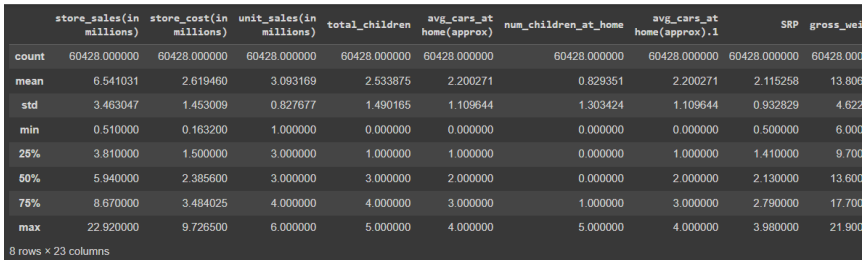
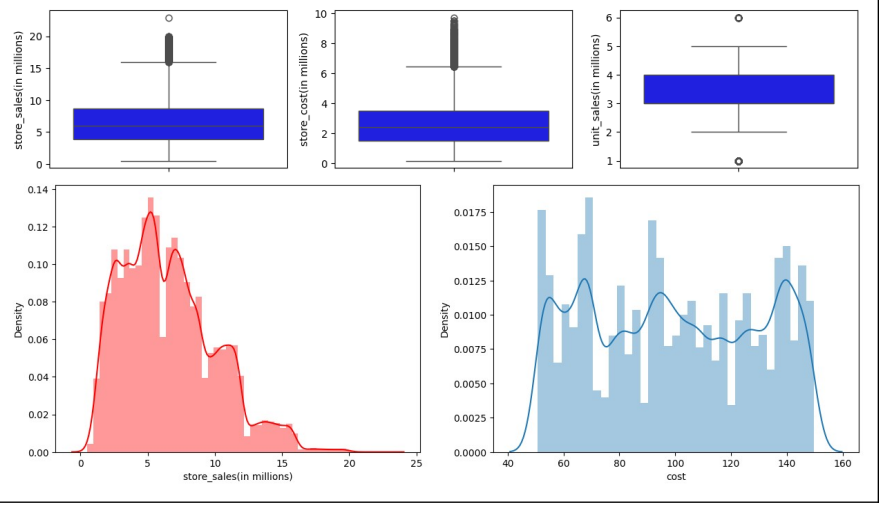
Data Collection and Preprocessing Phase

Date	4th July 2024
Team ID	739983
Project Title	Cost Prediction of Acquiring a Customer
Maximum Marks	6 Marks

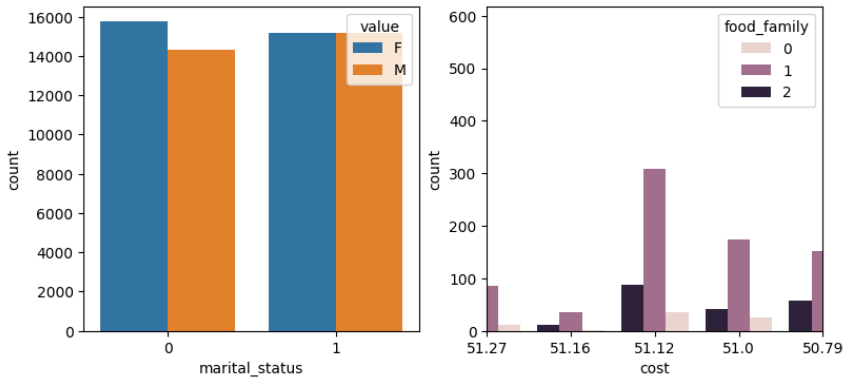
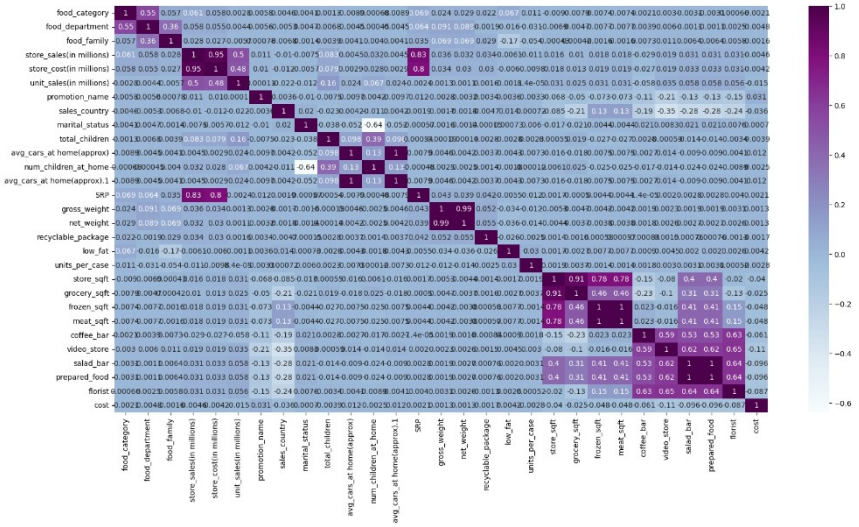
Data Exploration and Preprocessing Template

