In [1]:
```
#imports
import pandas as pd
import numpy as np
from sklearn.linear_model import Perceptron
```

In [15]:
```
# Create the dataset
dict = {'Input': ['x1', 'x2', 'x3'],
       'Viagra': [1, 0, 0],
       'Learning': [0, 1, 0],
       'The': [1, 1, 0],
        'Dating': [0, 0, 0],
       'Nigerian': [0, 0, 1],
       'Spam': [1, -1, 1]}

data = pd.DataFrame(dict)
data
```

Out[15]:

	Input	Viagra	Learning	The	Dating	Nigerian	Spam
0	x1	1	0	1	0	0	1
1	x2	0	1	1	0	0	-1
2	x3	0	0	0	0	1	1

In [24]:
```
# Defininig the target input and the output
data_i = pd.DataFrame(data, columns=['Viagra', 'Learning', 'The', 'Dating', 'N
igerian'])
output = pd.DataFrame(data, columns=['Spam'])
```

In [25]:
```
data_i
```

Out[25]:

	Viagra	Learning	The	Dating	Nigerian
0	1	0	1	0	0
1	0	1	1	0	0
2	0	0	0	0	1

In [26]:
```
output
```

Out[26]:

	Spam
0	1
1	-1
2	1

In [27]:
```
# Splitting The data into training and testing
from sklearn.model_selection import train_test_split
```

In [28]:
```
data_i_train, data_i_test, output_train, output_test = train_test_split(data
, output, test_size=0.3, random_state=0)
```

In [29]:
```
data_i_train
```

Out[29]:

	Viagra	Learning	The	Dating	Nigerian
1	0	1	1	0	0
0	1	0	1	0	0

In [30]:
```
#Creating and the model
model = Perceptron(tol=1e-3, random_state=0)
model
```

Out[30]:
```
Perceptron()
```

In [33]:
```
#fitting the model
model.fit(data_i, output)
```

```
C:\Users\Miss Gloriah\anaconda3\lib\site-packages\sklearn\utils\validation.p
y:993: DataConversionWarning: A column-vector y was passed when a 1d array wa
s expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

Out[33]:
```
Perceptron()
```

In [36]:
```
#model prediction
output_pred = model.predict(data_i_train)
output_pred
```

Out[36]:
```
array([-1,  1], dtype=int64)
```

In [39]:
```
# Predicting a different array
new = model.predict([[1, 1, 0, 1, 0]])
new
```

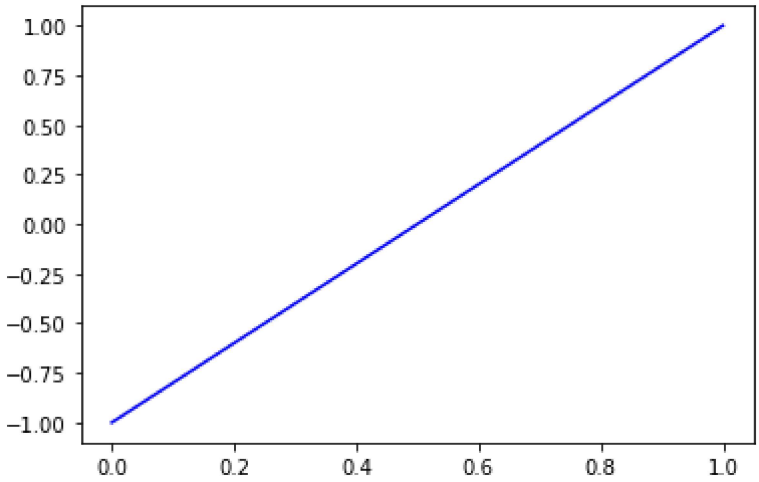
```
C:\Users\Miss Gloriah\anaconda3\lib\site-packages\sklearn\base.py:450: UserW
rning: X does not have valid feature names, but Perceptron was fitted with fe
ature names
  warnings.warn(
```

Out[39]:
```
array([1], dtype=int64)
```

In [41]:
```
#checing the model score
model.score(data_i, output)
```

Out[41]:
```
1.0
```

In [43]:
```
import matplotlib.pyplot as plt
plt.plot(data_i_train['Viagra'], output_pred, color='blue')
plt.Xlabel =('Actual Values')
plt.Ylabel = ('Predicted Outcome')
plt.show()
```



In [ ]: