

知能プログラミング演習Ⅰ

## 第2回: 3層ニューラルネットワーク

---

梅津 佑太

2号館 404A: umezu.yuta@nitech.ac.jp

前回作ったディレクトリに移動して今日の課題のダウンロードと解凍

**step1:** `cd ./DLL`

**step2:** `wget http://www-als.ics.nitech.ac.jp/~umezu/DLL19/Lec2.zip`

**step3:** `unzip Lec2.zip`

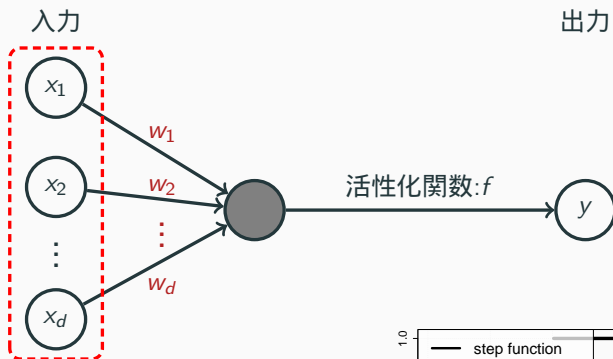
✓ まだ DLL のフォルダを作っていない人は, step1 の前に

`mkdir -p DLL`

でフォルダを作成する

## 1. 3層ニューラルネットワークの学習

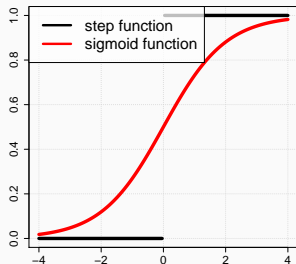
## 前回の復習: パーセプトロン



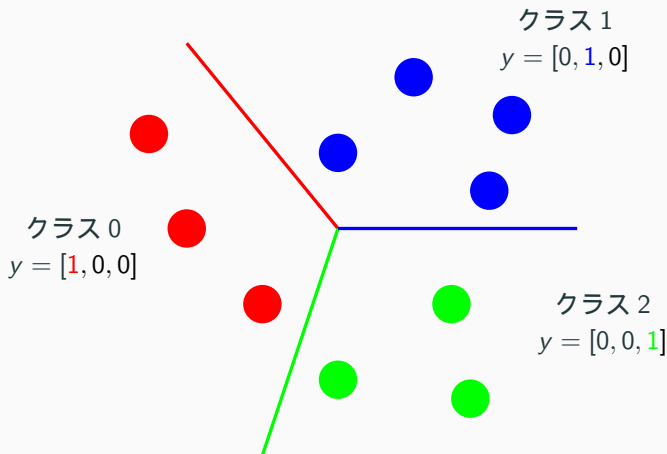
活性化関数  $f$  を用いて

$$y = f(w_0 + w_1x_1 + \cdots + w_dx_d)$$

によって入出力関係をモデル化し、**確率**  
**的勾配降下法**で重みを学習

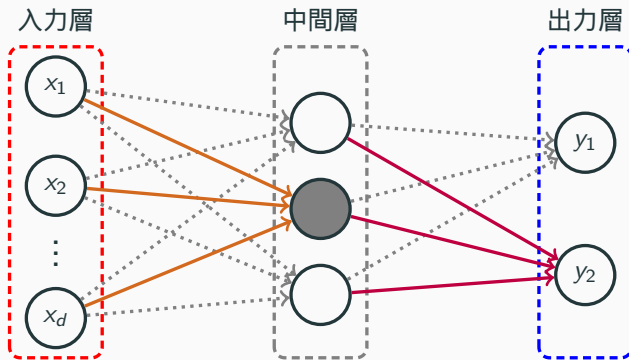


# 多クラス分類

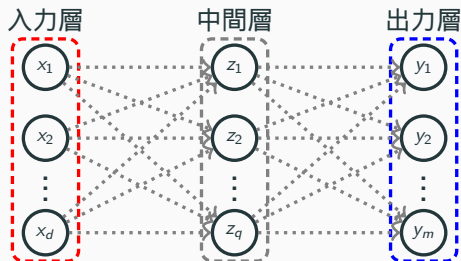


- 入力 ●, ●, ● をうまく分離する境界 (判別境界) を求める問題
- 各入力には正解ラベルがあり, ベクトルで表現する (1-of- $K$  表記)

### 3層ニューラルネットワーク



- パーセプトロンを組み合わせることで複雑なモデルを記述
  - ✓ すべての矢線には, 学習すべき重みがかかっている
  - ✓ 多クラス分類などの出力が多次元のモデルも学習可能
  - ✓ 順方向にネットワークが流れることを順伝播と呼ぶ



- 活性化関数を  $f$  (入力層  $\rightarrow$  中間層),  $g$  (中間層  $\rightarrow$  出力層) とする<sup>1</sup>

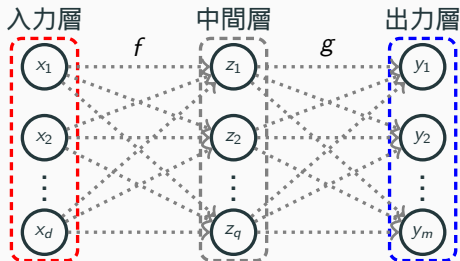
$$z_j = f(w_{j0} + w_{j1}x_1 + \cdots + w_{jd}x_d) = f(\mathbf{w}_j^\top \mathbf{x}), \quad j = 1, 2, \dots, q$$

$$y_k = g(v_{k0} + v_{k1}z_1 + \cdots + v_{kq}z_q) = g(\mathbf{v}_k^\top \mathbf{z}), \quad k = 1, 2, \dots, m$$

- 誤差関数  $E(\mathbf{w}_1, \dots, \mathbf{w}_q, \mathbf{v}_1, \dots, \mathbf{v}_m) = E(W, V)$  をできるだけ小さくするようにパラメータ  $W$  と  $V$  を学習

<sup>1</sup> $\mathbf{x} = (1, x_1, \dots, x_d)$ ,  $\mathbf{z} = (1, z_1, \dots, z_q)$ ,  $\mathbf{w} = (w_0, \dots, w_d)$  と表す

### 3層ニューラルネットワークのモデルの例



- 簡単のため,  $w_{j0} = v_{k0} = 0$  かつ  $q < m$  とする
- $z_j = f(\mathbf{w}_j^\top \mathbf{x}) = \mathbf{w}_j^\top \mathbf{x}, y_k = g(\mathbf{v}_k^\top \mathbf{z}) = \mathbf{v}_k^\top \mathbf{z}$  とすれば,  
 $\mathbf{z} = (z_1, \dots, z_q)^\top = \mathbf{W}\mathbf{x}$  なので,

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} \mathbf{v}_1^\top \mathbf{W}\mathbf{x} \\ \vdots \\ \mathbf{v}_m^\top \mathbf{W}\mathbf{x} \end{bmatrix} = \mathbf{V}\mathbf{W}\mathbf{x}$$

このモデルは縮小ランク回帰モデルとも呼ばれる。



# 活性化関数の例

- 入力層から出力層への活性化関数

- ✓ ReLU<sup>2</sup>:  $f(x) = \max\{0, x\}$

- ✓ シグモイド関数:  $f(x) = 1/(1 + e^{-x})$

- ✓ ハイパボリックタンジェント:  $f(x) = \tanh x = (e^x - e^{-x})/(e^x + e^{-x})$

- 中間層から出力層への活性化関数

- ✓ 恒等写像: 回帰問題で用いられる

$$g(x) = x$$

- ✓ ソフトマックス関数: 多クラス分類問題で用いられるもので, 出力されたベクトルは,  $x$  の所属確率を表す

$$g(x) = \frac{1}{\sum_{k=1}^m e^{x_k}} (e^{x_1}, \dots, e^{x_m})^T$$

---

<sup>2</sup>Rectified Linear Unit

- ReLU<sup>3</sup>:  $f(x) = \max\{0, x\}$

$$f'(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{その他} \end{cases}$$

- シグモイド関数:  $f(x) = 1/(1 + e^{-x})$

$$f'(x) = f(x)(1 - f(x))$$

- ハイパボリックタンジェント:  $f(x) = \tanh x$

$$f'(x) = 1 - f(x)^2$$

---

<sup>3</sup>Rectified Linear Unit

教師データ<sup>4</sup>  $y_1, \dots, y_m$  とモデルの出力  $g(\mathbf{v}_1^\top \mathbf{z}), \dots, g(\mathbf{v}_m^\top \mathbf{z})$  に対して<sup>5</sup>,

- $\ell_2$ -誤差: 回帰問題 ( $g$ : 恒等写像)

$$E(W, V) = \sum_{j=1}^m (y_j - g(\mathbf{v}_j^\top \mathbf{z}))^2$$

- クロスエントロピー: 分類問題 ( $g$ : ソフトマックス関数)

$$E(W, V) = - \sum_{i=1}^m y_i \log g(\mathbf{v}_i^\top \mathbf{z})$$

✓ 特に,  $m = 2$  で  $y \in \{0, 1\}$  の場合は

$$E(\mathbf{w}) = -y \log f(\mathbf{w}^\top \mathbf{x}) - (1 - y) \log(1 - f(\mathbf{w}^\top \mathbf{x}))$$

---

<sup>4</sup>正解のラベルや観測した実数値

<sup>5</sup>各  $k$  に対して,  $g(\mathbf{v}_k^\top \mathbf{z}) = g(v_{k0} + v_{k1}f(\mathbf{w}_1^\top \mathbf{x}) + \dots + v_{kq}f(\mathbf{w}_q^\top \mathbf{x}))$

- 関数  $f(W)$  の行列微分<sup>6</sup>:  $W = (\mathbf{w}_1, \dots, \mathbf{w}_p)^\top \in \mathbb{R}^{p \times q}$  としたとき,

$$\frac{\partial f(W)}{\partial W} = \left( \frac{\partial f(W)}{\partial w_{ij}} \right)_{i=1, \dots, p, j=1, \dots, q} = \left( \frac{\partial f(W)}{\partial \mathbf{w}_1}, \dots, \frac{\partial f(W)}{\partial \mathbf{w}_p} \right)^\top$$

✓ 例:  $f(W) = \text{tr}(W^\top W) = \sum_{i=1}^p \sum_{j=1}^q w_{ij}^2$  ならば

$$\frac{\partial f(W)}{\partial w_{ij}} = 2w_{ij} \quad \Rightarrow \quad \frac{\partial f(W)}{\partial W} = 2W$$

—— パラメータの更新規則 ——

$$W^{(t+1)} = W^{(t)} - \eta_t \left. \frac{\partial E(W, V^{(t)})}{\partial W} \right|_{W=W^{(t)}}$$

$$V^{(t+1)} = V^{(t)} - \eta_t \left. \frac{\partial E(W^{(t)}, V)}{\partial V} \right|_{V=V^{(t)}}$$

<sup>6</sup> $f: \mathbb{R}^{p \times q} \rightarrow \mathbb{R}$  は行列  $W$  を変数として、実数  $f(W)$  を返す関数

## 準備: 誤差関数の勾配の導出 I

中間層から出力層への活性化関数をソフトマックス関数, 誤差関数としてクロスエントロピーを考える.

- パラメータ  $V \in \mathbb{R}^{m \times (q+1)}$  (中間層から出力層へのパラメータ) に関する誤差関数の微分は

$$\begin{aligned}\frac{\partial E(W, V)}{\partial \mathbf{v}_k} &= -\frac{\partial}{\partial \mathbf{v}_k} \sum_{i=1}^m y_i \log g(\mathbf{v}_i^\top \mathbf{z}) = -y_k \frac{\partial \log g(\mathbf{v}_k^\top \mathbf{z})}{\partial \mathbf{v}_k} \\ &= -y_k \underbrace{\frac{1}{g(\mathbf{v}_k^\top \mathbf{z})} \frac{\partial g(\mathbf{v}_k^\top \mathbf{z})}{\partial \mathbf{v}_k}}_{\text{対数関数の合成関数の微分}} = (g(\mathbf{v}_k^\top \mathbf{z}) - y_k) \mathbf{z}\end{aligned}$$

つまり,  $g(V\mathbf{z}) = (g(\mathbf{v}_1^\top \mathbf{z}), \dots, g(\mathbf{v}_m^\top \mathbf{z}))^\top$  とすれば,

$$\Rightarrow \frac{\partial E(W, V)}{\partial V} = \left( \frac{\partial E(W, V)}{\partial \mathbf{v}_1}, \dots, \frac{\partial E(W, V)}{\partial \mathbf{v}_m} \right)^\top = (g(V\mathbf{z}) - \mathbf{y}) \mathbf{z}^\top$$

## 準備: 誤差関数の勾配の導出 II

$z = f(Wx)$  は入力層から中間層へのパラメータ  $W$  に依存するので,

$$\begin{aligned}\frac{\partial E(W, V)}{\partial \mathbf{w}_j} &= -\frac{\partial}{\partial \mathbf{w}_j} \sum_{i=1}^m y_i \log g(\mathbf{v}_i^\top \mathbf{z}) = -\sum_{i=1}^m y_i \frac{\partial \log g(\mathbf{v}_i^\top \mathbf{z})}{\partial \mathbf{w}_j} \\ &= -\sum_{i=1}^m y_i \underbrace{\frac{\partial \log g(\mathbf{v}_i^\top \mathbf{z})}{\partial \mathbf{v}_i^\top \mathbf{z}} \frac{\partial \mathbf{v}_i^\top \mathbf{z}}{\partial \mathbf{w}_j}}_{\text{合成関数の微分}} = \sum_{i=1}^m (g(\mathbf{v}_i^\top \mathbf{z}) - y_i) \frac{\partial \mathbf{v}_i^\top \mathbf{z}}{\partial \mathbf{w}_j}\end{aligned}$$

ここで,

$$\begin{aligned}\frac{\partial \mathbf{v}_i^\top \mathbf{z}}{\partial \mathbf{w}_j} &= \frac{\partial}{\partial \mathbf{w}_j} \{v_{i0} + v_{i1} f(\mathbf{w}_1^\top \mathbf{x}) + \cdots + v_{iq} f(\mathbf{w}_q^\top \mathbf{x})\} \\ &= v_{ij} \frac{\partial f(\mathbf{w}_j^\top \mathbf{x})}{\partial \mathbf{w}_j} = v_{ij} \nabla f(\mathbf{w}_j^\top \mathbf{x}) \mathbf{x}\end{aligned}$$

より,

## 準備: 誤差関数の勾配の導出 III

$\tilde{v}_j = (v_{1j}, \dots, v_{mj})^\top$  とすれば,

$$\frac{\partial E(W, V)}{\partial \mathbf{w}_j} = \sum_{i=1}^m (g(\mathbf{v}_i^\top \mathbf{z}) - y_i) v_{ij} \nabla f(\mathbf{w}_j^\top \mathbf{x}) \mathbf{x} = \underbrace{\tilde{v}_j^\top (g(V\mathbf{z}) - \mathbf{y}) \nabla f(\mathbf{w}_j^\top \mathbf{x})}_{\text{スカラー}} \mathbf{x}$$

なので,  $\tilde{V} = (\tilde{v}_1, \dots, \tilde{v}_q)$  を,  $V$  から 1 列目を取り除いた行列<sup>7</sup> として,

$$\frac{\partial E(W, V)}{\partial W} = \left[ \tilde{V}^\top (g(V\mathbf{z}) - \mathbf{y}) \odot \nabla f(W\mathbf{x}) \right] \mathbf{x}^\top$$

となる. ただし, 同じ長さのベクトル  $\mathbf{v}, \mathbf{w}$  に対して,

$$\mathbf{v} \odot \mathbf{w} = (v_i w_i)_i$$

は成分ごとの積 (アダマール積) を表す.

---

<sup>7</sup>  $V$  から切片項に対応する部分を取り除いたもの

## 誤差逆伝播法

まとめると、確率的勾配降下法の各ステップにおいて、パラメータは以下の通り更新すれば良い:

### 誤差逆伝播法

$$V^{(t+1)} = V^{(t)} - \eta_t (g(V^{(t)\top} \mathbf{z}^{(t)}) - \mathbf{y}) \mathbf{z}^{(t)\top}$$

$$W^{(t+1)} = W^{(t)} - \eta_t \left[ \tilde{V}^{(t)\top} (g(V^{(t)} \mathbf{z}^{(t)}) - \mathbf{y}) \odot \nabla f(W^{(t)} \mathbf{x}) \right] \mathbf{x}^\top$$

- 勾配を計算する際には,

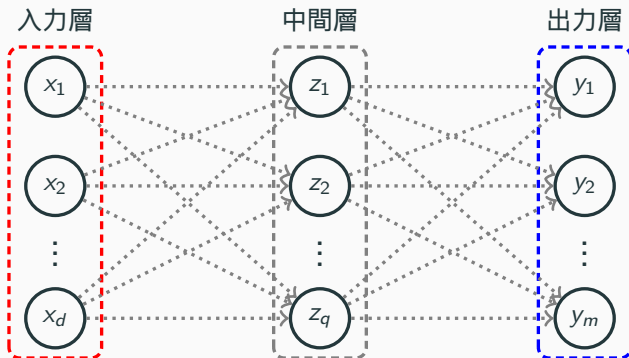
$$\delta_2 = g(V^{(t)\top} \mathbf{z}^{(t)}) - \mathbf{y}$$

$$\delta_1 = \tilde{V}^{(t)\top} (g(V^{(t)} \mathbf{z}^{(t)}) - \mathbf{y}) \odot \nabla f(W^{(t)} \mathbf{x})$$

を  $\delta_2, \delta_1$  の順に定義しておくと便利 (逆伝播の由来)



# 誤差逆伝播法

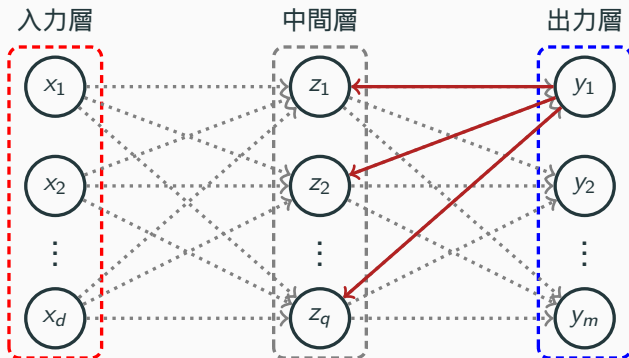


## 誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

# 誤差逆伝播法

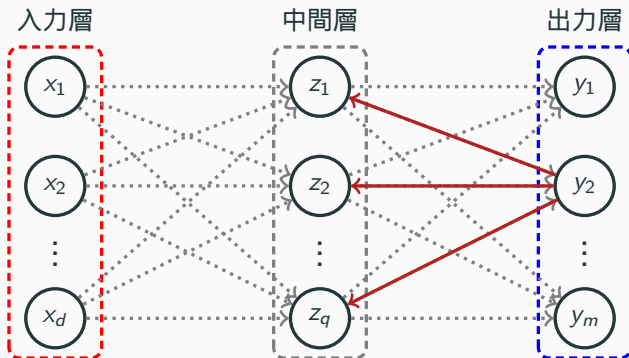


誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

# 誤差逆伝播法

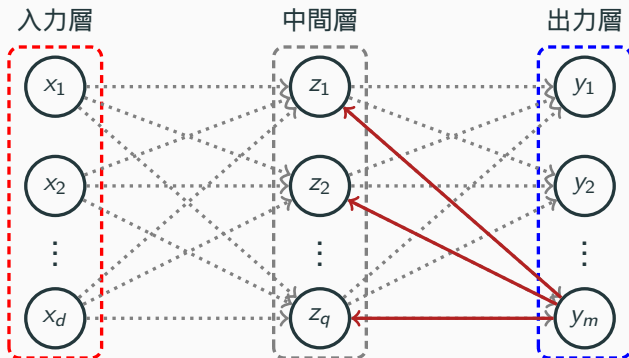


誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

# 誤差逆伝播法

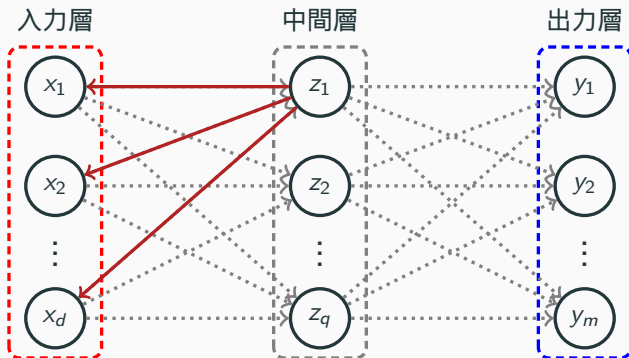


誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

# 誤差逆伝播法

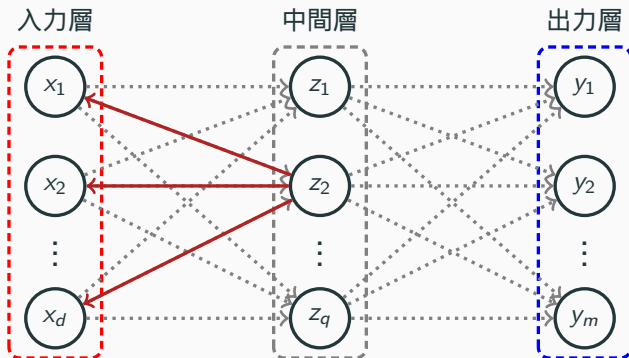


## 誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

# 誤差逆伝播法

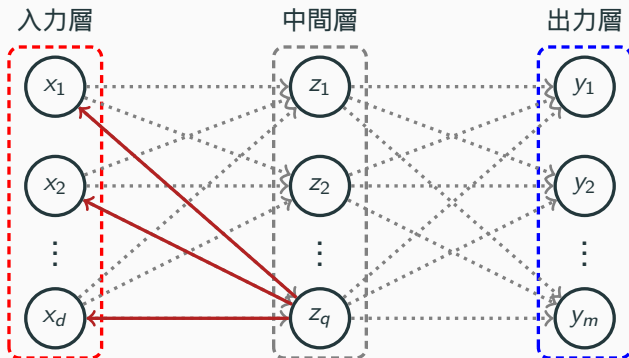


誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

# 誤差逆伝播法



誤差逆伝播法

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t - \eta \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) \mathbf{z}^t$$

$$\mathbf{w}_j^{t+1} = \mathbf{w}_j^t - \eta \sum_{k=1}^m \nabla g(\mathbf{v}_k^{t\top} \mathbf{z}^t) (g(\mathbf{v}_k^{t\top} \mathbf{z}^t) - y_k) v_{jk}^t \nabla f(\mathbf{w}_j^{t\top} \mathbf{x}) \mathbf{x}$$

## 課題のための準備: 分類結果の評価

- 学習後のモデルの出力 (各クラスの所属確率) が最大のクラスにテストデータを分類
  - ✓ 例えば, ソフトマックス関数の出力が  $[0.8, 0.04, 0.1, 0.06]$  なら, 予測結果はクラス 0

		予測結果			
		0	1	2	3
実際の クラス	0	10	3	3	4
	1	2	8	5	5
	2	0	5	12	3
	3	3	2	3	12

- 対角成分は分類結果の正答数を表しており, 正答率は (正答数の合計)/(データ数) で評価する
  - ✓ 上の例なら正答率は  $42/80 = 52.5\%$