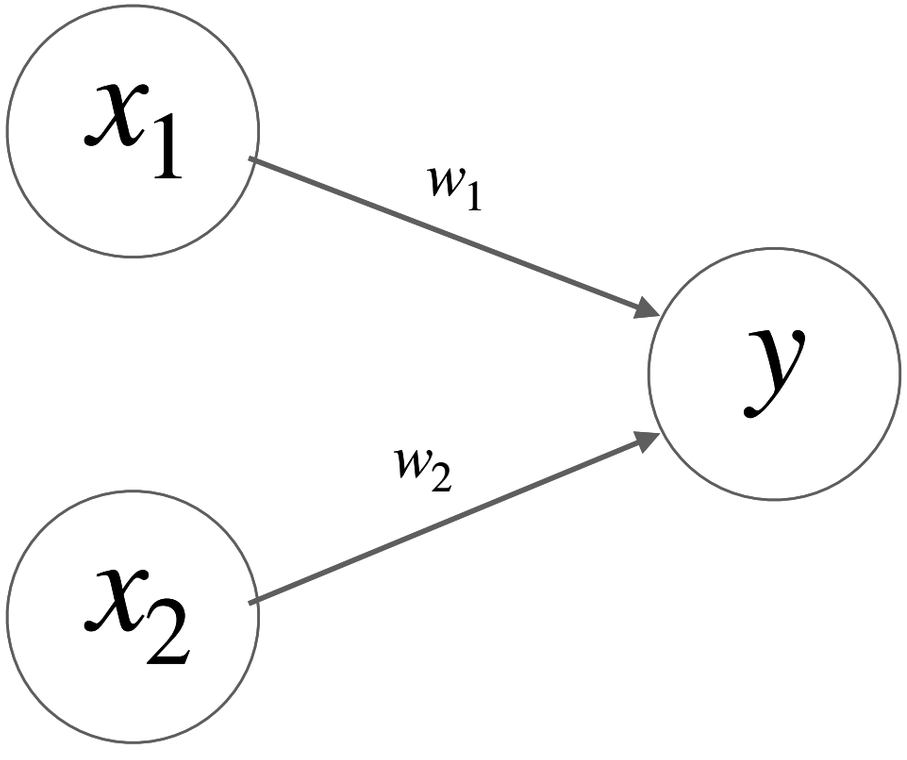
ゼロから作るDeep learning
第二章　パーセプトロン

s1240094 Miyuka Nakamura

目次

2.1 パーセプトロンとは

複数の信号を入力(x)と重み(w)の積和が閾値θより大きい時”1”、そうでない時 に”0”を出力する



$$y = \begin{cases} 1 & (\sum_{i=1}^{n} x_i w_i > \theta) \\ 0 & (otherwise) \end{cases}$$

2.2 単純な論理ゲート

ANDゲートをパーセプトロンで表現する

x_1	x_2	y
0	0	0
1	0	0
0	1	0
1	1	1

真理値表を満たす(w1,w2,θ)を考える

例えば (w1,w2,θ) = (0.5, 0.5 0.7)

NAND, ORゲートについて考える

x_1	x_2	y
0	0	1
1	0	1
0	1	1
1	1	0

- NANDゲート

ANDのパラメータを反転する

例）

(x1,x2, θ) = (-0.5, -0.5, -0.7)

x_1	x_2	y
0	0	0
1	0	1
0	1	1
1	1	1

- ORゲート

例）

(x1,x2, θ) = (1, 1, 0.5)

2.3 パーセプトロンの実装

ANDゲートの実装　(w1,w2,θ) = (0.5, 0.5 0.7)の時

```
def AND(x1, x2):
    w1, w2, theta = 0.5, 0.5, 0.7
    tmp = x1*w1 + x2*w2
    if tmp <= theta:
        return 0
    elif tmp > theta:
        return 1

if __name__ == '__main__':
    for xs in [(0, 0),(1, 0),(0, 1),(1, 1)]:
        y = AND(xs[0], xs[1])
        print(str(xs) + " -> " + str(y))
```

実行結果

(0,0) -> 0

(1,0) -> 0

(0,1) -> 0

(1,1) -> 1

重みとバイアスを導入した実装

```
import numpy as np

def AND(x1, x2):
    x = np.array([x1, x2])
    w = np.array([0.5, 0.5])
    b = -0.7
    tmp = np.sum(w*x) + b
    if tmp <= 0:
        return 0
    else:
        return 1

if __name__ == '__main__':
    for xs in [(0, 0),(1, 0),(0, 1),(1, 1)]:
        y = AND(xs[0], xs[1])
        print(str(xs) + " -> " + str(y))
```

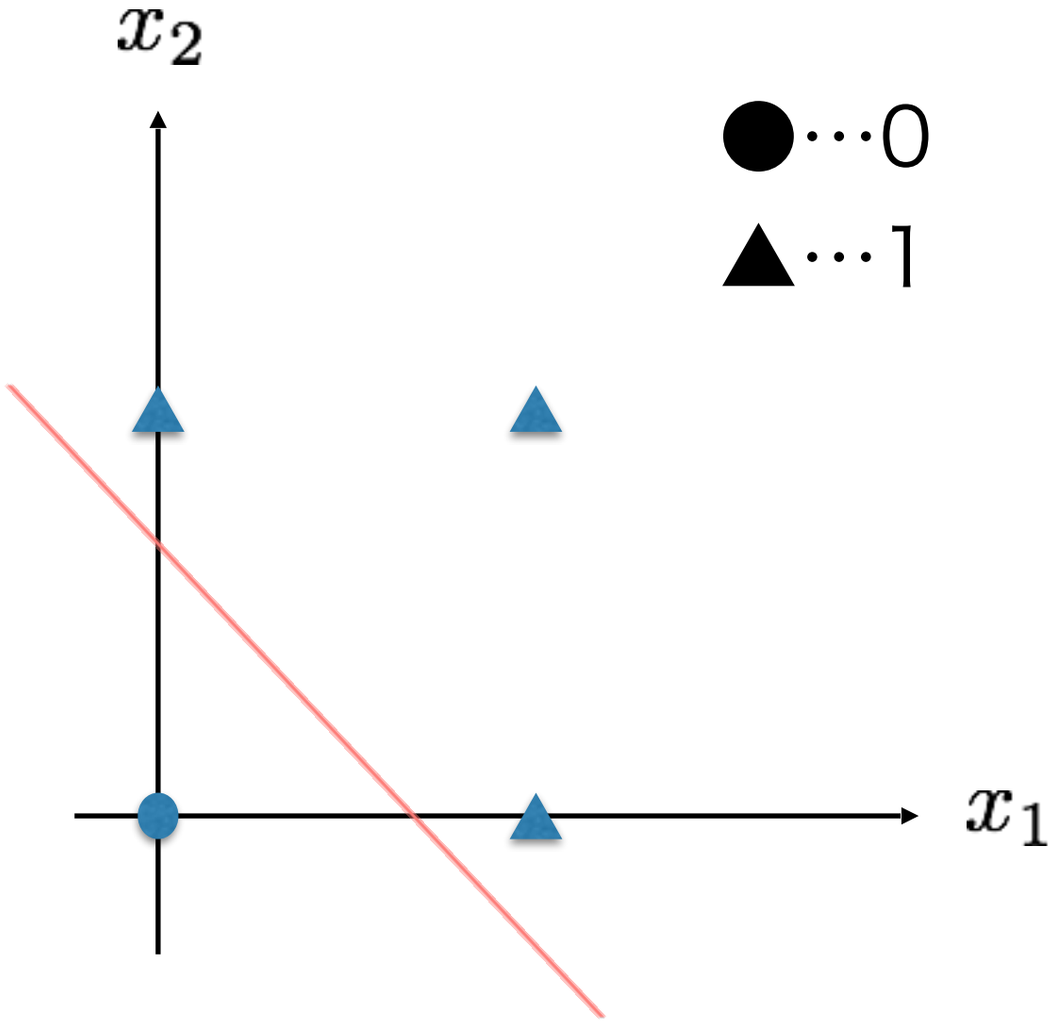
2.4 パーセプトロンの限界

XORを考える

x_1	x_2	y
0	0	0
1	0	1
0	1	1
1	1	0

今までの（単層）パーセプトロンでは実現できない

ORゲートの挙動を視覚的に考える



パラメータは

(b, w1, w2)=(-0.5, 1.0, 1.0)

$$y = \begin{cases} 0 & (-0.5 + x_1 + x_2 \leqq 0) \\ 1 & (-0.5 + x_1 + x_2 > 0) \end{cases}$$

