

```
In [16]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [17]: bank_df=pd.read_csv(r"F:\FSDS\Data Files\bank.csv",sep=';')
```

```
In [18]: bank_df
```

```
Out[18]:
```

	age	job	marital	education	default	balance	housing	loan	contact	day	month
0	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct
1	33	services	married	secondary	no	4789	yes	yes	cellular	11	may
2	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr
3	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun
4	59	blue-collar	married	secondary	no	0	yes	no	unknown	5	may
...
4516	33	services	married	secondary	no	-333	yes	no	cellular	30	jul
4517	57	self-employed	married	tertiary	yes	-3313	yes	yes	unknown	9	may
4518	57	technician	married	secondary	no	295	no	no	cellular	19	aug
4519	28	blue-collar	married	secondary	no	1137	no	no	cellular	6	feb
4520	44	entrepreneur	single	tertiary	no	1136	yes	yes	cellular	3	apr

4521 rows × 17 columns

```
In [19]: cat=bank_df.select_dtypes(include='object').columns
num=bank_df.select_dtypes(exclude='object').columns
```

```
In [20]: cat
```

```
Out[20]: Index(['job', 'marital', 'education', 'default', 'housing', 'loan', 'contact',
               'month', 'poutcome', 'y'],
               dtype='object')
```

```
In [21]: num
```

```
Out[21]: Index(['age', 'balance', 'day', 'duration', 'campaign', 'pdays', 'previous'], dtype
               ='object')
```

LabelEncoder

```
In [22]: from sklearn.preprocessing import LabelEncoder
```

```
In [23]: le=LabelEncoder()
```

```
In [24]: bank_df['y']=le.fit_transform(bank_df['y'])
```

```
In [25]: bank_df
```

```
Out[25]:
```

	age	job	marital	education	default	balance	housing	loan	contact	day	month
0	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct
1	33	services	married	secondary	no	4789	yes	yes	cellular	11	may
2	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr
3	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun
4	59	blue-collar	married	secondary	no	0	yes	no	unknown	5	may
...
4516	33	services	married	secondary	no	-333	yes	no	cellular	30	jul
4517	57	self-employed	married	tertiary	yes	-3313	yes	yes	unknown	9	may
4518	57	technician	married	secondary	no	295	no	no	cellular	19	aug
4519	28	blue-collar	married	secondary	no	1137	no	no	cellular	6	feb
4520	44	entrepreneur	single	tertiary	no	1136	yes	yes	cellular	3	apr

4521 rows × 17 columns

```
In [26]: bank_df=pd.read_csv(r"F:\FSDS\Data Files\bank.csv",sep=';')
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
bank_df['y']=le.fit_transform(bank_df['y'])
```

```
In [28]: bank_df=pd.read_csv(r"F:\FSDS\Data Files\bank.csv",sep=';')
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
le.fit_transform(bank_df)
```

```

-----
ValueError                                Traceback (most recent call last)
Cell In [28], line 4
      2 from sklearn.preprocessing import LabelEncoder
      3 le=LabelEncoder()
----> 4 le.fit_transform(bank_df)

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\preprocessing\
_label.py:114, in LabelEncoder.fit_transform(self, y)
    101 def fit_transform(self, y):
    102     """Fit label encoder and return encoded labels.
    103
    104     Parameters
    (...)
    112         Encoded labels.
    113     """
--> 114     y = column_or_1d(y, warn=True)
    115     self.classes_, y = _unique(y, return_inverse=True)
    116     return y

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\utils\valida
tion.py:1406, in column_or_1d(y, dtype, warn)
    1395         warnings.warn(
    1396             (
    1397                 "A column-vector y was passed when a 1d array was"
    (...)
    1402             stacklevel=2,
    1403         )
    1404         return _asarray_with_order(xp.reshape(y, (-1,)), order="C", xp=xp)
-> 1406 raise ValueError(
    1407     "y should be a 1d array, got an array of shape {} instead.".format(shape)
    1408 )

ValueError: y should be a 1d array, got an array of shape (4521, 17) instead.

```

In [27]: bank_df

Out[27]:

	age	job	marital	education	default	balance	housing	loan	contact	day	month
0	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct
1	33	services	married	secondary	no	4789	yes	yes	cellular	11	may
2	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr
3	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun
4	59	blue-collar	married	secondary	no	0	yes	no	unknown	5	may
...
4516	33	services	married	secondary	no	-333	yes	no	cellular	30	jul
4517	57	self-employed	married	tertiary	yes	-3313	yes	yes	unknown	9	may
4518	57	technician	married	secondary	no	295	no	no	cellular	19	aug
4519	28	blue-collar	married	secondary	no	1137	no	no	cellular	6	feb
4520	44	entrepreneur	single	tertiary	no	1136	yes	yes	cellular	3	apr

4521 rows × 17 columns

In [29]: `bank_df['y'].values`

Out[29]: `array(['no', 'no', 'no', ..., 'no', 'no', 'no'], dtype=object)`

```
In [30]: bank_df=pd.read_csv(r"F:\FSDS\Data Files\bank.csv",sep=';')
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
for i in cat[1:]:
    bank_df[i]=le.fit_transform(bank_df[i])
```

In [31]: `bank_df`

Out[31]:

	age	job	marital	education	default	balance	housing	loan	contact	day	month
0	30	unemployed	1	0	0	1787	0	0	0	19	10
1	33	services	1	1	0	4789	1	1	0	11	8
2	35	management	2	2	0	1350	1	0	0	16	0
3	30	management	1	2	0	1476	1	1	2	3	6
4	59	blue-collar	1	1	0	0	1	0	2	5	8
...
4516	33	services	1	1	0	-333	1	0	0	30	5
4517	57	self-employed	1	2	1	-3313	1	1	2	9	8
4518	57	technician	1	1	0	295	0	0	0	19	1
4519	28	blue-collar	1	1	0	1137	0	0	0	6	3
4520	44	entrepreneur	2	2	0	1136	1	1	0	3	0

4521 rows × 17 columns

In [33]:

```
bank_df=pd.read_csv(r"F:\FSDS\Data Files\bank.csv",sep=';')
pd.get_dummies(bank_df, dtype='int')
```

Out[33]:

	age	balance	day	duration	campaign	pdays	previous	job_admin.	job_blue-collar	job_entrepreneur
0	30	1787	19	79	1	-1	0	0	0	
1	33	4789	11	220	1	339	4	0	0	
2	35	1350	16	185	1	330	1	0	0	
3	30	1476	3	199	4	-1	0	0	0	
4	59	0	5	226	1	-1	0	0	1	
...
4516	33	-333	30	329	5	-1	0	0	0	
4517	57	-3313	9	153	1	-1	0	0	0	
4518	57	295	19	151	11	-1	0	0	0	
4519	28	1137	6	129	4	211	3	0	1	
4520	44	1136	3	345	2	249	7	0	0	

4521 rows × 53 columns

In []: