Brazilian E-commerce

Presented by Long Bui & Alissa Dao

Table of Content

I. Project Proposal

Objectives, Dataset, Methods of the Projects

Objectives

01

Determining different segments of customers

02

Understanding the characteristics and behaviors of each customer segment

03

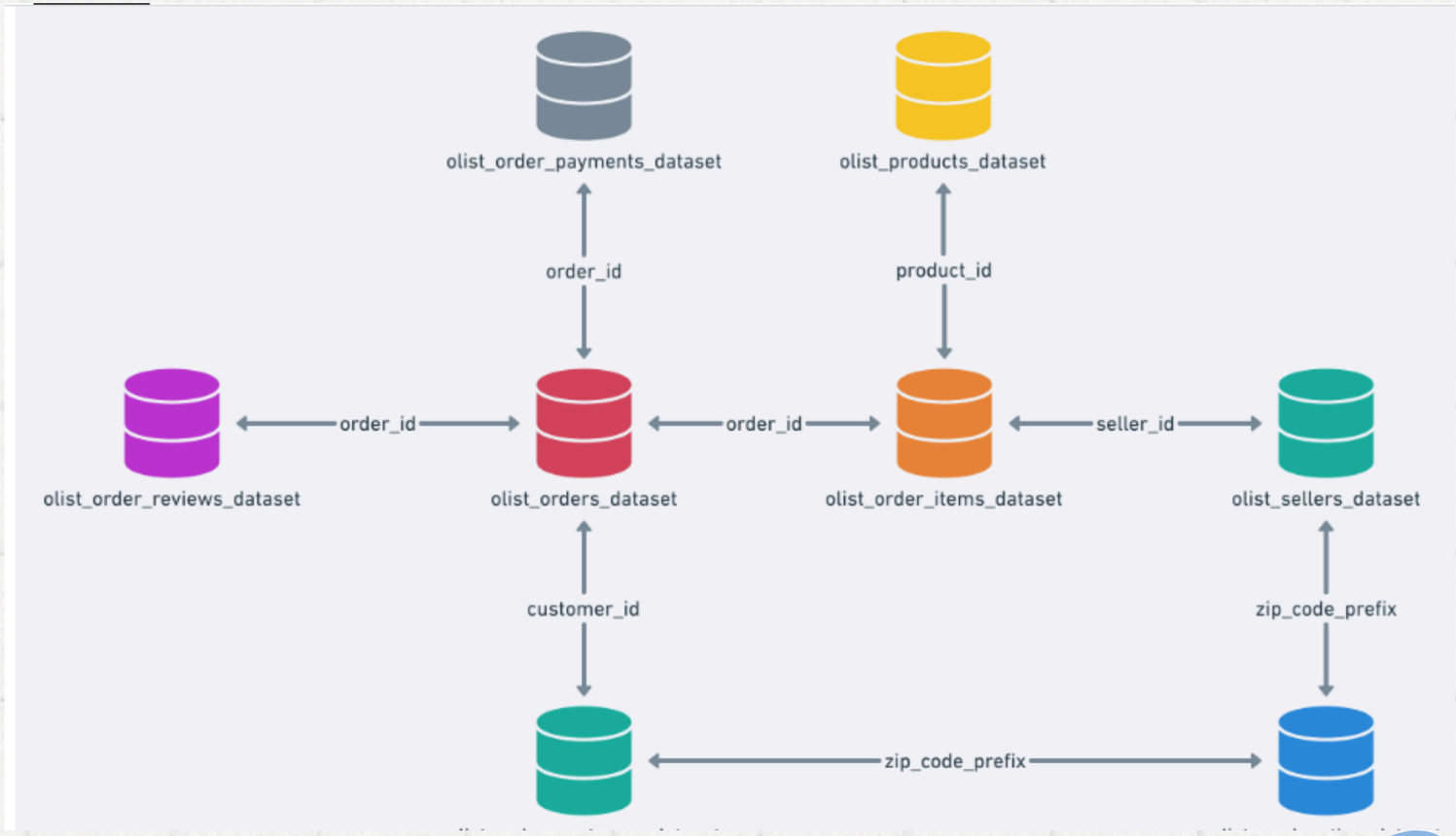
Determining the pairs of items that are mostly bought together

04

Regression to predict Customer LLifetime Value

Dataset

- A set of Brazilian ecommerce public dataset of orders made at Olist Store
- The dataset has information of 100k orders from 2016 to 2018 made at multiple marketplaces in Brazil.
- Variables: order status, price, payment and freight performance to customer location, product attributes and finally reviews written by customers.



Methods

Association	Classification
Apriori	KNNImputer

Linear Regression	Clustering
RandomForest Regressor	KMeans

II. Data Cleaning

	Dataset	Shape	null-amount	num_of_null_col	null_col
0	customer_df	(99441, 5)	0	0	
1	geolocation_df	(1000163, 5)	0	0	
2	orders_df	(99441, 8)	4908	3	order_approved_at, order_delivered_carrier_dat...
3	items_df	(112650, 7)	0	0	
4	payments_df	(103886, 5)	0	0	
5	reviews_df	(99224, 7)	145903	2	review_comment_title, review_comment_message
6	products_df	(32951, 9)	2448	8	product_category_name, product_name_lenght, pr...
7	sellers_df	(3095, 4)	0	0	

NULL VALUES

1) Null String Values: fill with empty string

5	reviews_df	(99224, 7)	145903	2	review_comment_title, review_comment_message

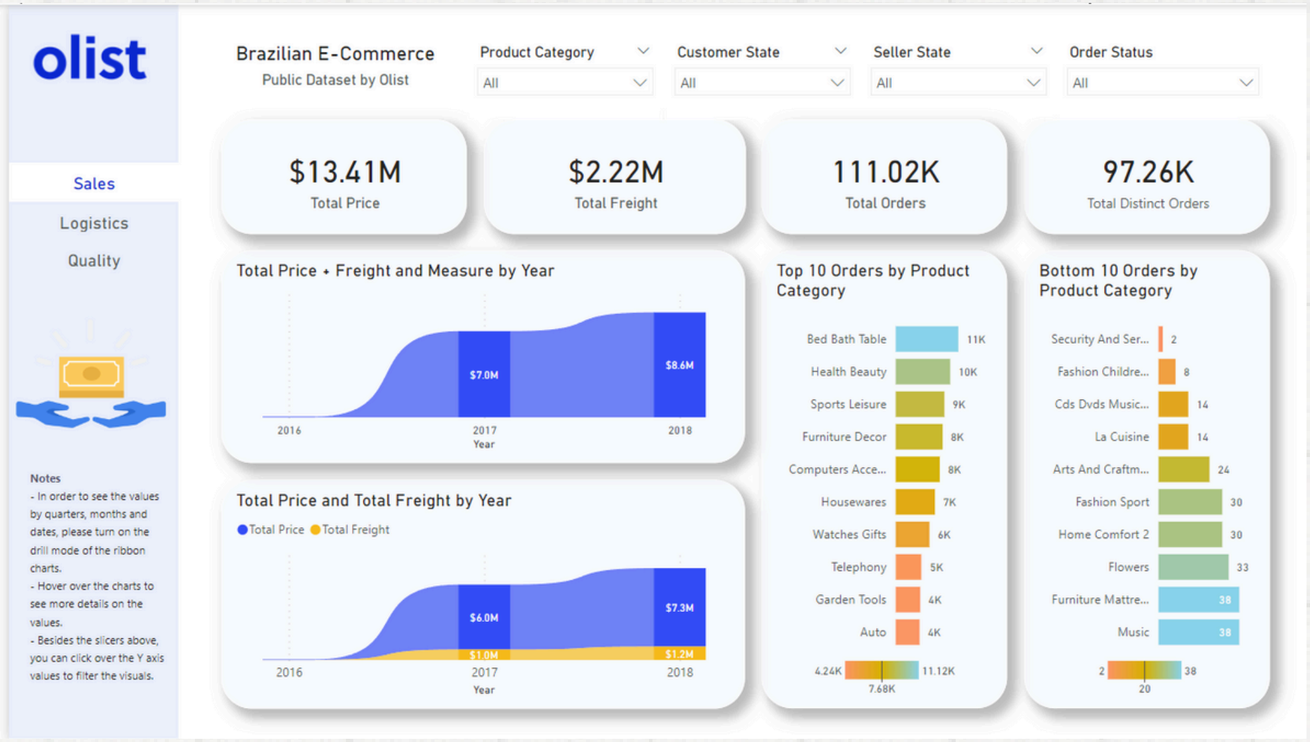
2) Time data: Convert to datetime type --> replace by median time

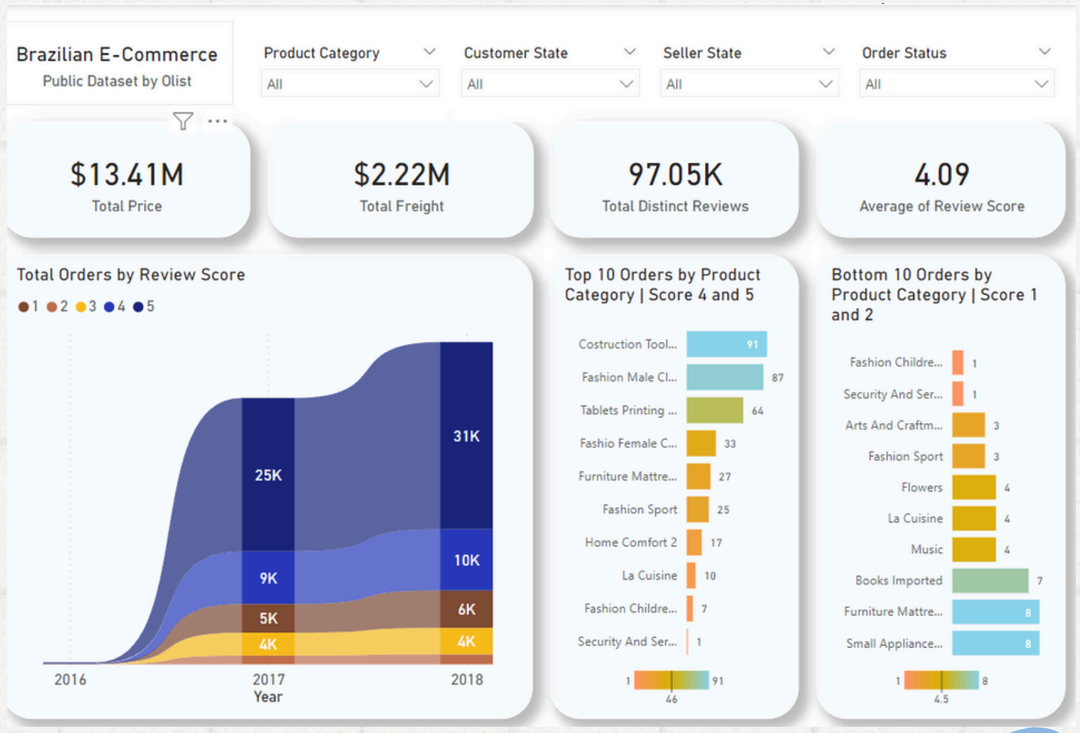
orders_df	(99441, 8)	4908	3	order_approved_at, order_delivered_carrier_

3) Missing values in quantity_df: KNN Imputer

6	products_df	(32951, 9)	2448	8	product_category_name, product_name_lenght, pr...