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	Dimensions: (60428, 40)
Univariate Analysis	
Bivariate Analysis	

	store_sales(in millions)	store_cost(in millions)	unit_sales(in millions)	total_children	avg_cars_at home(approx)	num_children_at_home	avg_cars_at home(approx).1	SRP	gross_wei
count	60428.000000	60428.000000	60428.000000	60428.000000	60428.000000	60428.000000	60428.000000	60428.000000	60428.000
mean	6.541031	2.619460	3.093169	2.533875	2.200271	0.829351	2.200271	2.115258	13.806
std	3.463047	1.453009	0.827677	1.490165	1.109644	1.303424	1.109644	0.932829	4.622
min	0.510000	0.163200	1.000000	0.000000	0.000000	0.000000	0.000000	0.500000	6.000
25%	3.810000	1.500000	3.000000	1.000000	1.000000	0.000000	1.000000	1.410000	9.700
50%	5.940000	2.385600	3.000000	3.000000	2.000000	0.000000	2.000000	2.130000	13.600
75%	8.670000	3.484025	4.000000	4.000000	3.000000	1.000000	3.000000	2.790000	17.700
max	22.920000	9.726500	6.000000	5.000000	4.000000	5.000000	4.000000	3.980000	21.900

8 rows × 23 columns

	
Multivariate Analysis	
Outliers and Anomalies	-

Data Preprocessing Code Screenshots

Loading Data	`df = pd.read_csv('/content/media prediction and its cost.csv')`
Handling Missing Data	`df.isnull().any()`

Data Transformation	`from sklearn.preprocessing import LabelEncoder`
	
	`def encoder(data, variable):  # Corrected parameter name to 'variable'`
	`    lb = LabelEncoder()`
	`    df[variable] = lb.fit_transform(df[variable])`
	`    return lb`

Feature Engineering	food_category_le = encoder(df,'food_category')
	brand_name_le = encoder(df,'brand_name')
	food_department_le = encoder(df,'food_department')
	food_family_le = encoder(df,'food_family')
	promotion_name_le = encoder(df,'promotion_name')
	store_city_le = encoder(df,'store_city')
	#unit_per_case_le = encoder(df,'unit_per_case')
	net_weight_le = encoder(df,'net_weight')
	sales_le = encoder(df,'sales_country')
	martial_le = encoder(df,'marital_status')

Save Processed Data	import pickle pickle.dump(rf,open('customers.pkl','wb')) pickle.dum p(food_category_le,open('food_category_le.pkl','wb')) pickle.dum p(brand_name_le,open('brand_name_le.pkl','wb')) pickle.dum p(promotion_name_le,open('promotion_name_le.pkl','w b')) pickle.dum p(store_city_le,open('store_city_le.pkl','wb'))