FINAL PROJECT REPORT

IST 652_WINTER 2021-2022

TO : Professor: Dr Debbie Landowski

Contents

OVERVIEW

The purpose of this document is to provide the data analysis that has been carried out as part of final project for the current class of scripting for data analysis

The topic I've chosen is to analyze the publicly available ecommerce data of an ecommerce website.

This from the Wish ecommerce data set, however the data set at the being of the analysis seemed to be huge as the analysis progress it was realized that the data set is only for clothing category of women and men only with few attributes.

The main aim is to apply the techniques, rules, that are learned as part of course rather than the volume of data.

DATA SOURCE

Clothing data of wish.com that was dated in 2020 summer sales input for few clothing products of men and women

Data files will be attached as part of the submission.

This data is available at Kaggle and Data world.

URL: https://www.kaggle.com/datasets/jmmvutu/summer-products-and-sales-in-ecommerce-wish

INSIGHT OF THE DATA

It's so common that when we look up for a clothing related product information general tendency is to look for size and the favorite color and the rest of things like brand, price, ratings and discounts will be observed before any purchase is made.

Along with that based on the type of the product like shirts(multiple variety of shirts), bottoms, outer wear, ..etc, would be available.

GOAL OF THE CURRENT PROJECT

To identify the following

1) Is there any relation between items sold based on ratings

2) Can we derive any conclusions the purchases based on size and color

3) Do we get any insight which region (country) has more purchases

4) Does any ads, tags have any impact on items sold

5) Is there any data available to view how price would have influenced product purchase (i.,e lowest price item versus highest prices items)

6) Any predictions can be made based on ratings

7) Additionally, if I can apply any machine learning techniques to identify the trends, accuracy of data .etc.,

DATA PREPROCESSING

This is the first and foremost important step and many tasks need to be performed before we consider the data is good for analysis.

Typical checks are like

i) Identifying the duplicates

ii) Examine the same and remove them from the data set to be analyzed

iii) Identify the columns that has null values in the rows

iv) Removing the null rows based on the importance of the column

v) Identify the key columns and check the data type of the column

vi) Based on the type convert them to list and create dictionaries if necessary

For this current project analysis did all the above

As part of data acquisition, got the required data in the form of CSV from Kaggle,

Loaded the data set to the working directory for the better coordination of puling the rows and columns.

As part of data preprocessing and cleaning performed all the above tasks.

Import all the required packages both at Jupyter notebook level and anaconda prompt with the required commands

Here are the results for a quick view

Total Rows and column information

```
In [3]: productsdf=pd.read_csv(datapath, sep=',')
        productsdf.shape

Out[3]: (1573, 43)
```

After removing the duplicates and identifying the nulls

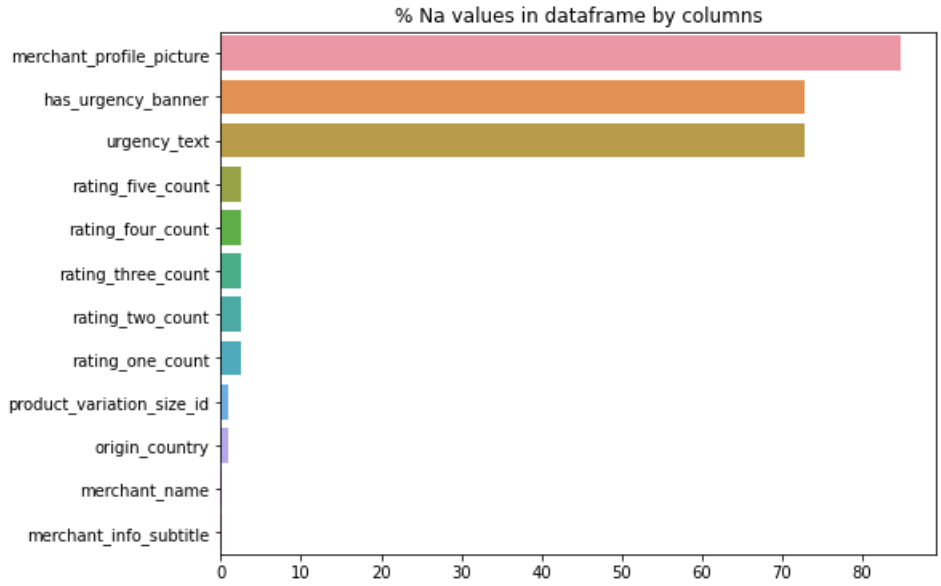
Quick data info:-

- There are 198 duplicate product IDs
- 34 duplicate records
- And the below information regarding the null values across various attributes.
- Total products with price greater than retail_price: 477
- Buyer currency as EURO

```
In [19]: #Number of columns and rows after removing the duplicates
         productsdf.shape

Out[19]: (1505, 43)

In [20]: # Before removing duplicates it was (1573, 43) and after removing duplicates the data shows as (
         #have removed 34 duplicate rows information
         #Now we have to identify if there are any null values
         null_val = pd.DataFrame(productsdf.isnull().sum())
         null_val.columns = ['null_val']

In [21]: null_val['percent_'] = round(null_val['null_val'] / len(productsdf.index), 2) * 100
         null_val.sort_values('percent_', ascending = False)[:10]

Out[21]:
```

	null_val	percent_
merchant_profile_picture	1281	85.0
has_urgency_banner	1042	69.0
urgency_text	1042	69.0
rating_two_count	41	3.0
product_color	41	3.0
rating_five_count	41	3.0
rating_four_count	41	3.0
rating_three_count	41	3.0
rating_one_count	41	3.0

```
plot_missing_data(productsdf)
```



DATA EXPLORING

By taking a deep dive into the data the following understandings are drawn in accordance with questions that thought through

SIZE DATA ANALYSIS

As we know that this data set is with clothing products information and the sizes of the different types of products are mentioned in various forms by merchandisers, important observation is that there won't be any common and single approach for this declaration of sizes as these comes from different merchants

Here are the different types are mentioned.

```
pd.DataFrame(productsdf['product_variation_size_id'].value_counts())
```

	product_variation_size_id
S	619
XS	332
M	196
XXS	88
L	49
...	...
S Diameter 30cm	1
6XL	1
AU plug Low quality	1
5PAIRS	1
SIZE S	1

105 rows × 1 columns

The data results with "105" different sizes, it will be very difficult to perform any type of analysis with such a categorization, hence we need to figure out the common approach by combining or grouping 2 to 3 sizes as single size and then identify how products were sold.

The row name as 'product_variation_size_id', and the simplest approach I tried was if and else if statements with conditions. Code is available in the Code file

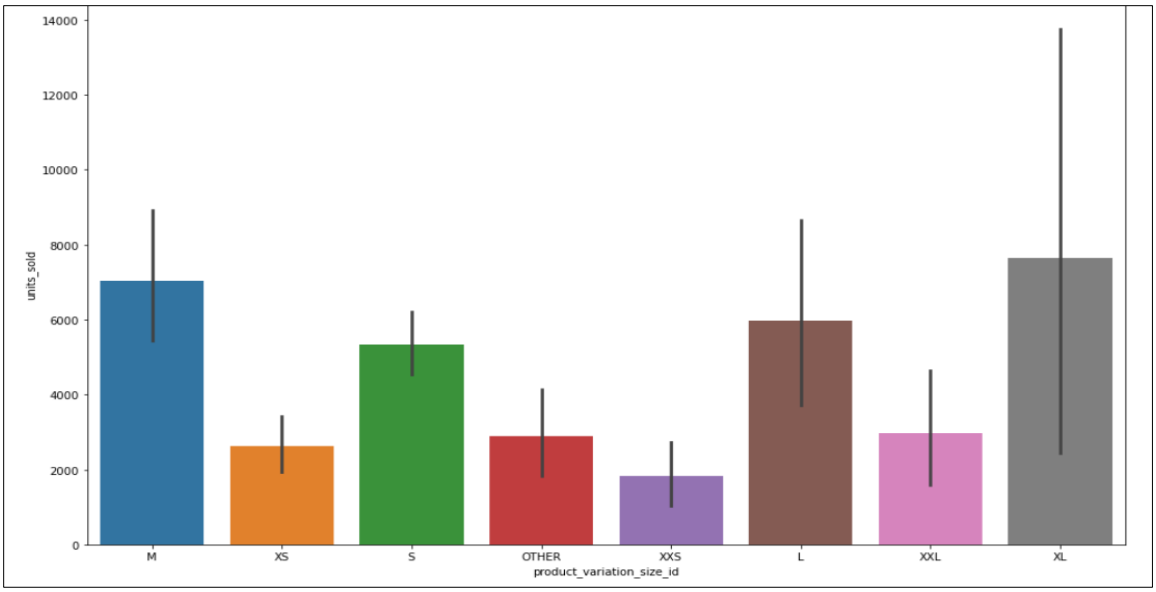
Below is the output

```
pd.DataFrame(productsdf['product_variation_size_id'].value_counts())
```

]:

	product_variation_size_id
S	636
XS	341
M	200
OTHER	125
XXS	97
L	51
XXL	37
XL	18

By using this Size analysis, we could know how many products are sold based on sizes. Definitely there is a relationship between sizes and items sold.



There are few more analysis done, and graphs are plotted in the code file which results that, merchants should have more "stock" for few sizes based on the items that are getting sold.

However, this analysis is not going to give complete picture of sizes with respect to product type like Pants, shirts, bottoms, sleepwear ..etc., as we don't have category and sub category like class and department or style of the product name

COLOR ANALYSIS

The next one comes in line is about color of the product as color plays a vital role which in turn can predict which color fabric can be made available in stock.

There are very interesting findings when the color data was analyzed, it was nearly 102 colors as the naming convention would be different from different merchants and the merchandising tools used.

However, this data was brought down as follows with the grouping and combining certain combinations. The Final result is

```
In [43]: count = productsdf['product_color'].value_counts()
         count

Out[43]: black     270
         white     212
         blue      138
         green     126
         red       120
         other      98
         pink       93
         yellow     82
         grey       79
         purple     50
         dual       34
         orange     26
         brown      13
         Name: product_color, dtype: int64
```

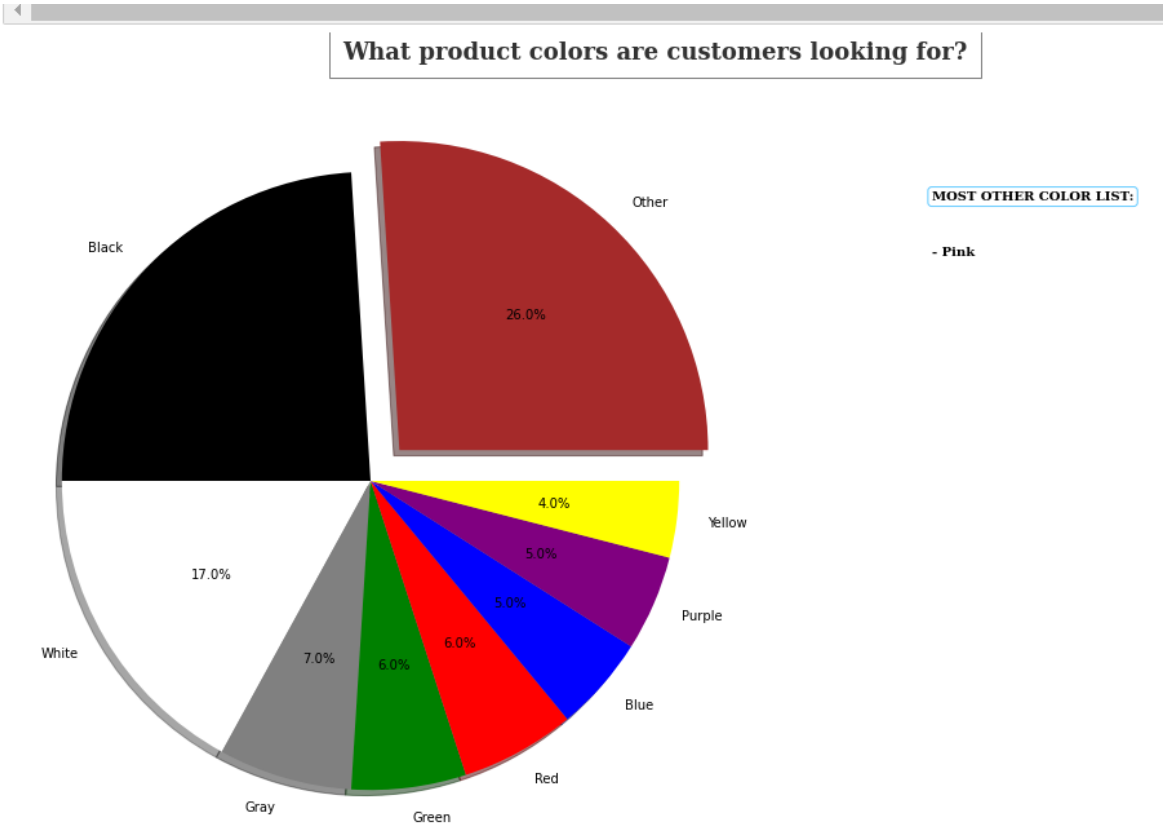
```
In [44]: col_df = productsdf.groupby('product_color').agg('sum')['units_sold'].to_frame()
         col_df.reset_index(level=0, inplace=True)
         col_df
```

Out[44]:

	product_color	units_sold
0	black	1482393
1	blue	557348
2	brown	31750
3	dual	123350
4	green	569575
5	grey	497420
6	orange	183758
7	other	628122
8	pink	298950
9	purple	338320
10	red	467050
11	white	1063211
12	yellow	223262

Note : the code file has the coding elements, how the data is grouped .

Below is one of the sample plot that shows which color products are brought more by the customers



From this one more prediction can be conveyed to the merchant

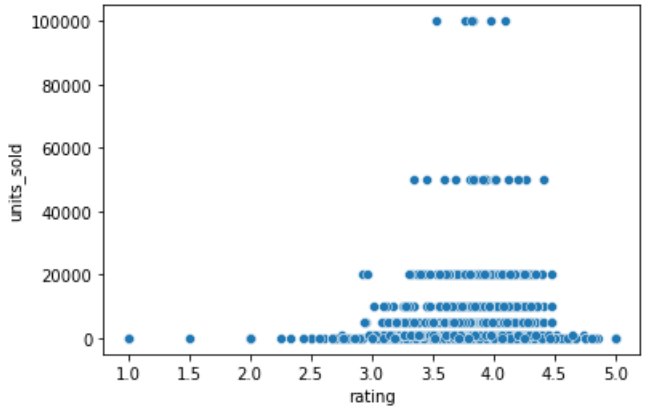
1) Having more stock of the items that are black in color 25% of the items sold are black in color

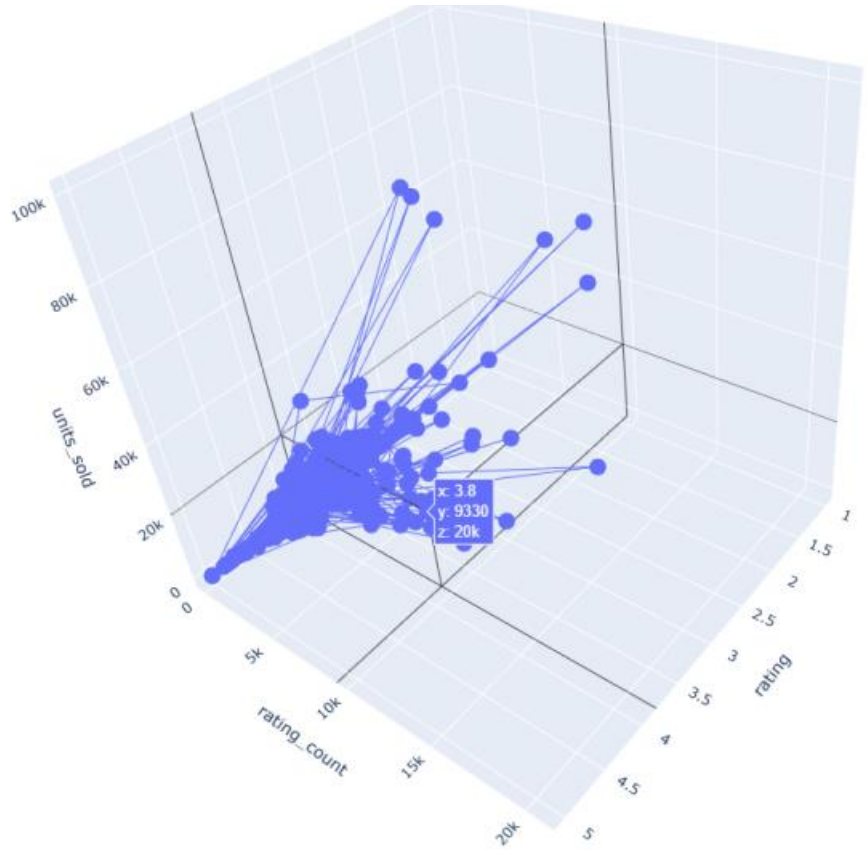
2) At the same time the 25 % in other color which is mix category of various color, in this sector merchant can put little more effort to have the parent color and combination color so that accurate result can be attained.

RATINGS ANALYSIS

In this data set Ratings played another important role. More ratings more items are sold here is the plot that shows how the ratings will influence the successful purchase.

```
In [21]: sns.scatterplot(data=productsdf,x='rating',y='units_sold')
Out[21]: <AxesSubplot:xlabel='rating', ylabel='units_sold'>
```





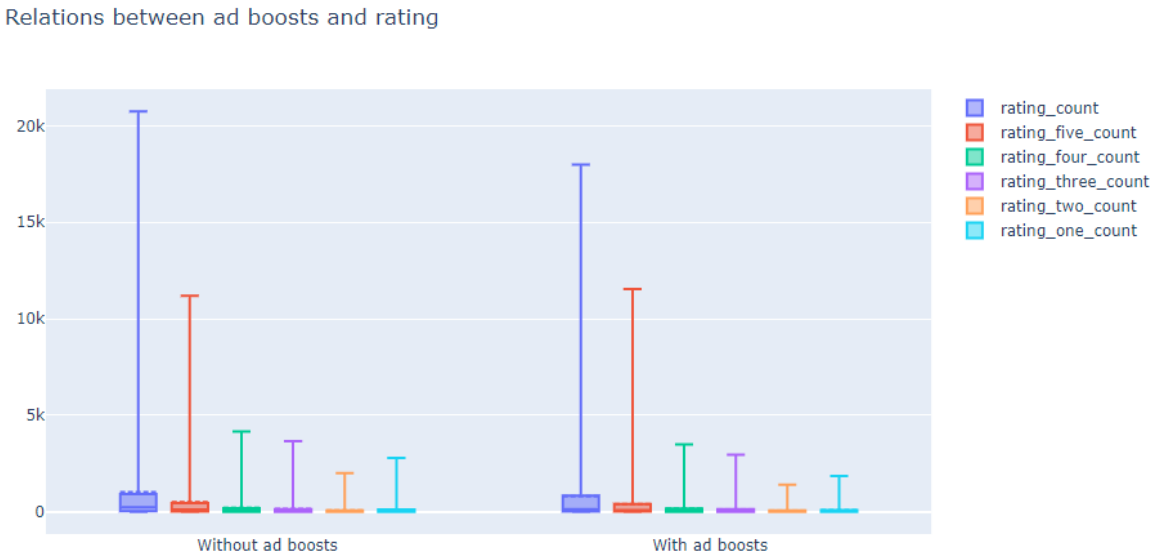
The code and data preparation for these plots are available in the code files Part1 file.

