

A decorative header at the top of the slide with a blue and white textured, water-like appearance.

# Customer Segmentation

Marc Rodriguez

# Project Introduction

*My project will be working with Customer Segmentation through using the K-Means algorithm and applying it to the industry of online retail through an open source data set. The dataset will consist of information regarding online transactions. The categorical columns of this dataset range from Invoice Number, to Quantity to Country, which provide a series of topics to aid in the segmentation.*



# What is Customer Segmentation?

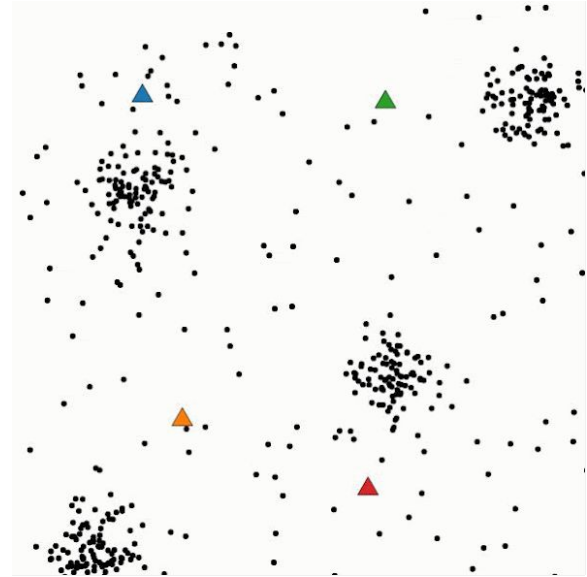
*Customer Segmentation is the general practice of dividing clientele into groups that may be similar in certain characteristics or patterns. This is often done to better understand the customer of the business and evolve certain strategies or approaches taken to gain clientele or general business. This often also aids the customer by having businesses better understand them and their holistic needs or desires. Customer segmentation can be applicable to various industries but I decided to focus on retail as it was the one that seemed the most crucial in perspective.*



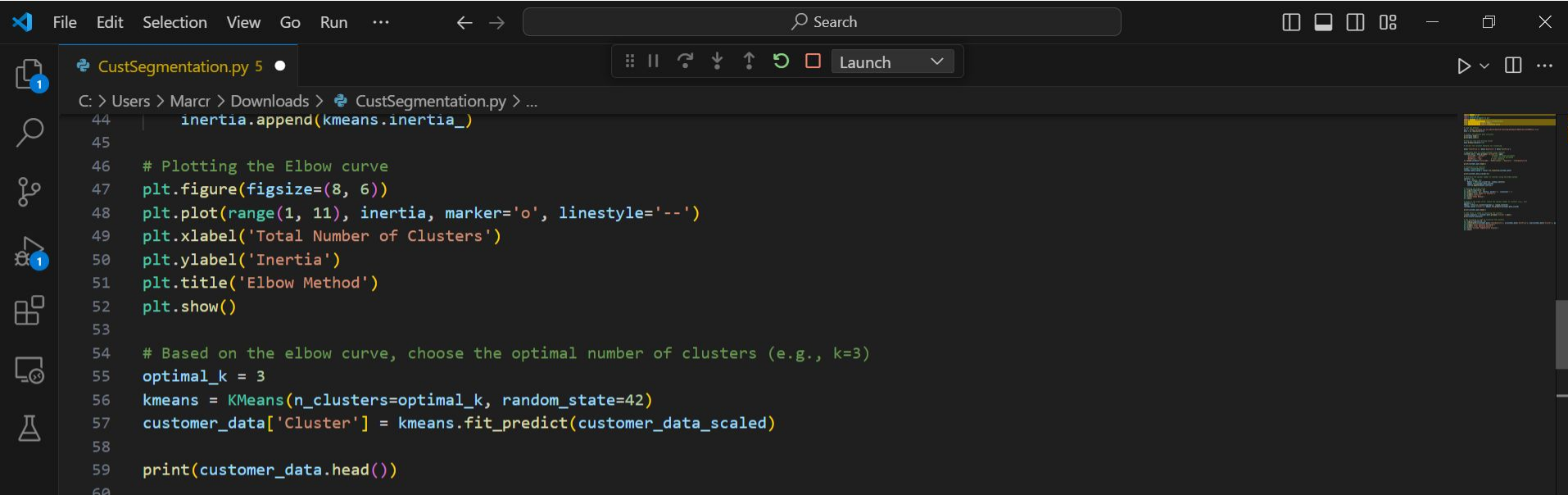
# What is K-Means Algorithm?

A popular clustering methodology that is often implemented in the realm of machine learning. Focuses upon dividing a dataset into a series of clusters. Data points in these datasets eventually belong to a certain cluster based upon proximity.

*Note: This is just a figure not attributed to my code, but a simple demonstration of k-means clustering and its implementation*



# Implementation of K Means



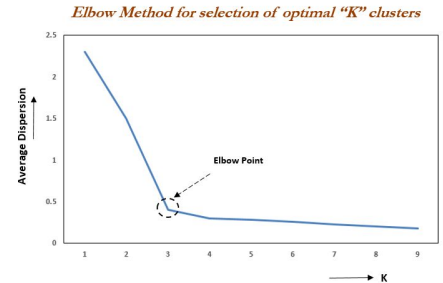
```
CustSegmentation.py 5 •
C: > Users > Marcr > Downloads > CustSegmentation.py > ...
44     inertia.append(kmeans.inertia_)
45
46 # Plotting the Elbow curve
47 plt.figure(figsize=(8, 6))
48 plt.plot(range(1, 11), inertia, marker='o', linestyle='--')
49 plt.xlabel('Total Number of Clusters')
50 plt.ylabel('Inertia')
51 plt.title('Elbow Method')
52 plt.show()
53
54 # Based on the elbow curve, choose the optimal number of clusters (e.g., k=3)
55 optimal_k = 3
56 kmeans = KMeans(n_clusters=optimal_k, random_state=42)
57 customer_data['Cluster'] = kmeans.fit_predict(customer_data_scaled)
58
59 print(customer_data.head())
60
```

# Expected Outcome

*Regretfully I couldn't get my imports sorted in time but based upon the code I would expect:*

- An output of several rows of data relevant to Dataframe with any rows missing values to be removed
- Aggregation of data and 5 rows of standardized data to be output
- Elbow Plot Curve and visualization of mean of clusters

*Example of elbow plot curve:*



# Works Cited

- Mohd Mazz, Customer Segmentation, Kaggle Data Set,  
<https://www.kaggle.com/code/maaz0511/9-customer-segmentation#Dataset-Details>