test-code

May 6, 2025

[8]:
```
import pandas as pd
dataset_sample=pd.read_csv("shopee_sample_data.csv")
dataset_sample.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 42425 entries, 0 to 42424
Data columns (total 22 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   price_ori             42267 non-null  float64
 1   delivery              42363 non-null  object 
 2   item_category_detail  42363 non-null  object 
 3   specification         42363 non-null  object 
 4   title                 42363 non-null  object 
 5   w_date                42363 non-null  float64
 6   link_ori              42363 non-null  object 
 7   item_rating           38623 non-null  float64
 8   seller_name           42363 non-null  object 
 9   idElastic             42363 non-null  object 
 10  price_actual          42335 non-null  float64
 11  sitename              42363 non-null  object 
 12  idHash                42363 non-null  object 
 13  seller_details        41671 non-null  object 
 14  location              33756 non-null  object 
 15  total_rating          42363 non-null  object 
 16  id                    42363 non-null  float64
 17  pict_link             42363 non-null  object 
 18  total_sold            38623 non-null  object 
 19  favorite              39708 non-null  object 
 20  timestamp             42363 non-null  float64
 21  desc                  42363 non-null  object 
dtypes: float64(6), object(16)
memory usage: 7.1+ MB
```

[ ]:
```

```

[9]:
```
def isfloat(value):
    try:
        float(value)  # Try to convert the value to float
        return True   # If successful, it's a float
    except ValueError:  # If it fails, it's not a float
        return False
dataset_sample.iloc[dataset_sample[dataset_sample['item_category_detail'].
 ↪apply(lambda x: isfloat(x)) == True].index]
dataset_sample.dropna(thresh=10,inplace=True,axis=0)
dataset_sample['item_category_detail'] = dataset_sample['item_category_detail'].
 ↪apply(lambda x: x.lower())
dataset_sample['item_category_detail'] = dataset_sample['item_category_detail'].
 ↪apply(lambda x: x.split('|'))
dataset_sample['item_category_detail'] = dataset_sample['item_category_detail'].
 ↪apply(lambda x: [st.strip() for st in x])
```

[10]:
```
dataset_sample['Type of product']=dataset_sample["item_category_detail"].
 ↪apply(lambda x: x[1])
dataset_sample['Type of product_2']=dataset_sample["item_category_detail"].
 ↪apply(lambda x : x[2] if len(x)>3 else x[1])
dataset_sample.loc[:,['Type of product','Type of product_2']]
```

[10]:

	Type of product	Type of product_2
0	women's clothing	women's clothing
1	women's clothing	dresses
2	women's clothing	dresses
3	women's clothing	dresses
4	women's clothing	dresses
…	…	…
42420	women's clothing	traditional wear
42421	women's clothing	women's clothing
42422	mobile & gadgets	cases & covers
42423	mobile & gadgets	cases & covers
42424	mobile & gadgets	cases & covers

[42363 rows x 2 columns]

[11]:
```
import pprint
pd.reset_option('display.max_colwidth')
dataset_sample[dataset_sample['Type of product']=='baby & toys']['Type of␣
 ↪product_2'].unique()
```

[11]:
```
array(['baby clothing', 'baby gear', 'feeding & nursing',
       'bath & toiletries', 'formula & food', 'boys fashion',
       'toys & education', 'nursery', 'girls fashion',
       'diapers & potties', 'kids fashion accessories & bags',
       'baby & toys', 'baby & toddler play', 'kids sports & outdoor play',
```

```
       'kids health & skincare', 'baby safety', 'maternity care'],
      dtype=object)
```

[ ]:
```
```

[12]:
```
fashion_names=["women's clothing", "men's clothing", 'muslim fashion',
               'baby & toys', 'watches',
               "men's bags & wallets",
               "women's bags", 'sports & outdoor', 'fashion accessories'
               , "men's shoes", "women's shoes", 'women clothes']
fashion_names_2=['baby clothing', 'boys fashion', 'girls fashion', 'kids␣
 ↪fashion accessories & bags']
```

[13]:
```
new_set=dataset_sample[(dataset_sample['Type of product'].isin(fashion_names))␣
 ↪| (dataset_sample['Type of product_2'].isin(fashion_names_2))]
```

[15]:
```
print(new_set.columns)
new_set
```

```
Index(['price_ori', 'delivery', 'item_category_detail', 'specification',
       'title', 'w_date', 'link_ori', 'item_rating', 'seller_name',
       'idElastic', 'price_actual', 'sitename', 'idHash', 'seller_details',
       'location', 'total_rating', 'id', 'pict_link', 'total_sold', 'favorite',
       'timestamp', 'desc', 'Type of product', 'Type of product_2'],
      dtype='object')
```

[15]:

	price_ori	delivery	\
0	29.16	Shipping Pre-Order (ships in 11 days) Shipping…	
1	57.78	Shipping Free shipping Free shipping for order…	
2	82.00	Shipping Free shipping Shipping from overseas …	
3	115.00	Shipping Free shipping Shipping to KL City, Ku…	
4	40.90	Shipping Free shipping Shipping from overseas …	
…	…	…	
42417	6.00	Shipping Free shipping Shipping to KL City, Ku…	
42418	49.99	Shipping Free shipping Shipping to KL City, Ku…	
42419	32.00	Shipping Shipping from Mainland China to KL Ci…	
42420	105.00	Shipping Pre-Order (ships in 14 days) Shipping…	
42421	80.00	Shipping Shipping to KL City, Kuala Lumpur shi…	

	item_category_detail	\
0	[shopee, women's clothing, skirts]	
1	[shopee, women's clothing, dresses, midi dresses]	
2	[shopee, women's clothing, dresses, maxi dresses]	
3	[shopee, women's clothing, dresses, maxi dresses]	
4	[shopee, women's clothing, dresses, maxi dresses]	
…	…	
42417	[shopee, women's clothing, tops]	

```
42418         [shopee, men's shoes, sneakers, plimsolls]
42419  [shopee, women's clothing, traditional wear, s…
42420  [shopee, women's clothing, traditional wear, s…
42421       [shopee, women's clothing, traditional wear]

                                           specification  \
0      Category Shopee Women's Clothing Skirts Brand …
1      Category Shopee Women's Clothing Dresses Midi …
2      Category Shopee Women's Clothing Dresses Maxi …
3      Category Shopee Women's Clothing Dresses Maxi …
4      Category Shopee Women's Clothing Dresses Maxi …
…                                                    …
42417  Category Shopee Women's Clothing Tops Brand No…
42418  Product Specifications Category Shopee Men's S…
42419  Category Shopee Women's Clothing Traditional W…
42420  Category Shopee Women's Clothing Traditional W…
42421  Category Shopee Women's Clothing Traditional W…

                                                 title      w_date  \
0      Alice's new elegant style ultra-fairy French d…  20201123.0
1       Korean Vintage Style Square Neck Slim Midi D…  20201123.0
2      ZANZEA Women Sleeveless Drawstring Pleated Swi…  20201123.0
3      AIR SPACE V-Neck Ruffle Sleeve Tassel Side Tie…  20201123.0
4      ZANZEA Women Crew Neck Long Sleeve Ethnic Vint…  20201123.0
…                                                    …         …
42417        [Live Only] KNIT BY MEMZ | Shopee Malaysia  20201202.0
42418  BUM Equipment Unisex Canvas Shoes  - Black/Blu…  20201202.0
42419  Zumba T002061 T-shirt closet for fitness | Sho…  20201202.0
42420      Georgette Embroidery Saree | Shopee Malaysia  20201202.0
42421  SONGKET SUT 8 [ORIGINAL PAKISTAN MADE] | Shope…  20201202.0

                                            link_ori  item_rating  \
0      https://shopee.com.my/Alice's-new-elegant-styl…          NaN
1      https://shopee.com.my/ Korean-Vintage-Style-S…          NaN
2      https://shopee.com.my/ZANZEA-Women-Sleeveless-…          5.0
3      https://shopee.com.my/AIR-SPACE-V-Neck-Ruffle-…          NaN
4      https://shopee.com.my/ZANZEA-Women-Crew-Neck-L…          4.8
…                                                    …          …
42417  https://shopee.com.my/-Live-Only-KNIT-BY-MEMZ-…          5.0
42418  https://shopee.com.my/BUM-Equipment-Unisex-Can…          4.9
42419  https://shopee.com.my/Zumba-T002061-T-shirt-cl…          NaN
42420  https://shopee.com.my/Georgette-Embroidery-Sar…          NaN
42421  https://shopee.com.my/SONGKET-SUT-8-ORIGINAL-P…          5.0

             seller_name                         idElastic  …  \
0             8ysl9a1301  6e1e3d7b51a4c1099c368114ab91a88a  …
1      showcasemywardore  a4248a77a54045ea69f18cb87ada6fb2  …
```

```
2              zanzea.os  8ef1159741c3ad608ab376a1def1a8d5  …
3          airspacemy.os  e5f6337f3faa4badad794fb211b35c61  …
4              zanzea.os  3dbb5a357043464af36d6591d215a19c  …
…                    …                                 …  …
42417           ariel_83  7c0483d90b1c96e7dd11d1449a636788  …
42418       bumequipment  2e161a63b7837a7aedcc43cf41a4b876  …
42419        fitfunky.my  1043ad99bffb185eb11800469572f0b8  …
42420         padmavathi  04843cb146021c2102fb25d16889c5dd  …
42421           msalimjb  3c39f275ac5351c2d3bba188836d5904  …

                            location total_rating            id  \
0          Lubok China, 000001 Melaka            0  6.445109e+09
1            KLCC, 50088 Kuala Lumpur            1  5.445225e+09
2                      Mainland China            3  4.843064e+09
3        Kuala Langat, 42500 Selangor            0  6.245303e+09
4                      Mainland China          123  3.430108e+09
…                                 …          …           …
42417       Shah Alam, 40200 Selangor          491  3.735841e+09
42418  Seri Kembangan, 43300 Selangor          608  9.802219e+08
42419                             NaN            1  4.956021e+09
42420        Johor Bahru, 81300 Johor            2  6.318889e+09
42421        Johor Bahru, 81200 Johor            1  3.957804e+09

                                              pict_link total_sold favorite  \
0      https://cf.shopee.com.my/file/63256421fb228665…        NaN       80
1      https://cf.shopee.com.my/file/559058f73c771491…        NaN       61
2      https://cf.shopee.com.my/file/5ce472482e927cf4…          3       24
3      https://cf.shopee.com.my/file/fdab91ad3c366064…        NaN       28
4      https://cf.shopee.com.my/file/0af4b832a74478f1…        242      744
…                                                    …        …       …
42417  https://cf.shopee.com.my/file/5ccd1c7ff04ede72…        723        7
42418  https://cf.shopee.com.my/file/f42c1c57c9343878…        880      480
42419  https://cf.shopee.com.my/file/0829c6971f5e5170…        NaN      NaN
42420  https://cf.shopee.com.my/file/090cfbc76c52d1bb…        NaN        1
42421  https://cf.shopee.com.my/file/edb35b6b5e13ead7…          1        1

          timestamp                                               desc  \
0      1.606064e+12  Promotions: Summer's new elegant style ins sup…
1      1.606064e+12  - Color: Multicolor\n\n- Material: Polyester\n…
2      1.606064e+12  Item Type:Dress\nMaterial: Cotton \nColors:Gre…
3      1.606064e+12  Welcome to Air Space Malaysia Official where w…
4      1.606064e+12  Material:Cotton\nPackage?included:1Dress\nColo…
…               …                                                 …
42417  1.606842e+12  NAK CARI BAJU BUNDLE MURAH ? CANTIK ? BRANDED …
42418  1.606842e+12  Welcome To Our Online Store BUM Equipment ! \n…
42419  1.606842e+12  zumba fitness clothes zumba T02061 - Buy Zumba…
42420  1.606842e+12  *DN:HD50*\n\n*FABRIC:GEORGETTE WITH EMBROIDERY…
```

```
42421  1.606842e+12  BAJU MELAYU - 2PCS\nSAMPIN            - 1PCS\n…

        Type of product Type of product_2
0      women's clothing  women's clothing
1      women's clothing           dresses
2      women's clothing           dresses
3      women's clothing           dresses
4      women's clothing           dresses
…                   …                 …
42417  women's clothing  women's clothing
42418       men's shoes          sneakers
42419  women's clothing  traditional wear
42420  women's clothing  traditional wear
42421  women's clothing  women's clothing

[21105 rows x 24 columns]
```

[ ]:
```
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(10,10))
sns.heatmap(new_set.isnull(),yticklabels=False,cbar=False,cmap='viridis')
grouped = new_set[['price_ori', 'delivery', 'item_category_detail',␣
 ↪'specification',
       'title', 'w_date', 'link_ori', 'item_rating', 'seller_name',
       'idElastic', 'price_actual', 'sitename', 'idHash', 'seller_details',
       'location', 'total_rating', 'total_sold', 'Type of product']].
 ↪groupby('Type of product').agg('sum').reset_index()
grouped
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
Cell In[7], line 5
      3 plt.figure(figsize=(10,10))
      4 sns.heatmap(new_set.isnull(),yticklabels=False,cbar=False,cmap='viridis')
----> 5 grouped = new_set.groupby('Type of product').agg('sum').reset_index()
      6 grouped

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\core\groupby\gene
 ↪py:1432, in DataFrameGroupBy.aggregate(self, func, engine, engine_kwargs,␣
 ↪*args, **kwargs)
   1429     kwargs["engine_kwargs"] = engine_kwargs
   1431 op = GroupByApply(self, func, args=args, kwargs=kwargs)
-> 1432 result = op.agg()
   1433 if not is_dict_like(func) and result is not None:
   1434     # GH #52849
   1435     if not self.as_index and is_list_like(func):
```

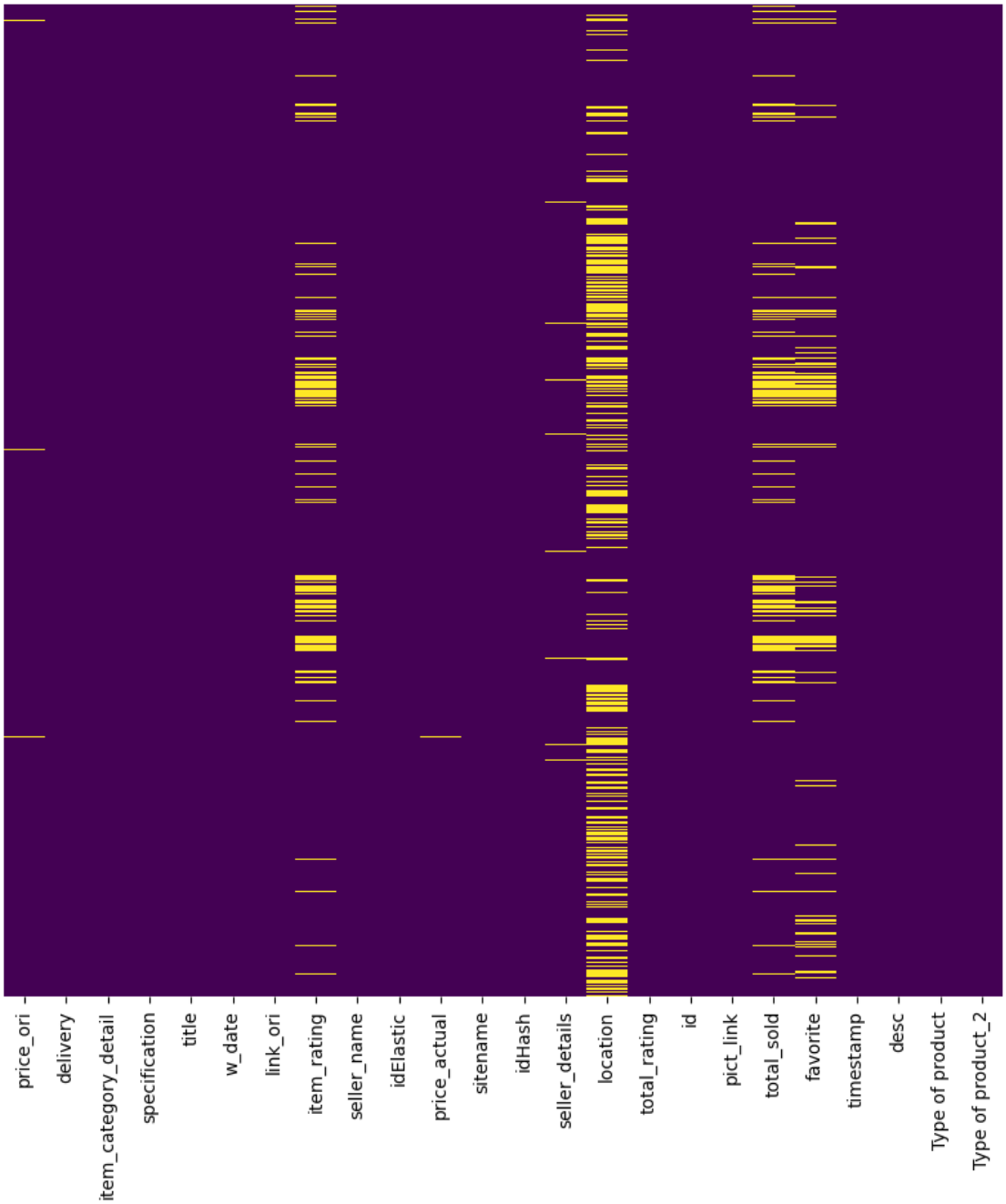
```
File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\apply.
↪py:187, in Apply.agg(self)
    184 kwargs = self.kwargs
    186 if isinstance(func, str):
--> 187     return self.apply_str()
    189 if is_dict_like(func):
    190     return self.agg_dict_like()

File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\apply.
↪py:603, in Apply.apply_str(self)
    601         else:
    602             self.kwargs["axis"] = self.axis
--> 603 return self._apply_str(obj, func, *self.args, **self.kwargs)

File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\apply.
↪py:693, in Apply._apply_str(self, obj, func, *args, **kwargs)
    691 f = getattr(obj, func)
    692 if callable(f):
--> 693     return f(*args, **kwargs)
    695 # people may aggregate on a non-callable attribute
    696 # but don't let them think they can pass args to it
    697 assert len(args) == 0

File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\grou
↪py:3146, in GroupBy.sum(self, numeric_only, min_count, engine, engine_kwargs)
  3141 else:
  3142     # If we are grouping on categoricals we want unobserved categories to
  3143     # return zero, rather than the default of NaN which the reindexing in
  3144     # _agg_general() returns. GH #31422
  3145     with com.temp_setattr(self, "observed", True):
-> 3146         result = self._agg_general(
  3147             numeric_only=numeric_only,
  3148             min_count=min_count,
  3149             alias="sum",
  3150             npfunc=np.sum,
  3151         )
  3153     return self._reindex_output(result, fill_value=0)

File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\grou
↪py:1906, in GroupBy._agg_general(self, numeric_only, min_count, alias, npfunc␣
↪**kwargs)
  1896 @final
  1897 def _agg_general(
```

```
   1898     self,
   (…)
   1904     **kwargs,
   1905 ):
-> 1906     result = self._cython_agg_general(
   1907         how=alias,
   1908         alt=npfunc,
   1909         numeric_only=numeric_only,
   1910         min_count=min_count,
   1911         **kwargs,
   1912     )
   1913     return result.__finalize__(self.obj, method="groupby")

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor e\groupby\grou
 ↪py:1998, in GroupBy._cython_agg_general(self, how, alt, numeric_only,␣
 ↪min_count, **kwargs)
   1995     result = self._agg_py_fallback(how, values, ndim=data.ndim, alt=alt
   1996     return result
-> 1998 new_mgr = data.grouped_reduce(array_func)
   1999 res = self._wrap_agged_manager(new_mgr)
   2000 if how in ["idxmin", "idxmax"]:

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor e\internals\ma
 ↪py:1469, in BlockManager.grouped_reduce(self, func)
   1465 if blk.is_object:
   1466     # split on object-dtype blocks bc some columns may raise
   1467     #  while others do not.
   1468     for sb in blk._split():
-> 1469         applied = sb.apply(func)
   1470         result_blocks = extend_blocks(applied, result_blocks)
   1471 else:

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor e\internals\bl
 ↪py:393, in Block.apply(self, func, **kwargs)
    387 @final
    388 def apply(self, func, **kwargs) -> list[Block]:
    389     """
    390     apply the function to my values; return a block if we are not
    391     one
    392     """
--> 393     result = func(self.values, **kwargs)
    395     result = maybe_coerce_values(result)
    396     return self._split_op_result(result)
```

```
File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\grou
↪py:1973, in GroupBy._cython_agg_general.<locals>.array_func(values)
   1971 def array_func(values: ArrayLike) -> ArrayLike:
   1972     try:
-> 1973         result = self._grouper._cython_operation(
   1974             "aggregate",
   1975             values,
   1976             how,
   1977             axis=data.ndim - 1,
   1978             min_count=min_count,
   1979             **kwargs,
   1980         )
   1981     except NotImplementedError:
   1982         # generally if we have numeric_only=False
   1983         # and non-applicable functions
   1984         # try to python agg
   1985         # TODO: shouldn't min_count matter?
   1986         # TODO: avoid special casing SparseArray here
   1987         if how in ["any", "all"] and isinstance(values, SparseArray):

File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\ops
↪py:831, in BaseGrouper._cython_operation(self, kind, values, how, axis,␣
↪min_count, **kwargs)
    829 ids, _, _ = self.group_info
    830 ngroups = self.ngroups
--> 831 return cy_op.cython_operation(
    832     values=values,
    833     axis=axis,
    834     min_count=min_count,
    835     comp_ids=ids,
    836     ngroups=ngroups,
    837     **kwargs,
    838 )

File c:\Users\wak␣
↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\ops
↪py:550, in WrappedCythonOp.cython_operation(self, values, axis, min_count,␣
↪comp_ids, ngroups, **kwargs)
    539 if not isinstance(values, np.ndarray):
    540     # i.e. ExtensionArray
    541     return values._groupby_op(
    542         how=self.how,
    543         has_dropped_na=self.has_dropped_na,
   (…)
    547         **kwargs,
    548     )
--> 550 return self._cython_op_ndim_compat(
```

```
    551     values,
    552     min_count=min_count,
    553     ngroups=ngroups,
    554     comp_ids=comp_ids,
    555     mask=None,
    556     **kwargs,
    557 )

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\ops
 ↪py:344, in WrappedCythonOp._cython_op_ndim_compat(self, values, min_count,␣
 ↪ngroups, comp_ids, mask, result_mask, **kwargs)
    341     # otherwise we have OHLC
    342     return res.T
--> 344 return self._call_cython_op(
    345     values,
    346     min_count=min_count,
    347     ngroups=ngroups,
    348     comp_ids=comp_ids,
    349     mask=mask,
    350     result_mask=result_mask,
    351     **kwargs,
    352 )

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\ops
 ↪py:505, in WrappedCythonOp._call_cython_op(self, values, min_count, ngroups,␣
 ↪comp_ids, mask, result_mask, **kwargs)
    498 result = result.T
    500 if self.how not in self.cast_blocklist:
    501     # e.g. if we are int64 and need to restore to datetime64/timedelta6
    502     # "rank" is the only member of cast_blocklist we get here
    503     # Casting only needed for float16, bool, datetimelike,
    504     #  and self.how in ["sum", "prod", "ohlc", "cumprod"]
--> 505     res_dtype = self._get_result_dtype(orig_values.dtype)
    506     op_result = maybe_downcast_to_dtype(result, res_dtype)
    507 else:

File c:\Users\wak␣
 ↪computer\AppData\Local\Programs\Python\Python313\Lib\site-packages\pandas\cor\groupby\ops
 ↪py:284, in WrappedCythonOp._get_result_dtype(self, dtype)
    281             out_dtype = "object"
    282     return np.dtype(out_dtype)
--> 284 def _get_result_dtype(self, dtype: np.dtype) -> np.dtype:
    285     """
    286     Get the desired dtype of a result based on the
    287     input dtype and how it was computed.
   (…)
    296         The desired dtype of the result.
```

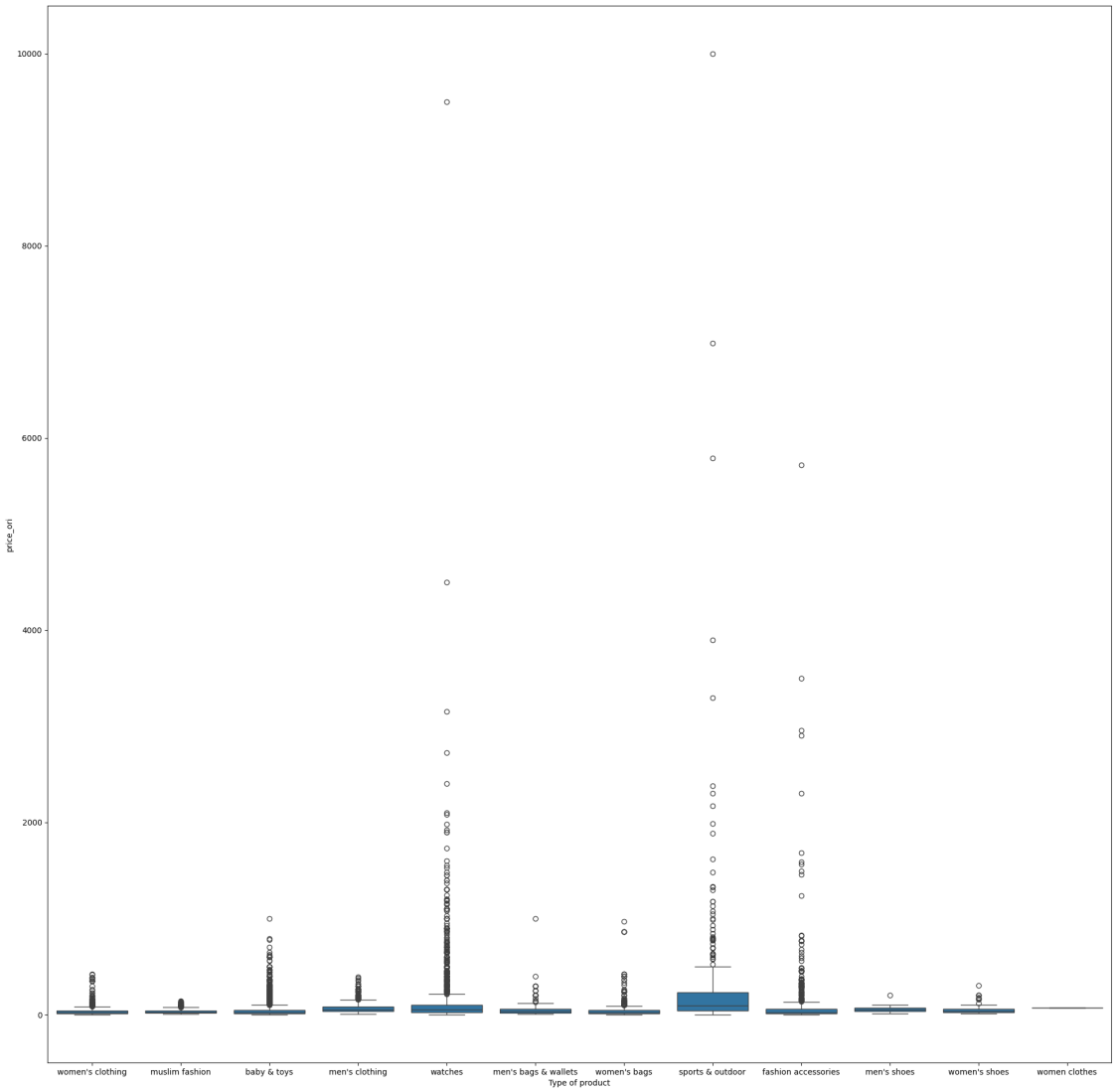
```
    297     """
    298     how = self.how

KeyboardInterrupt:
```



[ ]:
```
import tkinter as tk
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
import seaborn as sns

# Ensure loc is initialized
loc = 0

def on_click(event=None):
    global loc
    loc = (loc + 1) % 4
    ax.cla()

    if loc == 0:
        sns.heatmap(new_set.isnull(), cbar=False, yticklabels=False, ax=ax)
    if loc == 1:
        grouped = new_set.groupby('Type of product').agg('sum').reset_index()
        sns.histplot(data=grouped, x='Type of product', y='total_sold', ax=ax)
    if loc == 2:
        sns.boxplot(data=new_set[(new_set['price_actual']<100) ],␣
 ↪x='item_rating', y='price_actual', ax=ax)
    if loc==3:
        sns.boxplot(data=new_set[(new_set['price_actual']>100) &␣
 ↪(new_set['price_actual']<1000) ], x='item_rating', y='price_actual', ax=ax)
    canvas.draw()


# Initialize the Tkinter window
app_test_null = tk.Tk()
app_test_null.title("Missing Data")

# Create Matplotlib figure and axis
fig, ax = plt.subplots(nrows=1, ncols=1, figsize=(12, 12))
button = tk.Button(app_test_null, text="Next Plot", command=on_click)
button.pack()
# Create the canvas to embed the Matplotlib figure
canvas = FigureCanvasTkAgg(fig, master=app_test_null)
canvas.get_tk_widget().pack(fill=tk.BOTH, expand=True)
# Initial plot (Seaborn heatmap)
sns.heatmap(new_set.isnull(), cbar=False, yticklabels=False, ax=ax)
canvas.draw()



# Start the Tkinter event loop
app_test_null.mainloop()
```

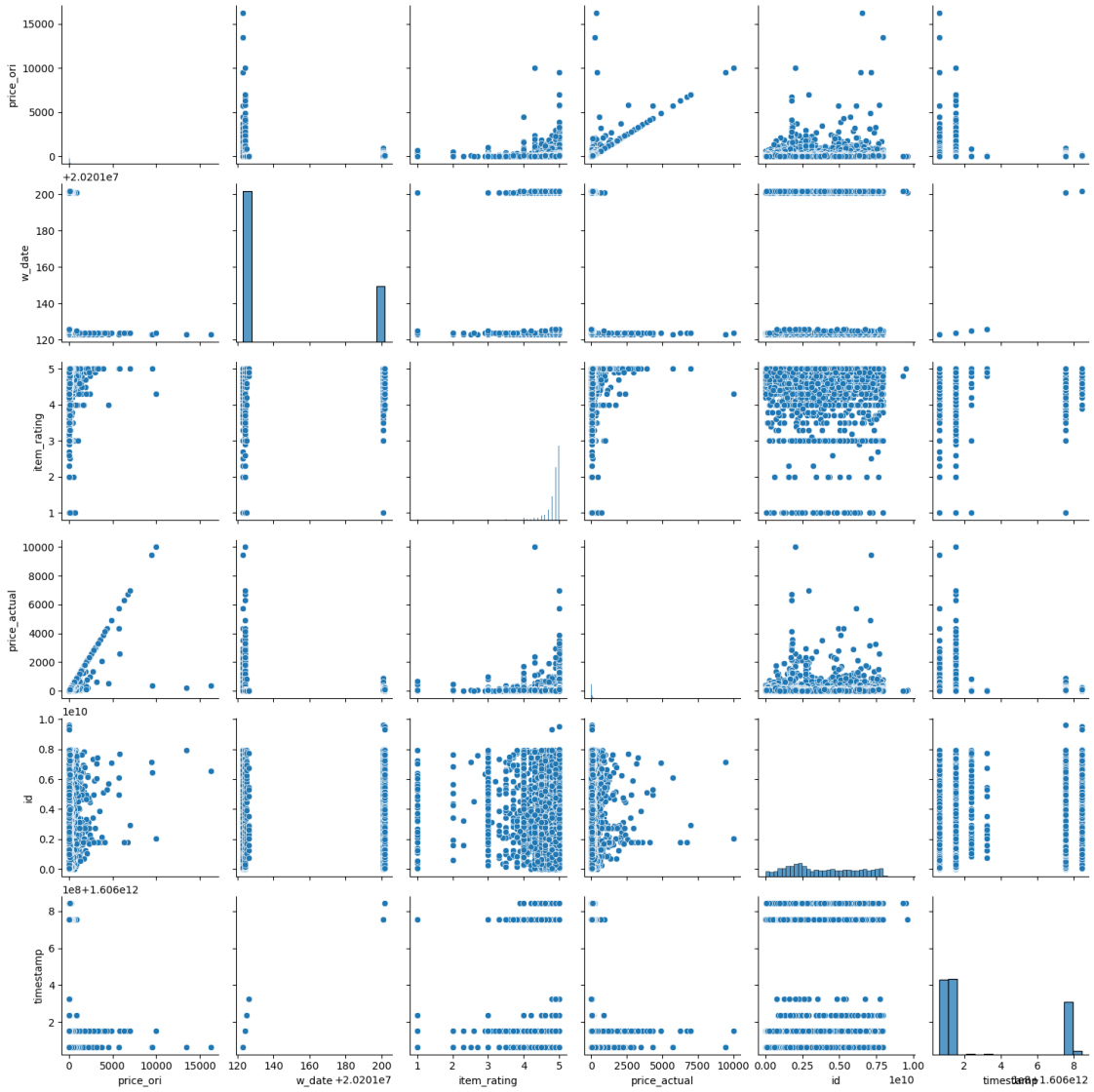
[ ]:
```
plt.figure(figsize=(24,24))
sns.boxplot(x='Type of product',y='price_ori',data=new_set.
 ↪dropna(inplace=False))
```

[ ]: <Axes: xlabel='Type of product', ylabel='price_ori'>



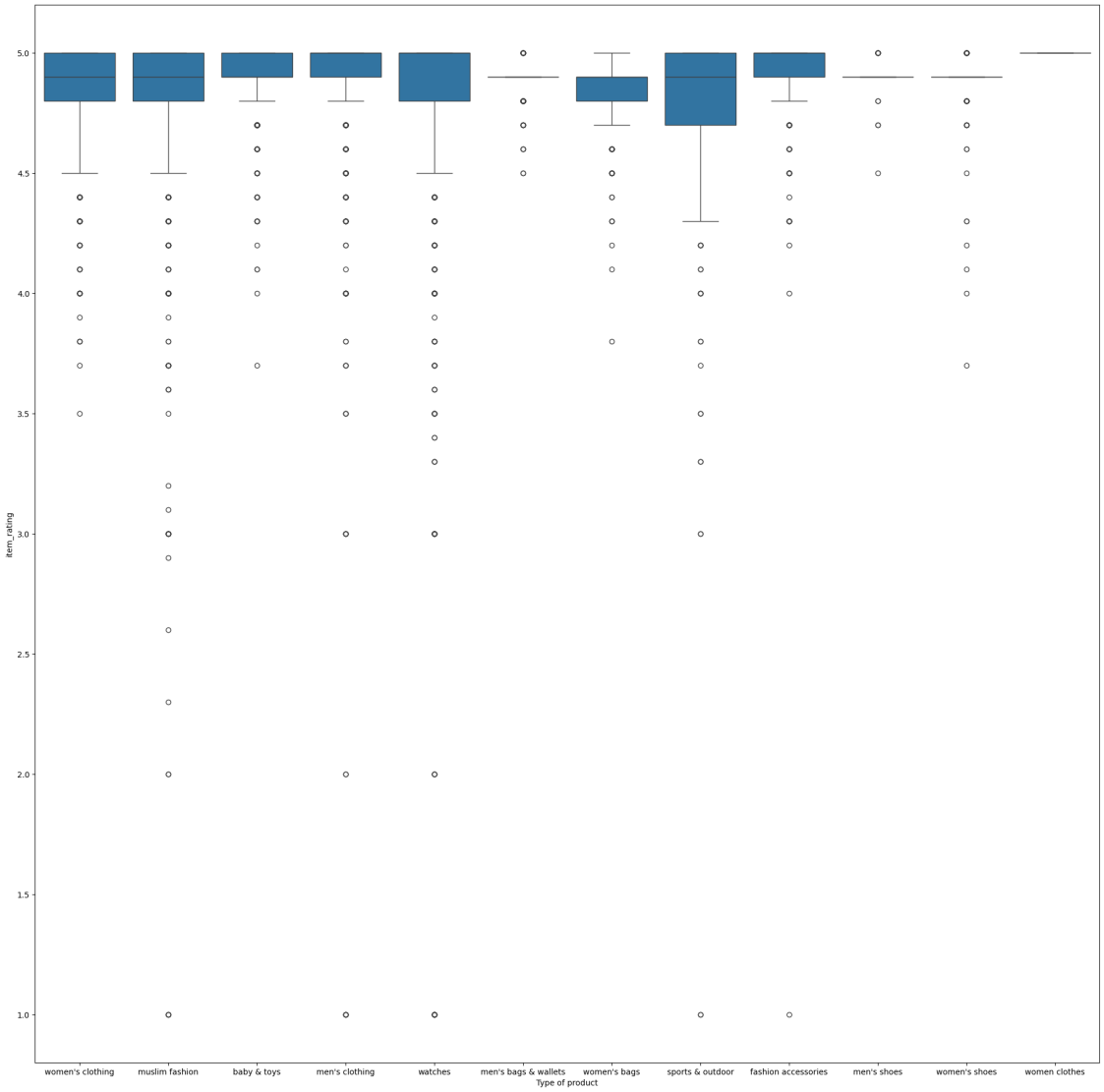
[ ]:
```
sns.pairplot(data=new_set)
```

[ ]: <seaborn.axisgrid.PairGrid at 0x23234dea660>



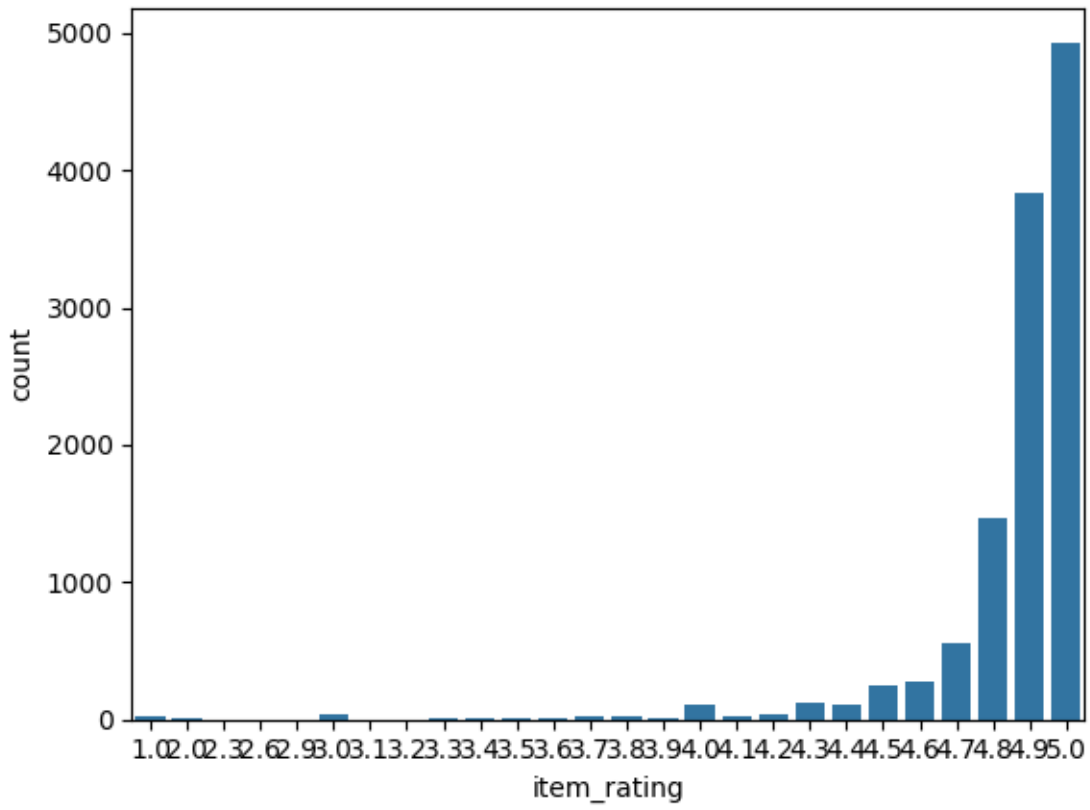
[ ]:
```
plt.figure(figsize=(24,24))
sns.boxplot(x='Type of product',y='item_rating',data=new_set.
  ↪dropna(inplace=False))
```

[ ]: <Axes: xlabel='Type of product', ylabel='item_rating'>



[ ]:
```
sns.countplot(data=new_set.dropna(inplace=False),x='item_rating')
```

[ ]: <Axes: xlabel='item_rating', ylabel='count'>



[ ]:
```
import plotly.express as px

px.histogram(new_set, x='item_rating', color='Type of product',␣
↪barmode='overlay')
```

[ ]:
```
px.histogram(
    data_frame=new_set,
    x='price_actual'
    )
```

[ ]:
```

```