Architecture

Customer Segmentation using k-prototype algorithm

Revision Number: 1.7

Last date of revision: 23/05/2022

Rajdeep Mondal

Arpan Das

Document Version Control

Date Issued	Version	Description	Author
17th April 2022	1.1	Selected and Downloaded the Dataset	Both
18th April 2022	1.2	Studied the dataset using Pandas	Both
20th April 2022	1.3	Stated some questions regarding the project	Both
24th April 2022	1.4	Performed EDA based on the questions	Both
28th April 2022	1.5	Used tableau and Power BI for dashboards	Both
2nd May 2022	1.6	Performed Clustering for segments	Both
8th May 2022	1.7	Build a model for validation of clusters	Both
12th May 2022	1.8	Explained the Model using LIME	Both
18th May 2022	1.9	Written the Medium Blog.	Both
23rd May 2022	1.10	Restructure and reformatted the LLD.	Both

Contents

Document Version Control

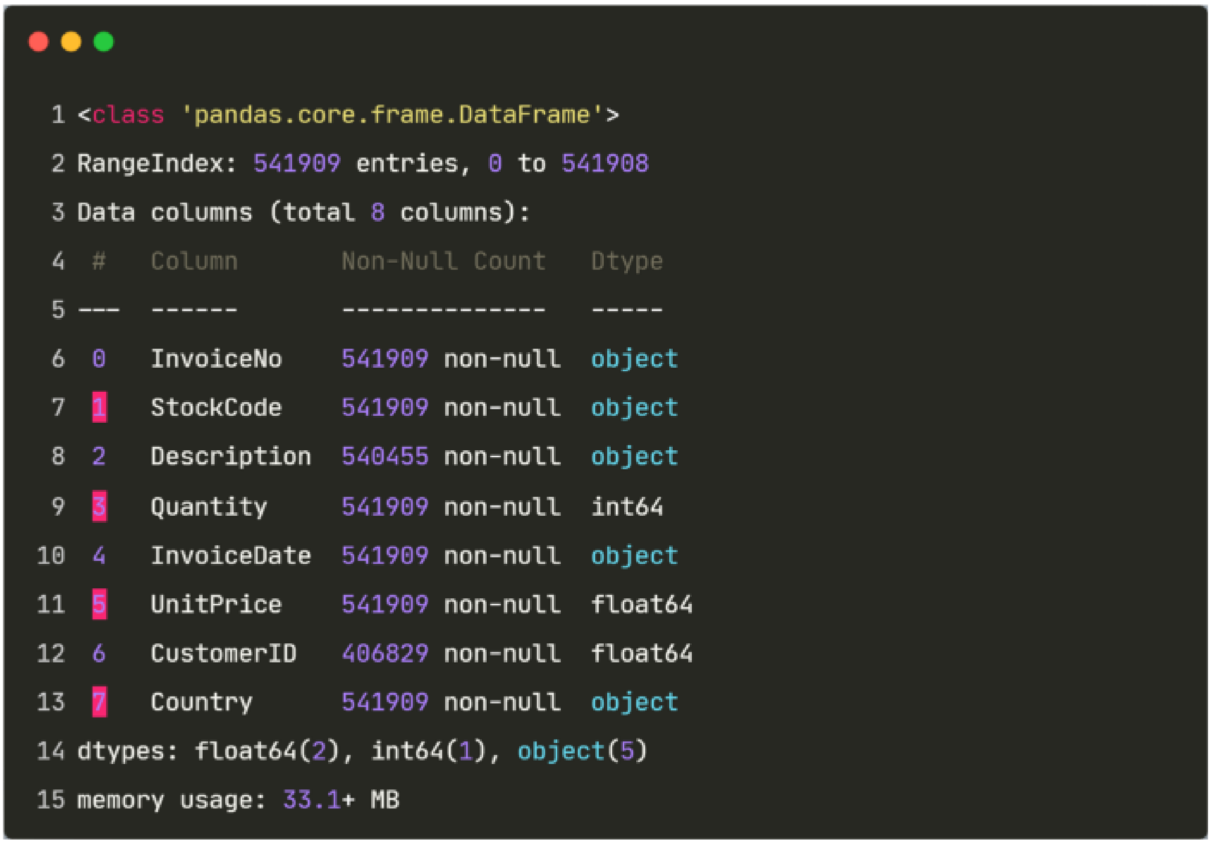
Abstract

This project aims to analyze E-Commerce data that list purchases made by nearly 4000 customers from December 2010 to December 2021. Based on this database we performed Exploratory Data Analysis with Statistical Methods for gaining data-driven insights with machine learning. Here we used Unsupervised techniques with Python for grouping the customers by their behavioral patterns.