III. DATA UNDERSTANDINGS



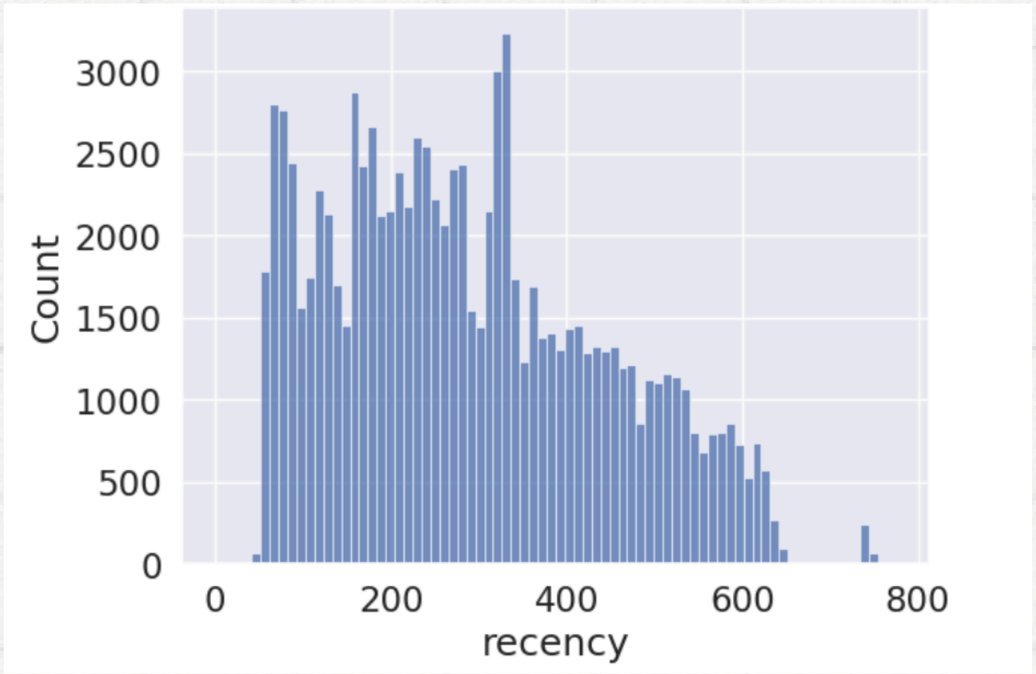


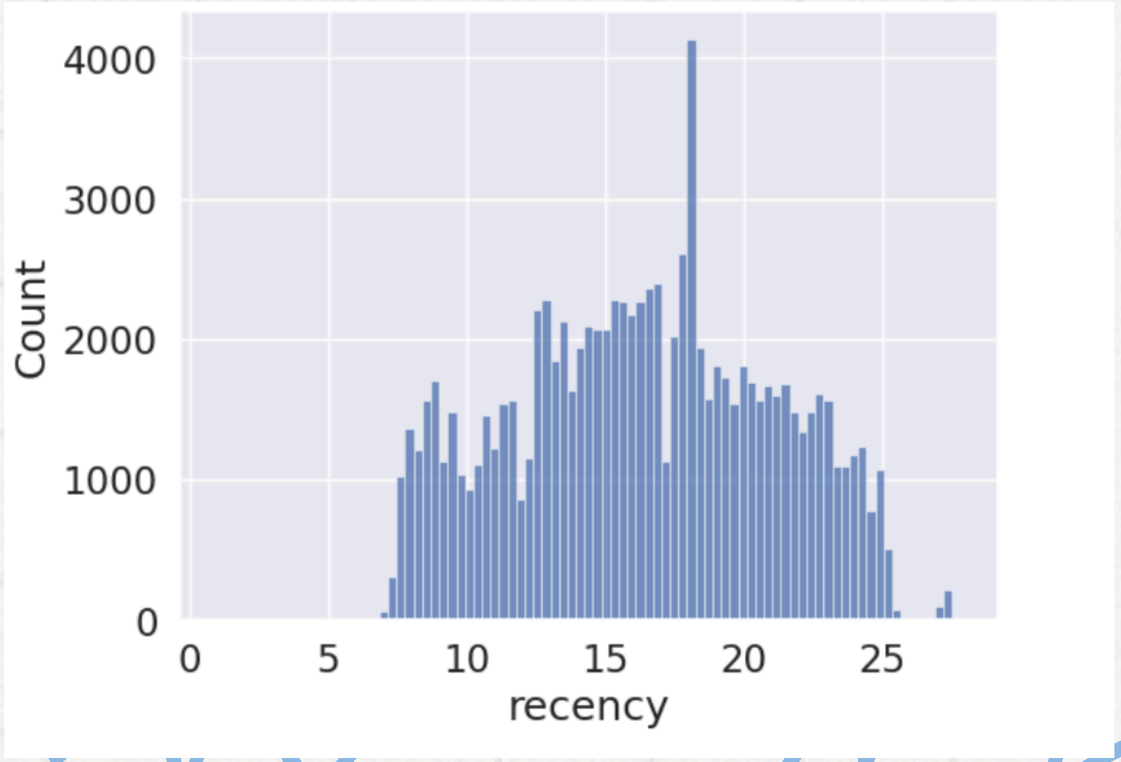
IV. Machine Learning Tasks

Clustering

RFM MODEL

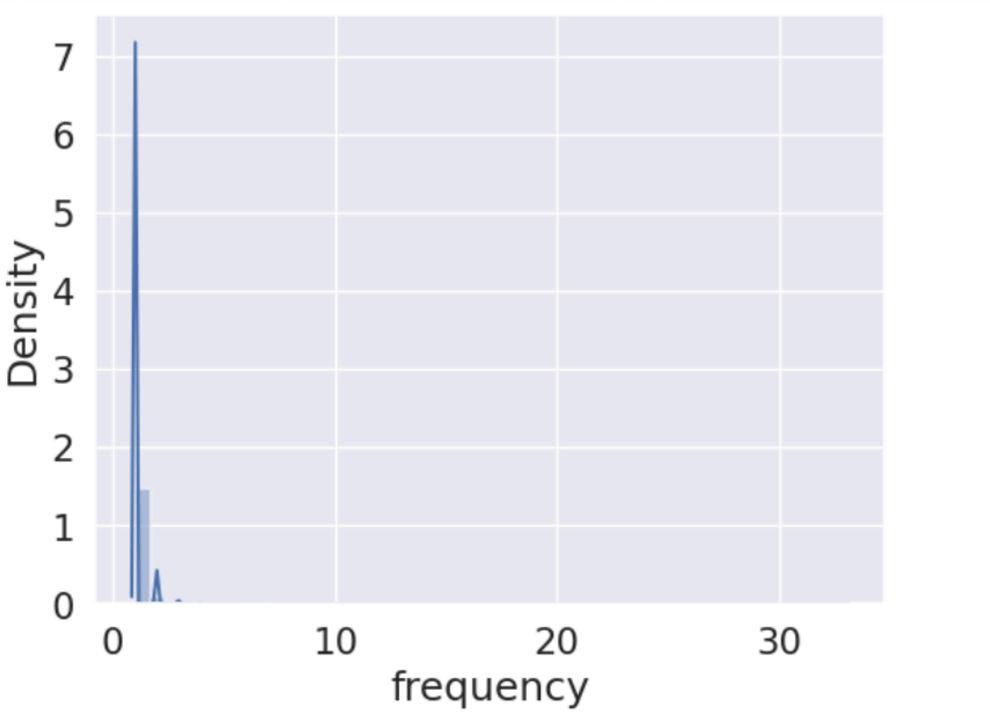
Recency: How recent a customer makes a purchase





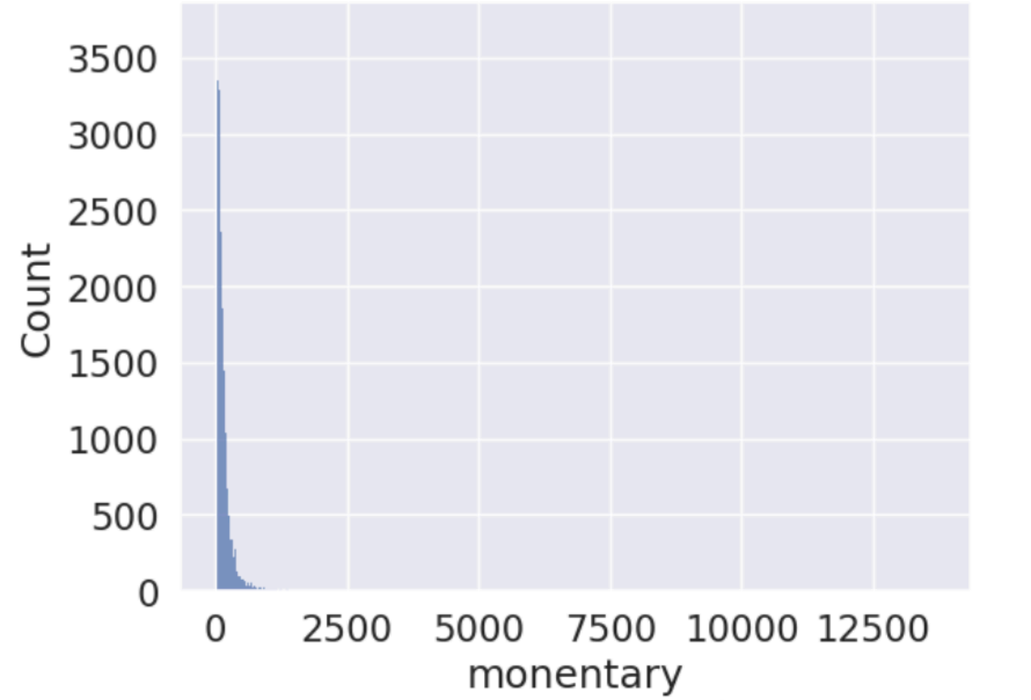
RFM MODEL

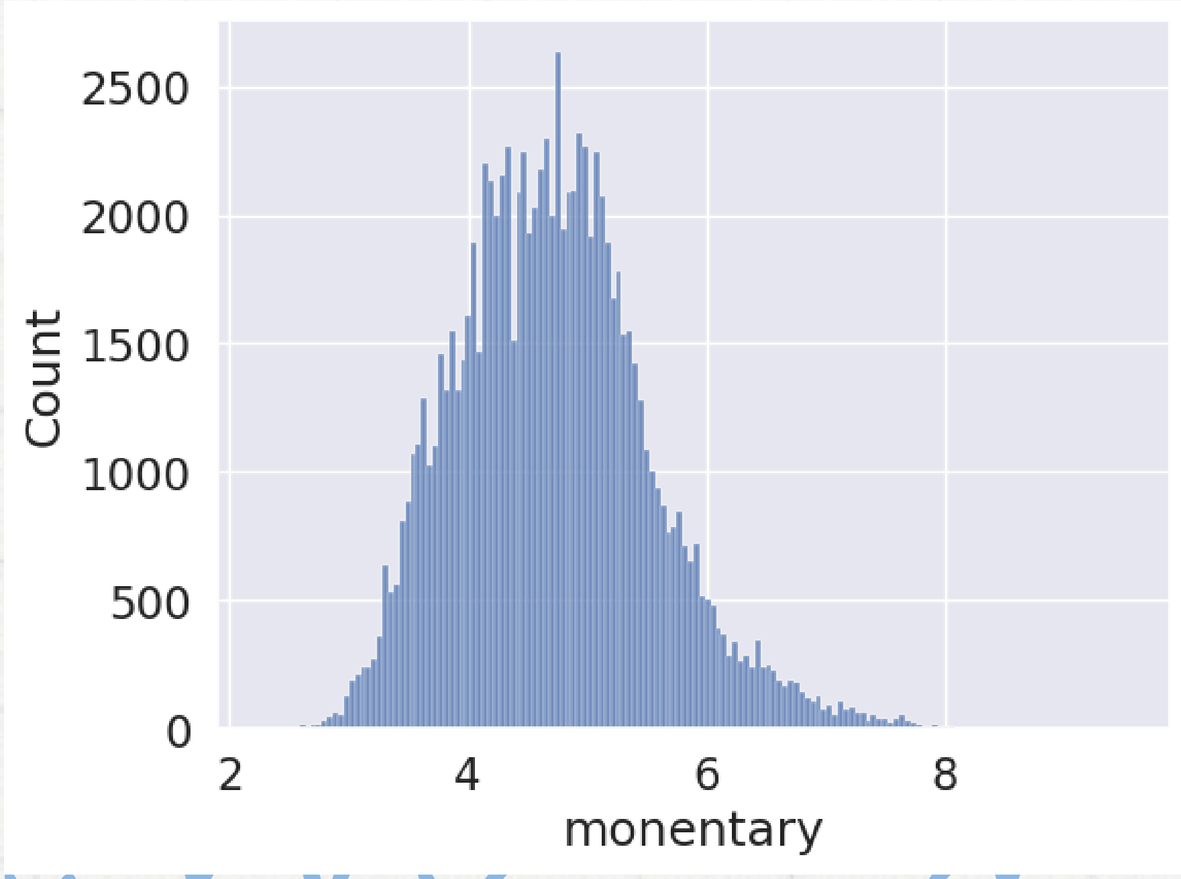
Frequency: How frequent a customer makes a purchase



RFM MODEL

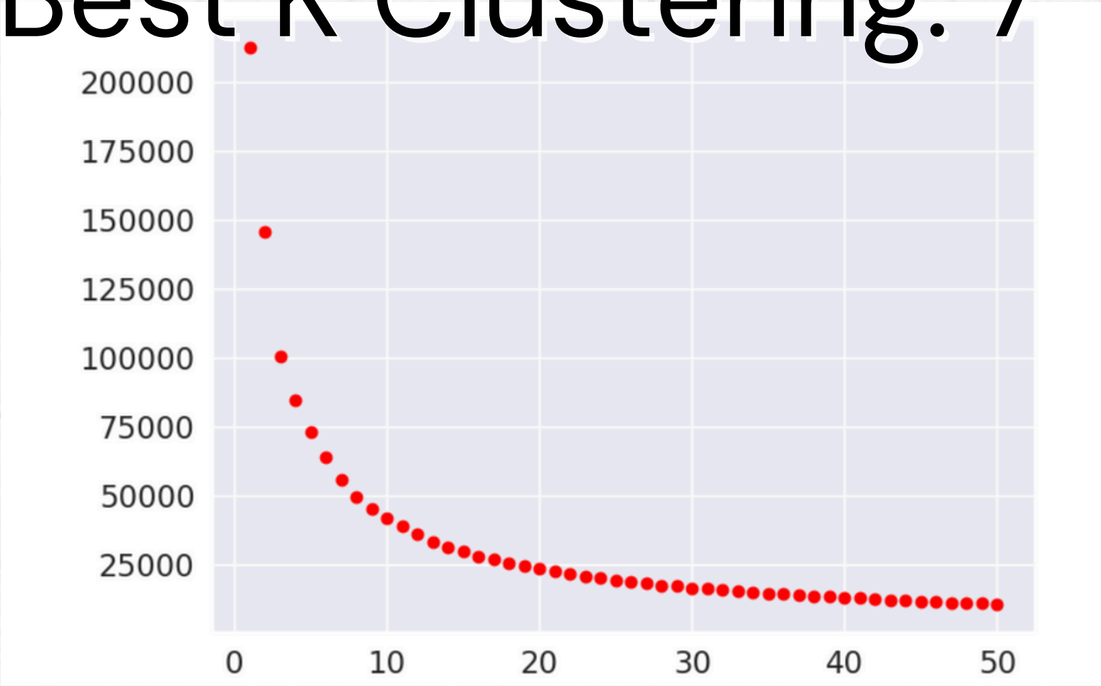
Monentary: Total money a customer makes a purchase





KMEANS CLUSTERING

Best K Clustering: 7



KMEANS CLUSTERING

1st Cluster: Loyal Customers

2nd Cluster: No Spending Incentive Customer

3rd Cluster: Lost Customers (highest recency)

4th Cluster: Whales (high recency)

5th Cluster: Potential Customters (decent recency & payment)

7th Cluster: Worst Customers (low in monentary, high recency)

Clusters Attributes:

clusters_7	monentary	recency	frequency
0	222.123725	318.544413	6.177650
1	56.472862	115.950276	1.033735
2	150.510091	444.603337	1.064795
3	592.312783	320.956713	1.133844
4	233.538668	93.510889	1.094475
5	138.468845	215.231908	1.056264
6	51.696758	370.803279	1.036501

Association

Association

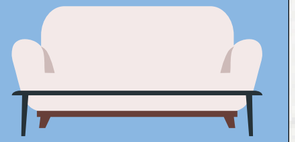
Given a really large number of observations in the dataset, we calculated the associations rules based on each year and each cluster.

The Association Rules are the most popular pair of items in different categories

Cluster 3

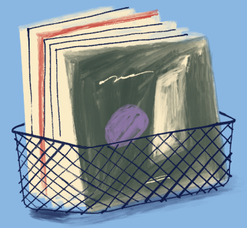
Perfumery - Health Beauty

Furniture Decoration - Bed, Bath, Table



Cluster 4

Furniture Decoration - Bed, Bath, Table



Year 2016

Association Rules

Year 2017

Association Rules

Cluster 1

Housewares - Bed, Bath, Table

Food Drink - Sport Leisure

Cluster 3

Bed, Bath, Table - Home Comfort

Cluster 4

Computer Accessories - Luggage Accessories

Cluster 5

Cool Stuff - Watches Gifts

Furniture Decore - Bed, Bath, Table

Cluster 6

Housewares - Bed, Bath, Table

Toys - Baby

Cluster 7

Housewares - Market Place

Year 2018

Association Rules

Cluster 1

Bed, Bath, Table - Baby

Health beauty - Luggage accessories

Cluster 2

Telephone - Telephone

Cluster 4

Computer Accessories - Luggage Accessories

Cluster 5

Bed, Bath, Table - Bed, Bath- Table

Cluster 6

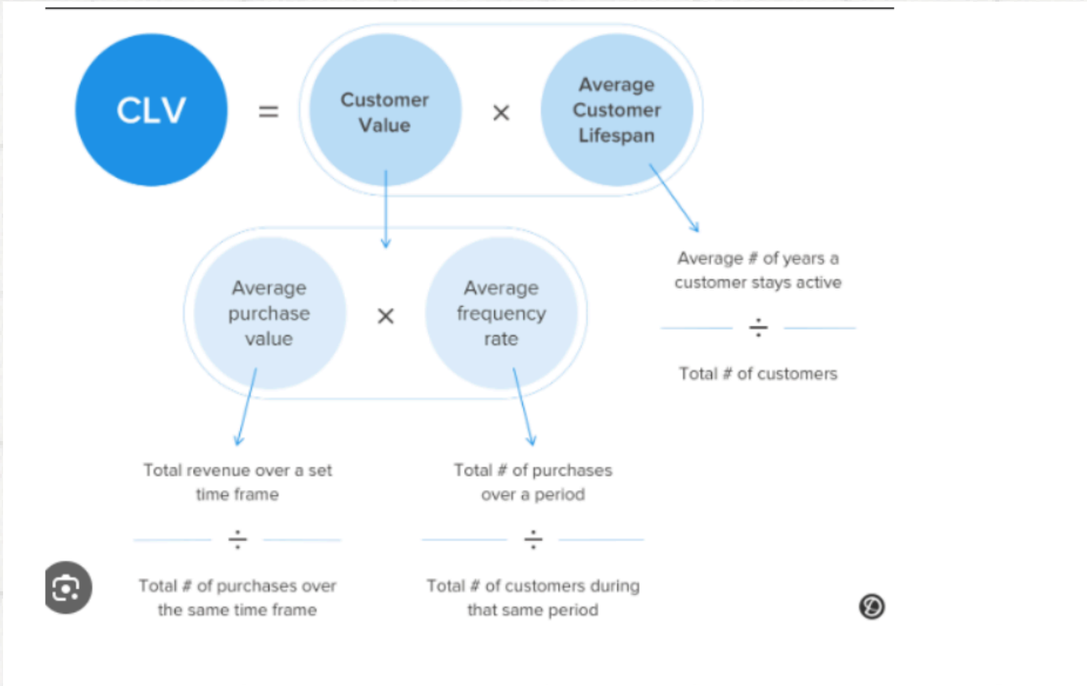
Computer Accessories -Computer Accessories

Cluster 7

Telephone - Telephone

Decision Tree Regression

LINEAR REGRESSION FOR CUSTOMER LIFETIME VALUE



	Customer ID	recency	frequency	monetary	recency_groups	frequency_groups	monetary_groups	overall_score
0	12346.0	325	34	6463.038333	1	2	3	6
1	12347.0	2	222	615.191250	3	3	3	9
2	12348.0	75	51	403.880000	2	2	3	7
3	12349.0	18	175	1107.172500	3	3	3	9
4	12350.0	310	17	334.400000	2	1	3	6

RANDOM FOREST REGRESSOR

Out[58]:
```
0.8610443949901239
```

In [60]:
```
param_dist = {"max_depth": [3, None],

           "min_samples_leaf": range(1,9),
             "criterion": ["gini", "entropy"]}
```

In [61]:
```
tree= DecisionTreeClassifier()
rf= RandomForestClassifier()

tree_cv= RandomizedSearchCV(tree,param_dist,cv=5)
rf_cv= RandomizedSearchCV(rf, param_dist,cv=5)
```

In [62]:
```
tree_cv.fit(X,y)

rf_cv.fit(X,y)
```

Out[62]:
```
RandomizedSearchCV(cv=5, estimator=RandomForestClassifier(),
                   param_distributions={'criterion': ['gini', 'entropy'],
                                        'max_depth': [3, None],
                                        'min_samples_leaf': range(1, 9)})
```

[62]:
```
tree_cv.fit(X,y)

rf_cv.fit(X,y)
```

[62]:
```
RandomizedSearchCV(cv=5, estimator=RandomForestClassifier(),
                   param_distributions={'criterion': ['gini', 'entropy'],
                                        'max_depth': [3, None],
                                        'min_samples_leaf': range(1, 9)})
```

[63]:
```
print(tree_cv.best_score_)

print(rf_cv.best_score_)
```

```
0.8929902806660472
0.8988325000368714
```

MODEL SCORE

```
comparison_data.groupby(['Actual','Prediction'])['Actual'
```

Out[66]:

		Actual	Prediction
Actual	**Prediction**		
	High_ltv	574	574
High_ltv	**Low_ltv**	205	205
	Mid_ltv	12	12
	High_ltv	94	94
Low_ltv	**Low_ltv**	4759	4759
	Mid_ltv	1	1
	High_ltv	73	73
Mid_ltv	**Low_ltv**	25	25
	Mid_ltv	79	79

IV. Conclusion and Discussion

Applications

01

Planning different marketing programs for different customer segments

02

Offering different sales/promotion program for different pairs of items

03

Rearraning the distance between the aisle of the categories that people usually buy together

04

Regression to predict Customer Lifetime Value

Thank you very much!