

ニコニコAIスクール「脳型人工知能開発者入門コース」#5

勾配法と誤差逆伝搬法

長野 祥大

2018/02/03

目指すゴール

本講義のコンセプト

9

モデルとプログラムの橋渡し

数式で記述されたモデルを、正しくプログラムとして
書き下す練習をする

(1) 数式の理解

数理的に何が起きて
いるか

(2) 実装の理解

数式をどのようにして
コードに書き起こすか

$$y = \sigma(W^T x)$$



```
F.Sigmoid(  
np.dot(w.T, x))
```

NIC02AI SCHOOL

[初回講義 by 八木さん]

NIC02AI SCHOOL

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 $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^N$ に対する線形モデル,

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

の最小二乗回帰を考える.

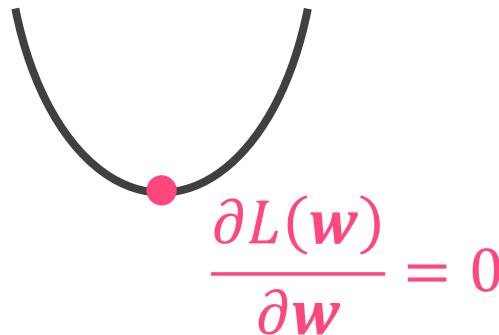
$$\min_{\mathbf{w}} L(\mathbf{w}) = \frac{1}{2} \|X\mathbf{w} - \mathbf{y}\|^2$$

2次関数の極小値は大域解となるので,

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

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

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



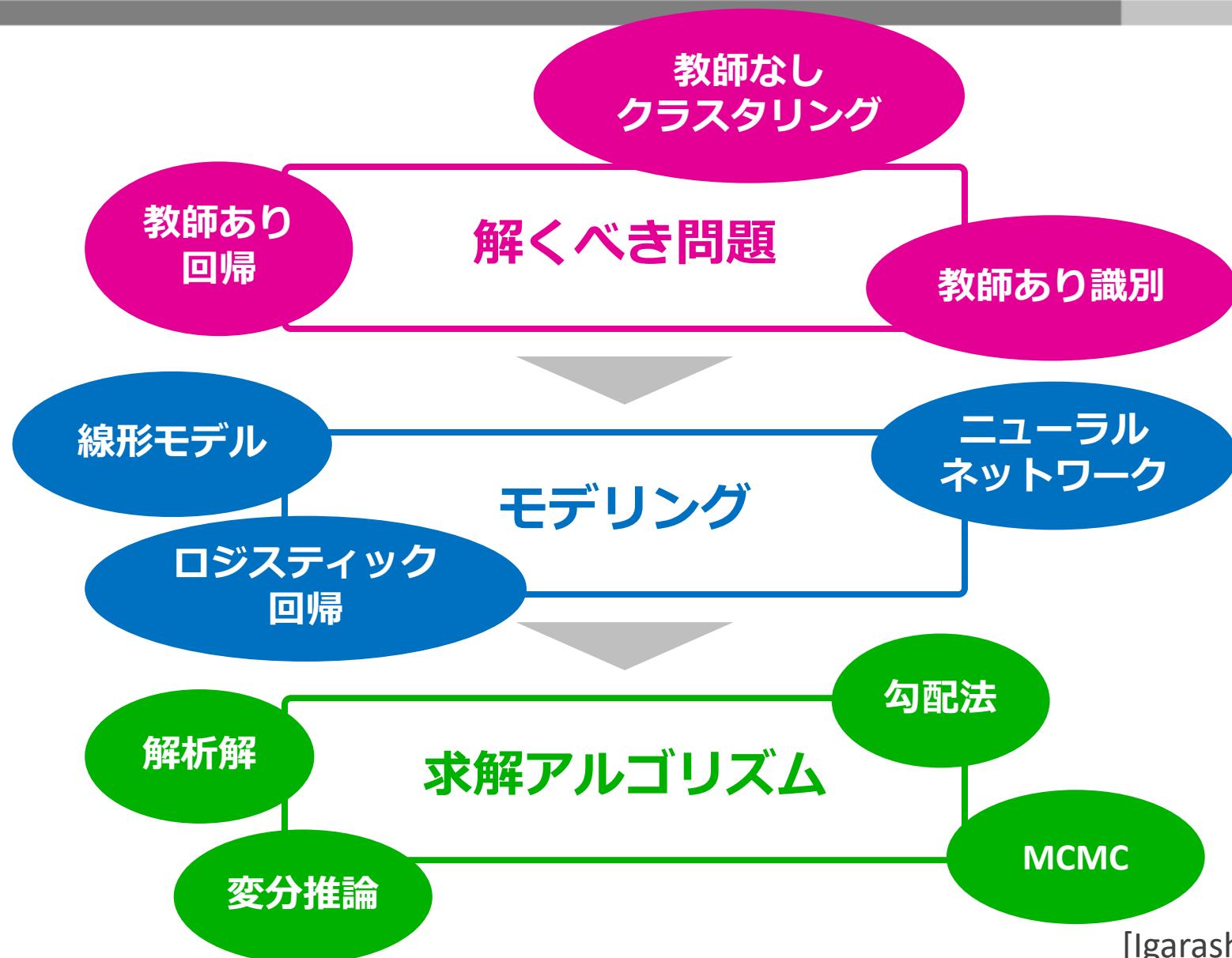
$$\begin{matrix} \mathbf{y} \\ = \\ X \\ \mathbf{w} \end{matrix}$$

解くべき問題

モデリング

求解アルゴリズム

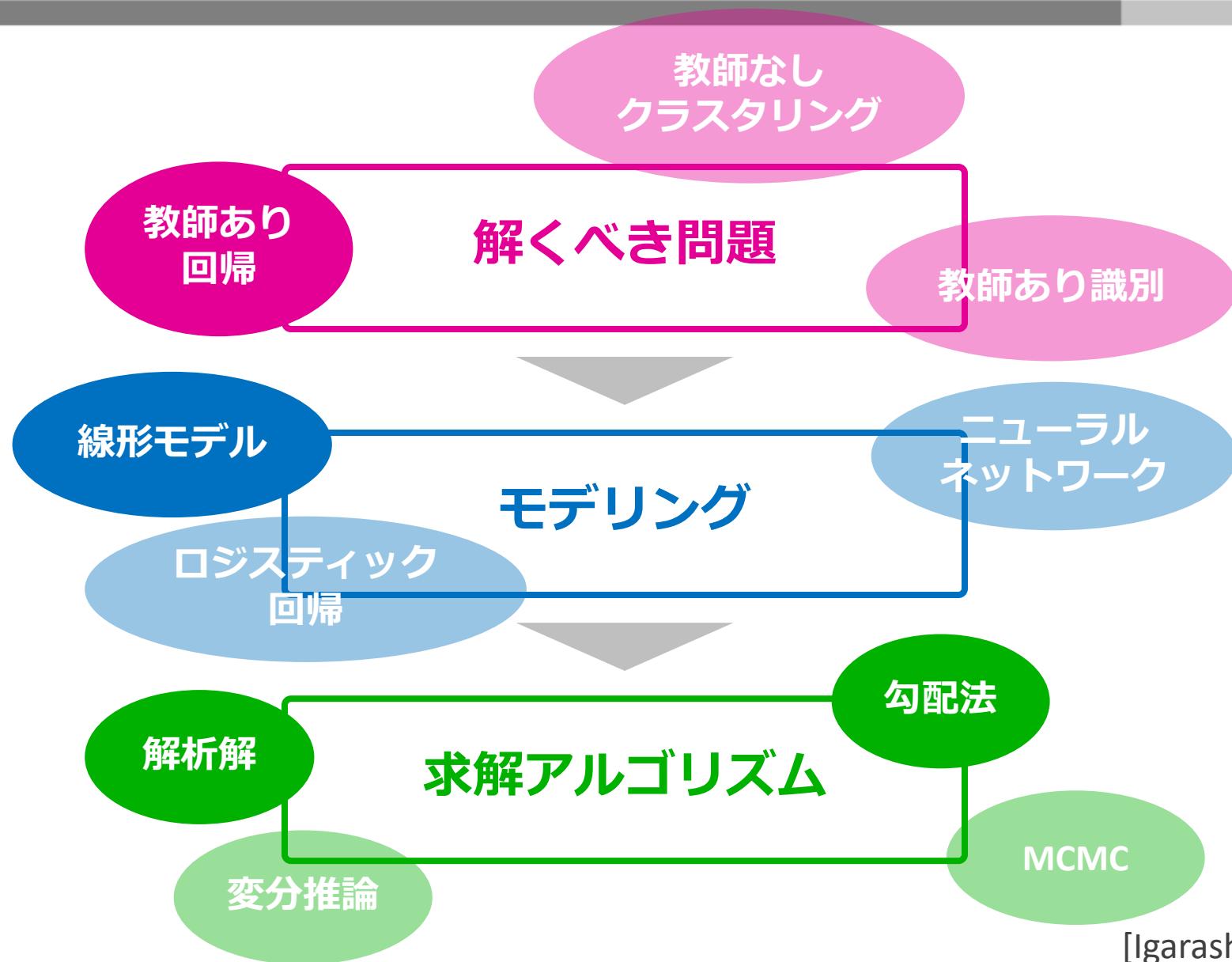
機械学習の階層性



[Igarashi+ 2016]

NICO2AISCHOOL

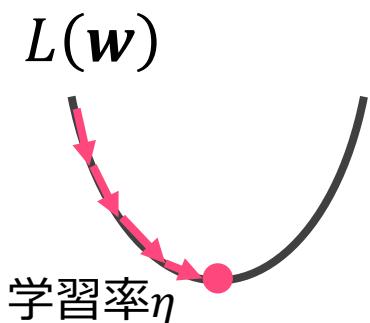
機械学習の階層性



線形モデルの二乗誤差最小化: 勾配法

for t in 1...T

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



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

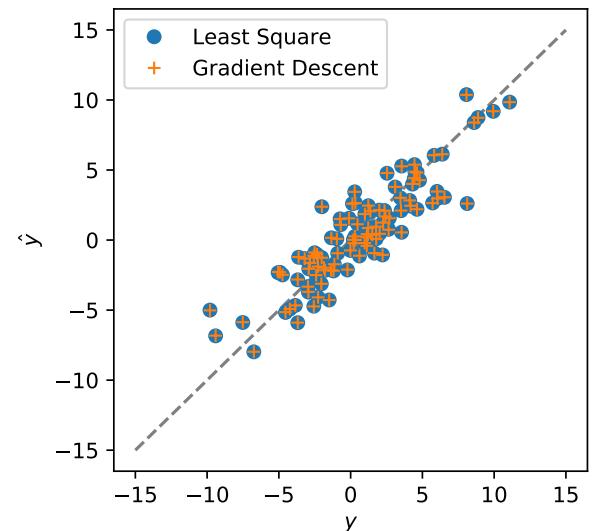
課題

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

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

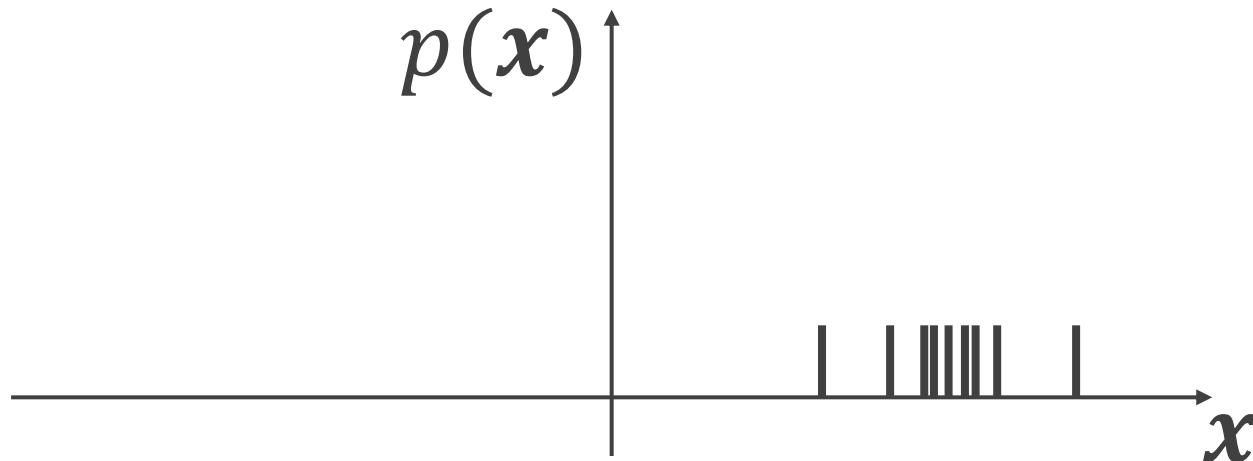
※2 データ数N: 100, 次元数D: 20



対数尤度

データ x に対するパラメーター $\theta = (\mu, \sigma)$ の対数尤度

$$\log p(x|\theta)$$



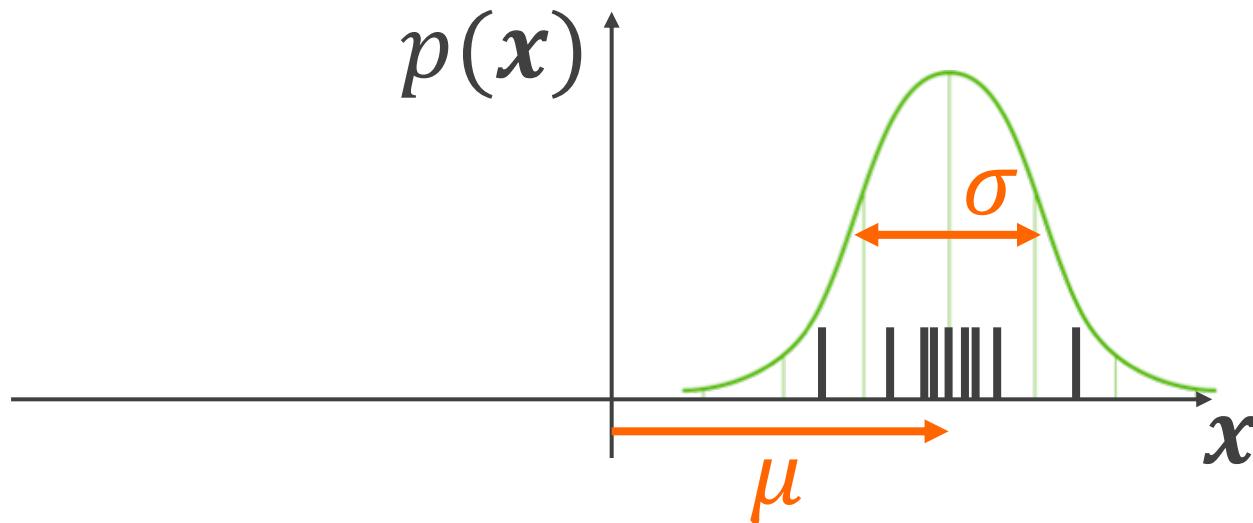
(対数)尤度は与えられたデータに対するモデルのもっともらしさ

⇒ (対数)尤度を最大化を最大化する学習を考える

対数尤度

データ x に対するパラメーター $\theta = (\mu, \sigma)$ の対数尤度

$$\log p(x|\theta)$$



(対数)尤度は与えられたデータに対するモデルのもっともらしさ

⇒ (対数)尤度を最大化を最大化する学習を考える

最小二乗法と最尤推定

データの生成モデルとして $y = \mathbf{w}^T \mathbf{x} + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$ を仮定する。この時, y の条件付き確率 $p(y|\mathbf{w}; \mathbf{x})$ はガウス分布 $\mathcal{N}(\mathbf{w}^T \mathbf{x}, \sigma^2)$ 対数尤度 $\log p(y|\mathbf{w}; \mathbf{x})$ の最大化を考えると,

$$\begin{aligned}\arg \max_{\mathbf{w}} \log p(y|\mathbf{w}; \mathbf{x}) &= \arg \max_{\mathbf{w}} \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y - \mathbf{w}^T \mathbf{x})^2}{2\sigma^2}\right) \\ &= \arg \max_{\mathbf{w}} \left[\log C + \log \exp\left(-\frac{(y - \mathbf{w}^T \mathbf{x})^2}{2\sigma^2}\right) \right] \\ &= \arg \max_{\mathbf{w}} -\frac{(y - \mathbf{w}^T \mathbf{x})^2}{2\sigma^2} \\ &= \arg \min_{\mathbf{w}} (y - \mathbf{w}^T \mathbf{x})^2\end{aligned}$$

となり, 最小二乗法が導出できる。(ここでノイズ分散 σ^2 は既知とした。)

L2正則化のベイズ的解釈

本当は対数尤度 $\log p(y|w; x)$ ではなくパラメーター w の(対数)事後分布 $\log p(w|y; x)$ を最大化したい。対数事後分布はベイズの定理から対数尤度と事前分布の和に分解できる。

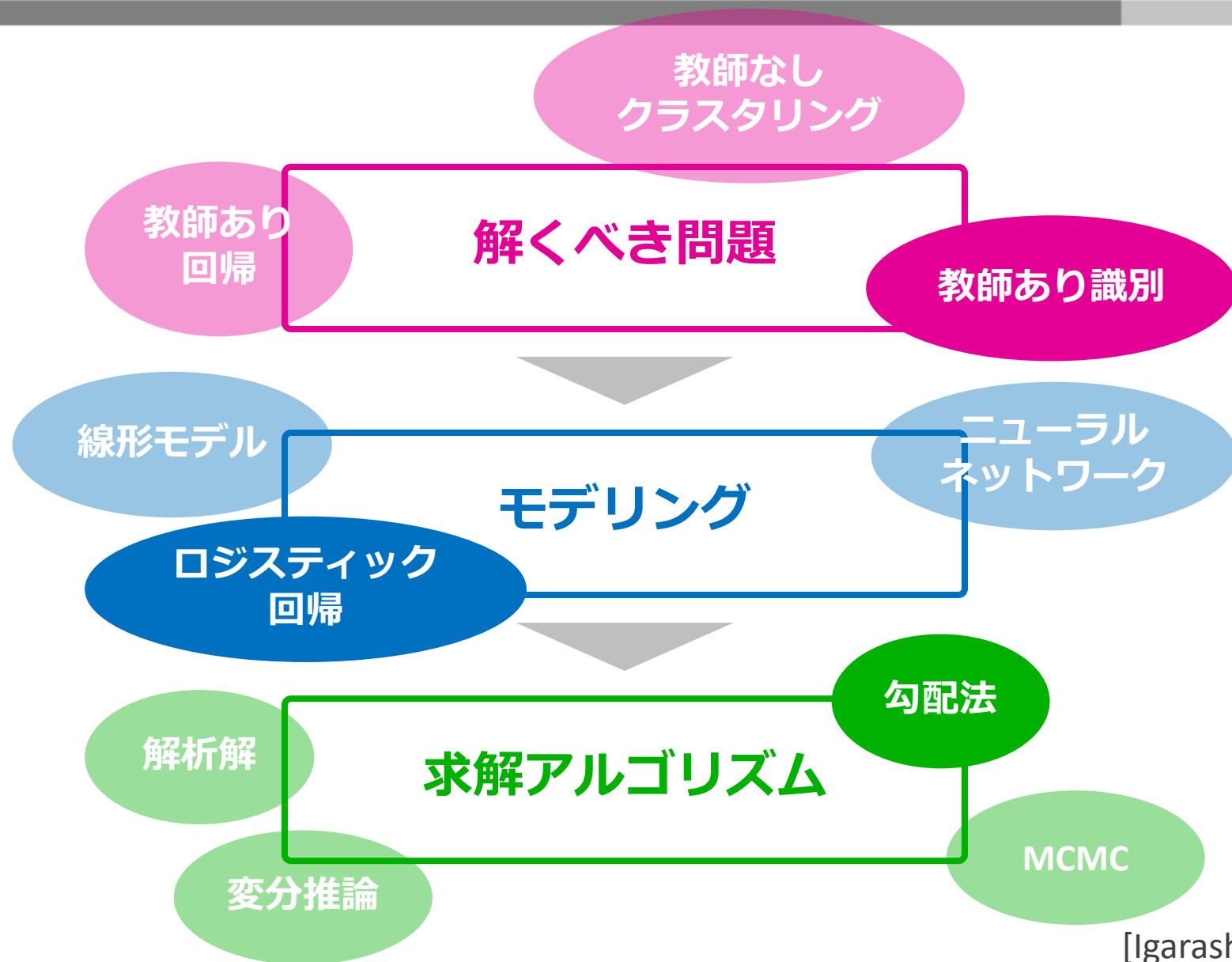
$$\log p(w|y; x) = \log \frac{p(y|w; x)p(w)}{p(y|x)} \propto \log p(y|w; x) + \log p(w)$$

事前分布 $p(w_i)$ にガウス分布 $\mathcal{N}(0, \tau^2)$ を仮定すると対数事後分布の最大化からL2正則化の目的関数

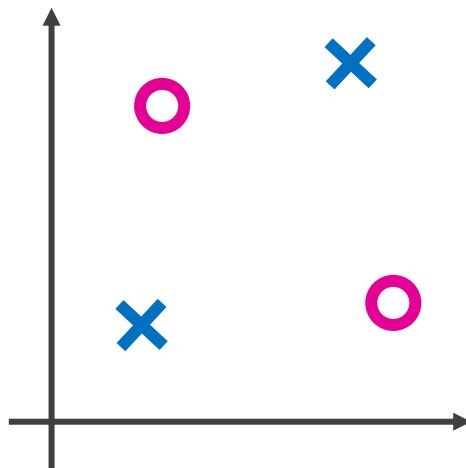
$$\arg \min_w L(w) = \arg \min_w \left\{ (y - w^T x)^2 + \lambda \|w\|^2 \right\}$$

が導出できる(ここで $\lambda = \sigma^2 / \tau^2$)。 (宿題)

→ L2正則化 = パラメーターの事前分布へのガウス分布の導入！



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

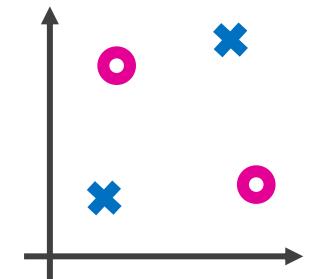
教師あり識別に用いられるモデル

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

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

$$\Pr(d = 1|x) \quad X \quad \mathbf{w}$$

logit  =  



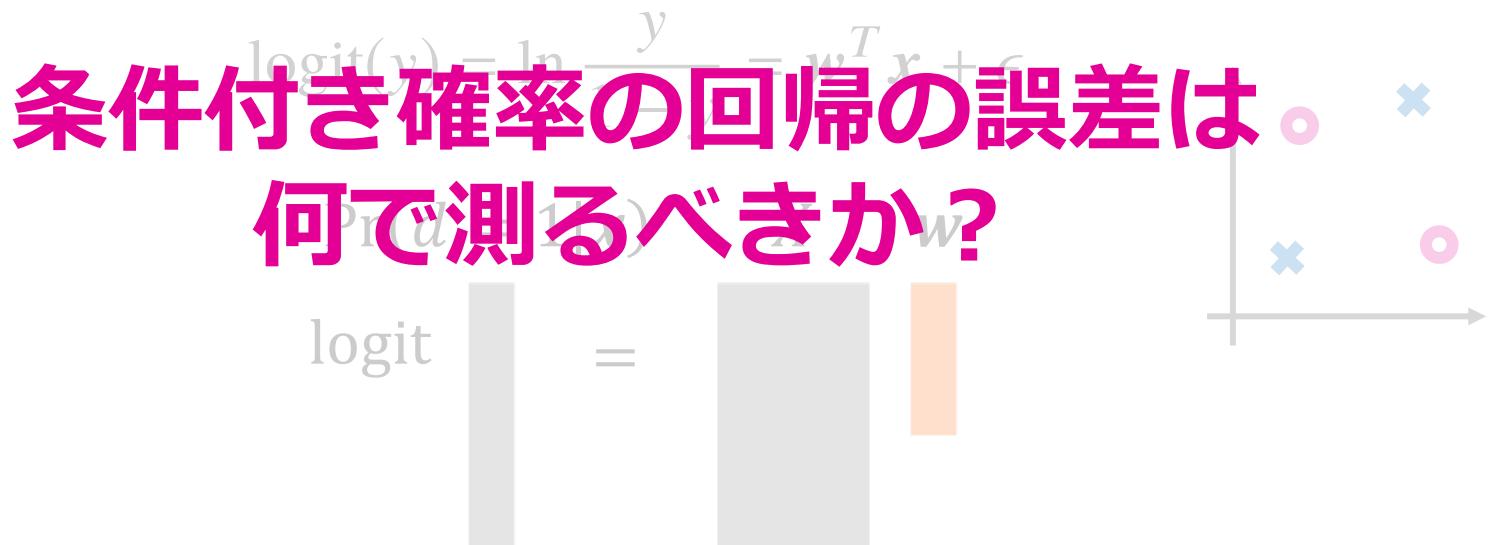
シグモイド関数 $1/(1 + e^{-ax})$ を用いて以下のようにも書き表せる

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

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 $\Pr(d = 1|x)$ の回帰 = 識別問題

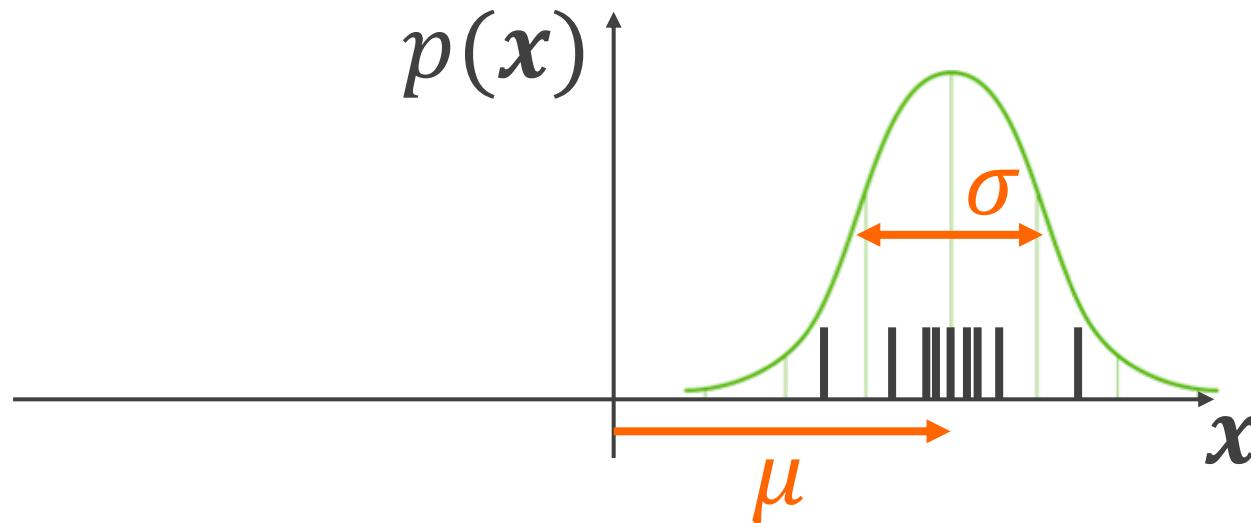


シグモイド関数 $1/(1 + e^{-ax})$ を用いて以下のようにも書き表せる

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

データ x に対するパラメーター $\theta = (\mu, \sigma)$ の対数尤度

$$\log p(x|\theta)$$



(対数)尤度は与えられたデータに対するモデルのもっともらしさ

⇒ (対数)尤度を最大化を最大化する学習を考える

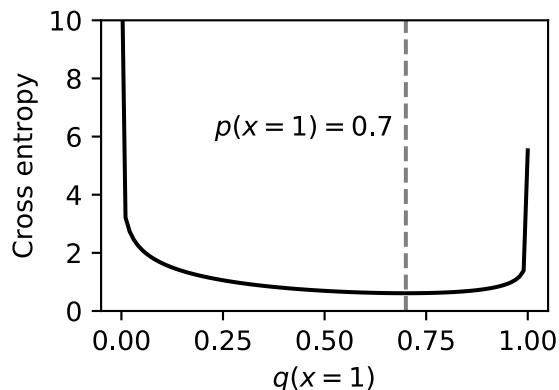
負の対数尤度の最小化

データ 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

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

条件付き確率のモデリング

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

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

勾配法による求解

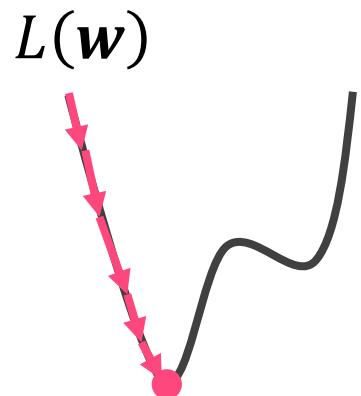
$$\begin{aligned} \frac{\partial L(w)}{\partial w} &= \frac{\partial H(y)}{\partial y} \frac{\partial y}{\partial w} \\ &= \frac{y - d}{y(1 - y)} \cdot \frac{\partial y}{\partial w} \\ &= \frac{y - d}{y(1 - y)} \cdot y(1 - y) \cdot x \\ &= (y - d)x \end{aligned}$$

シグモイド関数の微分

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

for t in 1...T

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

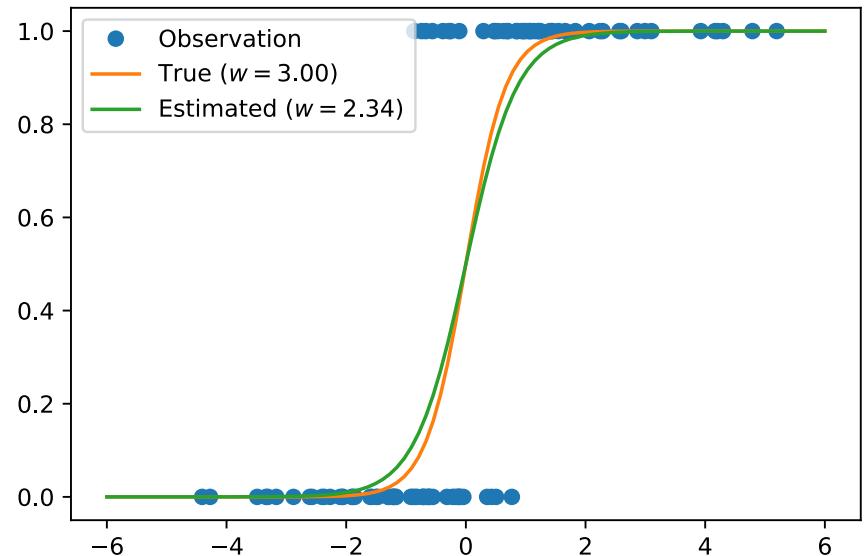


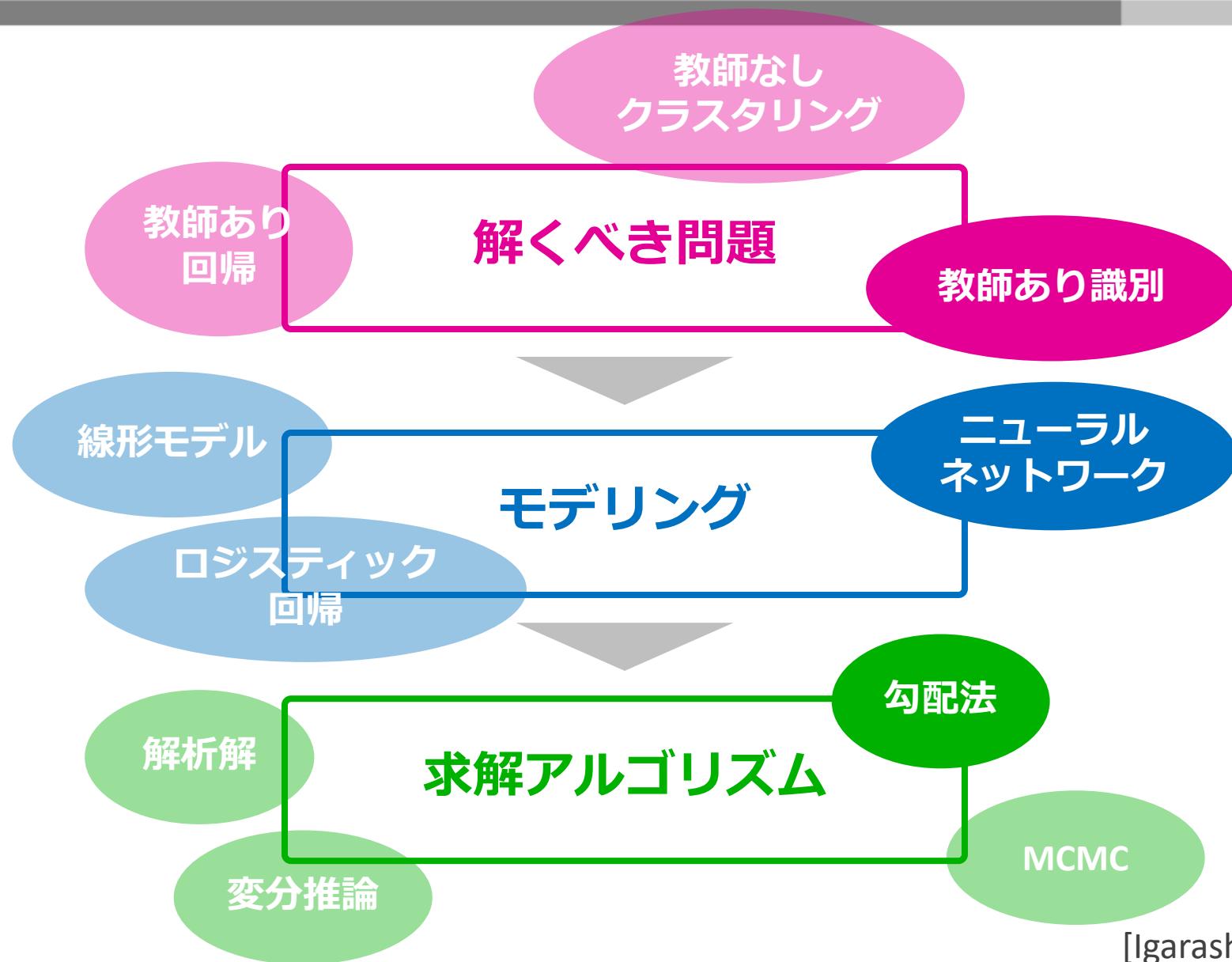
誤差関数の形は一見してよくわからないが、
勾配を計算することで(局所)解が求まる！

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

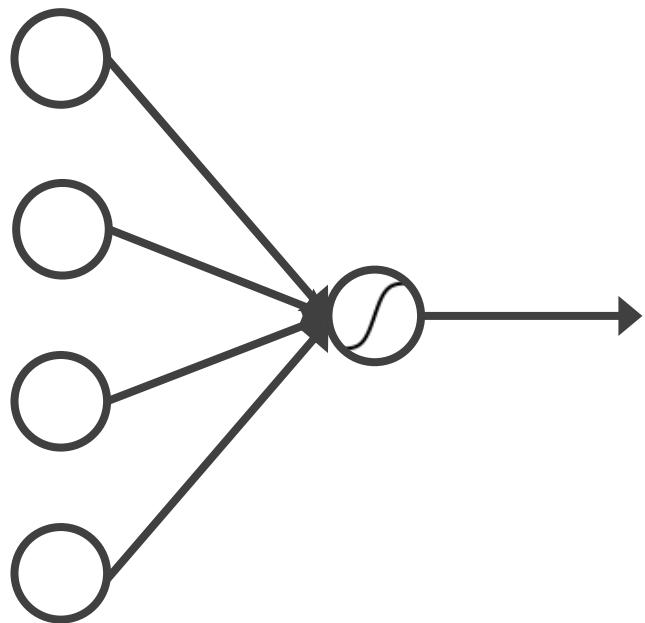




線形写像と非線形活性化関数からなる計算グラフ

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

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



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

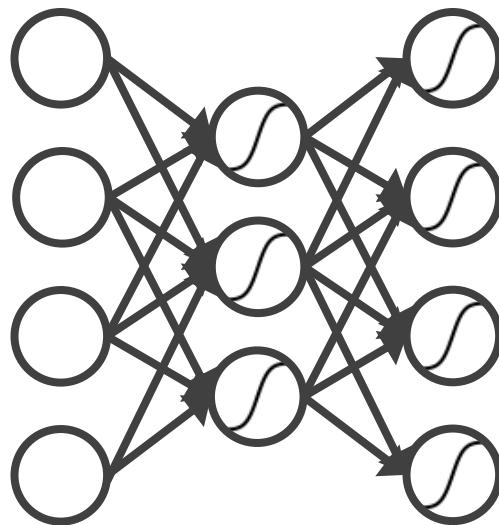
最適化するパラメーターは W と \mathbf{b}

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

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

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



これまでのモデルとは異なり、パラメーター(W^l, 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

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

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。

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

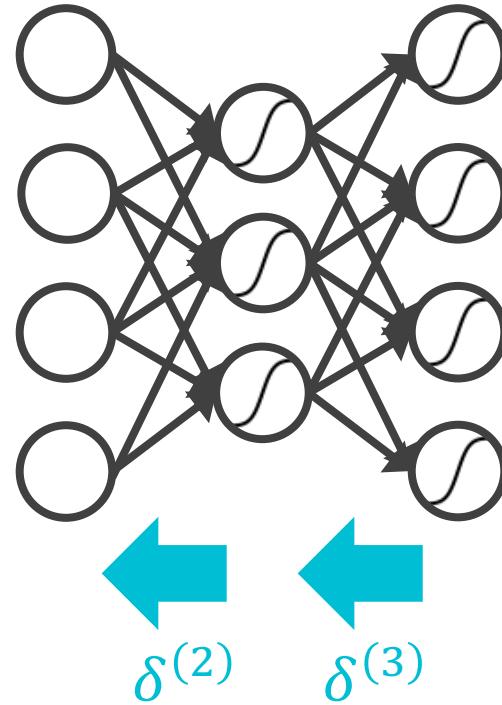
条件付き確率のモデリング

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

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

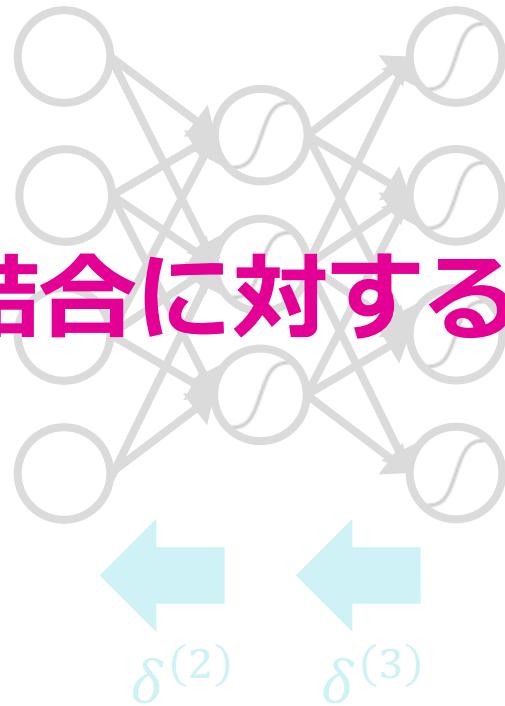
$$L(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^K d_{nk} \log y_k(\mathbf{x}_n; \mathbf{w})$$

⇒ 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？



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

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

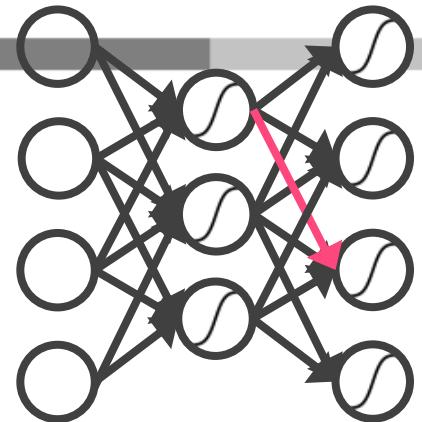
各層の結合の勾配

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

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



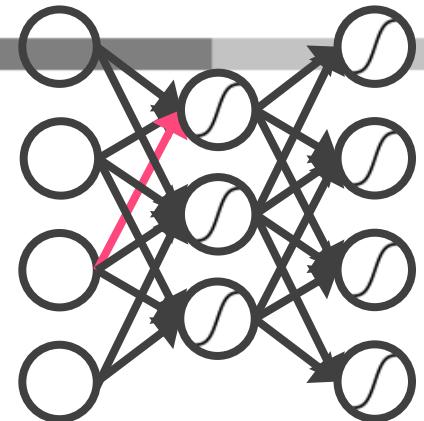
NNの定式

$$\begin{aligned} u_j^{(l+1)} &= \sum_i W_{ji}^{(l+1)} z_i^{(l)} + b_j^{(l+1)} \\ z_i^{(l)} &= f(u_i^{(l)}) \end{aligned}$$

各層の結合の勾配

3層目の結合に関する勾配

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



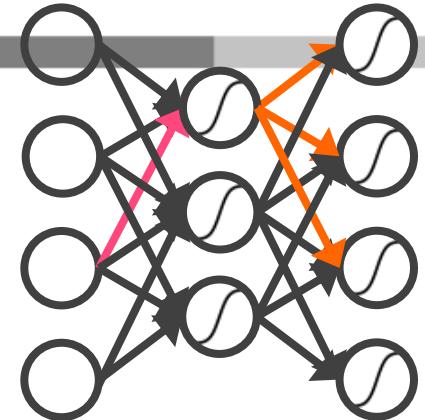
2層目の結合に関する勾配

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

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

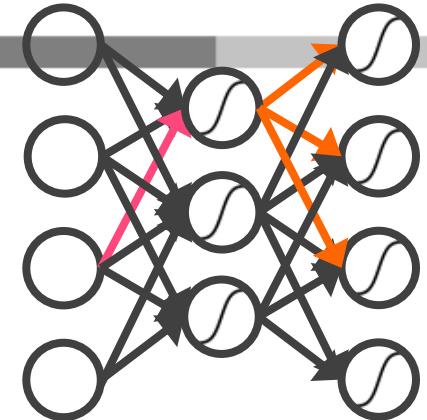
$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \boxed{\sum_k \frac{\partial L}{\partial u_k^{(3)}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}}}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

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

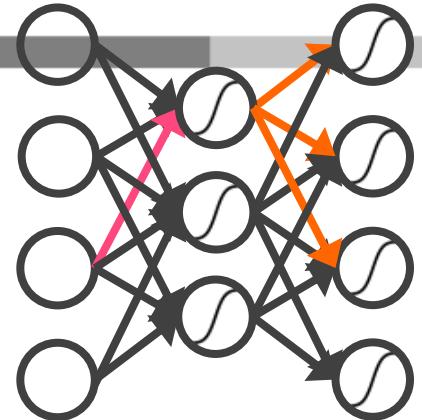
$$\begin{aligned} \frac{\partial L}{\partial W_{ji}^{(2)}} &= \frac{\partial L}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}} \\ &= \sum_k \boxed{\frac{\partial L}{\partial u_k^{(3)}}} \frac{\partial u_k^{(3)}}{\partial u_j^{(2)}} \frac{\partial u_j^{(2)}}{\partial W_{ji}^{(2)}}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

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

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

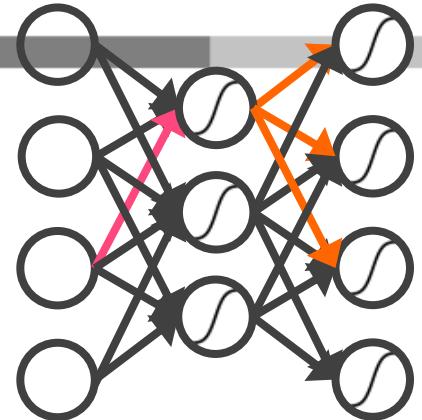
各層の結合の勾配

3層目の結合に関する勾配

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

2層目の結合に関する勾配

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



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！？

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、

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

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

行列表記

	N	N_{dim_1}	1
$U, Z, \Delta = N_{\text{dim}}$		$W = N_{\text{dim}_2}$	

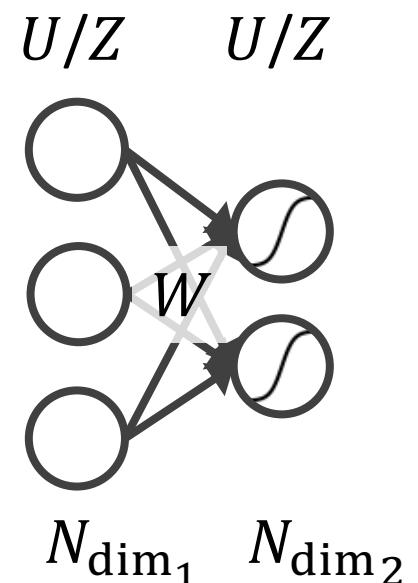
モデルの定式とデルタの更新則

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

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

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)

N

$U, Z, \Delta = N_{\text{dim}}$

データが得られた元での真の勾配の計算はデータ数 N をサイズを持つ行列演算を**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

確率的勾配降下法 (SGD)

1. 入力: データ $\{(x_i, y_i)\}_{i=1}^N = (X, Y)$, モデル $y = f(x; w)$
2. 重みパラメーター $\{(W^{(l)}, b^{(l)})\}_{l=1}^L$ を初期化
3. 学習率 η を設定

for t in 1...T

 for b in 1...B

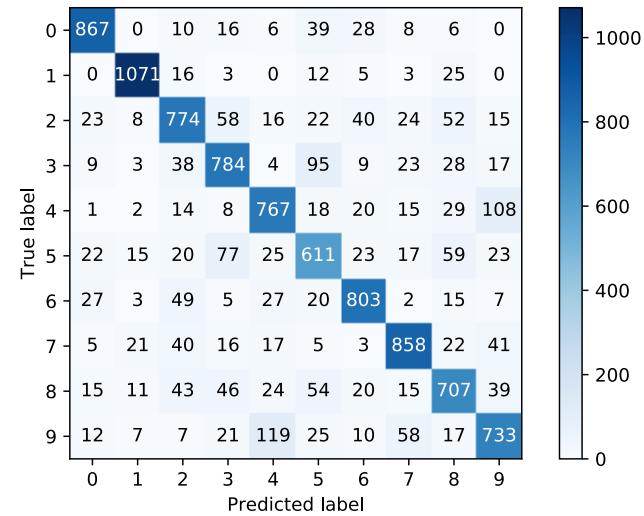
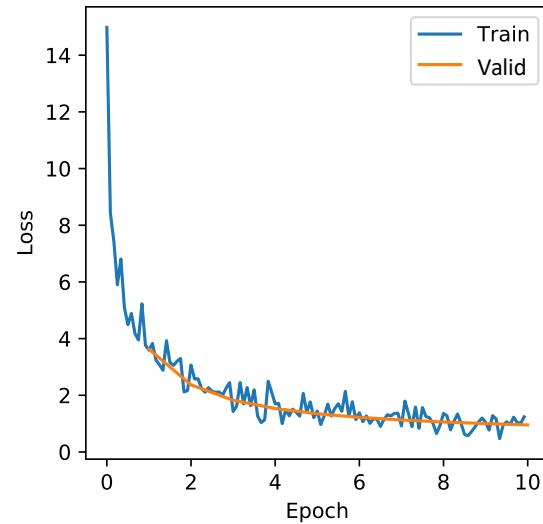
4. N_B 個のデータ $(x_{bN_B:(b+1)N_B}, y_{bN_B:(b+1)N_B})$ をランダムに選択する
5. $\{(U^{(l)}, Z^{(l)}, \Delta^{(l)})\}_{l=1}^L$ を計算する
6. パラメーターに関する勾配を計算

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

7. パラメーターを更新

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{1}{N_B} \Delta^{(l)} Z^{(l-1)T}, \quad b^{(l)} \leftarrow \eta \frac{1}{N_B} \Delta^{(l)} \mathbb{1}_N$$

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
二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

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

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]

N_B	N_{\dim_1}	1
$U, Z, \Delta = N_{\dim}$	$W = N_{\dim_2}$	$b = N_{\dim_2}$

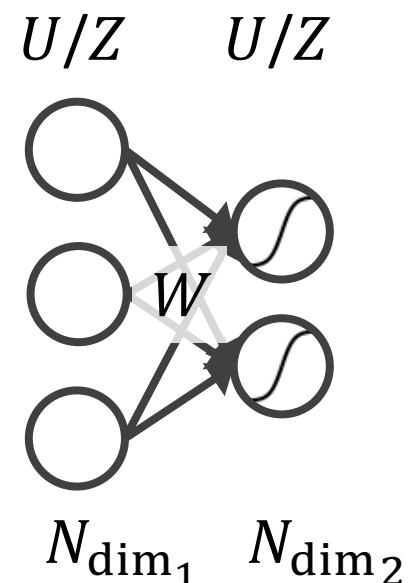
各層の関係とデルタの更新則

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

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

各パラメーターの勾配

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



参考書籍

機械学習プロフェッショナルシリーズ

『深層学習』岡谷貴之 著

講談社 (2015/4/7)

amzn.to/2E7Q1C9



参考スライド

数式をnumpyに落としこむコツ

中谷 秀洋

bit.ly/2ACaJqz

CybozuLabs

数式を numpy に落としこむコツ
～機械学習を題材に～

2011/10/15

中谷 秀洋@サイボウズ・ラボ
@shuyo / id:n_shuyo