

In [1]:

```
# Importing the Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score
from sklearn.model_selection import train_test_split
from sklearn.datasets import make_classification
```

In [2]:

```
bank=pd.read_csv('C:/Users/Hp/Downloads/bank-full.csv',sep=";")
bank.head(10)
```

Out[2]:

	age	job	marital	education	default	balance	housing	loan	contact	day	mont
0	58	management	married	tertiary	no	2143	yes	no	unknown	5	ma
1	44	technician	single	secondary	no	29	yes	no	unknown	5	ma
2	33	entrepreneur	married	secondary	no	2	yes	yes	unknown	5	ma
3	47	blue-collar	married	unknown	no	1506	yes	no	unknown	5	ma
4	33	unknown	single	unknown	no	1	no	no	unknown	5	ma
5	35	management	married	tertiary	no	231	yes	no	unknown	5	ma
6	28	management	single	tertiary	no	447	yes	yes	unknown	5	ma
7	42	entrepreneur	divorced	tertiary	yes	2	yes	no	unknown	5	ma
8	58	retired	married	primary	no	121	yes	no	unknown	5	ma
9	43	technician	single	secondary	no	593	yes	no	unknown	5	ma

Generate the dataset

In [3]:

```
x, y= make_classification(
    n_samples=100,
    n_features=1,
    n_classes=2,
    n_clusters_per_class=1,
    flip_y=0.03,
    n_informative=1,
    n_redundant=0,
    n_repeated=0,
)
print(y)
```

```
[1 1 0 1 0 1 0 1 1 1 1 0 1 1 0 0 0 0 0 1 0 1 1 1 1 1 1 1 0 0 0 1 1 0 1 0 1
 1 0 0 1 0 0 0 1 1 0 0 1 0 0 0 1 0 0 1 1 1 0 1 0 0 1 0 0 0 1 1 1 0 1 0 0 0
 1 0 0 1 0 1 0 0 0 1 0 1 0 1 1 0 1 0 1 0 1 1 1 1 1 0 1]
```

EDA

In [4]:

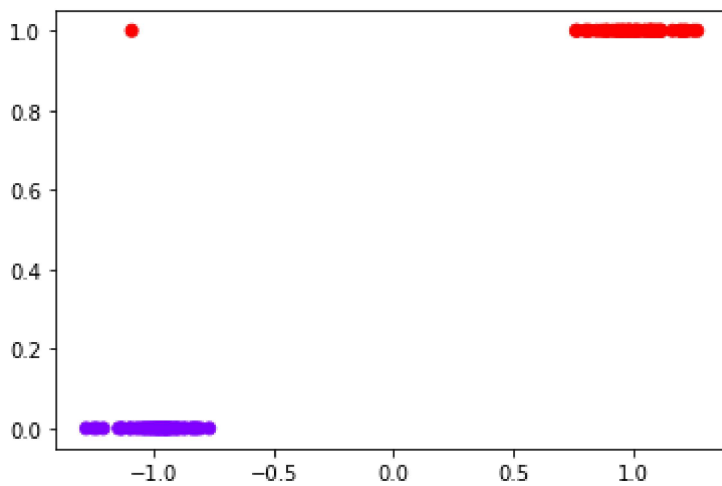
```
bank.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45211 entries, 0 to 45210
Data columns (total 17 columns):
#   Column      Non-Null Count  Dtype
---  -
0   age         45211 non-null  int64
1   job         45211 non-null  object
2   marital     45211 non-null  object
3   education   45211 non-null  object
4   default     45211 non-null  object
5   balance     45211 non-null  int64
6   housing     45211 non-null  object
7   loan        45211 non-null  object
8   contact     45211 non-null  object
9   day         45211 non-null  int64
10  month       45211 non-null  object
11  duration    45211 non-null  int64
12  campaign    45211 non-null  int64
13  pdays      45211 non-null  int64
..   ..         ..         ..         ..
```

visualize the dataset

In [5]:

```
plt.scatter(x,y,c=y,cmap='rainbow')  
plt.show()
```



split the dataset

In [6]:

```
x_train, x_test, y_train, y_test = train_test_split(x, y, random_state=1)
```

In [18]:

```
x_train.shape
```

Out[18]:

(75, 1)

perform logistic regression

In [22]:

```
log_reg = LogisticRegression()  
log_reg.fit(x_train, y_train)
```

Out[22]:

LogisticRegression()

In [26]:

```
print(log_reg.coef_)  
print(log_reg.intercept_)
```

```
[[2.9079778]]  
[0.26649412]
```

make prediction using model

In [27]:

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

display the confusion matrix

In [28]:

```
confusion_matrix(y_test, y_pred)
```

Out[28]:

```
array([[14,  0],  
       [ 0, 11]], dtype=int64)
```

In []: