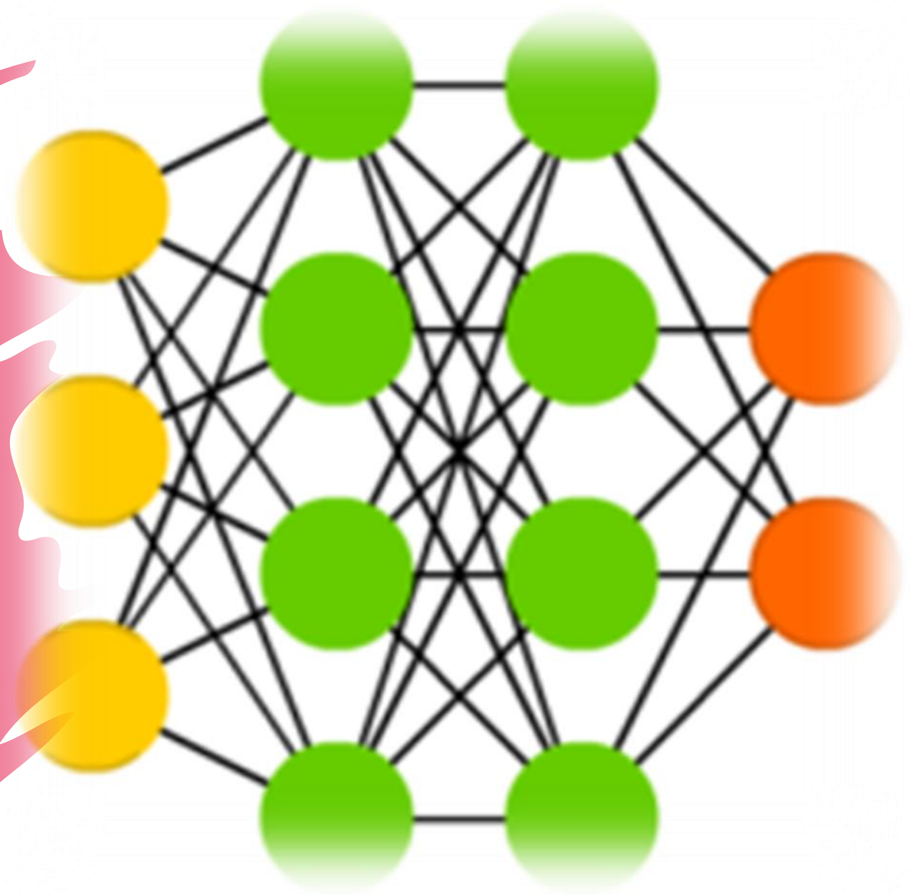
手計算
ニューラルネットワーク
入門　③実装1



予定

第１回　理論：順伝播

第２回　理論：逆伝播

第３回　実装：実装1

第４回　実験：実装2iris,titanic

第５回　実験：実装3mnist

ニューラルネットワーク
を完全に理解したい

最終目標（最低限）

☑ ニューラルネットワークを完全に理解する

C++でニューラルネットワークを実装し
何かしらの分類問題を解く

C++でニューラルネットワークを実装し
何かしらの回帰問題を解く

最終目標（理想）

Kaggle Titanic in C++

k KAGGLE · GETTING STARTED PREDICTION COMPETITION · ONGOING

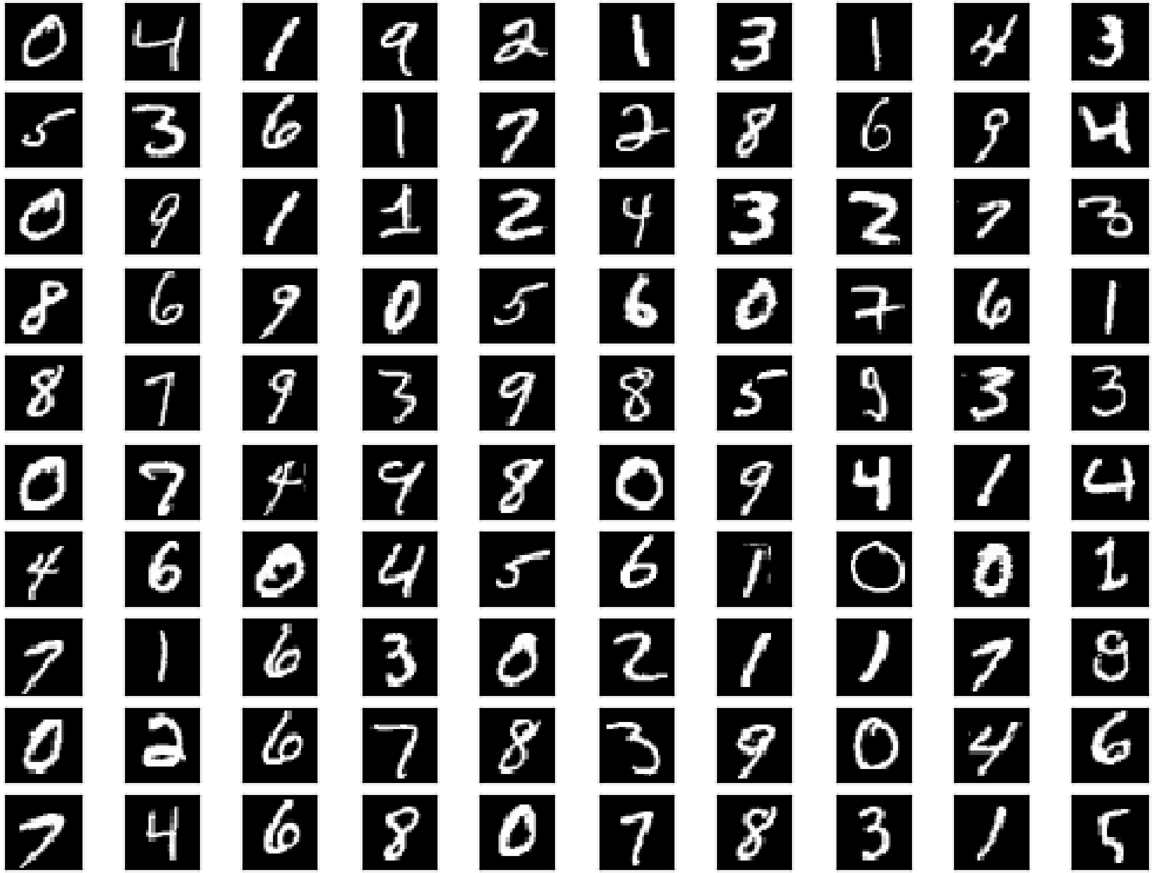
Submit Prediction

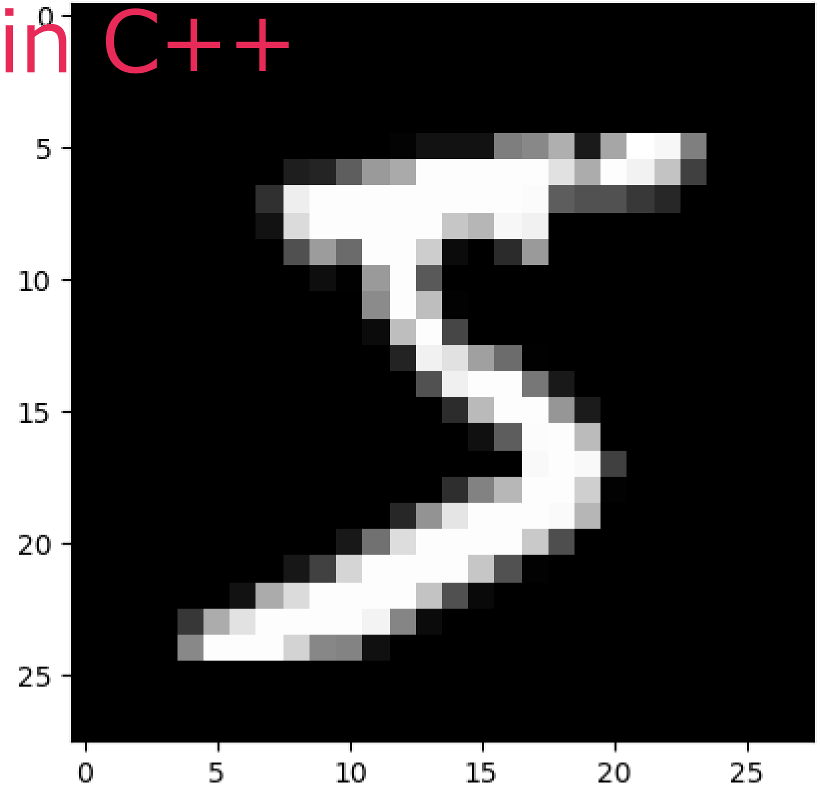
Titanic - Machine Learning from Disaster

Start here! Predict survival on the Titanic and get familiar with ML basics

最終目標（理想）

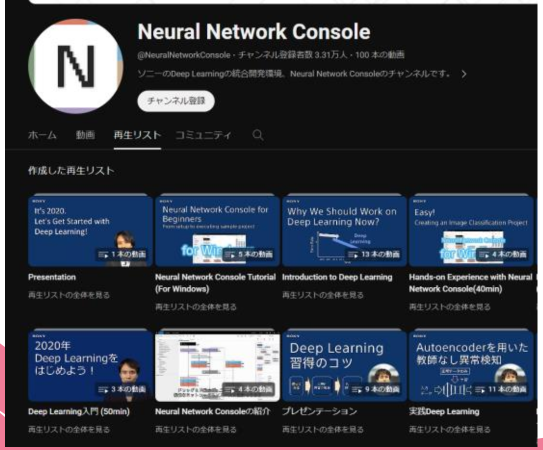
MNISTデータセットの手書き文字認識 in C++

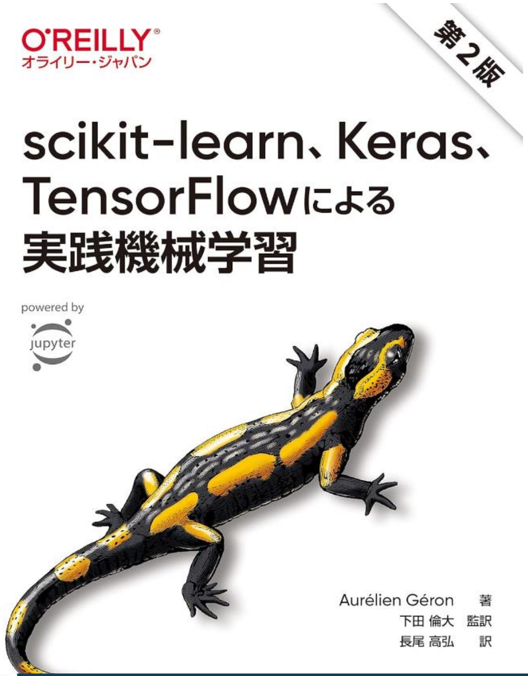


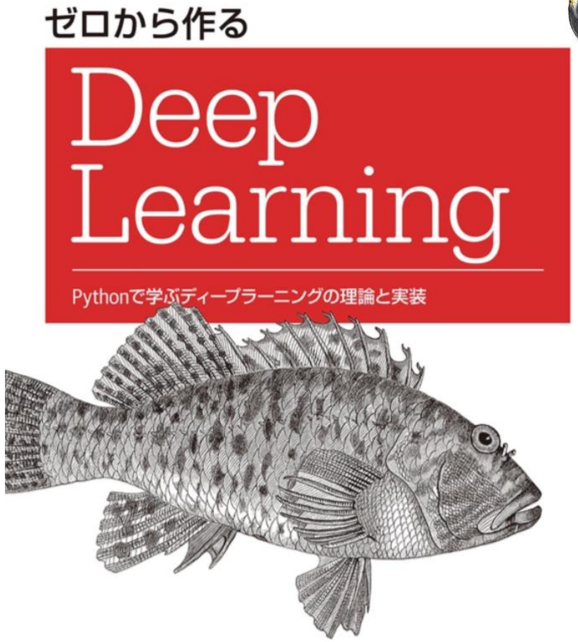


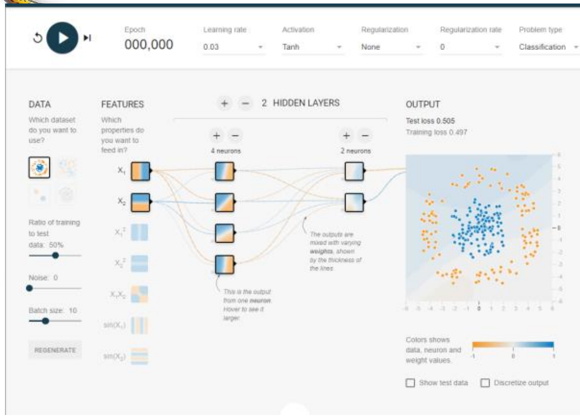
28×28×256

参考文献

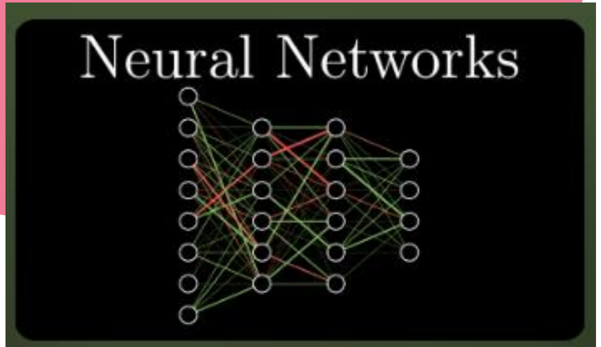








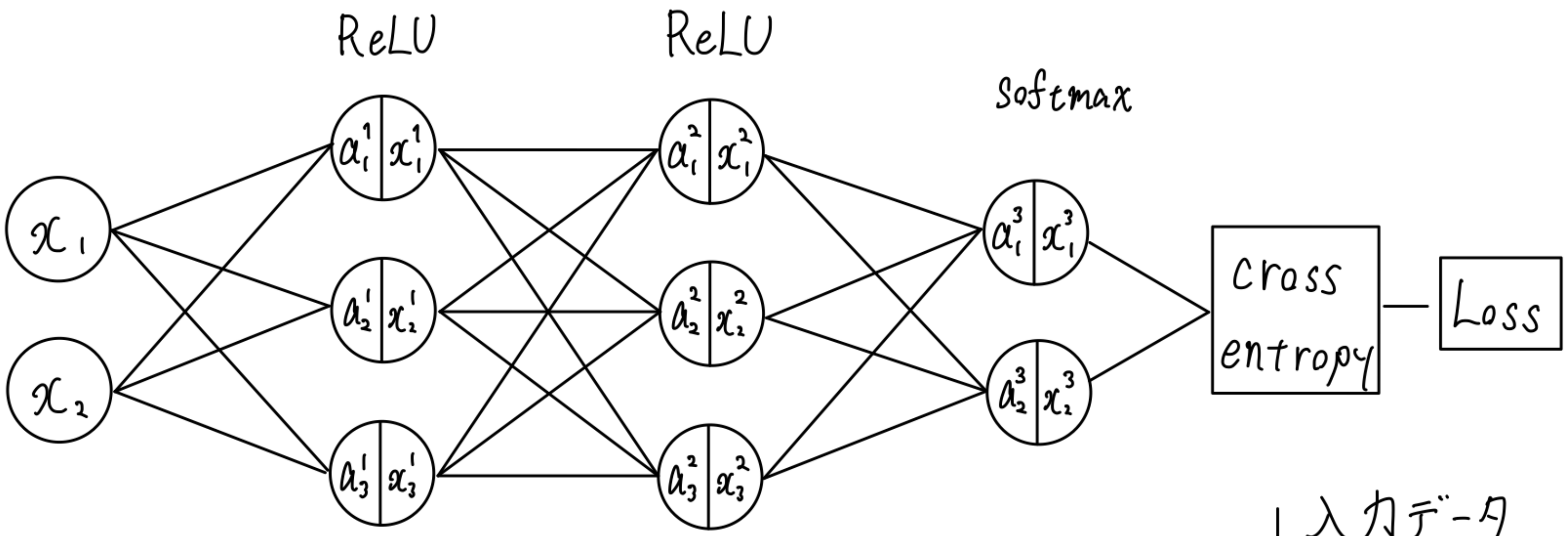
ディープラーニング入門：仕組み理解から実装まで
某処生活_LiveSomewhere
5 本の動画 996 回視聴 最終更新日: 2021/02/28



Neural networks
3Blue1Brown・コース
4 本の動画 本日更新

前回の復習と訂正

誤差逆伝播-演習③手計算でNNを学習せよ



左のNNを学習せよ

＝すべてのパラメータを1回更新せよ

Lossはいくら改善するか？

$$W^1 = \begin{pmatrix} 1 & -2 & 3 \\ -1 & 2 & -3 \end{pmatrix},\ W^2 = \begin{pmatrix} -1 & 2 & -3 \\ 1 & -2 & 3 \\ -1 & 2 & -3 \end{pmatrix},\ W^3 = \begin{pmatrix} -1 & 2 \\ -3 & 1 \\ -2 & 3 \end{pmatrix}$$

$$B^1 = \begin{pmatrix} 1 & -2 & 3 \end{pmatrix},\ B^2 = \begin{pmatrix} 1 & -2 & 3 \end{pmatrix},\ B^3 = \begin{pmatrix} 34 & -54 \end{pmatrix}$$

入力データ

$X = \begin{pmatrix} -1 & 2 \end{pmatrix}$ 1×2

教師ラベル

$T = \begin{pmatrix} 1 & 0 \end{pmatrix}$

学習率

$\eta = 0.1$

誤差逆伝播-公式 まとめ 行列表現

出力層

$$\frac{\partial L}{\partial W^K} = X^{(K-1)T} \Delta^K$$

$$\frac{\partial L}{\partial B^K} = \Delta^K$$

$$\Delta^K = \frac{\partial f_L}{\partial X^K} \odot \frac{\partial h_K}{\partial A^K}$$

ここが間違ってた

中間層

$$\frac{\partial L}{\partial W^l} = X^{(l-1)T} \Delta^l$$

$$\frac{\partial L}{\partial B^l} = \Delta^l$$

$$\Delta^l = \Delta^{l+1} W^{(l+1)T} \odot \frac{\partial h_l}{\partial A^l}$$

誤差逆伝播-公式 まとめ 誤差/活性化関数の微分

誤差関数(cross-entropy)

$$\frac{\partial f_L}{\partial {x_j}^K} = \left(-\frac{t_j}{{x_j}^K} + \sum_{i=1 \wedge i \neq j}^{n} \frac{t_i}{{x_i}^K} \right)$$

活性化関数(ReLU)

$$\frac{\partial h_l}{\partial {a_j}^l} = \begin{cases} 1({a_j}^l > 0) \\ 0({a_j}^l \leq 0) \end{cases}$$

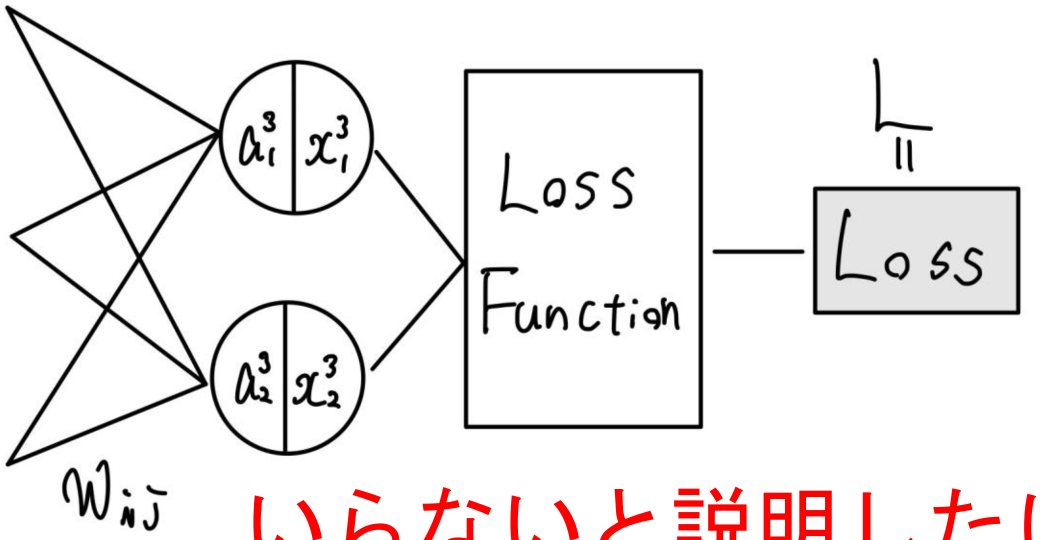
活性化関数(soft-max)

$$\frac{\partial h_K}{\partial {a_j}^K} = {x_j}^K\left(1 - {x_j}^K\right)$$

誤差逆伝播-出力層の重みの更新

$$\frac{\partial L}{\partial w_{ij}{}^K} = \frac{\partial L}{\partial x_j{}^K}\frac{\partial x_j{}^K}{\partial a_j{}^K}\frac{\partial a_j{}^K}{\partial w_{ij}{}^K}$$

第2回のこの部分
から
間違ってた

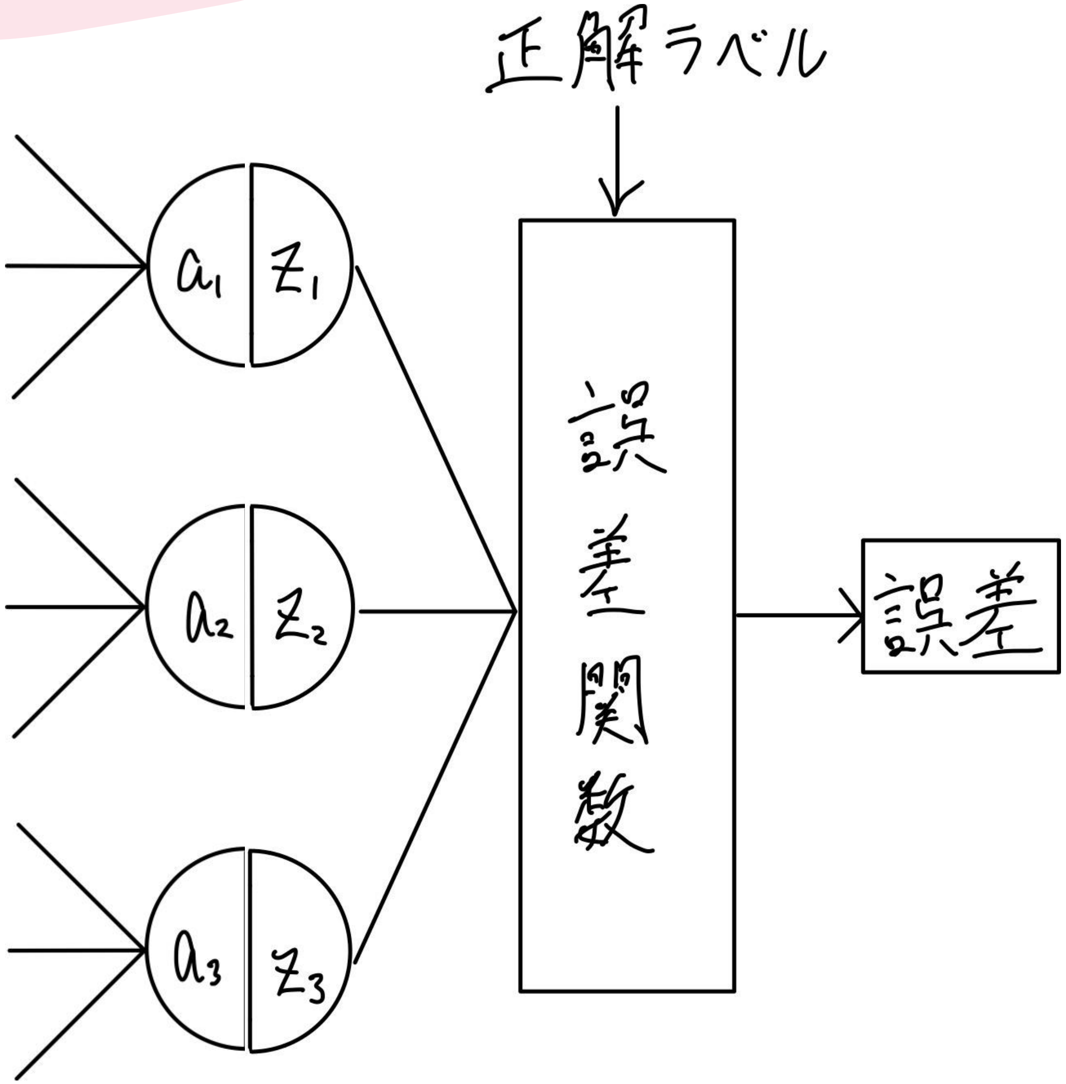


さっきのシグマは？

$$\frac{\partial z}{\partial x_k} = \sum_{i=1}^{n} \frac{\partial z}{\partial u_i}\frac{\partial u_i}{\partial x_k}$$

いらないと説明したけど→
出力層の活性化関数が
softmaxのようなとき
(1入力1出力関数でないとき)は必要

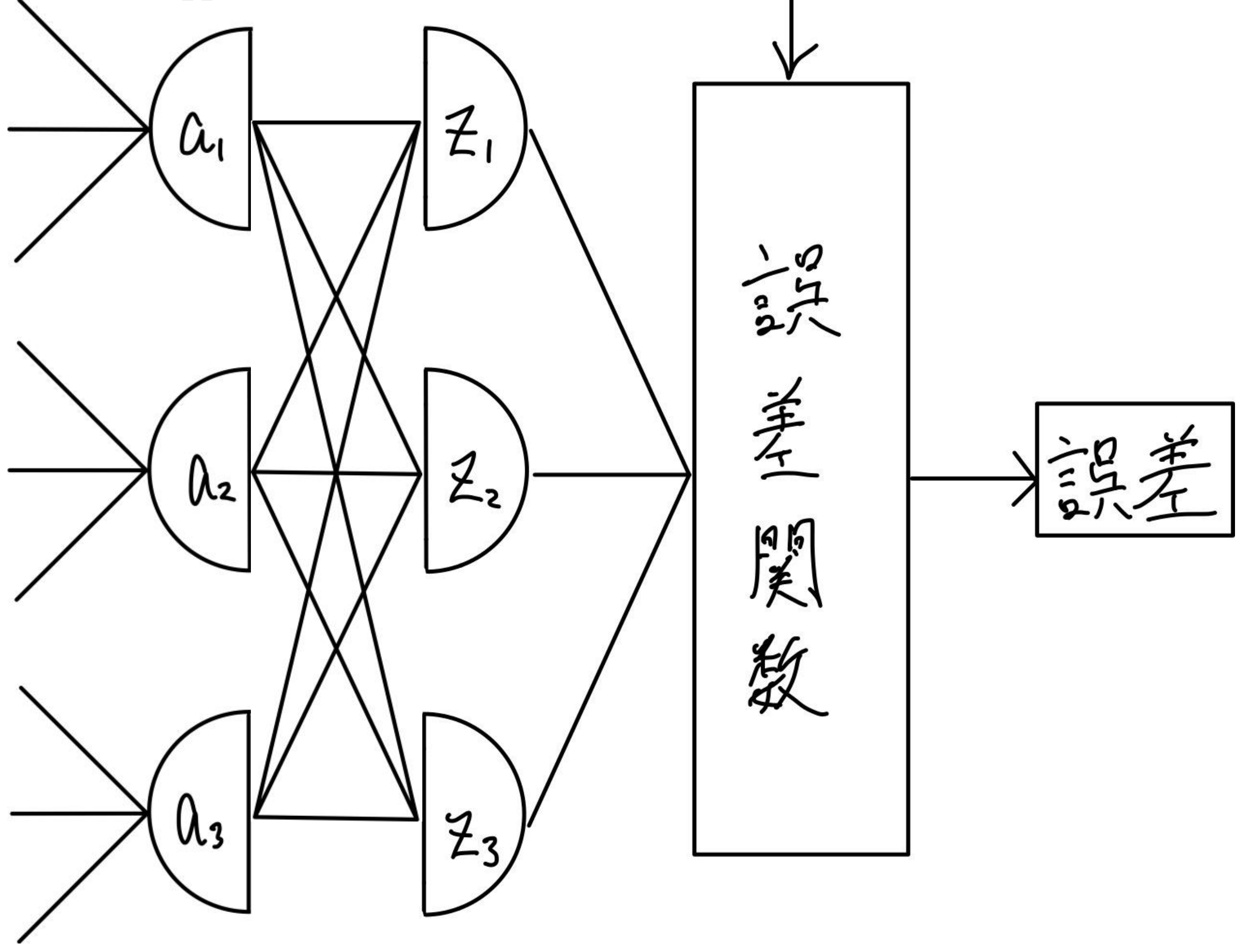
ニューラルネットワーク-softmax



$$z_k = \frac{e^{a_i}}{\sum_{i=1}^{n} e^{a_i}}$$

ニューラルネットワーク-softmax

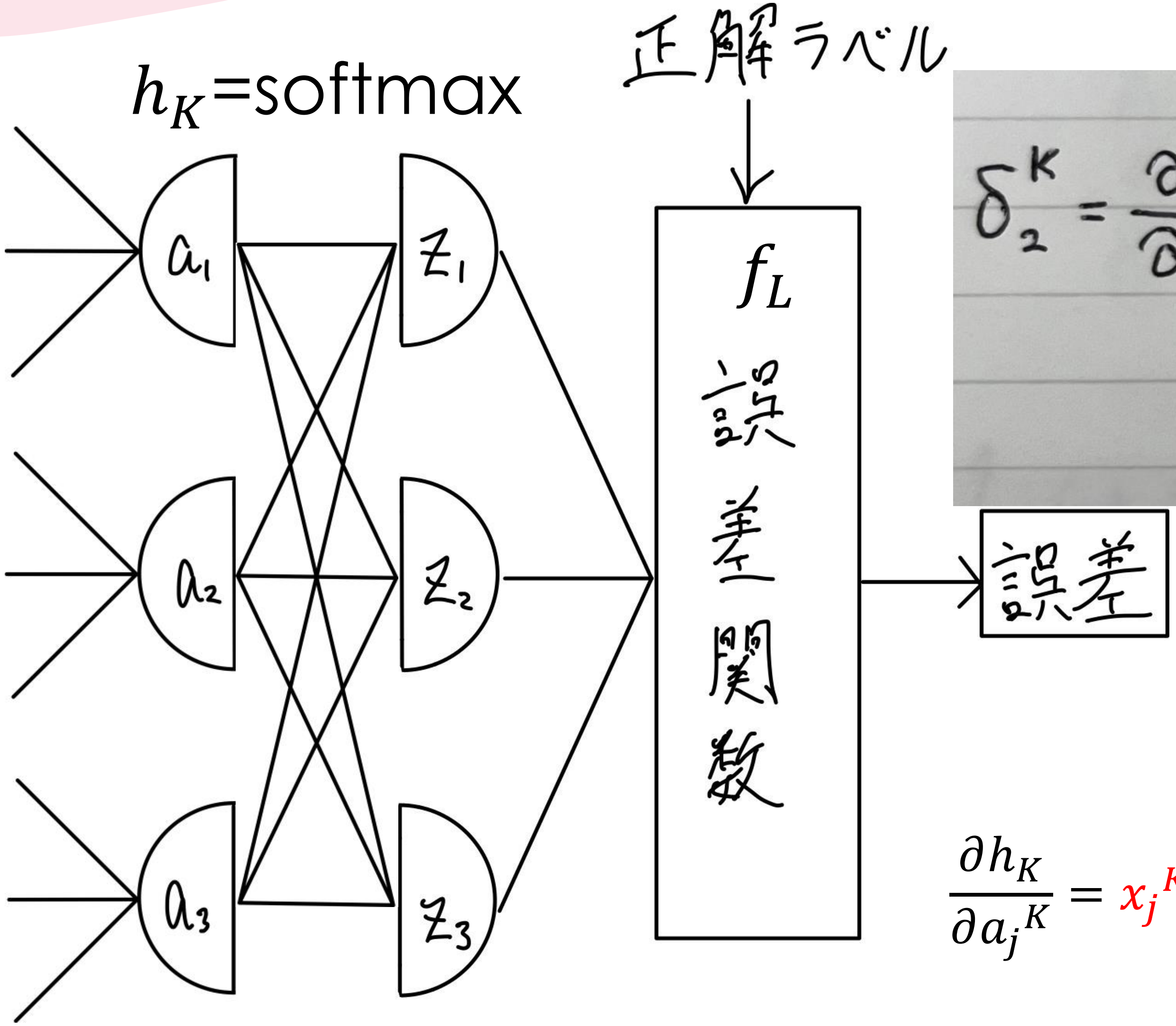
h_K=softmax



$$z_k = \frac{e^{a_i}}{\sum_{i=1}^{n} e^{a_i}}$$

ニューラルネットワーク-softmaxの誤差逆伝播

h_K=softmax



$$\delta_2^K = \frac{\partial L}{\partial a_2^K} = \frac{\partial L}{\partial x_1^K}\frac{\partial x_1^K}{\partial a_2^K} + \frac{\partial L}{\partial x_2^K}\frac{\partial x_2^K}{\partial a_2^K} + \frac{\partial L}{\partial x_3^K}\frac{\partial x_3^K}{\partial a_2^K}$$

$$= \sum_i \frac{\partial L}{\partial x_i^K}\frac{\partial x_i^K}{\partial a_2^K}$$

$$\frac{\partial h_K}{\partial a_j{}^K} = x_j{}^K\left(1 - x_j{}^K\right)$$

ニューラルネットワーク-softmaxの誤差逆伝播

$$\delta_2^K = \frac{\partial L}{\partial a_2^K} = \frac{\partial L}{\partial x_1^K}\frac{\partial x_1^K}{\partial a_2^K} + \frac{\partial L}{\partial x_2^K}\frac{\partial x_2^K}{\partial a_2^K} + \frac{\partial L}{\partial x_3^K}\frac{\partial x_3^K}{\partial a_2^K}$$

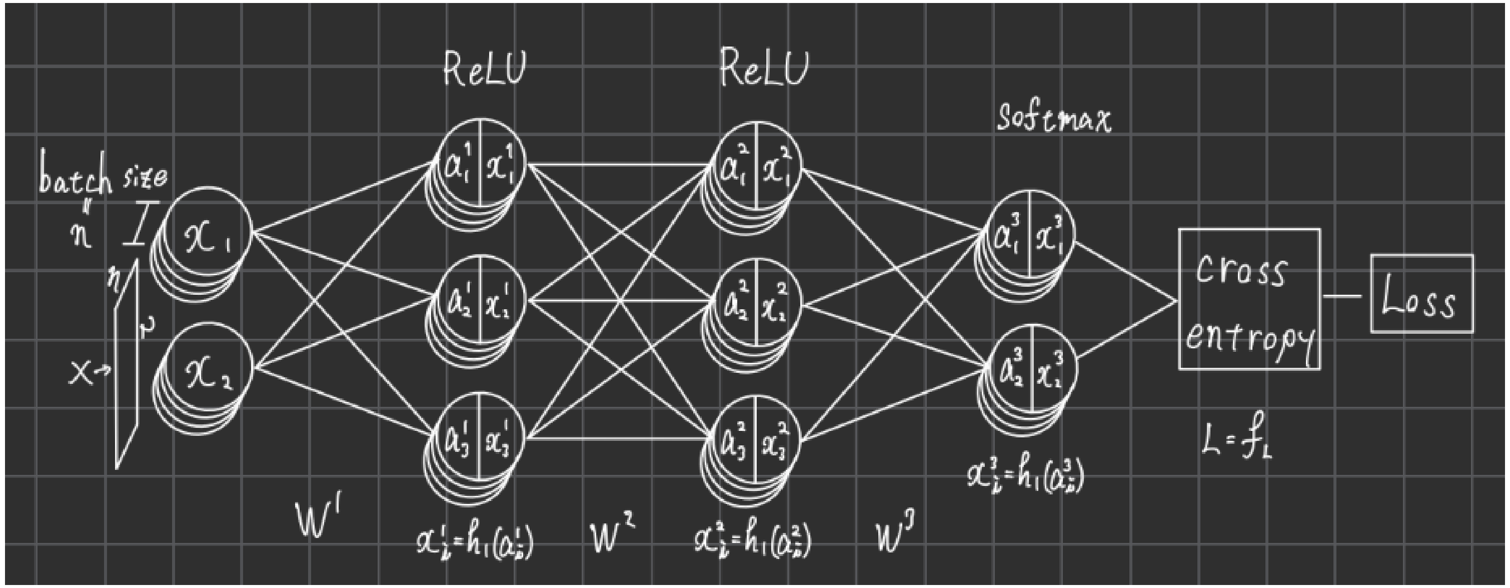
$$= \sum_i \frac{\partial L}{\partial x_i^K}\frac{\partial x_i^K}{\partial a_2^K}$$

$$\Delta^K = \begin{pmatrix} {\delta_1}^K & {\delta_2}^K & {\delta_3}^K \end{pmatrix}$$

$$= \begin{pmatrix} \sum_i \frac{\partial L}{\partial {x_i}^K}\frac{\partial {x_i}^K}{\partial {a_1}^K} & \sum_i \frac{\partial L}{\partial {x_i}^K}\frac{\partial {x_i}^K}{\partial {a_2}^K} & \sum_i \frac{\partial L}{\partial {x_i}^K}\frac{\partial {x_i}^K}{\partial {a_3}^K} \end{pmatrix}$$

誤差逆伝播法
ミニバッチ学習

ミニバッチ学習の順伝播

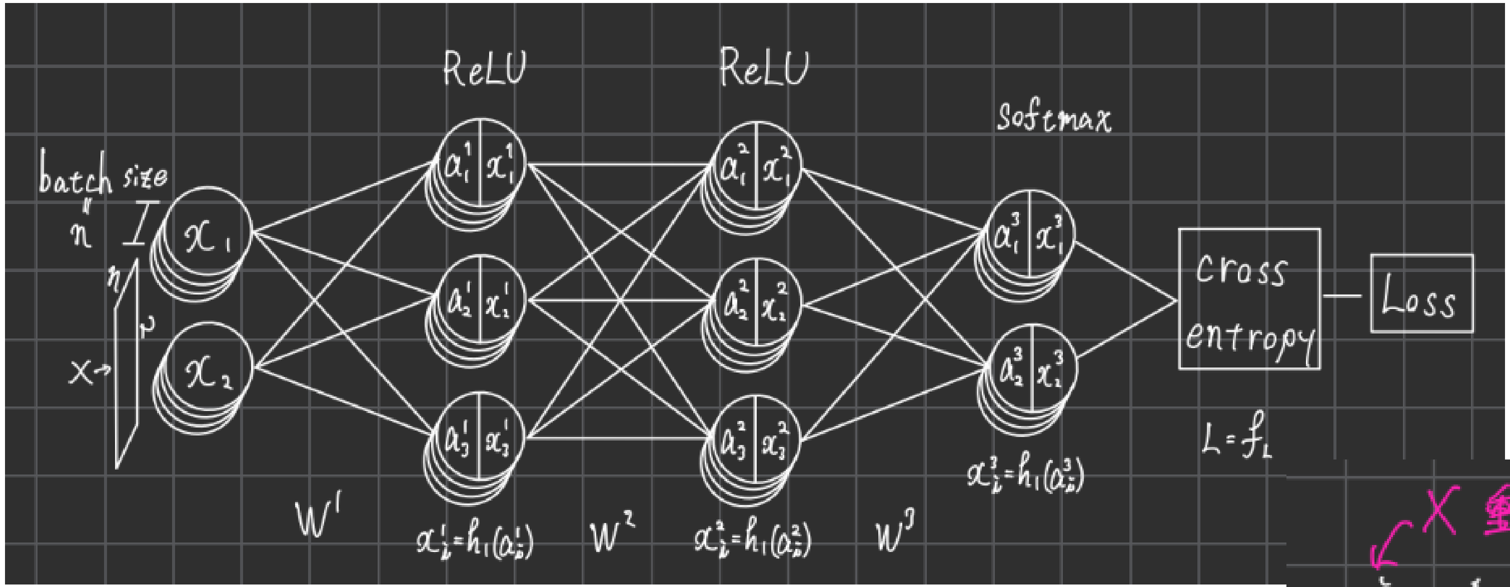


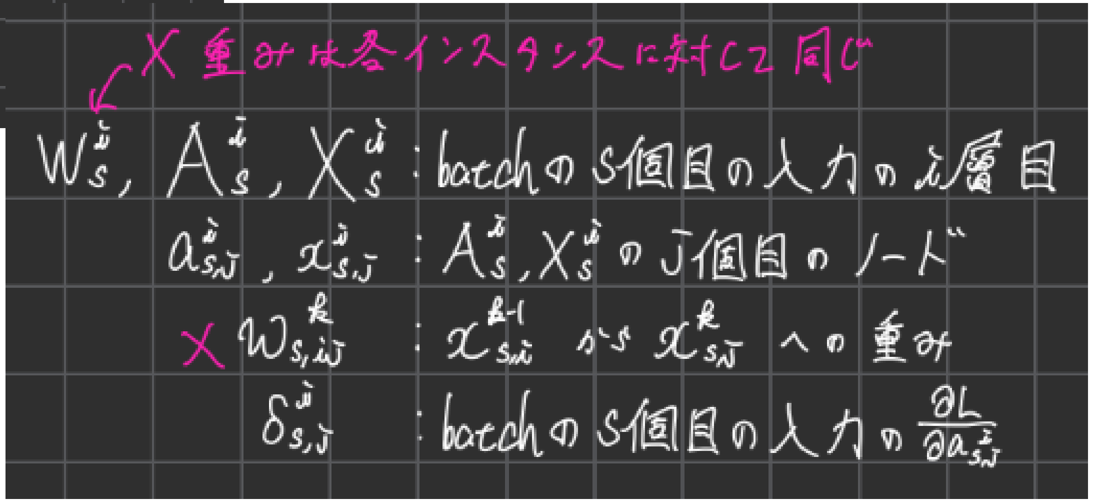
入力 X^0 $(n\times 2)$

$$A^1 = X^0W^1 + B^1$$
$$X^1 = \mathrm{ReLU}(A^1)$$
$$A^2 = X^1W^2 + B^2$$
$$X^2 = \mathrm{ReLU}(A^2)$$
$$A^3 = X^2W^3 + B^3$$
$$X^3 = \mathrm{SoftMax}(A^3)$$
$$L = \mathrm{CrossEntropy}(X^3)$$

バイアスの行列(例えばB_1)は
n×2になるよう1行目をコピーする

ミニバッチ学習の順伝播





パラメータ(W,B)は
任意のインスタンスに対して
同じである（それはそう）

ミニバッチ学習の順伝播

$$\Delta^K = \begin{pmatrix} \sum_i \frac{\partial L}{\partial {x_{1,i}}^K} \frac{\partial {x_{1,i}}^K}{\partial {a_{1,1}}^K} & \cdots & \sum_i \frac{\partial L}{\partial {x_{1,i}}^K} \frac{\partial {x_{1,i}}^K}{\partial {a_{1,3}}^K} \\ \vdots & \ddots & \vdots \\ \sum_i \frac{\partial L}{\partial {x_{n,i}}^K} \frac{\partial {x_{n,i}}^K}{\partial {a_{n,1}}^K} & \cdots & \sum_i \frac{\partial L}{\partial {x_{n,i}}^K} \frac{\partial {x_{n,i}}^K}{\partial {a_{n,3}}^K} \end{pmatrix}$$

$$\frac{\partial L}{\partial X^K} = \begin{pmatrix} \frac{\partial L}{\partial {x_{1,1}}^K} & \cdots & \frac{\partial L}{\partial {x_{1,3}}^K} \\ \vdots & \ddots & \vdots \\ \frac{\partial L}{\partial {x_{n,1}}^K} & \cdots & \frac{\partial L}{\partial {x_{n,3}}^K} \end{pmatrix}$$

$$\frac{\partial h_K}{\partial A^K} = \begin{pmatrix} \frac{\partial {x_{1,1}}^K}{\partial {a_{1,1}}^K} & \cdots & \frac{\partial {x_{1,1}}^K}{\partial {a_{1,3}}^K} \\ \vdots & \ddots & \vdots \\ \frac{\partial {x_{1,3}}^K}{\partial {a_{1,1}}^K} & \cdots & \frac{\partial {x_{1,3}}^K}{\partial {a_{1,3}}^K} \end{pmatrix} \cdots$$

ミニバッチ学習の順伝播

$$\Delta^K = \begin{pmatrix} \sum_i \frac{\partial L}{\partial {x_{1,i}}^K} \frac{\partial {x_{1,i}}^K}{\partial {a_{1,1}}^K} & \cdots & \sum_i \frac{\partial L}{\partial {x_{1,i}}^K} \frac{\partial {x_{1,i}}^K}{\partial {a_{1,3}}^K} \\ \vdots & \ddots & \vdots \\ \sum_i \frac{\partial L}{\partial {x_{n,i}}^K} \frac{\partial {x_{n,i}}^K}{\partial {a_{n,1}}^K} & \cdots & \sum_i \frac{\partial L}{\partial {x_{n,i}}^K} \frac{\partial {x_{n,i}}^K}{\partial {a_{n,3}}^K} \end{pmatrix}$$

$$\frac{\partial f_L}{\partial X^K} = \frac{\partial L}{\partial X^K} = \begin{pmatrix} \frac{\partial L}{\partial {x_{1,1}}^K} & \cdots & \frac{\partial L}{\partial {x_{1,3}}^K} \\ \vdots & \ddots & \vdots \\ \frac{\partial L}{\partial {x_{n,1}}^K} & \cdots & \frac{\partial L}{\partial {x_{n,3}}^K} \end{pmatrix}$$

$$\frac{\partial h_K}{\partial A^K} = \begin{pmatrix} \frac{\partial {x_{1,1}}^K}{\partial {a_{1,1}}^K} & \cdots & \frac{\partial {x_{1,1}}^K}{\partial {a_{1,3}}^K} \\ \vdots & \ddots & \vdots \\ \frac{\partial {x_{1,3}}^K}{\partial {a_{1,1}}^K} & \cdots & \frac{\partial {x_{1,3}}^K}{\partial {a_{1,3}}^K} \end{pmatrix} \cdots$$

n個

$$\Delta^K = \frac{\partial f_L}{\partial X^K} \odot \frac{\partial h_K}{\partial A^K}$$

誤差逆伝播-公式 まとめ 誤差/活性化関数の微分

誤差関数(cross-entropy)

$$\frac{\partial f_L}{\partial {x_j}^K} = \left(-\frac{t_j}{{x_j}^K} + \sum_{i=1 \land i \neq j}^{n} \frac{t_i}{{x_i}^K} \right)$$

活性化関数(ReLU)

$$\frac{\partial h_l}{\partial {a_j}^l} = \begin{cases} 1({a_j}^l > 0) \\ 0({a_j}^l \leq 0) \end{cases}$$

活性化関数(soft-max)

$$\frac{\partial h_K}{\partial {a_j}^K} = {x_j}^K(1 - {x_j}^K)$$

$\longrightarrow$ $\frac{\partial {x_i}^K}{\partial {a_j}^K}$を求める

誤差逆伝播-公式

活性化関数(soft-max)

$$\frac{\partial {x_i}^K}{\partial {a_j}^K}$$

$$x_k^K = \frac{e^{a_k^K}}{\sum_{i=1}^{n} e^{a_i^K}}$$

$$\frac{\partial x_k^K}{\partial a_j^K} = \frac{e^{a_k^K} \sum_{i=1}^{n} e^{a_i^K} - e^{a_k^K} e^{a_j^K}}{\left(\sum_{i=1}^{n} e^{a_i^K}\right)^2} = \frac{e^{a_k^K}}{\sum_{i=1}^{n} e^{a_i^K}} \frac{\sum_{i=1}^{n} e^{a_i^K} - e^{a_j^K}}{\sum_{i=1}^{n} e^{a_i^K}}$$

$k=j$

$$= x_k^K \left(1 - x_j^K\right)$$

$$\frac{\partial x_k^K}{\partial a_j^K} = \frac{0 - e^{a_k^K} e^{a_j^K}}{\left(\sum_{i=1}^{n} e^{a_i^K}\right)^2} = -x_k^K x_j^K$$

$$= x_k^K \left(0 - x_j^K\right)$$

$k \neq j$

誤差逆伝播-公式 まとめ 行列表現

出力層

$$\frac{\partial L}{\partial W^K} = X^{(K-1)T} \Delta^K$$

$$\frac{\partial L}{\partial B^K} = \Delta^K$$

$$\Delta^K = \frac{\partial f_L}{\partial X^K} \odot \frac{\partial h_K}{\partial A^K}$$

中間層

$$\frac{\partial L}{\partial W^l} = X^{(l-1)T} \Delta^l$$

$$\frac{\partial L}{\partial B^l} = \Delta^l$$

$$\Delta^l = \Delta^{l+1} W^{(l+1)T} \odot \frac{\partial h_l}{\partial A^l}$$

学習に関する
テクニック

重みの初期値

重みの初期値として適切なものを選べ
１． 全部０
２． 全部同じ値
３． 一様乱数
４． 正規分布　Heの初期値

特徴量スケーリング

特徴量　身長・指の長さ　→　体の特徴
-1~1の実数にスケーリングする

House housing
経度・緯度とか，家の中のベッドの数，世帯年収

同じスケールにしないと桁が違いすぎて
対等な特徴量として扱えない

分布が正規分布になるとより良い

正則化

パラメータは値が小さい方が過学習しない

L1正則化

α sum(|w|)を誤差関数に足す

：スパースなモデルができる

L2正則化

α sum(|w|^2)を誤差関数に足す

：単に過学習が起こりにくい

グリッドサーチ

ハイパーパラメータ
・NNの層の数
・正則化項の係数
・学習率
・．．．