HOME LONE DATA ANALYSIS

Abstract:

In the domain of financial lending, ensuring a safe and secure lending experience requires accurate prediction models to assess the likelihood of loan defaults. This project focuses on the development of a deep learning model aimed at predicting the probability of default for future loans based on historical loan data. The dataset used in this project is highly imbalanced, with a disproportionate number of default versus non-default instances, adding complexity to the predictive task. Additionally, the dataset contains a large number of features, increasing the challenge of effectively capturing relevant patterns while mitigating overfitting. The proposed solution employs advanced deep learning techniques, such as neural networks, to handle class imbalance and leverage the rich feature set for accurate predictions. Evaluation metrics, such as precision, recall, and the area under the ROC curve, are utilized to assess the model's performance in real-world scenarios, ensuring that it provides robust and reliable predictions for future loan default risks.

Objective:

Creating a model that predicts whether or not an applicant will be able to repay a loan using historical data

Domain:

Finance

Analysis to be done:

Perform data preprocessing and build a deep learning prediction model

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter   End Lab   Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code   Python 3 [3.10]

[1]:
```python
import pandas as pd
import sklearn
import numpy as np
import matplotlib.pyplot as plt
import os
import warnings
import seaborn as sns
from sklearn.preprocessing import OneHotEncoder
from sklearn.datasets import make_blobs
from sklearn.impute import SimpleImputer
from sklearn.pipeline import Pipeline
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import StandardScaler
from sklearn.svm import LinearSVC
from sklearn.metrics import roc_auc_score
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import roc_auc_score
from sklearn.calibration import CalibratedClassifierCV
from sklearn.metrics import confusion_matrix
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.linear_model import SGDClassifier
import plotly.offline as py
import plotly.graph_objs as go
from plotly.offline import init_notebook_mode, iplot
from sklearn.model_selection import train_test_split
init_notebook_mode(connected=True)
```

Simple 0 0 Python 3 [3.10] | Idle   Mode: Command   Ln 3, Col 18   homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter   End Lab   Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code   Python 3 [3.10]

```python
import pickle
import gc
warnings.filterwarnings('ignore')
%matplotlib inline
```

[2]:
```python
# Load the dataset
house_loan = pd.read_csv('loan_data (1).csv')
house_loan.describe()
```

[2]:

	SK_ID_CURR	TARGET	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	REGION_POPULATION_RELATIVE	
count	307511.000000	307511.000000	307511.000000	3.075110e+05	3.075110e+05	307499.000000	3.072330e+05	307511.000000	3
mean	278180.518577	0.080729	0.417052	1.687979e+05	5.990260e+05	27108.573909	5.383962e+05	0.020868	-
std	102790.175348	0.272419	0.722121	2.371231e+05	4.024908e+05	14493.737315	3.694465e+05	0.013831	
min	100002.000000	0.000000	0.000000	2.565000e+04	4.500000e+04	1615.500000	4.050000e+04	0.000290	-
25%	189145.500000	0.000000	0.000000	1.125000e+05	2.700000e+05	16524.000000	2.385000e+05	0.010006	-
50%	278202.000000	0.000000	0.000000	1.471500e+05	5.135310e+05	24903.000000	4.500000e+05	0.018850	-
75%	367142.500000	0.000000	1.000000	2.025000e+05	8.086500e+05	34596.000000	6.795000e+05	0.028663	-
max	456255.000000	1.000000	19.000000	1.170000e+08	4.050000e+06	258025.500000	4.050000e+06	0.072508	

8 rows × 106 columns

Simple 0 0 Python 3 [3.10] | Idle   Mode: Command   Ln 1, Col 1   homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
[3]: house_loan.columns
```

```
[3]: Index(['SK_ID_CURR', 'TARGET', 'NAME_CONTRACT_TYPE', 'CODE_GENDER',
       'FLAG_OWN_CAR', 'FLAG_OWN_REALTY', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL',
       'AMT_CREDIT', 'AMT_ANNUITY',
       ...
       'FLAG_DOCUMENT_18', 'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_HOUR',
       'AMT_REQ_CREDIT_BUREAU_DAY', 'AMT_REQ_CREDIT_BUREAU_WEEK',
       'AMT_REQ_CREDIT_BUREAU_MON', 'AMT_REQ_CREDIT_BUREAU_QRT',
       'AMT_REQ_CREDIT_BUREAU_YEAR'],
      dtype='object', length=122)
```

```
[4]: house_loan.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 286.2+ MB
```

```
[5]: house_loan.isnull().sum()
```

```
[5]: SK_ID_CURR                 0
TARGET                     0
NAME_CONTRACT_TYPE         0
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
TARGET                         0
NAME_CONTRACT_TYPE             0
CODE_GENDER                    0
FLAG_OWN_CAR                   0
                             ...
AMT_REQ_CREDIT_BUREAU_DAY    41519
AMT_REQ_CREDIT_BUREAU_WEEK   41519
AMT_REQ_CREDIT_BUREAU_MON    41519
AMT_REQ_CREDIT_BUREAU_QRT    41519
AMT_REQ_CREDIT_BUREAU_YEAR   41519
Length: 122, dtype: int64
```

```
[6]: house_loan.head()
```

[6]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT
0	100002	1	Cash loans	M	N	Y	0	202500.0	406597.5	
1	100003	0	Cash loans	F	N	N	0	270000.0	1293502.5	
2	100004	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	
3	100006	0	Cash loans	F	N	Y	0	135000.0	312682.5	
4	100007	0	Cash loans	M	N	Y	0	121500.0	513000.0	

5 rows × 122 columns

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
TARGET                        0
NAME_CONTRACT_TYPE            0
CODE_GENDER                   0
FLAG_OWN_CAR                  0
                            ...
AMT_REQ_CREDIT_BUREAU_DAY     41519
AMT_REQ_CREDIT_BUREAU_WEEK    41519
AMT_REQ_CREDIT_BUREAU_MON     41519
AMT_REQ_CREDIT_BUREAU_QRT     41519
AMT_REQ_CREDIT_BUREAU_YEAR    41519
Length: 122, dtype: int64
```

[6]:
```
house_loan.head()
```

[6]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT
0	100002	1	Cash loans	M	N	Y	0	202500.0	406597.5	
1	100003	0	Cash loans	F	N	N	0	270000.0	1293502.5	
2	100004	0	Revolving loans	M	Y	Y	0	67500.0	135000.0	
3	100006	0	Cash loans	F	N	Y	0	135000.0	312682.5	
4	100007	0	Cash loans	M	N	Y	0	121500.0	513000.0	

5 rows × 122 columns

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

5 rows × 122 columns

[7]:
```
defaulters=(house_loan.TARGET==1).sum()
payers=(house_loan.TARGET==0).sum()
print((defaulters/payers)*100)
```

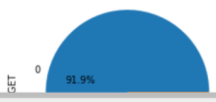
```
8.781828601345662
```

[8]:
```
without_id=[column for column in house_loan.columns if column!='SK_ID_CURR']

#check for duplicate values

na=house_loan[house_loan.duplicated(subset=without_id,keep=False)]
print("Duplicates are: ",na.shape[0])
```

```
Duplicates are:  0
```

[9]:
```
house_loan.TARGET.value_counts().plot(kind='pie',autopct='%1.1f%%')
```

[9]: <AxesSubplot: ylabel='TARGET'>



Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

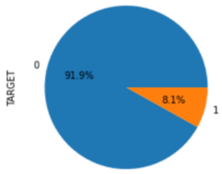
jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

[9]: <AxesSubplot: ylabel='TARGET'>




```
[10]: import matplotlib as plt
```

```
[11]: shuffled_data=house_loan.sample(frac=1,random_state=3)
unpaid_home_loan=shuffled_data.loc[shuffled_data['TARGET']==1]
paid_home_loan=shuffled_data.loc[shuffled_data['TARGET']==0].sample(n=24825,random_state=69)
normalised_home_loan=pd.concat([unpaid_home_loan,paid_home_loan])
normalised_home_loan.TARGET.value_counts().plot(kind='pie',autopct="%1.1f%%")
```

[11]: <AxesSubplot: ylabel='TARGET'>

1

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]


```
[53]: import tensorflow as tf
```

```
[13]: normalised_home_loan.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 49650 entries, 207339 to 121862
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 46.6+ MB
```

```
[14]: normalised_home_loan.head
```

```
[14]: <bound method NDFrame.head of         SK_ID_CURR  TARGET NAME_CONTRACT_TYPE CODE_GENDER FLAG_OWN_CAR  \
207339      340318       1         Cash loans           F            N
8756        110186       1         Cash loans           M            Y
230344      366811       1         Cash loans           F            N
178329      306645       1         Cash loans           M            Y
55586       164407       1         Cash loans           M            N
...            ...     ...                ...         ...          ...
130947      251878       0         Cash loans           F            Y
40467       146875       0         Cash loans           F            N
187004      316791       0         Cash loans           M            N
131755      252811       0         Cash loans           F            N
121862      241287       0         Cash loans           M            N

       FLAG_OWN_REALTY  CNT_CHILDREN  AMT_INCOME_TOTAL  AMT_CREDIT  \
207339               N             0          112500.0    405000.0
```


Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
8756               0                  0.0                        0.0
230344             0                  NaN                        NaN
178329             0                  NaN                        NaN
55586              0                  0.0                        0.0
...              ...                  ...                        ...
130947             0                  0.0                        0.0
40467              0                  0.0                        0.0
187004             0                  0.0                        0.0
131755             0                  0.0                        0.0
121862             0                  0.0                        0.0

        AMT_REQ_CREDIT_BUREAU_WEEK  AMT_REQ_CREDIT_BUREAU_MON  \
207339                         0.0                        0.0
8756                           0.0                        0.0
230344                         NaN                        NaN
178329                         NaN                        NaN
55586                          0.0                        0.0
...                            ...                        ...
130947                         0.0                        1.0
40467                          0.0                        0.0
187004                         0.0                        0.0
131755                         0.0                        0.0
121862                         0.0                        0.0

        AMT_REQ_CREDIT_BUREAU_QRT  AMT_REQ_CREDIT_BUREAU_YEAR
207339                        0.0                         3.0
8756                          0.0                         0.0
230344                        NaN                         NaN
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

[15]:
```
normalised_home_loan.dropna(axis=0)
normalised_home_loan.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 49650 entries, 207339 to 121862
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 46.6+ MB
```

[16]:
```
normalised_home_loan.isnull().sum()
```

[16]:
```
SK_ID_CURR                       0
TARGET                           0
NAME_CONTRACT_TYPE               0
CODE_GENDER                      0
FLAG_OWN_CAR                     0
                              ...
AMT_REQ_CREDIT_BUREAU_DAY     7648
AMT_REQ_CREDIT_BUREAU_WEEK    7648
AMT_REQ_CREDIT_BUREAU_MON     7648
AMT_REQ_CREDIT_BUREAU_QRT     7648
AMT_REQ_CREDIT_BUREAU_YEAR    7648
Length: 122, dtype: int64
```

[17]:
```
#print(normalised_home_loan.apply())
```

[18]:
```
print(pd.unique(normalised_home_loan.AMT_REQ_CREDIT_BUREAU_DAY))
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
[17]: #print(normalised_home_loan.apply())
```

```
[18]: print(pd.unique(normalised_home_loan.AMT_REQ_CREDIT_BUREAU_DAY))
print(pd.unique(normalised_home_loan.AMT_REQ_CREDIT_BUREAU_WEEK))
print(pd.unique(normalised_home_loan.AMT_REQ_CREDIT_BUREAU_MON))
print(pd.unique(normalised_home_loan.AMT_REQ_CREDIT_BUREAU_QRT))
print(pd.unique(normalised_home_loan.AMT_REQ_CREDIT_BUREAU_YEAR))
```

```
[ 0. nan  1.  2.  4.  3.  9.]
[ 0. nan  1.  2.  4.  3.  5.  6.]
[ 0. nan  1.  3.  5.  9.  2.  6.  8.  4. 11. 12.  7. 13. 10. 17. 15. 14.
 16. 18. 27.]
[ 0. nan  2.  3.  1.  4.  5.  6. 19.  7.]
[ 3.  0. nan  1.  5.  4.  2.  6.  7.  8.  9. 10. 14. 13. 12. 11. 22. 16.
 23. 17.]
```

```
[19]: normalised_home_loan.dropna(axis=0)
```

[19]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT
279124	423360	1	Cash loans	M	Y	N	1	157500.0	1125000.0
216116	350411	1	Cash loans	M	Y	N	0	112500.0	225000.0
133687	255050	1	Cash loans	M	Y	N	1	337500.0	704844.0
4159	104863	1	Cash loans	M	Y	N	0	265500.0	521280.0

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
[20]: print(normalised_home_loan.info())
print(normalised_home_loan.isnull().sum())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 49650 entries, 207339 to 121862
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 46.6+ MB
None
SK_ID_CURR                    0
TARGET                        0
NAME_CONTRACT_TYPE            0
CODE_GENDER                   0
FLAG_OWN_CAR                  0
                           ...
AMT_REQ_CREDIT_BUREAU_DAY     7648
AMT_REQ_CREDIT_BUREAU_WEEK    7648
AMT_REQ_CREDIT_BUREAU_MON     7648
AMT_REQ_CREDIT_BUREAU_QRT     7648
AMT_REQ_CREDIT_BUREAU_YEAR    7648
Length: 122, dtype: int64
```

```
[21]: normalised_home_loan.TARGET.value_counts().plot(kind='pie',autopct="%1.1f%%")
```

```
[21]: <AxesSubplot: ylabel='TARGET'>
```

1

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter — End Lab — Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb — Code — Python 3 [3.10]

[20]:
```
print(normalised_home_loan.info())
print(normalised_home_loan.isnull().sum())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 49650 entries, 207339 to 121862
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 46.6+ MB
None
SK_ID_CURR                     0
TARGET                         0
NAME_CONTRACT_TYPE             0
CODE_GENDER                    0
FLAG_OWN_CAR                   0
                            ...
AMT_REQ_CREDIT_BUREAU_DAY     7648
AMT_REQ_CREDIT_BUREAU_WEEK    7648
AMT_REQ_CREDIT_BUREAU_MON     7648
AMT_REQ_CREDIT_BUREAU_QRT     7648
AMT_REQ_CREDIT_BUREAU_YEAR    7648
Length: 122, dtype: int64
```

[21]:
```
normalised_home_loan.TARGET.value_counts().plot(kind='pie',autopct="%1.1f%%")
```

[21]: `<AxesSubplot: ylabel='TARGET'>`

Simple 0 0 Python 3 [3.10] | Idle — Mode: Command — Ln 1, Col 1 — homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

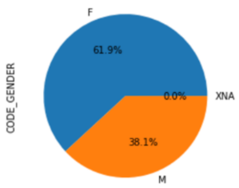
jupyterlab jupyter — End Lab — Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb — Code — Python 3 [3.10]

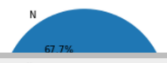
[23]:
```
normalised_home_loan.CODE_GENDER.value_counts().plot(kind='pie',autopct="%1.1f%%")
#roughly equal amount
```

[23]: `<AxesSubplot: ylabel='CODE_GENDER'>`



[24]:
```
normalised_home_loan.FLAG_OWN_CAR.value_counts().plot(kind='pie',autopct="%1.1f%%")
```

[24]: `<AxesSubplot: ylabel='FLAG_OWN_CAR'>`



Simple 0 0 Python 3 [3.10] | Idle — Mode: Command — Ln 1, Col 1 — homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

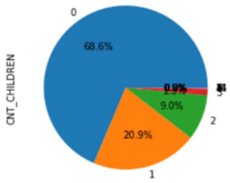
jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]


```
[25]: normalised_home_loan.CNT_CHILDREN.value_counts().plot(kind='pie',autopct="%1.1f%%")
[25]: <AxesSubplot: ylabel='CNT_CHILDREN'>
```




```
[26]: (normalised_home_loan[normalised_home_loan['AMT_INCOME_TOTAL']>1000000]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['AMT
[26]: 0    64.864865
      1    35.135135
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
Name: TARGET, dtype: float64
[27]: #print((normalised_home_loan[normalised_home_loan['CNT_CHILDREN']>1]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CNT_CH
print((normalised_home_loan[normalised_home_loan['CNT_CHILDREN']>2]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CNT_CHI
print((normalised_home_loan[normalised_home_loan['CNT_CHILDREN']>5]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CNT_CHI
#as number of children is increasing lone defaulters are increasing

1    57.047872
0    42.952128
Name: TARGET, dtype: float64
1    81.818182
0    18.181818
Name: TARGET, dtype: float64

