



Data Collection and Preprocessing Phase

Section	Des	scripti	ion										
				s, dir	nensi	ons	, and	d struc	ture o	of the d	ata.		
	[22]:	data.describ	e()									•	
		Cred	itScore	Age	Tenure	В	alance M	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	
				- 5	10000,000000			10000.000000		10000,000000	10000,000000	10000.000000	
	1			38.921800	5,012800	76485.	889288	1,530200	0.70550	0.515100	100090.239881	0.203700	
				10.487806	2.892174	62397	405202	0.581654	0.45584	0.499797	57510.492818	0.402769	
	1			18.000000	0.000000		000000	1.000000	0.00000	0.000000	11.580000	0.000000	
				32.000000	3.000000		000000	1.000000	0.00000	0.000000		0.000000	
				37.000000	5.000000			1.000000	1.00000	1.000000		0.000000	
				44.000000	7.000000			2.000000		1.000000		0.000000	
	1			92.000000	10.000000			4.000000	1.00000	1.000000		1.000000	
		max 650	000000	52.000000	10.000000	2300303	090000	4.000000	1.00000	1.000000	199992.400000	1.00000	
	[30]:	data.corr()											
Data Overview	[30]:	uaca.com()	CreditScore		Gender	****	Tenure		000-1-1-1		Member Estimated	Salary Exited	
	13071	CreditScore				-0.003965	0.000842		0.012238			01384 -0.027094	
	1	Geography	0.00788			0.022812	0.003739		0.003972			01369 0.035943	
		Gende				-0.027544	0.014733		-0.021859			08112 -0.106512	
		Age	-0.003969	5 0.022812	-0.027544	1.000000	-0.009997	0.028308	-0.030680	-0.011721	0.085472 -0.0	07201 0.285323	
		Tenure	0.00084	0.003739	0.014733	-0.009997	1.000000	-0.012254	0.013444	0.022583	0.028362 0.0	07784 -0.014001	
		Balance	0.00626	0.069408	0.012087	0.028308	-0.012254	1.000000	-0.304180	-0.014858 -	0.010084 0.0	12797 0.118533	
		NumOfProducts	0.01223	0.003972	-0.021859	-0.030680	0.013444	-0.304180	1.000000	0.003183	0.009612 0.0	14204 -0.047820	
		HasCrCard	-0.00545	9 -0,008523	0.005766	-0.011721	0.022583	-0.014858	0.003183	1.000000 -	0.011866 -0.0	09933 -0.007138	
		IsActiveMember				0.085472	-0.028362		0.009612			11421 -0.156128	
		EstimatedSalary				-0.007201	0.007784		0.014204			00000 0.012097	
		Exited	-0.02709	0.035943	-0.106512	0.285323	-0.014001	0.118533	-0.047820	-0.007138	0.156128 0.0	12097 1.000000	
	_					_							
Univariate Analysis	Exp	Exploration of individual variables (mean, median, mode, etc)											
_	1												





Bivariate Analysis	Relationships between two variables (correlation, scatter plots).
--------------------	---

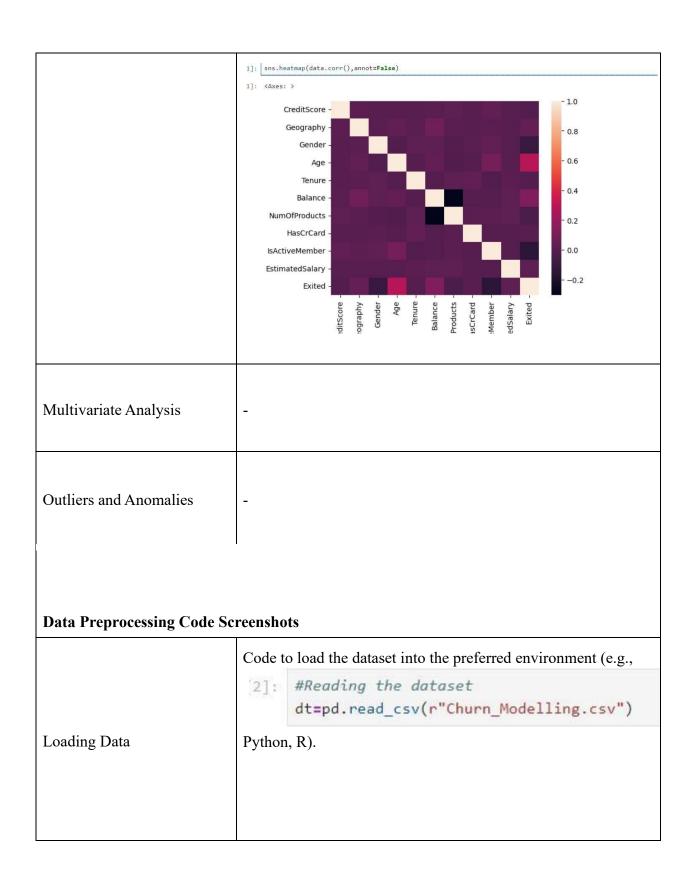
Date	15 July 2024
Team ID	740671
Project Title	Telecom Customer Churn Prediction
Maximum Marks	6 Marks

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.











Handling Missing Data	[5]: dt.isnull().any() Code for identifying and handling missing values.
Data Transformation	[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) Code for transforming variables (scaling, normalization).
Feature Engineering	7]: #Removing columns data=dt.drop(['RowNumber', 'CustomerId', 'Surname'], axis=True) Code for creating new features or modifying existing ones.





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["Geography"]=le.fit_transform(data["Geography"]) #data["Geoder"]) #data["Age"]=le.fit_transform(data["Age"]) #data["Age"]=le.fit_transform(data["Tenure"]) #data["Balance"]-le.fit_transform(data["Tenure"]) #data["Balance"]-le.fit_transform(data["NumOfProducts"]) #data["NumOfProducts"]-le.fit_transform(data["IsActiveNember"]) #data["IsActiveNember"]-le.fit_transform(data["IsActiveNember"]) #data["EstimatedSalary"]-le.fit_transform(data["EstimatedSalary"]) #data["Exited"]=le.fit_transform(data["Exited"])