

【要点まとめ】

関数についての総復習を講義いただいた。

関数とは、与えられた入力値に対して何らかの処理を行い、新たな出力値を生成・返却するものである。

関数は $f(h, g)$ が後続) で表現され、例えば入力値を2.5倍して5を足す関数は

$$f(x) = 2.5x + 5$$

と表現される。(実装は実装演習に記載する)

活性化関数は以下のようなものがある。(メジャーなものを抜粋)

【中間層で使用する活性化関数】

- ステップ関数
- シグモイド関数
- tanh
- ReLU
- Leaky ReLU
- Swish

【出力層で使用する活性化関数】

- シグモイド関数：2値分類
- ソフトマックス関数：多値分類
- 恒等関数：回帰 (何もしないので活性化関数ではない)

【実装演習】

pythonで $f(x) = 2.5x + 5$ を表現

```
def f(x):  
    return 2.5x + 5
```

シグモイド関数

```
def sigmoid(x):  
    return 1.0 / (1.0 + np.exp(-x))
```

ReLU関数

```
def relu(x):  
    return np.maximum(0, x)
```

【実装演習考察】

シグモイド関数はグラフで表現すると複雑に見えますが、実装では一行で書くことができました。(np.expの力かも

しませんが)

活性化関数はとても種類豊富に見えるので、講義いただいた基本的な分だけでも実装まで抑えておく必要があると感じました。また恒等関数（恒等写像）のような何もしない関数は何を意味するのかなど、キチンと理解しておく必要があります。

【自己学習】

※参考資料：ゼロから作るDeepLearning①

恒等関数（恒等写像）は回帰問題の出力層で使用されることがある。

出力層の出力値の値が、そのまま確率を意味することができる場合など、値を変化させても最終的な出力値の比率が変わらない（意味を成さない場合）に使用される。