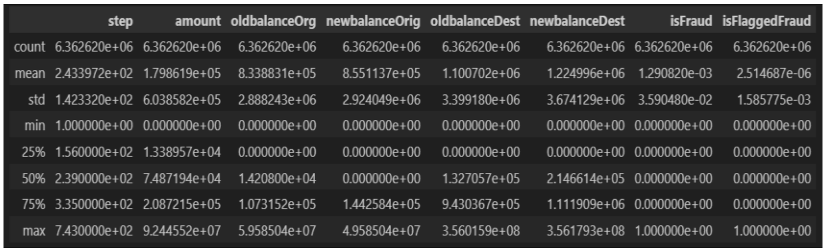
Data Collection and Preprocessing Phase

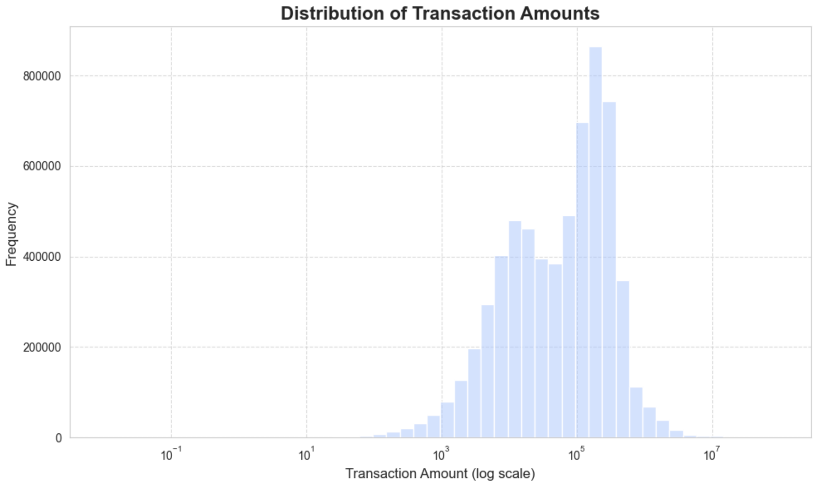
Date	16 June 2025
Team ID	SWTID1749710444
Project Title	Online Payment Fraud Detection using ML
Maximum Marks	6 Marks

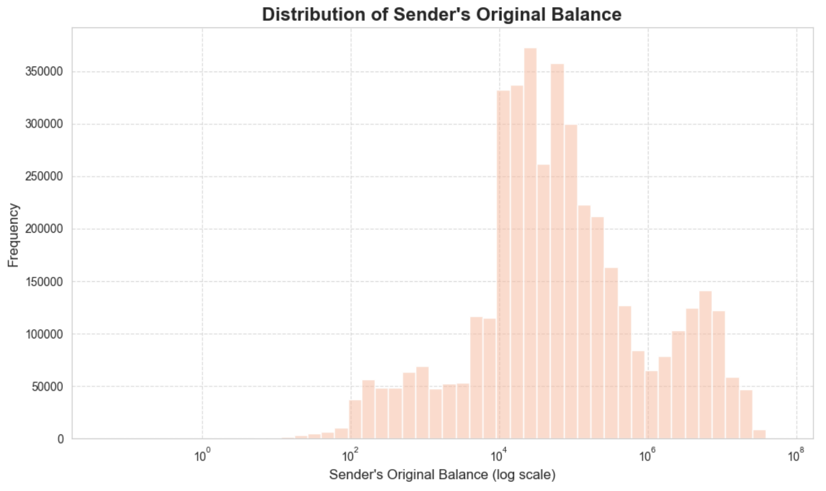
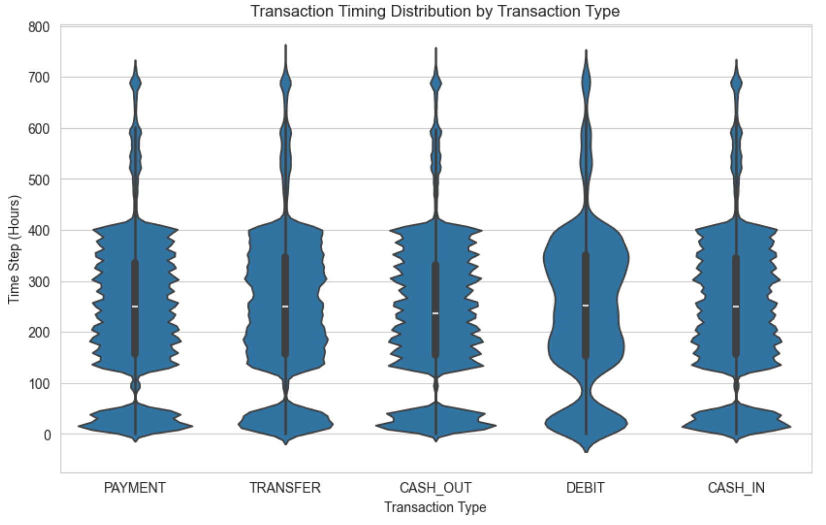
Data Exploration and Preprocessing

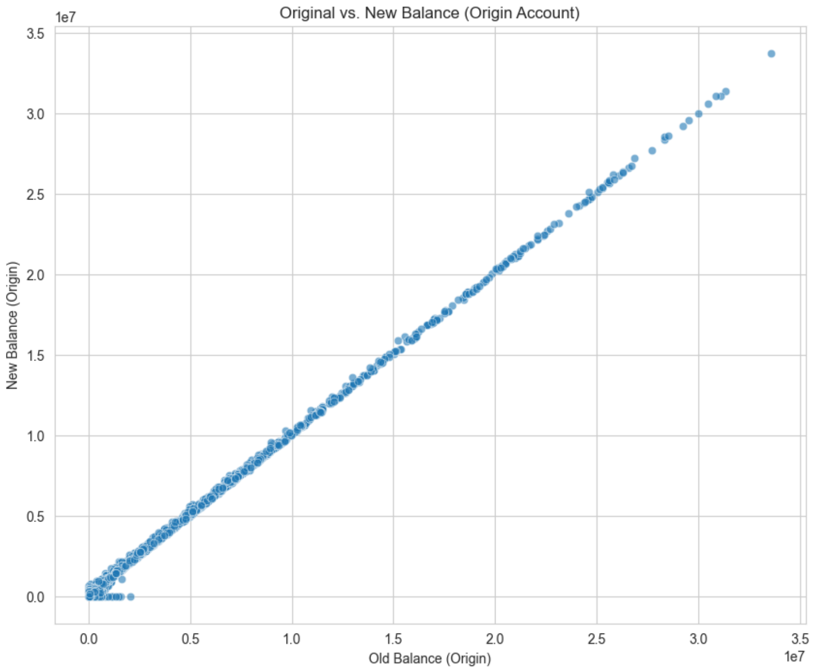
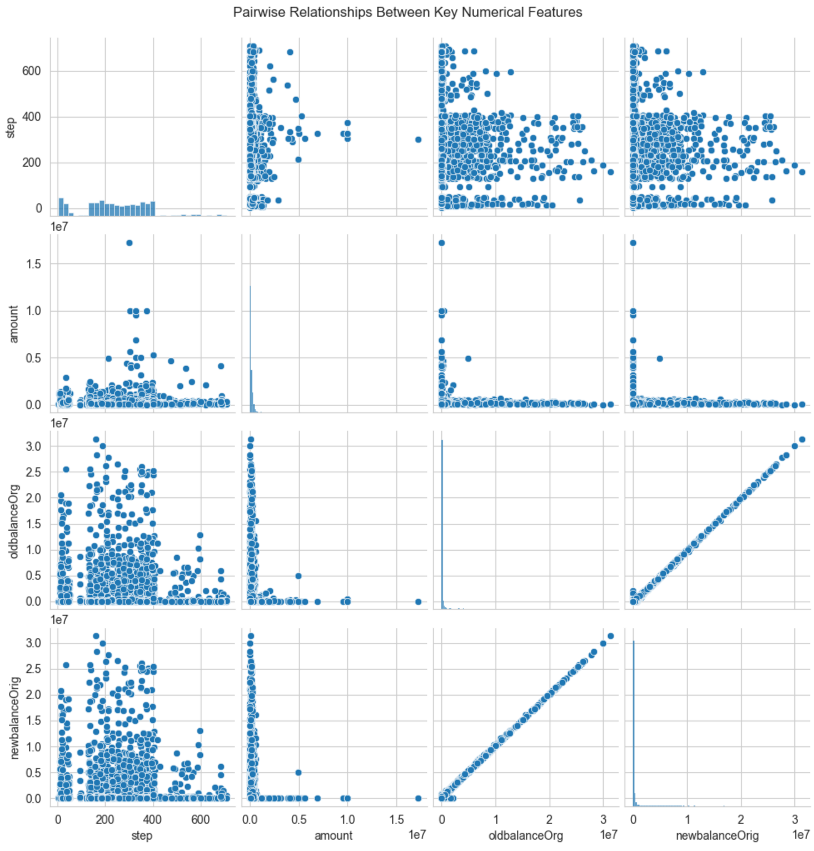
Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	

