Predicting Personalized Target Points Offers

Contents

Executive Summary

This report was commissioned to examine ways to improve Loblaw targets personalized offering to customers at the right time to make Loblaw more relevant when customers are making their shopping decision. The method used for this analysis are Data exploration, Imputations, Outlier's detection, Visualization and Modeling. Codes can be found in the appendix of this report. The model was based on our most loyal group of customers (Recent_LastTwoWeeks), the LR model's accuracy score is 69% which is good. The classification report shows a precision with 73%, recall with 90% and f1-score with 81% of correctly identified user who will return to shop in 2 weeks

By using this model, Loblaw would be able to target our most loyal customers with relevant personalized offering just in time when customers are planning their grocery shop. We recommend the upselling of Organics (meat, seafood, deli and dry grocery) and Plant based products in the stores, as the model shows that our loyal customers are buying Organic Produces. Our recommendation is back with the increasing trend of healthy living due to the pandemic.

Problem Statement

Loblaw is one of the key players offering grocery loyalty program, the PC point offers are not timely or representative of customer buying pattern.

The ability to predict a customer shopping pattern and customized a point offer to materialize just in time when the customer will be getting ready for a grocery shop, would lead to an overall lift in stores sales and increase Loblaw's market share.

Our goal is to have a sales lift of 2%.

The division leads (discount- Superstore/ No frills/ Market – Loblaw/Zehrs/Fortino) would be the sponsors of this project because they are responsible for the financial heath of the stores and promotions. Digital team are responsible for ensure that the target offers are uploaded on time while the merchants ensure they make the right deals with the vendors. The flyer team would use the information when creating a member only promotion in the flyer. Allocation and supply chain would ensure that stock is allocated, brought and shipped out on time to stores.

The ability to understand customers shopping pattern to better provide relevant point offerings is very important in helping the merchant and the flyer team in their planning sessions.

This will be an agile project which would give us the flexibility to implement, test and make changes as we go to ensure we have a prefect solution for this opportunity.

Most customer forget to scan their loyalty card during check out which makes it difficult for the system to calculate when the item was bought by the customer.

.

Research Methodology & Ethics

PC Opinium offers customers the ability to save and collect point on frequent purchased items, these points are redeemable for groceries (10,000 points = $10). Target offers such as a spend stretch (where a customer's weekly spend is $100 but they are encouraged to spend $150 and receive 20,000 in points), points on new items or departments are to increase a customer basket size. For Loblaw repeat customers means increase revenue and protecting market share, while for the customers it means that they are appreciated and valued.

According to Wright, Claire; Sparks and Leigh, customers are actively seeking an involving relationship with their preferred grocery stores. Snice the grocery store is believed to offer psychological reassurance of risk attached to purchasing an item and create a sense of belonging. Customers put their trust in Loblaw when they make their shopping plans, it is up to Loblaw to ensure that their trust is not misplaced. Ensuring the grocery items sold are of superior quality and in turn thanking the customers for choosing to shop with them by rewarding customers with timely and relevant point offers.

Another aspect that Loblaw need to consider is price discrimination when offering target offers or member's only pricing to PC Opinium members. With the amount of information that PC loyalty members provides when shopping, Loblaw is able to use that information ethically to learn and understand their members shopping behaviours.

According to Smith, Rimler, data mining is regarded as an ethically practice since it provides information about the buyers' behaviour to the business. Which the business incorporates into its strategic decision to recover a prescription pricing strategy with respect to price discrimination which can be considered to be "unethical". The business might try to steal market share by adjusting its pricing strategy or by sacrificing market share in return for higher margins. The strategic decision then becomes an ethical dilemma, whether or not it is considered a violation of the governing competition policy. Although the spirit of our antitrust laws would regard such behavior as a violation, it may not be technically so. Therein lies the ethical implications of using data mining to extract information about consumer behavior.

Exploratory Data Analysis

Three datasets were merged to derived the key column for my analysis. These features are important to my report because it provide a timeline of the customers shopping process which will help drive the understanding of customers shopping pattern toward predicting the next shop day and personalized promotional points offerings. Refer to appendix figure 1 for codes.

The dataset produces a weak correlation, days_since_prior _order has a correlation with order_number. Order_number also has a weak correlation to both order_dow and order_hour_of_day. Refer to appendix figure 2 for chart.

Imputation

Days_since_piror_order has missing values which I will be making an assumption that they are missing because these are new PC Optimum customers who are using their card for the first time. We have about 20875 missing values which would be filled with a value of zero. Refer to appendix figure 3 for chart and code.

Outlier Detection

Three column shows outlier in the visual below but these are not true outliers, I will be keeping the outliers in the data set. For order_number, a customer can have as many orders as they can afford. Stores are opened 24hrs through the help of online ordering (PC Express portal) and customers (cherry picker) who visit ours stores when there is an attractive promotional will have a longer prior shopping day.

Order_hour_of_day shows the store key busy times are between 9am and 5pm, this would provide stores colleagues an indication on where to invest labour hours to ensure we keep our customers happy.

Order_number shows a high frequency of order between 1 and 20 orders, then it begins to falls. Order_dow shows most of the order are placed or made during Sunday/Monday which helps in the allocation and shipment of inventory from the DCs to the stores. In ONT, the break of a new promotional week is on Thursday, which means that shipment should arrive store on Tuesday or Wednesday to give store colleagues enough time to stock up the shelves.

It also provides an indicator to stores/online when colleagues are mostly needed.

Days_since_piror_order graph provides a view on who are our loyal customers, which would be those who have returned to shop with us in the past two weeks. Refer to appendix figure 4 for chart.

Feature Engineering

In order to gain a better understanding of our loyal customers to target value PC point offering to those groups. I have extracted 3 new columns from day_since_prior_order.

Recent_LastTwoWeeks- This will be my most loyal customers, who think of shopping with Loblaw when they run or of groceries or about to run out.

Recent_LastFourWeeks- With current state of COVID-19, many customers have changed their buying pattern. I will be making the assumption that it is the case for this group.

Recent_LaggardMoreThanFourWeeks- This are the cherry-picking customers, who are in the store for the promotional item or have received a point offer on an item.

I have extracted five columns from Order_hour_of_day to understand key busy shopping times for customers (Early morning, Morning, Lunch, Afternoon and Evening). This information will help in scheduling of labour hours to ensure we have coverage for the key busy shopping times.

I have extracted Seven columns from Order_dow to understand key busy shopping days for customers which would help the stores to plan and ensure the shelves are stocked. It would help in the planning/allocation and shipment of good from the DC to the stores.

Dummies are applied to the dataset to make it easy to quantify the data, duplicate columns some of which were derived after feature engineering are dropped from dataset. Order_number is removed because it has a strong correlation to the response variable (day_since_prior_order). The product_name was removed because it does not have any value to the dataset since product_id is indexing the data. Refer to appendix figure 5 for code.

Modelling

In the creation of my model for prediction, I looked at various models that would give me the best accuracy. First, I applied cross validation to estimate how the model would perform and based on the figure below this dataset is consistent and would be able to trained a dataset that it has not seen. I picked LR because it will answer all the probability questions ("Would the customer buy this item or return back to shop again') returning a 0 or 1.

I looked at the KNN and Dtree but the model accuracy of LR was higher at 69%. The LR model also produce a better confusion matrix. Refer to appendix figure 6 for chart.

- True positive – The model correctly predicted that 1267 customers will shop every two weeks.
- True negative- The model correctly predicated that 61 customers will not shop every two weeks.
- False Positive- The model incorrectly predicted that 457 customers will shop in two weeks.
- False Negative- The model incorrectly predicted that 139 customers will not shop in two weeks.

Classification report show a precision with 73%, recall with 90% and F1-score with 81% of correctly identified users who will return to shop in 2weeeks.

For this business problem, predictive modelling is the best solution since we are interested in the future activities of the customer. PC Optimum customer provide a lot of information to the business every time their loyalty card is swiped during checkout. These information become liquid gold and gives access into the brains of our customers which help Loblaw plan target offers.

Which would enable the business to effectively sent out these offers in a timely manner and offer suggestions based on complimentary items or try to upsell certain items to the customer.

According to the express analytics blog, there are five area where predicative modelling is effective in the retail industry. They are promotions, shopper targeting, marketing campaign management, pricing and inventory management. The ability to test a promotion or marketing plan with the help of predicative analysis before rolling it out. Gone are the days of waiting for the mailman to drop off the flyer for the new week's promotions, customers have easy access to the flyers on the smart devices.

In order to protect market share, Loblaw has to proactive in recognising and actively reacting to changes in the market place. Predicative modelling makes it possible to stay active and relevant to the customers.

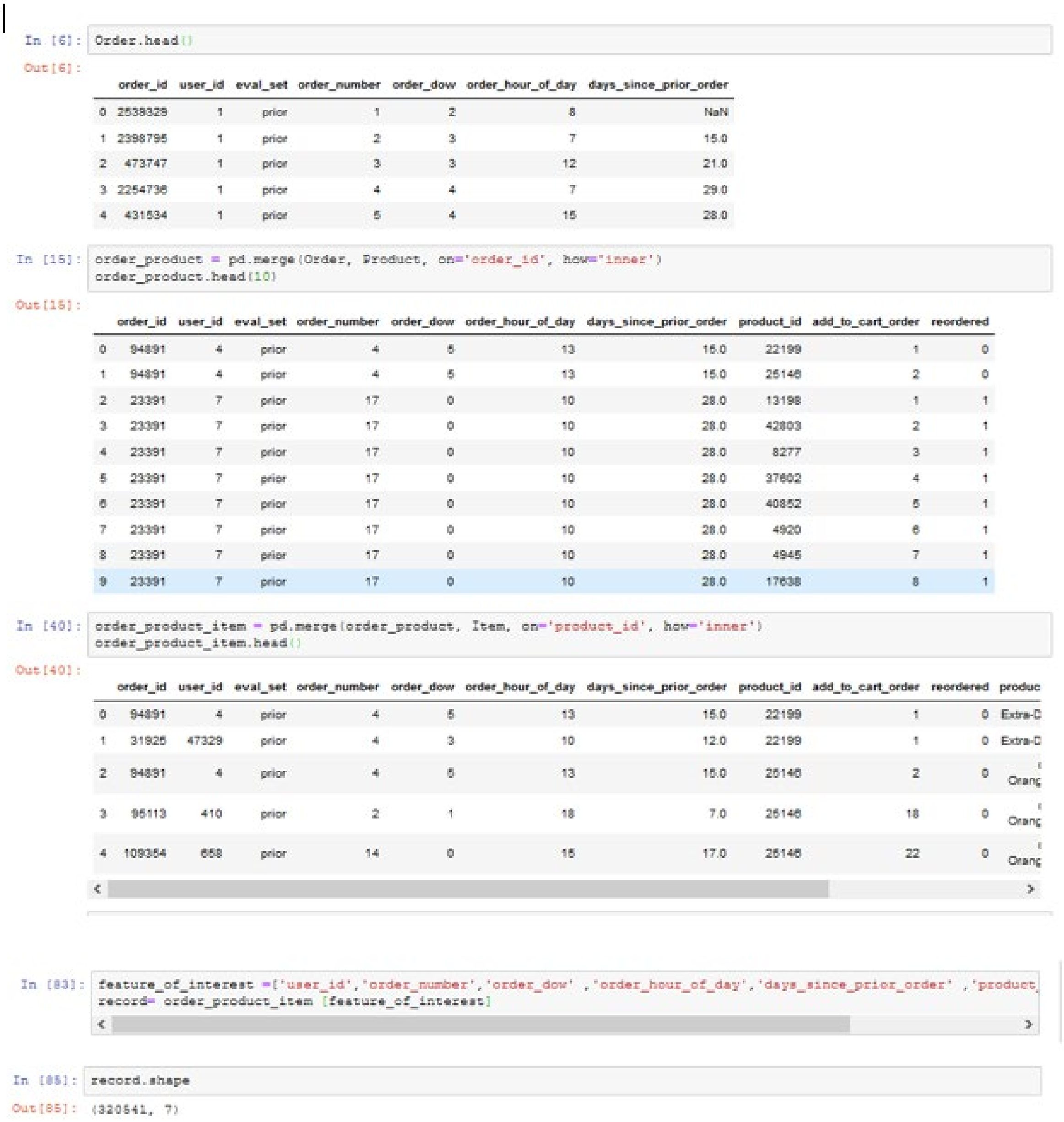
Recommendation

We are recommending offering target offers to upsell Organics (meat, seafood, deli and dry grocery) and Plant based products in the stores to our loyal customers since they are already buying organic produces. This offer would be very relevant to this group of customers, who based on their buying pattern show they are health conscious. Refer to appendix figure 7 for chart and link below for visualization.

https://public.tableau.com/profile/ekefon#!/vizhome/assignment4_up/Dashboard2?publish=yes

Appendix

Figure 1



```
In [6]: Order.head()
```

Out[6]:

	order_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	days_since_prior_order
0	2539329	1	prior	1	2	8	NaN
1	2398795	1	prior	2	3	7	15.0
2	473747	1	prior	3	3	12	21.0
3	2254736	1	prior	4	4	7	29.0
4	431534	1	prior	5	4	15	28.0

```
In [15]: order_product = pd.merge(Order, Product, on='order_id', how='inner')
         order_product.head(10)
```

Out[15]:

	order_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	days_since_prior_order	product_id	add_to_cart_order	reordered
0	94891	4	prior	4	5	13	15.0	22199	1	0
1	94891	4	prior	4	5	13	15.0	25146	2	0
2	23391	7	prior	17	0	10	28.0	13198	1	1
3	23391	7	prior	17	0	10	28.0	42803	2	1
4	23391	7	prior	17	0	10	28.0	8277	3	1
5	23391	7	prior	17	0	10	28.0	37602	4	1
6	23391	7	prior	17	0	10	28.0	40852	5	1
7	23391	7	prior	17	0	10	28.0	4920	6	1
8	23391	7	prior	17	0	10	28.0	4945	7	1
9	23391	7	prior	17	0	10	28.0	17638	8	1

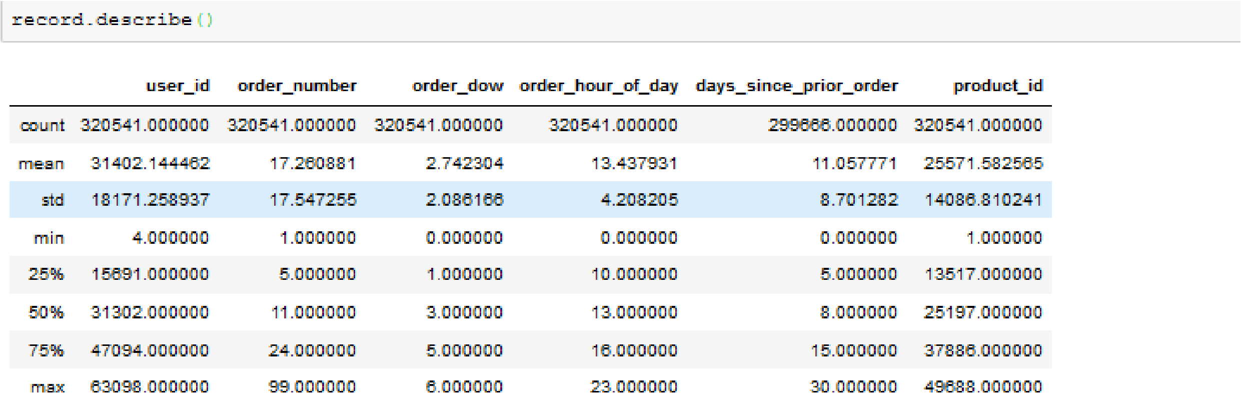
```
In [40]: order_product_item = pd.merge(order_product, Item, on='product_id', how='inner')
         order_product_item.head()
```

Out[40]:

	order_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	days_since_prior_order	product_id	add_to_cart_order	reordered	produc
0	94891	4	prior	4	5	13	15.0	22199	1	0	Extra-D
1	31925	47329	prior	4	3	10	12.0	22199	1	0	Extra-D
2	94891	4	prior	4	5	13	15.0	25146	2	0	Orang
3	95113	410	prior	2	1	18	7.0	25146	18	0	Orang
4	109354	658	prior	14	0	15	17.0	25146	22	0	Orang

```
In [83]: feature_of_interest =['user_id','order_number','order_dow' ,'order_hour_of_day','days_since_prior_order' ,'product_
         record= order_product_item [feature_of_interest]
```

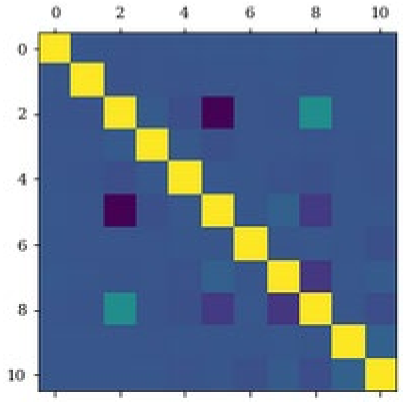
```
In [85]: record.shape
Out[85]: (320541, 7)
```

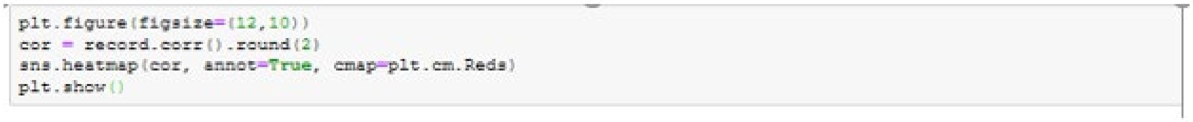


```
record.describe()
```

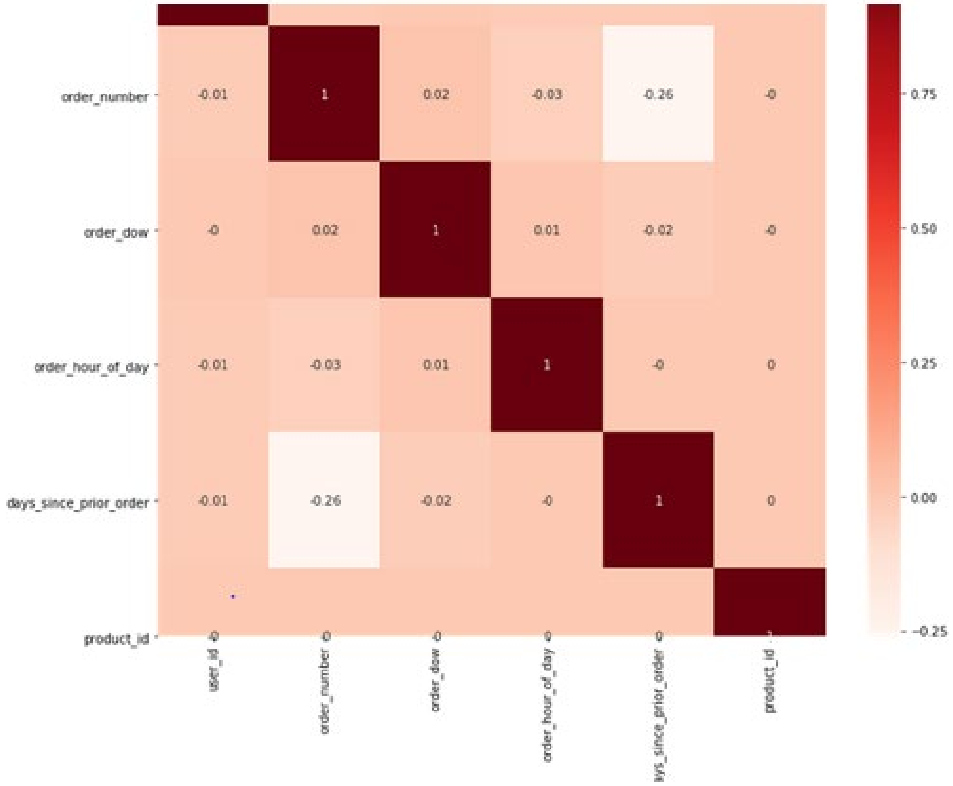
	user_id	order_number	order_dow	order_hour_of_day	days_since_prior_order	product_id
count	320541.000000	320541.000000	320541.000000	320541.000000	299666.000000	320541.000000
mean	31402.144462	17.260881	2.742304	13.437931	11.057771	25571.582565
std	18171.258937	17.547255	2.086166	4.208205	8.701282	14086.810241
min	4.000000	1.000000	0.000000	0.000000	0.000000	1.000000
25%	15691.000000	5.000000	1.000000	10.000000	5.000000	13517.000000
50%	31302.000000	11.000000	3.000000	13.000000	8.000000	25197.000000
75%	47094.000000	24.000000	5.000000	16.000000	15.000000	37886.000000
max	63098.000000	99.000000	6.000000	23.000000	30.000000	49688.000000

Figure 2





```
plt.figure(figsize=(12,10))
cor = record.corr().round(2)
sns.heatmap(cor, annot=True, cmap=plt.cm.Reds)
plt.show()
```



