

知能工学特別講義 第4講

担当：和田山 正

名古屋工業大学

本講義の内容

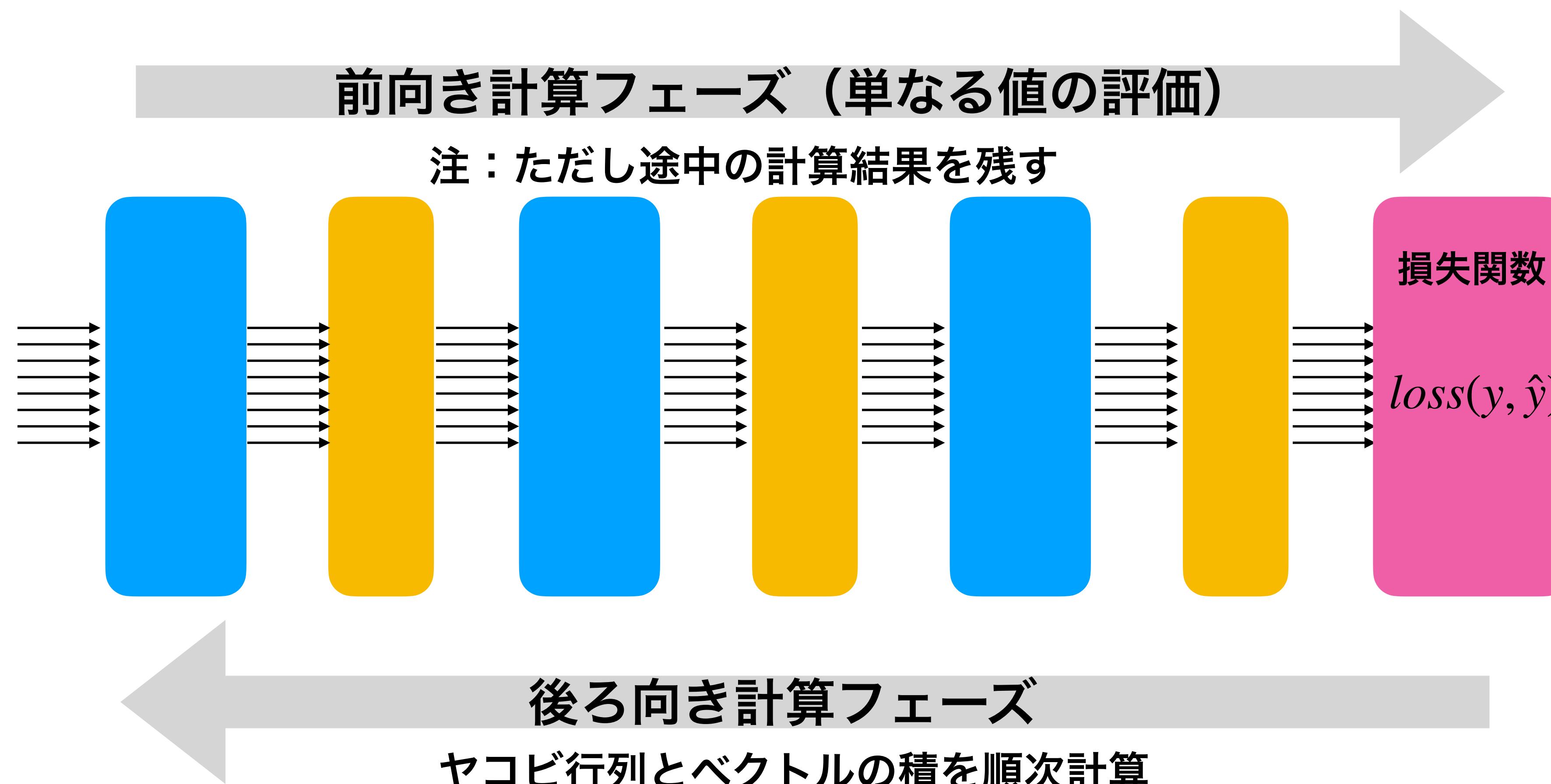
- 誤差逆伝播法の原理
- 回帰問題における予測モデル

誤差逆伝播法の原理



誤差逆伝播法(backprop)

- ・パラメータの勾配ベクトルを効率良く求めることが目的

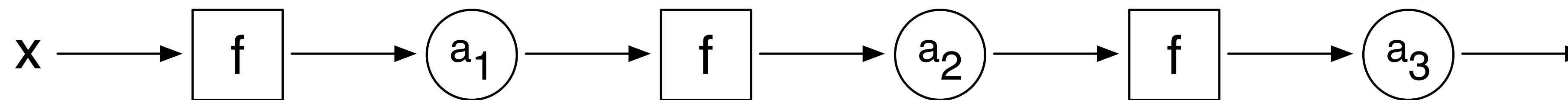


学習パラメータの勾配計算

極端に簡単化した順方向ネットワークを考える。

$$F(x; A) = a_3 f(a_2 f(a_1 f(x)))$$

という関数が与えられている。ここで、 $A = \{a_1, a_2, a_3\}$ である。



$$\frac{\partial F(x; A)}{\partial a_1}$$

を求めよ。

合成関数の微分

$$y = f(g(x))$$

について、 $\frac{\partial y}{\partial x}$ を求めたい。このとき、まず

$$\begin{aligned} u &= g(x) \\ y &= f(u) \end{aligned}$$

と分ける。このとき、合成関数の微分公式は(微分の連鎖律)は次のとおり:

$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial u} \times \frac{\partial u}{\partial x}$$

連鎖律をもとに計算してみる

$F(x; A) = a_3 f(a_2 f(a_1 f(x)))$ を

$$u_1 = a_1 f(x)$$

$$u_2 = a_2 f(u_1)$$

$$u_3 = a_3 f(u_2)$$

と書き換える。ここで、連鎖律を使うと

$$\frac{\partial u_3}{\partial a_1} = \frac{\partial u_3}{\partial u_2} \times \frac{\partial u_2}{\partial u_1} \times \frac{\partial u_1}{\partial a_1}$$

である。ここで、

$$\frac{\partial u_1}{\partial a_1} = f(x), \quad \frac{\partial u_2}{\partial u_1} = a_2 f'(u_1), \quad \frac{\partial u_3}{\partial u_2} = a_3 f'(u_2)$$

より、

$$\frac{\partial u_3}{\partial a_1} = a_3 f'(u_2) a_2 f'(u_1) f(x)$$

を得る。

前向き・後ろ向き計算

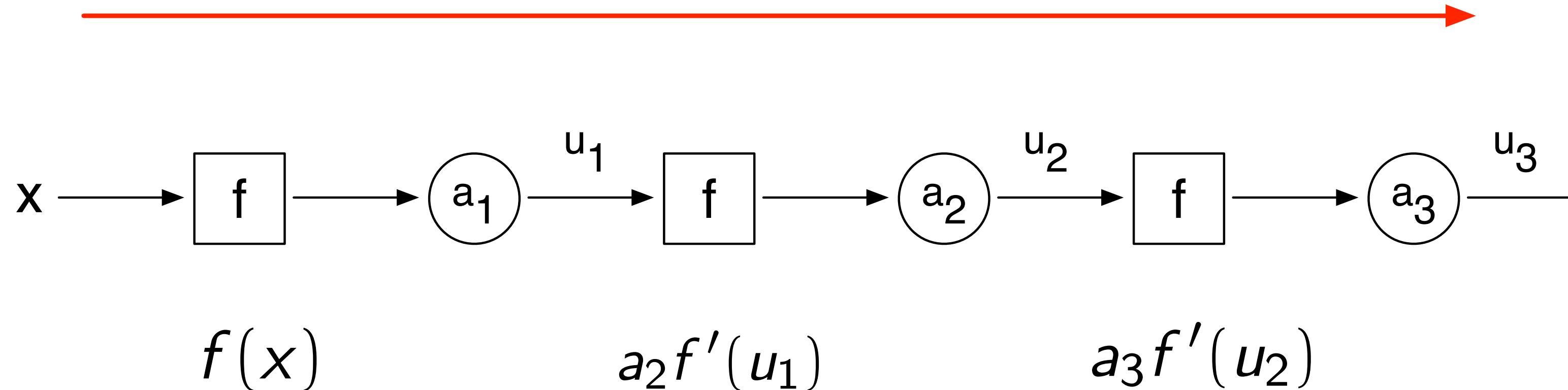
$$u_1 = a_1 f(x)$$

$$u_2 = a_2 f(u_1)$$

$$u_3 = a_3 f(u_2)$$

$$\frac{\partial u_3}{\partial a_1} = a_3 f'(u_2) a_2 f'(u_1) f(x)$$

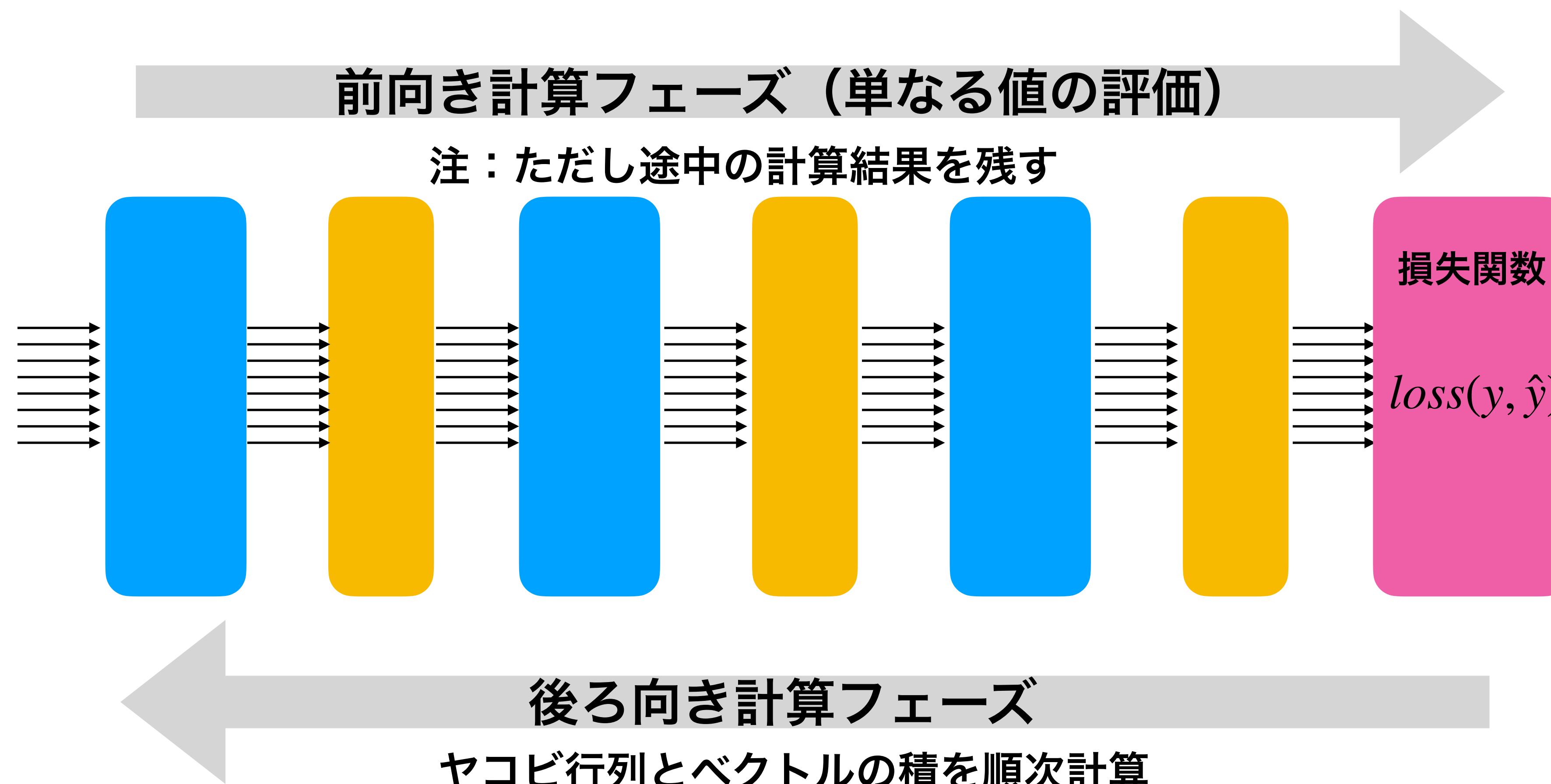
前向き計算



後ろ向き計算

誤差逆伝播法(backprop)

