Ashwin Ravi
3122 21 5001 014
CSE-A

A2:- Loan Amount Prediction Using Linear Regression and Visualization

Aim

To Develop a python program to predict the loan amount to be sanctioned using Linear Regression (LR) Model using Scikit-learn library. Visualize the features from the dataset and interpret the results obtained by the model using Matplotlib library.

Code and output

Loading the dataset:

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler,LabelEncoder
from sklearn.linear_model import LinearRegression
from sklearn.metrics import
accuracy_score,classification_report,confusion_matrix,mean_squared_error,r2
_score
import matplotlib.pyplot as plt
import seaborn as sns
#Load train dataset
train_df=pd.read_csv('/content/train.csv')
train_df.head(5)
```

	Customer ID	Name	Gender	Age	Income (USD)	Income Stability	Profession	Type of Employment	Location	Loan Amount Request (USD)	...	Credit Score	No. of Defaults	Has Active Credit Card	Property ID	Property Age	Property Type	Property Location	Co-Applicant	Property Price	Loan Sanction Amount (USD)
0	C-36995	Frederica Shealy	F	56	1933.05	Low	Working	Sales staff	Semi-Urban	72809.58	...	809.44	0	NaN	746	1933.05	4	Rural	1	119933.46	54607.18
1	C-33999	America Calderone	M	32	4952.91	Low	Working	NaN	Semi-Urban	46837.47	...	780.40	0	Unpossessed	608	4952.91	2	Rural	1	54791.00	37469.98
2	C-3770	Rosetta Verne	F	65	988.19	High	Pensioner	NaN	Semi-Urban	45593.04	...	833.15	0	Unpossessed	546	988.19	2	Urban	0	72440.58	36474.43
3	C-26480	Zoe Chitty	F	65	NaN	High	Pensioner	NaN	Rural	80057.92	...	832.70	1	Unpossessed	890	NaN	2	Semi-Urban	1	121441.51	56040.54
4	C-23459	Afton Venema	F	31	2614.77	Low	Working	High skill tech staff	Semi-Urban	113858.89	...	745.55	1	Active	715	2614.77	4	Semi-Urban	1	208567.91	74008.28

5 rows × 24 columns

Ashwin Ravi
3122 21 5001 014
CSE-A

Pre-Processing the data (Handling missing values, Encoding, Normalization,Standardization).

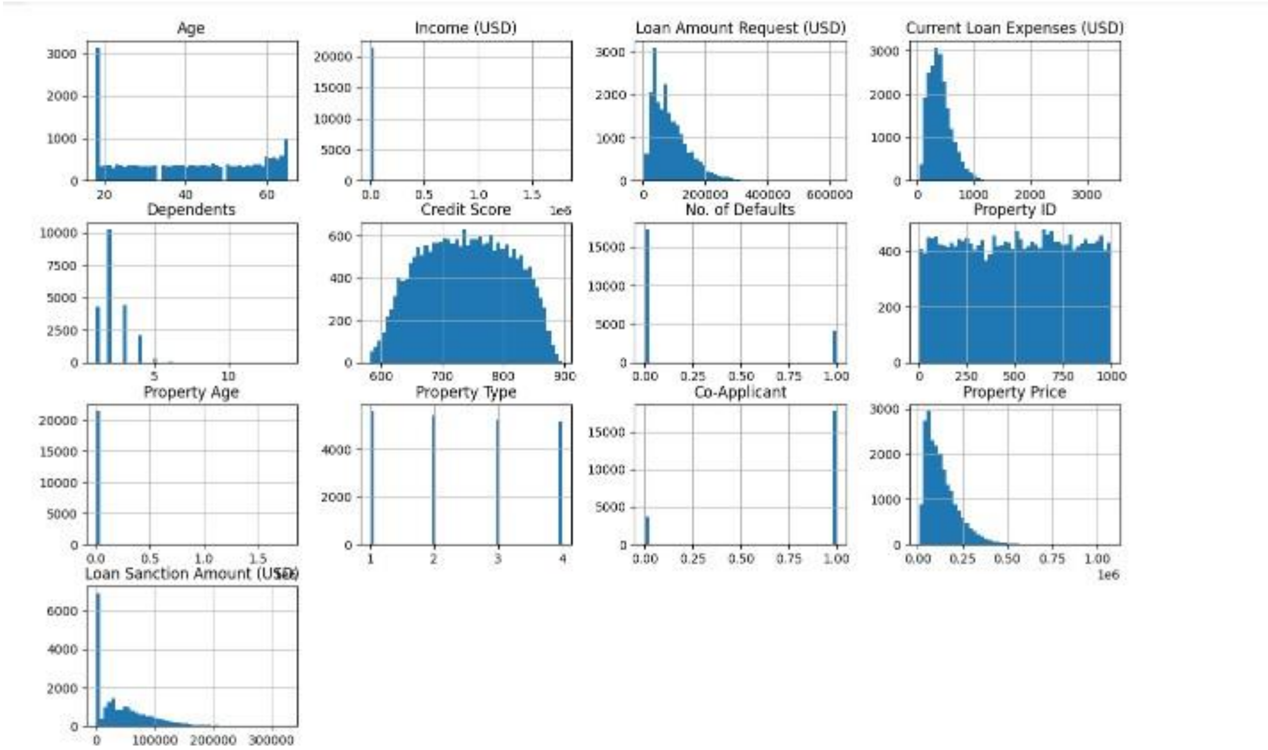
```
#Preprocessing

indexes = train_df[train_df['Loan Sanction Amount (USD)'].isna()].index

#Loan Amount empty is removed(invalid)
train_df.drop(index=indexes, inplace=True)
#Loan Amount < 0
indexes1 = train_df[train_df['Loan Sanction Amount (USD)'] < 0].index
train_df.drop(index=indexes1,inplace=True)
#Coapplicant <  0 is made 0

indexes10 = train_df[train_df['Co-Applicant']<0].index
train_df.drop(index=indexes10,inplace=True)

#property age empty is removed
indexes11 = train_df[train_df['Property Age'].isna()].index
train_df.drop(index=indexes11,inplace=True)

#credit score null is
indexes1 = train_df[train_df['Credit Score'].isna()].index
train_df.drop(index=indexes1,inplace=True)

indexes2 = train_df[train_df['Income (USD)'].isna()].index
train_df.drop(index=indexes2,inplace=True)
indexes4 = train_df[train_df['Current Loan Expenses (USD)'].isna()].index
train_df.drop(index=indexes4,inplace=True)

indexes5 = train_df[train_df['Current Loan Expenses (USD)'] < 0].index
train_df.drop(index=indexes5,inplace=True)
indexes14 = train_df[train_df['Dependents'].isna()].index
train_df.drop(index=indexes14,inplace=True)
indexes3 = train_df[train_df['Property Price']<0].index

train_df.drop(index=indexes3,inplace=True)
```

	Customer ID	Name	Gender	Age	Income (USD)	Income Stability	Profession	Type of Employment	Location	Loan Amount Request (USD)	...	Credit Score	No. of Defaults	Has Active Credit Card	Property ID	Property Age	Property Type	Property Location	Co-Applicant	Property Price	Loan Sanction Amount (USD)
0	C-36995	Frederica Shealy	F	56	1933.05	Low	Working	Sales staff	Semi-Urban	72809.58	...	809.44	0	NaN	746	1933.05	4	Rural	1	119933.46	54607.18
1	C-33999	America Calderone	M	32	4952.91	Low	Working	NaN	Semi-Urban	46837.47	...	780.40	0	Unpossessed	608	4952.91	2	Rural	1	54791.00	37469.98
2	C-3770	Rosetta Verne	F	65	988.19	High	Pensioner	NaN	Semi-Urban	45593.04	...	833.15	0	Unpossessed	546	988.19	2	Urban	0	72440.58	36474.43
5	C-17688	Polly Crumpler	F	60	1234.92	Low	State servant	Secretaries	Rural	34434.72	...	684.12	1	Inactive	491	1234.92	2	Rural	1	43146.82	22382.57
6	C-23855	Nathalie Olivier	M	43	2361.56	Low	Working	Laborers	Semi-Urban	152561.34	...	637.29	0	Unpossessed	227	2361.56	1	Semi-Urban	1	221050.80	0.00
8	C-26934	Kenny Ankrom	F	38	1296.07	Low	Working	Cooking staff	Rural	35141.99	...	705.29	1	Active	241	1296.07	4	Rural	1	54903.44	22842.29
9	C-24944	Barbie Goetsch	M	18	1546.17	Low	Working	Laborers	Rural	42091.29	...	613.24	0	Unpossessed	883	1546.17	2	Urban	1	67993.43	0.00
10	C-40801	Laree Staton	M	18	2416.86	Low	State servant	Core staff	Semi-Urban	25765.72	...	652.41	0	Active	325	2416.86	2	Rural	1	32423.71	16747.72
11	C-37677	Xenia Browder	F	39	2719.74	Low	Commercial associate	High skill tech staff	Semi-Urban	20879.96	...	646.21	0	NaN	196	2719.74	2	Rural	0	33568.47	0.00
12	C-30073	Brinda Vaz	F	48	777.25	Low	Working	NaN	Semi-Urban	96080.60	...	764.11	0	Active	678	777.25	1	Semi-Urban	1	146073.26	67256.42
13	C-34993	Brendon Swanson	F	43	997.25	Low	Working	NaN	Rural	48894.06	...	726.28	0	NaN	578	997.25	4	Rural	1	80607.40	34225.84
14	C-35991	Gricelda Lamphere	M	61	1664.52	Low	Working	Core staff	Semi-Urban	72448.95	...	781.51	0	Active	500	1664.52	4	Urban	1	113464.01	54336.71
15	C-39716	Marietta Alverson	F	54	3716.54	Low	Working	Drivers	Rural	116487.58	...	749.33	1	Active	458	3716.54	1	Urban	1	194442.39	75716.93
16	C-45549	Elidia Mccluney	F	61	2077.42	Low	Working	Realty agents	Semi-Urban	70815.01	...	779.55	0	Inactive	395	2077.42	4	Rural	1	102502.20	0.00

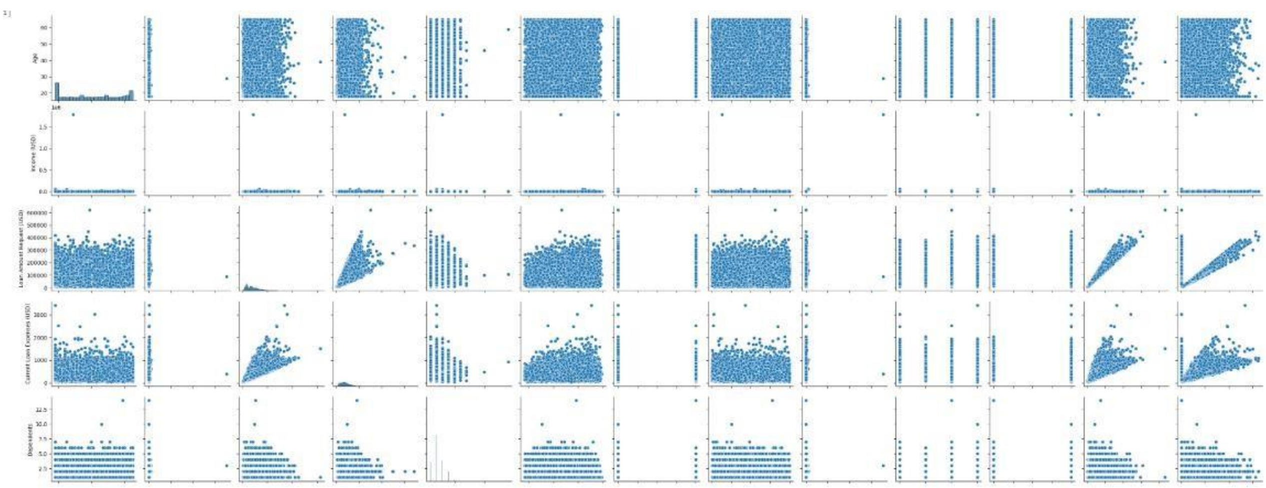
Ashwin Ravi
3122 21 5001 014
CSE-A

Exploratory Data Analysis.

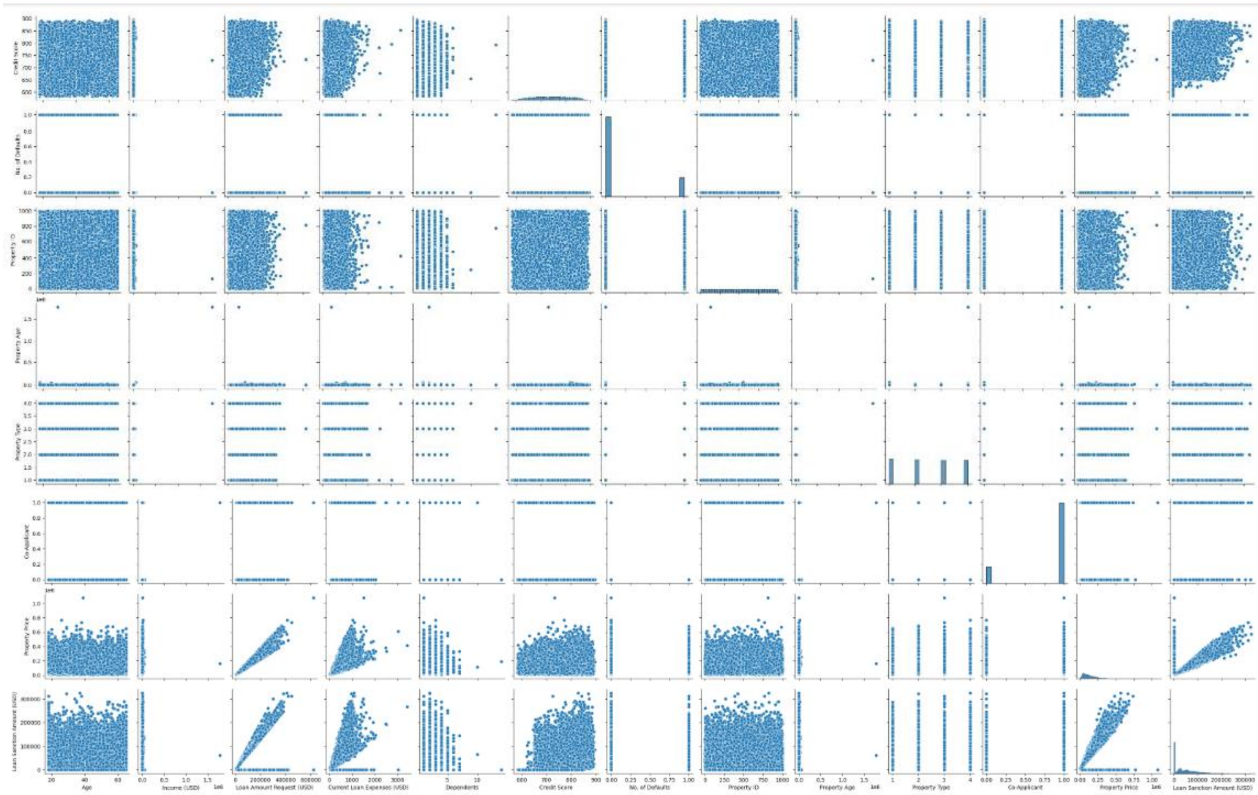
```
#Exploratory data analysis - Histogram
train_df.hist(bins=50,figsize=(15,10))
plt.show()
```




```
#Exploratory data analysis - Pairplot
sns.pairplot(train_df)
plt.show()
```



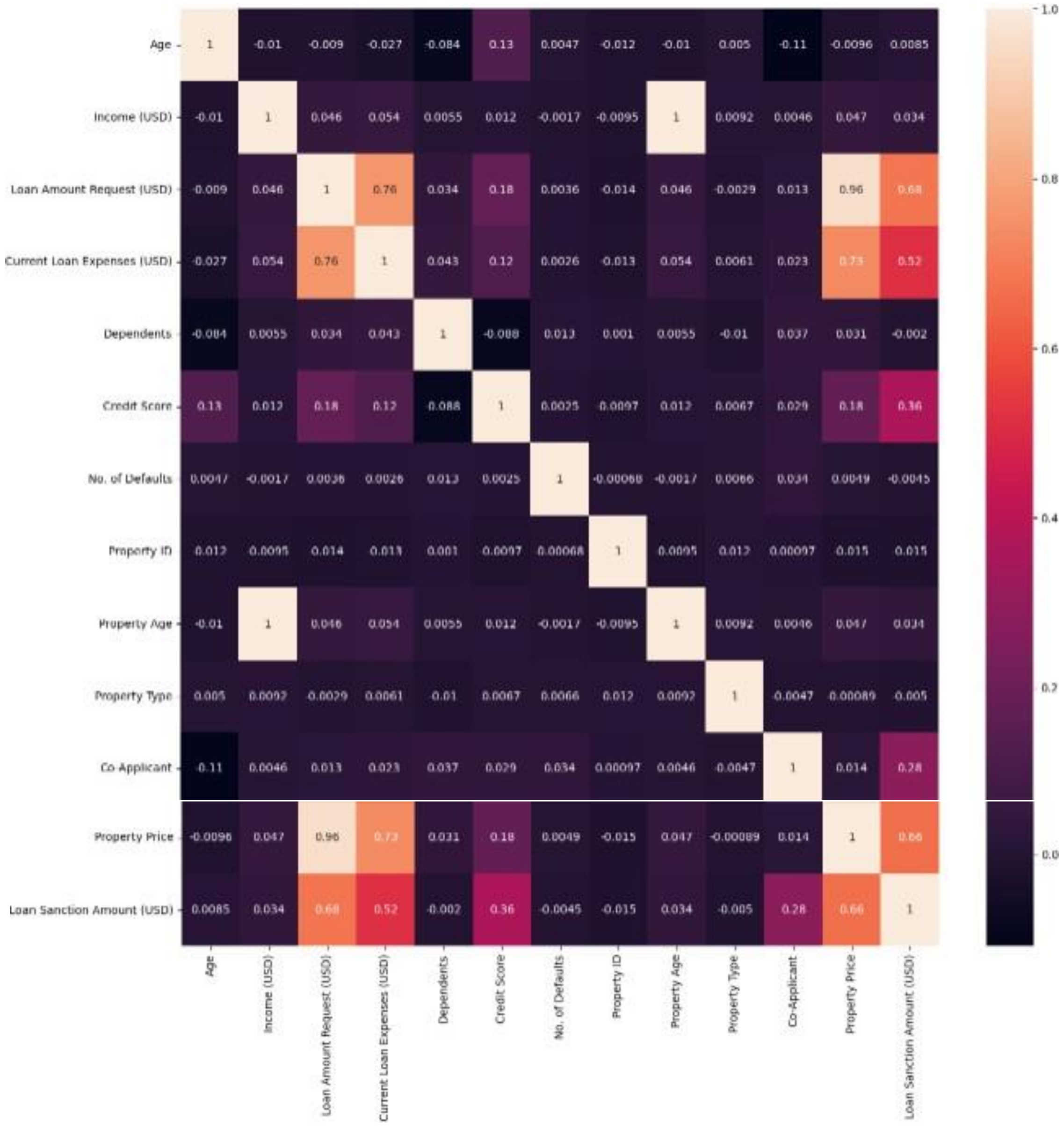
Ashwin Ravi
3122 21 5001 014
CSE-A



Feature Engineering techniques.

```
# Heatmap
plt.figure(figsize=(15,15))
sns.heatmap(train_df.corr(),annot=True)
```

Ashwin Ravi
3122 21 5001 014
CSE-A



Split the data into training and testing sets.

```
y = train_df['Loan Sanction Amount (USD)']

x = train_df[['Loan Amount Request (USD)','Current Loan Expenses (USD)','Credit Score','No. of Defaults','Co-Applicant','Property Price']]
x_train,x_test,y_train,y_test =
train_test_split(x,y,test_size=0.2,random_state=42)
```

Ashwin Ravi
3122 21 5001 014
CSE-A

```
[302] x_train.head(5)
```

	Income (USD)	Loan Amount Request (USD)	Current Loan Expenses (USD)	Credit Score	No. of Defaults	Co-Applicant	Property Price
7897	3076.13	82096.95	420.95	848.74	0	1	132251.22
23113	1998.77	33781.05	360.14	884.18	0	0	48809.75
24001	1321.36	68048.04	240.06	827.19	1	1	91988.47
168	1637.51	75095.84	417.84	815.75	1	1	114087.85
1751	1338.47	38659.63	267.70	830.23	0	1	50087.64

```
y_train.head(5)

7897     65677.56
23113    27024.84
24001        0.00
168      60076.67
1751     27061.74
Name: Loan Sanction Amount (USD), dtype: float64
```

```
[304] x_test.head(5)
```

	Income (USD)	Loan Amount Request (USD)	Current Loan Expenses (USD)	Credit Score	No. of Defaults	Co-Applicant	Property Price
20027	2791.49	136069.08	458.01	800.97	0	1	195424.20
4583	1275.47	38363.06	303.98	682.39	0	1	50345.07
3999	5724.06	172728.96	769.63	735.13	0	1	207933.22
7194	3882.05	159042.53	657.08	831.32	0	0	230691.23
15165	913.74	26430.91	264.71	868.00	0	0	47666.31

```
y_test.head(5)

20027    102051.81
4583          0.00
3999     112273.82
7194     127234.02
15165     21144.73
Name: Loan Sanction Amount (USD), dtype: float64
```

Train the model

```
model = LinearRegression()
model.fit(x_train,y_train)

▾ LinearRegression
LinearRegression()
```

Test the model.

```
y_pred = model.predict(x_test)
```

Measure the performance of the trained model.

```
#Measure the performance of the trained model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print("Mean Squared Error:", mse)
print("R-squared Score:", r2)
```

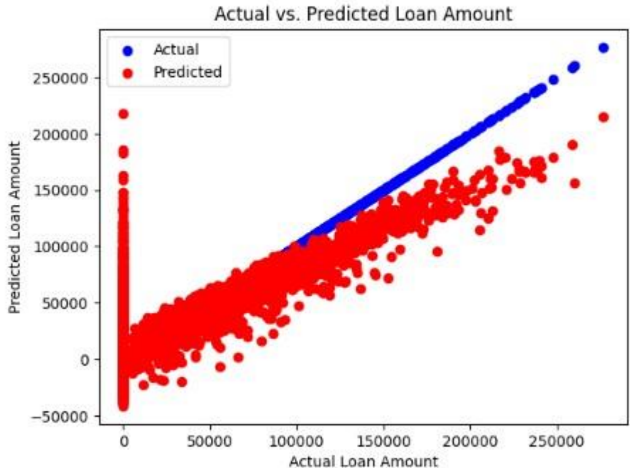
```
y_pred = model.predict(x_test)

mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print("r2: ", r2)
print("mse: ", mse)
print("score : ",model.score(x_train, y_train))

r2 :  75.33250273700457
mse :  564458841.8228692
score :  0.7996466211806167
```

Represent the results using graphs.

```
# Scatter plot for actual loan amounts
plt.scatter(y_test, y_test, color='blue', label='Actual')

# Scatter plot for predicted loan amounts
plt.scatter(y_test, y_pred, color='red', label='Predicted')

plt.xlabel("Actual Loan Amount")
plt.ylabel("Predicted Loan Amount")
plt.title("Actual vs. Predicted Loan Amount")
```

Ashwin Ravi
3122 21 5001 014
CSE-A

```
plt.legend()
plt.show()
```



Predicting loan amount for test dataset

```
#Load test dataset
test_df = pd.read_csv('/content/test.csv')

test_df.head(5)
```

	Customer ID	Name	Gender	Age	Income (USD)	Income Stability	Profession	Type of Employment	Location	Loan Amount Request (USD)	...	Dependents	Credit Score	No. of Defaults	Has Active Credit Card	Property ID	Property Age	Property Type	Property Location	Co-Applicant	Property Price
0	C-26247	Tandra Olszewski	F	47	3472.69	Low	Commercial associate	Managers	Semi-Urban	137088.98	...	2.0	799.14	0	Unpossessed	843	3472.69	2	Urban	1	236644.5
1	C-35067	Jeannette Cha	F	57	1184.84	Low	Working	Sales staff	Rural	104771.59	...	2.0	833.31	0	Unpossessed	22	1184.84	1	Rural	1	142357.3
2	C-34590	Keva Godfrey	F	52	1266.27	Low	Working	NaN	Semi-Urban	176684.91	...	3.0	627.44	0	Unpossessed	1	1266.27	1	Urban	1	300991.24
3	C-16668	Elva Sackett	M	65	1369.72	High	Pensioner	NaN	Rural	97009.18	...	2.0	833.20	0	Inactive	730	1369.72	1	Semi-Urban	0	125612.1
4	C-12196	Sade Constable	F	60	1939.23	High	Pensioner	NaN	Urban	109980.00	...	NaN	NaN	0	NaN	356	1939.23	4	Semi-Urban	1	180908.0

5 rows × 23 columns

```
test_df.loc[(test_df['Co-Applicant'] == '?'),'Co-Applicant'] = 0
test_df.loc[(test_df['Co-Applicant'] == '0'),'Co-Applicant'] = 0

test_df['Property Age'].fillna(train_df['Property Age'].mean().round(2),
inplace=True)
test_df['Credit Score'].fillna(0, inplace=True)
```

(20000, 23)

	Customer ID	Name	Gender	Age	Income (USD)	Income Stability	Profession	Type of Employment	Location	Loan Amount Request (USD)	...	Dependents	Credit Score	No. of Defaults	Has Active Credit Card	Property ID	Property Age	Property Type	Property Location	Co-Applicant	Property Price
0	C-26247	Tandra Olszewski	F	47	3472.69	Low	Commercial associate	Managers	Semi-Urban	137088.98	...	2.0	799.14	0	Unpossessed	843	3472.69	2	Urban	1	236644.5
1	C-35067	Jeannette Cha	F	57	1184.84	Low	Working	Sales staff	Rural	104771.59	...	2.0	833.31	0	Unpossessed	22	1184.84	1	Rural	1	142357.3
2	C-34590	Keva Godfrey	F	52	1266.27	Low	Working	NaN	Semi-Urban	176684.91	...	3.0	627.44	0	Unpossessed	1	1266.27	1	Urban	1	300991.24
3	C-16668	Elva Sackett	M	65	1369.72	High	Pensioner	NaN	Rural	97009.18	...	2.0	833.20	0	Inactive	730	1369.72	1	Semi-Urban	0	125612.1
4	C-12196	Sade Constable	F	60	1939.23	High	Pensioner	NaN	Urban	109980.00	...	NaN	0.00	0	NaN	356	1939.23	4	Semi-Urban	1	180908.0

5 rows × 23 columns

Ashwin Ravi
3122 21 5001 014
CSE-A

```
test_df1=test_df[['Loan Amount Request (USD)','Current Loan Expenses
(USD)','Credit Score','No. of Defaults','Co-Applicant','Property Price']]
test_pred = model.predict(test_df1)
op = pd.DataFrame({'Customer ID' : CUST_ID, 'Loan Sanction Amount (USD)':test_pred
})
op.to_csv('output.csv',index=False)
op.head(10)
```

	Customer ID	Loan Sanction Amount (USD)
0	C-26247	87276.701206
1	C-35067	66414.686587
2	C-34590	5243.481243
3	C-16668	44926.956728
4	C-12196	67851.018732
5	C-2600	-2826.468323
6	C-9047	91288.476652
7	C-2206	78632.937794
8	C-25607	513.293304
9	C-11606	14897.476773