REPORT

PRML MINOR PROJECT

B21EE043 and B21EE055

PIPELINE –

Major Problems and their Solution Approaches—

1. The Dataset is too large(5,42,221 rows) and Sparse. Sampling helped wherever needed.
2. Problem with customer id column. Column is playing a major role(clustering 100s of data points) as well as having a large number of NULL values. Using a Supervised Learning approach (Decision Tree) to fill the NULL values.
3. Making the DateTime useful. Separating the DateTime column into 4 different columns: date, month, year and time(further divided into bins).
4. Visualization of the dataset is also a problem since the dataset is very large. Using some useful libraries like DataPrep.
5. We are unable to train many models on the dataset due to their high time complexity. Not able to take average linkage in hierarchical clustering .So I used sampling as a tool to handle this problem.
6. As the no. of features are more. So how to visualize a scatter plot. So we used features having highest eigenvalues as they conserve the most of the variance of the data. And used 3d plots.

Outline:

7. Preprocessing : Handling outliers, handling NULL values, column selection using correlation, Encoding, Visualization.
8. Algorithms used: PCA, K Means, hierarchical , DBSCAN (because of the large dataset we had limited no. of algorithms).
9. Also Used **elbow method** to decide best no. of clusters.
10. Evaluation methods: silhouette score, calinski harabasz score , davies bouldin score, sse method. Why are these used? (as we are having an unsupervised learning problem without labels so used intrinsic measure methods.
11. As a result **hierarchical and DBscan** are working best here. And dividing the data in **two clusters**.

DETAILED REPORT :-

Retail dataset Analysis(Preprocessing) :

To analyze this dataset we need maximum information to be extracted from the dataset. To do this we started with the **InvoiceDate column.** This column is not very useful unless it is converted into useful information.

- So we used the **data in this column to create 4 different columns for date, month, year and time. we binned the as time column to convert the discrete values of time to useful intervals.** Month(1 to 12), Date(1 to 31), time(6 to 9, 9 to 12 , 12 to 15, 15 to 19 , 19 to 21) the bins were made.
- The bins for the time are specifically chosen based on the minimum, maximum time and as well as the most probable time in the dataset. The time is first divided in intervals using cut function and then intervals are encoded using the label encoder.
- Moving ahead in the analysis of the data we have first checked the **total number of discrete values and the null values in each column** of the dataset. This gives us a fair idea about the distribution of data along the columns as well as the ambiguity in the data. (**description(1454)** and **customer id(135080)** having null values.)
- To handle the null values we have **deleted the rows corresponding to the column with a lower number of null values.** But we also have a **column with a large number** of null values in the dataset and the column(CustomerID) **is an important driving factor in separating different types of customers** from the dataset. **This column is handed later in the program.**
- Now shifting our focus on creating a **new valuable column (TotalSale)** from two columns of the dataset(**Quantity and UnitPrice**). This column contains the total sale in a single bill.

Handling outliers:

- Presence of outliers in the dataset is a major issue which can cause large impacts on statistical and hypothetical analysis. So we checked the presence of outliers in some specific columns chosen based on the type of column (column with some sort of distribution). **Box plots may be useful here but due to the large amount of data we are unable to go with box plots. (drawback of large dataset).**
- The presence of an outlier is decided if the value in these columns is **out of range ($\mu-3\sigma,\mu+3\sigma$).** The index of such rows are stored and then **these rows are deleted** from the dataset.

Handling ambiguous data in Description Column:

- looking at the dataset we noticed that the description column contains some ambiguous data. From our observation we find that most of the noisy data in this column doesn't have a capital letter beginning whereas the useful data is Starting with a capital letter. So we removed these rows from our dataset.

Negative value test:

- Our dataset must contain only non-negative for a selected number of columns(unit price, quantity) . So we checked for negative values in those columns and if such data is present it is removed from the dataset.
- **Now our dataset is almost free from noisy data**. So we moved towards encoding of columns which either contain values either in the form of string or numerical values which need to be encoded.

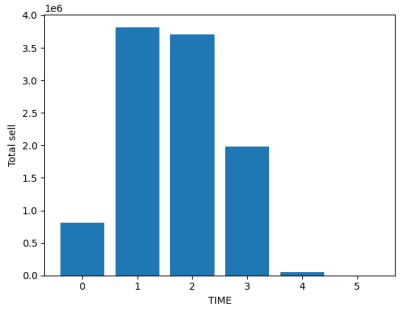
Handling NULL values in CustomerID column:

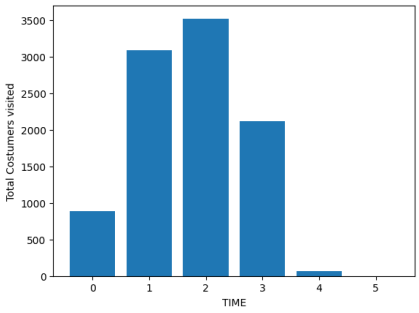
- There are about 20% null values in this column. Deleting the rows with null values is not a good idea as it will cause loss of huge data and also this column can't be deleted because it puts together all the rows associated with a particular customer. So we came up with an idea of filling the null values based on a classification algorithm trained on rows with non-null values. The null values of each row is then filled with the prediction by this classification algorithm.

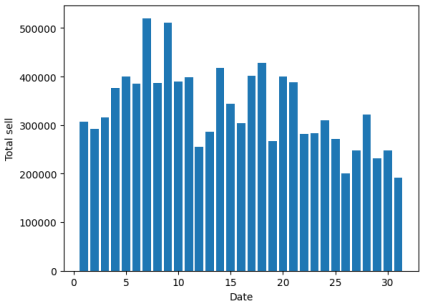
Why is the **decision tree** used **for predicting customer id**?

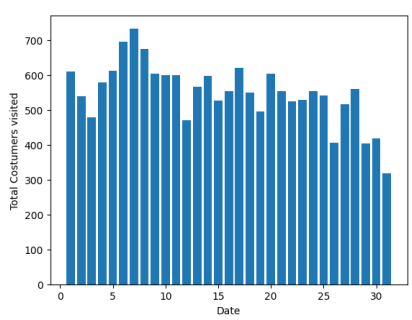
- We tried working with some other more efficient algorithms but since the dataset is very large those algorithms are taking a lot of time and also we faced the problem with RAM used in some algorithms. So we continued with the Decision tree Algorithm.

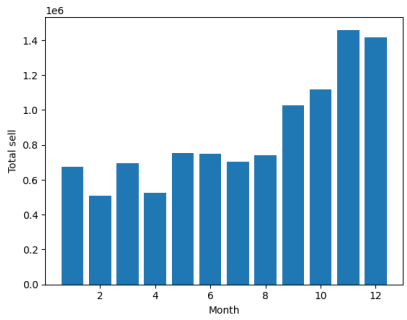
Now, at last we headed towards the visualization of the dataset. We tried getting a complete exposure of the dataset using the plots.

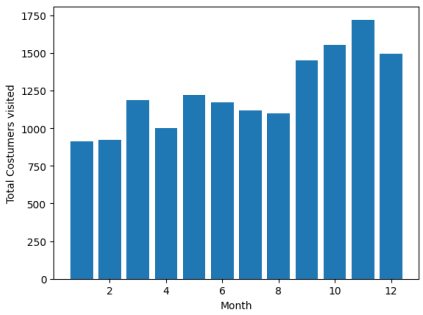


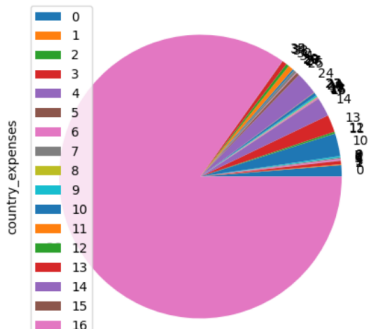


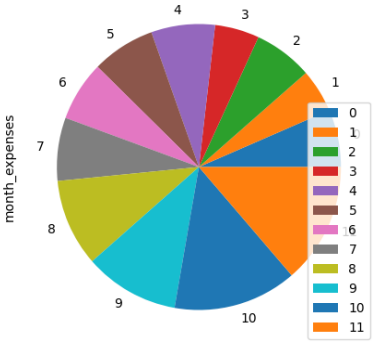


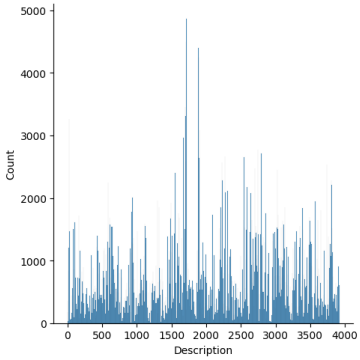


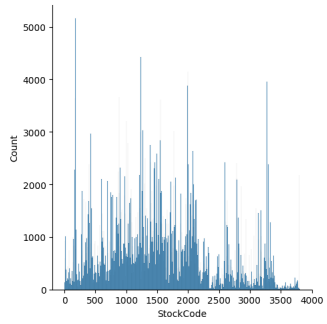


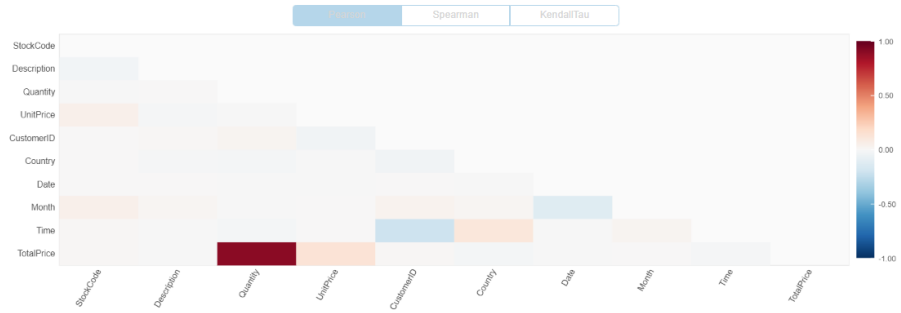












Observations:

1. We observed that **more than 80%** of the data is from just a single country(**United kingdom**). So we can call this data biased for a particular country.
2. From the monthly total sale we can see that about 50% of the sale is from the last 4 months of the year. Also, from the total sales and total customers visit analysis we can see that more customers are **visiting during the last few months**.
3. Also we can see that on a day time from(**9 am to 3 pm the sale is maximum or 70 percent of the whole day sale**).
4. Now using the DataPrep Report **on this updated dataset** we get a complete analysis report of the data. From this report we can see that there are **10 remaining features, no null values in the dataset and count of duplicate data in the dataset.**

Why to use the DataPrep Report inbuilt library function ?

It is fast and more efficient than the individual algorithms that you run for various plots for visualization. Since we faced problems with RAM and time consumed by the individual plots algorithms we moved with this report.

5. Now in the Data Report this we get the count of positive, zero and negative values for each column. Also it gives the distribution of the feature along with min-max and mean values of a respective column.
6. Also this DataPrep Report has a feature to to visualize the distribution of this dataset with taking two features at a time. But these plots are not very useful to comment about the overall distribution at this time.
7. From the correlation chart we can see that **TotalPrice and Quantity are highly correlated**. Also TotalPrice is also having a good amount of correlation with UnitPrice. Hence we need to remove **TotalPrice** from our dataset (so at the end it was just created to analyze the dataset efficiently).

Data Reduction and Clustering Model Training:

Problems faced and their solution approaches —

- Only a limited number of models can be trained on this dataset and those models also have their own limitations. For example, **hierarchical clustering can't be trained on type average linkage**(more efficient here) as the **dataset is very large** and even sampling data to a valuable extent doesn't help here.
- Sampling is a solution for this problem in many algorithms for large datasets. **Sampling takes x(assigned number) random samples from the dataset with replacement**. Although **sampling may increase biasness.** Even computing the performance of the model is another problem. For example, computation of **silhouette score** takes a lot of time to compute and consumes a large amount of RAM(**greater than 16GB**). So from our perspective and analysis we can say that using **sampling is a nice solution** to **deal with large datasets.**
- I plotted the cluster diagram only from 10,000 points(however I trained the model for all 5 lakh points but just tested for 10,000) because as I increase no. of points all points look very dense so it is very difficult to visualize the data.
- We cannot apply the **LDA algorithm** here because it needs the **labels** of train data and as this is an **unsupervised problem** we are not having the **labels of the dataset.**
- As the no. of features are more. So how to visualize a cluster using scatter plot. So we used features having highest eigenvalues as they conserve the most of the variance of the data.

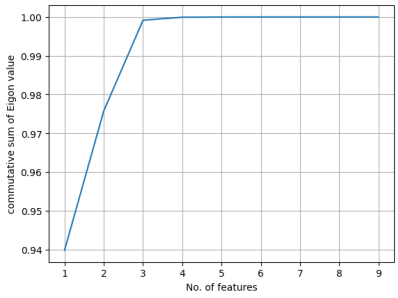
PCA(Principle Component Analysis):

- Since we are using the in-built PCA algorithm from sklearn library there is no need to scale the data. Then to **find the best no. of n_component** i used eigenvalues because eigenvalues gives us the measure of variances of features of a dataset .
- To **conserve the variance of our dataset** we have to take the no. of features so that about more than 90 percent of the eigenenergy is conserved . So by the values and plot we can see taking the top 3 features of the dataset is **conserving about 99.99 percent of variance (as a result of good preprocessing**). So from the above analysis we can say that **3 is the best value for the number of components** to be taken.

The values of **eigenenergy** are as follows.

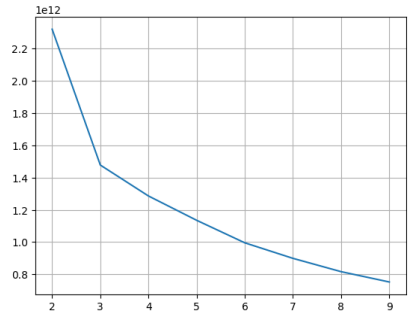
[0.93976082 0.97576186 0.9991659 0.99994079 0.99999613 0.99999844

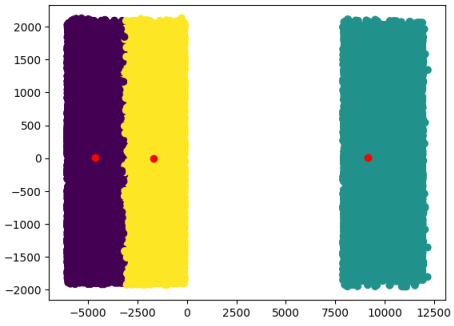
0.99999961 0.99999998 1.   ]

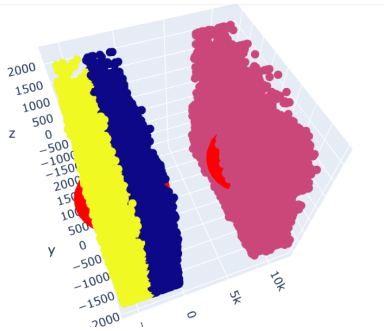


K Means:

- **k means it is working properly on the whole data set(no need of sampling).**
- As k means can be applied only on the continuous data not on the categorical data. So **applied pca with n_components equal to no. of columns**. and **converted my dataset into the same dataset but a continuous dataset.**
- Now I applied the **elbow method** to find the best no. of clusters. For this training the k means model and found parameters on that model and plotted that and the point where I am getting more steep is my best no. of clusters.(**got 3 as best no. of centroids.)**
- Then I applied the k means algorithm as my final model with **no. of clusters equal to 3**. And then plotted my clustering results.

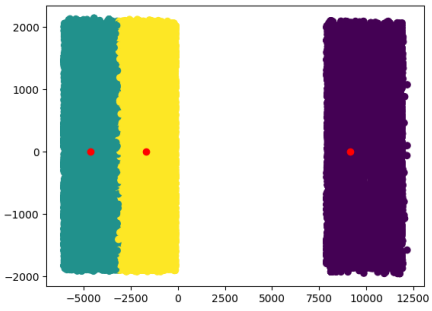


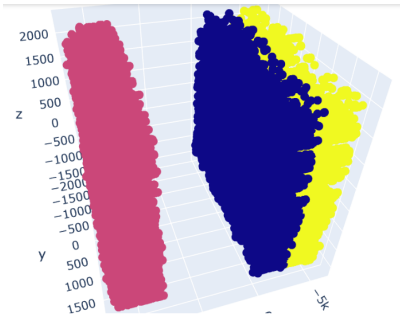




Then I plotted it without pca data . and got these results.

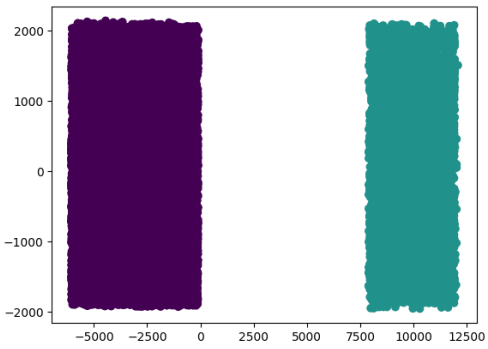
Why the cluster formed with pca and without pca almost the same: because variance is conserved (99.99 percent). Only data that is transformed in another direction.

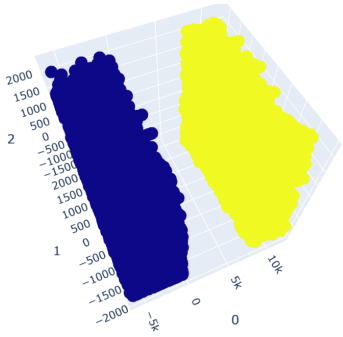




Hierarchical clustering:

- For this I used pca because this is a very time complex algorithm so it is not able to work on the whole dataset. (**here we saw the most use of pca algorithm**)
- For this clustering algorithm I sampled the data as it is a very time consuming process so I sampled (randomly generated) data points and then got the cluster results. It is **classifying the dataset in 2 classes**.
- Unable to plot dendrogram as ram limit exceeded.
- The cluster diagrams.

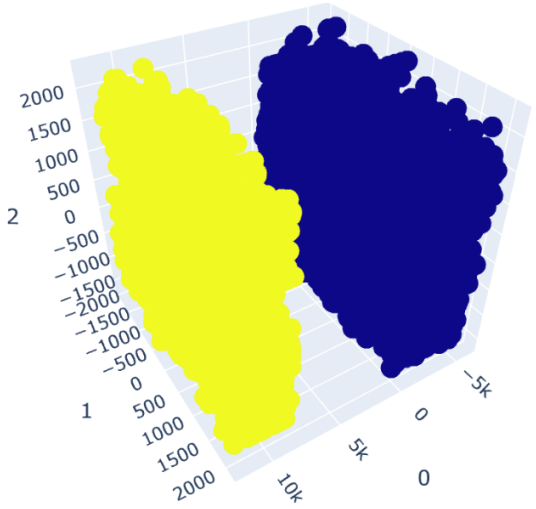




- I faced some problems like at all data points the ram limit exceeded . also time was around 5 hours. So I had to do **sampling** although it may affect the accuracy of our model but that is also necessary.

DBSCAN:

- This is one of the base algorithms for density based clustering. It can make clusters of different shapes and sizes and is very sensitive to outliers. To apply this algorithm first I have computed an optimal distance(maximum distance for the point to be included in the cluster) for input to the algorithm.
- Then I have trained the model on a sampled dataset and optimal distance calculated previously. Then to get a fair idea of the model I have plotted it and also calculated some applicable scores for it. From these analyses we get to know that this algorithm is not very efficient and is not working very well when compared to other algorithms.



Comparison Between Algorithms:

Silhouette range : (-1 to 1) 1 is best score and -1 is the worst score.

Calinski score: the ratio of the sum of between-clusters dispersion and of inter-cluster dispersion for all clusters

Davies score: ranged between 0 to 1 good model having higher value.

Algorithm	Silhouette score	Calinski score	Davies score	No. of clusters/centroids.	Some points	Some points
PCA	.....	.....	.....	3	Best as it reduced data from 10 to 3 features.	Conserved 99 percent variance.
K Means (same for pca and non pca)	0.4f8	572051.43	0.78	3	Good model	Without sampling
Hierarchical	0.64	570912.35	0.78	2	Best model	Used sampling
DBSCAN	0.64	136645.27	0.30	2	Best model	Used sampling