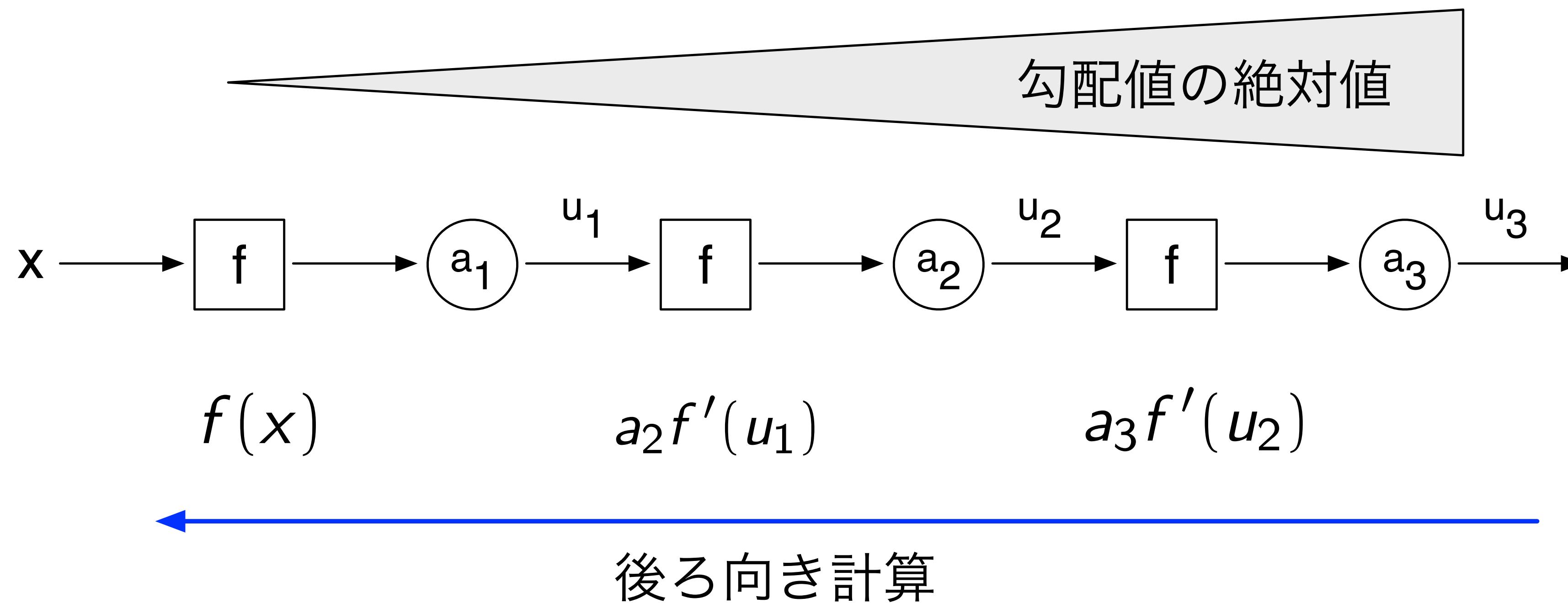
- ・パラメータの勾配ベクトルを効率良く求めることが目的



勾配消失問題

勾配消失問題とは、入力層に近い層の勾配値がほとんどゼロになってしまい、入力層に近い部分に位置する学習パラメータの更新がほとんど進まない現象を指す。