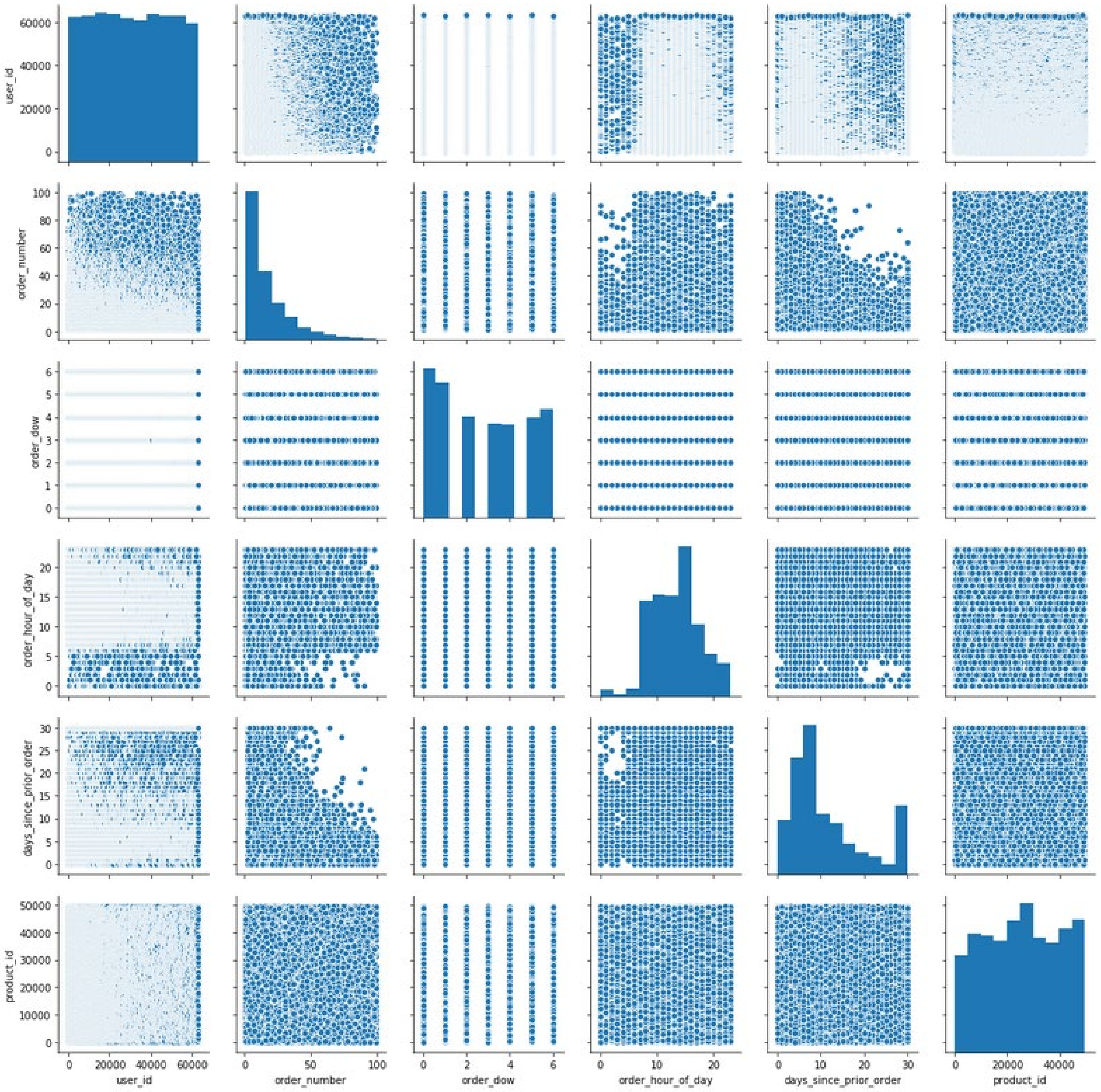
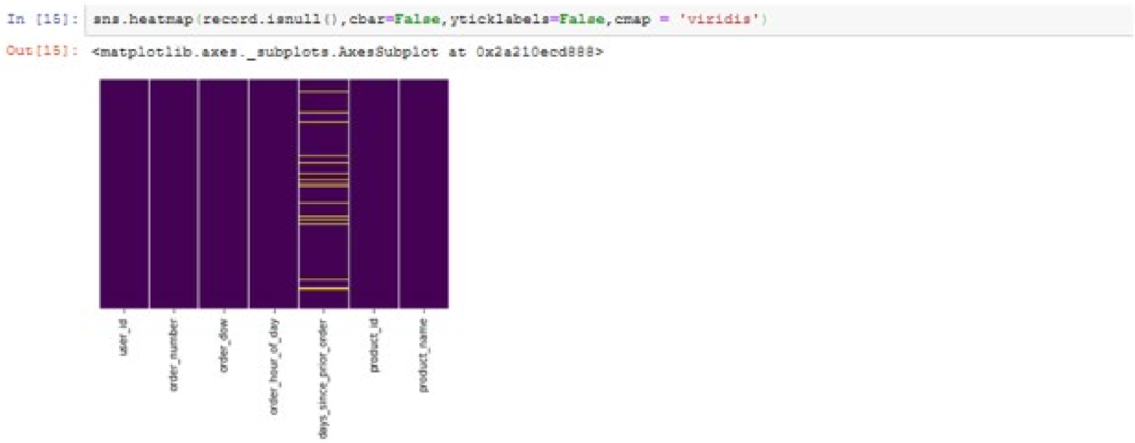
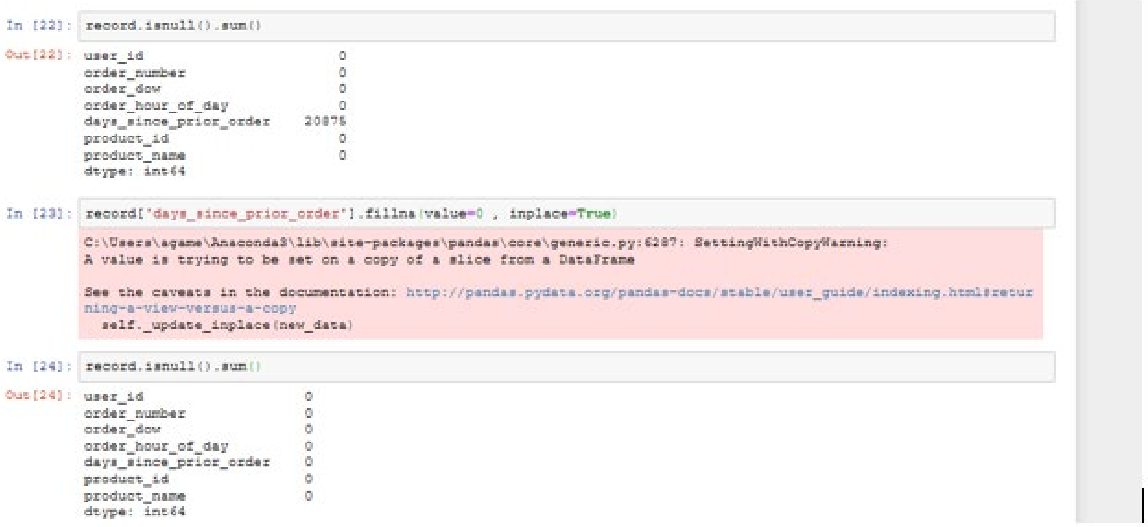

user_id
60000
40000
20000
0
order_number
100
80
60
40
20
0
order_dow
6
5
4
3
2
1
0
order_hour_of_day
20
15
10
5
0
days_since_prior_order
30
25
20
15
10
5
0
product_id
50000
40000
30000
20000
10000
0
0
20000
40000
60000
user_id
0
50
100
order_number
0
2
4
6
order_dow
0
10
20
order_hour_of_day
0
10
20
30
days_since_prior_order
0
20000
40000
product_id

Figure 3

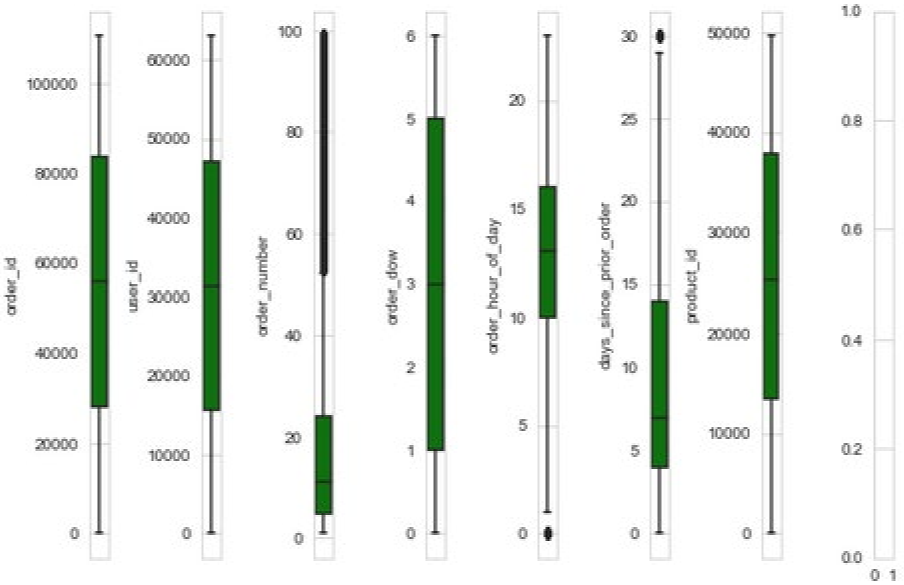


```
In [15]: sns.heatmap(record.isnull(),cbar=False,yticklabels=False,cmap = 'viridis')

Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x2a210ecd888>
```

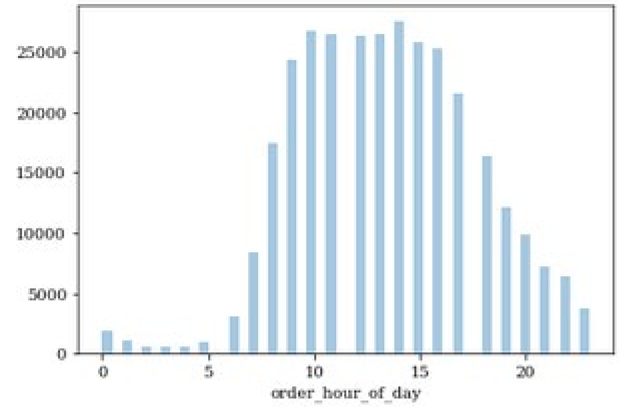


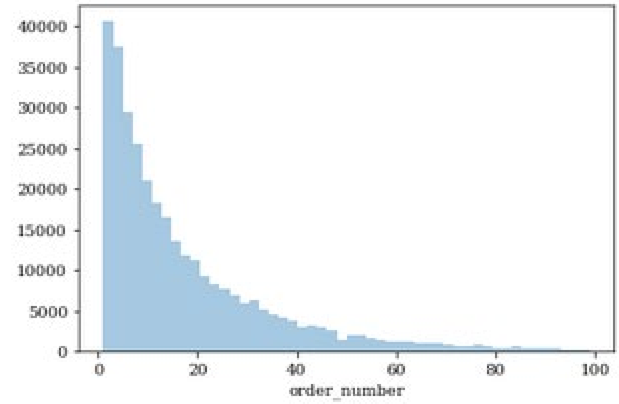
```
In [22]: record.isnull().sum()

Out[22]: user_id                       0
         order_number                  0
         order_dow                     0
         order_hour_of_day             0
         days_since_prior_order    20875
         product_id                    0
         product_name                  0
         dtype: int64

In [23]: record['days_since_prior_order'].fillna(value=0 , inplace=True)

         C:\Users\agame\Anaconda3\lib\site-packages\pandas\core\generic.py:6287: SettingWithCopyWarning:
         A value is trying to be set on a copy of a slice from a DataFrame

         See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#retur
         ning-a-view-versus-a-copy
           self._update_inplace(new_data)

In [24]: record.isnull().sum()

Out[24]: user_id                   0
         order_number              0
         order_dow                 0
         order_hour_of_day         0
         days_since_prior_order    0
         product_id                0
         product_name              0
         dtype: int64
```

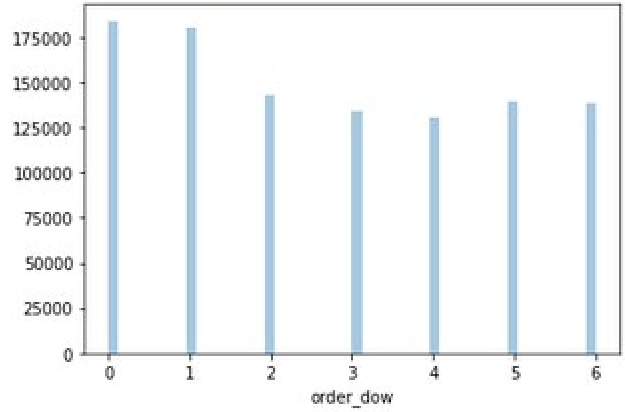
Figure 4



order_id
0
20000
40000
60000
80000
100000
user_id
0
10000
20000
30000
40000
50000
60000
order_number
0
20
40
60
80
100
order_dow
0
1
2
3
4
5
6
order_hour_of_day
0
5
10
15
20
days_since_prior_order
0
5
10
15
20
25
30
product_id
0
10000
20000
30000
40000
50000
0.0
0.2
0.4
0.6
0.8
1.0
0
1


25000
20000
15000
10000
5000
0
0
5
10
15
20
order_hour_of_day


40000
35000
30000
25000
20000
15000
10000
5000
0
0
20
40
60
80
100
order_number


175000
150000
125000
100000
75000
50000
25000
0
0
1
2
3
4
5
6
order_dow

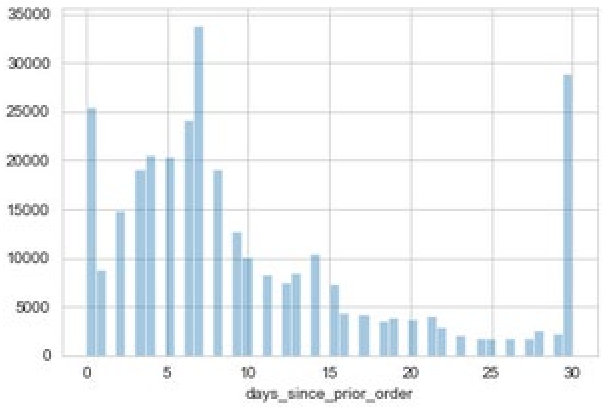
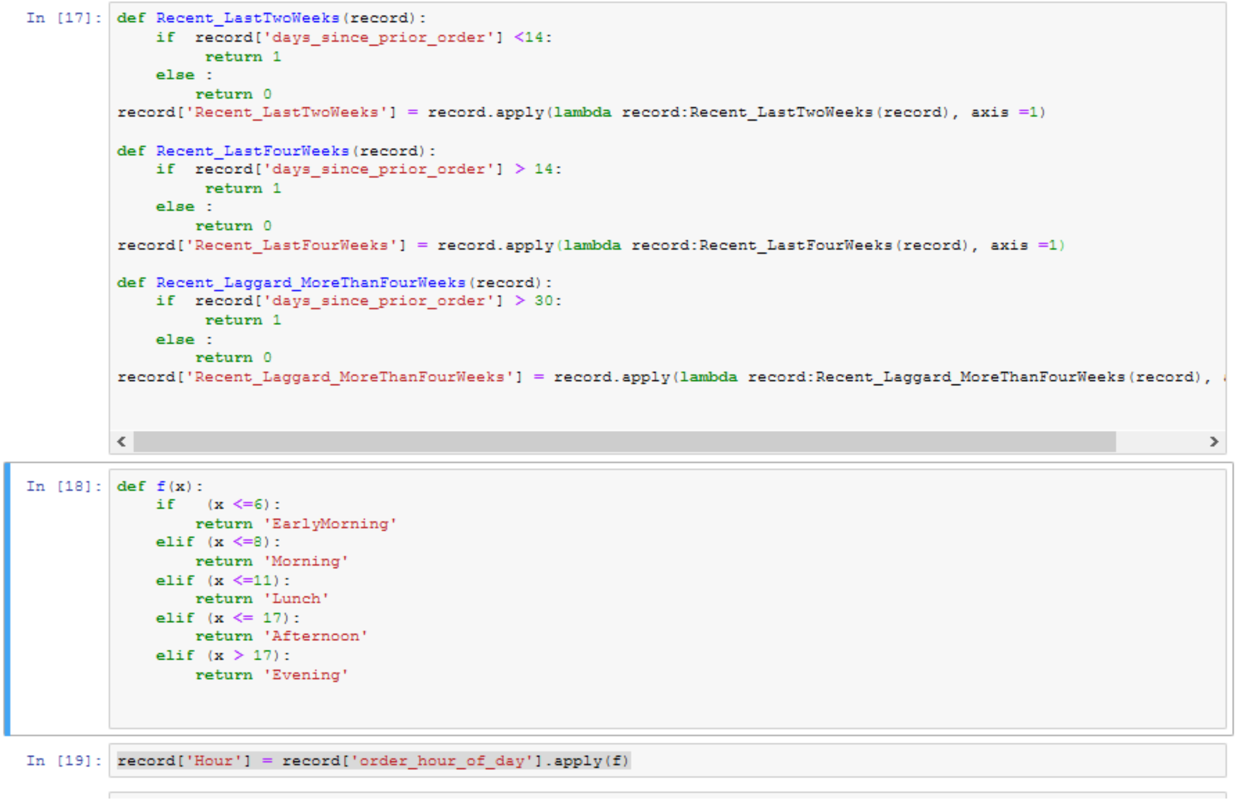


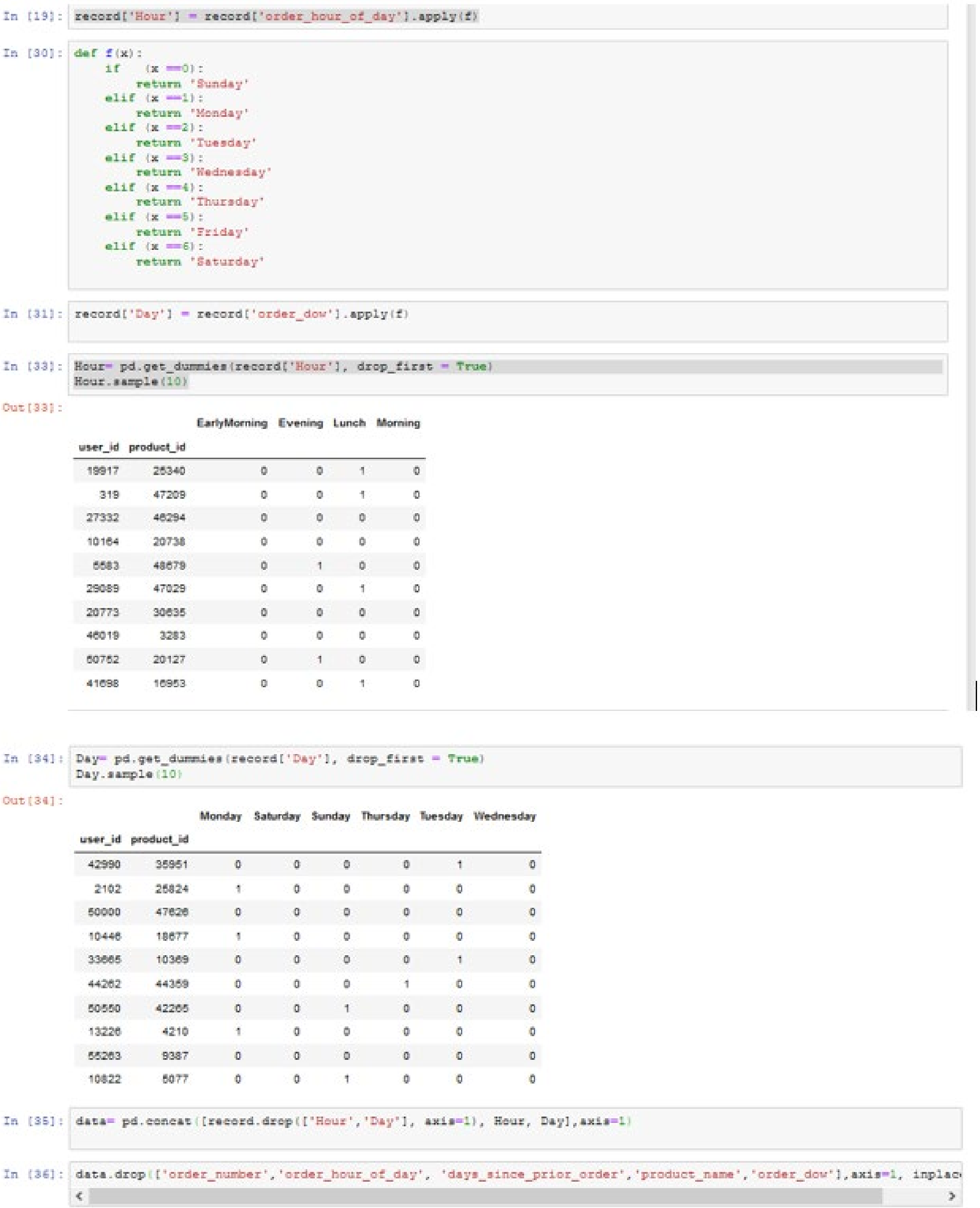
Figure 5



```
In [17]: def Recent_LastTwoWeeks(record):
             if  record['days_since_prior_order'] <14:
                  return 1
             else :
                 return 0
         record['Recent_LastTwoWeeks'] = record.apply(lambda record:Recent_LastTwoWeeks(record), axis =1)

         def Recent_LastFourWeeks(record):
             if  record['days_since_prior_order'] > 14:
                  return 1
             else :
                 return 0
         record['Recent_LastFourWeeks'] = record.apply(lambda record:Recent_LastFourWeeks(record), axis =1)

         def Recent_Laggard_MoreThanFourWeeks(record):
             if  record['days_since_prior_order'] > 30:
                  return 1
             else :
                 return 0
         record['Recent_Laggard_MoreThanFourWeeks'] = record.apply(lambda record:Recent_Laggard_MoreThanFourWeeks(record),
```

```
In [18]: def f(x):
             if   (x <=6):
                 return 'EarlyMorning'
             elif (x <=8):
                 return 'Morning'
             elif (x <=11):
                 return 'Lunch'
             elif (x <= 17):
                 return 'Afternoon'
             elif (x > 17):
                 return 'Evening'
```

```
In [19]: record['Hour'] = record['order_hour_of_day'].apply(f)
```



```
In [19]: record['Hour'] = record['order_hour_of_day'].apply(f)
```

```
In [30]: def f(x):
             if   (x ==0):
                 return 'Sunday'
             elif (x ==1):
                 return 'Monday'
             elif (x ==2):
                 return 'Tuesday'
             elif (x ==3):
                 return 'Wednesday'
             elif (x ==4):
                 return 'Thursday'
             elif (x ==5):
                 return 'Friday'
             elif (x ==6):
                 return 'Saturday'
```

```
In [31]: record['Day'] = record['order_dow'].apply(f)
```

```
In [33]: Hour= pd.get_dummies(record['Hour'], drop_first = True)
         Hour.sample(10)
```

Out[33]:

		EarlyMorning	Evening	Lunch	Morning
user_id	product_id				
19917	25340	0	0	1	0
319	47209	0	0	1	0
27332	46294	0	0	0	0
10164	20738	0	0	0	0
5583	48679	0	1	0	0
29089	47029	0	0	1	0
20773	30635	0	0	0	0
46019	3283	0	0	0	0
60752	20127	0	1	0	0
41698	16953	0	0	1	0

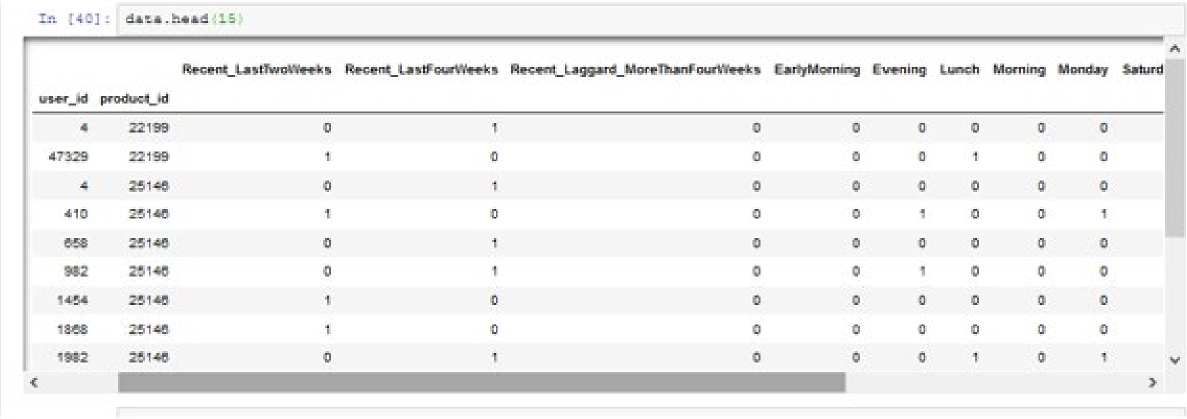
```
In [34]: Day= pd.get_dummies(record['Day'], drop_first = True)
         Day.sample(10)
```

Out[34]:

		Monday	Saturday	Sunday	Thursday	Tuesday	Wednesday
user_id	product_id						
42990	35951	0	0	0	0	1	0
2102	25824	1	0	0	0	0	0
50000	47626	0	0	0	0	0	0
10446	18677	1	0	0	0	0	0
33665	10369	0	0	0	0	1	0
44262	44359	0	0	0	1	0	0
50550	42265	0	0	1	0	0	0
13226	4210	1	0	0	0	0	0
55263	9387	0	0	0	0	0	0
10822	5077	0	0	1	0	0	0

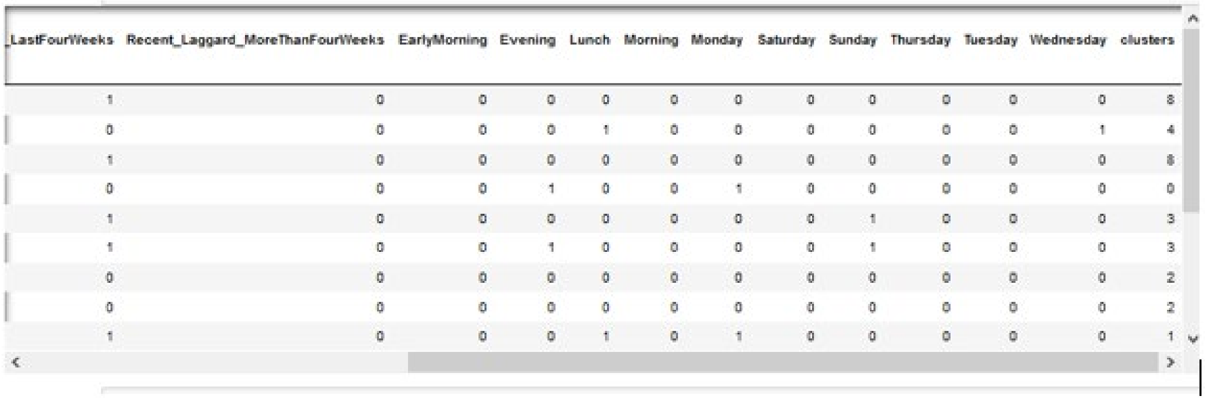
```
In [35]: data= pd.concat([record.drop(['Hour','Day'], axis=1), Hour, Day],axis=1)
```

```
In [36]: data.drop(['order_number','order_hour_of_day', 'days_since_prior_order','product_name','order_dow'],axis=1, inplac
```



