

NICO2AI パイロット講義 #5

勾配法と誤差逆伝搬法

長野 祥大

2017/07/15

本日の概要

線形回帰モデルの復習 (解析解)

線形回帰モデルを異なるアルゴリズム (勾配法) で解く

線形回帰モデルを識別問題に拡張したロジスティック回帰モデル

問題を多クラス分類に拡張し、順伝搬ニューラルネットワークを導入する

順伝搬ニューラルネットワークの最適化アルゴリズムとして誤差逆伝搬法を理解する

(復習) 線形モデルの二乗誤差最小化: 解析解

データ $\{(\mathbf{x}^{(i)}, y^{(i)})\}$ に対する線形モデル,

$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$

の empirical risk minimization (ERM) 問題を考える.

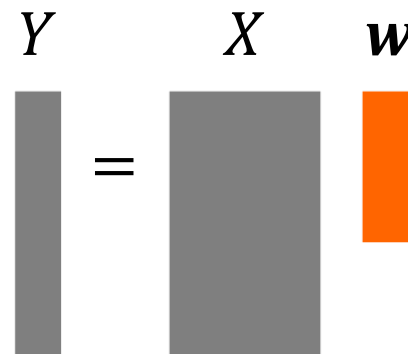
$$\min_{\mathbf{w}} L(\mathbf{w}) = \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

2次関数の極小値は大域解となるので,

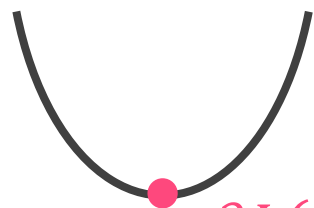
$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} = \mathbf{X}^T (\mathbf{X}\mathbf{w} - \mathbf{y}) = 0$$