Ratings in relation with Ads

Along with ratings ad Boosts up the below is the same analysis, however I kept this analysis under ratings as a sub part

The kind of stars influenced the Ad Boosts

This shows that ads might influence with lesser rating products to get be sold, however the higher rated items doesn't required much ads to Boost up sales.




```
productsdf.groupby(['uses_ad_boosts'])['units_sold'].describe()
```

uses_ad_boosts	count	mean	std	min	25%	50%	75%	max
0	757.0	5027.158520	10125.796202	1.0	100.0	1000.0	5000.0	100000.0
1	584.0	4552.996575	9714.160788	10.0	100.0	1000.0	5000.0	100000.0

```
rating_cols=['rating_count','rating_five_count','rating_four_count',
             'rating_three_count','rating_two_count','rating_one_count']
ratings_data=productsdf[rating_cols+['uses_ad_boosts']]
```

```
ratings_data.groupby('uses_ad_boosts').describe()
```

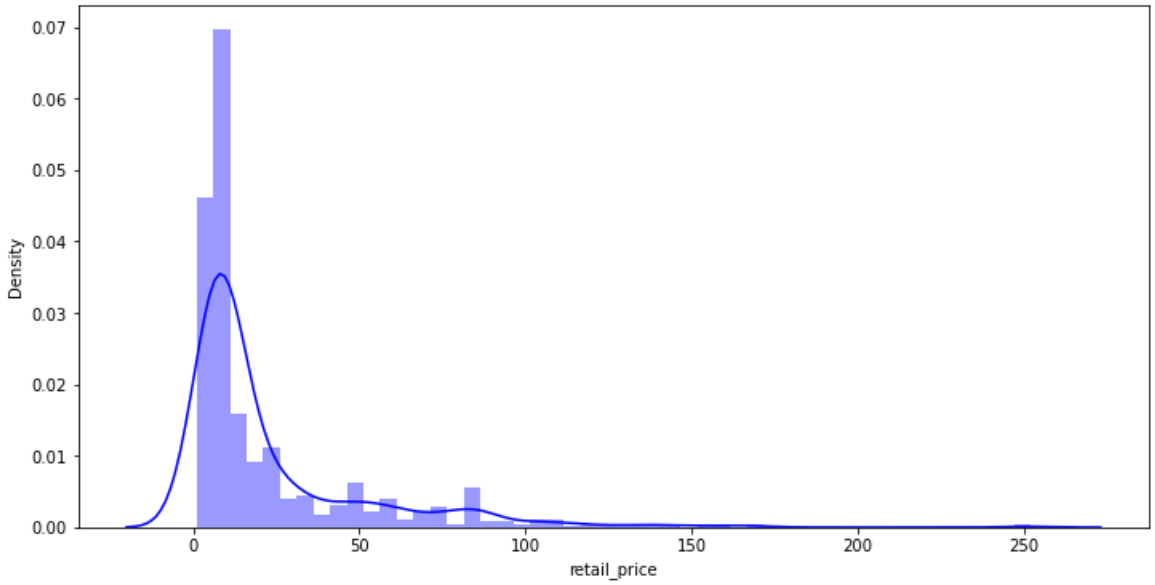
	rating_count								rating_five_count		...	rating_two_count				
	count	mean	std	min	25%	50%	75%	max	count	mean	...	75%	max	count	mean	std
uses_ad_boosts																
0	757.0	1080.096433	2255.718034	0.0	35.0	277.0	999.0	20744.0	736.0	536.010870	...	77.00	2003.0	736.0	115.293478	245.94377
1	584.0	854.518836	1856.903440	0.0	28.0	150.5	901.5	17980.0	570.0	424.305263	...	63.75	1410.0	570.0	91.182456	195.42917

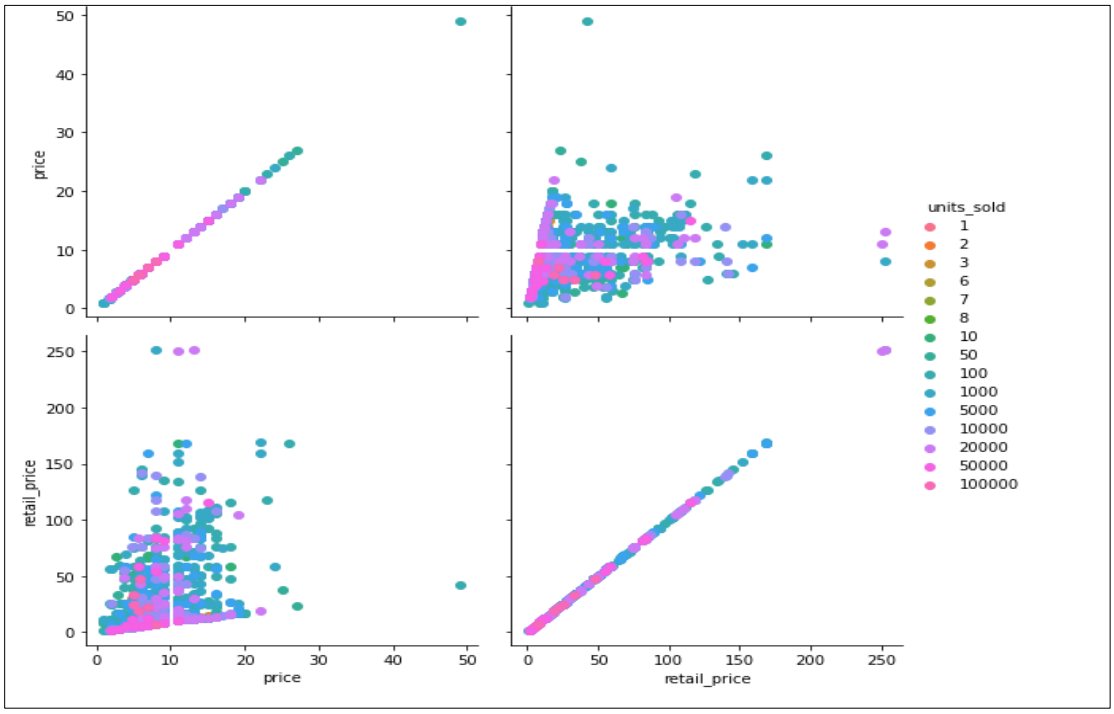
2 rows × 48 columns

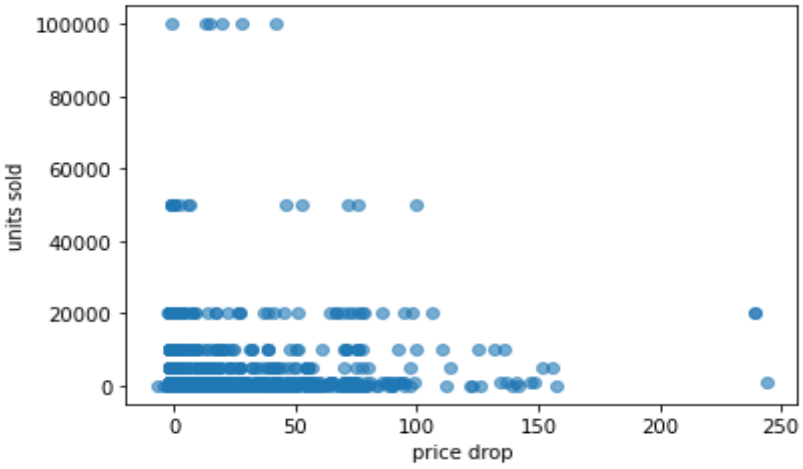
Relation Between Sale Price vs Retail Price

There is the data that shows that sale price is less than the retail price and it influenced the no of units sold. As we have the data for these three attributes unit sold, price and retail price.

The below picture shows the density of retail price







This plot clearly shows where the items are sold more the more spread is across the less price range than that of high ticked items.

DISCOUNT ANALYSIS

Not only the sale price and retail price there will be lot of encouragement to have the items sold with discounts. Here are the plots the represents how the items are sold with and without discounts

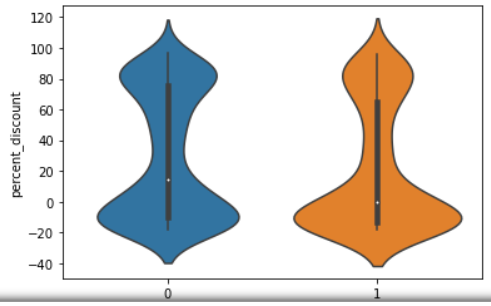
```
productsdf['percent_discount'] = (productsdf['retail_price'] - productsdf['price']) / productsdf['retail_price'] * 100
productsdf[['retail_price', 'price', 'percent_discount']].head()
```

t[36]:

	retail_price	price	percent_discount
0	14	16.00	-14.285714
1	22	8.00	63.636364
2	43	8.00	81.395349
3	8	8.00	0.000000
4	3	2.72	9.333333

```
sns.violinplot(data=productsdf, y='percent_discount', x='is_success')
```

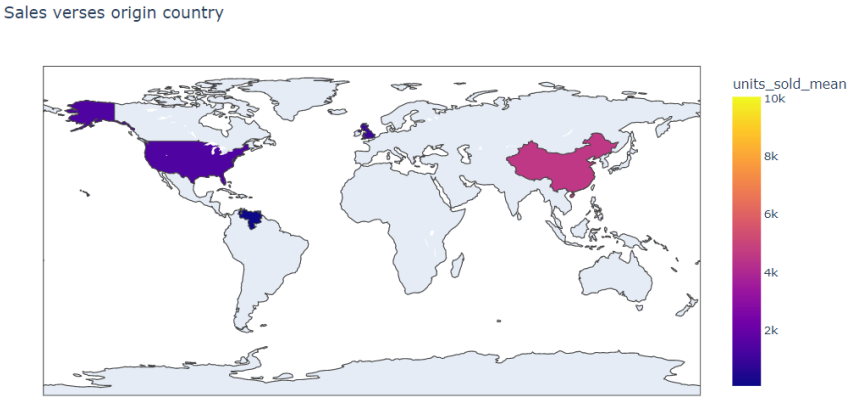
t[37]: <AxesSubplot:xlabel='is_success', ylabel='percent_discount'>



SHIPPING COST & SHIPPED COUNTRIES ANALYSIS

The data set provides information about the items that that were shipped across globe. The below figure represents the how many countries are on the top list also we can visualize how the shipping cost influences the shipping

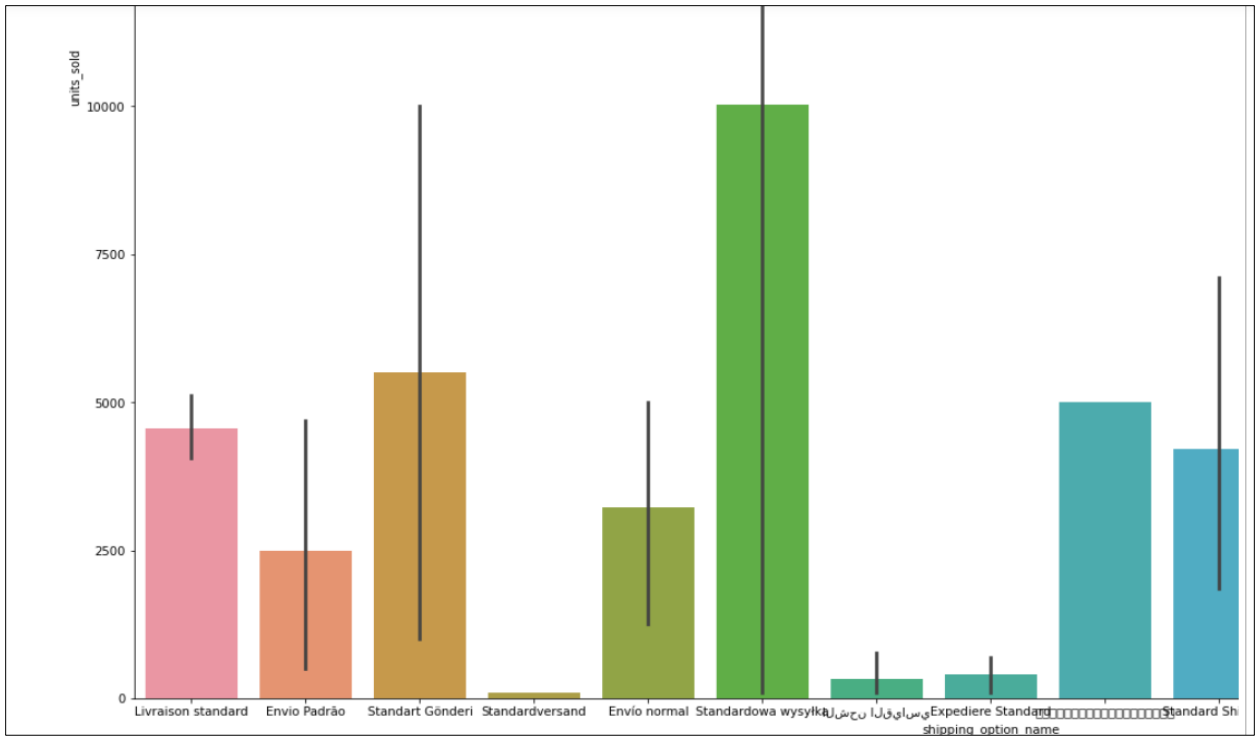
However, there is a caveat to do wholistic analysis as this data set is only for one month and for only clothing with summer sales.




```
productsdf.groupby('shipping_option_name').agg(['count', 'sum'])['units_sold']
```

52]:

shipping_option_name	count	sum
Ekspresowa wysyłka	1	10000
Envio Padrão	9	22400
Envio normal	5	16100
Expediere Standard	6	2400
Livraison Express	3	1200
Livraison standard	1440	6572589
Spedizione standard	2	1100
Standard Shipping	21	88550
Standardowa wysyłka	3	30100
Standardversand	3	300
Standart Gönderi	2	11000
Стандартная доставка	3	10100
الشحن القياسي	4	1300
การส่งสินค้ามาตรฐาน	2	10000
ការដឹកជញ្ជូនតាមស្តង់ដារ	1	10000



Along with these analysis it shows that

Out[65]:

	origin_country	units_sold_mean	code
0	AT	100.000000	NaN
1	CN	4554.703168	CHN
2	GB	1000.000000	GBR
3	SG	10050.000000	SGP
4	US	1405.161290	USA
5	VE	83.333333	VEN

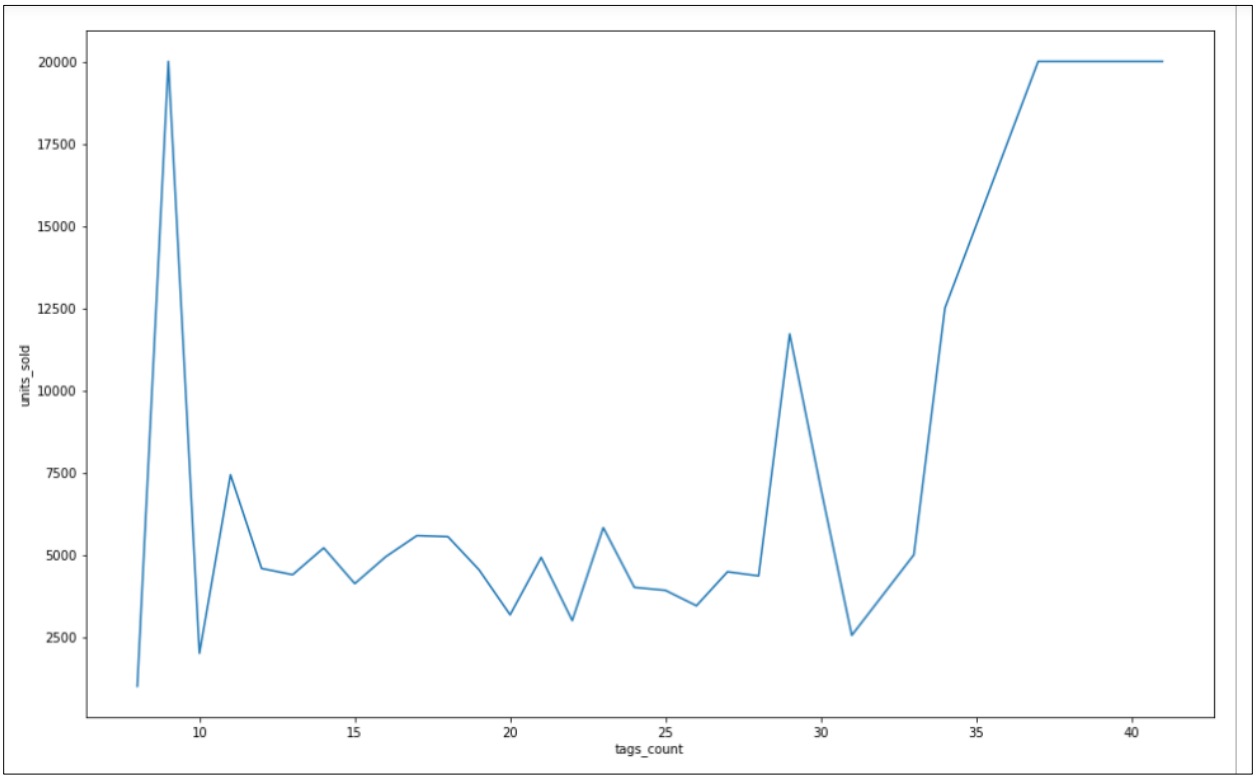
66]: `import plotly`

China is the highest country where the items are being shipped.

There has been some overall analysis carried out with the available data after data clean up

One of the interesting find out was about the item tagging , the more the tags to items the more have been sold this shows the social media tagging has pretty much influence in customer's mindset.

Its was high for few items and low for certain items and then a steady line for few items, below depiction shows the same.

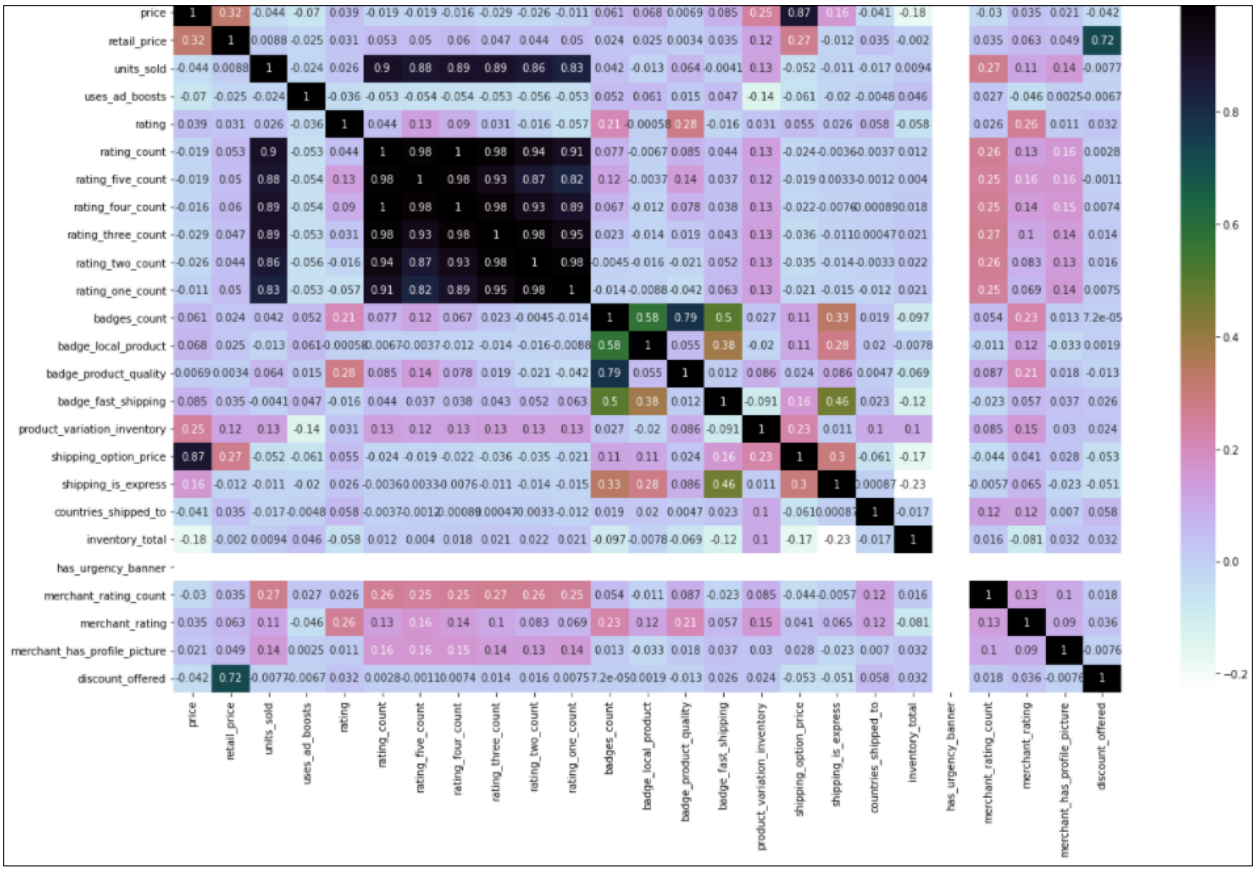


Over all Analysis

The principles of the data analysis can be found more useful and the most trusted techniques are such as correlation, regression and machine learning (Random forest model, Clustering, Train the model with and test them)

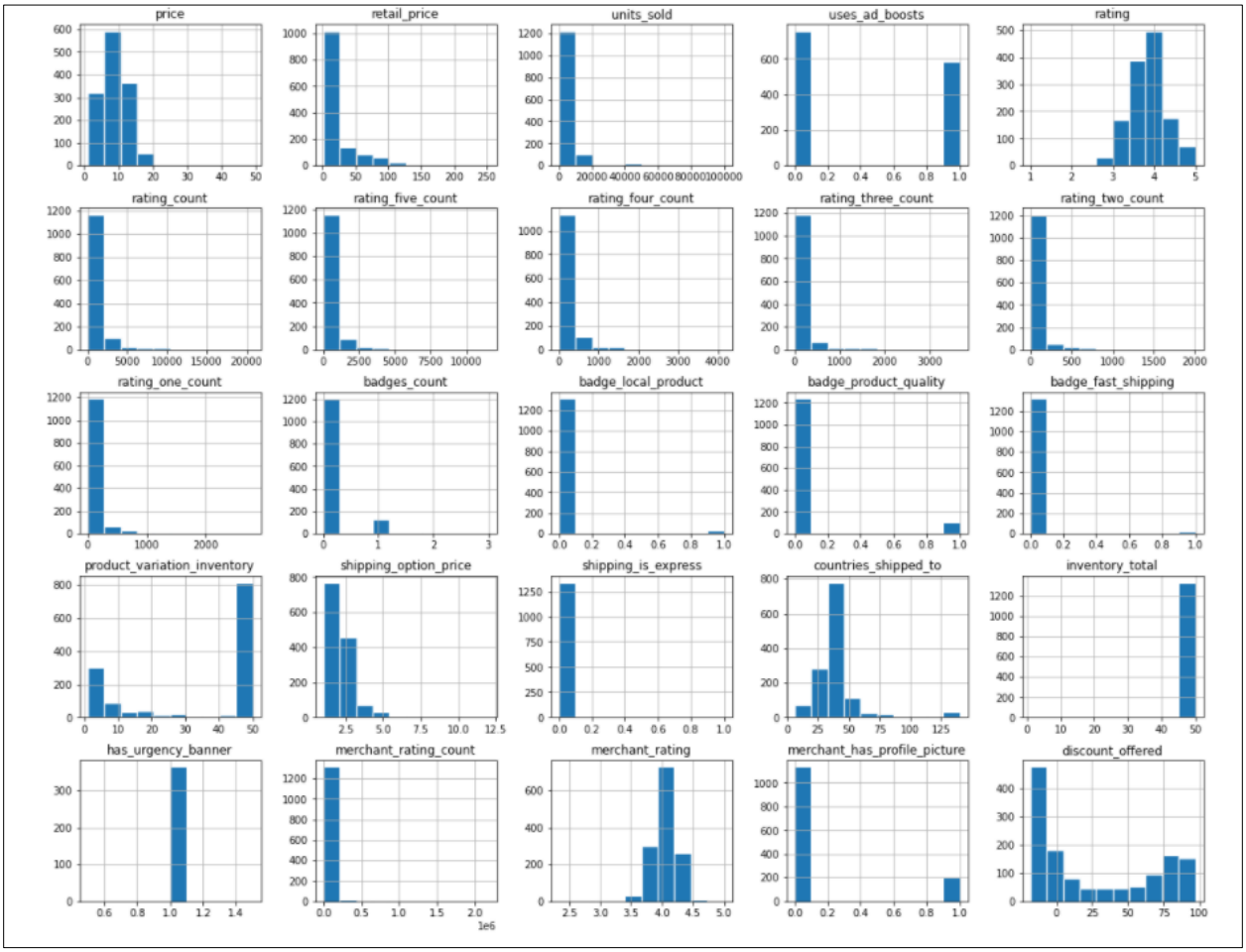
I've tried K means for the unit sold, I ended up in errors.

However I could get the over all correlation matrix with the default iris and the plot as overall heat map.


price
retail_price
units_sold
uses_ad_boosts
rating
rating_count
rating_five_count
rating_four_count
rating_three_count
rating_two_count
rating_one_count
badges_count
badge_local_product
badge_product_quality
badge_fast_shipping
product_variation_inventory
shipping_option_price
shipping_is_express
countries_shipped_to
inventory_total
has_urgency_banner
merchant_rating_count
merchant_rating
merchant_has_profile_picture
discount_offered
0.8
0.6
0.4
0.2
0.0
−0.2

Over all histogram of data

The below depiction shows over all histogram of available and useful data and how the items are sold based on price, color, size, ratings, discounts, tags, ads and the shipping costs.



Simplest K means for Units sold

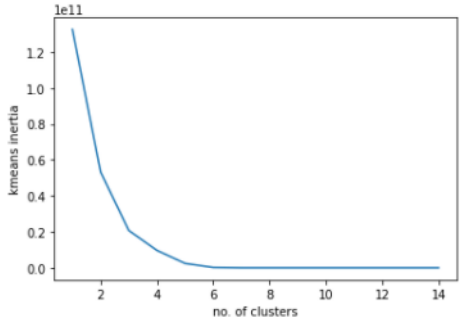
I've strong wish to experiment on the actual machine learning techniques and try to see how the model works, as mentioned due to errors I couldn't proceed further with that aspect

However, I've one small K means worked out and provided the analysis how the

tags_count

```
from sklearn.cluster import KMeans

clusters = {}
for i in range(1,15):
    kmeans = KMeans(n_clusters=i).fit(productsdf[['units_sold']])
    clusters[i] = kmeans.inertia_

plt.plot(list(clusters.keys()), list(clusters.values()));
plt.xlabel('no. of clusters');
plt.ylabel('kmeans inertia');
```

```
C:\Users\praso\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:881: UserWarning: KMeans is known to have a memory lea
k on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variab
le OMP_NUM_THREADS=6.
  warnings.warn(
```



SWEETVIZ REPORT

Apart from the analysis of the data by using a lot of libraries and cleansing the data and developing the models with the help of online and google I could figure out a package called “sweetviz” this helps to train the data automatically and provides an overall report

However this is not recommended for the current submission but in the real world for quick understanding of raw data these tools will help and then analysts can do a exploratory analysis of the categories or departments that they wish to have

I’m attaching the pdf version of the report as part of submission

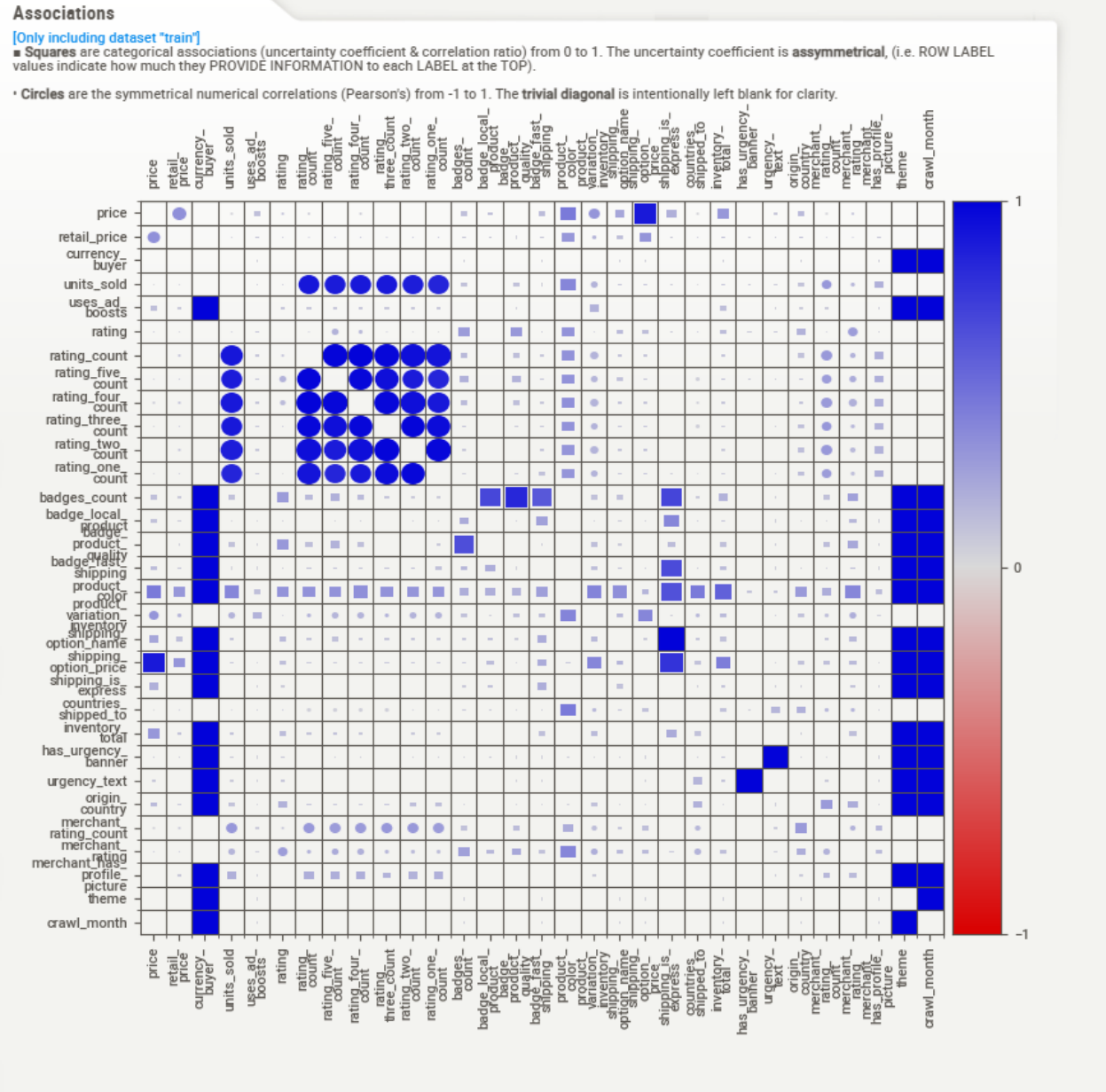
import sweetviz as sv

train = productsdf

report = sv.analyze([train,'train'],target_feat='price')

report.show_html('report.html')

Whole view of associations



Final Conclusions

This current data set is with only clothing for women and men and it was for only one month in July summer sales.

With the data that was available the following conclusions are drawn

1) Higher the ratings and higher the purchases

2) Ad Boost ups and Tagging doesn't have major influences in purchases made

3) Most of the purchases shows that black color and Size S are sold more this might be that data that made available, it may be different if we have whole data of various categories

4) Discounts and Price drop made sales to boost up

5) Also, the predictions are that CHINA is the country which has more items that are shipped

6) This is for merchandising team how they are merchandising at their site like have 102 different sizes and 120 colors this would be tough when someone has to analyze the whole website

Either their must be certain pattern based on the categories like clothing, accessories, shoes …etc., and avoid the duplication of product IDS and Color coding and names. The same applies to the Size clothing, shoes and non-apparel items.

7) Shipping costs this would be definitely vary based on the promotions that they would run on their site, though the analysis is done with the available data this can't be taken as a conclusion that the shipping costs and shipping methods that are followed by the ecommerce site.

8) One of the most influential factor with current trend is tagging, with the current data set it shows that the mechanism how the merchandise teams tags the items need to be improvised so that results can be more successful rather than just being the tag counts.

Lessons Learnt

As part this 1500+ row information and 43 columns data set there are lots of lessons learnt for me

1) I couldn't apply LAMDA function to convert the sizes and color in upper case string I was encountering an "type error saying float can be used with Lamda" even though I've converted them as list of sizes and colors, Finally I switched to arrays and got the result

2) Like wise I could not get the word cloud even after multiple times reinstalling the required software.

References

Along with Asynch material and labs below is the help findout from internet

1. Why the digital shelf is key for eCommerce analytics providers - Import.io
2. Analytics Ready Data | Enthought, Inc.
3. Data Science vs. Data Analytics vs. Machine Learning [2022 Edition] (simplilearn.com)
4. The Numbers Game Deciphered | Simplilearn
5. Top 10 Business Analytics Tools Used by Companies Today [2022 Edition] (simplilearn.com)
6. What is Data Analysis? Types, Methods and Techniques [2022 Edition] | Simplilearn
7. American Statistical Association (amstat.org)
8. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
9. https://towardsdatascience.com/boosting-showdown-scikit-learn-vs-xgboost-vs-lightgbm-vs-catboost-in-sentiment-classification-f7c7f46fd956
10. https://machinelearningmastery.com/random-forest-ensembles-with-xgboost/
11. https://catboost.ai/en/docs/concepts/python-usages-examples
12. https://realpython.com/python-keyerror/
13. https://realpython.com/python-dicts/
14. https://www.lfd.uci.edu/~gohlke/pythonlibs/#wordcloud
15. https://www.codegrepper.com/code-examples/python/%27float%27+object+has+no+attribute+%27replace%27