Customer segmentation

Mile stone 2 Report

STUDENT'S NAME

Doha Abdelaal Zakaria

Yosr Mohammed Abdel Haleem

Salma Mohammed Hamed

Menna Abdel Rahim Ali

Yasmien Ahmed Abdel Hamied

Omnia Ahmed Mustafa

Pattern recognition
Dr. Manal Tantawy

1) Data analysis:

1.1) Data type of each column:

```
data.info()
```

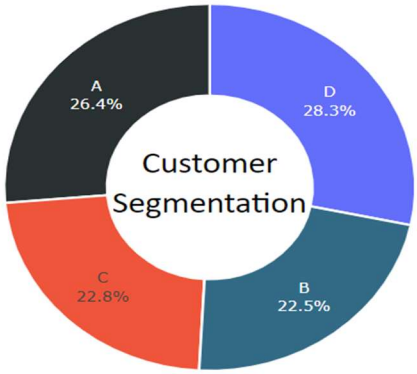
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7165 entries, 0 to 7164
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   ID               7165 non-null   int64
 1   Gender           7165 non-null   object
 2   Ever_Married     7044 non-null   object
 3   Age              7165 non-null   int64
 4   Graduated        7096 non-null   object
 5   Profession       7060 non-null   object
 6   Work_Experience  6440 non-null   float64
 7   Spending_Score   7165 non-null   object
 8   Family_Size      6864 non-null   float64
 9   Var_1            7101 non-null   object
 10  Segmentation     7165 non-null   object
dtypes: float64(2), int64(2), object(7)
memory usage: 615.9+ KB
```

1.2) We count the null values in each column

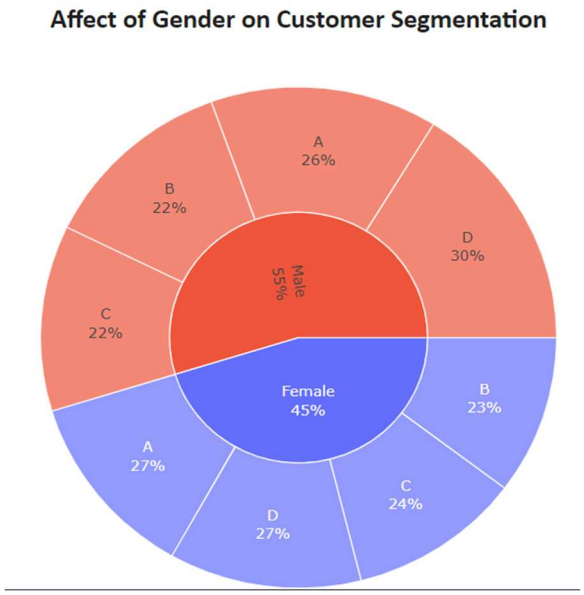
```
print(data.isna().sum())
```

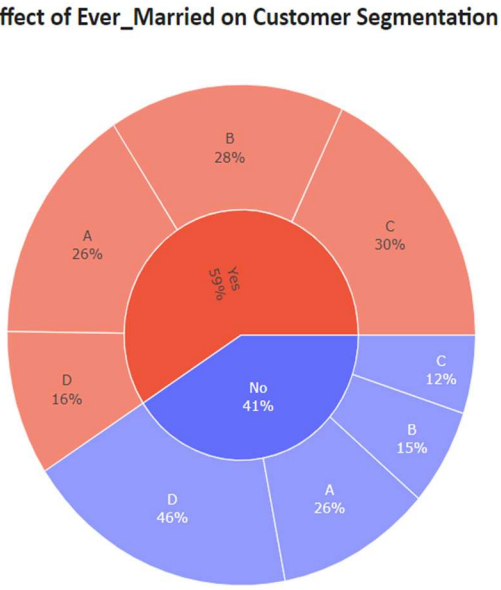
```
ID                  0
Gender              0
Ever_Married      121
Age                 0
Graduated          69
Profession        105
Work_Experience   725
Spending_Score      0
Family_Size       301
Var_1              64
Segmentation        0
dtype: int64
```

1.3) Plot that describe the percentage of each segment in the data

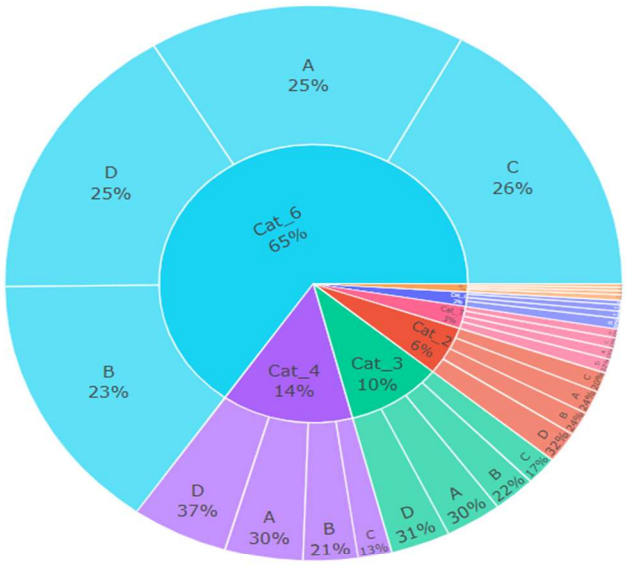


1.4) The relation between each column and segmentation column

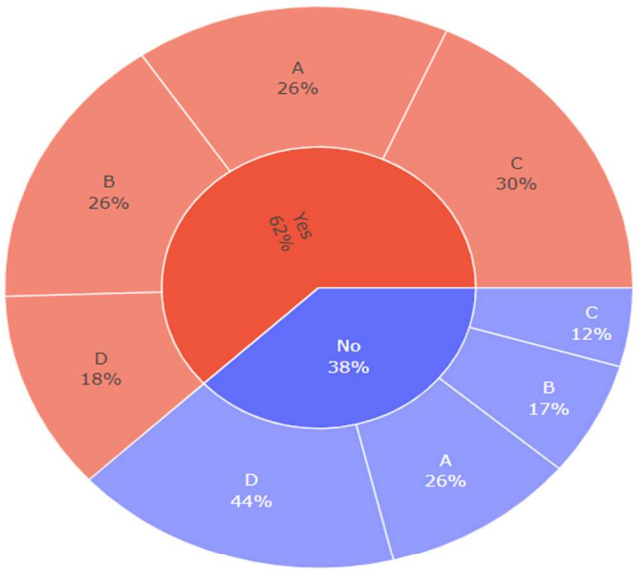




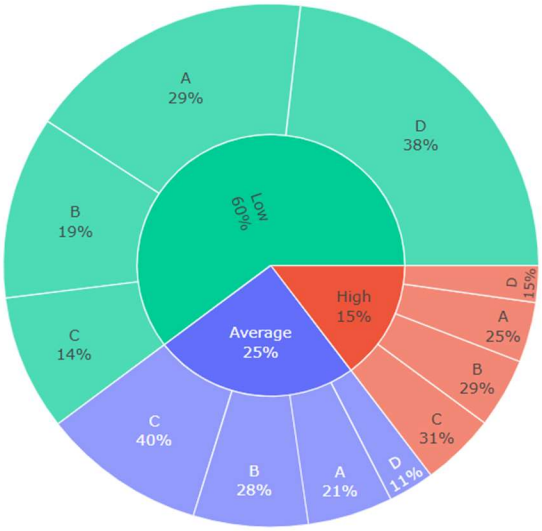
Affect of Var_1 on Customer Segmentation



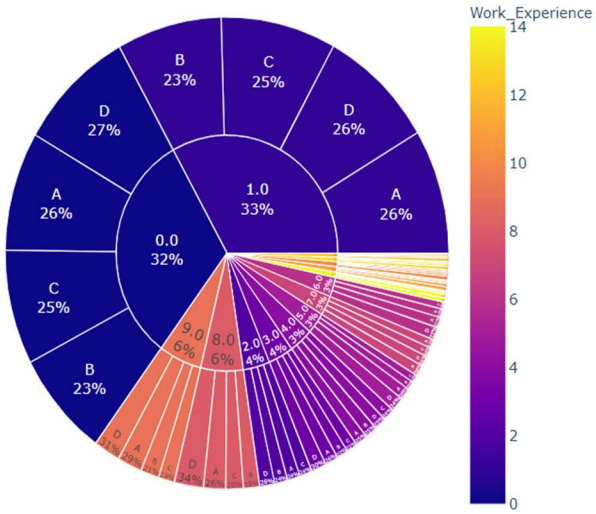
Affect of Graduated on Customer Segmentation



Affect of Spending_Score on Customer Segmentation



Affect of Work_Experience on Customer Segmentation



2) Preprocessing techniques:

2.1) Filling the missing values:

We used the mode method to fill the missing values (used in train data). Mode fills the missing values with most common values in the dataset. It is applied to the column that has null values like

And we dropped other rows. (Used in test data)

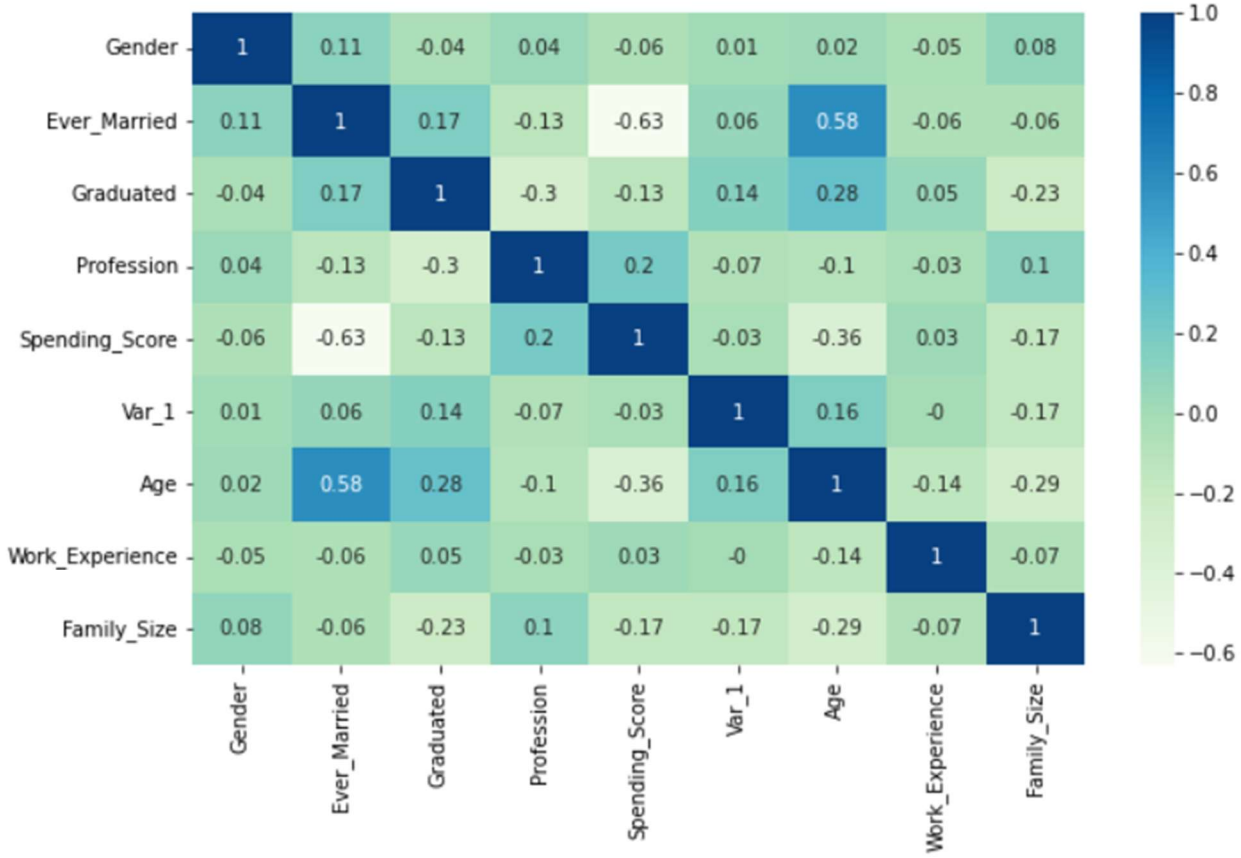
2.2) Using Label encoding:
It is a method used to strings values to encode them to numerical values.
Like Gender, Ever_Married, Graduated, Var_1, professions, Spending_Score

2.3) Dropping columns with low-correlation: work_experience/Ever married
2.4) Standardization: Applied to some numerical column such as Age, Family_Size

3) Features used /discarded:

3.1) used features: Gender, Ever_Married, Graduated, Spending_Score, Var_1, Age, Family_Size, professions
3.2) Discarded features: 'Work_Experience'

3.2) Correlation:



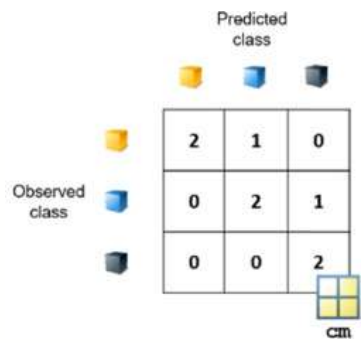
4) Classification techniques:

3.1) XGB Classifier
3.2) Gradient boosting.
3.4) SVM
3.5) Logistic Regression
3.5)

5) Training & testing size:

The data was divided into 20% for testing, 80 % training.

6) Confusion matrix:



AS an example

Confusion matrix

```
from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(Y_test, prediction)
print(cm)
accuracy_score(Y_test, prediction)
```

```
[[155  49  57  75]
 [ 92  68  78  50]
 [ 55  42 169  38]
 [ 73  31  17 262]]
```

7) Decision boundary: