

# day 1 Section 3 : 出力層

Q) 誤差関数 (二乗和誤差) において

- ・なぜ、引き算ではなく二乗するのか
- ・下式の  $1/2$  はどついつ意味をもつか述べよ。

A) ・誤差には正負があるため単純に足すと相殺されて全体の誤差を正しく表せない。  
そのため、二乗して各々のレベルでの誤差を正の値になるようにする。

- ・誤差逆伝搬の計算で、誤差関数の微分を用いるが、その際の計算式を  $2 \times \frac{1}{2} = 1$  を利用して楽にするため。

Q) ①~③の式に該当するノードを示し、一行づつ処理の説明をせよ。①  $f(i, u)$

A) `def softmax(x):`  
`if x.ndim == 2:`  
`x = x.T`  
`x = x - np.max(x, axis=0)`  
`y = np.exp(x) / np.sum(np.exp(x), axis=0)`  
`return y.T`  
`x = x - np.max` # オバ-フロー対策 安定化  
`return np.exp(x) / np.sum(np.exp(x))`

★本題的な部分  
 はこの箇所  
 あり、②と、  
 ③と、  
 ①として返却する。

①  $f(i, u)$   
 ②  $e^{v_i}$   
 ③  $\sum_{k=1}^K e^{v_k}$

Q) ①~②の式に該当するソースコードをえし。  
- 実行処理の説明をせよ。

①  $E_n(W) = - \sum_{i=1}^I d_i \log y_i$  ② 交差エントロピー

A) # クロスエントロピー  
def cross\_entropy\_error(d, y):

```
    if y.ndim == 1:  
        d = d.reshape(1, d.size)  
        y = y.reshape(1, y.size)
```

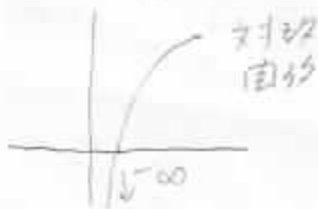
# 教師データが one-hot-vector の場合. 正解ラベルのインデックスにマッピング

```
    if d.size == y.size:  
        d = d.argmax(axis=1)
```

```
    batch_size = y.shape[0]
```

```
    return (-np.sum(np.log(y[np.arange(batch_size), d]  
                        + 1e-7))) / batch_size
```

★ 本質的な部分はこの箇所であり①を求めている。



•  $+1e-7$  している理由は、対数関数は 0 に近づくと  $-\infty$  になるが、それを回避するため

•  $y$  には 0 が 1 が含まれている。(正解の部分は 1)

•  $d$  が NN が正解だと思った場所