	step	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
count	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06
mean	2.433972e+02	1.798619e+05	8.338831e+05	8.551137e+05	1.100702e+06	1.224996e+06	1.290820e-03	2.514687e-06
std	1.423320e+02	6.038582e+05	2.888243e+06	2.924049e+06	3.399180e+06	3.674129e+06	3.590480e-02	1.585775e-03
min	1.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
25%	1.560000e+02	1.338957e+04	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
50%	2.390000e+02	7.487194e+04	1.420800e+04	0.000000e+00	1.327057e+05	2.146614e+05	0.000000e+00	0.000000e+00
75%	3.350000e+02	2.087215e+05	1.073152e+05	1.442584e+05	9.430367e+05	1.111909e+06	0.000000e+00	0.000000e+00
max	7.430000e+02	9.244552e+07	5.958504e+07	4.958504e+07	3.560159e+08	3.561793e+08	1.000000e+00	1.000000e+00

Univariate Analysis	

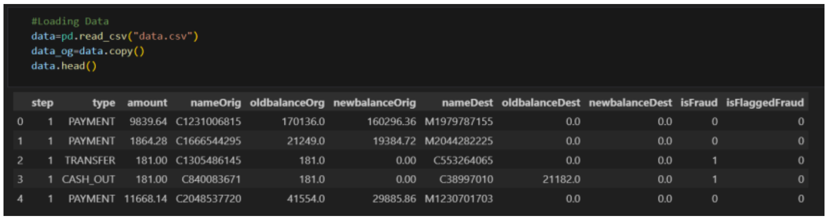
	
Bivariate Analysis	

	
Multivariate Analysis	
Outliers and Anomalies	Due to the extremely wide distribution and heavy-tailed nature of financial transaction data in this dataset, traditional outlier

detection methods are not effective. The data spans several orders of magnitude (e.g., transaction amounts range from $0 to $92+ million), making it difficult to distinguish between legitimate large transactions and true anomalies using standard statistical methods.

Data Preprocessing Code Screenshots

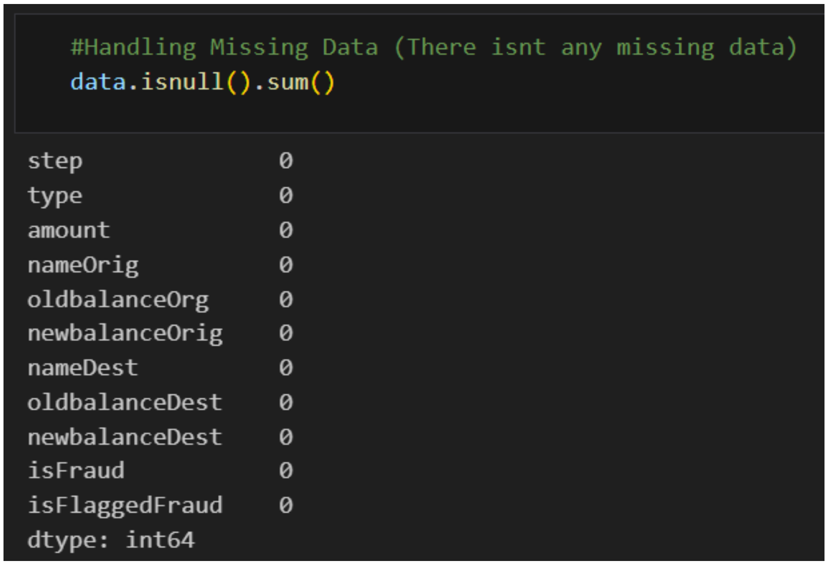
Loading Data



```
#Loading Data
data=pd.read_csv("data.csv")
data_og=data.copy()
data.head()
```

```
   step      type    amount     nameOrig  oldbalanceOrg  newbalanceOrig     nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
0     1   PAYMENT   9839.64  C1231006815       170136.0       160296.36  M1979787155             0.0             0.0        0               0
1     1   PAYMENT   1864.28  C1666544295        21249.0        19384.72  M2044282225             0.0             0.0        0               0
2     1  TRANSFER    181.00  C1305486145          181.0            0.00   C553264065             0.0             0.0        1               0
3     1  CASH_OUT    181.00   C840083671          181.0            0.00    C38997010         21182.0             0.0        1               0
4     1   PAYMENT  11668.14  C2048537720        41554.0        29885.86  M1230701703             0.0             0.0        0               0
```

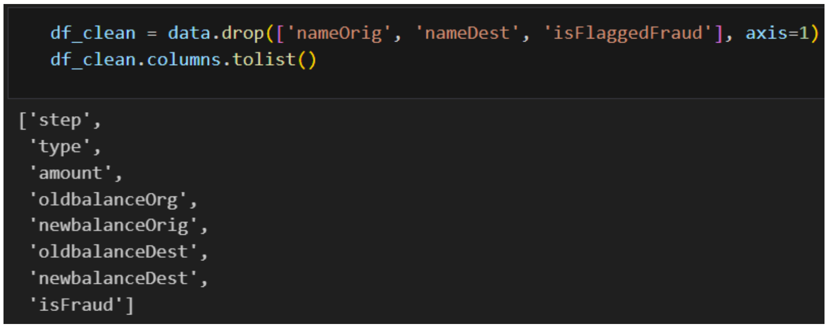
Handling Missing Data



```
#Handling Missing Data (There isnt any missing data)
data.isnull().sum()
```

```
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

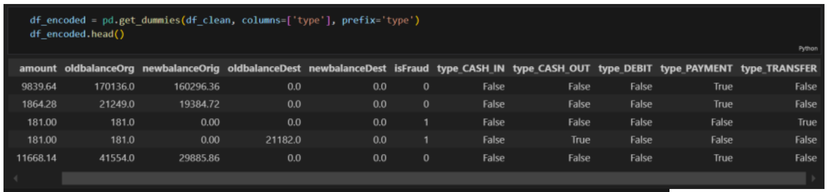
Data Transformation



```
df_clean = data.drop(['nameOrig', 'nameDest', 'isFlaggedFraud'], axis=1)
df_clean.columns.tolist()
```

```
['step',
 'type',
 'amount',
 'oldbalanceOrg',
 'newbalanceOrig',
 'oldbalanceDest',
 'newbalanceDest',
 'isFraud']
```

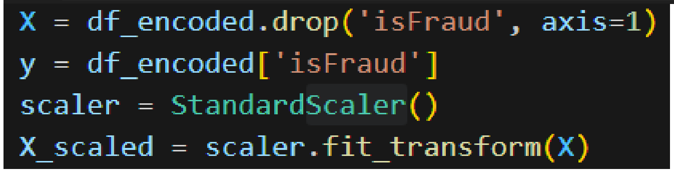
<table>
<tr>
<td></td>
<td>



```python
df_encoded = pd.get_dummies(df_clean, columns=['type'], prefix='type')
df_encoded.head()
```

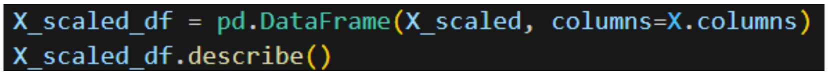
Python

amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	type_CASH_IN	type_CASH_OUT	type_DEBIT	type_PAYMENT	type_TRANSFER
9839.64	170136.0	160296.36	0.0	0.0	0	False	False	False	True	False
1864.28	21249.0	19384.72	0.0	0.0	0	False	False	False	True	False
181.00	181.0	0.00	0.0	0.0	1	False	False	False	False	True
181.00	181.0	0.00	21182.0	0.0	1	False	True	False	False	False
11668.14	41554.0	29885.86	0.0	0.0	0	False	False	False	True	False



```python
X = df_encoded.drop('isFraud', axis=1)
y = df_encoded['isFraud']
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

</td>
</tr>
<tr>
<td>Feature Engineering</td>
<td>New features such as balance changes and ratios were considered but not used in the baseline due to time/resource constraints.</td>
</tr>
<tr>
<td>Save Processed Data</td>
<td>



```python
X_scaled_df = pd.DataFrame(X_scaled, columns=X.columns)
X_scaled_df.describe()
```

</td>
</tr>
</table>