Global Superstores Data Exploration and Analysis for website Recommender System

Malungisa Mndzebele, DS-490-2021SP: Senior Project, Instructor: Dr. Shafqat Ali Shad

Introduction

Motivation:

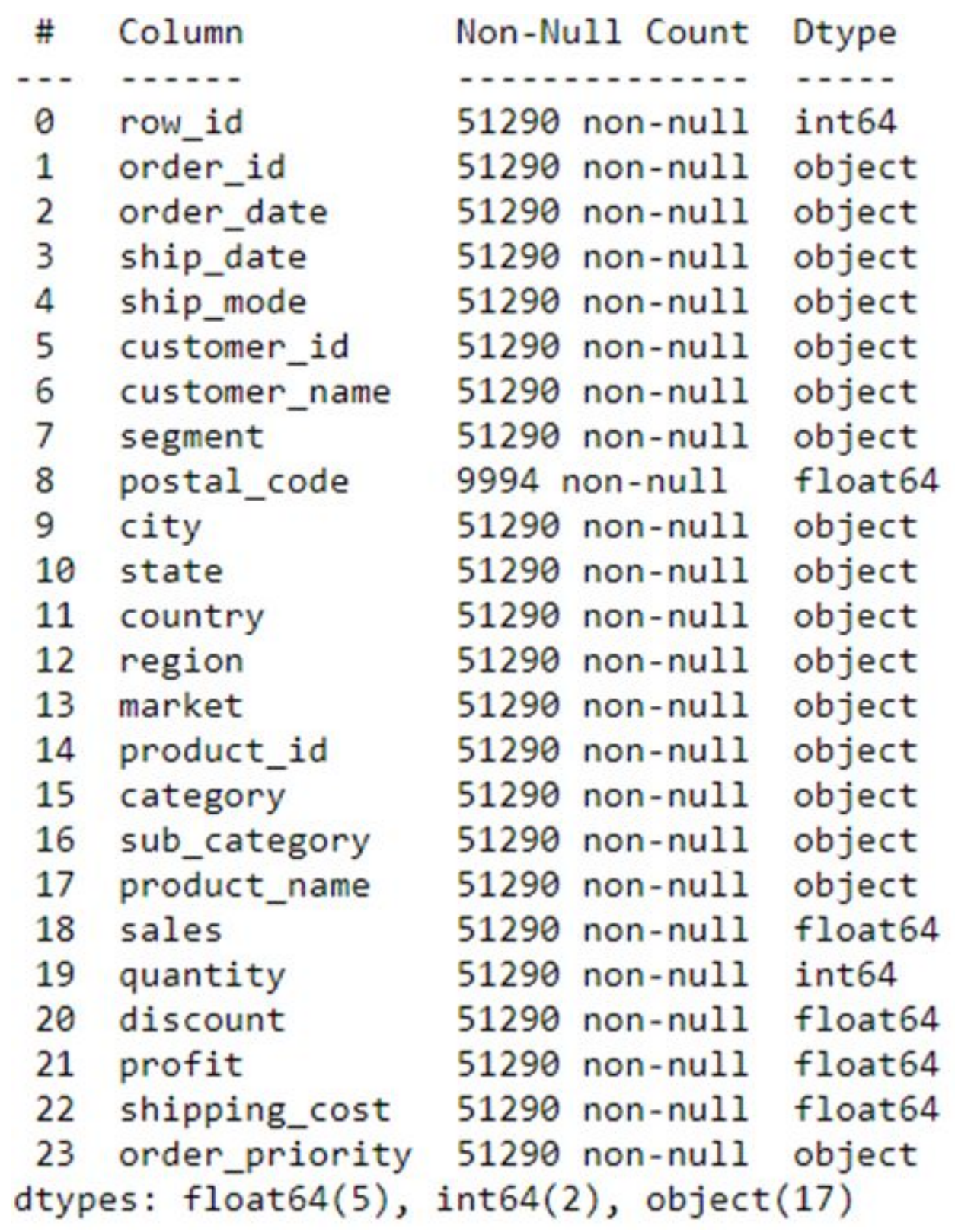
The main problem to be considered in this project is; Using grocery store customer and transaction data, can we make a reliable recommendation system for future customers? I thought it would be important to do my senior project focusing on this question because I think since all businesses depend on some form of recommender system to ensure efficiency, this would be a great skill to have. Hopefully by the end of this project I should have a good basic understanding of how recommender system like that of amazon or Netflix works.

Some insights from this project should also include the relationship between products bought by customers. I plan to make an interactive page where a customer can add a product to their cart and based on that the system should recommend other products to buy based on customer preference and other customers' activities.

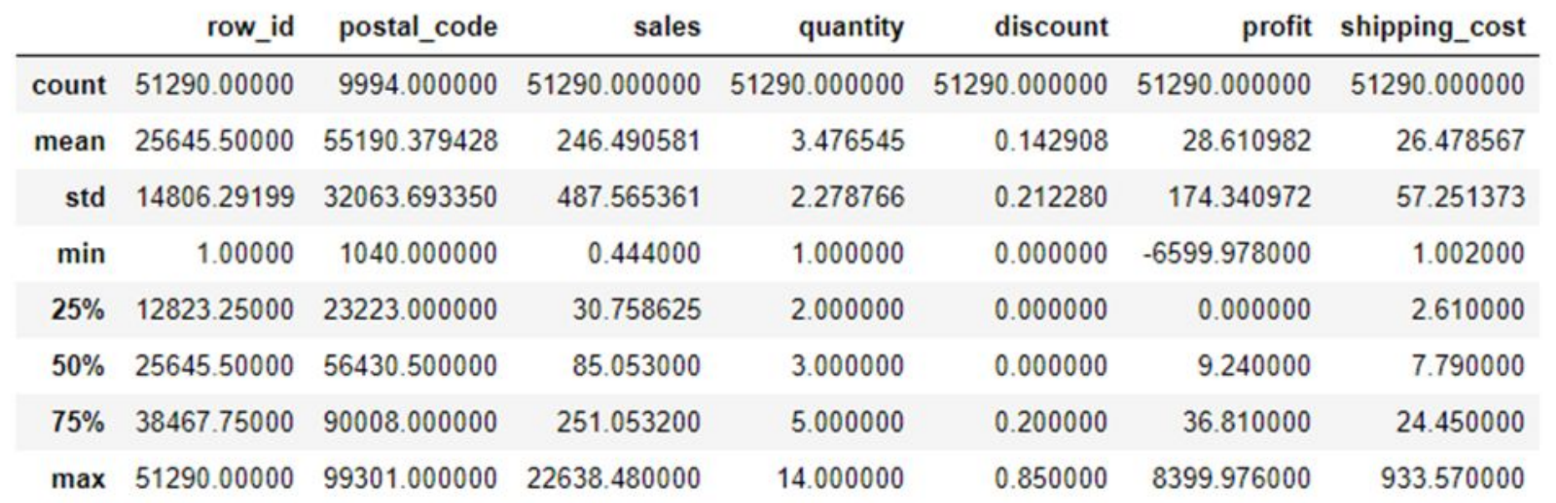
Data:

Word data - https://www.kaggle.com/paultimothymooney/latitude-and-longitude-for-every-country-and-state. The Global Superstore data can be found at: https://data.world/tableauhelp/superstore-data-sets. The data contains 24 columns and 51 290 rows.

The data has 17 415 unique customer ID so this means the some customers have bought more that one product whether at the same time or at different times. There is transactional data but there is no specific column for preference. This can be solved by assuming that the product a customer buys is what should be in their preference or add a product to someone's preference if they buy more than one unit of buy if multiple times.



```
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   row_id          51290 non-null  int64  
 1   order_id        51290 non-null  object 
 2   order_date      51290 non-null  object 
 3   ship_date       51290 non-null  object 
 4   ship_mode       51290 non-null  object 
 5   customer_id     51290 non-null  object 
 6   customer_name   51290 non-null  object 
 7   segment         51290 non-null  object 
 8   postal_code     9994 non-null   float64
 9   city            51290 non-null  object 
 10  state           51290 non-null  object 
 11  country         51290 non-null  object 
 12  region          51290 non-null  object 
 13  market          51290 non-null  object 
 14  product_id      51290 non-null  object 
 15  category        51290 non-null  object 
 16  sub_category    51290 non-null  object 
 17  product_name    51290 non-null  object 
 18  sales           51290 non-null  float64
 19  quantity        51290 non-null  int64  
 20  discount        51290 non-null  float64
 21  profit          51290 non-null  float64
 22  shipping_cost   51290 non-null  float64
 23  order_priority  51290 non-null  object 
dtypes: float64(5), int64(2), object(17)
```



	row_id	postal_code	sales	quantity	discount	profit	shipping_cost
count	51290.00000	9994.000000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000
mean	25645.50000	55190.379428	246.490581	3.476545	0.142908	28.610982	26.478567
std	14806.29199	32063.693350	487.565361	2.278766	0.212280	174.340972	57.251373
min	1.00000	1040.000000	0.444000	1.000000	0.000000	-6599.978000	1.002000
25%	12823.25000	23223.000000	30.758625	2.000000	0.000000	0.000000	2.610000
50%	25645.50000	56430.500000	85.053000	3.000000	0.000000	9.240000	7.790000
75%	38467.75000	90008.000000	251.053200	5.000000	0.200000	36.810000	24.450000
max	51290.00000	99301.000000	22638.480000	14.000000	0.850000	8399.976000	933.570000

METHODOLOGY

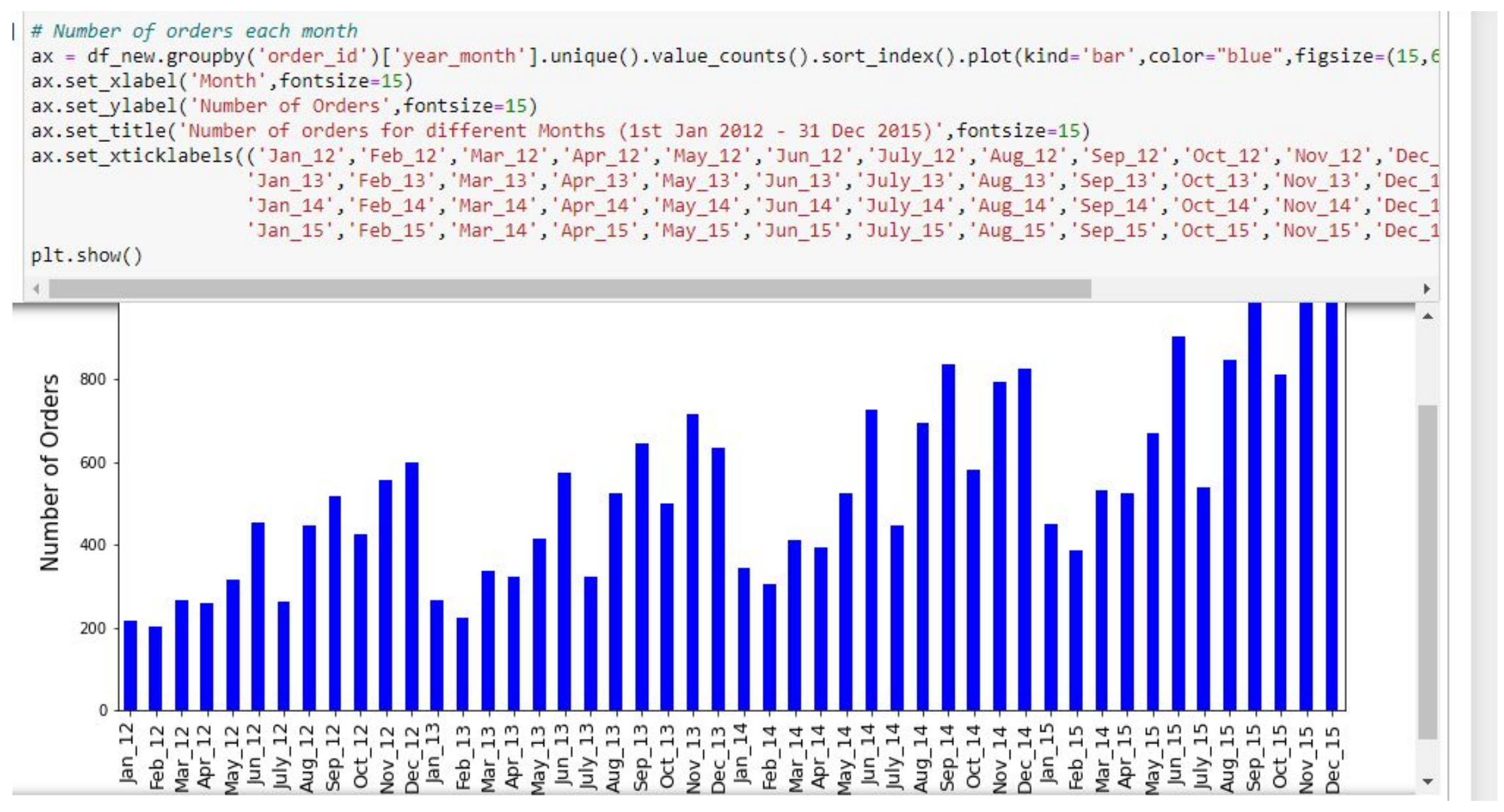
Firstly I did some data Exploration and visualization in python .

Then model building in python: 1. Apriori algorithm for category and product recommendation. 2. K-means for market segmentation

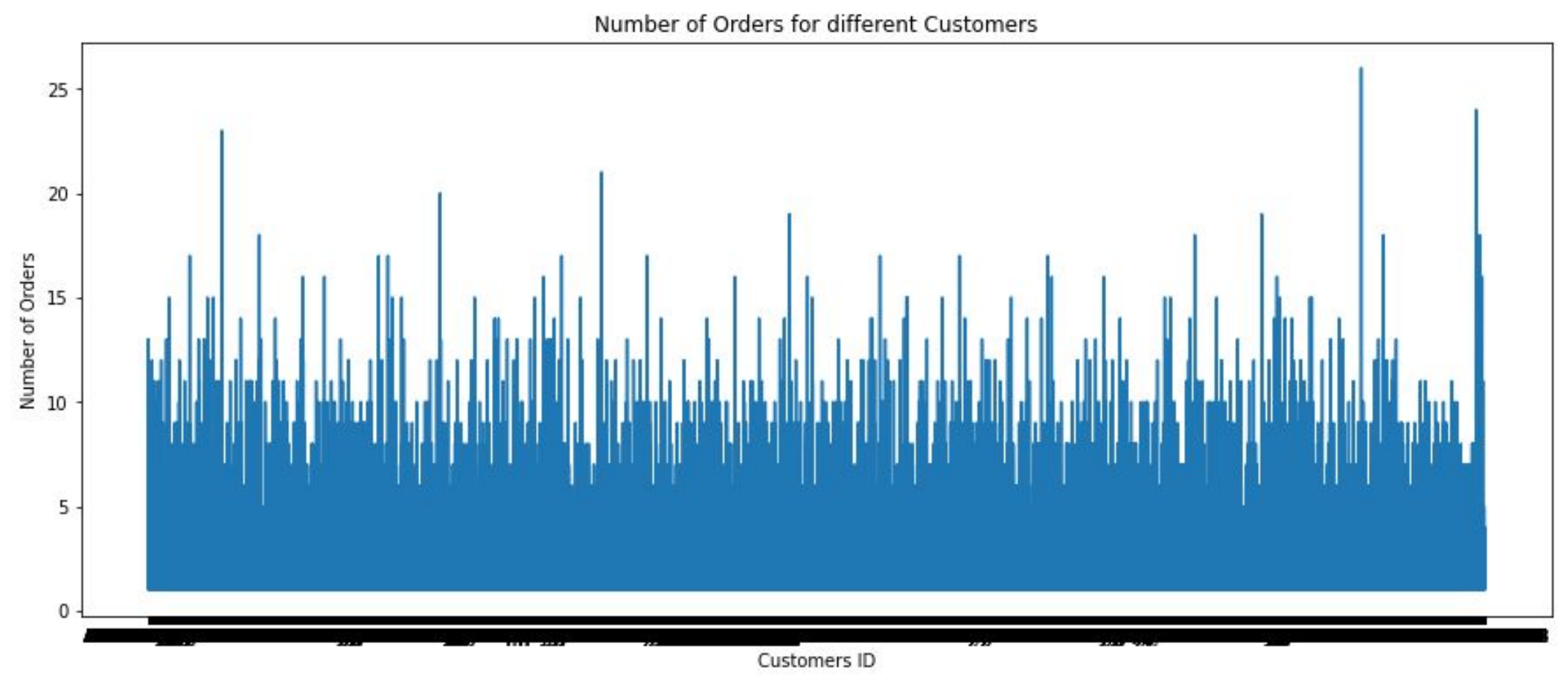
Then Build a dashboard in Tableau to present Customer purchasing data, sales, profit and locational data.

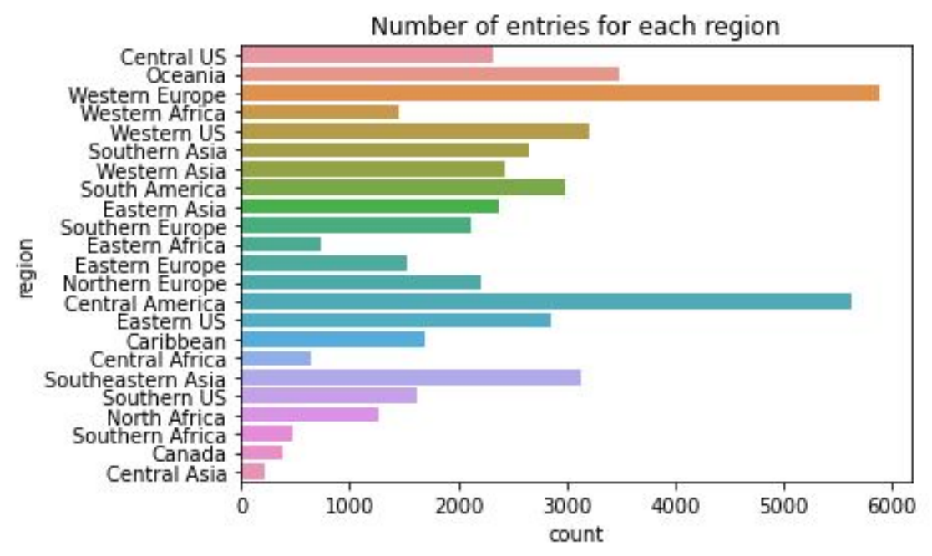
Then implement a model on R for the recommender system.

