BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

BÀI TẬP LỚN

PHÂN TÍCH DỮ LIỆU ……

NGÀNH: KHOA HỌC MÁY TÍNH

CHUYÊN NGÀNH: TRÍ TUỆ NHÂN TẠO VÀ KHOA HỌC DỮ LIỆU

SINH VIÊN: **NGUYỄN VĂN XXX**

MÃ LỚP: **12421TN**

NGƯỜI HƯỚNG DẪN: **TS. HOÀNG QUỐC VIỆT**

HƯNG YÊN – 2024

NHẬN XÉT

Nhận xét của giáo viên hướng dẫn

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

GIÁO VIÊN HƯỚNG DẪN

Hoàng Quốc Việt

LỜI CAM ĐOAN

Em xin cam đoan bài tập lớn môn lập trình Python nâng cao có tên là "Phân tích dữ liệu về bộ dữ liệu thời trang Anh-Mỹ" là sản phẩm của bản thân em.

Những phần sử dụng tài liệu tham khảo trong bài tập lớn đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong bài tập lớn hoàn toàn là kết quả do bản thân em thực hiện.

Nếu vi phạm lời cam đoan này, em xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

Hưng Yên, ngày ... tháng 06 năm 2024

Sinh viên

Nguyễn Văn X

LỜI CẢM ƠN

Để có thể hoàn thành bài tập lớn này, lời đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Khoa học máy tính, Khoa Công nghệ thông tin – Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện bài tập lớn môn học này.

Đặc biệt em xin chân thành cảm ơn thầy Hoàng Quốc Việt đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện bài tập lớn vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được bài tập lớn này.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hy vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy cô về những kết quả triển khai trong bài tập lớn.

Em xin trân trọng cảm ơn!

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN

1.1 Bài toán

Hiện nay, Thị trường thời trang Anh-Mỹ là một trong những thị trường thời trang lớn nhất và có sự đa dạng về phong cách và xu hướng. Bằng cách phân tích dữ liệu thời trang Anh-Mỹ, chúng ta có thể thu thập thông tin về một phạm vi rộng hơn và cung cấp cái nhìn tổng quan về xu hướng mua sắm và ưa chuộng của người tiêu dùng trong lĩnh vực thời trang, bao gồm nhiều loại sản phẩm từ quần áo, giày dép, phụ kiện đến trang phục và phụ kiện thời trang cho nam, nữ và trẻ em. Dữ liệu thời trang Anh-Mỹ thường được thu thập từ các nguồn uy tín và có tính tin cậy cao. Ngoài ra, dữ liệu thời trang Anh-Mỹ thường được cập nhật thường xuyên, giúp đảm bảo tính sẵn có của dữ liệu và đáng tin cậy trong quá trình phân tích

Ngoài ra thị trường thời trang Anh-Mỹ có sự ảnh hưởng toàn cầu đến các thị trường khác trên thế giới. Việc phân tích dữ liệu thời trang Anh-Mỹ giúp hiểu rõ hơn về ảnh hưởng của xu hướng và thị trường thời trang Anh-Mỹ đến các thị trường quốc tế khác, từ đó có thể áp dụng những kết quả phân tích để tối ưu hóa hoạt động kinh doanh và phát triển thị trường thời trang.

Chúng ta đã có dữ liệu và bây giờ chúng ta sẽ phải khai thác dữ liệu đó giúp người dùng biết rõ hơn về thông tin của các sản phẩm thời trang của Anh Mỹ ,…Giúp người dùng dễ dàng lựa chọn được trang phục và phụ kiện thời trang phù hợp nhất cho bản thân.

1.2 Trình bày dữ liệu bài toán

Dữ liệu được lấy từ trang web này:

Fashion Dataset UK-US | Kaggle

	Tên SP	Giá	Nhãn hiệu	Loại SP	Mô tả	Tổng đánh giá	Kiểu phong cách	Tổng size	size có sẵn	Màu sắc	Lịch sử mua hàng	Tuổi	Mùa	Đánh giá khách hàng	Phản hồi
0	T5D3	97.509966	Ralph Lauren	Footwear	Bad	492	Streetwear	M, L, XL	XL	Green	Medium	24	Fall/Winter	Mixed	Other
1	Y0V7	52.341277	Ted Baker	Tops	Not Good	57	Vintage	M, L, XL	XL	Black	Above Average	61	Winter	Negative	Other
2	N9Q4	15.430975	Jigsaw	Footwear	Very Bad	197	Streetwear	S, M, L	M	Blue	Average	27	Summer	Unknown	Neutral
3	V2T6	81.116542	Alexander McQueen	Outerwear	Not Good	473	Formal	S, M, L	L	Red	Very High	50	Fall/Winter	Neutral	Other
4	S7Y1	31.633686	Tommy Hilfiger	Bottoms	Very Good	55	Sporty	M, L, XL	S	Green	Above Average	23	Spring	Positive	Positive
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
599994	R4Z1	39.114987	Burberry	Swimwear	Very Bad	421	Minimalist	S, M, L	S	Blue	Above Average	39	Spring/Summer	Mixed	Unknown
599995	M1J9	34.856545	Tommy Hilfiger	Outerwear	Not Good	202	Vintage	S, M, L	XL	Green	Medium	54	Spring/Summer	Neutral	Mixed
599996	J9E1	18.324853	Burberry	Lingerie	Very Good	434	Edgy	M, L, XL	M	Black	Below Average	52	Fall	Neutral	Negative
599997	K6B6	41.904775	Ted Baker	Accessories	Worst	453	Bohemian	S, L, XL	L	Red	High	39	Summer	Neutral	Positive
599998	J5F9	56.454716	Ralph Lauren	Activewear	Good	403	Edgy	M, L, XL	XL	Red	Very Low	35	Summer	Unknown	Other

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 599999 entries, 0 to 599998
Data columns (total 15 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   Tên SP                599999 non-null  object
 1   Giá                   599999 non-null  float64
 2   Nhãn hiệu             599999 non-null  object
 3   Loại SP               599999 non-null  object
 4   Mô tả                 599999 non-null  object
 5   Tổng đánh giá         599999 non-null  int64
 6   Kiểu phong cách       599999 non-null  object
 7   Tổng size             599999 non-null  object
 8   size có sẵn           599999 non-null  object
 9   Màu sắc               599999 non-null  object
 10  Lịch sử mua hàng      599999 non-null  object
 11  Tuổi                  599999 non-null  int64
 12  Mùa                   599999 non-null  object
 13  Đánh giá khách hàng   599999 non-null  object
 14  Phản hồi              599999 non-null  object
dtypes: float64(1), int64(2), object(12)
memory usage: 68.7+ MB
None
```

-Dữ liệu bài toán gồm các feature sau:

+ Tên sản phẩm

+ Giá sản phẩm

+ Hãng sản phẩm

+ Loại sản phẩm

+ Mô tả

+ Tổng đánh giá

+ Kiểu phong cách

+ Tổng size

+ Size có sẵn

+ Màu sắc

+ Lịch sử mua hàng

+ Tuổi

+ Mùa

+ Đánh giá khách hàng

+ Phản hồi

- Dữ liệu bài toán là 1 file csv gồm 599999 rows × 15 columns

+ Có 15 feature và mỗi feature có 599999 dữ liệu đầu vào

-Sau khi mô tả dữ liệu ta có:

	Giá	Tổng đánh giá	Tuổi
count	599999.000000	599999.000000	599999.000000
mean	55.026482	249.933553	41.009558
std	25.972593	144.334560	13.566723
min	10.000297	0.000000	18.000000
25%	32.529910	125.000000	29.000000
50%	55.051762	250.000000	41.000000
75%	77.493413	375.000000	53.000000
max	99.999648	499.000000	64.000000

+ Giá sản phẩm trong dữ liệu này Max là 99,9999 và Min là 10

+Tổng đánh giá sản phẩm Max là 499 và Min là 0

+ Độ tuổi lớn nhất mua sản phẩm là 64 và độ tuổi nhỏ nhất là 18

1.3 Tiền xử lý dữ liệu

1.4 Thống kê dữ liệu

a) Thống kê loại sản phẩm bán chạy nhất

b) Thống kê số lượng sản phẩm Theo nhãn hiệu (articleType)

c) Thống kê số lượng các sản phẩm mỗi mùa

d) Thống kê độ tuổi mua nhiều sản phẩm nhất

e) Thống kê số lượng sản phẩm theo từng phong cách

f) Thống kê top 10 hãng được bán nhiều nhất

g) Thống kê số lượng sản phẩm theo đánh giá khách hàng

h) Thống kê số lượng sản phẩm theo trạng thái phản hồi

i) Thống kê 10 sản phẩm có giá cao nhất

j) Thống kê số lượng sản phẩm theo nhóm tuổi (trẻ từ 1 đến 30) và (già từ 31 đến 60)

1.5 Trực quan hoá dữ liệu

a) Biểu đồ thể hiện tỉ lệ số lượng sản phẩm mỗi loại
b) Hiển thị top 10 sản phẩm có giá bán cao nhất
c) Biểu đồ hiển thị top 10 hãng được bán nhiều nhất
d) Biểu đồ thể hiện số lượng sản phẩm theo nhóm tuổi (trẻ từ 1 đến 30) và (già từ 31 đến 60)
e) Biểu đồ thể hiện số lượng các phản hồi của khách hàng

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Pandas

2.2 Matplotlib

CHƯƠNG 3. GIẢI PHÁP

3.1. Mã nguồn tiền xử lý dữ liệu

```
print(df.info())
```

```
print(df.describe())
```

```
print(df.dtypes)
```

Đầu tiên em sử dụng hàm df.info() để kiểm tra các feature nào bị thiếu và có kiểu dữ liệu thế nào.Em hiển thị lại hàm df và dùng lại hàm info để kiểm tra xem dữ liệu. Và hình ảnh dưới là kết quả sau khi đã điền các dữ liệu:

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 599999 entries, 0 to 599998
Data columns (total 15 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   Tên SP               599999 non-null  object
 1   Giá                  599999 non-null  float64
 2   Hãng                 599999 non-null  object
 3   Loại SP              599999 non-null  object
 4   Mô tả                599999 non-null  object
 5   Tổng đánh giá        599999 non-null  int64
 6   Kiểu phong cách      599999 non-null  object
 7   Tổng size            599999 non-null  object
 8   size có sẵn          599999 non-null  object
 9   Màu sắc              599999 non-null  object
 10  Lịch sử mua hàng     599999 non-null  object
 11  Tuổi                 599999 non-null  int64
 12  Mùa                  599999 non-null  object
 13  Đánh giá khách hàng  599999 non-null  object
 14  Phản hồi             599999 non-null  object
dtypes: float64(1), int64(2), object(12)
memory usage: 68.7+ MB
```

3.2. Mã nguồn chức năng Thống kê dữ liệu

a, Thống kê loại sản phẩm bán chạy nhất

```
x=df[["Loại SP"]].assign(soluong=df['Tên SP']).groupby('Loại SP').count()
x.sort_values('soluong',ascending=False,ignore_index=True)
x=x.head(10)
x
```

- Đầu tiên em sử dụng hàm groupby(),hàm này cho phép chúng ta nhóm các dữ liệu giống nhau theo nhãn feature mà ta chọn trong dataframe ,vì ở đây là thống kê số lượng sản phẩm nên em sẽ nhóm các dữ liệu liên quan đến Tên loại sản phẩm vào với nhau;em

sẽ chọn dữ liệu "Loại SP" và "Tên SP" từ dataframe để nhóm với dữ liệu "Loại SP" giống nhau .

-Sử dụng hàm assign() phương thức gán các cột mới cho DataFrame, trả về một đối tượng mới (bản sao) với các cột mới được thêm vào các cột ban đầu. Các cột hiện có được gán lại sẽ bị ghi đè. Em sẽ gán cột mới tên "soluong" cho cột "Tên SP" trong dataframe để về sau chúng ta sẽ cột dữ liệu mới ứng dữ liệu như với các dữ liệu giống của "Loại SP"

-Sau đó em sử dụng hàm count() để đếm xem số lượng phim mà có cùng dữ liệu "Loai SP"giống nhau.

-Sau khi đã có được dataframe dữ liệu tổng hợp lại rồi em sử dụng hàm sort_values để sắp xếp dữ liệu lại theo thứ tự giảm dần và ta sẽ có dược kết quả như sau:

Loại SP	soluong
Accessories	60028
Activewear	60053
Bottoms	59541
Dresses	60119
Footwear	59643
Jewelry	60110
Lingerie	59973
Outerwear	59907
Swimwear	60346
Tops	60279

b, Thống kê số lượng sản phẩm Theo nhãn hiệu (articleType)

```
x2=df[["Nhãn hiệu"]].assign(soluong=df['Tên SP']).groupby("Nhãn hiệu").count()
x2.sort_values('soluong',ascending=False,ignore_index=True)
x2
```

- Đầu tiên em sử dụng hàm groupby(), hàm này cho phép chúng ta nhóm các dữ liệu giống nhau theo nhãn feature mà ta chọn trong dataframe ,vì ở đây là thống kê số lượng sản phẩm nên em sẽ nhóm các dữ liệu liên quan đến nhãn hiệu sản phẩm vào với nhau;em sẽ chọn dữ liệu "Nhãn hiệu" và "Tên SP" từ dataframe để nhóm với dữ liệu "Nhãn hiệu" giống nhau .

- Sử dụng hàm assign() phương thức gán các cột mới cho DataFrame, trả về một đối tượng mới (bản sao) với các cột mới được thêm vào các cột ban đầu. Các cột hiện có được gán lại sẽ bị ghi đè. Em sẽ gán cột mới tên "soluong" cho cột "Tên SP" trong dataframe để về sau chúng ta sẽ cột dữ liệu mới ứng dữ liệu như với các dữ liệu giống của "Nhãn hiệu"
- Sau đó em sử dụng hàm count() để đếm xem số lượng phim mà có cùng dữ liệu "Nhãn hiệu"giống nhau.
- Sau khi đã có được dataframe dữ liệu tổng hợp lại rồi em sử dụng hàm sort_values để sắp xếp dữ liệu lại theo thứ tự giảm dần và ta sẽ có dược kết quả như sau:

	soluong
Nhãn hiệu	
Alexander McQueen	75126
Burberry	74875
Calvin Klein	74703
Jigsaw	75555
Mulberry	74779
Ralph Lauren	75195
Ted Baker	74761
Tommy Hilfiger	75005

c, Thống kê số lượng các sản phẩm mỗi mùa

```
x=df[["Mùa"]].assign(soluongsanpham=df['Tên SP']).groupby('Mùa').count()
x.sort_values('soluongsanpham',ascending=False,ignore_index=True)
x
```

- Sử dụng groupby, hàm này cho phép chúng ta nhóm các dữ liệu giống nhau theo cột mà ta chọn trong Dataframe, vì đây là thống kê số lượng sản phẩm liên quan đến mùa nên e chọn 2 cột mùa và cột tên sản phẩm để nhóm với dữ liệu mùa giông nhau.
- Sử dụng hàm assign phương thức gán cột mới cho Dataframe. Trả về một đối tượng mới với các cột mới được thêm vào các cột ban đầu. Các cột hiện có được gán sẽ bị ghi đè.
- Sau khi đã có Datafame dữ liệu tổng hợp em sử dụng hàm sort_values để sắp xếp dữ liệu theo thứ tự giảm dần(ascending=False), ignore_index=True để giúp cho các thú tự của index không bị thay đổi vị trí sau khi sắp xếp và được kết quả sau:

Mùa	soluongsanpham
Fall	100446
Fall/Winter	99598
Spring	100515
Spring/Summer	99771
Summer	99754
Winter	99915

d, Thống kê độ tuổi mua nhiều sản phẩm nhất

```
df1=df[["Tuổi"]].assign(soluong=df['Tên SP']).groupby('Tuổi').count()
max_row = df1[df1["soluong"] == df1["soluong"].max()]
max_row
```

- Sử dụng groupby , hàm này cho chúng ta gom nhóm các thuộc tính giống nhau theo cột mà ta chọn trong dataframe. Vì đây là thống kê sản phẩm theo độ tuổi người mua nên ta chọn 2 cột là "Tuổi" và Số lượng để gom nhóm với dữ liệu tuổi giống nhau.
- Sử dụng hàm assign phương thức gán cột mới cho dataframe. Trả về một đối tượng gồm các cột mới cùng với các cột cũ, các cột cũ hiện có bị gán sẽ được ghi đè.
- Vì đây là thống kê ra độ tuổi mua nhiều sản phẩm nhất nên cho cột soluong bằng soluong lớn nhất ở trong dataframe bằng cách sử dùng hàm max .Từ có ta có được kết quả sau:

Tuổi	soluong
38	12966

e, Thống kê số lượng sản phẩm theo từng phong cách

```
x=df[["Kiểu phong cách"]].assign(soluongsanpham=df['Tên SP']).groupby('Kiểu phong cách').count()
x = x.sort_values('soluongsanpham', ascending=False).reset_index()
x
```

	Kiểu phong cách	soluongsanpham
0	Edgy	60360
1	Bohemian	60081
2	Sporty	60036
3	Minimalist	59982
4	Formal	59979
5	Vintage	59978
6	Streetwear	59934
7	Preppy	59896
8	Casual	59889
9	Glamorous	59864

f, Thống kê top 10 hãng được bán nhiều nhất

```
product_count_by_brand = df['Nhãn hiệu'].value_counts().head(10)
product_count_by_brand
```

