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

In [2]:
```
from sklearn.linear_model import LogisticRegression
```

In [3]:
```
df=pd.read_csv("detection.csv").dropna()

df
```

3	696168	pmason	opportunity strong policy which.	54	5	2242	True	1	Martinezberg	08- 22:27:
4	704441	noah87	Animal sign six data good or.	26	3	8438	False	1	Camachoville	202 04- 21:24:
5	570928	james00	See wonder travel this suffer less yard office...	41	4	3792	True	1	West Cheyenne	202 05- 22:24:
...	...	...	...	...	...	...	...	...	...	
49995	491196	uberg	Want but put card direction know miss former h...	64	0	9911	True	1	Lake Kimberlyburgh	202 04- 11:06:
			Provide							

In [4]:
```
df.dropna(inplace=True)
```

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 41659 entries, 1 to 49999
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   User ID         41659 non-null  int64 
 1   Username        41659 non-null  object
 2   Tweet           41659 non-null  object
 3   Retweet Count   41659 non-null  int64 
 4   Mention Count   41659 non-null  int64 
 5   Follower Count  41659 non-null  int64 
 6   Verified        41659 non-null  bool  
 7   Bot Label       41659 non-null  int64 
 8   Location        41659 non-null  object
 9   Created At      41659 non-null  object
 10  Hashtags        41659 non-null  object
dtypes: bool(1), int64(5), object(5)
memory usage: 3.5+ MB
```

```
In [19]: feature_matrix = df[['User ID','Retweet Count','Mention Count','Follower Count','Bot Label
         target_vector = df['Verified']
```

```
In [20]: feature_matrix.shape
```

```
Out[20]: (41659, 5)
```

```
In [21]: target_vector.shape
```

```
Out[21]: (41659,)
```

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

```
In [23]: fs = StandardScaler().fit_transform(feature_matrix)
```

```
In [24]: logr = LogisticRegression()
         logr.fit(fs,target_vector)
```

```
Out[24]: LogisticRegression()
```

```
In [25]: feature_matrix.shape
```

```
Out[25]: (41659, 5)
```

```
In [26]: target_vector.shape
```

```
Out[26]: (41659,)
```

```
In [27]: from sklearn.preprocessing import StandardScaler
```

```
In [28]: fs = StandardScaler().fit_transform(feature_matrix)
```

```
In [29]: logr = LogisticRegression()
         logr.fit(fs,target_vector)
```

```
Out[29]: LogisticRegression()
```

```
In [31]: observation=df[['User ID','Retweet Count','Mention Count','Follower Count','Bot Label']]
```

```
In [32]: prediction = logr.predict(observation)
         prediction
```

```
Out[32]: array([ True,  True,  True, ...,  True,  True,  True])
```

```
In [33]: logr.classes_
```

```
Out[33]: array([False,  True])
```

```
In [34]: logr.predict_proba(observation)[0][1]
```

```
Out[34]: 1.0
```