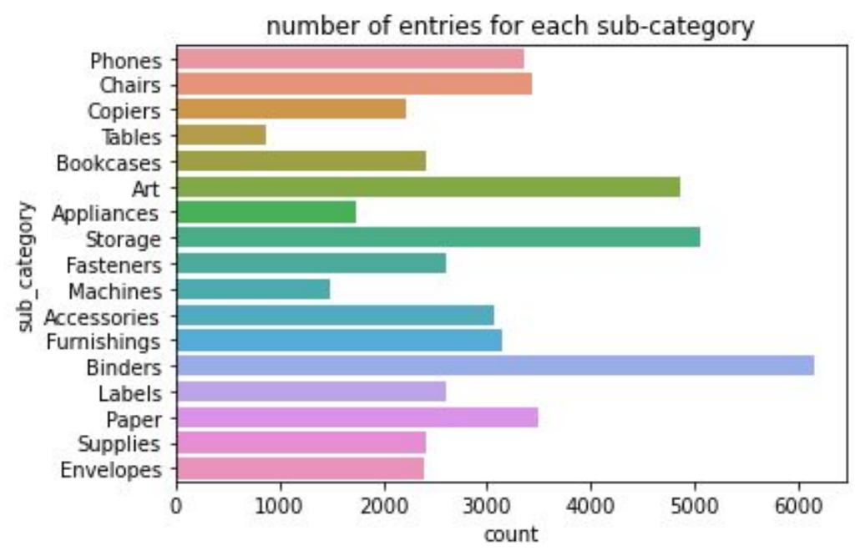
Data Exploration (python)

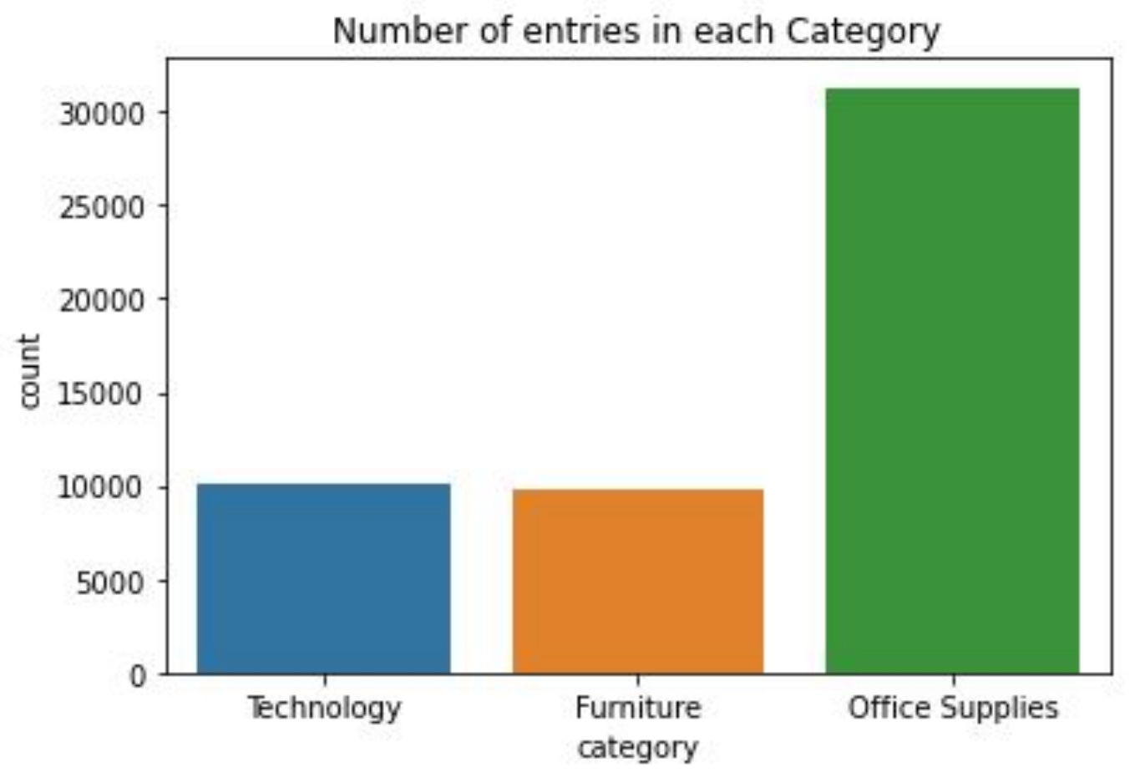


```
# Number of orders each month
ax = df_new.groupby('order_id')['year_month'].unique().value_counts().sort_index().plot(kind='bar',color="blue",figsize=(15,6
ax.set_xlabel('Month',fontsize=15)
ax.set_ylabel('Number of Orders',fontsize=15)
ax.set_title('Number of orders for different Months (1st Jan 2012 - 31 Dec 2015)',fontsize=15)
ax.set_xticklabels(('Jan_12','Feb_12','Mar_12','Apr_12','May_12','Jun_12','July_12','Aug_12','Sep_12','Oct_12','Nov_12','Dec_
                    'Jan_13','Feb_13','Mar_13','Apr_13','May_13','Jun_13','July_13','Aug_13','Sep_13','Oct_13','Nov_13','Dec_1
                    'Jan_14','Feb_14','Mar_14','Apr_14','May_14','Jun_14','July_14','Aug_14','Sep_14','Oct_14','Nov_14','Dec_1
                    'Jan_15','Feb_15','Mar_14','Apr_15','May_15','Jun_15','July_15','Aug_15','Sep_15','Oct_15','Nov_15','Dec_1
plt.show()
```



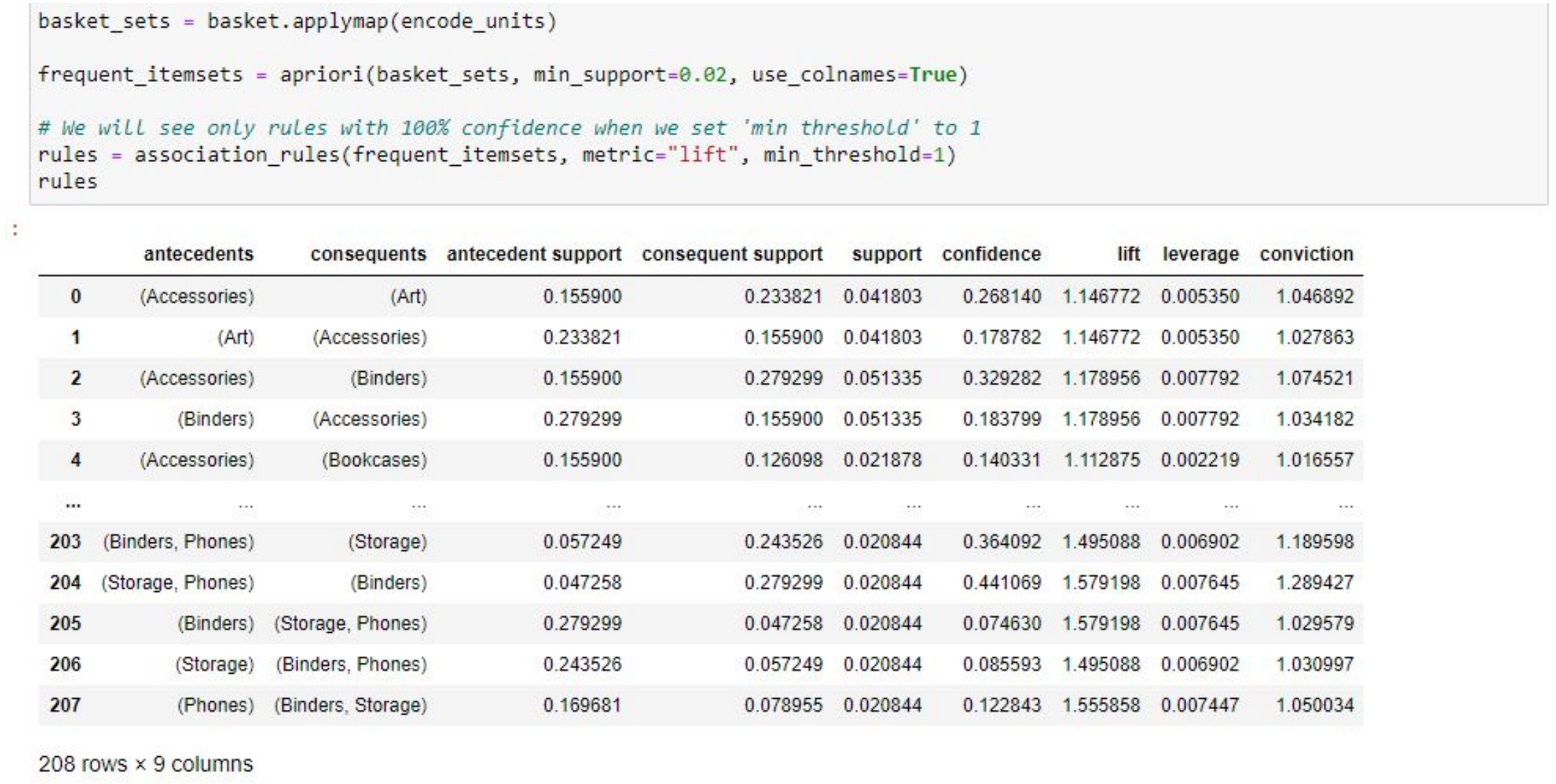






MODEL Building (python)

1. a. Apriori Algorithm for sub-category

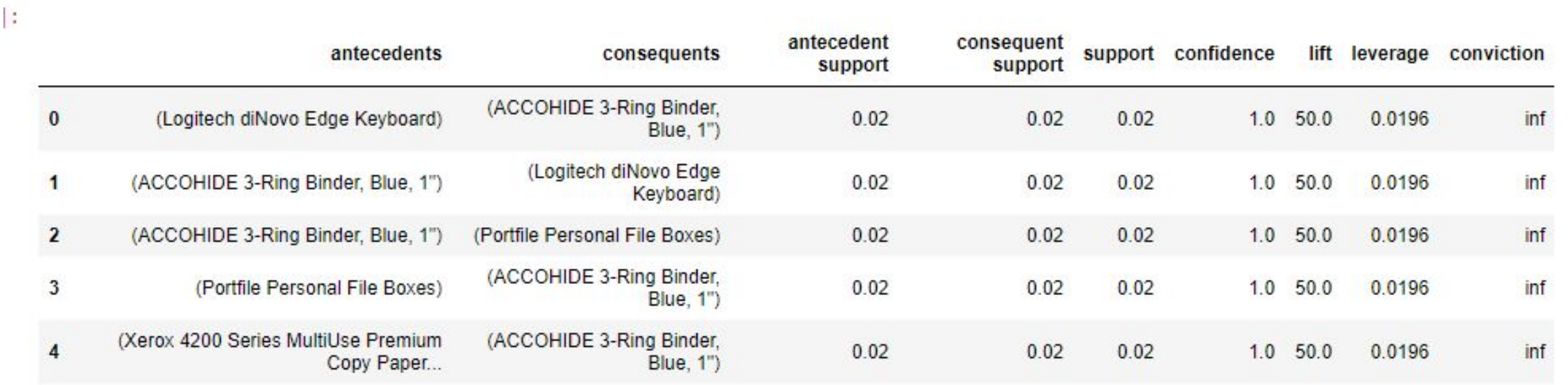


```
basket_sets = basket.applymap(encode_units)

frequent_itemsets = apriori(basket_sets, min_support=0.02, use_colnames=True)

# We will see only rules with 100% confidence when we set 'min threshold' to 1
rules = association_rules(frequent_itemsets, metric="lift", min_threshold=1)
rules
```

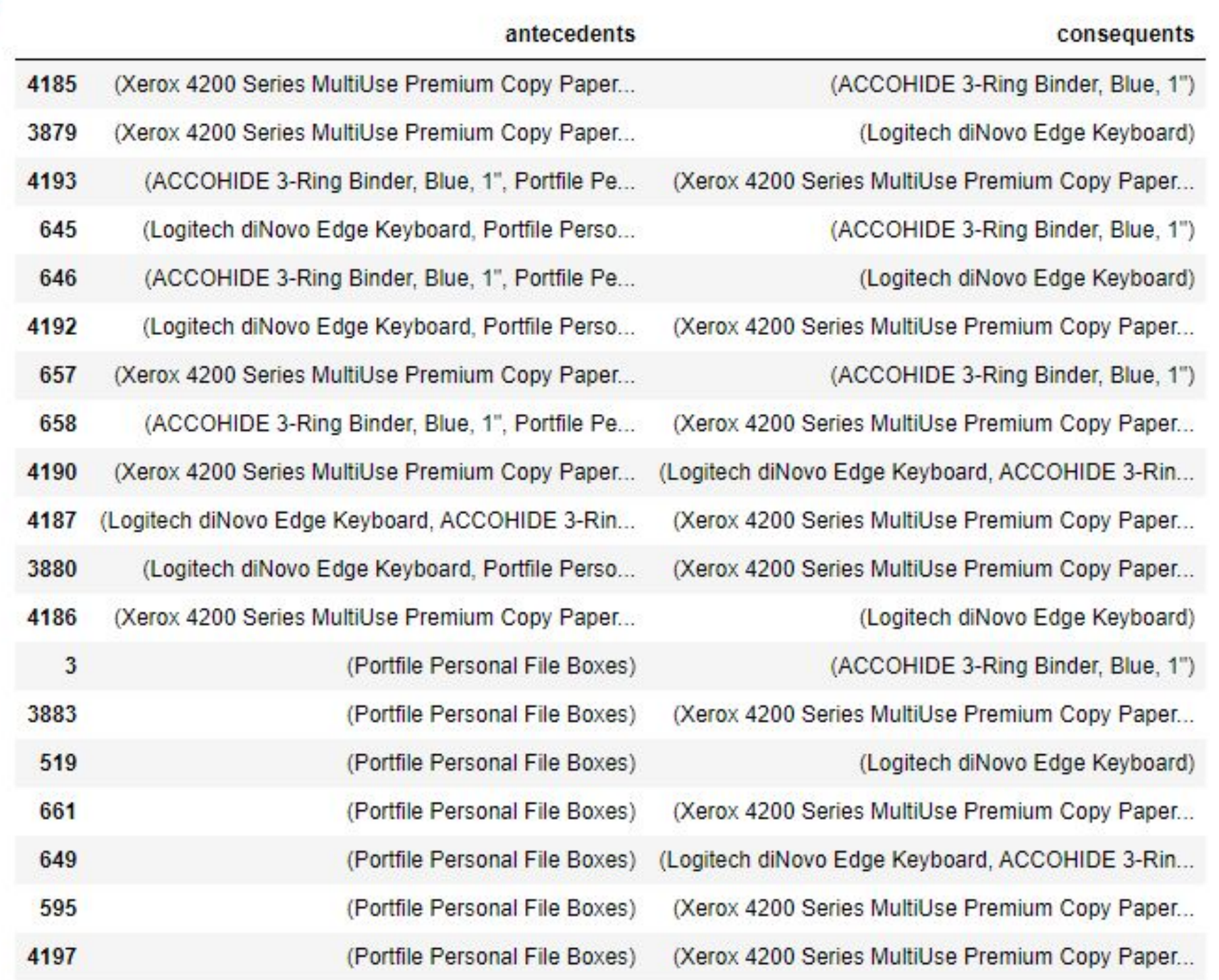
	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Accessories)	(Art)	0.155900	0.233821	0.041803	0.268140	1.146772	0.005350	1.046892
1	(Art)	(Accessories)	0.233821	0.155900	0.041803	0.178782	1.146772	0.005350	1.027863
2	(Accessories)	(Binders)	0.155900	0.279299	0.051335	0.329282	1.178956	0.007792	1.074521
3	(Binders)	(Accessories)	0.279299	0.155900	0.051335	0.183799	1.178956	0.007792	1.034182
4	(Accessories)	(Bookcases)	0.155900	0.126098	0.021878	0.140331	1.112875	0.002219	1.016557
...	...	...	...	...	...	...	...	...	...
203	(Binders, Phones)	(Storage)	0.057249	0.243526	0.020844	0.364092	1.495088	0.006902	1.189598
204	(Storage, Phones)	(Binders)	0.047258	0.279299	0.020844	0.441069	1.579198	0.007645	1.289427
205	(Binders)	(Storage, Phones)	0.279299	0.047258	0.020844	0.074630	1.579198	0.007645	1.029579
206	(Storage)	(Binders, Phones)	0.243526	0.057249	0.020844	0.085593	1.495088	0.006902	1.030997
207	(Phones)	(Binders, Storage)	0.169681	0.078955	0.020844	0.122843	1.555858	0.007447	1.050034

