Data Pre-Processing
Handling Categorical Values

Team ID	PNT2022TMID13933
Project Name	Project -Smart Lender - Applicant Credibility Prediction for Loan Approval

One-Hot Encoding:

One-Hot encoding technique is used when the features are nominal. In one hot encoding, for every categorical feature, a new variable is created. Categorical features are mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category. These newly created binary features are known as Dummy variables. This is also known as Dummy encoding.

Label Encoding:

This approach is very simple and it involves converting each value in a column to a number.

For example, if a dataset contains a variable 'Gender' with labels 'Male' and 'Female', then the label encoder would convert these labels into a number format and the resultant outcome would be [0,1].

```
import pandas as pd
import numpy as np
```

```
df = pd.read_csv("C:\\Users\\Komal T\\Downloads\\loan_prediction.csv")
df
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y
...	...	...	...	...	...	...	...	...	...	...	...	...	...
609	LP002978	Female	No	0	Graduate	No	2900	0.0	71.0	360.0	1.0	Rural	Y
610	LP002979	Male	Yes	3+	Graduate	No	4106	0.0	40.0	180.0	1.0	Rural	Y
611	LP002983	Male	Yes	1	Graduate	No	8072	240.0	253.0	360.0	1.0	Urban	Y
612	LP002984	Male	Yes	2	Graduate	No	7583	0.0	187.0	360.0	1.0	Urban	Y
613	LP002990	Female	No	0	Graduate	Yes	4583	0.0	133.0	360.0	0.0	Semiurban	N

614 rows × 13 columns

```
df.head()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y

```
df.tail()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
609	LP002978	Female	No	0	Graduate	No	2900	0.0	71.0	360.0	1.0	Rural	Y
610	LP002979	Male	Yes	3+	Graduate	No	4106	0.0	40.0	180.0	1.0	Rural	Y
611	LP002983	Male	Yes	1	Graduate	No	8072	240.0	253.0	360.0	1.0	Urban	Y
612	LP002984	Male	Yes	2	Graduate	No	7583	0.0	187.0	360.0	1.0	Urban	Y
613	LP002990	Female	No	0	Graduate	Yes	4583	0.0	133.0	360.0	0.0	Semiurban	N

```python
pd.get_dummies(data["Education"])
```

	0	1
0	1	0
1	1	0
2	1	0
3	0	1
4	1	0
...	...	...
609	1	0
610	1	0
611	1	0
612	1	0
613	1	0

614 rows × 2 columns

```python
pd.get_dummies(data["Property_Area"])
```

	0	1	2
0	0	0	1
1	1	0	0
2	0	0	1
3	0	0	1
4	0	0	1
...	...	...	...
609	1	0	0
610	1	0	0
611	0	0	1
612	0	0	1
613	0	1	0

614 rows × 3 columns

```python
from sklearn.preprocessing import LabelEncoder
l=LabelEncoder()
```

```python
l.fit_transform(data["Property_Area"])
```

```
array([2, 0, 2, 2, 2, 2, 2, 1, 2, 1, 2, 2, 2, 0, 2, 2, 2, 2, 0, 2, 2, 2,
       1, 0, 1, 1, 1, 2, 2, 1, 2, 2, 0, 1, 0, 2, 2, 1, 2, 1, 2, 2, 2, 1,
       2, 2, 2, 2, 2, 1, 1, 1, 1, 2, 2, 1, 1, 0, 2, 2, 2, 2, 0, 0, 1, 1,
       2, 2, 2, 1, 2, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 2, 1, 2, 1, 1, 1,
       2, 1, 1, 1, 2, 1, 1, 2, 1, 1, 1, 1, 2, 1, 2, 1, 2, 2, 2, 0, 2, 1,
       2, 1, 0, 1, 1, 0, 1, 2, 0, 2, 0, 1, 1, 1, 0, 0, 0, 0, 2, 0, 2, 2,
       1, 1, 1, 1, 0, 2, 1, 0, 0, 2, 1, 1, 2, 1, 2, 2, 0, 1, 0, 0, 2, 0,
       2, 1, 0, 2, 0, 1, 1, 2, 1, 0, 2, 0, 0, 0, 1, 1, 0, 2, 0, 1, 1, 0,
       0, 1, 1, 2, 2, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 2, 1, 0, 1, 0, 2,
       1, 2, 1, 1, 2, 2, 1, 1, 2, 0, 2, 1, 1, 1, 2, 0, 2, 1, 0, 1, 1, 1,
       2, 1, 1, 1, 1, 0, 2, 1, 1, 0, 1, 0, 0, 1, 1, 0, 2, 2, 0, 1, 0, 2,
       2, 0, 1, 2, 2, 2, 1, 2, 1, 2, 0, 1, 2, 0, 0, 2, 0, 1, 2, 1, 1, 0,
       1, 0, 1, 2, 0, 2, 2, 2, 0, 1, 1, 1, 1, 2, 1, 0, 2, 1, 2, 2, 0, 0,
       1, 0, 1, 0, 0, 1, 2, 2, 1, 2, 1, 2, 0, 2, 2, 1, 0, 2, 0, 2, 0, 2,
       0, 0, 1, 1, 0, 0, 0, 2, 1, 2, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 2, 2,
       2, 1, 2, 2, 2, 1, 0, 0, 2, 1, 0, 0, 2, 1, 0, 1, 0, 2, 1, 0, 1, 0,
       0, 0, 1, 2, 0, 2, 2, 1, 1, 1, 2, 2, 0, 0, 1, 0, 1, 0, 1, 1, 0, 2,
       2, 2, 0, 1, 2, 2, 1, 1, 2, 2, 2, 2, 1, 2, 2, 0, 0, 0, 2, 1, 2, 1,
       2, 2, 0, 1, 2, 0, 1, 1, 0, 1, 2, 0, 1, 0, 1, 2, 0, 0, 1, 2, 2, 2,
       0, 1, 0, 2, 2, 2, 1, 0, 0, 1, 0, 2, 1, 0, 1, 1, 2, 1, 1, 2, 2, 0,
       1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 2, 0, 0, 1, 1, 2, 2, 0, 1, 1, 2,
       0, 1, 1, 0, 2, 1, 1, 2, 1, 0, 1, 2, 0, 0, 1, 1, 1, 2, 0, 0, 1, 1,
       1, 0, 0, 2, 1, 2, 1, 2, 0, 1, 0, 1, 0, 2, 1, 0, 0, 1, 1, 0, 1, 0,
       2, 2, 2, 2, 0, 1, 2, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1,
       1, 1, 0, 2, 0, 1, 2, 0, 2, 1, 0, 0, 1, 1, 1, 2, 1, 0, 1, 0, 1, 0,
       0, 0, 2, 2, 0, 1, 2, 1, 1, 1, 1, 1, 0, 1, 2, 0, 2, 0, 2, 2, 2, 2,
       2, 1, 1, 2, 1, 2, 0, 2, 1, 2, 1, 0, 0, 0, 2, 1, 1, 1, 1, 1, 1, 0,
       2, 0, 0, 1, 0, 2, 2, 0, 2, 0, 1, 2, 1, 0, 0, 0, 0, 2, 2, 1],
      dtype=int64)
```

```python
for col in data:
    l=LabelEncoder()
    data[col]=l.fit_transform(data[col])
```

```python
data.head()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	Property_Area	Loan_Status
0	0	1	0	0	0	0	2	1
1	1	1	1	1	0	0	0	0
2	2	1	1	0	0	1	2	1
3	3	1	1	0	1	0	2	1
4	4	1	0	0	0	0	2	1

```
df.dtypes
```

```
Loan_ID               object
Gender                object
Married               object
Dependents            object
Education             object
Self_Employed         object
ApplicantIncome        int64
CoapplicantIncome    float64
LoanAmount           float64
Loan_Amount_Term     float64
Credit_History       float64
Property_Area         object
Loan_Status           object
dtype: object
```

```
df['Gender'] = df['Gender'].astype('category')
df.dtypes
```

```
Loan_ID                object
Gender               category
Married                object
Dependents             object
Education              object
Self_Employed          object
ApplicantIncome         int64
CoapplicantIncome     float64
LoanAmount            float64
Loan_Amount_Term      float64
Credit_History        float64
Property_Area          object
Loan_Status            object
dtype: object
```

```
df['Gender'] = df['Gender'].cat.codes
print(df)
```

```
      Loan_ID  Gender Married Dependents     Education Self_Employed  \
0    LP001002       2      No          0      Graduate            No
1    LP001003       2     Yes          1      Graduate            No
2    LP001005       2     Yes          0      Graduate           Yes
3    LP001006       2     Yes          0  Not Graduate            No
4    LP001008       2      No          0      Graduate            No
..        ...     ...     ...        ...           ...           ...
609  LP002978       1      No          0      Graduate            No
610  LP002979       2     Yes         3+      Graduate            No
611  LP002983       2     Yes          1      Graduate            No
612  LP002984       2     Yes          2      Graduate            No
613  LP002990       1      No          0      Graduate           Yes

     ApplicantIncome  CoapplicantIncome  LoanAmount  Loan_Amount_Term  \
0               5849                0.0         NaN             360.0
1               4583             1508.0       128.0             360.0
2               3000                0.0        66.0             360.0
3               2583             2358.0       120.0             360.0
4               6000                0.0       141.0             360.0
..               ...                ...         ...               ...
609             2900                0.0        71.0             360.0
610             4106                0.0        40.0             180.0
611             8072              240.0       253.0             360.0
612             7583                0.0       187.0             360.0
613             4583                0.0       133.0             360.0

     Credit_History Property_Area Loan_Status
0               1.0         Urban           Y
1               1.0         Rural           N
2               1.0         Urban           Y
3               1.0         Urban           Y
4               1.0         Urban           Y
..              ...           ...         ...
609             1.0         Rural           Y
610             1.0         Rural           Y
611             1.0         Urban           Y
612             1.0         Urban           Y
613             0.0     Semiurban           N

[614 rows x 13 columns]
```