$$\frac{\partial u_3}{\partial a_1} = a_3 f'(u_2) a_2 f'(u_1) f(x)$$



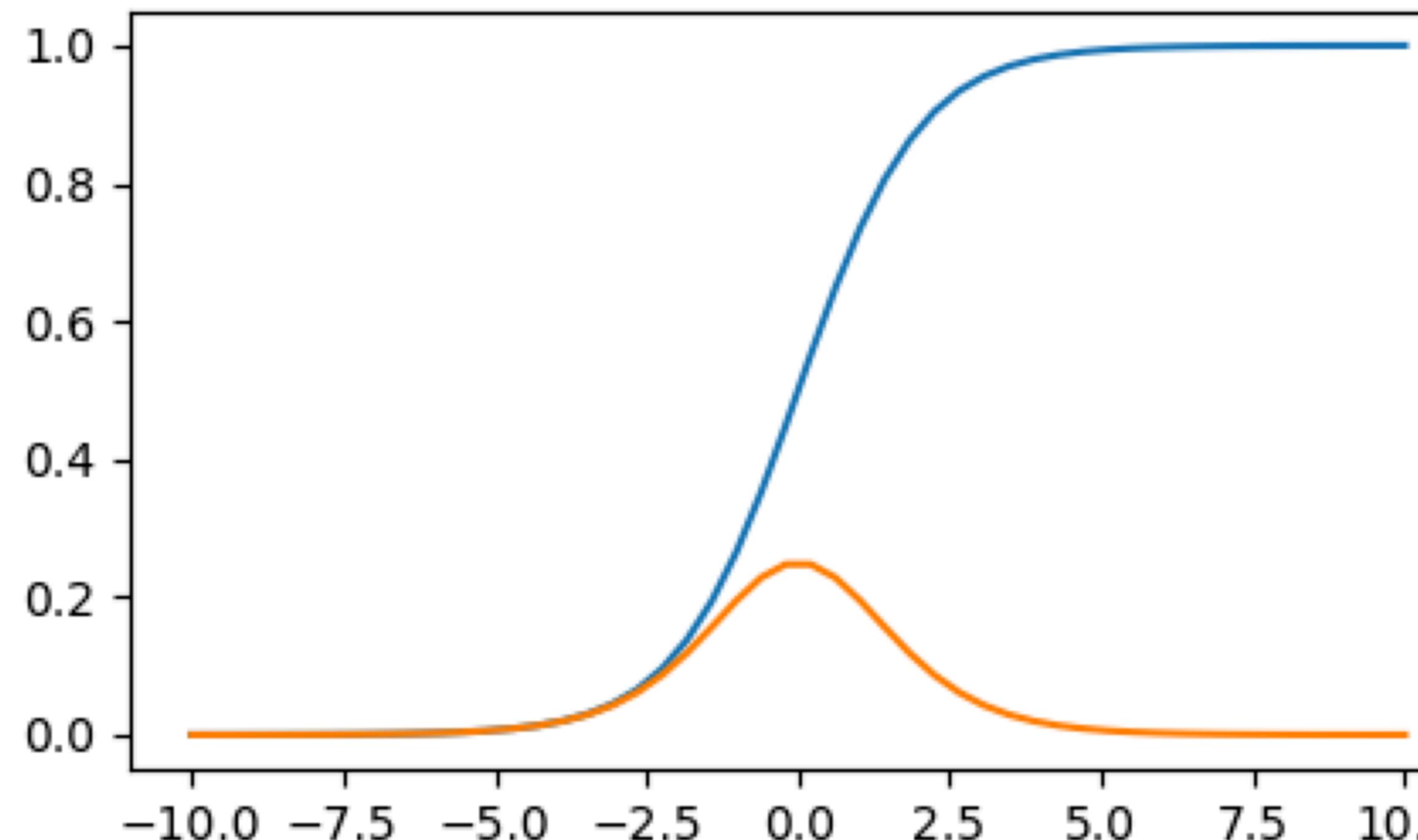