208 rows × 9 columns

1. b. Apriori Algorithm for each product sold

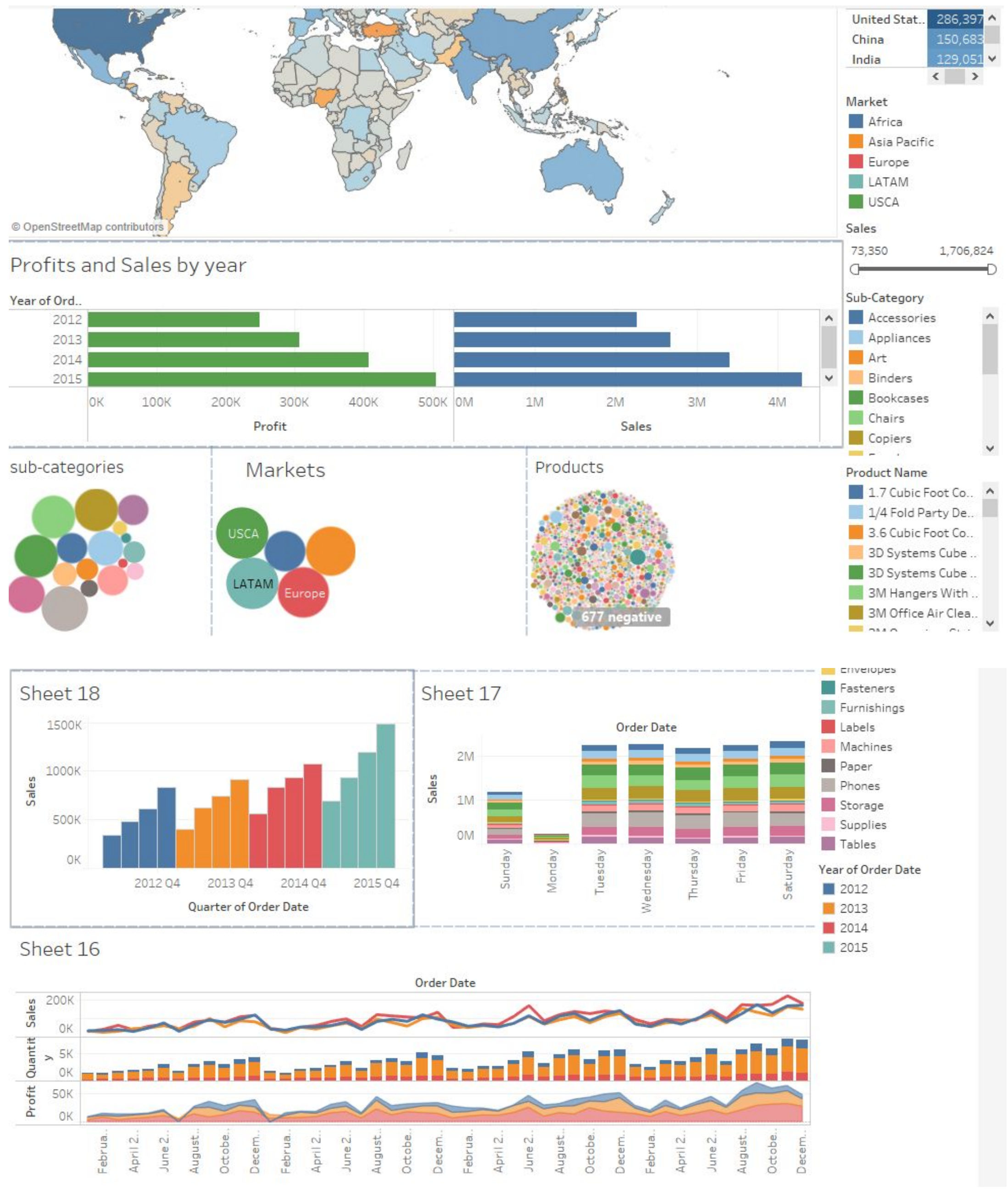


	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Logitech diNovo Edge Keyboard)	(ACCOHIDE 3-Ring Binder, Blue, 1")	0.02	0.02	0.02	1.0	50.0	0.0196	inf
1	(ACCOHIDE 3-Ring Binder, Blue, 1")	(Logitech diNovo Edge Keyboard)	0.02	0.02	0.02	1.0	50.0	0.0196	inf
2	(ACCOHIDE 3-Ring Binder, Blue, 1")	(Portfile Personal File Boxes)	0.02	0.02	0.02	1.0	50.0	0.0196	inf
3	(Portfile Personal File Boxes)	(ACCOHIDE 3-Ring Binder, Blue, 1")	0.02	0.02	0.02	1.0	50.0	0.0196	inf
4	(Xerox 4200 Series MultiUse Premium Copy Paper...	(ACCOHIDE 3-Ring Binder, Blue, 1")	0.02	0.02	0.02	1.0	50.0	0.0196	inf

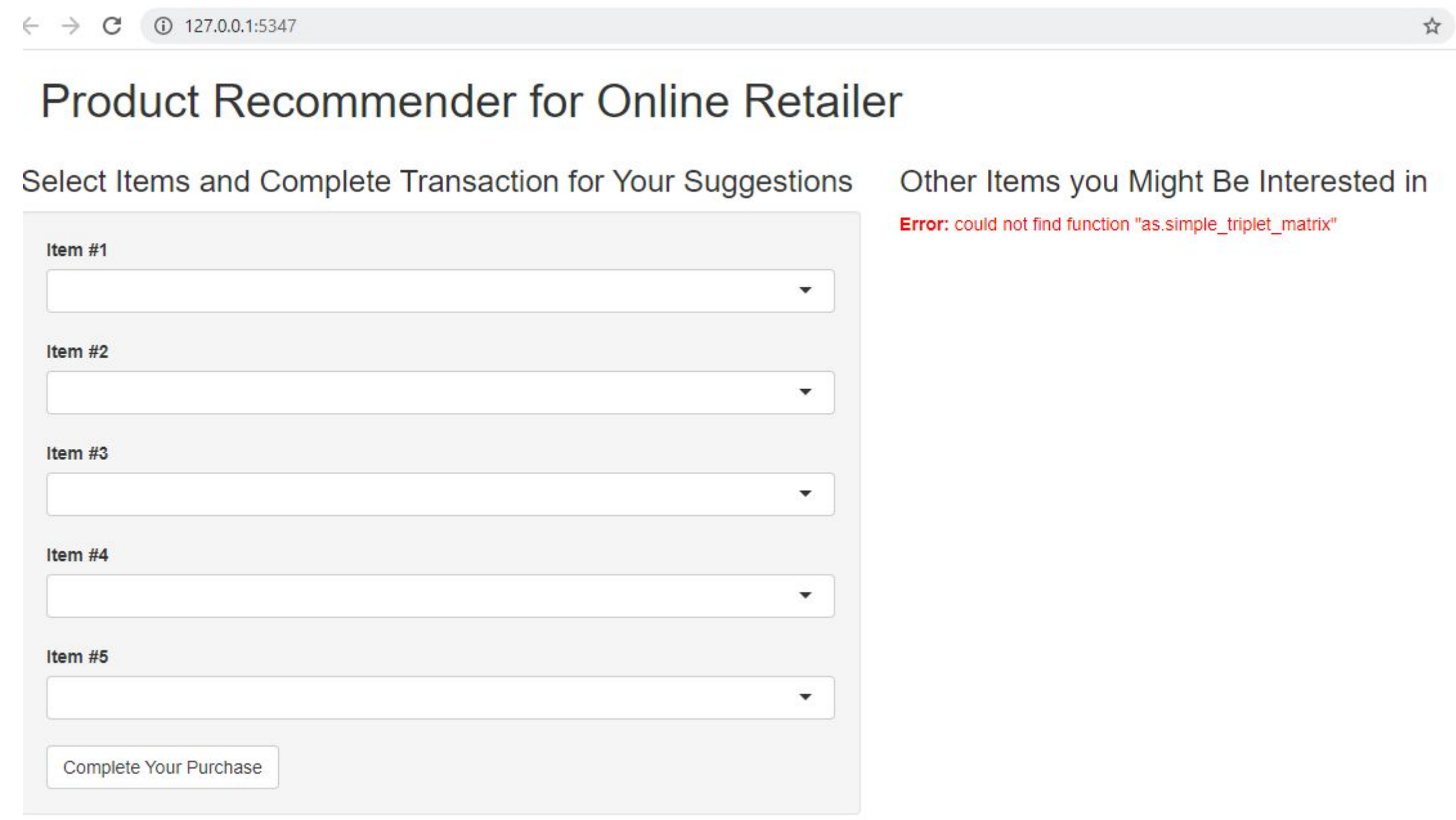


	antecedents	consequents
4185	(Xerox 4200 Series MultiUse Premium Copy Paper...	(ACCOHIDE 3-Ring Binder, Blue, 1")
3879	(Xerox 4200 Series MultiUse Premium Copy Paper...	(Logitech diNovo Edge Keyboard)
4193	(ACCOHIDE 3-Ring Binder, Blue, 1", Portfile Pe...	(Xerox 4200 Series MultiUse Premium Copy Paper...
645	(Logitech diNovo Edge Keyboard, Portfile Perso...	(ACCOHIDE 3-Ring Binder, Blue, 1")
646	(ACCOHIDE 3-Ring Binder, Blue, 1", Portfile Pe...	(Logitech diNovo Edge Keyboard)
4192	(Logitech diNovo Edge Keyboard, Portfile Perso...	(Xerox 4200 Series MultiUse Premium Copy Paper...
657	(Xerox 4200 Series MultiUse Premium Copy Paper...	(ACCOHIDE 3-Ring Binder, Blue, 1")
658	(ACCOHIDE 3-Ring Binder, Blue, 1", Portfile Pe...	(Xerox 4200 Series MultiUse Premium Copy Paper...
4190	(Xerox 4200 Series MultiUse Premium Copy Paper...	(Logitech diNovo Edge Keyboard, ACCOHIDE 3-Rin...
4187	(Logitech diNovo Edge Keyboard, ACCOHIDE 3-Rin...	(Xerox 4200 Series MultiUse Premium Copy Paper...
3880	(Logitech diNovo Edge Keyboard, Portfile Perso...	(Xerox 4200 Series MultiUse Premium Copy Paper...
4186	(Xerox 4200 Series MultiUse Premium Copy Paper...	(Logitech diNovo Edge Keyboard)
3	(Portfile Personal File Boxes)	(ACCOHIDE 3-Ring Binder, Blue, 1")
3883	(Portfile Personal File Boxes)	(Xerox 4200 Series MultiUse Premium Copy Paper...
519	(Portfile Personal File Boxes)	(Logitech diNovo Edge Keyboard)
661	(Portfile Personal File Boxes)	(Xerox 4200 Series MultiUse Premium Copy Paper...
649	(Portfile Personal File Boxes)	(Logitech diNovo Edge Keyboard, ACCOHIDE 3-Rin...
595	(Portfile Personal File Boxes)	(Xerox 4200 Series MultiUse Premium Copy Paper...
4197	(Portfile Personal File Boxes)	(Xerox 4200 Series MultiUse Premium Copy Paper...

Tableau Dashboards:



Implementation of model in R



127.0.0.1:5347

Product Recommender for Online Retailer

Select Items and Complete Transaction for Your Suggestions

Item #1

Item #2

Item #3

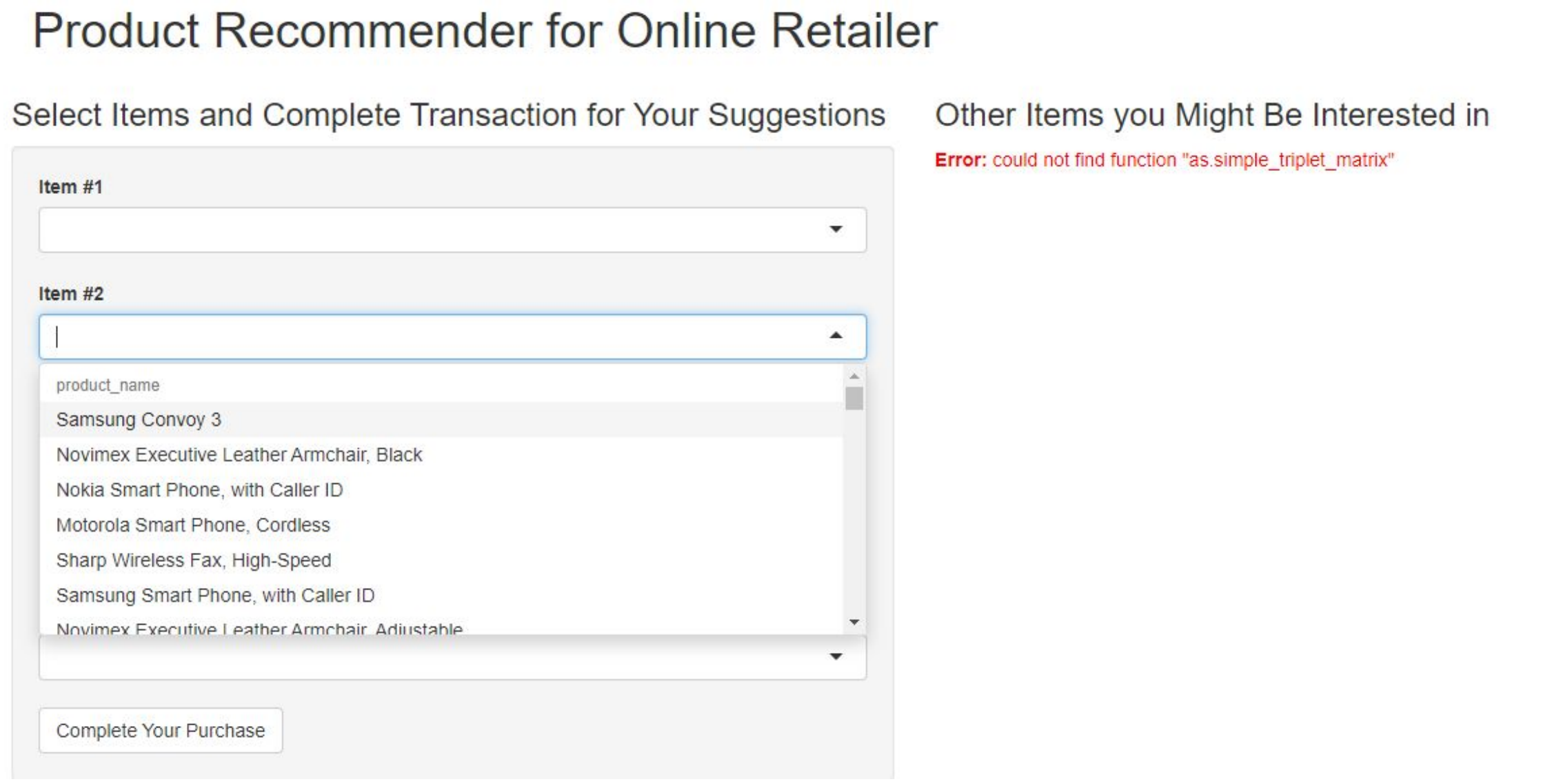
Item #4

Item #5

Complete Your Purchase

Other Items you Might Be Interested in

Error: could not find function "as.simple_triplet_matrix"



Product Recommender for Online Retailer

Select Items and Complete Transaction for Your Suggestions

Item #1

Item #2

product_name
Samsung Convoy 3
Novimex Executive Leather Armchair, Black
Nokia Smart Phone, with Caller ID
Motorola Smart Phone, Cordless
Sharp Wireless Fax, High-Speed
Samsung Smart Phone, with Caller ID
Novimex Executive Leather Armchair, Adjustable

Complete Your Purchase

Other Items you Might Be Interested in

Error: could not find function "as.simple_triplet_matrix"