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

In [2]:
```python
dataset = pd.read_csv('D:/data.csv')
dataset.head()
```

Out[2]:

	Area Code	Name	factor1	factor2	Pollution
0	15624510	street	19	19000	0
1	15810944	street	35	20000	0
2	15668575	street	26	43000	0
3	15603246	street	27	57000	0
4	15804002	street	19	76000	0

In [3]:
```python
x = dataset.iloc[:, [2, 3]].values
y = dataset.iloc[:, 4].values
```

In [4]:
```python
from sklearn.model_selection import train_test_split
xtrain, xtest, ytrain, ytest = train_test_split(x, y, test_size = 0.25, random_state =
```

In [5]:
```python
from sklearn.preprocessing import StandardScaler
sc_x = StandardScaler()
xtrain = sc_x.fit_transform(xtrain)
xtest = sc_x.transform(xtest)
print (xtrain[0:10, :])
```

```
[[ 0.58164944 -0.88670699]
 [-0.60673761  1.46173768]
 [-0.01254409 -0.5677824 ]
 [-0.60673761  1.89663484]
 [ 1.37390747 -1.40858358]
 [ 1.47293972  0.99784738]
 [ 0.08648817 -0.79972756]
 [-0.01254409 -0.24885782]
 [-0.21060859 -0.5677824 ]
 [-0.21060859 -0.19087153]]
```

In [6]:
```python
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(xtrain, ytrain)
```

Out[6]:
```
LogisticRegression(random_state=0)
```

In [7]:
```python
y_pred = classifier.predict(xtest)
```

In [8]:
```python
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(ytest, y_pred)
print ("Confusion Matrix : \n", cm)
```

```
Confusion Matrix :
 [[65  3]
 [ 8 24]]
```

In [9]:
```python
from sklearn.metrics import accuracy_score
print ("Accuracy : ", accuracy_score(ytest, y_pred))
```

Accuracy : 0.89

In [ ]: