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

課題のダウンロード

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`

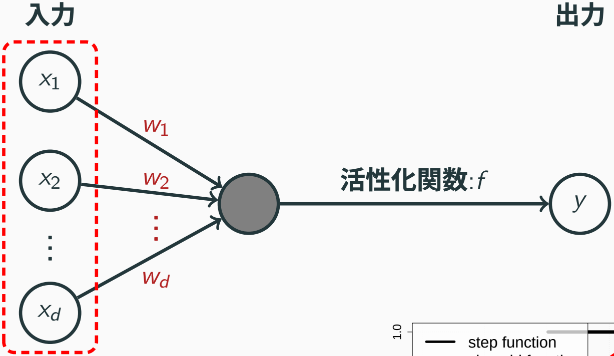
step3: `unzip Lec2.zip`

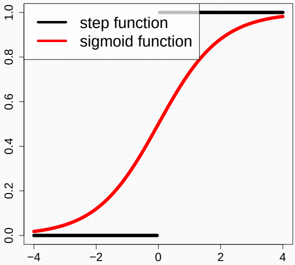
✓ **まだ** DLL **のフォルダを作ってない人は**, step1 **の前に**
`mkdir -p DLL`
でフォルダを作成する

今日の講義内容

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



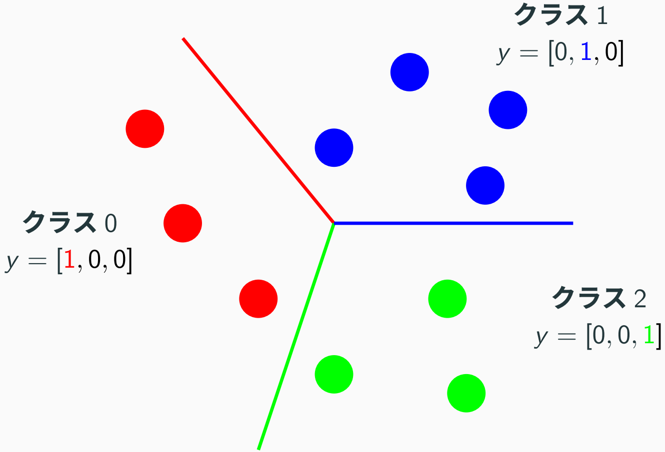


活性化関数 f を用いて

$$y = f(w_0 + w_1 x_1 + \cdots + w_d x_d)$$

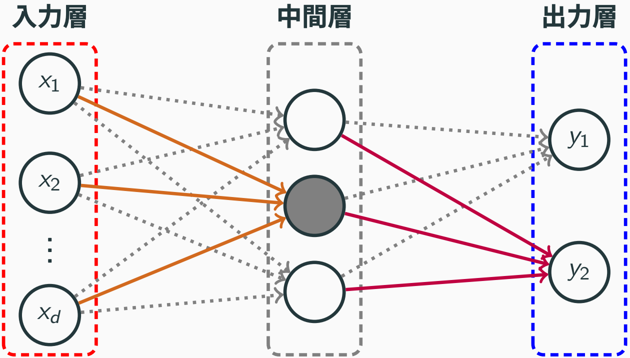
によって入出力関係をモデル化し，確率的勾配降下法で重みを学習

多クラス分類



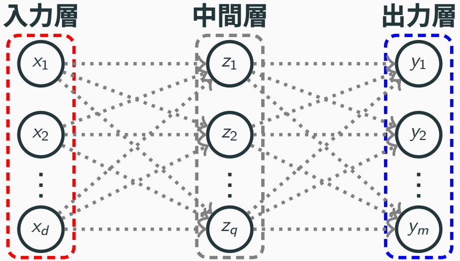
- **入力 ●, ●, ● をうまく分離する境界 (判別境界) を求める問題**
- **各入力には正解ラベルがあり, ベクトルで表現する (1-of-K 表記)**

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

モデル



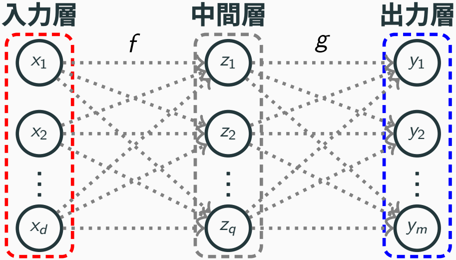
- **活性化関数を** f (**入力層** $\to$ **中間層**), g (**中間層** $\to$ **出力層**) **とする** [1]

$$z_j = f(w_{j0} + w_{j1}x_1 + \cdots + w_{jd}x_d) = f(\boldsymbol{w}_j{}^\top \boldsymbol{x}), \quad j = 1, 2, \ldots, q$$
$$y_k = g(v_{k0} + v_{k1}z_1 + \cdots + v_{kq}z_q) = g(\boldsymbol{v}_k{}^\top \boldsymbol{z}), \quad k = 1, 2, \ldots, m$$

- **誤差関数** $E(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_q, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_m) = E(W, V)$ **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**

[1] $\boldsymbol{x} = (1, x_1, \ldots, x_d), \boldsymbol{z} = (1, z_1, \ldots, z_q), \boldsymbol{w} = (w_0, \ldots, w_d)$ 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**
- $z_j = f(\boldsymbol{w}_j^\top \boldsymbol{x}) = \boldsymbol{w}_j^\top \boldsymbol{x}, y_k = g(\boldsymbol{v}_j^\top \boldsymbol{x}) = \boldsymbol{v}_j^\top z$ **とすれば**, $\boldsymbol{z} = (z_1, \ldots, z_q)^\top = W\boldsymbol{x}$ **なので**,

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

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[2]: $f(x) = \max\{0, x\}$
  - ✓ **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**

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

[2]Rectified Linear Unit

活性化関数の微分の例

- ReLU[3]: $f(x) = \max\{0, x\}$

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

- **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**: $f(x) = \tanh x$

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

[3] Rectified Linear Unit

誤差関数の例

教師データ[4] $y_1, \ldots, y_m$ とモデルの出力 $g(\boldsymbol{v}_1^\top \boldsymbol{z}), \ldots, g(\boldsymbol{v}_m^\top \boldsymbol{z})$ に対して[5],

- ℓ_2-誤差: 回帰問題 (g : 恒等写像)

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

- クロスエントロピー: 分類問題 (g : ソフトマックス関数)

$$E(W, V) = -\sum_{i=1}^{m} y_j \log g(\boldsymbol{v}_j^\top \boldsymbol{z})$$

✓ 特に, $m = 2$ で $y \in \{0, 1\}$ の場合は

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

[4] 正解のラベルや観測した実数値

[5] 各 k に対して, $g(\boldsymbol{v}_k^\top \boldsymbol{z}) = g(v_{k0} + v_{k1} f(\boldsymbol{w}_1^\top \boldsymbol{x}) + \cdots + v_{kq} f(\boldsymbol{w}_q^\top \boldsymbol{x}))$

勾配降下法

- **関数** $f(W)$ **の行列微分**[6]: $W = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_p)^\top \in \mathbb{R}^{p \times q}$ **としたとき**,

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

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

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

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

[6] $f : \mathbb{R}^{p \times q} \to \mathbb{R}$ **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

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

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

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

準備: 誤差関数の勾配の導出 II

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

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

$$= -\sum_{i=1}^{m} y_i \underbrace{\frac{\partial \log g(\boldsymbol{v}_i^\top \boldsymbol{z})}{\partial \boldsymbol{v}_i^\top \boldsymbol{z}} \frac{\partial \boldsymbol{v}_i^\top \boldsymbol{z}}{\partial \boldsymbol{w}_j}}_{\text{合成関数の微分}} = \sum_{i=1}^{m} (g(\boldsymbol{v}_i^\top \boldsymbol{z}) - y_i) \frac{\partial \boldsymbol{v}_i^\top \boldsymbol{z}}{\partial \boldsymbol{w}_j}$$

ここで,

$$\frac{\partial \boldsymbol{v}_i^\top \boldsymbol{z}}{\partial \boldsymbol{w}_j} = \frac{\partial}{\partial \boldsymbol{w}_j} \{v_{i0} + v_{i1} f(\boldsymbol{w}_1^\top \boldsymbol{x}) + \cdots + v_{iq} f(\boldsymbol{w}_q^\top \boldsymbol{x})\}$$

$$= v_{ij} \frac{\partial f(\boldsymbol{w}_j^\top \boldsymbol{x})}{\partial \boldsymbol{w}_j} = v_{ij} \nabla f(\boldsymbol{w}_j^\top \boldsymbol{x}) \boldsymbol{x}$$

より,

準備: 誤差関数の勾配の導出 III

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

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

なので, $\tilde{V} = (\tilde{\boldsymbol{v}}_1, \ldots, \tilde{\boldsymbol{v}}_q)$ を, V から 1 列目を取り除いた行列 [7] として,

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

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

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

は成分ごとの積 (アダマール積) を表す.

[7] 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

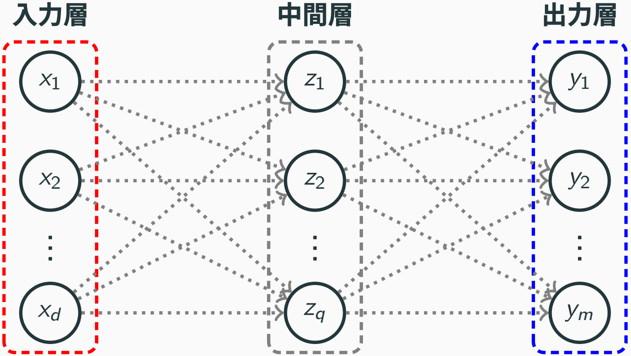
$$V^{(t+1)} = V^{(t)} - \eta_t (g(V^{(t)\top} \boldsymbol{z}^{(t)}) - \boldsymbol{y}) {\boldsymbol{z}^{(t)}}^\top$$
$$W^{(t+1)} = W^{(t)} - \eta_t \left[\tilde{V}^{(t)\top} (g(V^{(t)} \boldsymbol{z}^{(t)}) - \boldsymbol{y}) \odot \nabla f(W^{(t)} \boldsymbol{x}) \right] \boldsymbol{x}^\top$$

- 勾配を計算する際には，

$$\delta_2 = g(V^{(t)\top} \boldsymbol{z}^{(t)}) - \boldsymbol{y}$$
$$\delta_1 = \tilde{V}^{(t)\top} (g(V^{(t)} \boldsymbol{z}^{(t)}) - \boldsymbol{y}) \odot \nabla f(W^{(t)} \boldsymbol{x})$$

を δ_2, δ_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

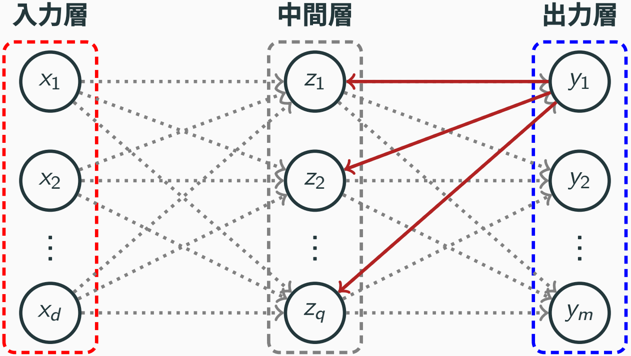


誤差逆伝播法

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

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

誤差逆伝播法

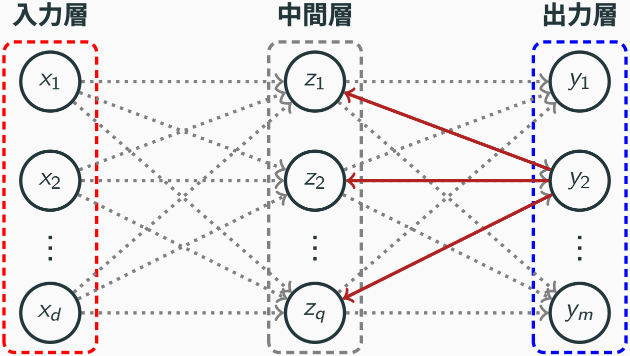


誤差逆伝播法

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

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

誤差逆伝播法

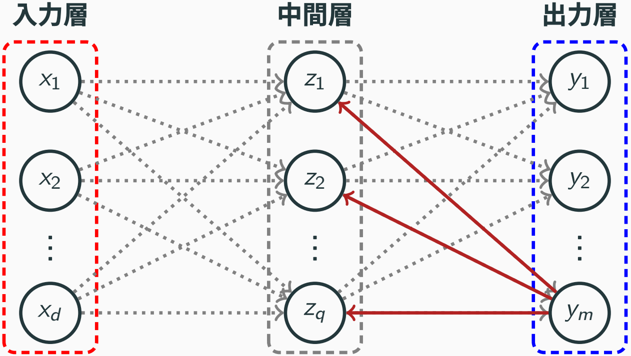


誤差逆伝播法

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

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

誤差逆伝播法

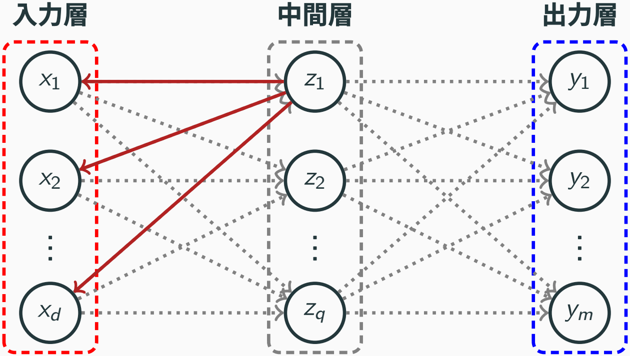


誤差逆伝播法

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

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

誤差逆伝播法

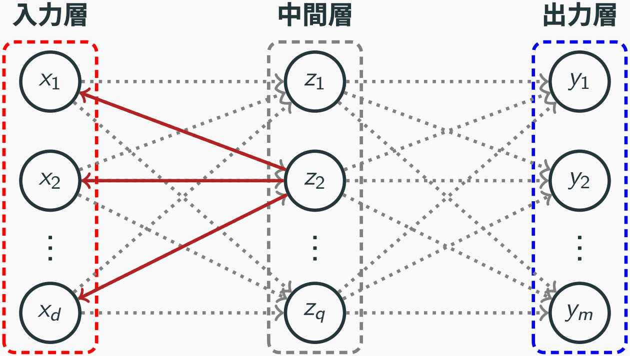


誤差逆伝播法

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

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

誤差逆伝播法

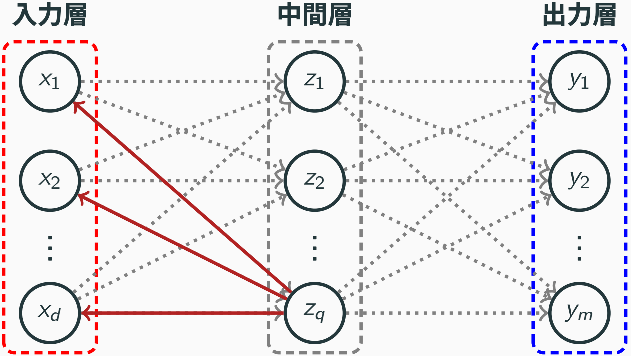


誤差逆伝播法

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

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

誤差逆伝播法



誤差逆伝播法

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

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

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 $42/80 = 52.5\%$