[28]: print((normalised_home_loan[normalised_home_loan['FLAG_OWN_CAR']=='N']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['FLAG
print((normalised_home_loan[normalised_home_loan['FLAG_OWN_CAR']=='Y']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['FLAG

1    51.350064
0    48.649936
Name: TARGET, dtype: float64
0    52.823962
1    47.176038
Name: TARGET, dtype: float64

[29]: print((normalised_home_loan[normalised_home_loan['CODE_GENDER']=='M']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CODE_
print((normalised_home_loan[normalised_home_loan['CODE_GENDER']=='F']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CODE_
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
Name: TARGET, dtype: float64
```

[27]:
```
#print((normalised_home_loan[normalised_home_loan['CNT_CHILDREN']>1]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CNT_CH
print((normalised_home_loan[normalised_home_loan['CNT_CHILDREN']>2]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CNT_CHI
print((normalised_home_loan[normalised_home_loan['CNT_CHILDREN']>5]['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CNT_CHI
#as number of children is increasing lone defaulters are increasing
```

```
1    57.047872
0    42.952128
Name: TARGET, dtype: float64
1    81.818182
0    18.181818
Name: TARGET, dtype: float64
```

[28]:
```
print((normalised_home_loan[normalised_home_loan['FLAG_OWN_CAR']=='N']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['FLAG
print((normalised_home_loan[normalised_home_loan['FLAG_OWN_CAR']=='Y']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['FLAG
```

```
1    51.350064
0    48.649936
Name: TARGET, dtype: float64
0    52.823962
1    47.176038
Name: TARGET, dtype: float64
```

[29]:
```
print((normalised_home_loan[normalised_home_loan['CODE_GENDER']=='M']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CODE_
print((normalised_home_loan[normalised_home_loan['CODE_GENDER']=='F']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CODE_
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

[28]:
```
print((normalised_home_loan[normalised_home_loan['FLAG_OWN_CAR']=='N']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['FLAG
print((normalised_home_loan[normalised_home_loan['FLAG_OWN_CAR']=='Y']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['FLAG
```

```
1    51.350064
0    48.649936
Name: TARGET, dtype: float64
0    52.823962
1    47.176038
Name: TARGET, dtype: float64
```

[29]:
```
print((normalised_home_loan[normalised_home_loan['CODE_GENDER']=='M']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CODE_
print((normalised_home_loan[normalised_home_loan['CODE_GENDER']=='F']['TARGET'].value_counts())/len(normalised_home_loan[normalised_home_loan['CODE_
```

```
1    56.280372
0    43.719628
Name: TARGET, dtype: float64
0    53.867691
1    46.132309
Name: TARGET, dtype: float64
```

[30]:
```
#men more likely to default in payment of loans
```

[31]:
```
print((normalised_home_loan[normalised_home_loan['NAME_CONTRACT_TYPE']=='Cash loans']['TARGET'].value_counts())/len(normalised_home_loan[normalised_
print((normalised_home_loan[normalised_home_loan['NAME_CONTRACT_TYPE']=='Revolving loans']['TARGET'].value_counts())/len(normalised_home_loan[normal:
#cash loans have a higher percent of defaulters
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
1    46.132309
Name: TARGET, dtype: float64
```

[30]:
```
#men more likely to default in payment of loans
```

[31]:
```
print((normalised_home_loan[normalised_home_loan['NAME_CONTRACT_TYPE']=='Cash_loans']['TARGET'].value_counts())/len(normalised_home_loan[normalised_l
print((normalised_home_loan[normalised_home_loan['NAME_CONTRACT_TYPE']=='Revolving_loans']['TARGET'].value_counts())/len(normalised_home_loan[normal:
#cash loans have a higher percent of defaulters
```

```
Series([], Name: TARGET, dtype: float64)
Series([], Name: TARGET, dtype: float64)
```

[32]:
```
normalised_home_loan=normalised_home_loan.sample(frac=1,random_state=5)
```

[33]:
```
from sklearn.preprocessing import OrdinalEncoder
ordenc=OrdinalEncoder()
normalised_home_loan['NAME_CONTRACT_TYPE_CODE']=ordenc.fit_transform(normalised_home_loan[['NAME_CONTRACT_TYPE']])
print(normalised_home_loan[['NAME_CONTRACT_TYPE','NAME_CONTRACT_TYPE_CODE']].head(20))
print(normalised_home_loan['NAME_CONTRACT_TYPE_CODE'].value_counts())
```

```
       NAME_CONTRACT_TYPE  NAME_CONTRACT_TYPE_CODE
302218         Cash loans                      0.0
167526         Cash loans                      0.0
159305         Cash loans                      0.0
275427         Cash loans                      0.0
8837           Cash loans                      0.0
192094         Cash loans                      0.0
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
Name: NAME_CONTRACT_TYPE_CODE, dtype: int64
```

[34]:
```
normalised_home_loan['CODE_GENDER_CODE']=ordenc.fit_transform(normalised_home_loan[['CODE_GENDER']])
print(normalised_home_loan[['CODE_GENDER','CODE_GENDER_CODE']].head(20))
print(normalised_home_loan['CODE_GENDER_CODE'].value_counts())
```

```
       CODE_GENDER  CODE_GENDER_CODE
302218           M               1.0
167526           F               0.0
159305           M               1.0
275427           F               0.0
8837             M               1.0
192094           M               1.0
235115           F               0.0
79051            F               0.0
123267           M               1.0
5517             F               0.0
128624           M               1.0
187583           F               0.0
143193           M               1.0
288269           F               0.0
44320            F               0.0
256898           F               0.0
118237           F               0.0
5980             M               1.0
96475            F               0.0
249976           F               0.0
0.0    30716
1.0    18932
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

```
Name: NAME_CONTRACT_TYPE_CODE, dtype: int64
```

[34]:

```
normalised_home_loan['CODE_GENDER_CODE']=ordenc.fit_transform(normalised_home_loan[['CODE_GENDER']])
print(normalised_home_loan[['CODE_GENDER','CODE_GENDER_CODE']].head(20))
print(normalised_home_loan['CODE_GENDER_CODE'].value_counts())
```

```
       CODE_GENDER  CODE_GENDER_CODE
302218           M               1.0
167526           F               0.0
159305           M               1.0
275427           F               0.0
8837             M               1.0
192094           M               1.0
235115           F               0.0
79051            F               0.0
123267           M               1.0
5517             F               0.0
128624           M               1.0
187583           F               0.0
143193           M               1.0
288269           F               0.0
44320            F               0.0
256898           F               0.0
118237           F               0.0
5980             M               1.0
96475            F               0.0
249976           F               0.0
0.0    30716
1.0    18932
```


https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow

jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]

```
normalised_home_loan.loc[normalised_home_loan['CODE_GENDER_CODE']==2]
```

[35]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT
83382	196708	0	Revolving loans	XNA	N	Y	1	135000.0	405000.0
189640	319880	0	Revolving loans	XNA	Y	Y	0	247500.0	540000.0

2 rows × 124 columns

[36]:

```
normalised_home_loan['FLAG_OWN_CAR_CODE']=ordenc.fit_transform(normalised_home_loan[['FLAG_OWN_CAR']])
print(normalised_home_loan[['FLAG_OWN_CAR','FLAG_OWN_CAR_CODE']].head(20))
print(normalised_home_loan['FLAG_OWN_CAR_CODE'].value_counts())
```

```
       FLAG_OWN_CAR  FLAG_OWN_CAR_CODE
302218            N                0.0
167526            N                0.0
159305            N                0.0
275427            N                0.0
8837              N                0.0
192094            N                0.0
235115            N                0.0
79051             N                0.0
123267            N                0.0
5517              N                0.0
128624            N                0.0
187583            N                0.0
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code | Python 3 [3.10]

```
[38]: normalised_home_loan=normalised_home_loan.sample(frac=1,random_state=45)

[39]: normalised_home_loan['TARGET'].value_counts()

[39]: 0    24825
      1    24825
      Name: TARGET, dtype: int64

[40]: y=normalised_home_loan.TARGET

[41]: #y=y.sample(frac=1,random_state=45)

[42]: normalised_home_loan_features=['SK_ID_CURR','NAME_CONTRACT_TYPE_CODE','CNT_CHILDREN_CODE','FLAG_OWN_CAR_CODE']

[43]: from sklearn.model_selection import train_test_split

[44]: X=normalised_home_loan[normalised_home_loan_features]

[45]: #X=X.sample(frac=1,random_state=45)

[46]: blobs_random_seed = 42
      centers = [(0,0), (5,5)]
      cluster_std = 1
      frac_test_split = 0.33
      num_features_for_samples = 2
      num_samples_total = 49650
      # Generate data
```

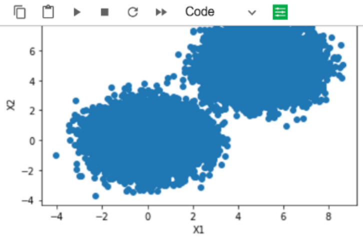
Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow
jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code | Python 3 [3.10]



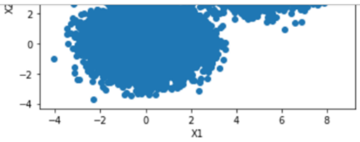

```
[56]: import matplotlib.pyplot as plt
      import numpy
      from sklearn import metrics
      from sklearn import svm

[57]: clf=svm.SVC(kernel='linear')

[58]: clf=clf.fit(X_train,y_train)

[59]: predictions = clf.predict(X_test)

      # Generate confusion matrix
      matrix = plot_confusion_matrix(clf, X_test, y_test,
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

File Edit View Run Kernel Tabs Settings Help

homeloan.ipynb

Code Python 3 [3.10]



[56]:
```python
import matplotlib.pyplot as plt
import numpy
from sklearn import metrics
from sklearn import svm
```

[57]:
```python
clf=svm.SVC(kernel='linear')
```

[58]:
```python
clf=clf.fit(X_train,y_train)
```

[59]:
```python
predictions = clf.predict(X_test)

# Generate confusion matrix
matrix = plot_confusion_matrix(clf, X_test, y_test,
                                 cmap=plt.cm.Blues,
                                 normalize='true')
plt.pyplot.title('Confusion matrix for our classifier')
plt.pyplot.show(matrix)
plt.pyplot.show()
```

Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 1, Col 1 homeloan.ipynb

https://lms.simplilearn.com
Practice Labs | Deep Learning with Keras and Tensorflow
jupyterlab jupyter End Lab Reset

File Edit View Run Kernel Tabs Settings Help

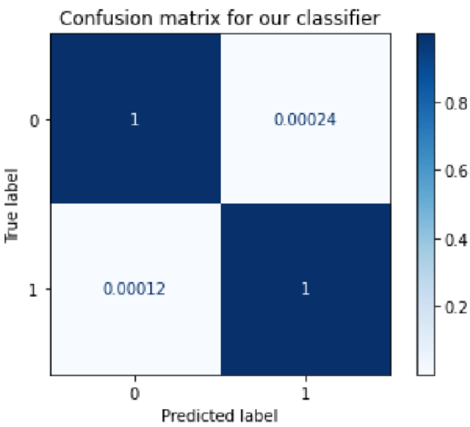
homeloan.ipynb

Code Python 3 [3.10]

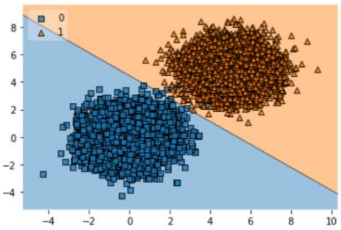
[61]:
```python
print(precision_score(y_test, predictions))
print(recall_score(y_test, predictions))
print(f1_score(y_test,predictions,average=None))
```

```
0.9992706053975201
1.0
[0.99963244 0.99963517]
```

[62]:
```python
support_vectors = clf.support_vectors_

# Visualize support vectors
plt.pyplot.scatter(X_train[:,0], X_train[:,1])
plt.pyplot.scatter(support_vectors[:,0], support_vectors[:,1], color='red')
plt.pyplot.title('Linearly separable data with support vectors')
plt.pyplot.xlabel('X1')
plt.pyplot.ylabel('X2')
plt.pyplot.show()
```



[63]:
```python
from mlxtend.plotting import plot_decision_regions
plot_decision_regions(X_test, y_test, clf=clf, legend=2)
plt.pyplot.show()
```



Simple 0 0 Python 3 [3.10] | Idle Mode: Command Ln 3, Col 18 homeloan.ipynb