CUSTOMER SEGMENTATION

CUSTOMER SEGMENTAION

Customer segmentation analysis is the process performed when looking to discover insights that define specific segments of customers. Marketers and brands leverage this process to determine what campaigns, offers, or products to leverage when communicating with specific segments.

Machine Learning for customer segmentation

Machine learning methodologies are a great tool for analyzing customer data and finding insights and patterns. Artificially intelligent models are powerful tools for decision-makers. They can precisely identify customer segments, which is much harder to do manually or with conventional analytical methods.

There are many machine learning algorithms, each suitable for a specific type of problem. One very common machine learning algorithm that's suitable for customer segmentation problems is the k-means clustering algorithm. There are other clustering algorithms as well such as DBSCAN, Agglomerative Clustering, and BIRCH, etc.

Exploring customer dataset and its features

Let's analyze a customer dataset. Our dataset has 24,000 data points and four features. The features are:

Customer ID – This is the id of a customer for a particular business.

Products Purchased – This feature represents the number of products purchased by a customer in a year.

Complaints – This column value indicates the number of complaints made by the customer in the last year

Money Spent – This column value indicates the amount of money paid by the customer over the last year.

```
customersdata.head()
```

	customer_id	products_purchased	complains	money_spent
0	649	1	0.0	260.0
1	1902	1	0.0	79.2
2	2155	3	0.0	234.2
3	2375	1	0.0	89.0
4	2407	2	0.0	103.0

PRE-PROCESSING THE DATASET

Before the data's are fetch to the corresponding algorithms.

The algorithm used in this project is K-means algorithm.

Before feeding k-means algorithm ,we need to pre process the dataset.let's implement the necessary pre-processing for the customer dataset.

Implementing K-means clustering

K-Means clustering is an efficient machine learning algorithm to solve data clustering problems. It's an unsupervised algorithm that's quite suitable for solving customer segmentation problems. Before let's explore two key concepts

Unsupervised Learning

Unsupervised machine learning is quite different from supervised machine learning. It's a special kind of machine learning algorithm that discovers patterns in the dataset from unlabelled data.

Unsupervised machine learning algorithms can group data points based on similar attributes in the dataset. One of the main types of unsupervised models is clustering models.

Note that, supervised learning helps us produce an output from the previous experience.

Clustering algorithms

A clustering machine learning algorithm is an unsupervised machine learning algorithm. It's used for discovering natural groupings or patterns in the dataset. It's worth noting that clustering algorithms just interpret the input data and find natural clusters in it.

Some of the most popular clustering algorithms are:

K-Means Clustering

Agglomerative Hierarchical Clustering

Expectation-Maximization (EM) Clustering

Density-Based Spatial Clustering

Mean-Shift Clustering

In the following section, we're going to analyze the customer segmentation problem using the k-means clustering algorithm and machine learning. However, before that, let's quickly discuss why we're using the k-means clustering algorithm.

Why use K-means clustering for customer segmentation?

Unlike supervised learning algorithms, K-means clustering is an unsupervised machine learning algorithm. This algorithm is used when we have unlabelled data. Unlabelled data means input data without categories or groups provided. Our customer segmentation data is like this for this problem.

The algorithm discovers groups (cluster) in the data, where the number of clusters is represented by the K value. The algorithm acts iteratively to assign each input data to one of K clusters, as per the features provided. All of this makes k-means quite suitable for the customer segmentation problem.

Given a set of data points are grouped as per feature similarity. The output of the K-means clustering algorithm is:

The centroids values for K clusters,

Labels for each input data point.

Finding the optimal number of clusters

Finding the optimal number of clusters is one of the key tasks when implementing a k-means clustering algorithm. It's worth noting that a k-means clustering model might converge for any value of K, but at the same time, not all values of K will produce the best model.

For some datasets, data visualization can help understand the optimal number of clusters, but this doesn't apply to all datasets. We have a few methods, such as the elbow method, gap statistic method, and average silhouette method, to assess the optimal number of clusters for a given dataset. We'll discuss them one by one.

The elbow method finds the value of the optimal number of clusters using the total within-cluster sum of square values. This represents how spread-apart the generated clusters are from one another. In this case, the K-means algorithm is evaluated for several values of k, and the within-cluster sum of square values is calculated for each value of k. After this, we plot the K versus the sum of square values. After analyzing this graph, the number of clusters is selected, so that adding a new cluster doesn't change the values of the sum of square values significantly.

Average silhouette method is a measure of how well each data point fits its corresponding cluster. This method evaluates the quality of clustering. As a general rule, a high average silhouette width denotes better clustering output.

Gap statistic method is a measure of the value of gap statistics. Gap statistics is the difference between the total intracluster changes for various values of k compared

to their expected values. This calculation is done using the null reference distribution of data points. The optimal number of clusters is the value that maximizes the value of gap statistics.

We're going to use the elbow method. The K-means clustering algorithm clusters data by separating given data points in k groups of equal variances. This effectively minimizes a parameter named inertia. Inertia is nothing but within-cluster sum-of-squares distances in this case.

When we use the elbow method, we gradually increase the number of clusters from 2 until we reach the number of clusters where adding more clusters won't cause a significant drop in the values of inertia.

The stage at this number of clusters is called the elbow of the clustering model. We'll see that in our case it's K =5.

For implementing the elbow method, the below function named "try_different_clusters" is created first. It takes two values as input:

•K (number of clusters),

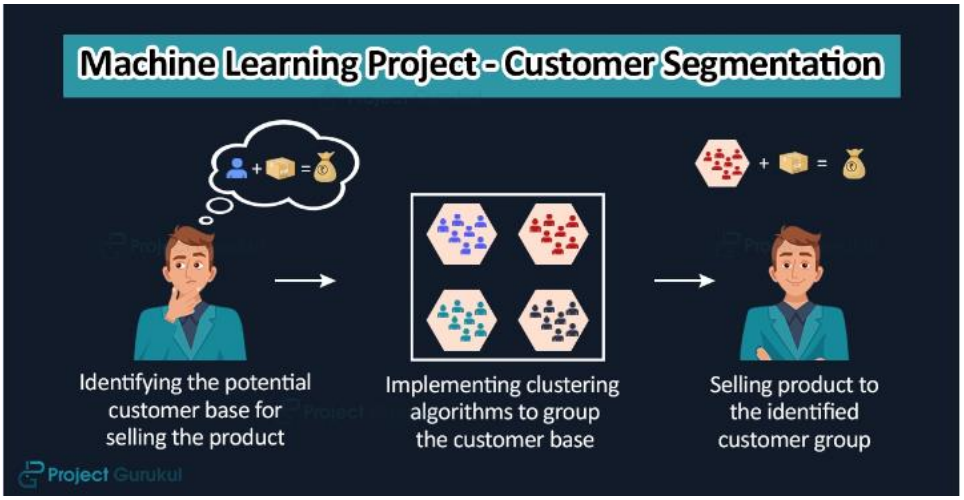
•data (input data).

Where we pass values of k from1-12 and calculate the inertia for each value of k.

And plot the value of k omnx-asxis againt y-axis for inertia avlue

Then ,lets plot the graph.

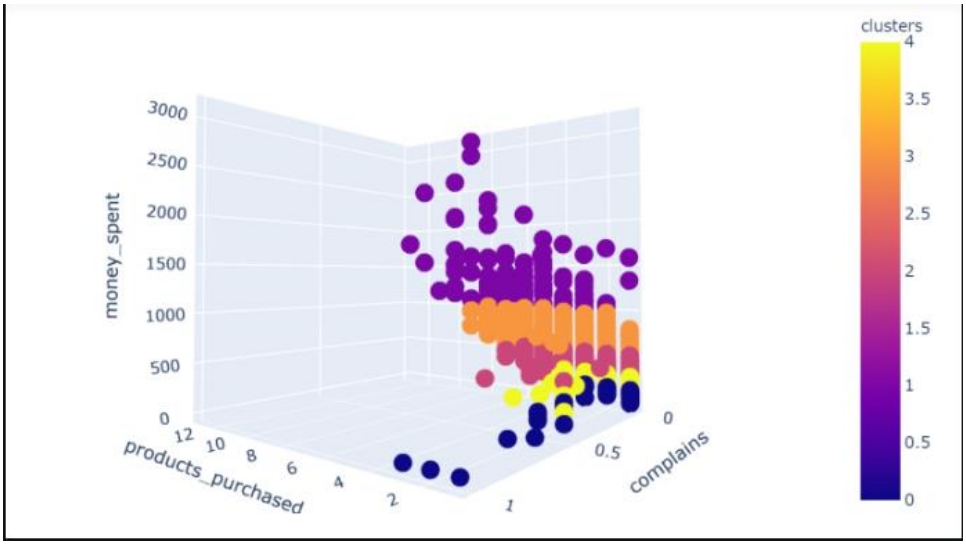
VISUALISATION



Generally, it's referred to as px. It's worth noting plotly express is the built-in module of the plotly library. This is the starting point of creating the most common plots as recommended. Note that each plotly express function creates graph objects internally and returns plotly.graph_objects.

A graph created by a single method call using plotly express can be also created using graph objects only. However, in that case, it takes around 5 to 100 times as much code.

As the 2D scatter plot, px.scatter plots individual data in a two-dimensional space, and the 3D method px.scatter_3d plots individual data in a three-dimensional space.



CONCLUSION

Thus the every process for the project