Dataset Description

This data contains 8 columns —

1. **InvoiceNo**: This is the Invoice number. There are 25,900 unique invoice data. It is a six-digit integral number uniquely assigned to each transaction. If this code starts with the letter 'C', it indicates a cancellation.
2. **StockCode**: This is the Product (item) code. There are 4,070 unique StockCode values. It is a five-digit integral number uniquely assigned to each distinct product. For some data, it contains special code like — `D`, `POST`, `M`, `C2`, `CRUK`, `Discount`, `POSTAGE`, `Manual`, `CARRIAGE`, `CRUK`, `Commission`.
3. **Description**: This describes the product, ie — Product Name. There are 4224 unique descriptions.
4. **Quantity**: This represents the quantities of each product (item) per transaction. It is a Numeric column.
5. **InvoiceDate**: This displays the Invoice Date and time which was generated when each transaction was completed. It is a Numeric column.
6. **UnitPrice**: This represents the Unit price of each product. It is a Numeric column.
7. **CustomerID**: This represents the unique Customer number. It is a five-digit integral number uniquely assigned to each customer.
8. **Country**: This represents the Country name where each customer resides.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   InvoiceNo    541909 non-null  object 
 1   StockCode    541909 non-null  object 
 2   Description  540455 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object 
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
```

How are we segmenting the dataset?

Customer analysis or customer segmentation is a process that is used to group customer behavior based on features. This may include their purchase history, demographics, and consumption.

We have used the transaction data of one e-commerce store in the United Kingdom. You can get the dataset from here.

For this particular problem, we have handled it by clustering using 2 Machine Learning models (K-Means and K-Prototype). The algorithm found 3 clusters or groups of clients.

Removed Outlier

An outlier is an observation of data that does not fit the rest of the data. It is sometimes called an extreme value. Some outliers are due to mistakes (for example, writing down 50 instead of 500) while others may indicate something unusual.

The most common and powerful tool for detecting outliers is Boxplot. It comes inside the seaborn library as well as on Plotly, but I recommend using Plotly since it provides much more information regarding IQR(Interquartile Range).

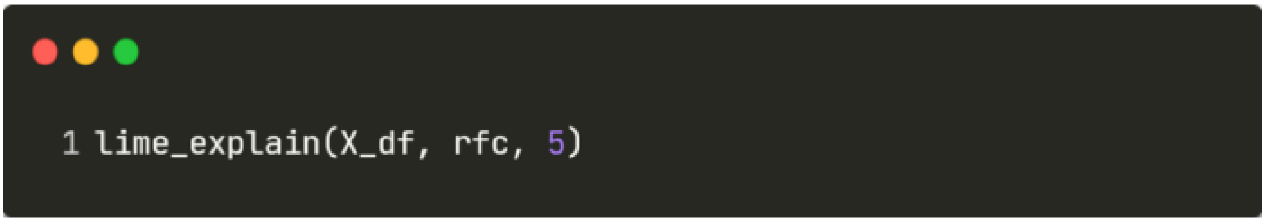
Clustering

For clustering, we need numerical features but in our dataset, there is a mixture of Categorial and Numerical features. KMeans can't be used here because of its limitation of only taking numerical variables. It calculates Euclidean distance. Hence we can use KModes but again Kmodes can't be used with numerical features because it uses Hamming distance, therefore, a new hybrid algorithm comes into the picture. **K-Prototype** proposed by Huang.

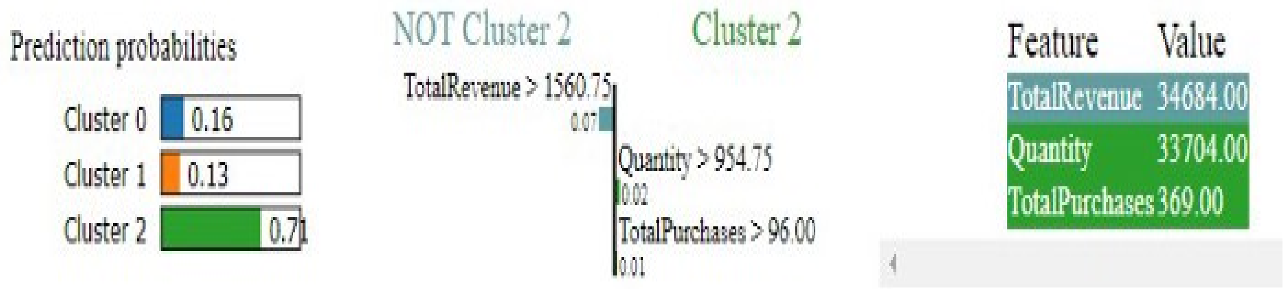
LIME to explain our model

Lime is able to explain any black-box classifier, with two or more classes. All it requires is that the classifier implements a function that takes in raw text or a NumPy array and outputs a probability for each class. For further details check out this paper.

The output of LIME provides a list of explanations, reflecting the contribution of each feature to the prediction of a data sample. This provides local interpretability, and it also allows to determine which feature changes will have the most impact on the prediction.



```
lime_explain(X_df, rfc, 5)
```



Feature	Value
TotalRevenue	34684.00
Quantity	33704.00
TotalPurchases	369.00

1. `TotalPurchases > 96.00` — Greater values than 96.00 positively correlate with a being in Cluster 2.
2. `Quantity > 954.75` — Greater values than 954.75 positively correlate with a being in Cluster 2.
3. `TotalRevenue > 1560.75` — Greater values than 1560.75 negatively correlates with not being in Cluster 2.

Conclusion

So finally we came to the end of this project. We learned a lot of things and faced several challenges along the way. This helped us improve our skills more in clustering and model explainability. There are many ways you could improve this model and we heartily encourage it. Do provide some feedback on what you learned from this blog and also provide new approaches that can be augmented to solve this problem in a better way.