Лабортаторная работа №2 по курсо ТМО

Бекетов Роман

ИУ5-62Б

Обработка пропусков в данных, кодирование категориальных признаков, масштабирование данных.

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns

import os

df = pd.read_csv("loan_data.csv")

df.sample(5)
```

```
      Loan_ID  Gender Married Dependents     Education
Self_Employed  \
36   LP001151  Female      No          0      Graduate            No

103  LP001581    Male     Yes          0  Not Graduate           NaN

374  LP002940    Male      No          0  Not Graduate            No

268  LP002361    Male     Yes          0      Graduate            No

153  LP001814    Male     Yes          2      Graduate            No


     ApplicantIncome  CoapplicantIncome  LoanAmount  Loan_Amount_Term
\
36              4000             2275.0       144.0             360.0

103             1820             1769.0        95.0             360.0

374             3833                0.0       110.0             360.0

268             1820             1719.0       100.0             360.0

153             9703                0.0       112.0             360.0


     Credit_History Property_Area Loan_Status
36              1.0     Semiurban           Y
103             1.0         Rural           Y
```

```
374             1.0         Rural           Y
268             1.0         Urban           Y
153             1.0         Urban           Y
```

```
df = df.drop(['Loan_ID'], axis=1)
```

```
df.sample(3)
```

```
    Gender Married Dependents     Education Self_Employed
ApplicantIncome  \
273   Male     Yes          0      Graduate            No
2920
58    Male     Yes          1      Graduate            No
3988
354   Male     Yes          0  Not Graduate            No
4467

     CoapplicantIncome  LoanAmount  Loan_Amount_Term
Credit_History  \
273          16.120001        87.0             360.0            1.0

58            0.000000        50.0             240.0            1.0

354           0.000000       120.0             360.0            NaN


    Property_Area Loan_Status
273         Rural           Y
58          Urban           Y
354         Rural           Y
```

```
df.shape
```

```
(381, 12)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 381 entries, 0 to 380
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Gender             376 non-null    object
 1   Married            381 non-null    object
 2   Dependents         373 non-null    object
 3   Education          381 non-null    object
 4   Self_Employed      360 non-null    object
 5   ApplicantIncome    381 non-null    int64
 6   CoapplicantIncome  381 non-null    float64
 7   LoanAmount         381 non-null    float64
 8   Loan_Amount_Term   370 non-null    float64
 9   Credit_History     351 non-null    float64
 10  Property_Area      381 non-null    object 
 11  Loan_Status        381 non-null    object 
dtypes: float64(4), int64(1), object(7)
memory usage: 35.8+ KB
```

```
df.isnull().sum()
```

```
Gender                5
Married               0
Dependents            8
Education             0
Self_Employed        21
ApplicantIncome       0
CoapplicantIncome     0
LoanAmount            0
Loan_Amount_Term     11
Credit_History       30
Property_Area         0
Loan_Status           0
dtype: int64
```

```
df_encoding = pd.get_dummies(df, columns=['Property_Area'],
prefix='Property_Area')
```

```
df_encoding.head(3)
```

```
  Gender Married Dependents     Education Self_Employed 
ApplicantIncome  \
0   Male     Yes          1      Graduate            No            
4583   
1   Male     Yes          0      Graduate           Yes            
3000   
2   Male     Yes          0  Not Graduate            No            
2583   

   CoapplicantIncome  LoanAmount  Loan_Amount_Term  Credit_History  \
0             1508.0       128.0             360.0             1.0   
1                0.0        66.0             360.0             1.0   
2             2358.0       120.0             360.0             1.0   

  Loan_Status  Property_Area_Rural  Property_Area_Semiurban  \
0           N                 True                    False   
1           Y                False                    False   
2           Y                False                    False   

   Property_Area_Urban  
0                False  
1                 True  
2                 True  
```

```python
bit_columns = [
    'Gender',
    'Married',
    'Education',
    'Self_Employed',
    'Loan_Status',
    'Property_Area_Rural',
    'Property_Area_Semiurban',
    'Property_Area_Urban'
]

for col in bit_columns:
    print(f"{col}:\n{df_encoding[col].unique()}\n")
```

```
Gender:
['Male' 'Female' nan]

Married:
['Yes' 'No']

Education:
['Graduate' 'Not Graduate']

Self_Employed:
['No' 'Yes' nan]

Loan_Status:
['N' 'Y']

Property_Area_Rural:
[ True False]

Property_Area_Semiurban:
[False  True]

Property_Area_Urban:
[False  True]
```

```python
df_encoding['Property_Area_Urban'] =
df_encoding['Property_Area_Urban'].astype(int)
df_encoding['Property_Area_Semiurban'] =
df_encoding['Property_Area_Semiurban'].astype(int)
df_encoding['Property_Area_Rural'] =
df_encoding['Property_Area_Rural'].astype(int)

df_encoding['Education'].replace(['Graduate', 'Not Graduate'], [1, 0],
inplace=True)
df_encoding['Married'].replace(['Yes', 'No'], [1, 0], inplace=True)
df_encoding['Loan_Status'].replace(['Y', 'N'], [1, 0], inplace=True)
```

```python
df_encoding.sample(4)
```

```
    Gender  Married Dependents  Education Self_Employed  
ApplicantIncome  \
72    Male        0          0          0           NaN  
7333  
41    Male        1          2          1            No  
2708  
199   Male        1          2          1           Yes  
5746  
244   Male        1          0          1            No  
2333  

     CoapplicantIncome  LoanAmount  Loan_Amount_Term  
Credit_History  \
72                 0.0       120.0             360.0             1.0

41              1167.0        97.0             360.0             1.0

199                0.0       144.0              84.0             NaN

244             2417.0       136.0             360.0             1.0


     Loan_Status  Property_Area_Rural  Property_Area_Semiurban  \
72             0                    1                        0   
41             1                    0                        1   
199            1                    1                        0   
244            1                    0                        0   

     Property_Area_Urban  
72                     0  
41                     0  
199                    0  
244                    1  
```

```python
df_encoding['Gender'].fillna('NotGiven', inplace=True)
df_encoding['Self_Employed'].fillna('NotGiven', inplace=True)
```

```python
for col in bit_columns:
    print(f"{col}:\n{df_encoding[col].unique()}\n")
```

```
Gender:
['Male' 'Female' 'NotGiven']

Married:
[1 0]

Education:
[1 0]
```

```
Self_Employed:
['No' 'Yes' 'NotGiven']

Loan_Status:
[0 1]

Property_Area_Rural:
[1 0]

Property_Area_Semiurban:
[0 1]

Property_Area_Urban:
[0 1]
```

```
df_encoding = pd.get_dummies(df_encoding, columns=['Gender',
'Self_Employed'], prefix=['Gender', 'Self_Employed'], dtype=int)

df_encoding.sample(5)
```

```
     Married Dependents  Education  ApplicantIncome  CoapplicantIncome
\
259        1          0          0             2167             2400.0

288        1          0          1             3948             1733.0

282        1          1          1             3466             1210.0

323        1          2          1             3283             2035.0

46         1         3+          1             3029                0.0


     LoanAmount  Loan_Amount_Term  Credit_History  Loan_Status  \
259       115.0             360.0             1.0            1
288       149.0             360.0             0.0            0
282       130.0             360.0             1.0            1
323       148.0             360.0             1.0            1
46         99.0             360.0             1.0            1

     Property_Area_Rural  Property_Area_Semiurban  Property_Area_Urban
\
259                    0                        0                    1

288                    1                        0                    0

282                    1                        0                    0

323                    0                        0                    1
```

```
46                     0                        0                    1


     Gender_Female  Gender_Male  Gender_NotGiven  Self_Employed_No  \
259              0            1                0                 1   
288              0            1                0                 1   
282              0            1                0                 0   
323              0            1                0                 1   
46               0            1                0                 1   

     Self_Employed_NotGiven  Self_Employed_Yes  
259                       0                  0  
288                       0                  0  
282                       0                  1  
323                       0                  0  
46                        0                  0  
```

```python
df_encoding['Loan_Amount_Term'].fillna(0, inplace=True)

feature_for_scaling = ['ApplicantIncome', 'CoapplicantIncome',
'LoanAmount', 'Loan_Amount_Term']

for col in feature_for_scaling:
    print(f"{col}:\n{df_encoding[df_encoding[col].isnull()].shape[0]}\
n")
```

```
ApplicantIncome:
0

CoapplicantIncome:
0

LoanAmount:
0

Loan_Amount_Term:
0
```

```python
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

df_encoding[feature_for_scaling] =
scaler.fit_transform(df_encoding[feature_for_scaling])

df_encoding.sample(5)
```

```
     Married Dependents  Education  ApplicantIncome  CoapplicantIncome
\
205        1         3+          1         0.537505          -0.465524
```

```
250        0          0          1        -0.281988          -0.546371

62         1          0          1        -0.438552           0.694141

189        1          0          0        -1.255225           0.708685

201        1          2          1         0.096728          -0.546371


     LoanAmount  Loan_Amount_Term  Credit_History  Loan_Status  \
205    0.671338           0.32791             1.0            1
250   -1.200050           0.32791             0.0            0
62     0.918503           0.32791             1.0            1
189   -0.423247           0.32791             0.0            0
201    0.530102           0.32791             1.0            1

     Property_Area_Rural  Property_Area_Semiurban  Property_Area_Urban
\
205                    0                        1                    0

250                    0                        0                    1

62                     0                        1                    0

189                    0                        0                    1

201                    0                        1                    0


     Gender_Female  Gender_Male  Gender_NotGiven  Self_Employed_No  \
205              0            1                0                 1
250              1            0                0                 1
62               0            1                0                 1
189              0            1                0                 1
201              0            1                0                 1

     Self_Employed_NotGiven  Self_Employed_Yes
205                       0                  0
250                       0                  0
62                        0                  0
189                       0                  0
201                       0                  0
```

```python
df_encoding.Dependents.fillna('NotGiven', inplace=True)

df_encoding.Dependents.isnull().sum()
```

```
0
```

```python
df_encoding = pd.get_dummies(df_encoding, columns=['Dependents'], 
prefix=['Dependents'], dtype=int)
```

```
df_encoding.Credit_History.isnull().sum()
```

```
30
```

```
df_encoding.Credit_History.unique()
```

```
array([ 1., nan,  0.])
```

```
df_encoding.Credit_History.fillna('NotGiven', inplace=True)
```

```
/var/folders/bg/1zs8qp8d26v62zscyhsgmff40000gq/T/
ipykernel_96609/999982145.py:1: FutureWarning: Setting an item of
incompatible dtype is deprecated and will raise an error in a future
version of pandas. Value 'NotGiven' has dtype incompatible with
float64, please explicitly cast to a compatible dtype first.
  df_encoding.Credit_History.fillna('NotGiven', inplace=True)
```

```
df_encoding.Credit_History.unique()
```

```
array([1.0, 'NotGiven', 0.0], dtype=object)
```

```
df_encoding = pd.get_dummies(df_encoding, columns=['Credit_History'],
prefix=['Credit_History'], dtype=int)
```

```
df_encoding.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 381 entries, 0 to 380
Data columns (total 24 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Married                  381 non-null    int64  
 1   Education                381 non-null    int64  
 2   ApplicantIncome          381 non-null    float64
 3   CoapplicantIncome        381 non-null    float64
 4   LoanAmount               381 non-null    float64
 5   Loan_Amount_Term         381 non-null    float64
 6   Loan_Status              381 non-null    int64  
 7   Property_Area_Rural      381 non-null    int64  
 8   Property_Area_Semiurban  381 non-null    int64  
 9   Property_Area_Urban      381 non-null    int64  
 10  Gender_Female            381 non-null    int64  
 11  Gender_Male              381 non-null    int64  
 12  Gender_NotGiven          381 non-null    int64  
 13  Self_Employed_No         381 non-null    int64  
 14  Self_Employed_NotGiven   381 non-null    int64  
 15  Self_Employed_Yes        381 non-null    int64  
 16  Dependents_0             381 non-null    int64  
 17  Dependents_1             381 non-null    int64  
 18  Dependents_2             381 non-null    int64  
 19  Dependents_3+            381 non-null    int64  
 20  Dependents_NotGiven      381 non-null    int64
 21  Credit_History_0.0       381 non-null    int64
 22  Credit_History_1.0       381 non-null    int64
 23  Credit_History_NotGiven  381 non-null    int64
dtypes: float64(4), int64(20)
memory usage: 71.6 KB
```

```
df_encoding.sample(4)
```

```
     Married  Education  ApplicantIncome  CoapplicantIncome
LoanAmount  \
205        1          1         0.537505          -0.465524
0.671338
372        1          1        -0.408932           0.914867   -
1.729688
281        0          0         0.794919           0.306160
1.518759
90         1          1        -0.881446           1.372573   -
0.176083

     Loan_Amount_Term  Loan_Status  Property_Area_Rural  \
205          0.327910            1                    0
372         -1.709055            1                    0
281          0.327910            1                    0
90           0.327910            1                    0

     Property_Area_Semiurban  Property_Area_Urban  ...  \
205                        1                    0  ...
372                        1                    0  ...
281                        1                    0  ...
90                         1                    0  ...

     Self_Employed_NotGiven  Self_Employed_Yes  Dependents_0
Dependents_1  \
205                       0                  0             0
0
372                       0                  0             1
0
281                       0                  0             0
0
90                        0                  0             1
0

     Dependents_2  Dependents_3+  Dependents_NotGiven
Credit_History_0.0  \
205             0              1                    0
0
372             0              0                    0
0
281             0              1                    0
```

```
0
90              0              0                    0
0

     Credit_History_1.0  Credit_History_NotGiven
205                   1                        0
372                   1                        0
281                   1                        0
90                    1                        0

[4 rows x 24 columns]
```

```
df_encoding.columns
```

```
Index(['Married', 'Education', 'ApplicantIncome', 'CoapplicantIncome',
       'LoanAmount', 'Loan_Amount_Term', 'Loan_Status',
'Property_Area_Rural',
       'Property_Area_Semiurban', 'Property_Area_Urban',
'Gender_Female',
       'Gender_Male', 'Gender_NotGiven', 'Self_Employed_No',
       'Self_Employed_NotGiven', 'Self_Employed_Yes', 'Dependents_0',
       'Dependents_1', 'Dependents_2', 'Dependents_3+',
'Dependents_NotGiven',
       'Credit_History_0.0', 'Credit_History_1.0',
'Credit_History_NotGiven'],
      dtype='object')
```

L1 reg are needed