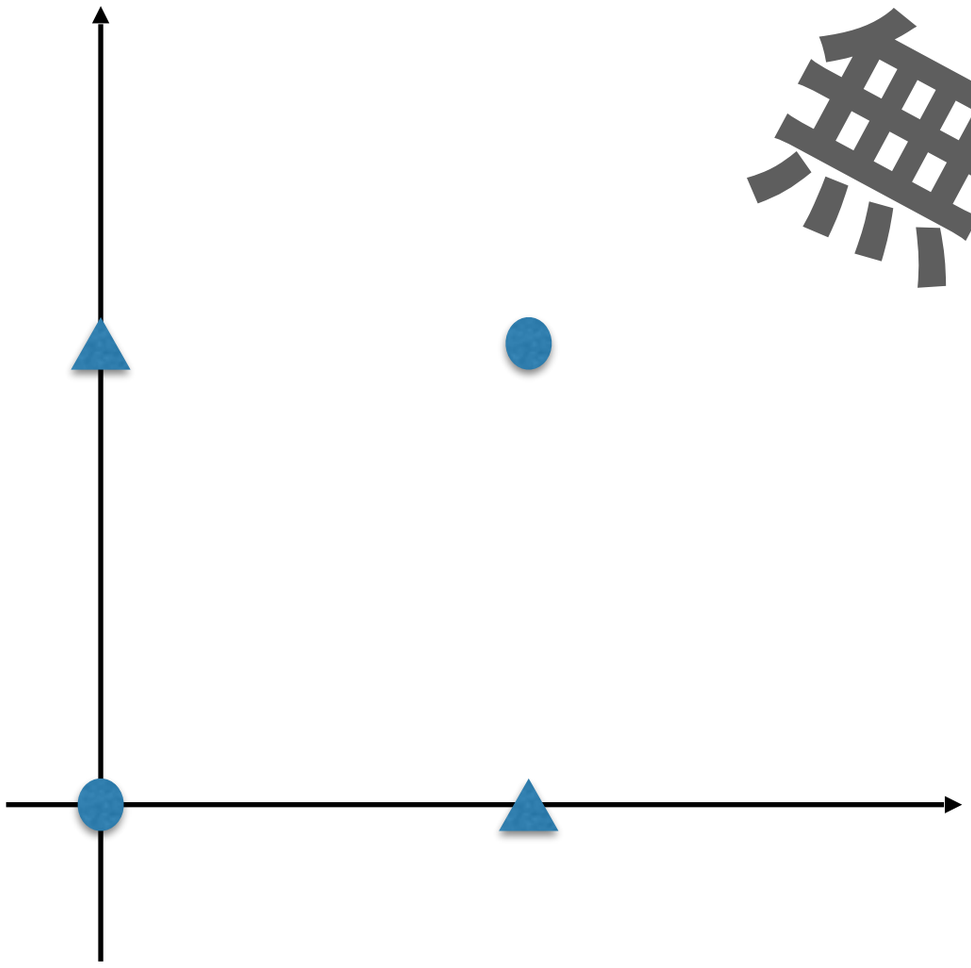
上記で表されるパーセプトロンは

$$-0.5 + x_1 + x_2 = 0$$

で分断された２つの領域を得る。

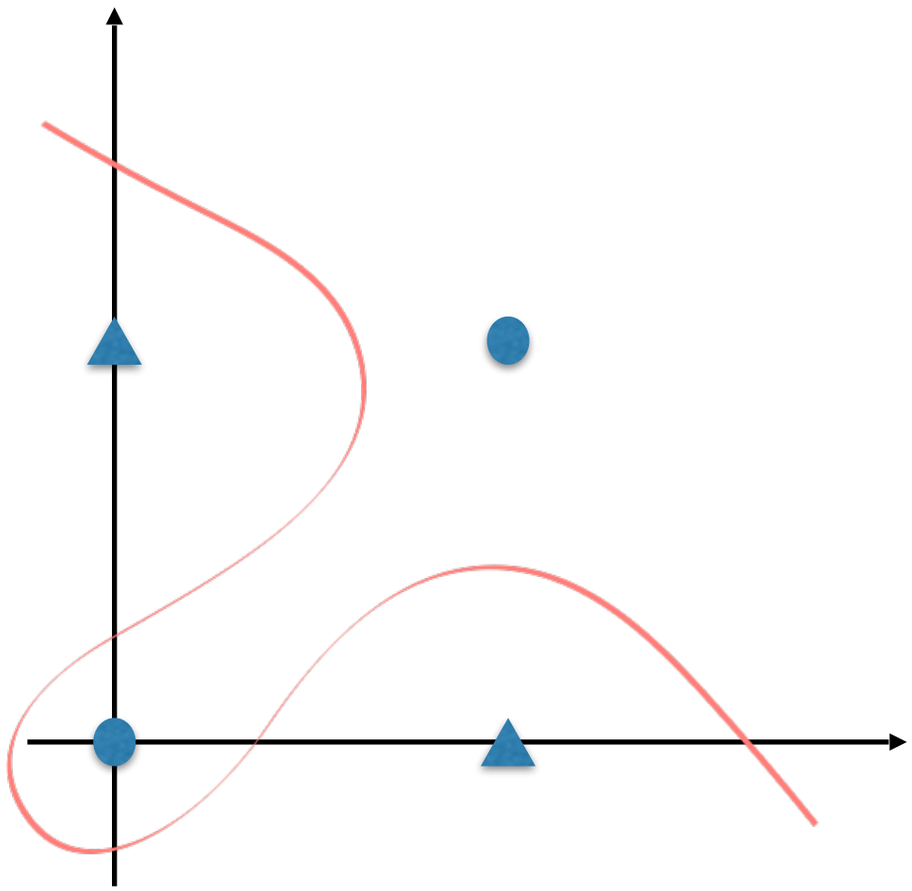
XORは？

XORゲートを表すために一本の直線で○と△を分けられるか？



線形と非線形

もし直線という制約を外すことができたら



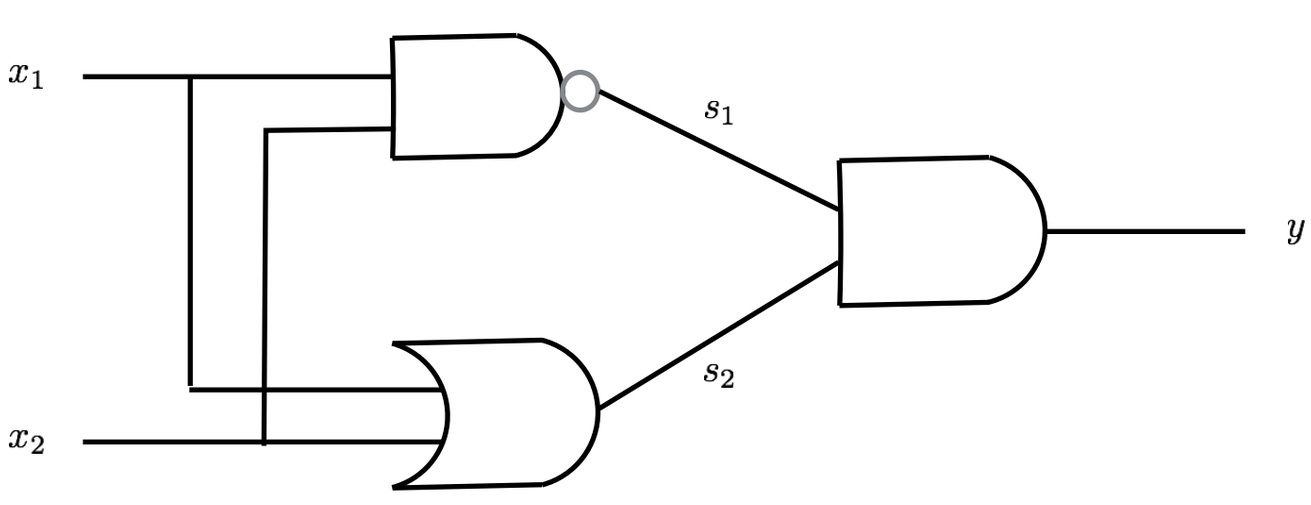
曲線による領域を非線形な領域

直線による領域を線形な領域

という

2.5 多層パーセプトロン

AND, NAND, ORゲートを組み合わせてXORを考える



x_1	x_2	s_1	s_2	y
0	0	1	0	0
1	0	1	1	1
0	1	1	1	1
1	1	0	1	0

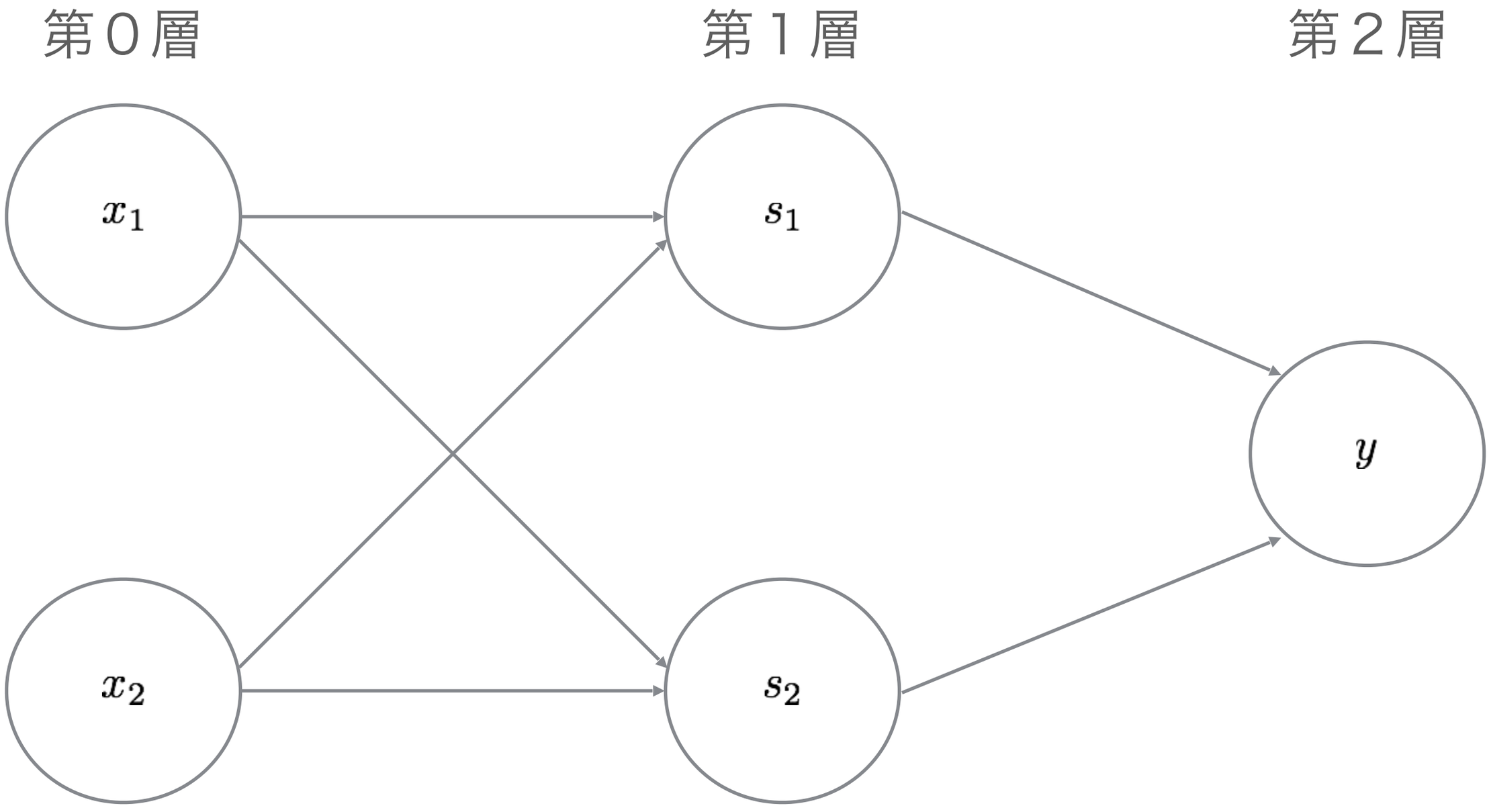
XORの実装

```
from AND_b import AND
from OR import OR
from NAND import NAND

def XOR(x1, x2):
    s1 = NAND(x1, x2)
    s2 = OR(x1, x2)
    y = AND(s1, s2)
    return y

if __name__ == '__main__':
    for xs in [(0, 0),(1, 0),(0, 1),(1, 1)]:
        y = XOR(xs[0], xs[1])
        print(str(xs) + " -> " + str(y))
```

XORのパーセプトロンによる表記



多層パーセプトロン

2.6 NANDからコンピュータへ

- パーセプトロンを多層にすることで、より複雑な回路も表現できる

- コンピュータはNANDの組み合わせだけで表現できる

- パーセプトロンでもコンピュータが行う処理も表現できる

2.7 まとめ

- パーセプトロンとは
    - 入出力を備えたアルゴリズム
    - 「重み」と「バイアス」をパラメータとして設定する
    - ANDやORゲートなどの論理回路を表現できる
    - 多層にすることでXOR等も表現できる
- 単層パーセプトロンでは非線形領域を表現できない
- 多層のパーセプトロンは（理論上）コンピュータを表現できる