$$\mathbf{X}^T \mathbf{X} \mathbf{w} = \mathbf{X}^T \mathbf{y}$$

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

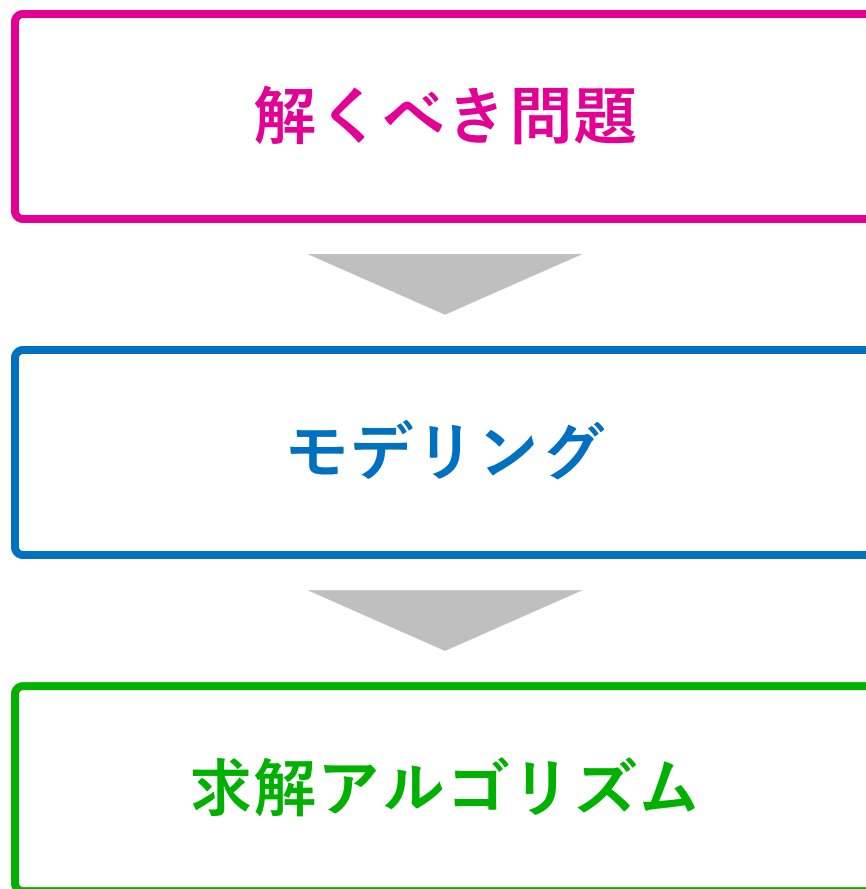
$$\mathbf{Y} = \mathbf{X} \mathbf{w}$$


$L(\mathbf{w})$

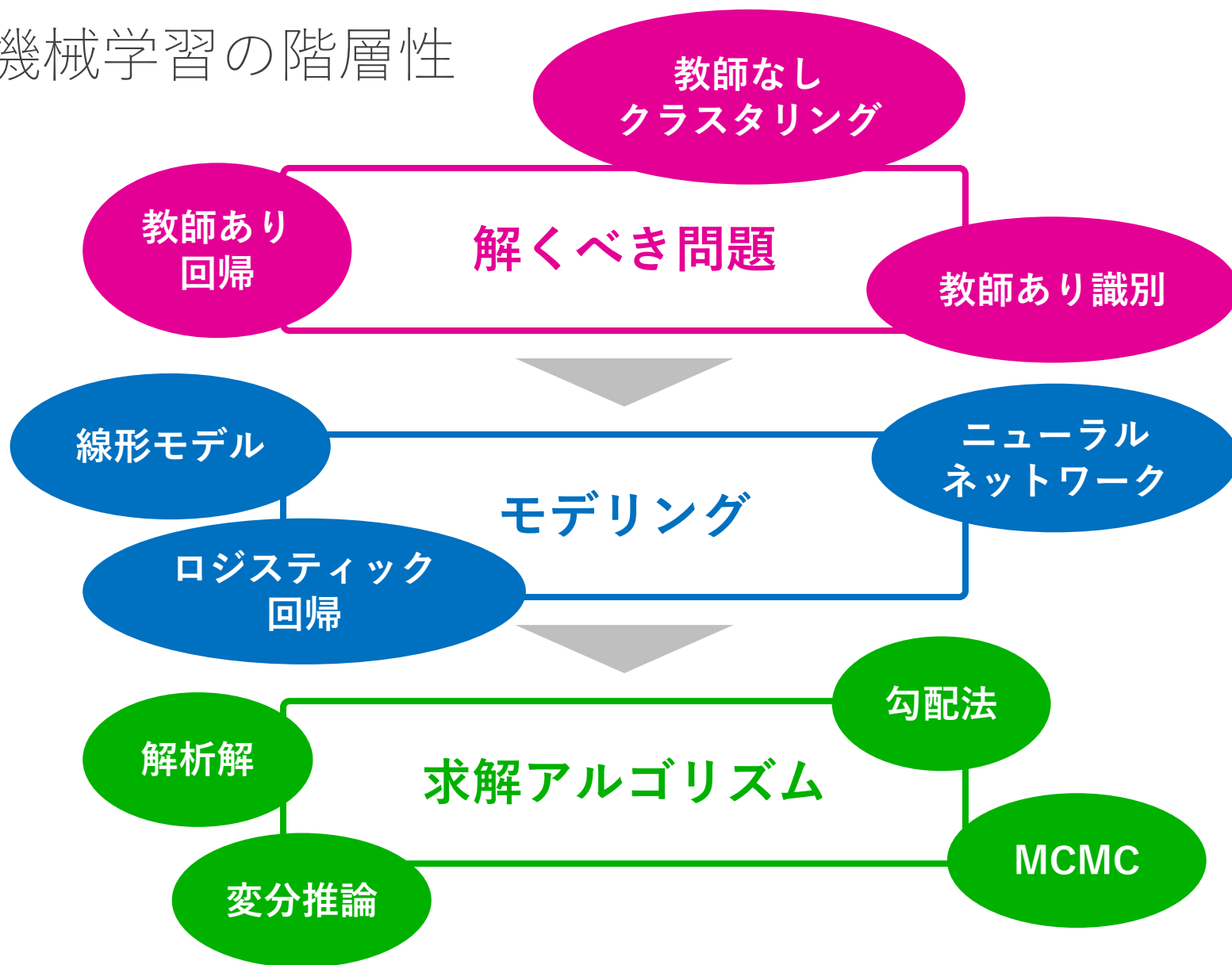


$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} = 0$$

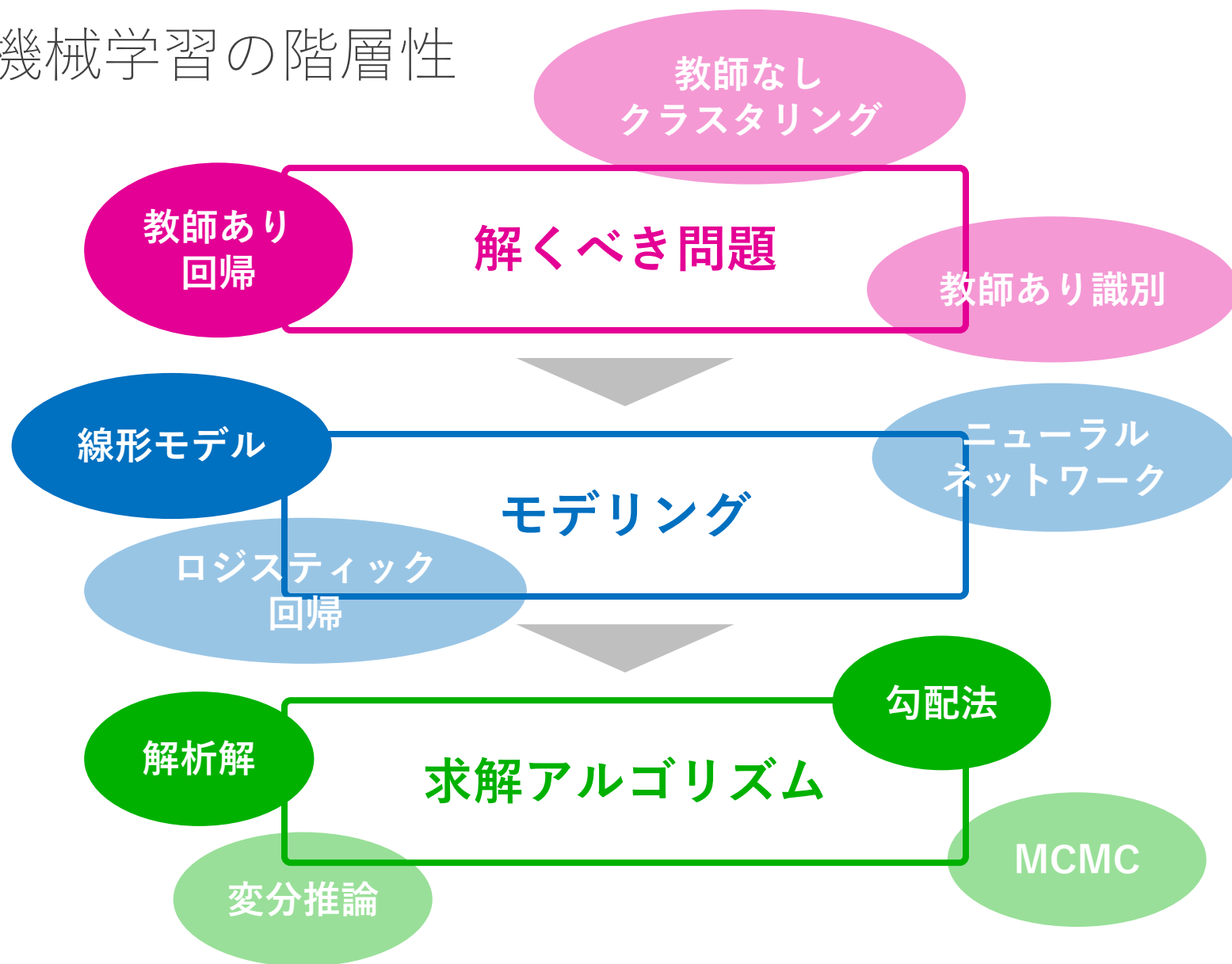
機械学習の階層性



機械学習の階層性



機械学習の階層性

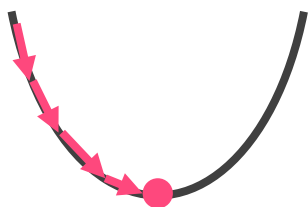


線形モデルの二乗誤差最小化: 勾配法

for t in 1...T

$$\begin{aligned}\mathbf{w}_{t+1} &= \mathbf{w}_t - \eta \frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} \\ &= \mathbf{w}_t - \eta X^T (X \mathbf{w}_t - y)\end{aligned}$$

$L(\mathbf{w})$



同じ**モデリング**でも異なる**アルゴリズム**で解を求めることができる！

解析解がわからなくても徐々に坂を下ること
で(局所)解が求まる

つくってみよう1

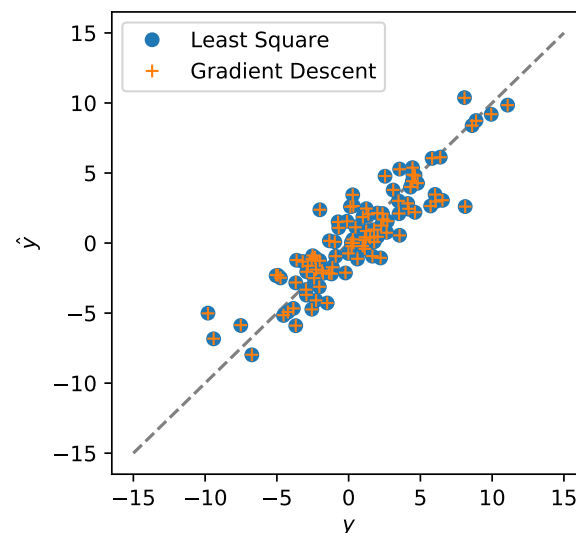
課題

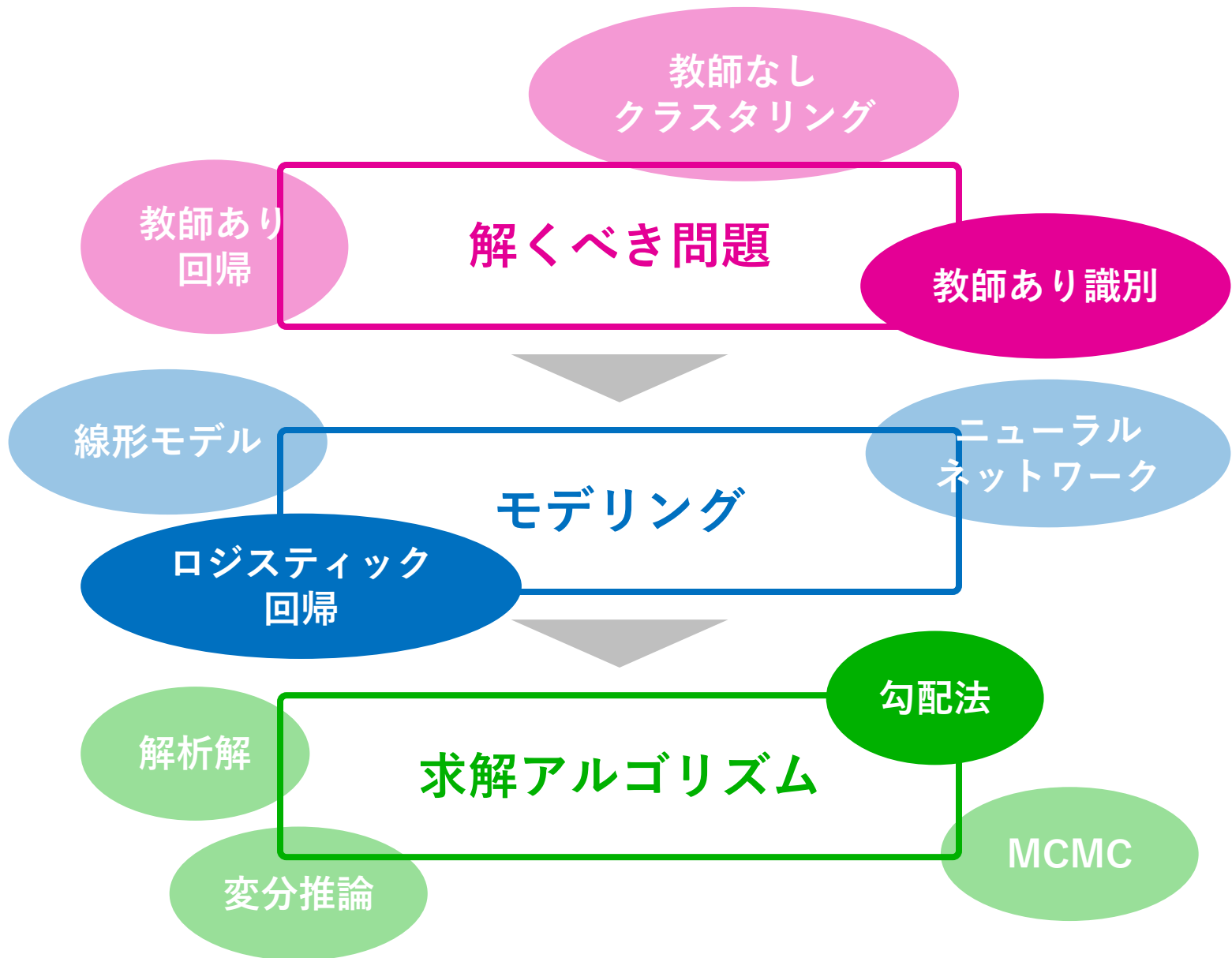
線形モデルの二乗誤差最小化を**解析解**と**勾配法**の2つで実装する

※1 データは以下の式から生成する

$$\begin{aligned} \mathbf{y} &= \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon} \\ X_{ij}, \mathbf{w}_i &\sim \mathcal{N}(0, 1.0) \\ \boldsymbol{\epsilon}_i &\sim \mathcal{N}(0, 4.0) \end{aligned}$$

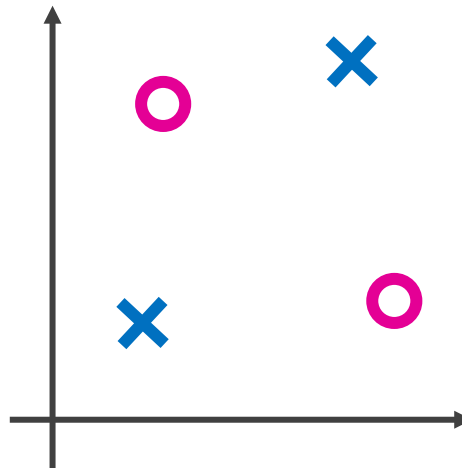
※2 データ数N: 100, 次元数D: 20





教師あり識別問題

データ x からデータのラベル $d \in \{0, 1\}$ を予測する問題.



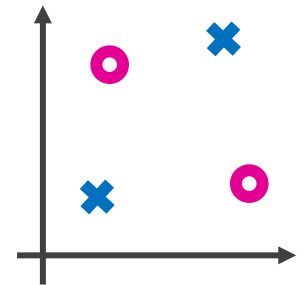
ロジスティック回帰

教師あり識別に用いられるモデル

名前がややこしい: 条件付き確率 $\Pr(d = 1|\mathbf{x})$ の回帰 = 識別問題

$$\text{logit}(y) = \ln \frac{y}{1-y} = \mathbf{w}^T \mathbf{x} + \epsilon$$

$$\text{logit} \begin{array}{|c|} \hline \Pr(d = 1|\mathbf{x}) \\ \hline \end{array} = \begin{array}{|c|} \hline X \\ \hline \end{array} \mathbf{w}$$



シグモイド関数 $1/(1 + e^{-ax})$ を用いて以下のようにも書き表せる

$$y(\mathbf{x}; \mathbf{w}) = \Pr(d = 1|\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$$

ロジスティック回帰

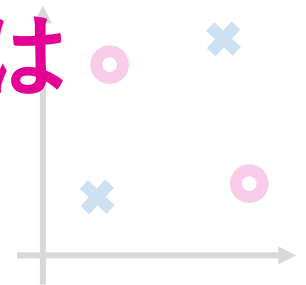
教師あり識別に用いられるモデル

名前がややこしい: 条件付き確率 $\Pr(d = 1|\mathbf{x})$ の回帰 = 識別問題

条件付き確率の回帰の誤差は
何で測るべきか?

logit

=



シグモイド関数 $1/(1 + e^{-ax})$ を用いて以下のようにも書き表せる

$$y(\mathbf{x}; \mathbf{w}) = \Pr(d = 1|\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$$

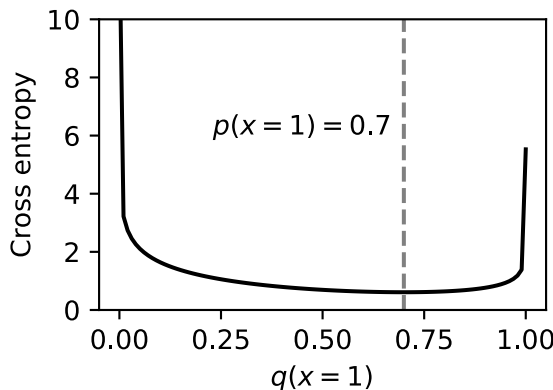
負の対数尤度の最小化

データ \mathbf{x} が与えられた時の d の尤度の最大化を考える

$$\begin{aligned} L(\mathbf{w}) &= -\log \prod_{i=1}^N p(d^{(i)} | \mathbf{x}^{(i)}; \mathbf{w}) = -\log \prod_{i=1}^N \left\{ y(\mathbf{x}^{(i)}; \mathbf{w}) \right\}^{d^{(i)}} \left\{ 1 - y(\mathbf{x}^{(i)}; \mathbf{w}) \right\}^{1-d^{(i)}} \\ &= -\sum_{i=1}^N \left[d^{(i)} \log y(\mathbf{x}^{(i)}; \mathbf{w}) + (1 - d^{(i)}) \log (1 - y(\mathbf{x}^{(i)}; \mathbf{w})) \right] \end{aligned}$$

この形は真の分布 p と推定した分布 q の**クロスエントロピー誤差**として知られる

$$H(p, q) = -\sum_d p(d) \log q(d)$$



左図は $p(d = 1) = 0.7$ とした時の y の値とクロスエントロピー誤差の関係

$y = 0.7$ でクロスエントロピー誤差は最小

ロジスティック回帰モデルのクロスエントロピー誤差最小化

条件付き確率のモデリング

$$y(\mathbf{x}; \mathbf{w}) = \Pr(d = 1 | \mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$$

$$H(y) = -\{d \log y + (1 - d) \log(1 - y)\}$$

解いてみよう1

目的関数を $L(\mathbf{w}) = H(y(\mathbf{x}; \mathbf{w}))$ として目的関数の勾配を求めよ

ヒント1: $\frac{\partial H}{\partial \mathbf{w}} = \frac{\partial H}{\partial y} \frac{\partial y}{\partial \mathbf{w}}$ の形に分解する (連鎖律: 合成関数の微分)

ヒント2: シグモイド関数 $\sigma(z)$ の微分が $\frac{\partial}{\partial z} \sigma(z) = \sigma(z) \cdot (1 - \sigma(z))$ となることを利用する

ロジスティック回帰モデルのクロスエントロピー誤差最小化

条件付き確率のモデリング

$$y(\mathbf{x}; \mathbf{w}) = \Pr(d = 1 | \mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$$

$$H(y) = -\{d \log y + (1 - d) \log(1 - y)\}$$

勾配法による求解

$$\begin{aligned} \frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} &= \frac{\partial H(y)}{\partial y} \frac{\partial y}{\partial \mathbf{w}} \\ &= \frac{y - d}{y(1 - y)} \cdot \frac{\partial y}{\partial \mathbf{w}} \\ &= \frac{y - d}{y(1 - y)} \cdot y(1 - y) \cdot \mathbf{x} \\ &= (y - d)\mathbf{x} \end{aligned}$$

シグモイド関数の微分

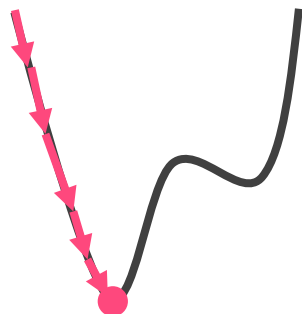
$$\begin{aligned} \frac{\partial}{\partial z} \sigma(z) &= \frac{\partial}{\partial z} \frac{1}{1 + e^{-z}} \\ &= -(1 + e^{-z})^{-2} \cdot -e^{-z} \\ &= \frac{1}{1 + e^{-z}} \cdot \frac{e^{-z}}{1 + e^{-z}} \\ &= \frac{1}{1 + e^{-z}} \cdot \left(1 - \frac{1}{1 + e^{-z}}\right) \\ &= \sigma(z) \cdot (1 - \sigma(z)) \end{aligned}$$

ロジスティック回帰モデルの勾配法による求解

for t in 1...T

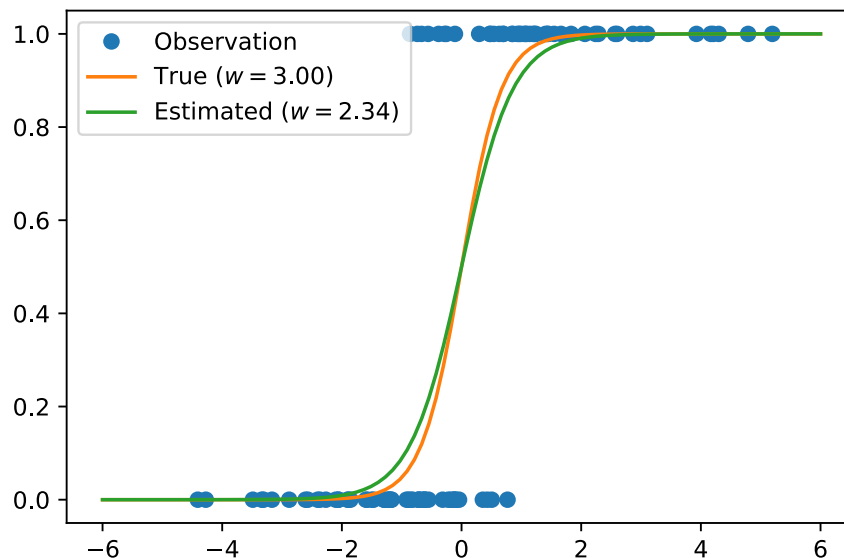
$$\begin{aligned}\mathbf{w}_{t+1} &= \mathbf{w}_t - \eta \frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} \\ &= \mathbf{w}_t - \eta X^T (y - d)\end{aligned}$$

$L(\mathbf{w})$



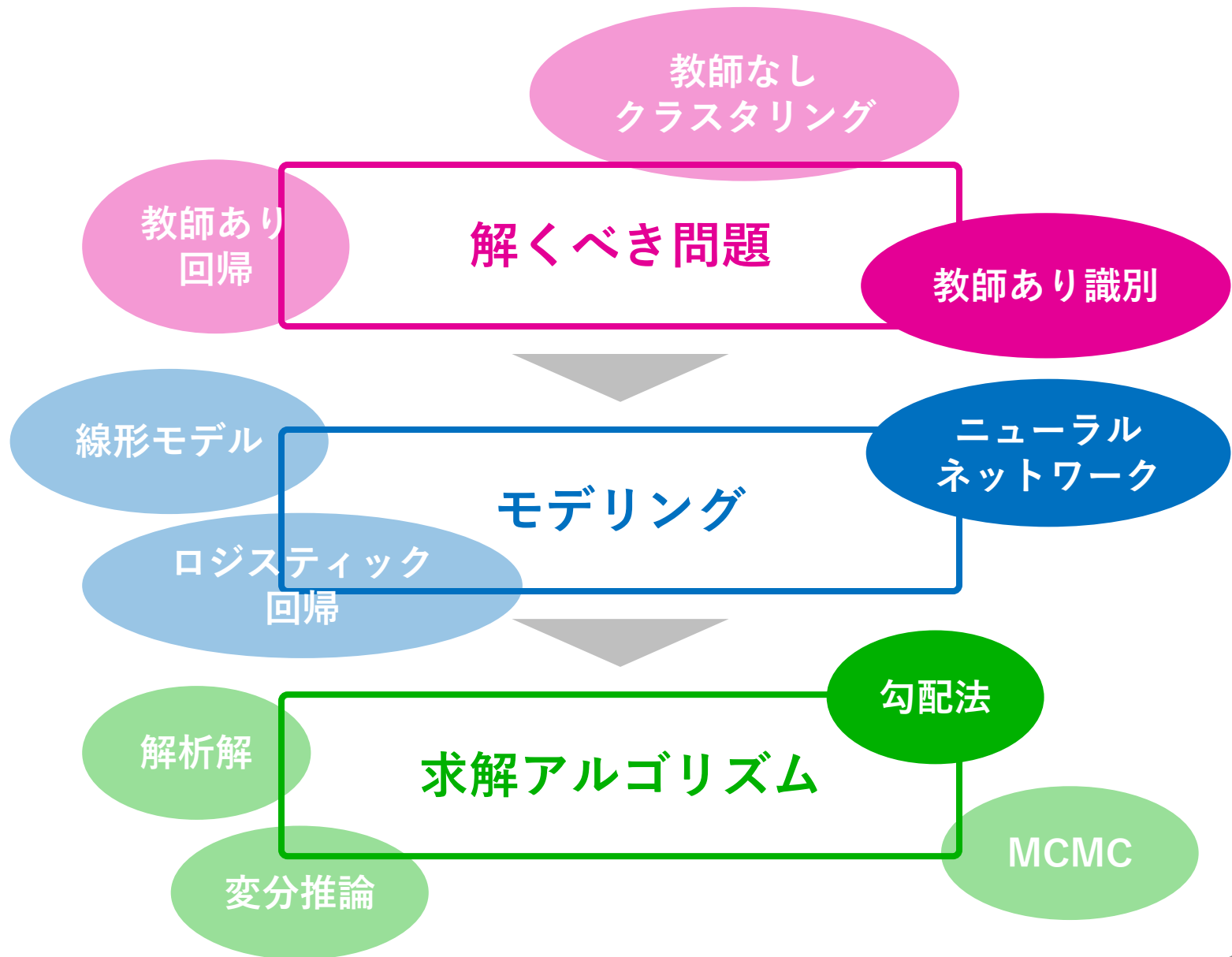
誤差関数の形は一見してよくわからないが、
勾配を計算することで(局所)解が求まる！

つくってみよう2



課題

1. シグモイド関数 $\sigma(z)$ を実装する
2. クロスエントロピー誤差関数 $H(d, y)$ を実装する
3. 1次元の**ロジスティック回帰モデル**のクロスエントロピー誤差最小化を**勾配法**で実装する

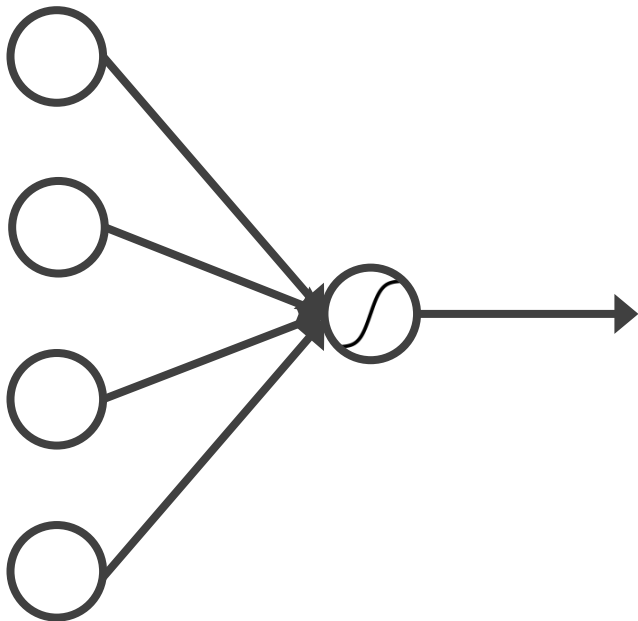


ニューラルネットワーク

線形写像と非線形活性化関数からなる計算グラフ

$$\mathbf{u} = \mathbf{W}\mathbf{x} + \mathbf{b}$$

$$\mathbf{z} = f(\mathbf{u})$$



例えば非線形活性化関数にシグモイド関数 $f(\mathbf{u}) = \frac{1}{1+e^{-u}}$ を取れば, **ロジスティック回帰と等価**

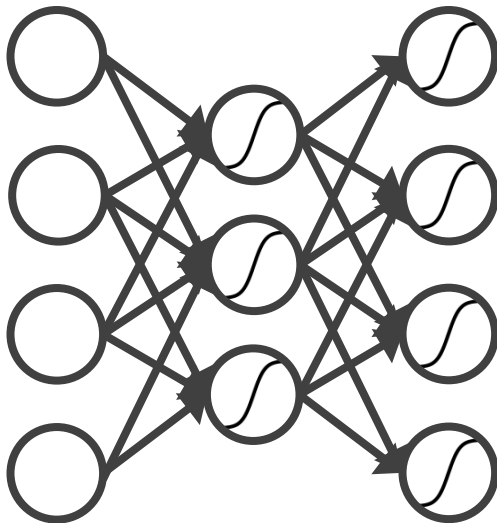
最適化するパラメーターは \mathbf{W} と \mathbf{b}

多層ニューラルネットワーク

線形写像と非線形活性化関数を多層に積んだもの

$$\mathbf{u}^{l+1} = \mathbf{W}^{l+1} \mathbf{z}^l + \mathbf{b}^{l+1}$$

$$\mathbf{z}^l = f(\mathbf{u}^l)$$



これまでのモデルとは異なり、**パラメーター($\mathbf{W}^l, \mathbf{b}^l$)に関して線形でないモデル**

⇒ 目的関数が非凸になる

e.g.) 3層NN

$$\mathbf{z}^3 = f(\mathbf{W}^3 f(\mathbf{W}^2 \mathbf{x} + \mathbf{b}^2) + \mathbf{b}^3)$$

多クラス分類

ソフトマックス関数

入力データを有限個のクラスに分類することを考える時，出力 y_1, \dots, y_K の総和が常に1になるように制約を加えた関数

$$y_k = z_k^{(L)} = \frac{\exp(u_k^{(L)})}{\sum_{j=1}^K \exp(u_j^{(L)})}$$

2値分類のときと同様に負の対数尤度の最小化を行えば以下のクロスエントロピー誤差が得られる.

$$L(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^K d_{nk} \log y_k(\mathbf{x}_n; \mathbf{w})$$

3層ニューラルネットワークのクロスエントロピー誤差最小化

条件付き確率のモデリング

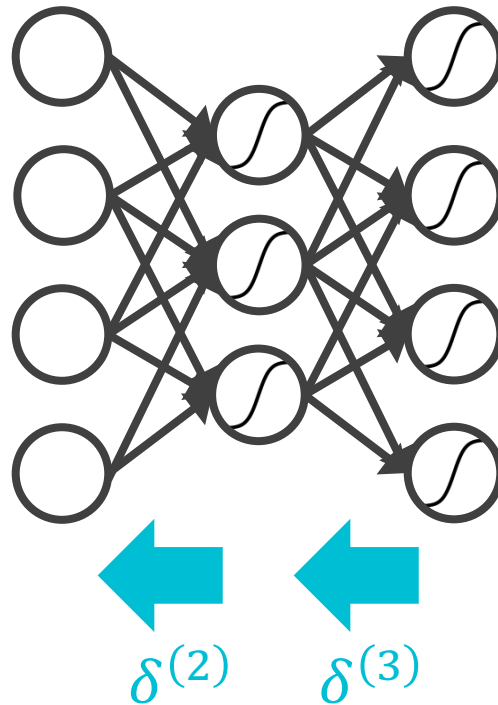
$$\mathbf{y}(\mathbf{x}; \mathbf{w}) = \mathbf{z}^{(3)} = f^{(3)}(W^{(3)} f^{(2)}(W^{(2)} \mathbf{x} + \mathbf{b}^{(2)}) + \mathbf{b}^{(3)})$$

$$z_k^{(3)} = f^{(3)}(u_k^{(3)}) = \frac{\exp(u_k^{(3)})}{\sum_{j=1}^K \exp(u_j^{(3)})}$$

$$L(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^K d_{nk} \log y_k(\mathbf{x}_n; \mathbf{w})$$

⇒ 深い(入力に近い)パラメーターに関する誤差の微分をいかに効率的に計算するか？

誤差逆伝搬法



ニューラルネットワークの重み W とバイアス b に関する勾配を出力に近い層から順番に伝搬しながら効率的に計算するアルゴリズム

誤差逆伝搬法

NNの各層の結合に対する勾配を考える



ニューラルネットワークの重み W とバイアス b に関する勾配を出力に近い層から順番に伝搬しながら効率的に計算するアルゴリズム

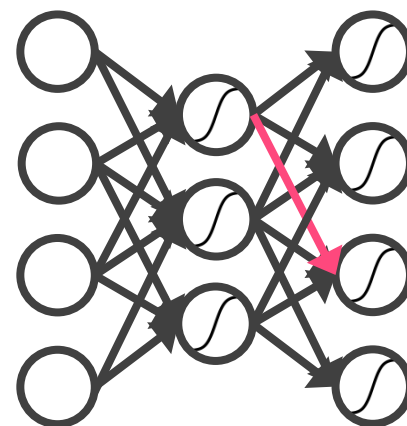
各層の結合の勾配

3層目の結合に関する勾配

$$\frac{\partial L}{\partial W_{ji}^{(3)}} = \frac{\partial L}{\partial u_j^{(3)}} \frac{\partial u_j^{(3)}}{\partial W_{ji}^{(3)}}$$

2層目の結合に関する勾配

$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \sum_k \frac{\partial L}{\partial u_k^{(3)}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \end{aligned}$$



NNの定式

$$u_j^{(l+1)} = \sum_i W_{ji}^{(l+1)} z_i^{(l)} + b_j^{(l+1)}$$

$$z_i^{(l)} = f(u_i^{(l)})$$

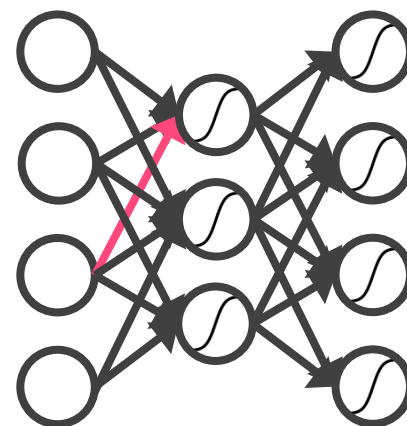
各層の結合の勾配

3層目の結合に関する勾配

$$\frac{\partial L}{\partial W_{ji}^{(3)}} = \frac{\partial L}{\partial u_j^{(3)}} \frac{\partial u_j^{(3)}}{\partial W_{ji}^{(3)}}$$

2層目の結合に関する勾配

$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \sum_k \frac{\partial L}{\partial u_k^{(3)}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \end{aligned}$$



各層の結合の勾配

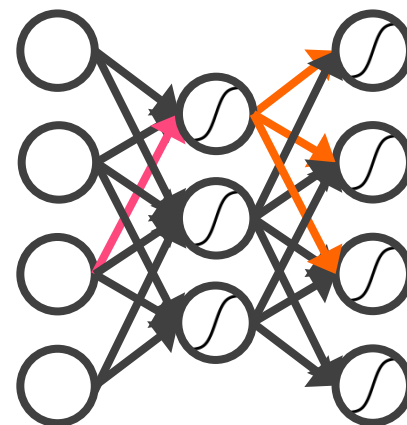
3層目の結合に関する勾配

$$\frac{\partial L}{\partial W_{ji}^{(3)}} = \frac{\partial L}{\partial u_j^{(3)}} \frac{\partial u_j^{(3)}}{\partial W_{ji}^{(3)}}$$

2層目の結合に関する勾配

$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \sum_k \frac{\partial L}{\partial u_k^{(3)}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \end{aligned}$$

流れ込む全てを考慮する



各層の結合の勾配

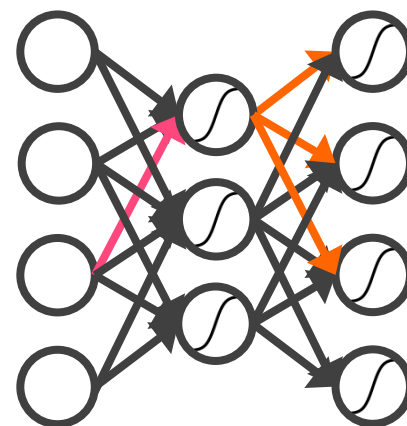
3層目の結合に関する勾配

$$\frac{\partial L}{\partial W_{ji}^{(3)}} = \boxed{\frac{\partial L}{\partial u_j^{(3)}}} \frac{\partial u_j^{(3)}}{\partial W_{ji}^{(3)}}$$

2層目の結合に関する勾配

$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \sum_k \boxed{\frac{\partial L}{\partial u_k^{(3)}}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \end{aligned}$$

共通要素が登場 $\Rightarrow \delta_k^{(3)}$ と定義



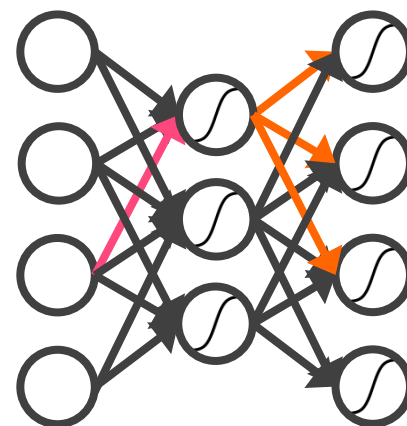
各層の結合の勾配

3層目の結合に関する勾配

$$\frac{\partial L}{\partial W_{ji}^{(3)}} = \boxed{\frac{\partial L}{\partial u_j^{(3)}}} \boxed{\frac{\partial u_j^{(3)}}{\partial W_{ji}^{(3)}}}$$

2層目の結合に関する勾配

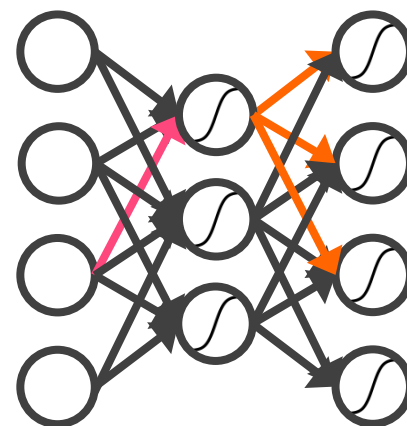
$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \boxed{\frac{\partial L}{\partial u_j^{(2)}}} \boxed{\frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}}} \\ &= \sum_k \frac{\partial L}{\partial u_k^{(3)}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \end{aligned}$$



各層の結合の勾配

3層目の結合に関する勾配

$$\frac{\partial L}{\partial W_{ji}^{(3)}} = \frac{\partial L}{\partial u_j^{(3)}} \frac{\partial u_j^{(3)}}{\partial W_{ji}^{(3)}}$$



2層目の結合に関する勾配

$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \sum_k \frac{\partial L}{\partial u_k^{(3)}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \end{aligned}$$

l 層目の W_{ji} に関する勾配が

$$\frac{\partial L}{\partial W_{ji}^{(l)}} = \delta_j^{(l)} z_i^{(l-1)}$$

で計算できそう！？

l 層目のパラメーターの勾配

W に関する勾配

$$\begin{aligned}\frac{\partial L}{\partial W_{ji}^{(l)}} &= \frac{\partial L}{\partial u_j^{(l)}} \frac{\partial u_j^{(l)}}{\partial W_{ji}^{(l)}} \\ &= \delta_j^{(l)} z_i^{(l-1)}\end{aligned}$$

b に関する勾配

$$\begin{aligned}\frac{\partial L}{\partial b_j^{(l)}} &= \frac{\partial L}{\partial u_j^{(l)}} \frac{\partial u_j^{(l)}}{\partial b_j^{(l)}} \\ &= \delta_j^{(l)}\end{aligned}$$

...残る謎は $\delta_j^{(l)}$ の計算方法

$\delta_j^{(l)}$ の計算

今 $\delta_j^{(l)} = \frac{\partial L}{\partial u_j^{(l)}}$ とする. ニューラルネットワークの定義

$$\begin{aligned} u_k^{(l+1)} &= \sum_j W_{kj}^{(l+1)} z_j^{(l)} + b_k^{(l+1)} \\ &= \sum_j W_{kj}^{(l+1)} f(u_j^{(l)}) + b_k^{(l+1)} \end{aligned}$$

を利用すれば

$$\frac{\partial u_k^{(l+1)}}{\partial u_j^{(l)}} = W_{kj}^{l+1} f'(u_j^{(l)})$$

より,

$$\begin{aligned} \delta_j^{(l)} &= \frac{\partial L}{\partial u_j^{(l)}} = \sum_k \frac{\partial L}{\partial u_k^{(l+1)}} \frac{\partial u_k^{(l+1)}}{\partial u_j^{(l)}} \\ &= \sum_k \delta_k^{(l+1)} W_{kj}^{l+1} f'(u_j^{(l)}) \end{aligned}$$

$\Rightarrow \delta_j^{(l)}$ が一つ上の層の $\delta_j^{(l+1)}$ から順番に求まる

行列表記

$$U, Z, \Delta = N_{\text{dim}} \begin{array}{c} N \\ \text{[matrix]} \end{array} \quad W = N_{\text{dim}_2} \begin{array}{c} N_{\text{dim}_1} \\ \text{[matrix]} \end{array} \quad b = N_{\text{dim}_2} \begin{array}{c} 1 \\ \text{[matrix]} \end{array}$$

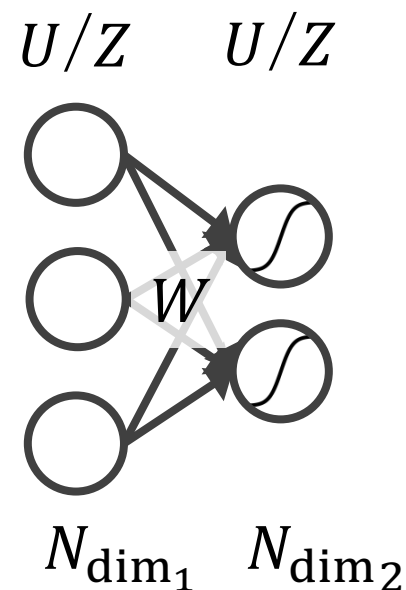
モデルの定式とデルタの更新則

$$U^{(l)} = W^{(l)} Z^{(l-1)} + \mathbf{b}^{(l)} \mathbb{1}_N^T$$

$$\Delta^{(l)} = f^{(l)'}(U^{(l)}) \odot \left(W^{(l+1)T} \Delta^{(l+1)} \right)$$

各パラメーターの勾配

$$\frac{\partial L}{\partial W^{(l)}} = \frac{1}{N} \Delta^{(l)} Z^{(l-1)T} \quad \frac{\partial L}{\partial \mathbf{b}^{(l)}} = \frac{1}{N} \Delta^{(l)} \mathbb{1}_N$$



解いてみよう2

目的関数が以下の**クロスエントロピー誤差**

$$L(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^K d_{nk} \log y_k(\mathbf{x}_n; \mathbf{w})$$

で表され, $y_k = z_k^{(3)}$ が以下の**ソフトマックス関数**

$$z_k^{(3)} = f^{(3)}(u_k^{(3)}) = \frac{\exp(u_k^{(3)})}{\sum_{j=1}^K \exp(u_j^{(3)})}$$

の時, $\delta_j^{(3)} = \frac{\partial L}{\partial u_j^{(3)}}$ が $\sum_{n=1}^N (y_j - d_{nj})$ となることを示せ.

ヒント1: $\frac{\partial L}{\partial u_j^{(3)}} = \sum_k \frac{\partial L}{\partial y_k} \frac{\partial y_k}{\partial u_j^{(3)}}$ の形に分解し, $k = j$ と $k \neq j$ に場合分けする

ヒント2: ソフトマックス関数の微分はソフトマックス関数自身が出てくることを利用する

確率的勾配降下法 (**SGD**)

$$U, Z, \Delta = N_{\text{dim}} \overset{N}{\boxed{\phantom{\text{matrix}}}}$$

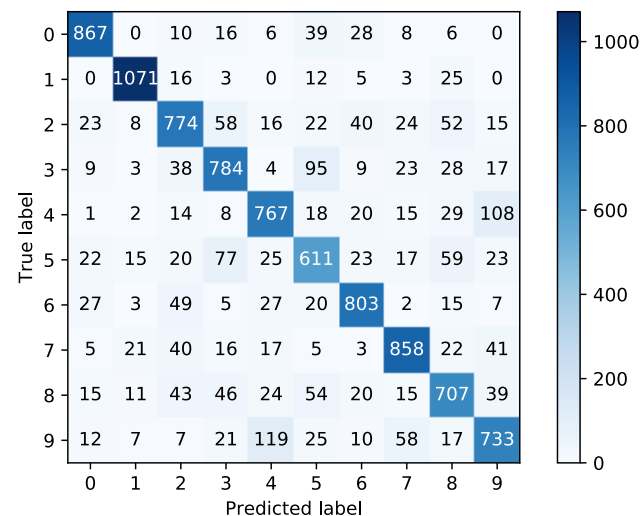
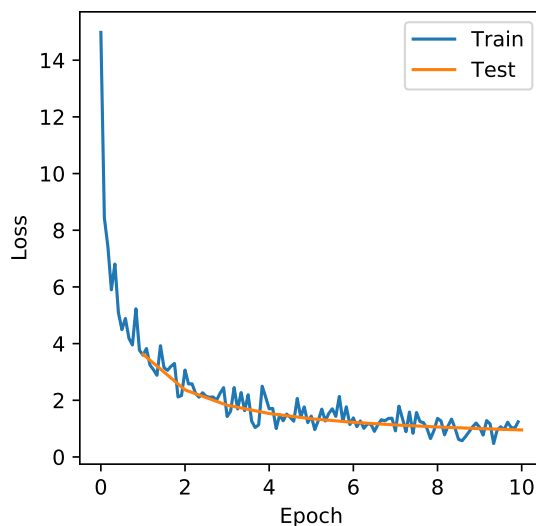
データが得られた元での真の勾配の計算はデータ数 N をサイズに持つ行列演算を**各更新ごと**に行う必要がある

e.g.)

- MNIST: $N=60,000$
- ImageNet: $N>400,000$

⇒ 全データ数 N ではなく、それよりも少ない数 N_B で勾配の近似値を計算する

実践演習



課題

1. **3層NNモデル**のクロスエントロピー誤差最小化を**ミニバッチ勾配降下法**で実装する
2. MNISTデータセットを用いて学習を行う

必要そうな式 in 行列表記 [1/2]

3層NNのモデル

$$Y(X) = Z^{(3)} = f^{(3)}\left(W^{(3)} f^{(2)}\left(W^{(2)} X + \mathbf{b}^{(2)} \mathbb{1}_N^T\right) + \mathbf{b}^{(3)} \mathbb{1}_N^T\right)$$

最終層の活性化関数 (ソフトマックス)

$$Y = Z^{(3)} = f^{(3)}(U^{(3)}) = \frac{\exp(U^{(3)})}{\mathbb{1}_{N_{\text{dim}}}^T \exp(U^{(3)})}$$

目的関数 (クロスエントロピー誤差)

$$L = -D \odot \log Y(X)$$

最終層の Δ

$$\Delta^{(3)} = Y - D$$

最終層以外の活性化関数 (シグモイド)

$$Z^{(l)} = f^{(l)}(U^{(l)}) = \frac{1}{1 + \exp(-U^{(l)})}$$

必要そうな式 in 行列表記 [2/2]

$$\begin{array}{c}
 N \\
 U, Z, \Delta = N_{\text{dim}} \begin{array}{|c|} \hline \\ \hline \end{array}
 \end{array}
 \quad
 \begin{array}{c}
 N_{\text{dim}_1} \\
 W = N_{\text{dim}_2} \begin{array}{|c|} \hline \\ \hline \end{array}
 \end{array}
 \quad
 \begin{array}{c}
 1 \\
 b = N_{\text{dim}_2} \begin{array}{|c|} \hline \\ \hline \end{array}
 \end{array}$$

各層の関係とデルタの更新則

$$U^{(l)} = W^{(l)} Z^{(l-1)} + \mathbf{b}^{(l)} \mathbb{1}_N^T$$

$$\Delta^{(l)} = f^{(l)'}(U^{(l)}) \odot \left(W^{(l+1)T} \Delta^{(l+1)} \right)$$

各パラメーターの勾配

$$\frac{\partial L}{\partial W^{(l)}} = \frac{1}{N} \Delta^{(l)} Z^{(l-1)T} \quad \frac{\partial L}{\partial \mathbf{b}^{(l)}} = \frac{1}{N} \Delta^{(l)} \mathbb{1}_N$$

