Implementing XOR Using Perceptron

There must be only some minor changes in labels and our Perceptron is ready for XOR

As XOR is the non-linear function it can’t be implemented using perceptron as it can only be implemented for linearly separable functions.

Here is the ScreenShot of changes and it outputs :-

Changes :-

Just make some changes the labels according to truth table of XOR

```
training_inputs = []
training_inputs.append(np.array([1,1]))
training_inputs.append(np.array([0,1]))
#training_inputs.append(np.array([1,0]))
training_inputs.append(np.array([0,0]))

labels  = np.array([0,1,1,0])

perceptron = Perceptron(2)
perceptron.train(training_inputs,labels)

inputs = np.array([1,0])
print(perceptron.predict(inputs))
```

When we run this code for Inputs : -

inputs = np.array[1,0]
inputs = np.array[0,0]

We get the following output :-

```
In [14]: runfile('/home/thisisbrijraj/Documents/
PythonProjects/RoyalAI/Perceptron2/run2.py',
wdir='/home/thisisbrijraj/Documents/PythonProjects/
RoyalAI/Perceptron2')
Reloaded modules: perceptron2
0
```

For input[1,0]

```
In [22]: runfile('/home/thisisbrijraj/Documents/
PythonProjects/RoyalAI/Perceptron3/run3.py',
wdir='/home/thisisbrijraj/Documents/PythonProjects/
RoyalAI/Perceptron3')
Reloaded modules: perceptron3
1
```

For input[0,0]

So the conclusion is that Perceptron doesn’t work for Non Linearly separable functions