

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [3]: from sklearn.linear_model import LogisticRegression
```

```
In [5]: df=pd.read_csv(r"C:\Users\Admin\Downloads\C3_bot_detection_data - C3_bot_detect  
df
```

Out[5]:

	User ID	Username	Tweet	Retweet Count	Mention Count	Follower Count	Verified	Bot Label	Loc
0	132131	flong	Station activity person against natural majori...	85	1	2353	False	1	Adki
1	289683	hinesstephanie	Authority research natural life material staff...	55	5	9617	True	0	Sand
2	779715	roberttran	Manage whose quickly especially foot none to g...	6	2	4363	True	0	Harris
3	696168	pmason	Just cover eight opportunity strong policy which.	54	5	2242	True	1	Martine
4	704441	noah87	Animal sign six data good or.	26	3	8438	False	1	Camact
...
49995	491196	uberg	Want but put card direction know miss former h...	64	0	9911	True	1	Kimberly
49996	739297	jessicamunoz	Provide whole maybe agree church respond most ...	18	5	9900	False	1	Gree
49997	674475	lynn cunningham	Bring different everyone international capital...	43	3	6313	True	1	Debor
49998	167081	richardthompson	Than about single generation itself seek sell ...	45	1	6343	False	0	Stephe
49999	311204	daniel29	Here morning class various room human true bec...	91	4	4006	False	0	Nova

50000 rows × 11 columns

```
In [6]: d1=df[['User ID','Retweet Count','Mention Count','Follower Count','Bot Label']]  
        d2=df['Verified']
```

```
In [7]: d1.shape
```

```
Out[7]: (50000, 5)
```

```
In [8]: d2.shape
```

```
Out[8]: (50000,)
```

```
In [9]: from sklearn.preprocessing import StandardScaler
```

```
In [10]: a=StandardScaler().fit_transform(d1)
```

```
In [11]: lr=LogisticRegression()  
        lr.fit(a,d2)
```

```
Out[11]: LogisticRegression()
```

```
In [16]: obs=[[10,12,13,14,5]]
```

```
In [17]: pdt=lr.predict(obs)  
        print(pdt)  
  
[ True]
```

```
In [18]: lr.classes_
```

```
Out[18]: array([False,  True])
```

```
In [19]: lr.predict_proba(obs)[0][0]
```

```
Out[19]: 0.4009324960343088
```

```
In [20]: lr.predict_proba(obs)[0][1]
```

```
Out[20]: 0.5990675039656912
```

```
In [ ]:
```