What is a neural network?

Neural networks form the base of deep learning, which is a subfield of machine learning , where the structure of the human brain inspires the algorithms. Neural networks take input data, train themselves to recognize patterns found in the data, and then predict the output for a new set of similar data. Therefore, a neural network can be thought of as the functional unit of deep learning, which mimics the behavior of the human brain to solve complex data-driven problems.

In [24]:

```
import numpy as np
```

In [25]:

```
input_features = np.array([[0,0],[0,1],[1,0],[1,1]])
```

In [26]:

```
print (input_features.shape)
print (input_features)
```

```
(4, 2)
[[0 0]
 [0 1]
 [1 0]
 [1 1]]
```

In [27]:

```
target_output = np.array([[0,1,1,1]])
```

In [28]:

```
target_output = target_output.reshape(4,1)
```

In [29]:

```
print(target_output.shape)
print (target_output)
```

```
(4, 1)
[[0]
 [1]
 [1]
 [1]]
```

In [30]:

```
weights = np.array([[0.1],[0.2]])
print(weights.shape)
print (weights)
bias = 0.3
lr = 0.05
```

```
(2, 1)
[[0.1]
 [0.2]]
```

In [31]:

```
def sigmoid(x):
 return 1/(1+np.exp(-x))
```

In [32]:

```
def sigmoid_der(x):
 return sigmoid(x)*(1-sigmoid(x))
```

In [33]:

```
inputs = input_features
in_o = np.dot(inputs, weights) + bias
out_o = sigmoid(in_o)
```

In [34]:

```
error = out_o - target_output
```

In [35]:

```
x = error.sum()
print(x)
```

```
-0.558754185648239
```

In [36]:

```
derror_douto = error
douto_dino = sigmoid_der(out_o)
```

In [37]:

```
deriv = derror_douto * douto_dino
inputs = input_features.T
deriv_final = np.dot(inputs,deriv)
```

In [38]:

```
weights -= lr * deriv_final
for i in deriv:
 bias -= lr * i
```

In [39]:

```
print (bias)
single_point = np.array([1,0])
result1 = np.dot(single_point, weights) + bias
result2 = sigmoid(result1)
print(result2)
single_point = np.array([1,1])
result1 = np.dot(single_point, weights) + bias
result2 = sigmoid(result1)
print(result2)
single_point = np.array([0,0])
result1 = np.dot(single_point, weights) + bias
result2 = sigmoid(result1)
print(result2)
```

```
[0.30626124]
[0.60225064]
[0.65093229]
[0.57597241]
```