

In [69]:

```
import numpy as np
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
```

In [70]:

```
df=pd.read_csv(r"C:\Users\mural\Downloads\archive.zip")
df
```

221	0	0	1.00000	-1.00000	1.00000	1.00000	1.00000	-1.00000	1.00000	1.00000	1.00000	-1.00000
222	1	0	0.97467	0.13082	0.94120	0.20036	0.88783	0.32248	0.89009	0.32711	0.85550	0.4521
223	0	0	0.00000	0.00000	1.00000	1.00000	0.00000	0.00000	1.00000	1.00000	0.00000	0.0000
224	1	0	0.92308	0.15451	0.86399	0.29757	0.72582	0.36790	0.70588	0.56830	0.57449	0.6271
225	0	0	0.00000	0.00000	1.00000	1.00000	1.00000	1.00000	0.00000	0.00000	0.00000	0.0000
226	1	0	0.88804	0.38138	0.65926	0.69431	0.29148	0.87892	-0.06726	0.90135	-0.39597	0.8044
227	0	0	1.00000	-1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.0000
228	1	0	0.73523	-0.38293	0.80151	0.10278	0.78826	0.15266	0.55580	0.05252	1.00000	0.2122
229	0	0	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.0000
230	1	0	0.94649	0.00892	0.97287	-0.00260	0.98922	0.00372	0.95801	0.01598	0.94054	0.0353
231	0	0	1.00000	1.00000	-1.00000	-1.00000	-1.00000	-1.00000	0.00000	0.00000	0.00000	0.0000
232	1	0	0.50466	-0.16900	0.71442	0.01513	0.71063	0.02258	0.68065	0.01282	0.34615	0.0559

In [71]:

```
pd.set_option('display.max_rows',10000000000)
pd.set_option('display.max_columns',10000000000)
pd.set_option('display.width',95)
```

In [72]:

```
print('This DataFrame has %d Rows and %d columns'%(df.shape))
```

This DataFrame has 350 Rows and 35 columns

In [73]:

```
df.head()
```

Out[73]:

	1	0	0.99539	-0.05889	0.85243	0.02306	0.83398	-0.37708	1.1	0.03760	0.85243.
0	1	0	1.00000	-0.18829	0.93035	-0.36156	-0.10868	-0.93597	1.00000	-0.04549	0.5087
1	1	0	1.00000	-0.03365	1.00000	0.00485	1.00000	-0.12062	0.88965	0.01198	0.7308
2	1	0	1.00000	-0.45161	1.00000	1.00000	0.71216	-1.00000	0.00000	0.00000	0.0000
3	1	0	1.00000	-0.02401	0.94140	0.06531	0.92106	-0.23255	0.77152	-0.16399	0.5279
4	1	0	0.02337	-0.00592	-0.09924	-0.11949	-0.00763	-0.11824	0.14706	0.06637	0.0378

In [74]:

```
features_matrix = df.iloc[:,0:34]
```

In [75]:

```
target_vector = df.iloc[:, -1]
```

In [76]:

```
print('The Features Matrix Has %d Rows And %d columns(s)%(features_matrix.shape))
print('The Target Matrix Has %d Rows And %d Columns(s)%(np.array(target_vector).reshape
```

The Features Matrix Has 350 Rows And 34 columns(s)  
 The Target Matrix Has 350 Rows And 1 Columns(s)

In [77]:

```
features_matrix_standardized = StandardScaler().fit_transform(features_matrix)
```

In [78]:

```
algorithm = LogisticRegression(penalty=None,dual=False, tol=1e-4,C=1.0, fit_intercept=True,
                               class_weight=None,random_state=None,solver='lbfgs',max_iter=1000,
                               verbose=0, warm_start=False, n_jobs=None,l1_ratio=None)
```

In [79]:

```
Logistic_Regression_Model = algorithm.fit(features_matrix_standardized,target_vector)
```

In [88]:

```
observation = [[1, 0, 0.99539, -0.05889, 0.8524299999999999, 0.02306, 0.8339799999999999,
0.8524299999999999, -0.17755, 0.59755, -0.44945, 0.60536, -0.38223, 0.84
0.58212, -0.32192, 0.56971, -0.29674, 0.36946, -0.47357, 0.56811, -0.511
-0.46168000000000003, 0.21266, -0.3409,0.112267,-0.54487,0.18641,-0.453]]
```

In [89]:

```
predictions = Logistic_Regression_Model.predict(observation)
print('The Model predicted The observation To Belong To Class %s'%(predictions))
```

The Model predicted The observation To Belong To Class ['g']

In [90]:

```
print('The Algorithm Was Trained To predict The One Of The Classes: %s'%(algorithm.class
```

The Algorithm Was Trained To predict The One Of The Classes: ['b' 'g']

In [91]:

```
print("""The Model Says The Probability Of The observation We Passed belonging To The Cl
print()
print("""The Model Says The Probability Of The observation We Passed belonging To The Cl
```

The Model Says The Probability Of The observation We Passed belonging To  
The Class ['b'] is 4.3938410904331704e-05

The Model Says The Probability Of The observation We Passed belonging To  
The Class ['g'] is 0.9999560615890957

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: