Data Collection and Preprocessing Phase

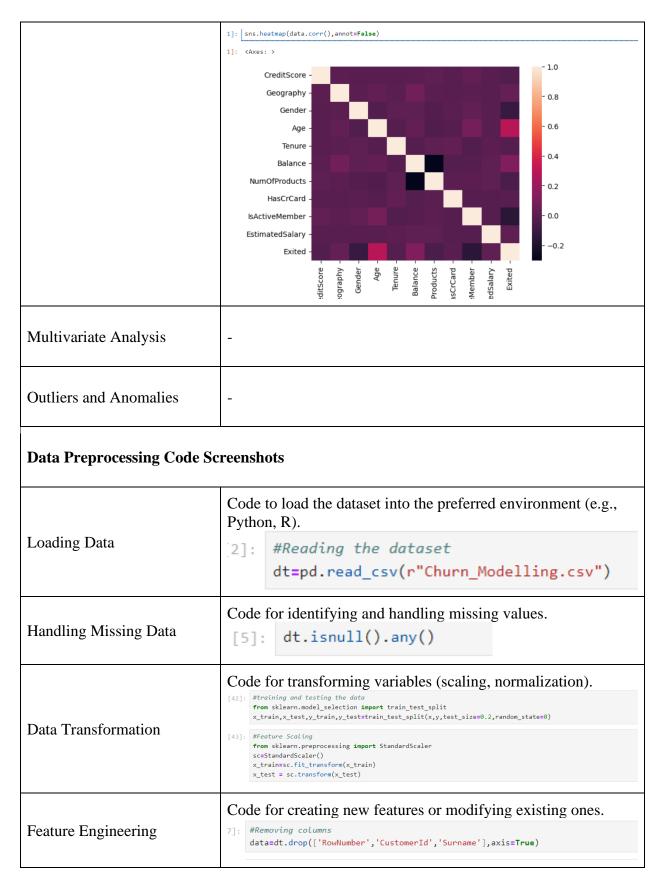| Date	15 July 2024
Team ID	739699
Project Title	Telecom Customer Churn Prediction
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	Basic statistics, dimensions, and structure of the data.
Univariate Analysis	Exploration of individual variables (mean, median, mode, etc..)
Bivariate Analysis	Relationships between two variables (correlation, scatter plots).

```
1]: sns.heatmap(data.corr(),annot=False)

1]: <Axes: >
```


Multivariate Analysis	-
Outliers and Anomalies	-

Data Preprocessing Code Screenshots

Loading Data	Code to load the dataset into the preferred environment (e.g., Python, R).
	
	 ```[2]: #Reading the dataset```
	 ```dt=pd.read_csv(r"Churn_Modelling.csv")```

Handling Missing Data	Code for identifying and handling missing values. ```[5]: dt.isnull().any()```

Data Transformation	Code for transforming variables (scaling, normalization).
	
	 ```[42]: #training and testing the data```
	 ```from sklearn.model_selection import train_test_split```
	 ```x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)```
	
	 ```[43]: #Feature Scaling```
	 ```from sklearn.preprocessing import StandardScaler```
	 ```sc=StandardScaler()```
	 ```x_train=sc.fit_transform(x_train)```
	 ```x_test = sc.transform(x_test)```

Feature Engineering	Code for creating new features or modifying existing ones.
	
	 ```7]: #Removing columns```
	 ```data=dt.drop(['RowNumber','CustomerId','Surname'],axis=True)```

Save Processed Data	Code to save the cleaned and processed data for future use.

```
[23]:  #LabelEncoding
```

```
[24]:  from sklearn.preprocessing import LabelEncoder
       le=LabelEncoder()
```

```
[25]:  #data["CreditScore"]=le.fit_transform(data["CreditScore"])
       data["Geography"]=le.fit_transform(data["Geography"])
       data["Gender"]=le.fit_transform(data["Gender"])
       #data["Age"]=le.fit_transform(data["Age"])
       #data["Tenure"]=le.fit_transform(data["Tenure"])
       #data["Balance"]=le.fit_transform(data["Balance"])
       #data["NumOfProducts"]=le.fit_transform(data["NumOfProducts"])
       #data["HasCrCard"]=le.fit_transform(data["HasCrCard"])
       #data["IsActiveMember"]=le.fit_transform(data["IsActiveMember"])
       #data["EstimatedSalary"]=le.fit_transform(data["EstimatedSalary"])
       #data["Exited"]=le.fit_transform(data["Exited"])
```