

演習：1. 入力層～中間層

入力層では、入力値に重みをかけた値にバイアスを足し合わせた合計値を総入力とし、中間層に渡す。

importと関数定義

```
In [1]: import numpy as np
        from common import functions
```

```
In [2]: def print_vec(text, vec):
        print(f'{text} : \n{vec}')
```

順伝播（単層・単ユニット）

```
In [3]: # 重み
        W = np.array([[0.1], [0.2]])
        print_vec("重み", W)
```

```
重み :
[[0.1]
 [0.2]]
```

```
In [4]: # バイアス
        b = 0.5
        print_vec("バイアス", b)
```

```
バイアス :
0.5
```

```
In [5]: # 入力値
        x = np.array([2, 3])
        print_vec("入力", x)
```

```
入力 :
[2 3]
```

```
In [7]: # 総入力
        u = np.dot(x, W) + b
        print_vec("総入力", u)
```

```
総入力 :
[1.3]
```

```
In [8]: # 中間層出力
        z = functions.relu(u)
        print_vec("中間層出力", z)
```

```
中間層出力 :
[1.3]
```

順伝播（単層・複数ユニット）

```
In [9]: # 重み
W = np.array([
    [0.1, 0.2, 0.3],
    [0.2, 0.3, 0.4],
    [0.3, 0.4, 0.5],
    [0.4, 0.5, 0.6]
])
```

```
In [10]: # バイアス
b = np.array([0.1, 0.2, 0.3])
```

```
In [11]: # 入力値
x = np.array([1.0, 5.0, 2.0, -1.0])
```

```
In [12]: # 総入力
u = np.dot(x, W) + b
print_vec("総入力", u)
```

総入力 :
[1.4 2.2 3.]

```
In [13]: # 中間層出力 (シグモイド関数)
z = functions.sigmoid(u)
print_vec("中間層出力", z)
```

中間層出力 :
[0.80218389 0.90024951 0.95257413]

```
In [ ]:
```