- Sử dụng groupby để gom nhóm các thuộc giống nhau theo cột . Vì đây là thống kê sản phẩm theo màu sắc nên e chọn cột “Sản phẩm” và “nhãn hiệu”.để gom nhóm với dữ liệu nhãn hiệu

```
Jigsaw               75555
Ralph Lauren         75195
Alexander McQueen    75126
Tommy Hilfiger       75005
Burberry             74875
Mulberry             74779
Ted Baker            74761
Calvin Klein         74703
Name: Nhãn hiệu, dtype: int64
```

g, Thống kê số lượng sản phẩm theo đánh giá khách hàng

```
product_count_by_review = df.groupby('Đánh giá khách hàng')['Tên SP'].count()
product_count_by_review
```

- Sử dụng groupby để gom nhóm các thuộc giống nhau theo cột . Vì đây là thống kê sản phẩm theo màu sắc nên e chọn cột “Đánh giá khách hàng” và “Tên sản phẩm”.để gom nhóm với dữ liệu màu sắc giống nhau.

```
Đánh giá khách hàng
Mixed       120096
Negative    120060
Neutral     120247
Positive    120066
Unknown     119530
Name: Tên SP, dtype: int64
```

h, Thống kê số lượng sản phẩm theo trạng thái phản hồi

```
# Thống kê số lượng các phản hồi của khách hàng
feedback_count = df['Phản hồi'].value_counts()
feedback_count
```

```
Positive    100160
Mixed       100153
Negative    100135
Other        99937
Unknown      99886
Neutral      99728
Name: Phản hồi, dtype: int64
```

i, Thống kê top 10 sản phẩm có giá bán cao nhất

```
df1=df[["Tên SP","Giá"]]
top_10_expensive_products = df1.sort_values('Giá', ascending=False).head(10)
top_10_expensive_products
```

	Tên SP	Giá
466303	Q5L6	99.999648
258891	F4N9	99.999372
263711	K0Y6	99.999271
233422	L4X7	99.999074
332179	Q2R2	99.998508
571754	L2A3	99.998443
436382	Y9N1	99.998310
514720	N7K8	99.998302
580746	I3E3	99.997778
191151	N2N8	99.997663

j , Thống kê thể hiện số lượng sản phẩm theo nhóm tuổi (trẻ từ 1 đến 30) và (già từ 31 đến 60)

```
df['Nhóm tuổi'] = pd.cut(df['Tuổi'], bins=[5, 30, 60], labels=['Trẻ', 'Già'])
# Thống kê số lượng sản phẩm theo nhóm tuổi
product_count_by_age = df['Nhóm tuổi'].value_counts().sort_index()
product_count_by_age
```

```
Trẻ    165926
Già    382856
Name: Nhóm tuổi, dtype: int64
```

3.3. Mã nguồn chức năng Trực quan hóa dữ liệu

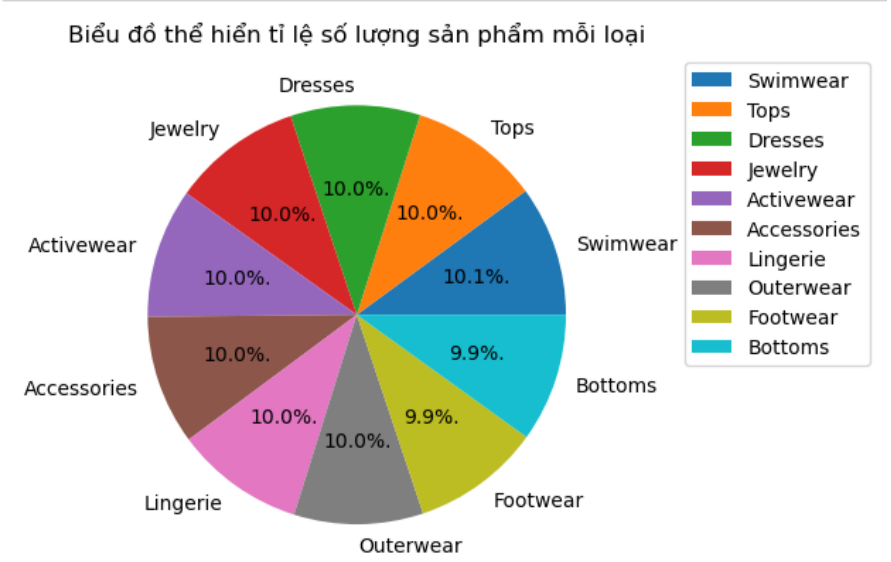
```
import matplotlib.pyplot as pl
```

Khai báo các thư viện cần thiết để trực quan hóa dữ liệu

a, Biểu đồ thể hiện tỉ lệ số lượng sản phẩm mỗi loại

```
Swimwear       60346
Tops           60279
Dresses        60119
Jewelry        60110
Activewear     60053
Accessories    60028
Lingerie       59973
Outerwear      59907
Footwear       59643
Bottoms        59541
Name: Loại SP, dtype: int64
```

```
pl.pie(df.values,labels=df.index,autopct="%1.1f%%.")
pl.legend(bbox_to_anchor=(1.5,1))
#pl.legend(bbox_to_anchor=(0,0,2,2,1),loc='upper right')
pl.title("Biểu đồ thể hiển tỉ lệ số lượng sản phẩm mỗi loại")
pass
```

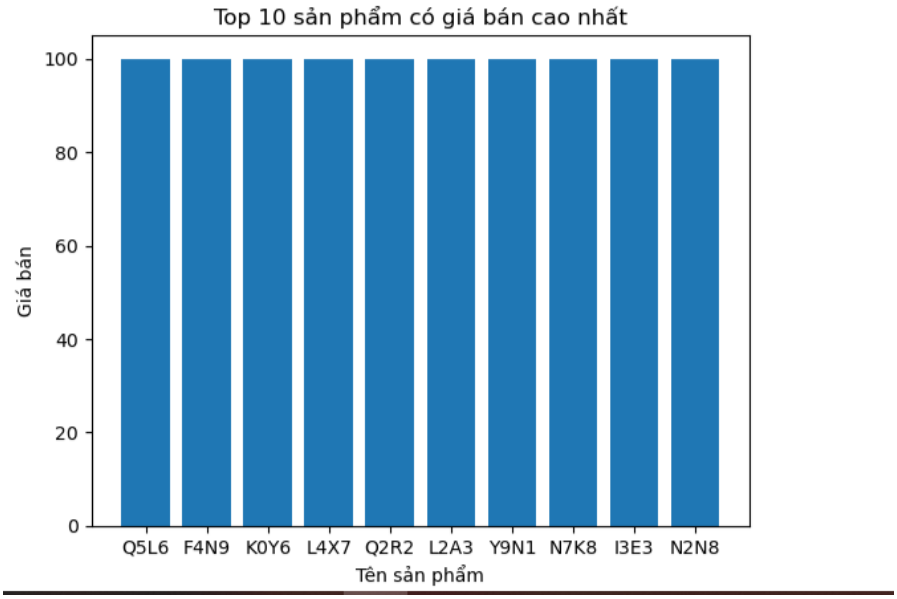


Đầu tiên chọn dataframe mà em vừa thống kê tỉ lệ số lượng sản phẩm mỗi loại ở trên. Sau đó chúng ta sử dụng pl.pie() để vẽ biểu đồ Pie. Truyền vào số lượng từ cột values của product_count_by_type và nhãn từ cột index. autopct='%1.1f%%' được sử dụng để hiển thị tỷ lệ phần trăm trên biểu đồ.

b, Hiển thị top 10 sản phẩm có giá bán cao nhất

	Tên SP	Giá
466303	Q5L6	99.999648
258891	F4N9	99.999372
263711	K0Y6	99.999271
233422	L4X7	99.999074
332179	Q2R2	99.998508
571754	L2A3	99.998443
436382	Y9N1	99.998310
514720	N7K8	99.998302
580746	I3E3	99.997778
191151	N2N8	99.997663

```
# Vẽ biểu đồ cột
pl.bar(top_10_expensive_products['Tên SP'], top_10_expensive_products['Giá'])
pl.xlabel('Tên sản phẩm')
pl.ylabel('Giá bán')
pl.title('Top 10 sản phẩm có giá bán cao nhất')
pl.show()
```



- pl.bar() được sử dụng để vẽ biểu đồ cột. Chúng ta truyền vào danh sách tên sản -phẩm từ cột 'Tên SP' và danh sách giá bán từ cột 'Giá' của DataFrame top_10_expensive_products.

pl.xlabel() được sử dụng để đặt nhãn cho trục x, trong trường hợp này là 'Tên sản phẩm'.

pl.ylabel() được sử dụng để đặt nhãn cho trục y, trong trường hợp này là 'Giá bán'.

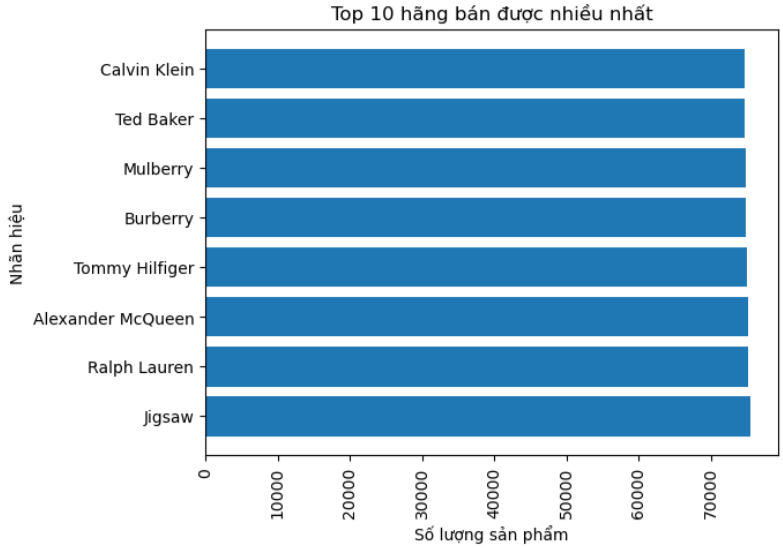
pl.title() được sử dụng để đặt tiêu đề cho biểu đồ, trong trường hợp này là 'Top 10 sản phẩm có giá bán cao nhất'.

Cuối cùng, plt.show() được sử dụng để hiển thị biểu đồ cột trên màn hình.

c, Biểu đồ hiển thị top 10 hãng được bán nhiều nhất

```
# Vẽ biểu đồ cột
plt.barh(product_count_by_brand.index, product_count_by_brand.values)
plt.xlabel('Số lượng sản phẩm')
plt.ylabel('Hãng')
plt.title('Top 10 hãng bán được nhiều nhất')
plt.show()
```

```
Jigsaw                75555
Ralph Lauren          75195
Alexander McQueen     75126
Tommy Hilfiger        75005
Burberry              74875
Mulberry              74779
Ted Baker             74761
Calvin Klein          74703
Name: Nhãn hiệu, dtype: int64
```



- pl.barh() được sử dụng để vẽ biểu đồ cột ngang. Chúng ta truyền vào danh sách tên hãng từ cột 'Hãng' và danh sách số lượng sản phẩm từ cột 'Số lượng sản phẩm' của DataFrame product_count_by_brand.

- pl.xlabel() được sử dụng để đặt nhãn cho trục x, trong trường hợp này là 'Số lượng sản phẩm'.

- pl.ylabel() được sử dụng để đặt nhãn cho trục y, trong trường hợp này là 'Hãng'.

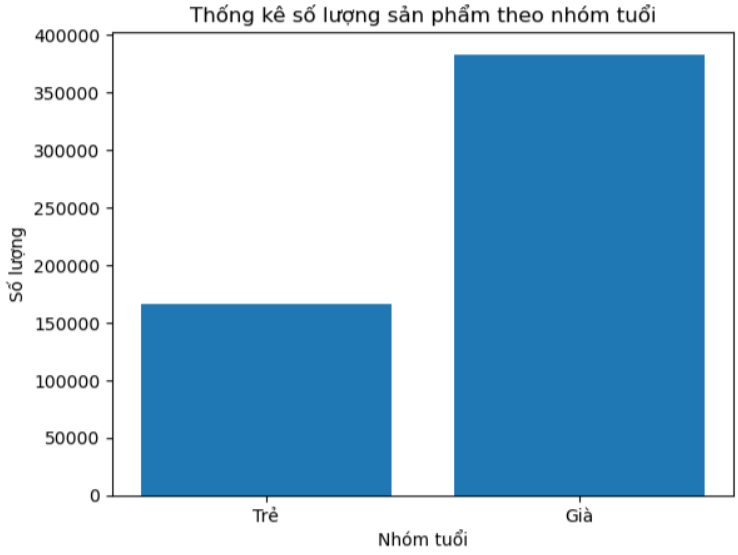
- pl.title() được sử dụng để đặt tiêu đề cho biểu đồ, trong trường hợp này là 'Top 10 hãng bán được nhiều nhất'.

- Cuối cùng, pl.show() được sử dụng để hiển thị biểu đồ cột ngang trên màn hình.

d, Biểu đồ thể hiện số lượng sản phẩm theo nhóm tuổi (trẻ từ 1 đến 30) và (già từ 31 đến 60)

```
Trẻ    165926
Già    382856
Name: Nhóm tuổi, dtype: int64
```

```
# Vẽ biểu đồ cột
plt.bar(product_count_by_age.index, product_count_by_age.values)
plt.xlabel('Nhóm tuổi')
plt.ylabel('Số lượng')
plt.title('Thống kê số lượng sản phẩm theo nhóm tuổi')
plt.show()
```



- pl.bar() được sử dụng để vẽ biểu đồ cột. Chúng ta truyền vào danh sách nhóm tuổi từ cột 'Nhóm tuổi' và danh sách số lượng sản phẩm từ cột 'Số lượng' của DataFrame product_count_by_age.

- pl.xlabel() được sử dụng để đặt nhãn cho trục x, trong trường hợp này là 'Nhóm tuổi'.

- pl.ylabel() được sử dụng để đặt nhãn cho trục y, trong trường hợp này là 'Số lượng'.

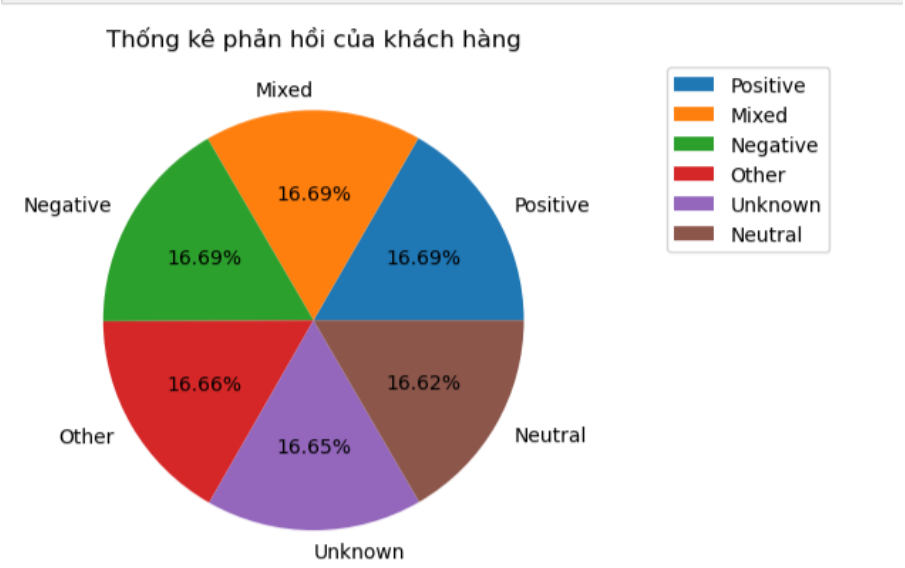
- pl.title() được sử dụng để đặt tiêu đề cho biểu đồ, trong trường hợp này là 'Thống kê số lượng sản phẩm theo nhóm tuổi'.

Cuối cùng, pl.show() được sử dụng để hiển thị biểu đồ cột trên màn hình.

e, Biểu đồ thể hiện số lượng các phản hồi của khách hàng

```
Positive    100160
Mixed       100153
Negative    100135
Other        99937
Unknown      99886
Neutral      99728
Name: Phản hồi, dtype: int64
```

```
# Vẽ biểu đồ pie
plt.pie(feedback_count.values, labels=feedback_count.index, autopct='%1.2f%%')
plt.title('Thống kê phản hồi của khách hàng')
pl.legend(bbox_to_anchor=(1.5,1))
plt.show()
```



- pl.pie() được sử dụng để vẽ biểu đồ Pie. Chúng ta truyền vào danh sách các giá trị từ cột 'Phản hồi' của DataFrame feedback_count và danh sách các nhãn tương ứng từ cột 'index' của feedback_count. Định dạng hiển thị tỉ lệ phần trăm sử dụng %1.2f%% để làm tròn đến 2 chữ số thập phân.
- pl.title() được sử dụng để đặt tiêu đề cho biểu đồ, trong trường hợp này là 'Thống kê phản hồi của khách hàng'.
- pl.legend() được sử dụng để đặt chú thích cho biểu đồ, ở đây là đặt chú thích vị trí bên phải của biểu đồ.

Cuối cùng, pl.show() được sử dụng để hiển thị biểu đồ Pie trên màn hình.

TÀI LIỆU THAM KHẢO

1. Dữ liệu

   Fashion Dataset UK-US | Kaggle

2. Tài liệu

   - Các tài liệu slide của thầy Nguyễn Văn Quyết
   - Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython 2nd Edition