

Bài tập Tiền xử lý dữ liệu 2 - Online Retail

Họ và tên: Lê Hoàng Vũ

Mã sinh viên: 23A4040156

Khai báo thư viện

Đọc dữ liệu

Có một số ký tự không thuộc bảng ASCII tiêu chuẩn. Truyền tham số encoding với giá trị unicode_escape Tham khảo một số nguồn tại <https://docs.python.org/3/library/codecs.html#standard-encodings> <https://stackoverflow.com/questions/22216076/unicodedecodeerror-utf8-codec-cant-decode-byte-0xa5-in-position-0-invalid-s>

Thông tin tổng quan bộ dữ liệu

```
-----
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
#   Column          Non-Null Count  Dtype
---  -
0   InvoiceNo        541909 non-null object
1   StockCode       541909 non-null object
2   Description     540455 non-null object
3   Quantity        541909 non-null int64
4   InvoiceDate     541909 non-null object
5   UnitPrice       541909 non-null float64
6   CustomerID      406829 non-null float64
7   Country         541909 non-null object
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
None
```

- Có tổng cộng 541909 bản ghi dữ liệu
- Thuộc tính Description, CustomerID có một vài giá trị Null. Nhưng Description là thuộc tính Optional, không quá quan trọng. Chủ yếu là CustomerID
- Thuộc tính CustomerID phải là dạng số nguyên
- InvoiceDate phải là kiểu thời gian
- Giả thuyết: InvoiceNo, StockCode cũng nên là dạng số nguyên, có một vài dữ liệu không thể ép sang số nguyên do có ký tự chữ? (Chưa làm ngay)

- UnitPrice, CustomerID, Quantity sẽ có một số giá trị ngoại lai

Đọc 10 hàng dữ liệu đầu tiên

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/01/2010 8:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	12/01/2010 8:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/01/2010 8:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/01/2010 8:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/01/2010 8:26	3.39	17850.0	United Kingdom
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/01/2010 8:26	7.65	17850.0	United Kingdom
6	536365	21730	GLASS STAR FROSTED T- LIGHT HOLDER	6	12/01/2010 8:26	4.25	17850.0	United Kingdom
7	536366	22633	HAND WARMER UNION JACK	6	12/01/2010 8:28	1.85	17850.0	United Kingdom
8	536366	22632	HAND WARMER RED POLKA DOT	6	12/01/2010 8:28	1.85	17850.0	United Kingdom
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	12/01/2010 8:34	1.69	13047.0	United Kingdom

Đọc 10 hàng dữ liệu cuối cùng

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
541899	581587	22726	ALARM CLOCK BAKELIKE GREEN	4	12/09/2011 12:50	3.75	12680.0	France
541900	581587	22730	ALARM CLOCK BAKELIKE IVORY	4	12/09/2011 12:50	3.75	12680.0	France
541901	581587	22367	CHILDRENS APRON SPACEBOY DESIGN	8	12/09/2011 12:50	1.95	12680.0	France
541902	581587	22629	SPACEBOY LUNCH BOX	12	12/09/2011 12:50	1.95	12680.0	France
541903	581587	23256	CHILDRENS CUTLERY SPACEBOY	4	12/09/2011 12:50	4.15	12680.0	France
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/09/2011 12:50	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/09/2011 12:50	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/09/2011 12:50	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/09/2011 12:50	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/09/2011 12:50	4.95	12680.0	France

Điền thay thế giá trị khuyết thiếu

Vì tệp dữ liệu khá lớn, nên chúng ta sẽ xóa một vài bản ghi thiếu CustomerID, ít bị ảnh hưởng tới các bài toán sau này. Chúng ta có thể thay thế các Description đang còn thiếu bằng giá trị như "Nothing" để có thể loại trừ các bản ghi chỉ thiếu CustomerID. Vì Description sau này có thể dựa vào Stock Code để trích xuất ra. Còn CustomerID thiếu thì khó có thể lấy lại được (hoặc tùy bài toán muốn xử lý)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
#   Column          Non-Null Count  Dtype
---  -
0   InvoiceNo        541909 non-null object
1   StockCode        541909 non-null object
2   Description      541909 non-null object
3   Quantity         541909 non-null int64
4   InvoiceDate       541909 non-null object
5   UnitPrice        541909 non-null float64
6   CustomerID       406829 non-null float64
7   Country          541909 non-null object
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
None
```

Xóa các dữ liệu khuyết thiếu

Xóa các bản ghi thiếu CustomerID và được tệp dữ liệu mới, đầy đủ các thuộc tính

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 406829 entries, 0 to 541908
Data columns (total 8 columns):
#   Column          Non-Null Count  Dtype
---  -
0   InvoiceNo        406829 non-null object
1   StockCode        406829 non-null object
2   Description      406829 non-null object
3   Quantity         406829 non-null int64
4   InvoiceDate       406829 non-null object
5   UnitPrice        406829 non-null float64
6   CustomerID       406829 non-null float64
7   Country          406829 non-null object
dtypes: float64(2), int64(1), object(5)
memory usage: 27.9+ MB
```

Chuyển đổi dạng dữ liệu

```

<class 'pandas.core.frame.DataFrame'>
Int64Index: 406829 entries, 0 to 541908
Data columns (total 8 columns):
#   Column          Non-Null Count  Dtype
---  -
0   InvoiceNo        406829 non-null object
1   StockCode        406829 non-null object
2   Description      406829 non-null object
3   Quantity         406829 non-null int64
4   InvoiceDate       406829 non-null datetime64[ns]
5   UnitPrice        406829 non-null float64
6   CustomerID       406829 non-null int64
7   Country          406829 non-null object
dtypes: datetime64[ns](1), float64(1), int64(2), object(4)
memory usage: 27.9+ MB
None

```

Kiểm tra các giá trị ngoại lai đối với kiểu dữ liệu số

Trong thống kê, nếu dữ liệu tương đồng nhau, sử dụng giá trị trung bình cho bạn kết quả phân tích chính xác nhất, nhưng nếu dữ liệu bị phân tán, có một vài giá trị mà chúng ta gọi là giá trị nhiễu, giá trị ngoại biên thì sử dụng số trung vị sẽ cho bạn kết quả chính xác nhất bởi số trung vị không phụ thuộc vào giá trị nhiễu.

<https://www.banhoituidap.com/p/3100/y-nghia-so-trung-vi-la-gi/>

Thông tin về thuộc tính Quantity

Thông tin tần suất các giá trị

1	73314
12	60033
2	58003
6	37688
4	32183

...

828	1
560	1
-408	1
512	1
-80995	1

Name: Quantity, Length: 436, dtype: int64

Giá trị trung bình: 12.06130339774205

Giá trị trung vị: 5.0

Giá trị lớn nhất: 80995

Giá trị nhỏ nhất: -80995

Thông tin về thuộc tính UnitPrice

Thông tin tần suất các giá trị

1.25	46555
1.65	37503
2.95	27211
0.85	26396
0.42	22032

...

3.56	1
4.37	1
6.89	1
0.98	1
224.69	1

Name: UnitPrice, Length: 620, dtype: int64

Giá trị trung bình: 3.4604710185298773

Giá trị trung vị: 1.95

Giá trị lớn nhất: 38970.0

Giá trị nhỏ nhất: 0.0

Thông tin về thuộc tính CustomerID

Thông tin tần suất các giá trị

17841	7983
14911	5903
14096	5128
12748	4642
14606	2782

...

```
15070      1
15753      1
17065      1
16881      1
16995      1
Name: CustomerID, Length: 4372, dtype: int64
```

```
Giá trị trung bình: 15287.690570239585
Giá trị trung vị: 15152.0
Giá trị lớn nhất: 18287
Giá trị nhỏ nhất: 12346
```

Như vậy, ta có thể thấy, ngoài thuộc tính **CustomerID** khá chính xác, thì hai thuộc tính **UnitPrice** và **Quantity** có các giá trị ngoại biên, sai lệch nhiều. Dựa vào giá trị trung bình, giá trị trung vị, giá trị lớn nhất và giá trị nhỏ nhất ta có thể thấy rõ điều đó.

Với **Quantity**, có một tập hợp giá trị đều là số âm. Chúng ta có thể giải quyết bằng cách thay thế nếu đây là tệp dữ liệu nhỏ, hoặc xóa những dữ liệu có giá trị Quantity là số âm đi với tệp dữ liệu lớn (do ít ảnh hưởng). Để kiểm tra xem có bao nhiêu bản ghi dữ liệu có giá trị Quantity âm, chúng ta sẽ sử dụng một vòng lặp.

```
Số giá trị Quantity âm: 8905
```

Như vậy, số giá trị Quantity âm không quá lớn so với tập dữ liệu hiện tại. Ta có thể xóa những bản ghi chứa những giá trị âm này đi. Nhưng nếu dùng phương thức `drop()` sẽ rất lâu đối với tập dữ liệu lớn, nên chúng ta sẽ làm theo cách lấy những bản ghi đạt điều kiện (>0). [Tham khảo: Theo StackOverflow](#)

```

<class 'pandas.core.frame.DataFrame'>
Int64Index: 397924 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column          Non-Null Count  Dtype
---  -
 0   InvoiceNo       397924 non-null object
 1   StockCode      397924 non-null object
 2   Description    397924 non-null object
 3   Quantity       397924 non-null int64
 4   InvoiceDate     397924 non-null datetime64[ns]
 5   UnitPrice      397924 non-null float64
 6   CustomerID     397924 non-null int64
 7   Country        397924 non-null object
dtypes: datetime64[ns](1), float64(1), int64(2), object(4)
memory usage: 27.3+ MB
Thông tin tần suất các giá trị
 1      73314
12     60033
 2     58003
 6     37688
 4     32183
...
4300      1
608        1
738        1
552        1
80995      1
Name: Quantity, Length: 302, dtype: int64

Giá trị trung bình: 13.021823262733587
Giá trị trung vị: 6.0
Giá trị lớn nhất: 80995
Giá trị nhỏ nhất: 1

```

Xuất tệp dữ liệu đã xử lý ra CSV

Kết luận và Giả thuyết

Kết luận: Với các giá trị lớn của UnitPrice và Quantity, chưa có cơ sở khẳng định rằng, các giá trị đó là chưa đúng. Vì có thể có sản phẩm có đơn giá rất cao và có doanh nghiệp mua một hàng hóa với số lượng rất lớn để bán lại. Ngoài ra, có một số sản phẩm có UnitPrice là 0.0, có thể sản phẩm đó là bán kèm hàng tặng, khuyến mãi.

Giả thuyết: Ngoài ra, còn một giả thuyết nữa là với StockCode và InvoiceNo còn một số giá trị không giống với số chung là do được thực hiện thủ công, có thể là mặt hàng hoặc hóa đơn đặc biệt.