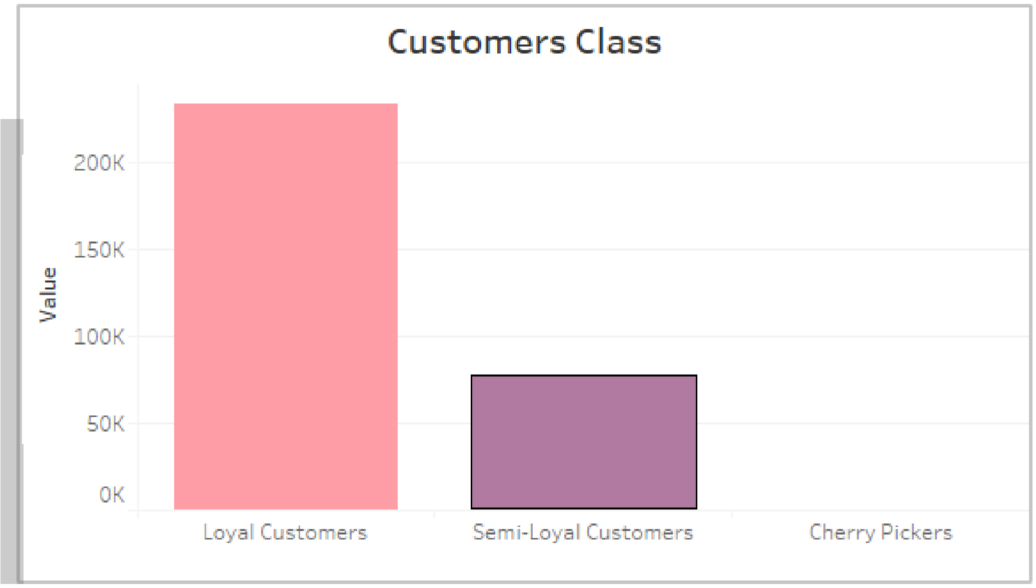
```
In [40]: data.head(15)
```

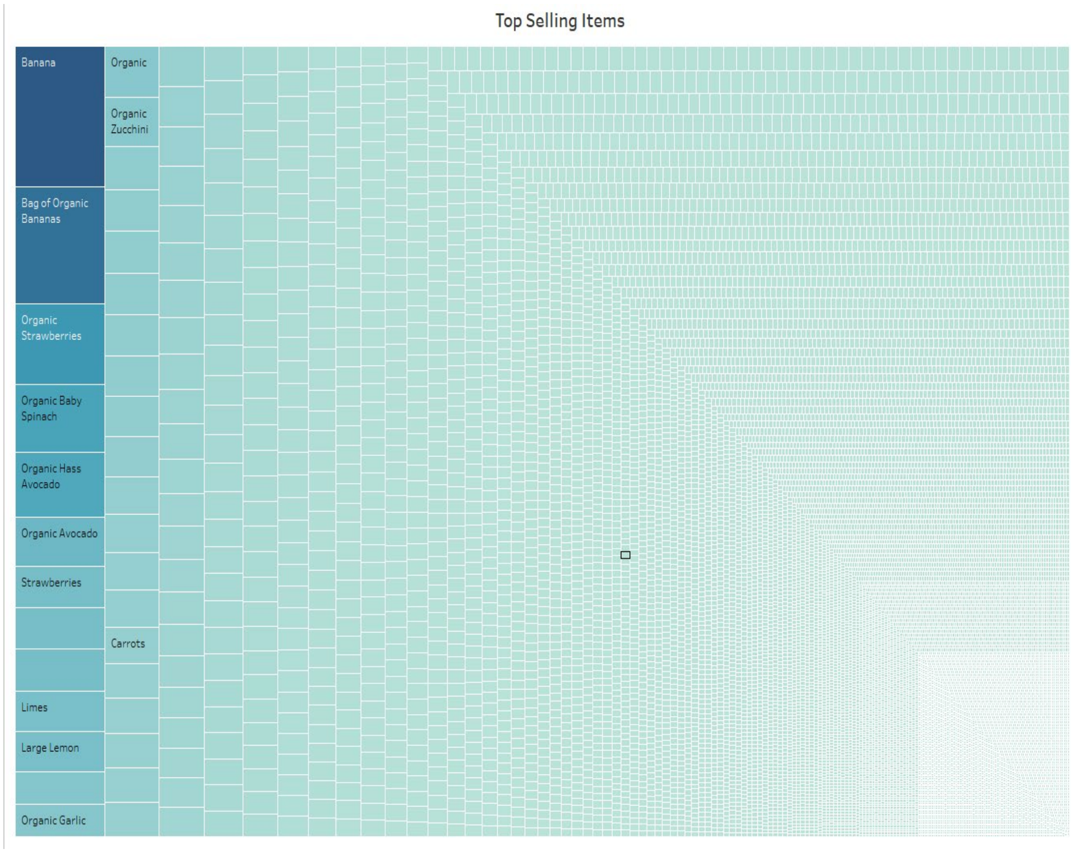
user_id	product_id	Recent_LastTwoWeeks	Recent_LastFourWeeks	Recent_Laggard_MoreThanFourWeeks	EarlyMorning	Evening	Lunch	Morning	Monday	Saturd
4	22199	0	1	0	0	0	0	0	0	
47329	22199	1	0	0	0	0	1	0	0	
4	25146	0	1	0	0	0	0	0	0	
410	25146	1	0	0	0	1	0	0	1	
658	25146	0	1	0	0	0	0	0	0	
982	25146	0	1	0	0	1	0	0	0	
1454	25146	1	0	0	0	0	0	0	0	
1868	25146	1	0	0	0	0	0	0	0	
1982	25146	0	1	0	0	0	1	0	1	



LastFourWeeks	Recent_Laggard_MoreThanFourWeeks	EarlyMorning	Evening	Lunch	Morning	Monday	Saturday	Sunday	Thursday	Tuesday	Wednesday	clusters
1	0	0	0	0	0	0	0	0	0	0	0	8
0	0	0	0	1	0	0	0	0	0	0	1	4
1	0	0	0	0	0	0	0	0	0	0	0	8
0	0	0	1	0	0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	0	0	3
1	0	0	1	0	0	0	0	1	0	0	0	3
0	0	0	0	0	0	0	0	0	0	0	0	2
0	0	0	0	0	0	0	0	0	0	0	0	2
1	0	0	0	1	0	1	0	0	0	0	0	1

Figure 6





Baris, Karaman (2019). Customer Segmentation

https://towardsdatascience.com/data-driven-growth-with-python-part-2-customer-segmentation-5c019d150444

Hemant Warudkar. Customer Segmentation, Data analytics, Retail marketing

https://expressanalytics.com/blog/top-5-uses-of-predictive-analytics-for-supermarkets-and-retail-grocers/

https://towardsdatascience.com/predicting-next-purchase-day-15fae5548027

Dima Shulga (2018). 5 Reasons why you should use Cross-Validation in your Data Science Projects

https://towardsdatascience.com/5-reasons-why-you-should-use-cross-validation-in-your-data-science-project-8163311a1e79

https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn

https://www.forbes.com/sites/hbsworkingknowledge/2014/02/24/six-myths-about-customer-loyalty-programs/?sh=4180d47147cc https://www.annexcloud.com/blog/10-pros-and-cons-of-loyalty-programs/

Gregory E. Smith,Michael S. Rimler(2009) WILL YOU BE MINED? ETHICAL CONSIDERATIONS OF OPT-IN LOYALTY PROGRAMS AND PRICE DISCRIMINATION.

http://iacis.org/iis/2009/P2009_1206.pdf

https://assets.kpmg/content/dam/kpmg/be/pdf/Markets/is-it-time-to-rethink-your-loyalty-program.pdf

https://www.customerexperienceinsight.com/retain-more-customers-2021/

Dumont, Jessica. Why customer loyalty programs are more important than ever

https://www.grocerydive.com/news/why-customer-loyalty-programs-are-more-important-than-ever/533642/

Improving Customer Loyalty in The Competitive Grocery Space

https://www.claruscommerce.com/blog/improving-grocery-loyalty/

Atty, Kate. (2018, October 5). The Next Phase for Grocery Store Loyalty Program

https://www.clutch.com/blog/the-next-phase-for-grocery-store-loyalty-programs/

Choy, Ben. (2019). Promotional Effectiveness Metric & Email Capture Benchmarks Across 10 Ecommerce Industries [2019 Report]

https://www.bigcommerce.com/blog/promotional-marketing/#promotional-effectiveness-lift-benchmarks

Instacart Market Basket Analysis: Which product will an Instacart consumer purchase again,

https://www.kaggle.com/c/instacart-market-basket-analysis