

```
In [1]: import random
import matplotlib
import matplotlib.pyplot as plt
```

```
In [2]: class Net:
    def __init__(self):
        self.w1 = 0
        self.w2 = 0
        self.b = 0

    def forward(self, train):
        wsum = self.w1 * train[0] + self.w2 * train[1] + self.b
        pred = 1 if wsum >= 0 else 0
        return (pred, wsum)
```

```
In [3]: gates = {'and': [[(0, 0), (0, 1), (1, 0), (1, 1)], [0, 0, 0, 1]],
                  'or':  [[(0, 0), (0, 1), (1, 0), (1, 1)], [0, 1, 1, 1]],
                  'nand': [[(0, 0), (0, 1), (1, 0), (1, 1)], [1, 1, 1, 0]],
                  'nor':  [[(0, 0), (0, 1), (1, 0), (1, 1)], [1, 0, 0, 0]]}
```

```
In [4]: def nn(data):
    x, y = data[0], data[1]

    net = Net()
    learning_rate = 0.1

    epoch = 40
    for i in range(epoch):
        index = random.randint(0, 3)
        train = x[index]
        test = y[index]

        predicted, wsum = net.forward(train)
        expected = test

        net.w1 += learning_rate * (expected - predicted) * train[0]
        net.w2 += learning_rate * (expected - predicted) * train[1]
        net.b += learning_rate * (expected - predicted)

    print('model trained!')
    print('testing-->')
    for i, j in zip(x, y):
        predicted = net.forward(i)[0]
        print(i, ' = ', predicted, end = ' ')
        print('(correct)' if predicted == j else print('(wrong)'))
```

In [5]: nn(gates['nand'])

```
model trained!  
testing-->  
(0, 0) = 1 (correct)  
(0, 1) = 1 (correct)  
(1, 0) = 1 (correct)  
(1, 1) = 0 (correct)
```

In []: