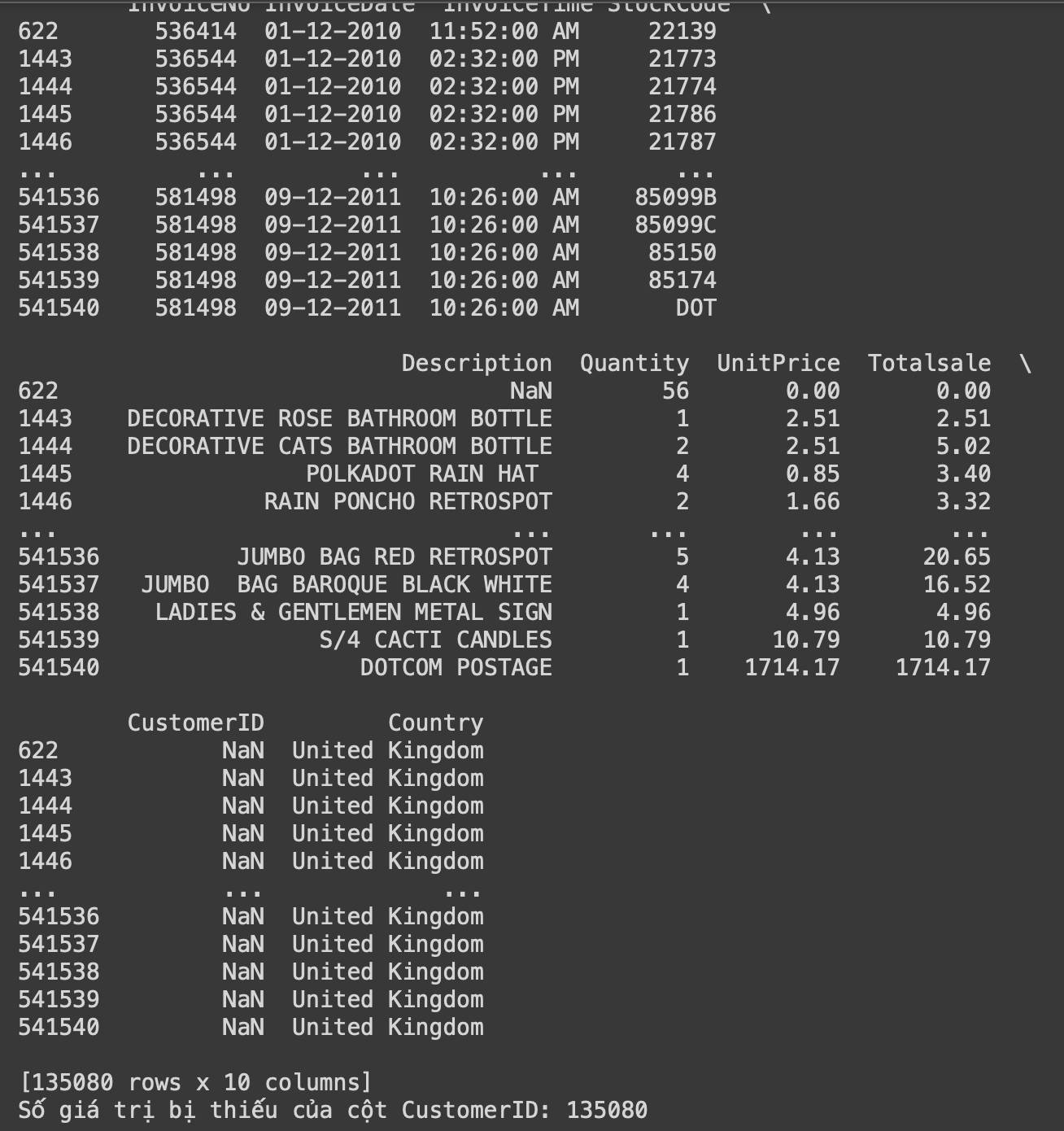
* Để thăm dò bộ dữ liệu, ta cần biết được tổng quan các thông tin về: số dòng, số cột, có tồn tại giá trị bị thiếu hay không, nếu có thì ở dòng nào, thuộc cột nào và chiếm bao nhiêu phần trăm của bộ dữ liệu.
* Xem số dòng, số cột hiện có của bộ dữ liệu nguyên bản để nắm được các thông tin sơ lược trước khi tiến hành tiền xử lý:



-       Tiến hành kiểm tra các dòng chứa giá trị bị thiếu:

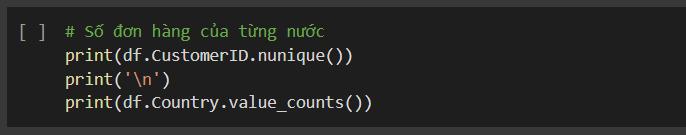


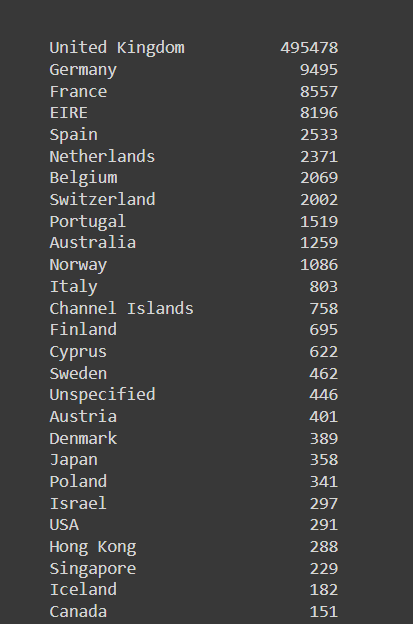
* Kết quả trả về:



*Nhận xét: Bộ dữ liệu trên có 10 cột, số dòng dữ liệu là**541909, tồn tại cột CustomerID với số giá trị bị thiếu là 135080, tức xấp xỉ 25% bộ dữ liệu tồn tại giá trị bị thiếu ở cột này. Vì vậy ta sẽ xử lý các giá trị thiếu này ở bước sau.*

* Để khám phá dữ liệu, chúng ta sẽ đưa ra số đơn hàng của từng nước, để tiện trong việc chọn các phân cụm clustering sau này

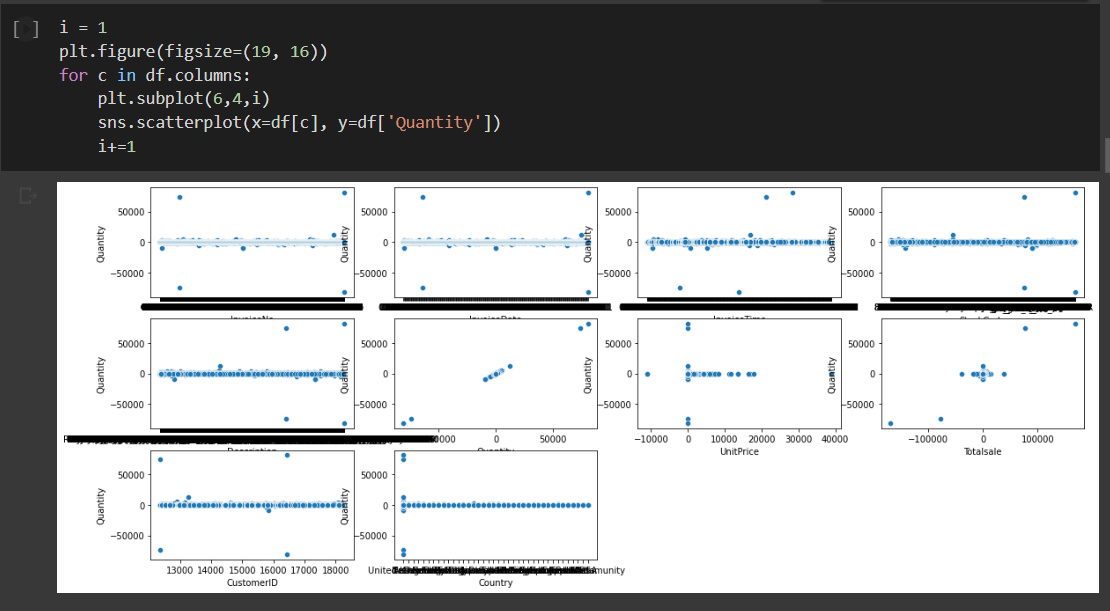
******

******

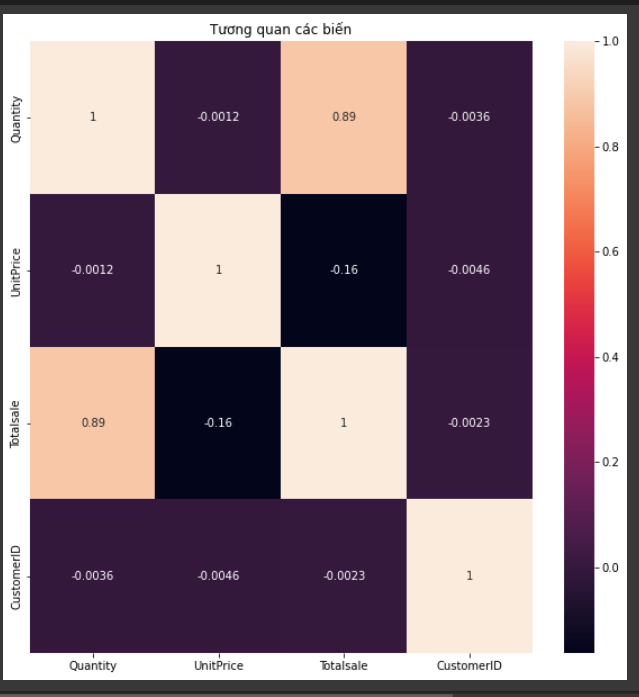
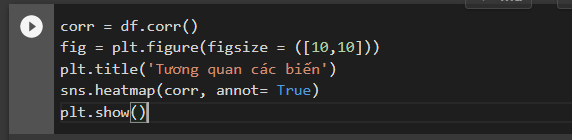
*-       Nhận thấy United Kingdom chiếm rất nhiều trong số đơn hàng (> 90%/ tổng số quốc gia), điều này giúp ta định hướng rằng, chúng ta sẽ tạo nhãn có thuộc tính country thành 2 loại : United Kingdom và các quốc gia khác.*

### b. Biểu diễn dữ liệu nguyên bản:

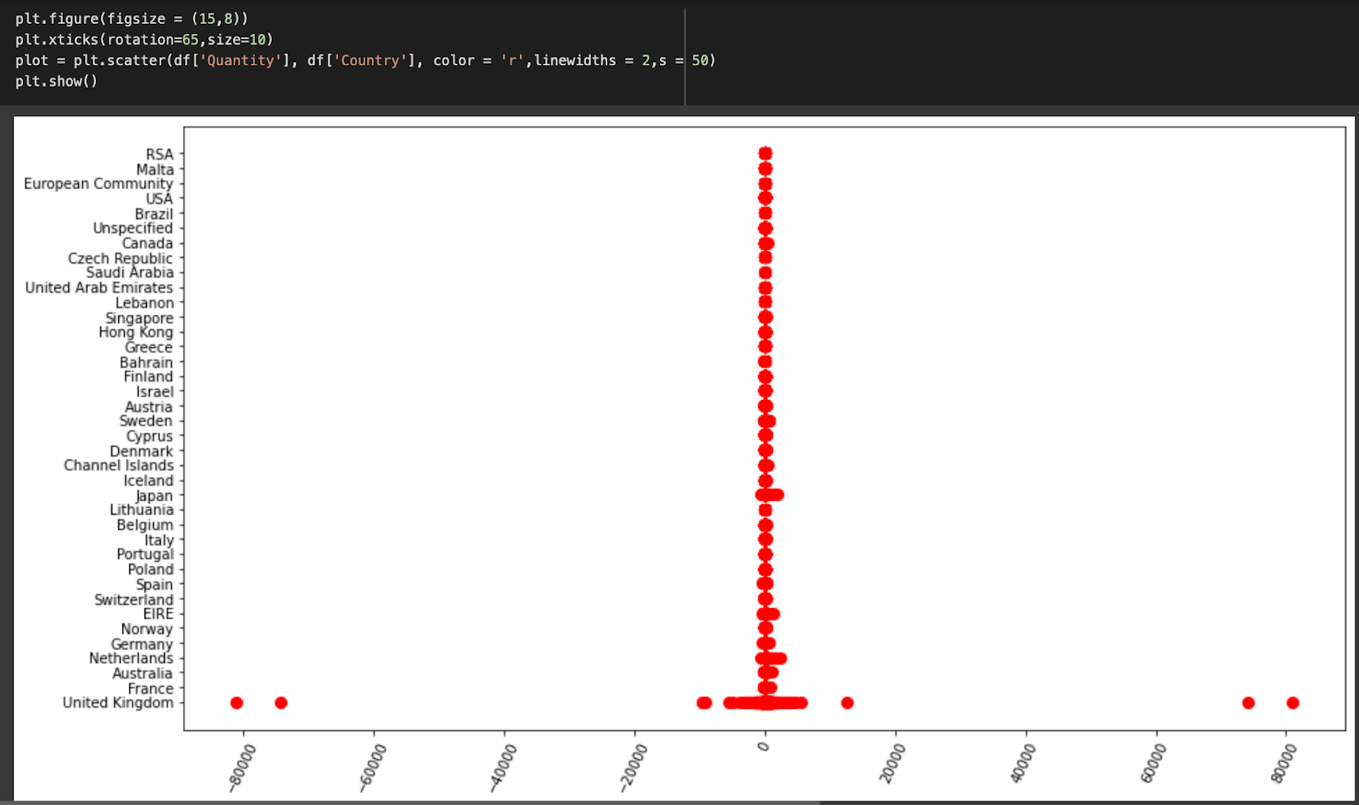
* Tiếp theo, việc phác hoạ biểu đồ tương quan giữa các biến làm nổi bật target của vấn đề. Qua biểu đồ cũng như các công thức liên quan, nhận thấy target của bộ dữ liệu này có thể sử dụng được biến Quantity, Totalsale cũng như UnitPrice như nhau



* Quá trình vẽ biểu đồ này sẽ tốn khoảng vài phút và vì bộ dữ liệu khá lớn nên phần biểu diễn biểu sẽ có những điểm đen dày đặc ở phần dưới mỗi biểu đồ.
* Dưới đây là biểu đồ tương quan giữa 4 biến, điểm số càng tiệm cận 1 chứng tỏ độ tương thích giữa chúng càng cao, và ngược lại, nếu điểm số về 0 hoặc bé hơn 0 thì chúng có tương quan ngược.



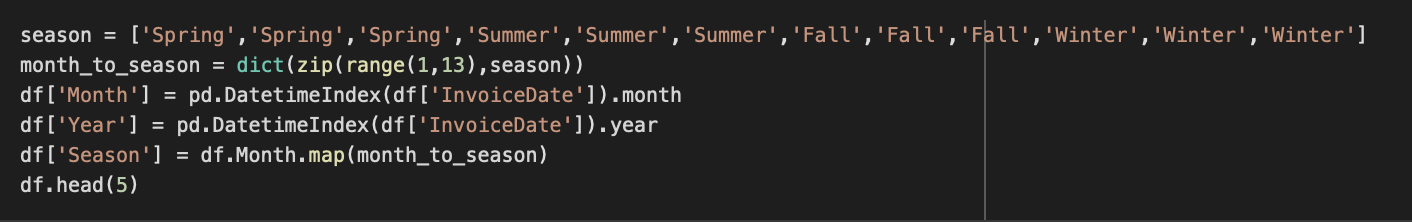
* *Nhận xét: EDA là bước giúp chúng ta định hình được các phương pháp mà chúng ta sẽ sử dụng sau này, ngoài ra với bộ dữ liệu này, nó giúp ta nhận thấy, muốn phân tích các trend theo thời gian, chúng ta cần phải tạo thêm các cột như tuần/ tháng hay gộp dữ liệu các ngày lại với nhau; Cần chỉnh dạng dữ liệu để hợp lý hoá các thông tin cần dùng*
* Kiểm tra xem bộ dữ liệu có tồn tại Outliers :

**

## 2. Chỉnh dạng dữ liệu

### a.     Thêm cột

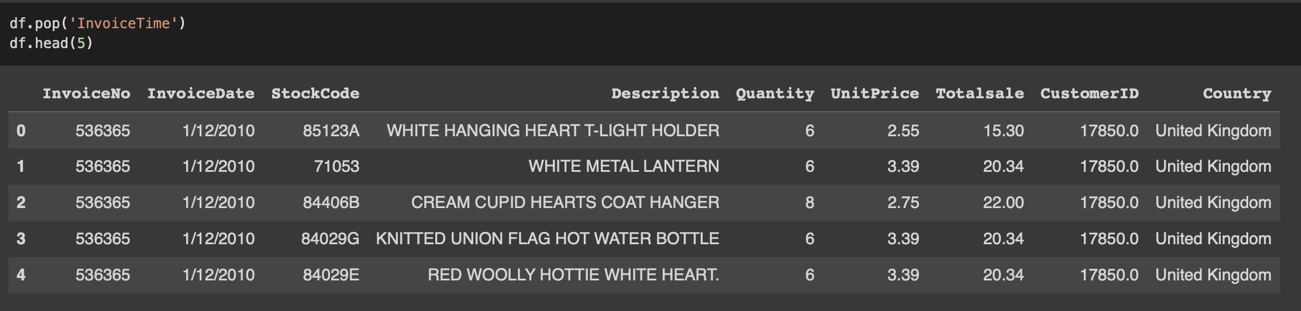
* Thêm vào cột Month:
* Lý do: Với mục tiêu phân loại các hóa đơn theo mùa, ta cần sắp xếp thời gian của các hoá đơn theo tháng (thay vì sắp xếp theo ngày) để thuận tiện cho việc phân loại. Nếu sắp xếp theo ngày thì không có ý nghĩa phân tích nhiều như tháng vì các ngày sẽ có sự trùng lặp rất nhiều.
* Thêm vào cột Year:
* Lý do: Bộ dữ liệu bao gồm thời gian bán hàng của năm 2010 và 2011, ta cần phân loại hoá đơn dựa theo năm để phục vụ cho việc so sánh 2 năm, từ đó biết được xu hướng mua hàng của người mua tập trung vào loại sản phẩm nào.
* Thêm vào cột Season:
* Lý do: Để phục vụ cho mục tiêu nghiên cứu ở mục 2: tìm ra các sản phẩm bán chạy theo mùa. Ta cần thêm vào bộ dữ liệu cột Season nhằm sắp xếp các hoá đơn có tháng 1/2/3 vào mùa xuân, tháng 4/5/6 vào mùa hè, tháng 7/8/9 vào mùa thu và các tháng 10/11/12 vào mùa đông. Như vậy, ở phần phân tích dữ liệu sẽ dễ dàng thực hiện hơn.





### b.     Xoá cột:

* Loại bỏ cột InvoiceTime.
* Lý do: Dữ liệu thời gian của tệp dữ liệu này không nhất quán giữa các khách hàng. Phụ thuộc vào mỗi vị trí địa lý khác nhau nên múi giờ đặt hàng sẽ có sự chênh lệch. Vì vậy, để khách quan và bao quát hơn, nhóm chỉ sử dụng dữ liệu dạng ngày của cột InvoiceDate

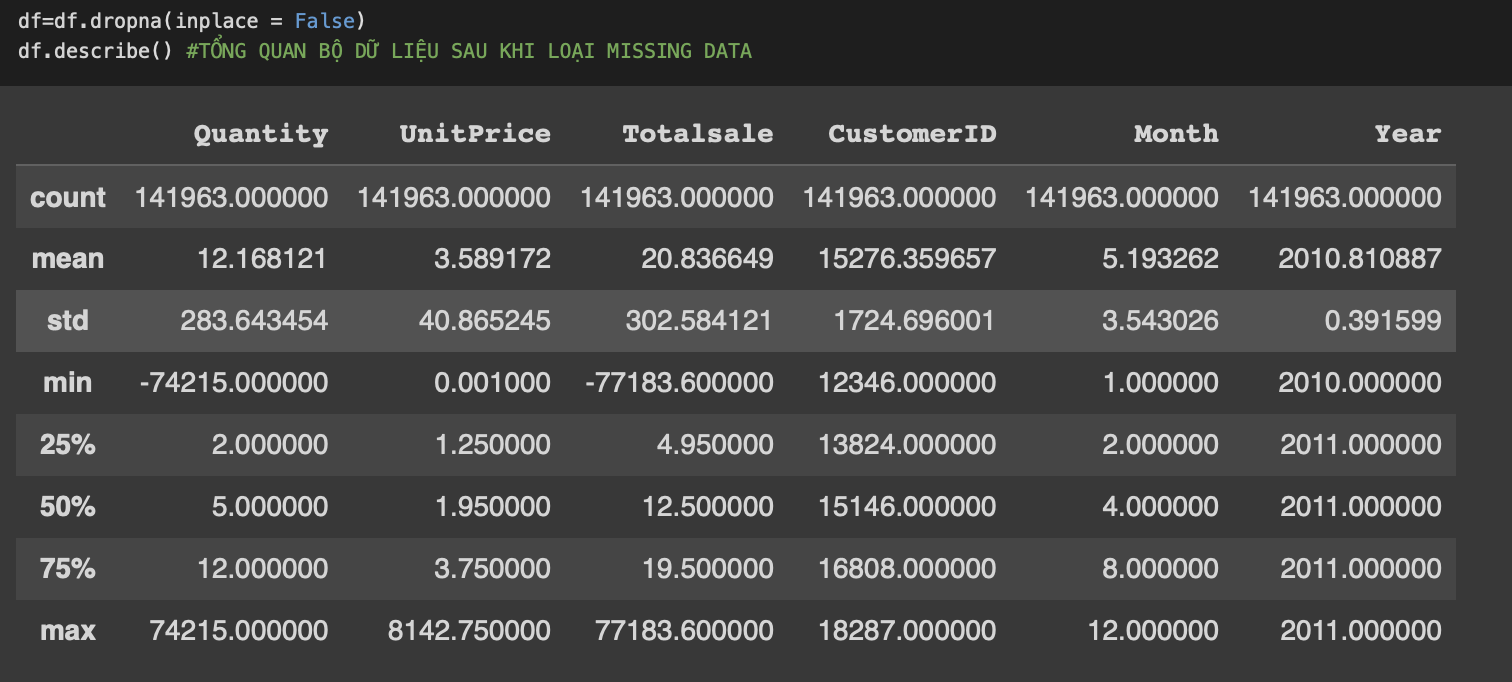


## 3.     Xử lý dữ liệu bị thiếu (Missing Values)

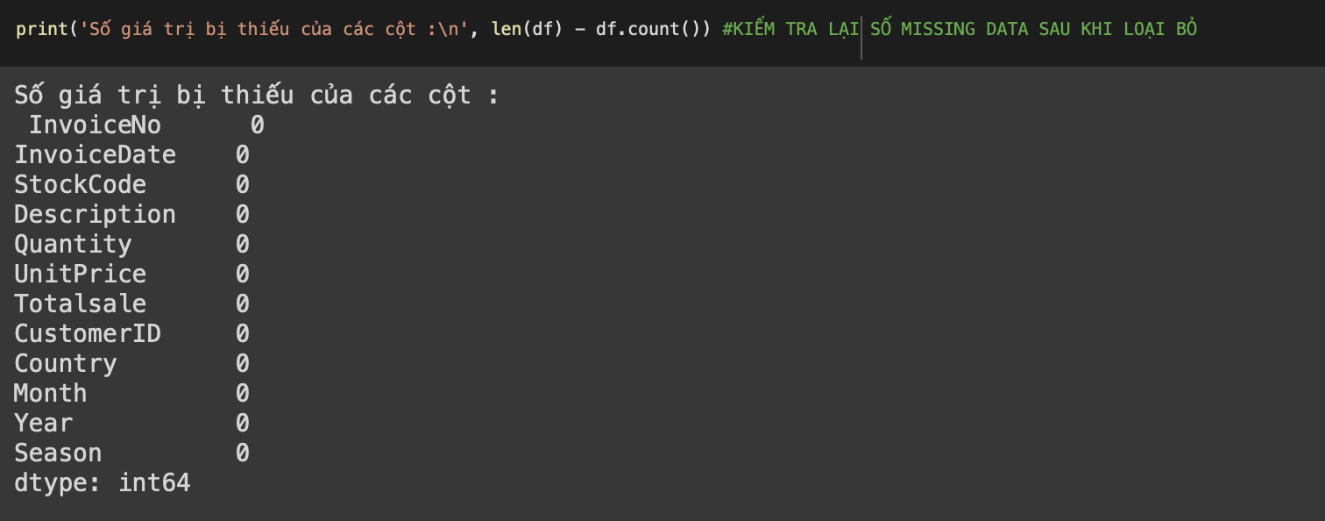
* Tiến hành kiểm tra số lượng dòng chứa giá trị bị thiếu của mỗi cột:



* Quan sát từ kết quả, ta thấy rằng chỉ có cột CustomerID là còn tồn lại missing values. Ta tiến hành loại bỏ luôn các dòng này ra khỏi bộ dữ liệu:



* Kiểm tra lại bộ dữ liệu sau khi xử lý:



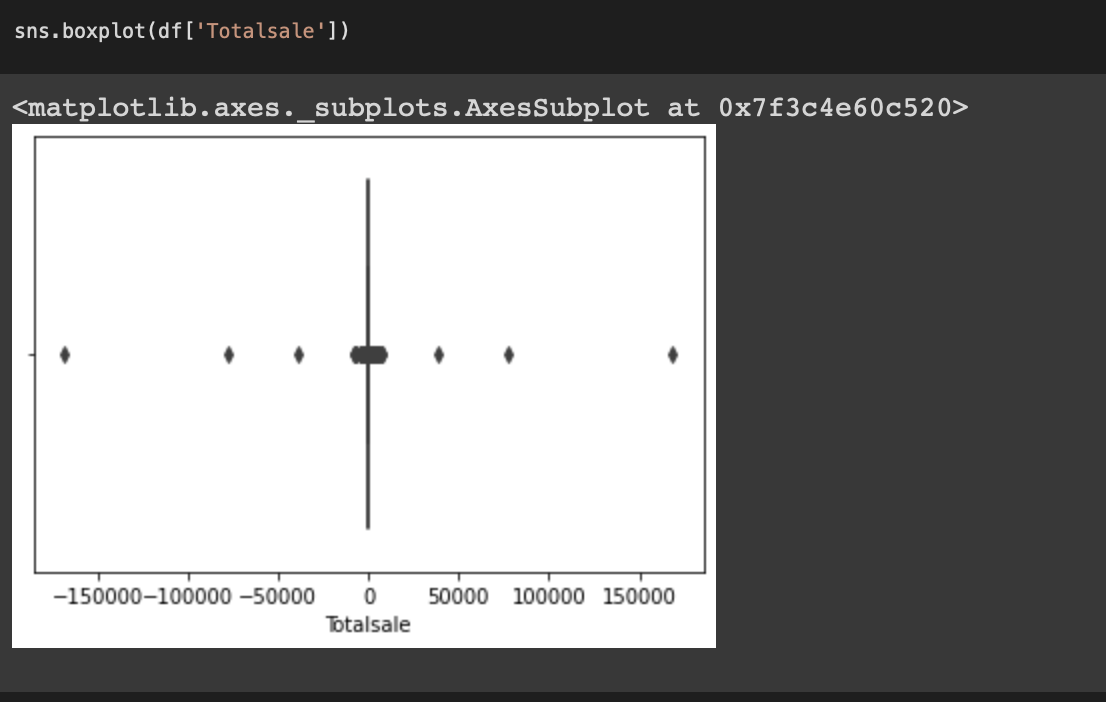
* Nhận xét: Sau khi đã bỏ đi các dòng chứa giá trị bị thiếu của cột CustomerID, bộ dữ liệu đã không còn missing values ở bất kỳ cột nào.



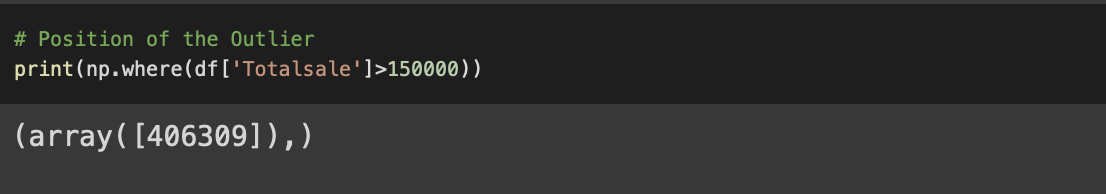
## 4.  Xử lý outliers của các biến định lượng

### a.     Biến Totalsale

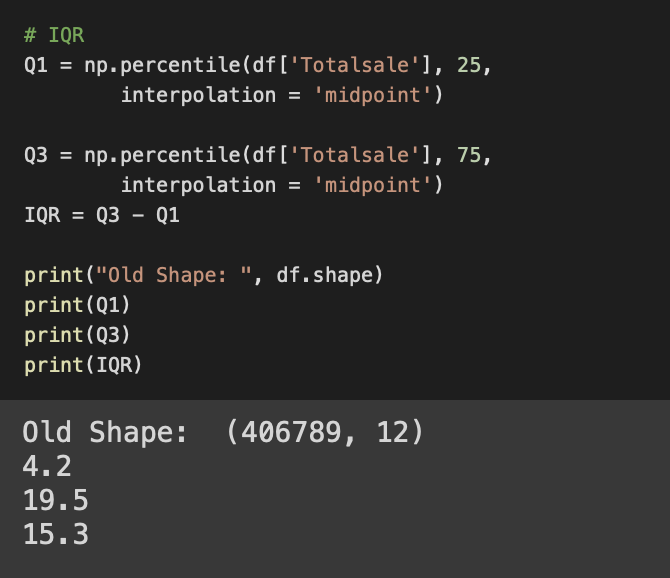
* Để quan sát trực quan dữ liệu của biến Totalsale, ta sẽ sử dụng biểu đồ hộp Boxplot để thể hiện các phân phối dữ liệu, nghĩa là giúp chúng ta biết được độ dàn trải của các điểm dữ liệu như thế nào, dữ liệu có đối xứng không, phân bố rộng hay hẹp, giá trị nhỏ nhất, lớn nhất và các điểm ngoại lệ.



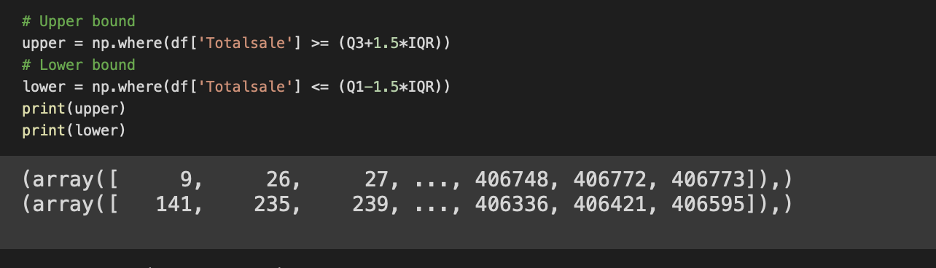
* Nhận xét: Dựa vào Boxplot phía trên, ta thấy giá trị Max = 150000 và Min = -150000 lần lượt là giá trị lớn nhất và bé nhất để một phần tử được xem là không phải Outliers. Vậy để xác định được Outliers của biến Total Sale, ta chỉ cần tìm các phần tử vượt quá 2 giá trị này.



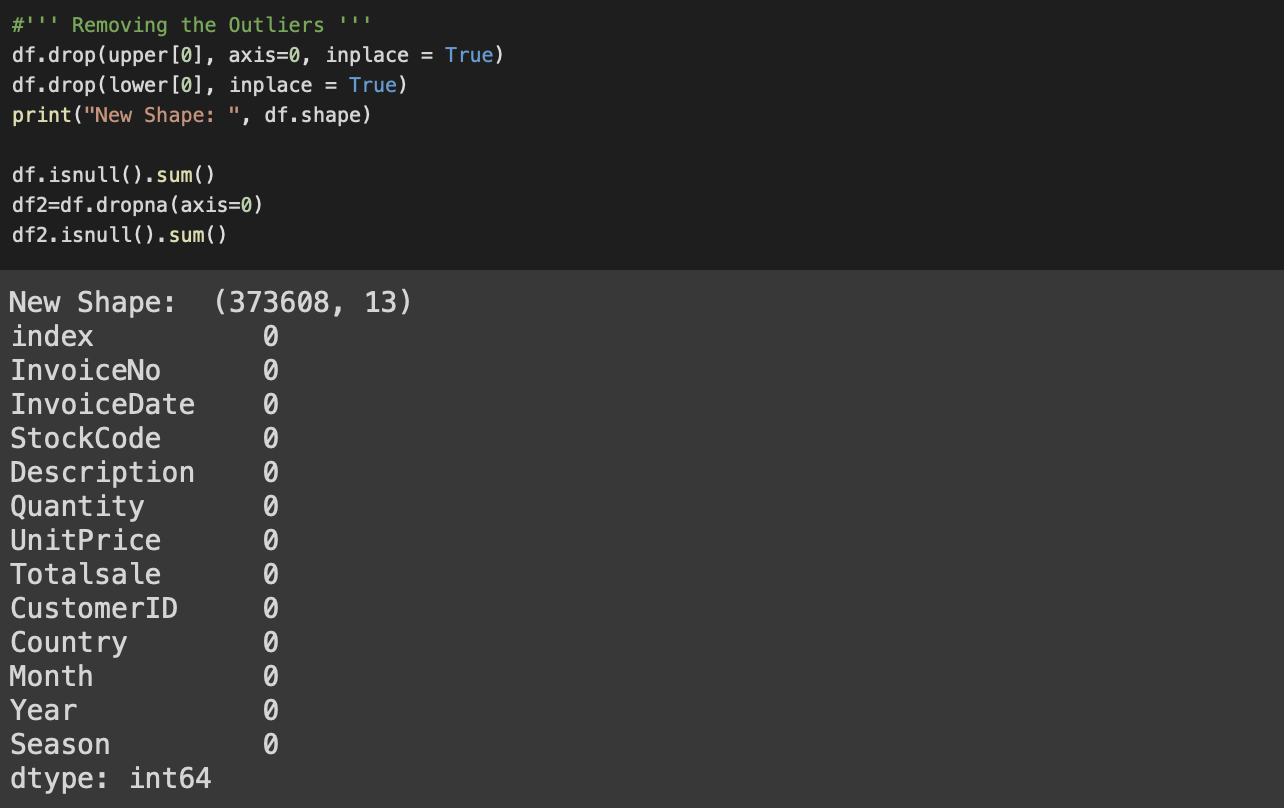
* Xác định giá trị của các tứ phân vị nhằm kiểm soát tác động của những giá trị ngoại lệ (Outliers) ở 2 đầu mút. Ở đây ta chỉ cần xác định giá trị của tứ phân vị thứ 1 và 3:



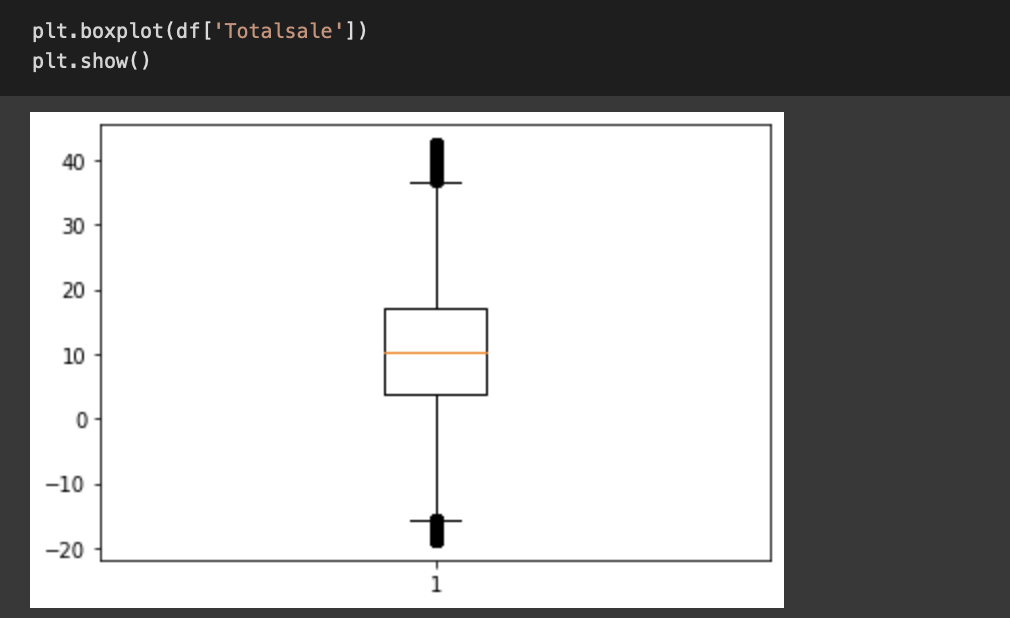
* Sau khi đã có Q1 và Q3, ta tiếp tục xác định giá trị biên của biến Total Sale, nếu giá trị của 1 phần tử bất kì nằm trong khoảng lower và upper thì gọi là expected data, ngược lại gọi là Outliers. Tạo 2 mảng chứa các index của phần tử là Outliers.



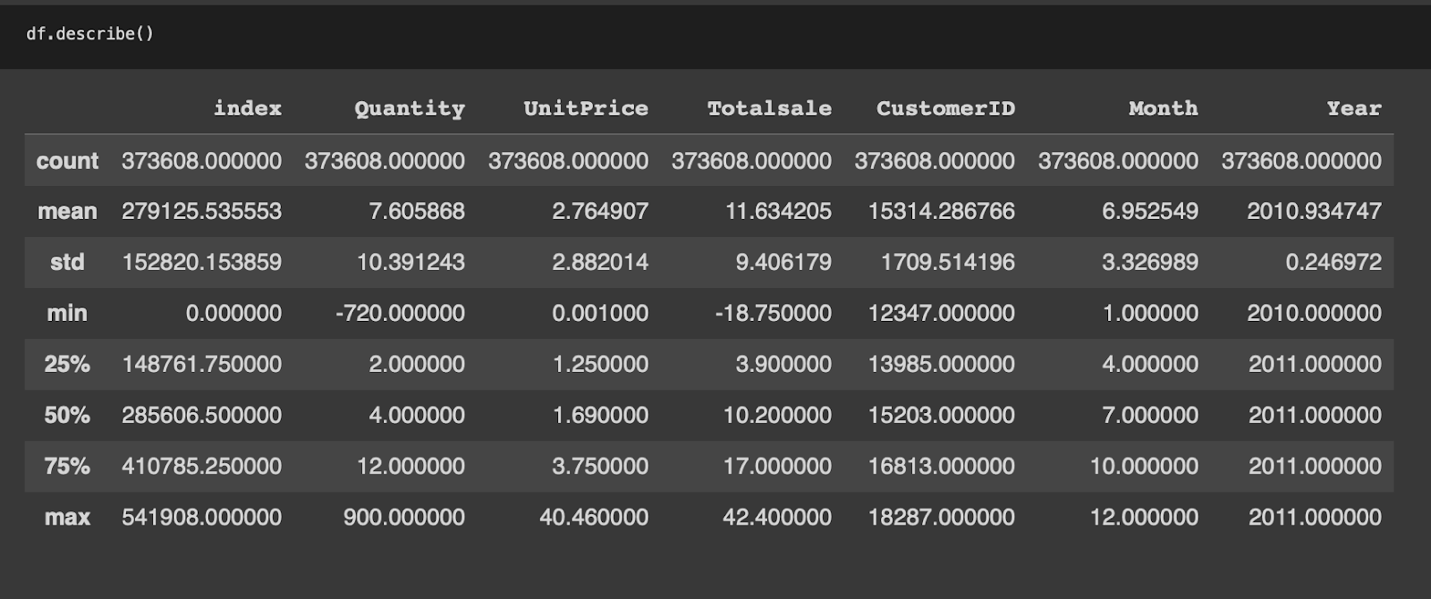
* Sau khi đã có index của các outliers, tiến hành loại bỏ outliers dựa trên index được lưu trong 2 array upper và lower.



* Sau khi đã xử lý các outliers, sử dụng Boxplot để quan sát bộ dữ liệu. Nếu phát hiện còn tồn tại Outliers thì ta tiến hành xử lý tiếp, ngược lại, kiểm tra bộ dữ liệu sau khi đã xử lý.

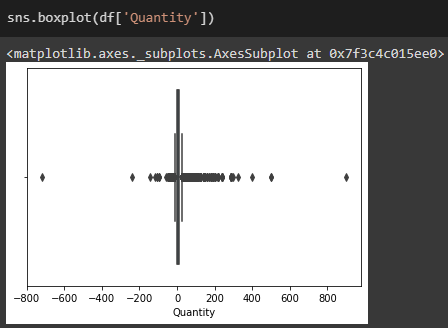


* Nhận xét: Biểu đồ hộp của biến Total Sale đã không còn Outliers, vậy ta sẽ xem tổng quan bộ dữ liệu đã thay đổi ra sao sau khi loại đi Outliers:



### b.   Biến Quantity

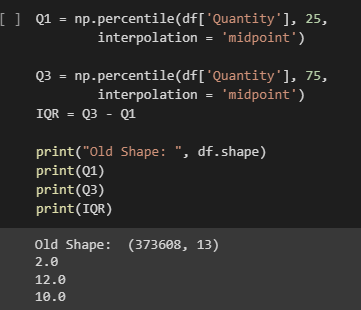
* Sử dụng biểu đồ hộp để xem các giá trị của Quantity phân phối như thế nào. Ở đây ta thấy chủ yếu dữ liệu dàn trải trong khoảng từ 0-300 và xuất hiện outliers ở 2 đầu mút -800 và 800. Vì vậy ta sẽ tiến hành loại bỏ các giá trị outliers này để tăng độ chính xác cho cho các mô hình phân tích phia sau.



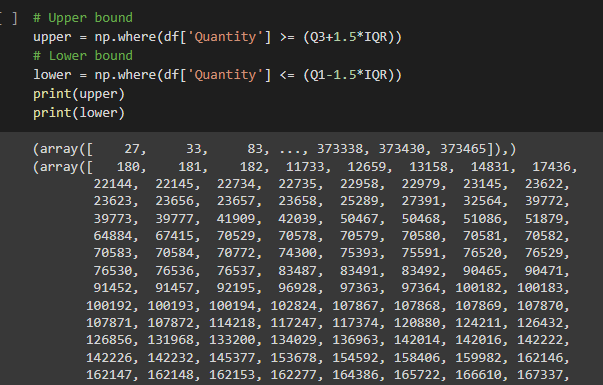
* Tìm Min, Max và các mô tả cơ bản về biến Quantity. Ở đây tồn tại giá trị âm là vì có các sản phẩm bị lỗi, hư hỏng và được hoàn về.



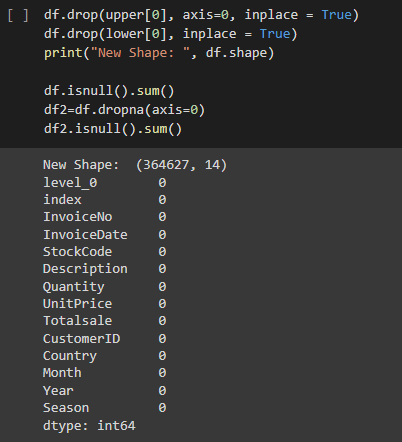
* Xác định giá trị của các tứ phân vị nhằm kiểm soát tác động của những giá trị ngoại lệ (Outliers) ở 2 đầu mút. Ở đây ta chỉ cần xác định giá trị của IQR, tứ phân vị thứ 1 và 3.



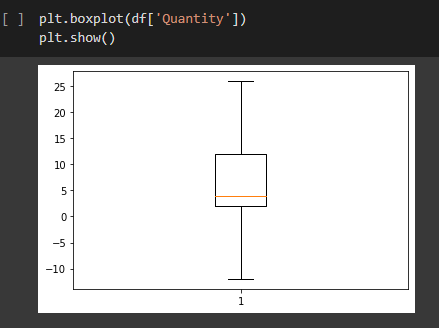
* Sau khi đã có Q1 và Q3, ta tiếp tục xác định giá trị biên của biến Total Sale, nếu giá trị của 1 phần tử bất kì nằm trong khoảng lower và upper thì gọi là expected data, ngược lại gọi là outliers. Tạo 2 mảng chứa các index của phần tử là outliers.



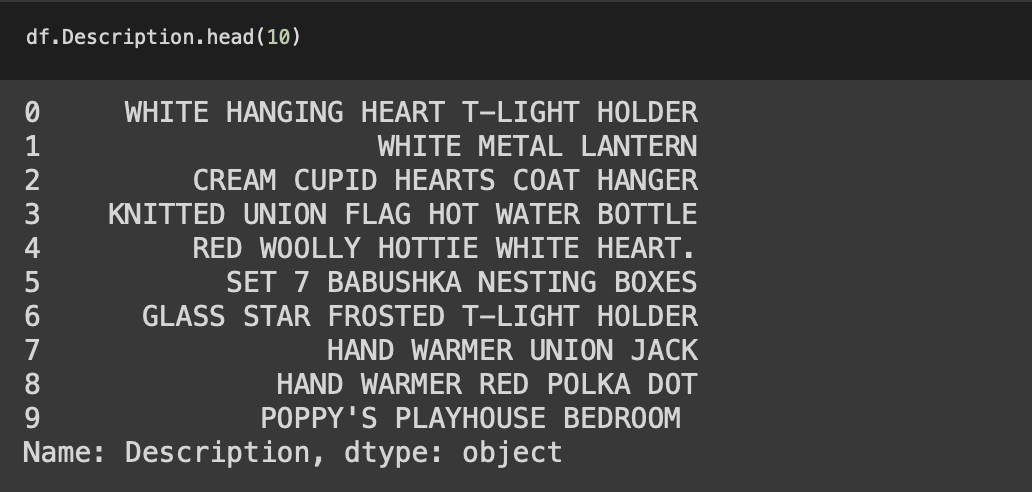
* Sau khi đã có index của các outliers, tiến hành loại bỏ outliers dựa trên index được lưu trong 2 array upper và lower.

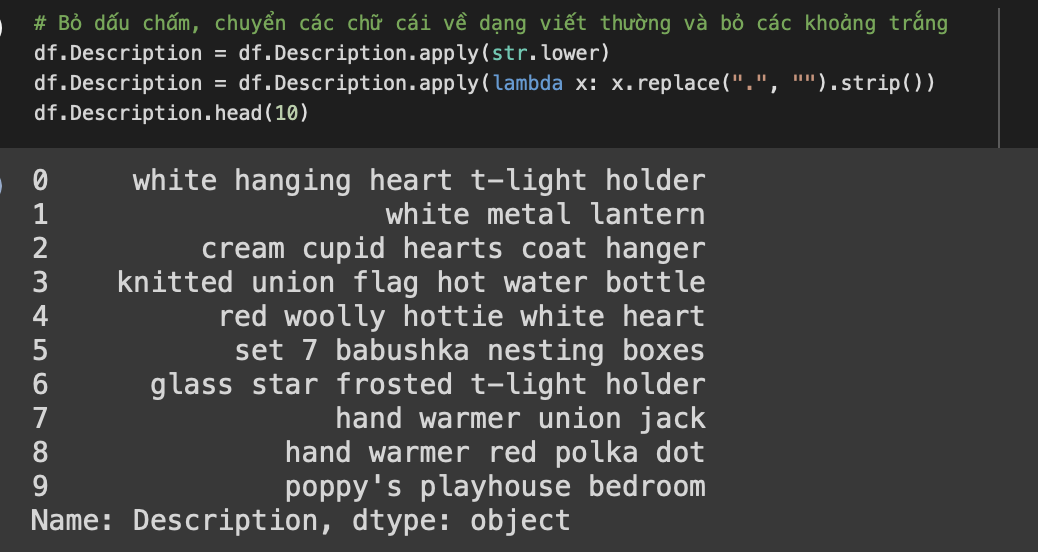


* Sau khi đã xử lý các outliers, sử dụng Boxplot để quan sát bộ dữ liệu. Nếu phát hiện còn tồn tại Outliers thì ta tiến hành xử lý tiếp, ngược lại, kiểm tra bộ dữ liệu sau khi đã xử lý.

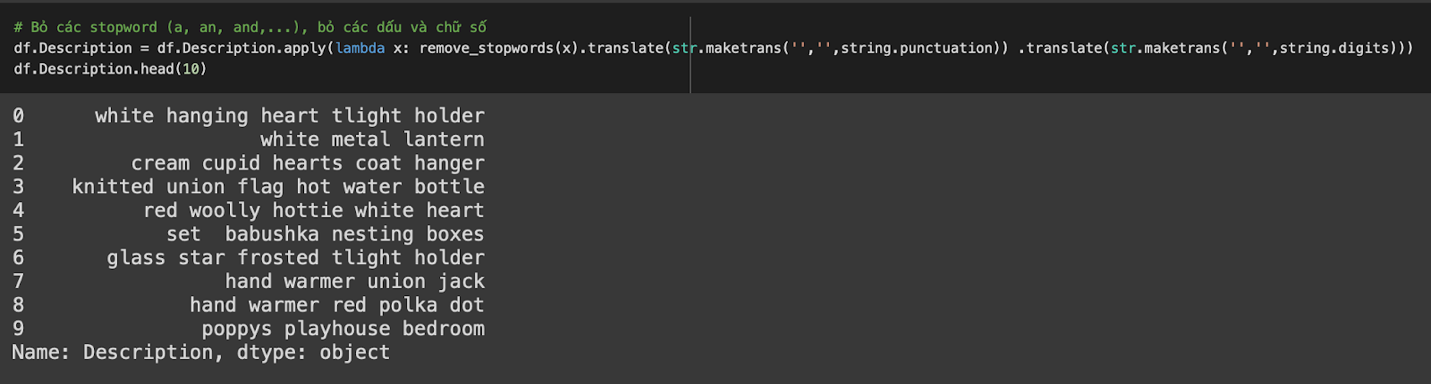


## 5.   Xử lý biến Description

* Biến Description bao gồm các mô tả sơ lược về loại hàng hóa mà người dùng đã mua, nên biến này sẽ có dạng chuỗi ký tự nhưng chưa thống nhất về cấu trúc. Chính vì vậy, ta cần cập nhật lại biến này sao cho ở các dòng, biến Description sẽ có cấu trúc giống nhau để phục vụ cho việc phân tích ở phần sau.
* Đầu tiên, ta cần xem sơ lược biến Description thông qua 10 dòng đầu của bộ dữ liệu:
* Quan sát thấy các kí tự đang ở dạng chữ in hoa nên ta tiến hành chuyển về chữ thường và loại luôn các khoảng trắng (nếu có):



* Bên cạnh đó, một vài chuỗi ký tự vẫn còn bao gồm số, các ký tự (-,’)  và các stopword (a, an, and,..) nên ta tiến hành loại bỏ.



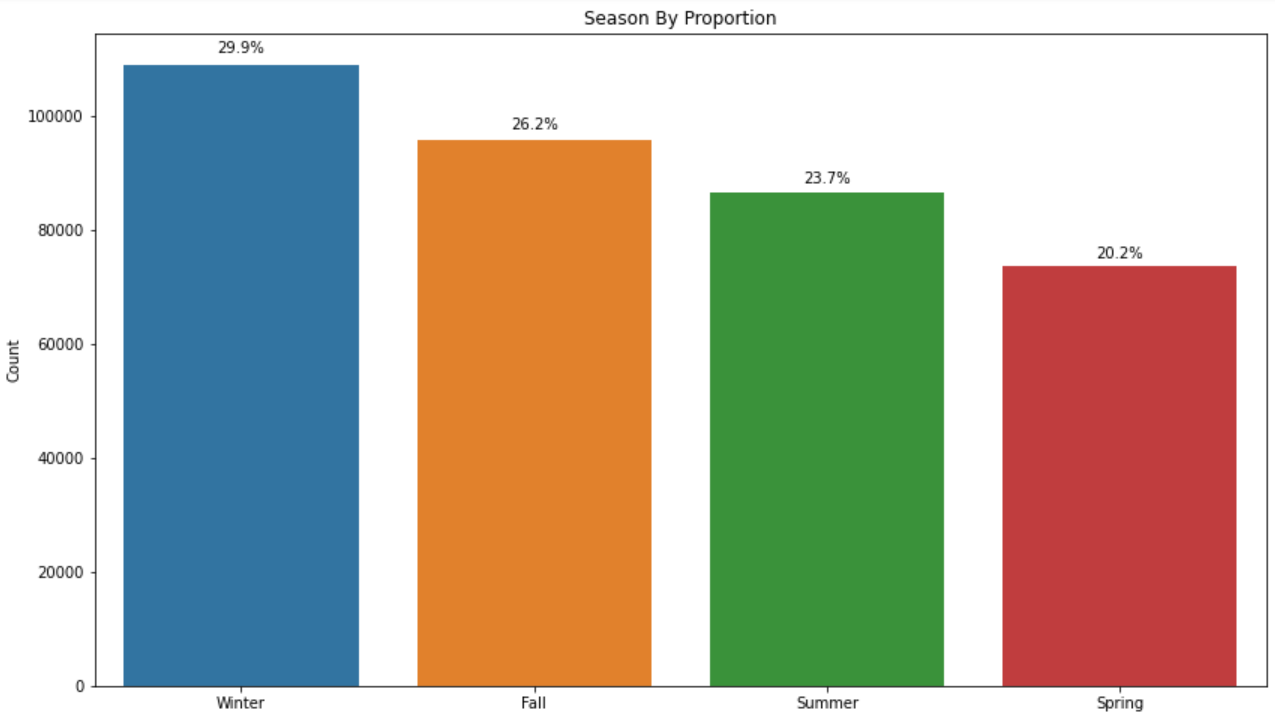
# Chương IV: Phân Tích Dữ Liệu

## Time Series:

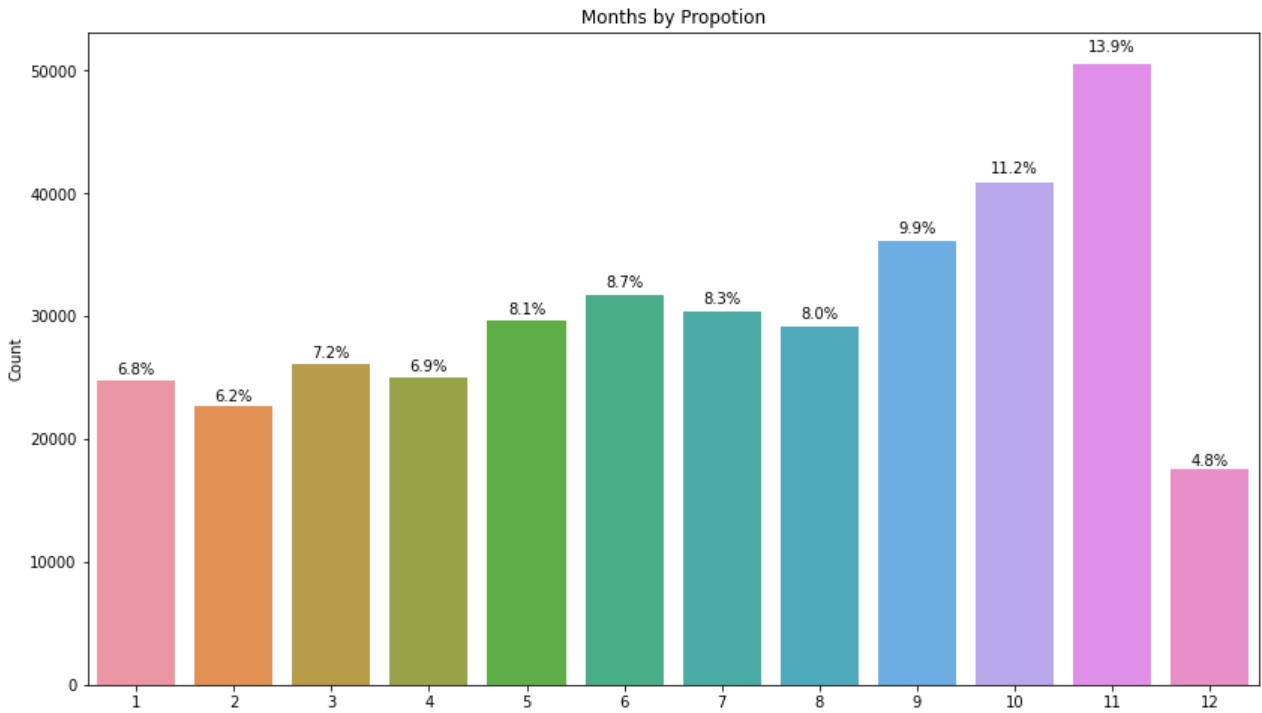
Trong xã hội hiện đại, khi lượng thông tin càng ngày càng tăng, việc phân tích dữ liệu time series trở nên cực kỳ quan trọng, nhưng cho dù có nhiều thuật toán hay mô hình trí tuệ nhân tạo sinh ra, việc dự đoán tương lai vẫn thất bại do sự thiếu ổn định trong chuỗi thời gian.

Bằng cách phân tích dữ liệu theo thời gian giúp chúng ta có một cái nhìn tổng quan về tình hình bán hàng của doanh nghiệp, nhận biết tốt hơn về xu hướng của dữ liệu và kiểm tra xem liệu có tính mùa vụ trong bộ dữ liệu không nhằm hỗ trợ đưa ra các quyết định bán hàng trong tương lai.

Một cái nhìn tổng quát, doanh nghiệp có xu hướng được đặt hàng tăng dần từ đầu năm đến cuối năm khi lượng đặt hàng cao nhất là vào mùa đông (~110,000 đơn hàng) và lượng đặt thấp nhất là vào mùa xuân (~78,000 đơn hàng).



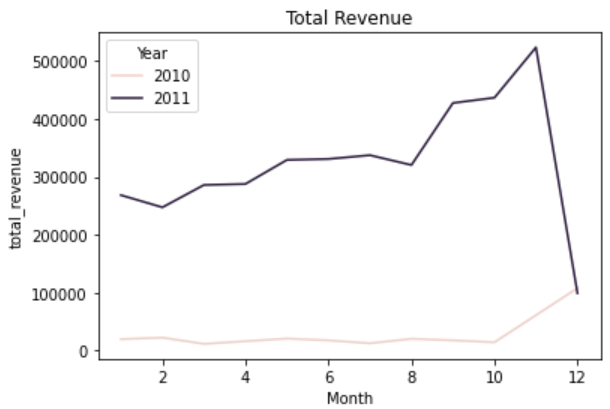
Để có một cái nhìn tốt hơn, ta hãy nhìn vào lượng đặt theo tháng. Mặc dù xu hướng đặt hàng vẫn tăng dần từ đầu năm đến cuối năm và đạt đỉnh vào tháng 11 (~50,000 đơn hàng), tuy nhiên  sau đó lại giảm mạnh ngay lập tức vào tháng 12 (~18,000 đơn hàng), thấp nhất trong tất cả các tháng.



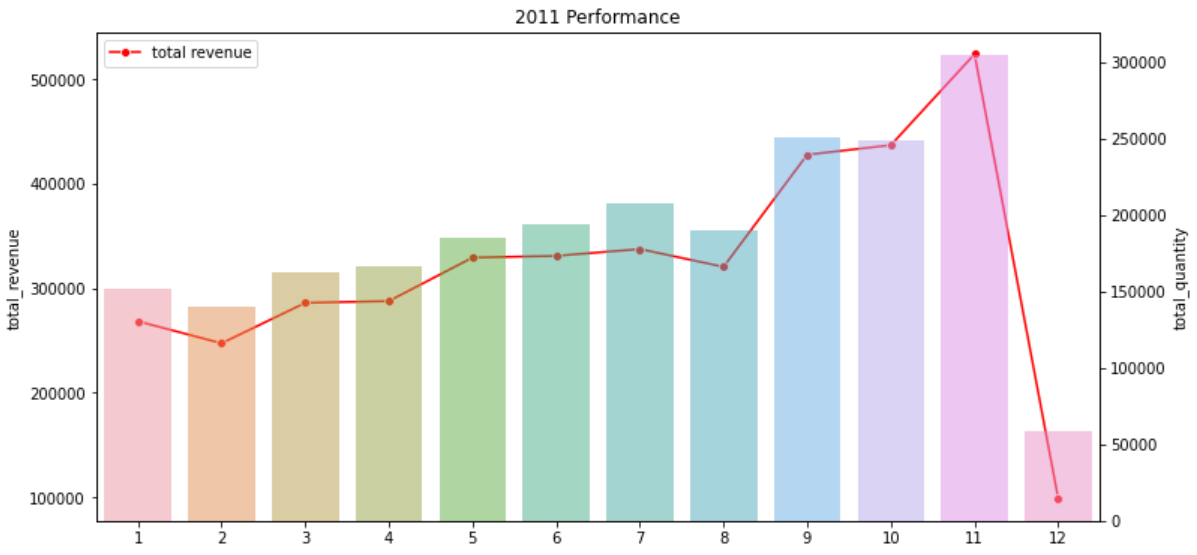
Bên cạnh đó, ta cũng có thể dễ dàng nhìn thấy lượng đơn hàng được đặt trong năm 2010 không đáng kể so với năm 2011 (6.5% đơn hàng của năm 2010 so với 93.5% đơn hàng của năm 2011 trên tổng gần 500,000 đơn hàng).



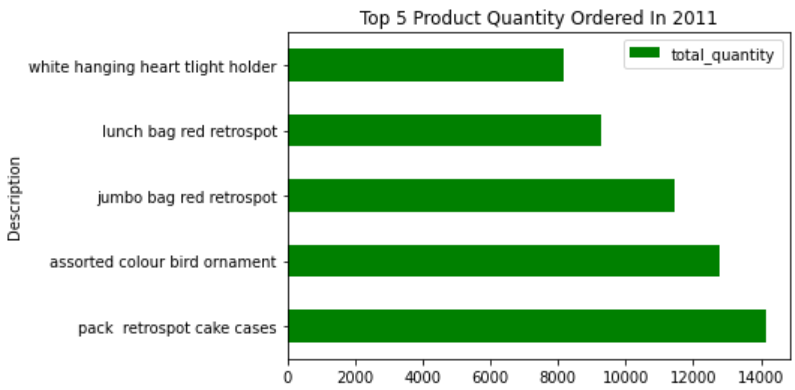
Chính vì lượng đơn hàng vượt trội trong năm 2011 so với năm 2010 đã giúp cho tổng doanh thu theo tháng của năm 2011 đều cao hơn năm 2010, riêng chỉ có khoảng thời gian tháng 12, doanh thu của năm 2011 giảm mạnh, khác với năm 2010 khi vào thời điểm cuối năm là lúc doanh thu nảy bật.



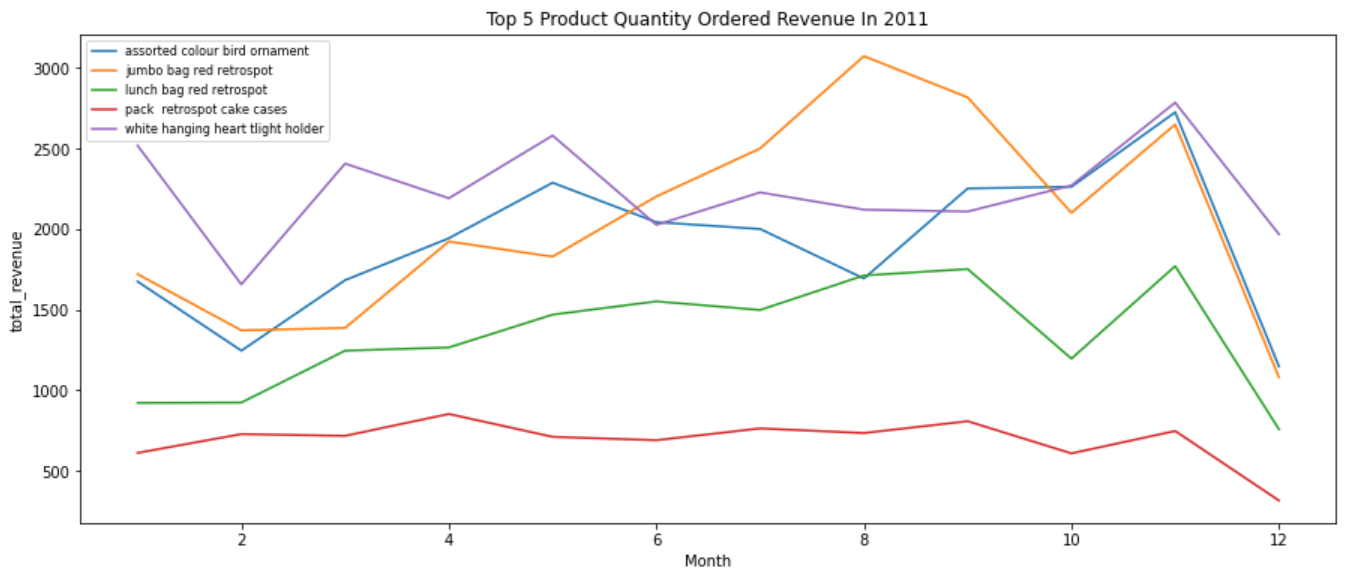
Ta có thể thấy ảnh hưởng của năm 2011 nhiều hơn đáng kể so với năm 2010, vậy nên ta sẽ cùng nhau có một cái nhìn rõ hơn vào năm 2011. Xu hướng tăng của lượng hàng và doanh thu song hành với nhau, Vậy ta có thể kết luận rằng, giá sản phẩm không tác động nhiều đến doanh thu như số lượng sản phẩm bán ra.



Tiếp đến ta cùng xem 5 sản phẩm có lượng hàng được bán ra cao nhất trong năm 2011. Đứng đầu danh sách là sản phẩm *pack retrospot cake cases* với số lượng ~ 14,000 sản phẩm được bán ra trong năm 2011.



Tuy *pack retrospot cake cases* là sản phẩm có số lượng bán hàng cao nhất nhưng doanh thu mang lại cho doanh nghiệp không đáng kể so với 4 sản phẩm còn lại khi doanh thu trung bình mỗi tháng chỉ rơi vào khoảng $600 - $700, chiếm một phần rất nhỏ khi doanh thu trung bình tháng của doanh nghiệp rơi vào khoảng $300,000 - $450,000.

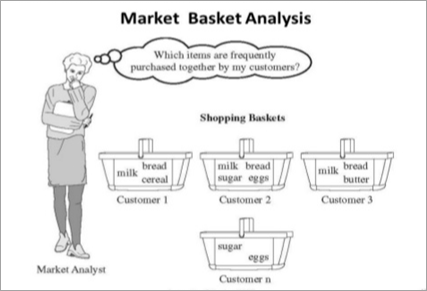


## FP-Growth:

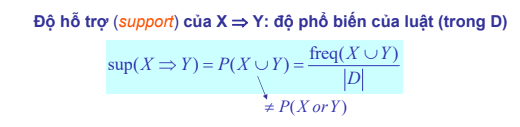
Mục tiêu: tìm ra các mặt hàng thường được bán chạy với nhau và mùa được bán chạy nhất để kết hợp sản xuất, quảng cáo giúp tăng lượng hàng được bán ra từ đó giúp tăng đáng kể doanh thu. Sử dụng FP- Growth để tìm ra các mặt hàng thường được bán chung với nhau (trong cùng một hoá đơn) và khoảng thời gian mà các mặt hàng được bán chạy nhất giúp tối ưu hoá lượng hàng được bán ra.

a) Lý thuyết FP-Growth:

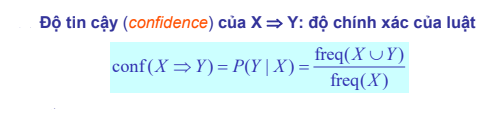
FP-Growth là một thuật toán được cải tiến từ thuật toán Apriori, một thuật toán dùng để giải quyết bài toán kinh điển “giỏ siêu thị”. Bài toán được đặt ra với việc làm sao để biết các mặt hàng thường xuyên được mua chung với nhau để từ đó, có các chương trình khuyến mãi, giảm giá, cách sắp xếp các mặt hàng trên kệ để tối ưu hóa được lượng hàng bán ra. Thuật toán được ứng dụng vào rất nhiều lĩnh vực và bài toán khác nhau ví dụ như: giám sát và phát hiện các tội phạm tiềm năng thông qua các chuỗi hành vi đã được nghiên cứu từ trước, các công ty có nhu cầu tìm ra các mặt hàng thường được bán cùng với nhau để có các biện pháp giúp gia tăng sức bán.



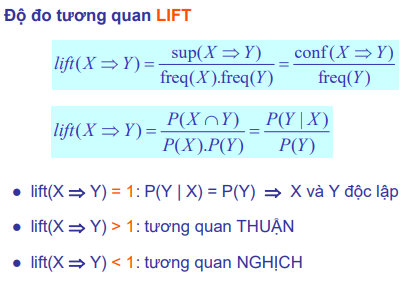
* **Các đại lượng được sử dụng trong thuật toán và các bước thực hiện**:
* Độ hỗ trợ (support) của X Y: độ phổ biến của luật kết hợp trong tập dữ liệu D



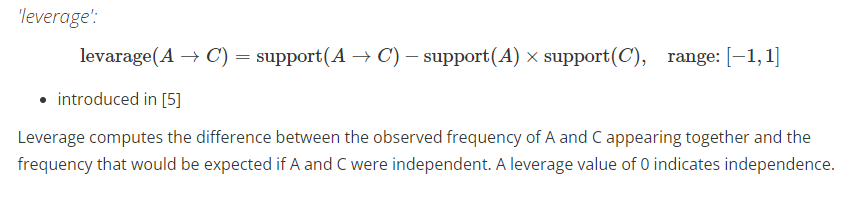
* Độ tin cậy (confidence) của X Y: độ chính xác của luật kết hợp



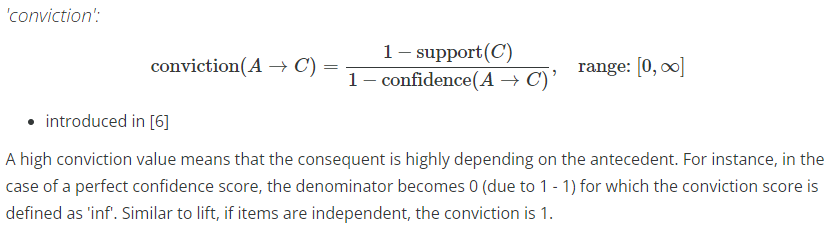
* Độ đo tương quan lift thường được sử dụng để đo tần suất xảy ra cùng nhau của antecedent A và consequence C của luật kết hợp AC so với mong đợi nếu chúng độc lập về mặt thống kê. Nếu A và C độc lập thì lift = 1. Đây là độ đo nhằm xác định trong trường hợp một luật kết hợp “mạnh” nhưng có cần thiết sử dụng hay không.



* Độ đo leverage tính toán sự khác biệt giữa tần suất quan sát được của A và C xuất hiện cùng nhau và tần suất được mong đợi nếu A và C độc lập. Leverage = 0 biểu thị tính độc lập.



* Độ đo conviction có giá trị cao có nghĩa là consequence C phụ thuộc nhiều vào antecedent A trong luật kết hợp AC. Chẳng hạn, trong trường hợp độ confidence hoàn hảo, mẫu số của độ đo conviction sẽ bằng 0 (do 1 - 1) với kết quả là conviction có giá trị “vô cực (inf)”'. Tương tự như độ đo leverage, nếu các item độc lập, kết quả sẽ là 1.



* **Các bước thực hiện thuật toán FP-Growth:**

Bước 1: Xây dựng F-list từ tập dữ liệu L1 (tập dữ liệu chứa tần số của từng item ban đầu)

Bước 2: Xây dựng FP-tree và Item Header table

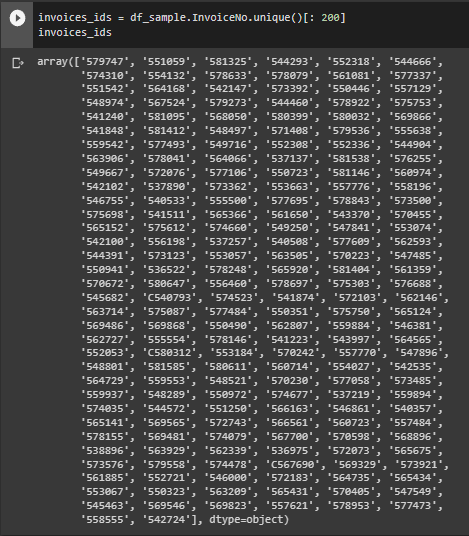
Bước 3: Duyệt “ngược” các item IK F-list để tạo các tập phổ biến

* Tạo cơ sở mẫu điều kiện CPB (conditional pattern base) từ prefix của các paths đến  IK
* Dùng CPB như CSDL để tạo FP-tree điều kiện.
* Tạo Item Header Table chỉ bao gồm các items phổ biến (cộng dồn freq(item) trong các paths).
* Phát sinh tập phổ biến từ mỗi path trong FP-tree điều kiện
* **Các bước thực hiện*:***

Bước 1: Lấy mẫu ngẫu nhiên 350000 dòng (lấy random\_state = 8 nhằm giữ nguyên kết quả cho mục đích nghiên cứu)

****

Bước 2: Lấy 200 dữ liệu InvoiceNo đầu nhằm giảm sự phức tạp của thuật toán FP-Growth. Lưu vào array có tên là invoice\_ids.

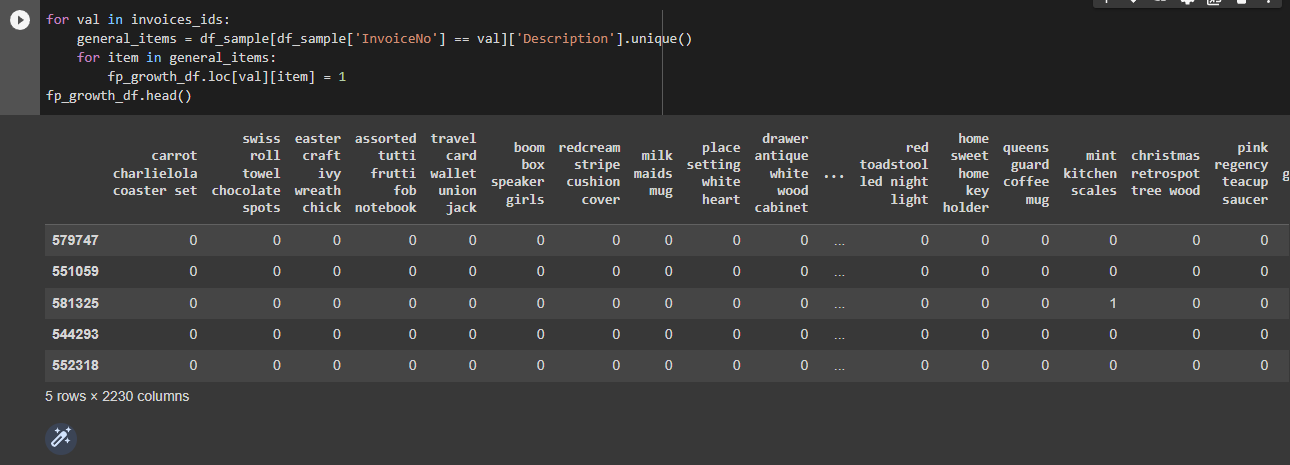
****

Bước 3: Tạo ra dataframe với dòng là các dữ liệu “InvoiceNo” và các cột là tên các sản phẩm (“Description”). Dựa trên array invoice\_ids, ta lấy các “Description” rồi đẩy dữ liệu vào một array khác có tên là all\_items. Các cột “Description” được tạo ra dựa vào việc ghép các “Description” đã được đưa array all\_items (ghép thành 1 set dữ liệu). Đưa hết toàn bộ dữ liệu của dataframe về 0.

Đây là bước chuẩn bị dataframe để tiếp đến đánh dấu các “Description” có trong “InvoiceNo”.

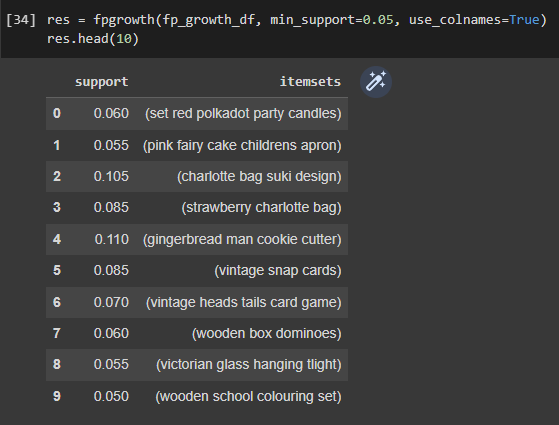
****

**Bước 4:** Đánh dấu các “Description” nếu nó có tồn tại trong “InvoiceNo” tương ứng.Đánh dấu bằng giá trị “1” đối với các “Description” có tồn tại trong “InvoiceNo” được xét.

****

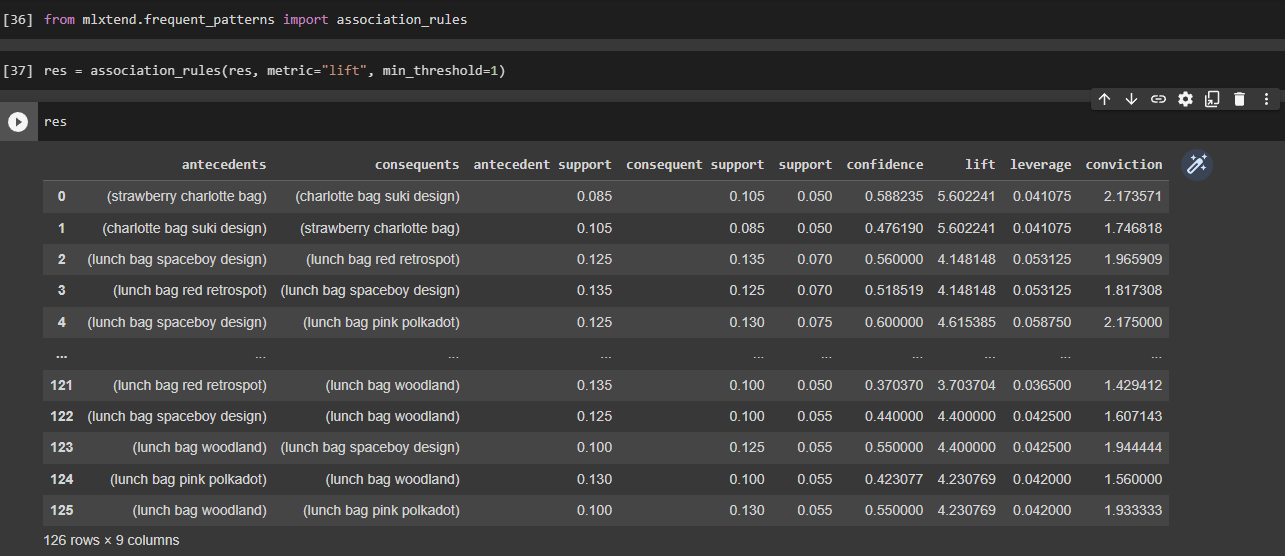
Bước 5:Sử dụng hàm fpgrowth để tính toán độ support của các items, lấy min\_support là 0.05.

**mlxtend.frequent\_patterns.fpgrowth(df, min\_support = 0.5, use\_colnames = False, max\_len = None, verbose = 0)**

****

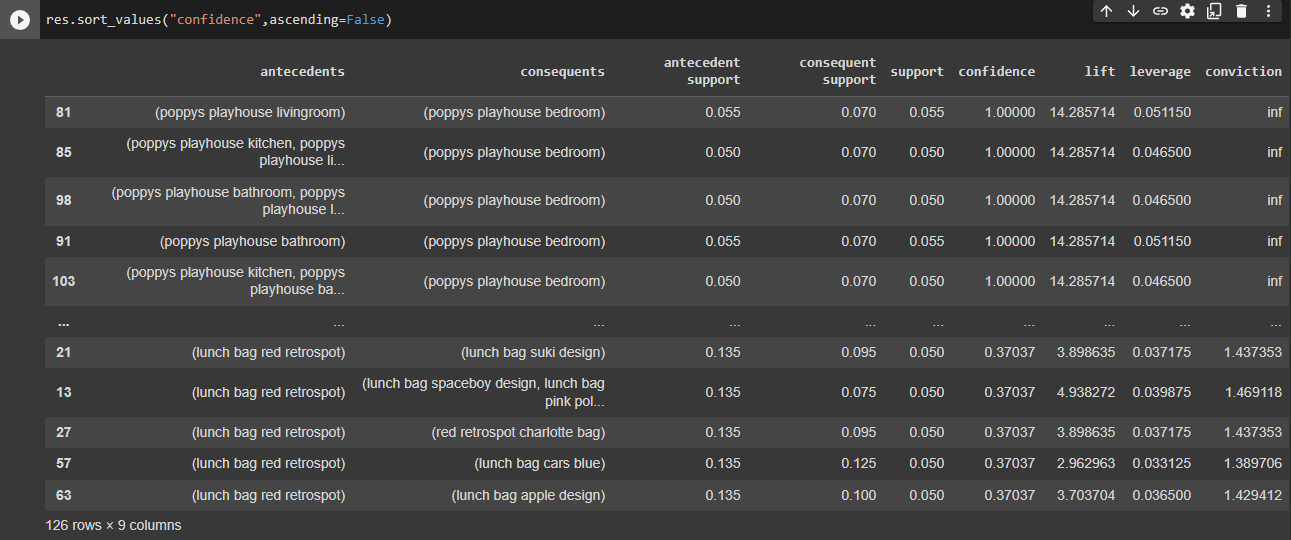
Bước 6:Lọc ra các luật kết hợp có độ đo lift 1

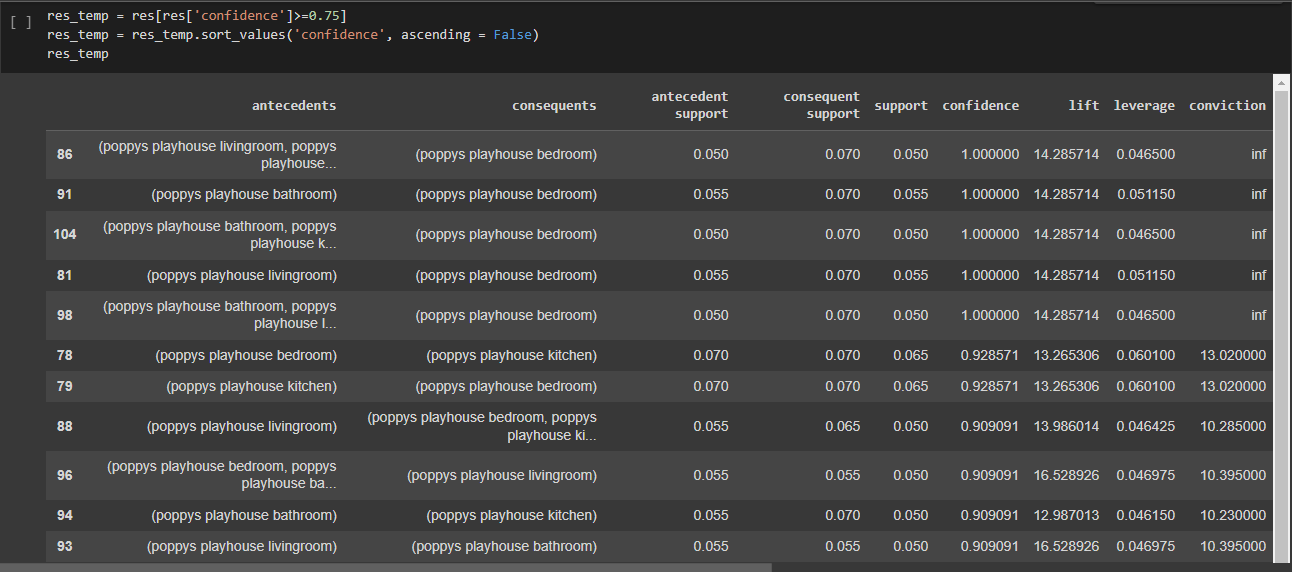
**mlxtend.frequent\_patterns.association\_rules(df, metric = “confidence”, min\_threshold = 0.8, support\_only = False)**

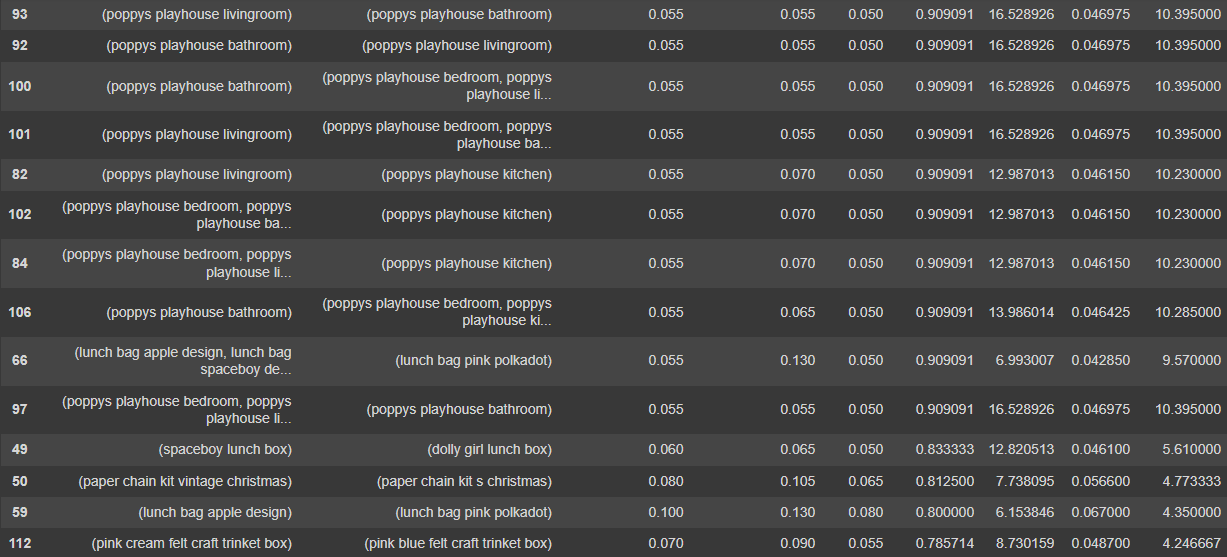
****

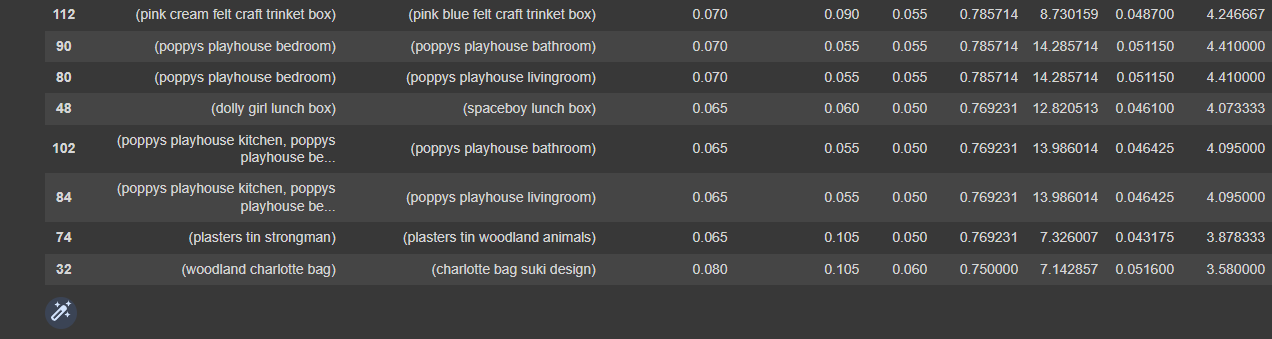
**Bước 7:** Sắp xếp các luật kết hợp giảm dần theo độ “confidence” và lọc ra các luật kết hợp có “confidence” 0.75.

*Nhận xét*: Ta quan sát thấy được rằng, đa phần các sản phẩm thường đi chung với nhau là “poppys playhouse livingroom”, “poppys playhouse bedroom”, “poppys playhouse bathroom”, “poppys playhouse kitchen”. Đặc biệt là 5 luật kết hợp đầu tiên có độ “confidence” và “conviction” tuyệt đối (confidence = 1, conviction = inf). Chính vì vậy, ta cần tập trung chú ý đến các mặt hàng này.

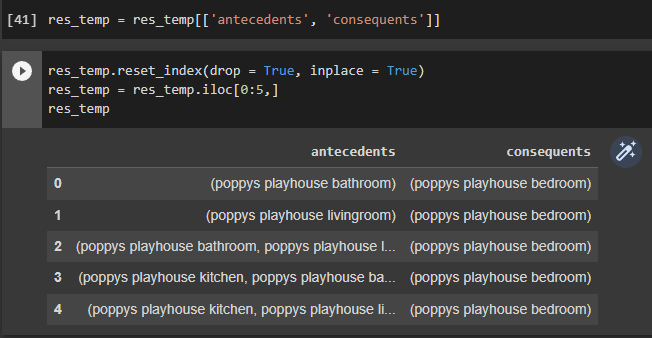
****

****

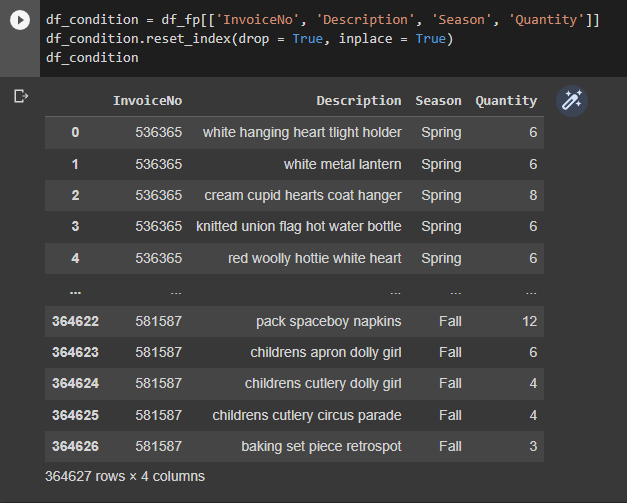
****

****

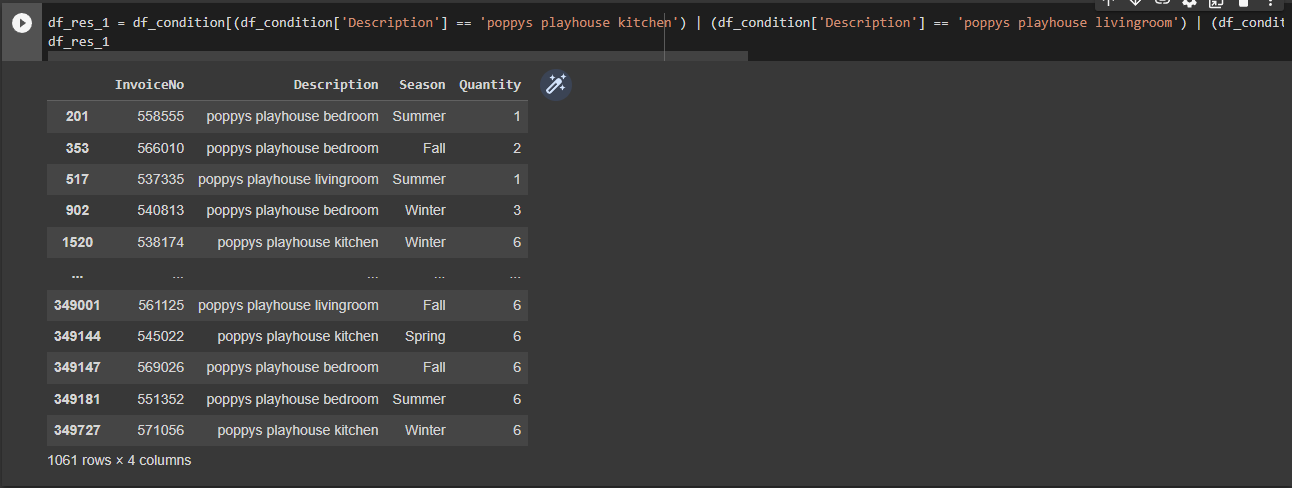
Bước 8:Lọc ra 5 luật kết hợp đầu tiên (5 luật kết hợp với confidence = 1 và conviction = inf)

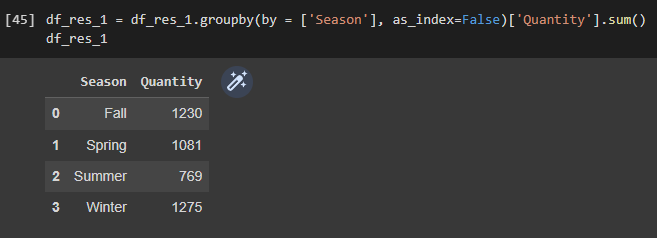
****

Bước 9**:** Lấy bộ dữ liệu gốc chưa lấy mẫu ngẫu nhiên với các cột “InvoiceNo”, “Description”, “Season”, “Quantity” để phục vụ cho mục đích so sánh đối chiếu lượng hàng được bán ra của các mặt hàng được chú ý.

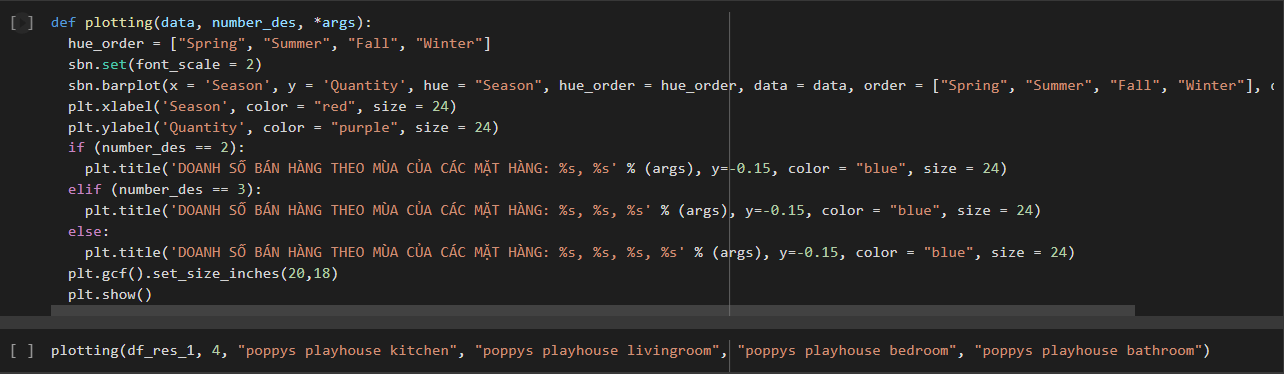
****

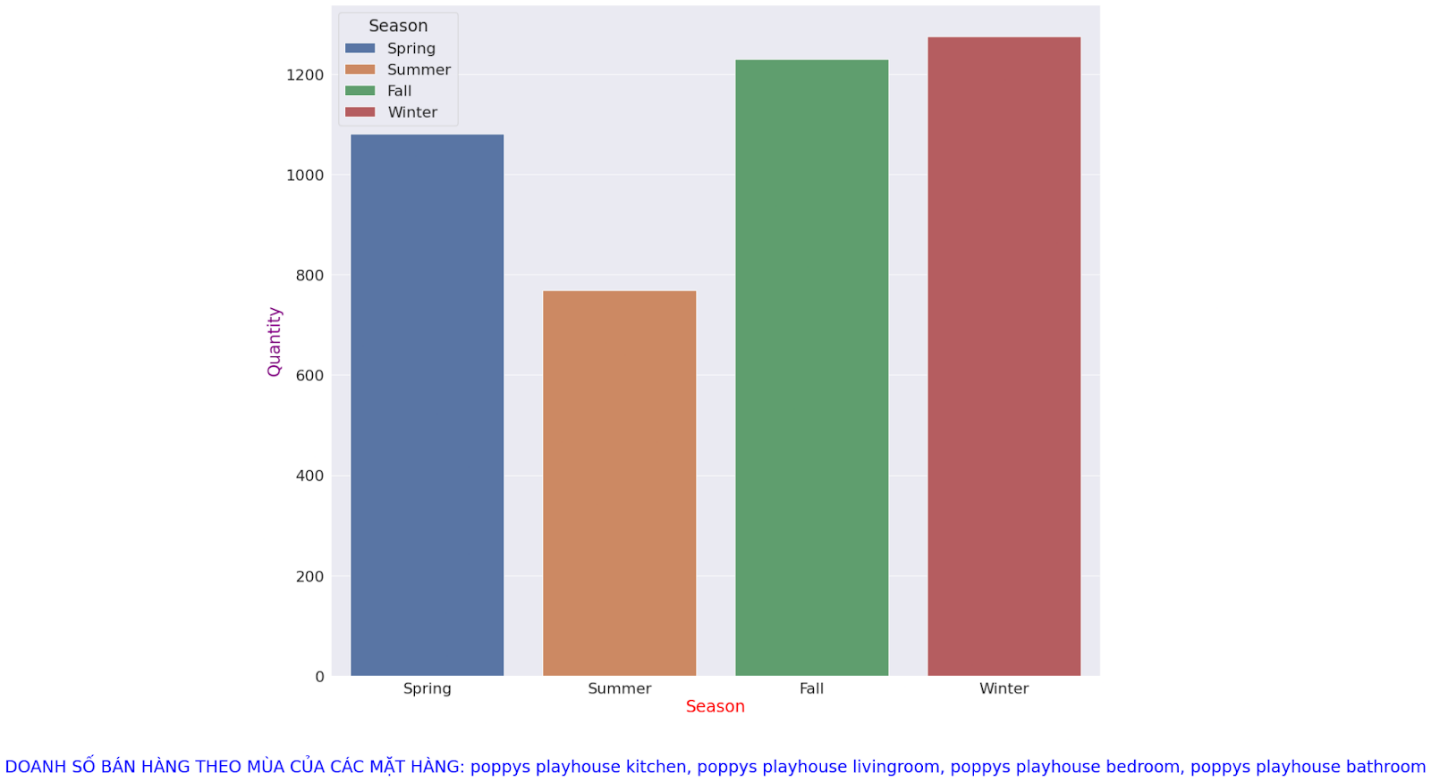
Bước 10**:** Thống kê số lượng mặt hàng “poppys playhouse kitchen”, “poppys playhouse livingroom”, “poppys playhouse bedroom” và “poppys playhouse bathroom” theo từng mùa. Ta không cần lọc theo InvoiceNo vì nếu các item được bán nhiều và đi chung với nhau ở mùa nào thì mùa đó sẽ có số lượng hàng bán ra nhiều hơn với các mặt hàng này nên việc đó là không cần thiết.

****

****

**Bước 11:** Vẽ biểu đồ số lượng hàng “**poppys playhouse kitchen”,** “**poppys playhouse livingroom**”, “**poppys playhouse bedroom**” và “**poppys playhouse bathroom**” đã được bán theo mùa.

****

****

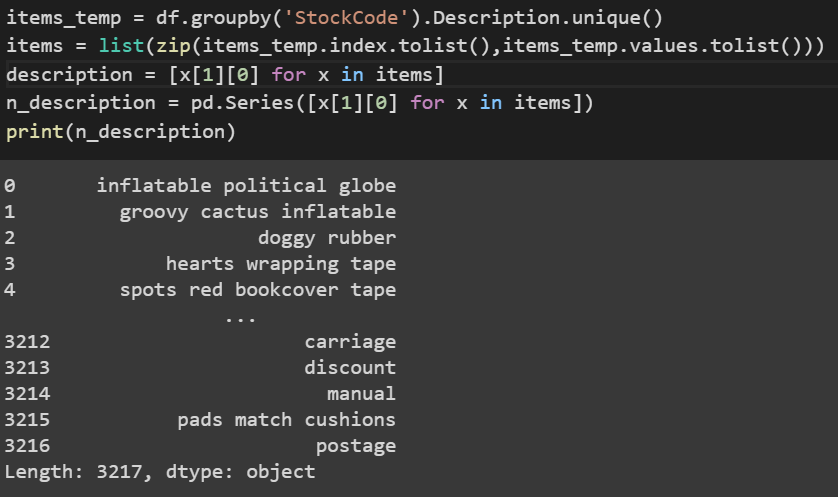
*Nhận xét*: Qua biểu đồ barchart trên, ta thấy được các mặt hàng “poppys playhouse kitchen”, “poppys playhouse bedroom”, “poppys playhouse livingroom” và “poppys playhouse bathroom” được bán nhiều nhất ở 2 mùa Thu (Fall) và Đông (Winter). Chính vì vậy, ta sẽ cần có các chính sách giảm giá, khuyến mãi bán theo combo các mặt hàng này để thúc đẩy lượng hàng được bán ra; gia tăng sản xuất các mặt hàng này, gia tăng quảng cáo để kích sức bán lên cao nhất có thể.

# Chương V: Machine Learning

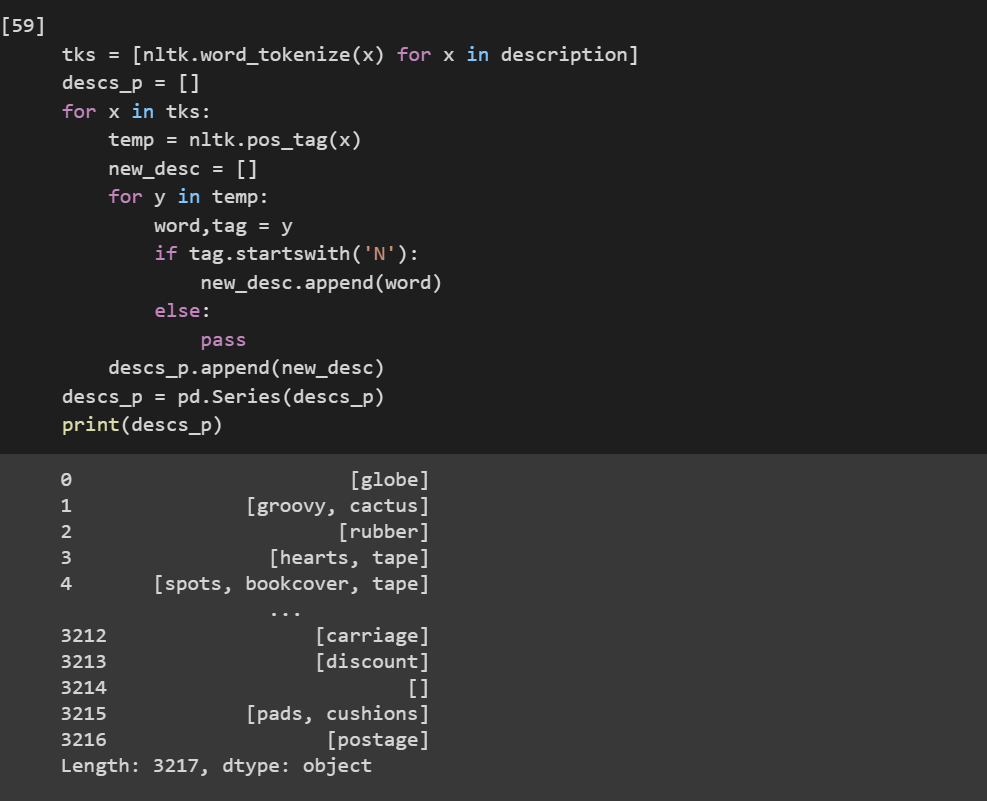
## Cluster Category by Product Name (Description) - K-means:

* 1. **Mục đích:**

Bộ dữ liệu có 3,217 mã sản phẩm (stock code) khác nhau tuy nhiên nhiều sản phẩm lại có tên (Description) tương tự nhau và có thể gộp chung lại thành một nhóm sản phẩm. Việc gom những sản phẩm tương tự nhau lại thành một nhóm giúp chúng ta có được cái nhìn tổng quát hơn về một mặt hàng được bán chạy thay vì từng sản phẩm riêng biệt. Bên cạnh đó, trực quan hóa dữ liệu cho 3,217 đầu sản phẩm cũng không hiệu quả nếu so với những nhóm sản phẩm chung hơn.



* 1. **Xử lý dữ liệu Description:**
* Dù đã được xử lý tốt ở bước tiền xử lý, chúng ta vẫn cần phải căn chỉnh lại dữ liệu **Description** cho phù hợp với với việc phân cụm.
* Đầu tiên, chúng ta cần phải tokenize từng vùng dữ liệu trong **Description** (tên của từng sản phẩm) để dễ dàng loại bỏ những từ sẽ gây nhiễu đến việc phân cụm. Tích hợp trong bước tokenize, chúng ta cũng sẽ loại bỏ những động từ, tính từ, trạng từ, …, chỉ giữ lại danh từ, phù hợp cho việc phân cụm.

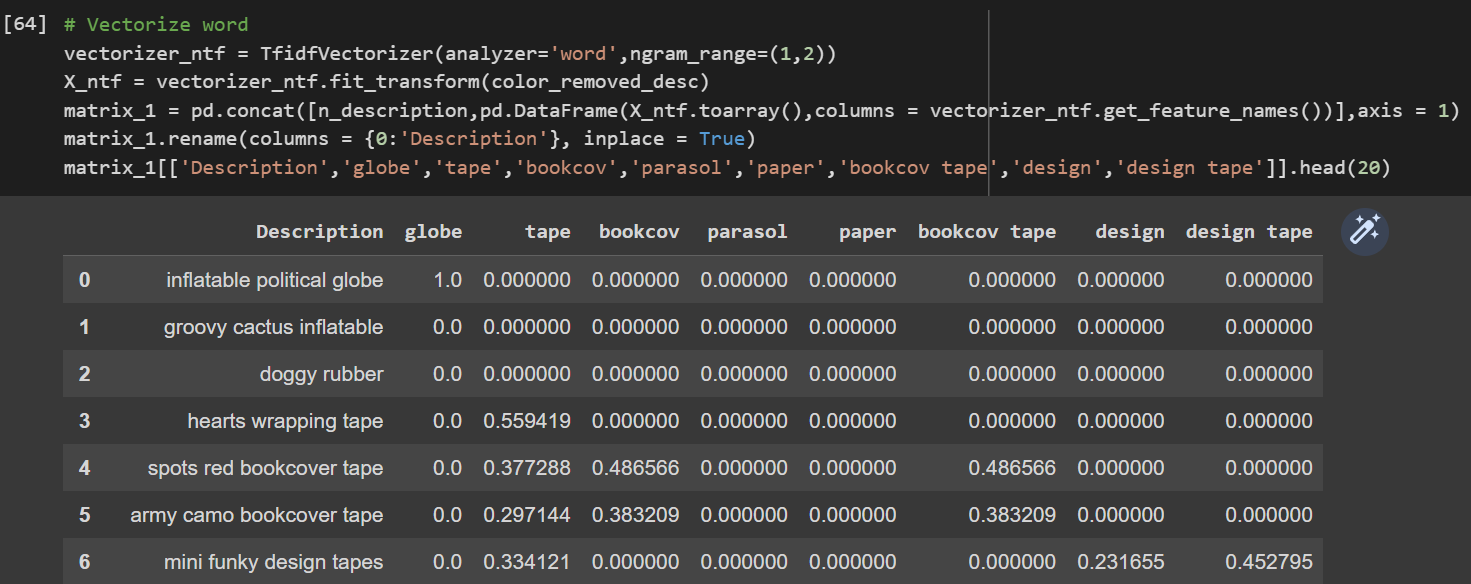


* Tiếp theo, Ta đưa các từ trên về dạng từ cán (ví dụ programming, programmer đưa về dạng từ cán là program) đồng thời loại bỏ các danh từ chỉ màu sắc vì có khả năng sẽ tồn tại và gây nhiễu trong quá trình chạy phân cụm.



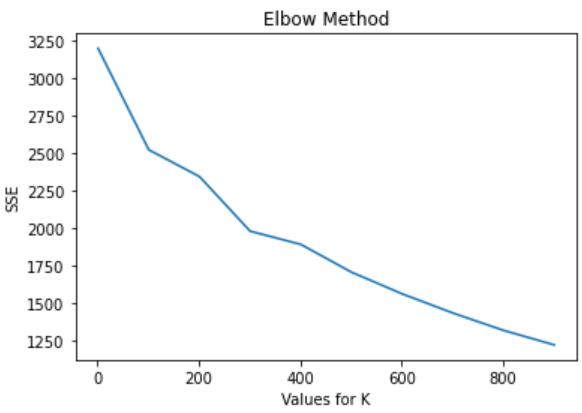
* 1. **Vector hóa**

Sau khi làm cho những dòng dữ liệu trở nên sạch sẽ hơn, ta tiến hành chuyển các dữ liệu dạng chuỗi của ta thành dữ liệu dạng số. Cụ thể hơn, trong bài này ta sẽ sử dụng phương pháp **TF - IDF (term frequency–inverse document frequency)** để gắn trọng số cho mỗi từ trong 1 tên sản phẩm. **TF - IDF** là một phương pháp đánh giá độ quan trọng của một từ hoặc một cụm từ trong tập văn bản khi văn bản đó nằm trong một tập hợp các văn bản.Phương pháp này được xác định dựa trên tần suất của từ trong văn bản và tần suất của văn bản chứa từ đó trong tập hợp các văn bản. Như ví dụ bên dưới ta có thể thấy, áp dụng phương pháp **Tf - IDF** để gắn trọng số cho mỗi từ trong tên sản phẩm, với sản phẩm có tên *“spots red bookcover tape”*  thì từ bookcover có trọng số ~ 0.486 và từ tape có trọng số ~ 0.377. Với giá trị trọng số càng cao, từ đó sẽ mang ý nghĩa cao hơn trong văn bản. Bên cạnh đó, trong bài này cũng có sử dụng n-gram level để thứ tự của các từ cũng được xem xét.

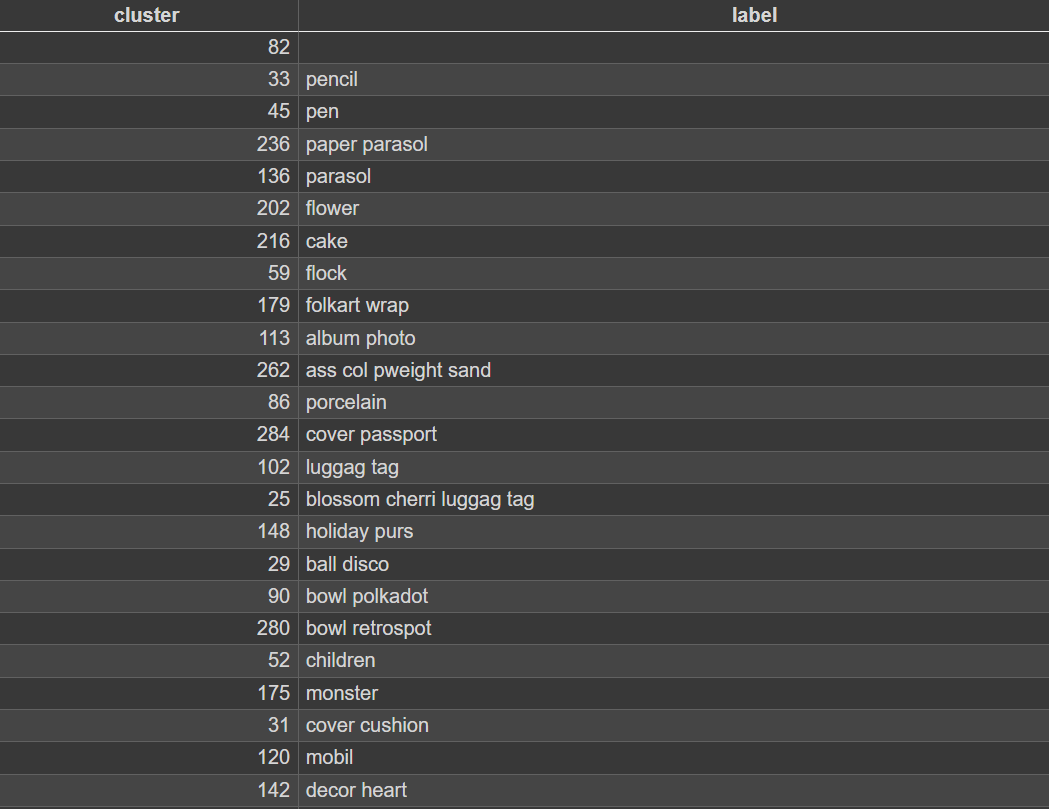


* 1. **Phân cụm với K-means:**

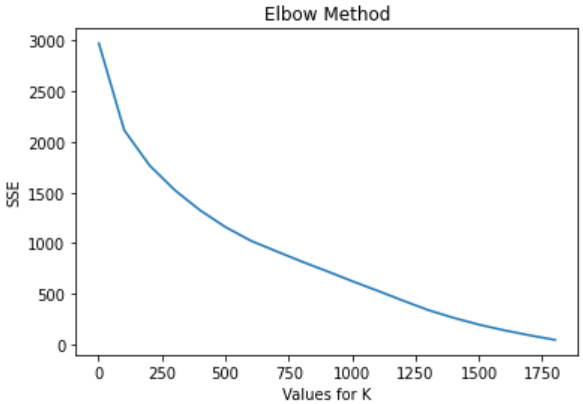
Sau khi đã có ma trận trọng số các từ, ta tiến hành phân cụm với k-means. Đầu tiên sử dụng phương pháp elbow để tìm được số cụm tối ưu nhất. Phương pháp elbow sử dụng độ tương quan nghịch giữa SSE (Sum of Square Error) và số cụm. Ta không muốn có quá nhiều cụm đồng thời cũng không muốn SSE quá cao nên ta chọn điểm elbow trong đồ thị, nơi có số cụm và điểm SSE thỏa điều kiện tốt nhất. Với bài này, số cụm tốt nhất ở khoảng K = 300.



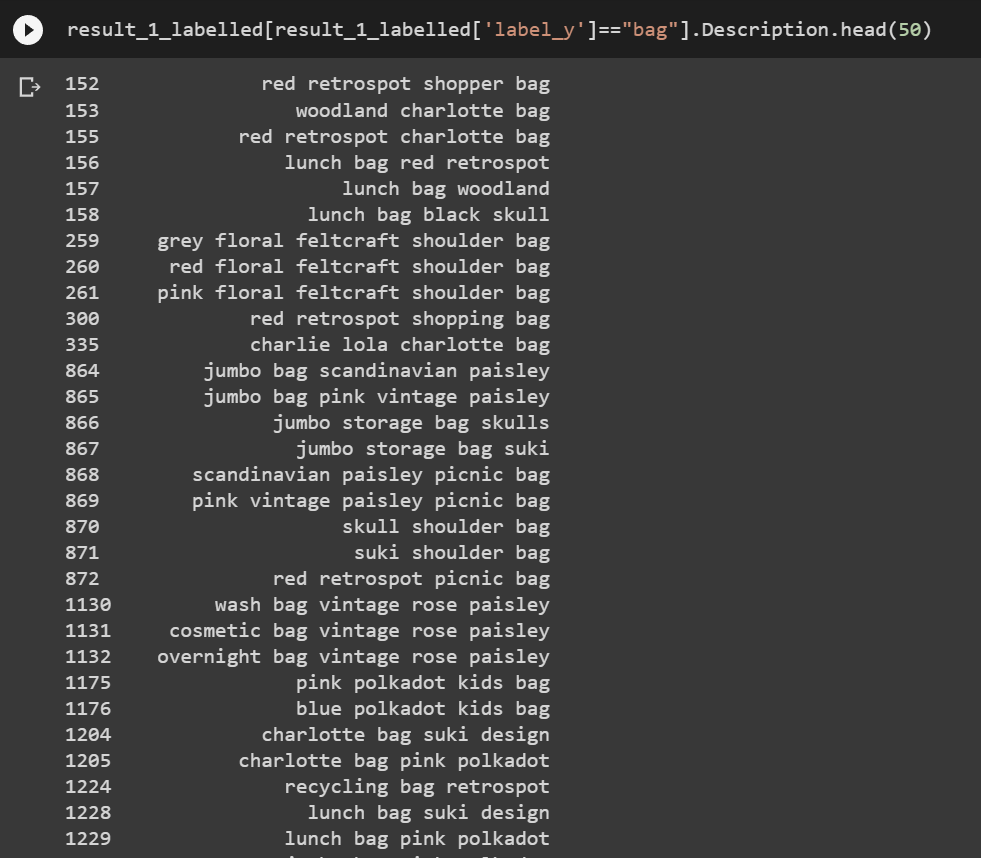
Sau khi được phân thành 300 cụm, ta tiến hành dán nhãn cho mỗi cụm để xem liệu các cụm được đặt tên hợp lý với các sản phẩm trong cụm hay không Sau khi kiểm tra các cụm được dán nhãn, ta dễ dàng nhìn thấy các nhãn không hợp lý như *Children* hay cùng một loại bowl thì được phân thành 2 cụm *bowl polkadot* và *bowl retrospot.* Bước này, ta cần thủ công loại bỏ những từ không phù để các cụm được nhóm lại hiệu quả hơn.



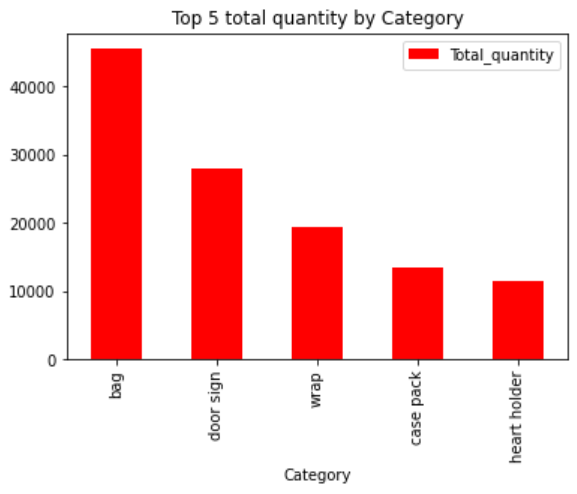
Sau khi loại bỏ các từ bất hợp lý, ta tiến hành vectorize lại phân cụm lại với bộ dữ liệu mới và thấy rằng điểm SSE chung đã giảm. Tại K = 300, điểm SSE ở bộ dữ liệu cũ rơi vào khoảng ~ 2,000 còn ở bộ dữ liệu đã được lọc từ không hợp lí chỉ còn khoảng ~ 1500.



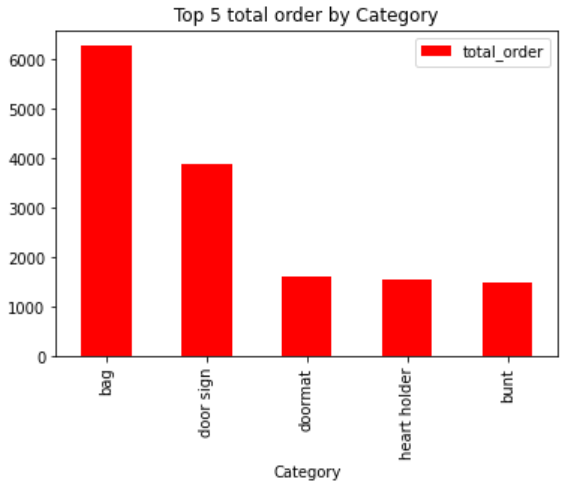
* Ta kiểm tra xem các sản phẩm được đưa vào trong 1 cụm có hợp lý hay không.Với cụm được dán nhãn lag *bag*, thì các sản phẩm đều là *bag*, một tín hiệu đáng mừng.



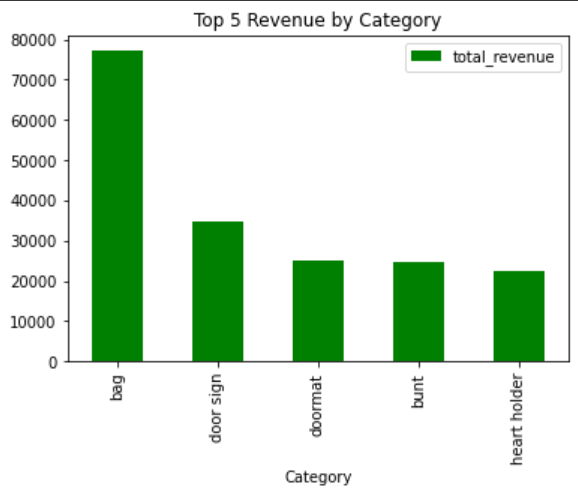
* Cuối cùng, sau khi phân những sản phẩm thành những nhóm, ta sử dụng những nhóm này cho mục đích phân tích dữ liệu bằng cách trực quan hóa.Ta có thể dễ dàng nhìn thấy *bag* đang là loại sản phẩm được bán nhiều nhất với số lượng đặt hàng ~ 45,000 (Một insight khác khi ta sử dụng category để phân tích vì sản phẩm bán nhiều nhất lại là *pack retrospot cake case* như đã đề cập ở trên không thuộc nhóm bag)



* Bên cạnh đó ta cũng có thể biết được loạt mặt hàng nào được đặt hàng nhiều nhất. Ta có thể thấy sự khác nhau ở top 3,4,5 giữa số lượng đặt và số lần đặt theo từng nhóm.

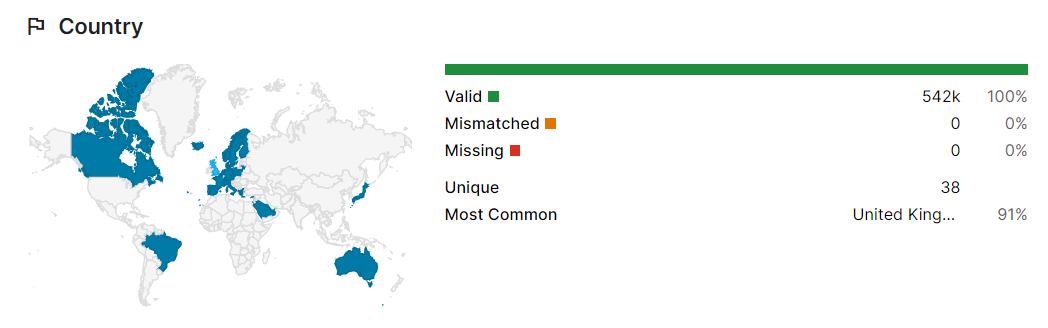


* Và cuối cùng ta cũng có thể thấy rằng *bag* là dòng sản phẩm bán chạy nhất và cũng là dòng sản phẩm mang lại doanh thu cao nhất cho doanh nghiệp với revenue ~ 80,000

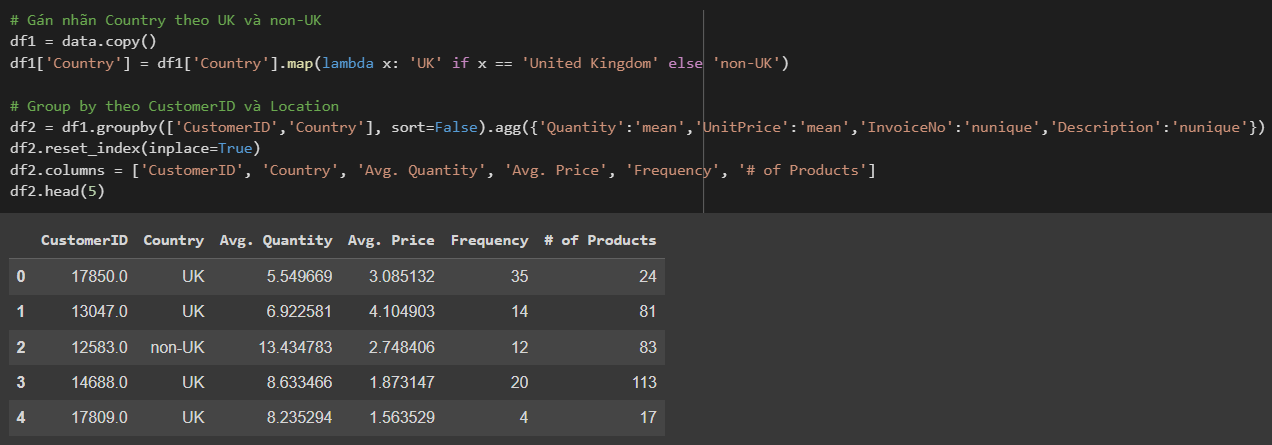


## Customer Segmentation by Country and Behavior - K-Prototypes

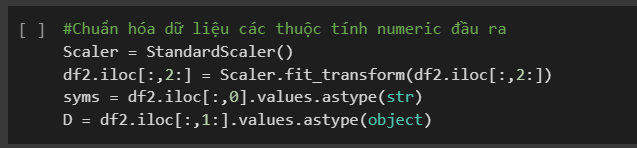
Tập dữ liệu có phần lớn khách hàng ở UK (91%) và phần nhỏ thuộc các quốc gia khác (9% còn lại). Vì vậy ta sẽ tiến hành phân cụm khách hàng khách hàng theo vị trí địa lý và hành vi mua hàng, xem xét những gì khách hàng mua, mức giá, tần suất và số lượng họ mua hàng để từ đó có thể đưa ra các chiến dịch tiếp thị phù hợp với từng nhóm khách hàng.



* Đầu tiên ta sẽ gán nhãn thuộc tính country thành UK (United Kingdom) và non-UK (các quốc gia khác):



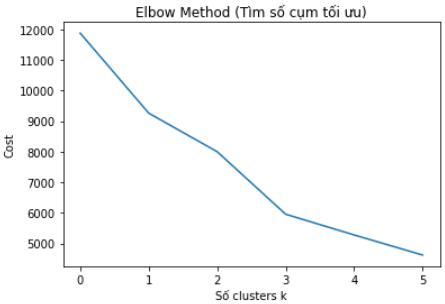
* Ta tiến hành chuẩn hóa dữ liệu cho các thuộc tính numeric bằng cách scale dữ liệu về một phân bố, trong đó giá trị trung bình của các quan sát bằng 0 và độ lệch chuẩn bằng 1, giúp giảm thiểu dữ liệu dư thừa, cải thiện quá trình phân cụm.



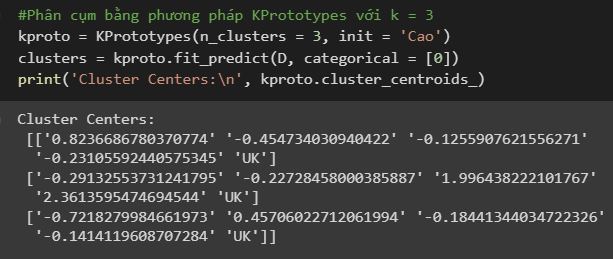
* Sử dụng phương pháp K-Prototypes - thuật toán cải tiến của K-Means để làm việc với các tập dữ liệu hỗn hợp (thuộc tính số và thuộc tính phân lớp)
* Ta tiến hành tìm số phân cụm tối ưu k bằng cách chạy vòng lặp với số cụm k từ 2 đến 8 để tìm ra số k phân cụm tối ưu bằng phương pháp Elbow.



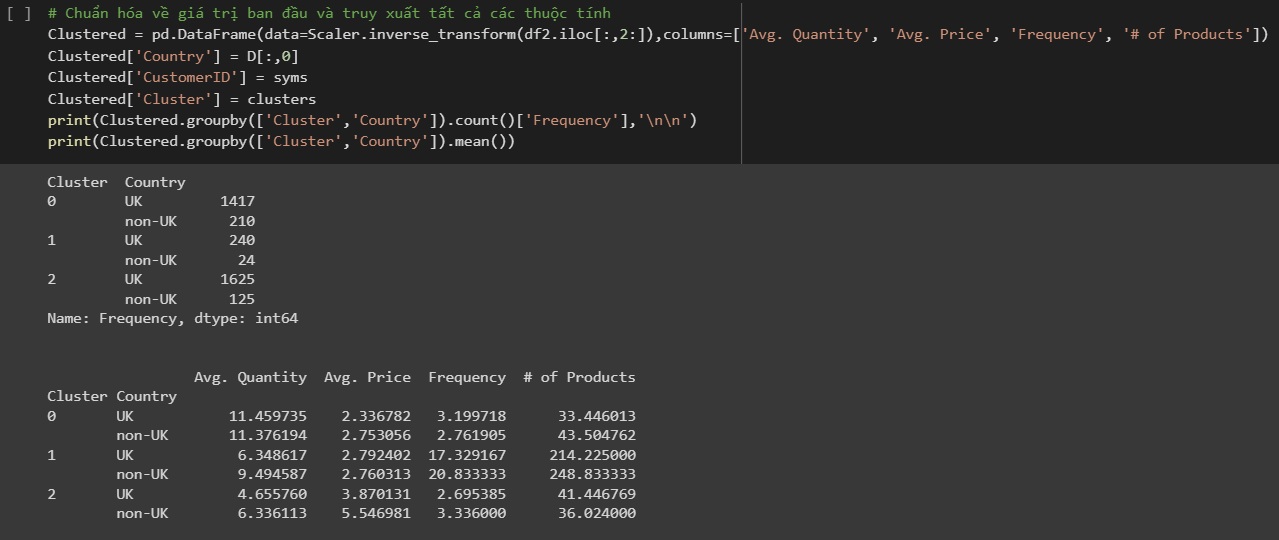




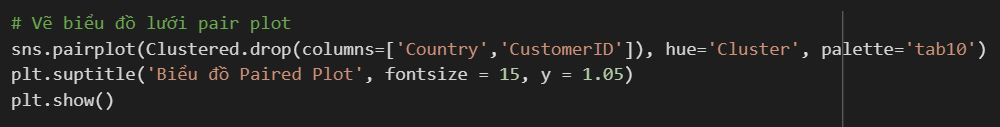
* Ta tiến hành phân cụm bằng phương pháp KPrototypes với k = 3

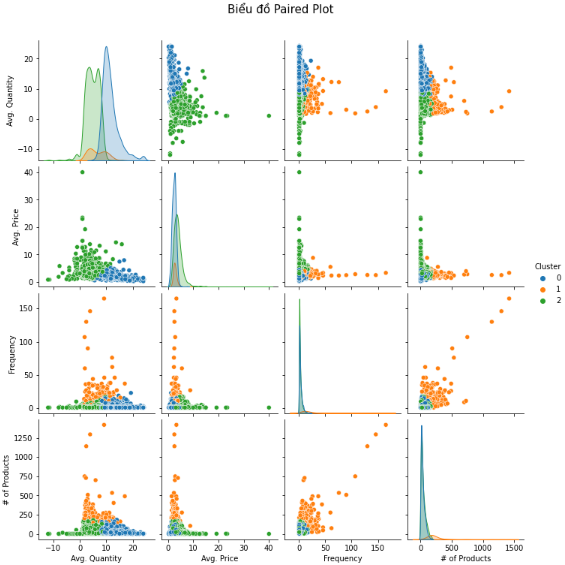


* Sau đó chuẩn hóa dữ liệu về giá trị ban đầu và truy xuất các thuộc tính



* Vẽ biểu đồ lưới pairplot thể hiện mối tương quan giữa các chiều dữ liệu (số lượng khách hàng, đơn giá, số lần mua hàng và mô tả sản phẩm theo vị trí địa lý) theo từng cặp với nhau





*Nhận xét*: Ta có thể thấy cách các cụm được phân phối theo từng biến trong biểu đồ phân tán.

* Trong cụm đầu tiên, khách hàng đã mua sản phẩm với số lượng lớn và mức giá thấp, các sản phẩm không đa dạng, tần suất khách hàng trở lại cũng khá thấp. Khách hàng chủ yếu nằm ngoài UK
* Cụm thứ hai đại diện cho những khách hàng mua sản phẩm với số lượng nhỏ, mức giá thấp nhưng đa dạng nhiều loại sản phẩm và họ có xu hướng thường xuyên quay trở lại mua hàng. Phần lớn khách hàng (cả ở UK và không ở UK) đều thuộc cụm này.
* Cụm thứ ba bao gồm những khách hàng đã mua những sản phẩm với số lượng thấp, mức giá cao và thỉnh thoảng mua hàng. Vì vậy những khách hàng mua hàng 1 lần có nhiều khả năng thuộc phân khúc này. Đồng thời phần lớn khách hàng (cả ở UK và không ở UK) đều thuộc cụm này

# CHƯƠNG VI: KẾT LUẬN

Với đề tài “*Phân tích dữ liệu kinh doanh*”, nhóm đã sử dụng bộ *Phân tích dữ liệu kinh doanh của cửa hàng bán lẻ trực tuyến* để tiến hành khám phá dữ liệu (EDA) và nhận ra bộ dữ liệu này có các biến tương quan sâu sắc với nhau như : “Quantiy”, “TotalSale” và “ Uniprice” ,sau đó đến bước tiền xử lí : Loại bỏ và thêm bớt những cột cần thiết, tách các outliers ra để giảm thiểu tối đa sự sai số. Kết quả sau khi phân tích nhóm đã xác định được xu hướng mua hàng (Timeseries) của khách hàng trong một khoảng thời gian, kết quả này giúp ích cho những ai muốn dự đoán xu hướng hành vi người tiêu dùng trong thời gian tới .Với phần FP-Growth, chúng ta thấy được các mặt hàng như “poppys playhouse kitchen”, “poppys playhouse bedroom”, “poppys playhouse livingroom” và “poppys playhouse bathroom” được bán nhiều nhất ở 2 mùa Thu (Fall) và Đông (Winter). Chính vì thế, cần đề xuất các chính sách giảm giá, khuyến mãi bán theo combo các mặt hàng này để thúc đẩy lượng hàng được bán ra; gia tăng sản xuất các mặt hàng này, gia tăng quảng cáo để kích sức bán lên cao nhất, hay cũng có thể sử dụng chính sách “số lượng hơn giá cả” tương tự như những công ty Trung Quốc. Việc đi gom lại những sản phẩm tương tự nhau lại thành một nhóm giúp chúng ta có được cái nhìn tổng quát hơn về một mặt hàng được bán chạy thay vì từng sản phẩm riêng biệt (Clustering).

# Tài Liệu Tham Khảo

[1] Empowering Business with Effective Insights, *“Online Retails Sale Dataset”,* 2022. Địa chỉ: <https://www.kaggle.com/datasets/rohitmahulkar/online-retails-sale-dataset>

[2] Big Data Uni, “Tìm hiểu về Time Series Phần 1”, 2021.

Địa chỉ: <https://bigdatauni.com/tin-tuc/tim-hieu-ve-time-series-phan-tich-chuoi-thoi-gian-p-1.html>

[3] Kaggle, “*Beginner Friendly EDA + FP-Growth Algorithm”,* 2022.

Địa chỉ: <https://www.kaggle.com/code/ahmedhammad01/beginner-friendly-eda-fp-growth-algorithm/notebook>

[4] Mlxtend, “ Apriori”.

Địa chỉ: [https://rasbt.github.io/mlxtend/api\_subpackages/mlxtend.frequent\_patterns/ - association\_rules](https://rasbt.github.io/mlxtend/api_subpackages/mlxtend.frequent_patterns/#association_rules)

[5] Bài giảng học phần Khai phá dữ liệu, *“Chương 4: Luật kết hợp”*, TS.Nguyễn An Tế, khoa Công nghệ thông tin kinh doanh, trường Đại học Kinh tế TPHCM, 2022.

[6] Lily Wu, “Clustering Product Names with Python - Part 1”, 2021.

Địa chỉ: <https://medium.com/m/global-identity?redirectUrl=https%3A%2F%2Ftowardsdatascience.com%2Fclustering-product-names-with-python-part-1-f9418f8705c8>

[7] Lily Wu, “Clustering Product Names with Python - Part 2”, 2021.

Địa chỉ: <https://medium.com/m/global-identity?redirectUrl=https%3A%2F%2Ftowardsdatascience.com%2Fclustering-product-names-with-python-part-1-f9418f8705c8>

[8] Dương Phạm, “*TF-IDF( term frequency – inverse document frequency)”,*2016.

Địa chỉ: <https://viblo.asia/p/tf-idf-term-frequency-inverse-document-frequency-JQVkVZgKkyd>

[9] Kaggle, “*Store Customers Clustering Analysis*”, 2021.

Địa chỉ: <https://www.kaggle.com/code/miguelfzzz/store-customers-clustering-analysis>

[10] Github, “Data analysis online retail transactions”, 2019.

Địa chỉ: <https://github.com/amir-hojjati/Data-Analysis-Online-Retail-Transactions>

# Bảng phân công:

|  |  |  |
| --- | --- | --- |
| Thành viên | Phân công | Đánh giá |
| Nguyễn Phúc Hải | FP-Growth, Tiền xử lý dữ liệu, Báo cáo phần Word. | 20% |
| Nguyễn Văn Hoàng Dũng | Kmeans, Timeseries, Tiền xử lý dữ liệu, Báo cáo phần Word. | 20% |
| Lê Đức Dũng | Khám phá dữ liệu (EDA), Timeseries, Báo cáo phần Word | 20% |
| Võ Ngọc Dung | Tiền xử lý dữ liệu, Báo cáo phần Word. | 20% |
| Nguyễn Quỳnh Khánh Hà | K-prototypes, Báo cáo phần Word . | 20% |