シグモイド関数とその微分

シグモイド関数

$$f(x) = \frac{1}{1 + e^{-x}}$$

シグモイド関数の導関数

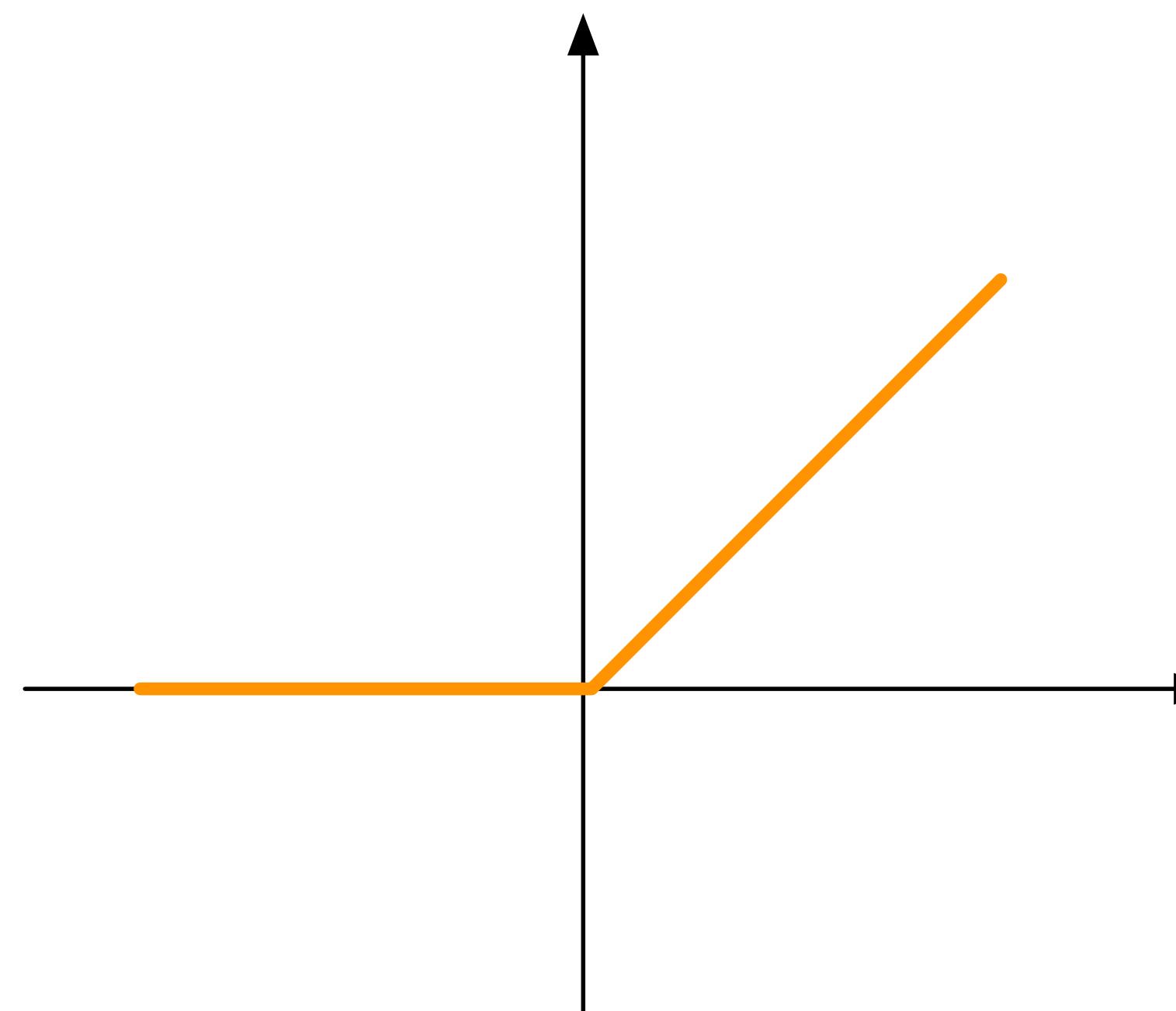
$$f'(x) = (1 - f(x))f(x)$$



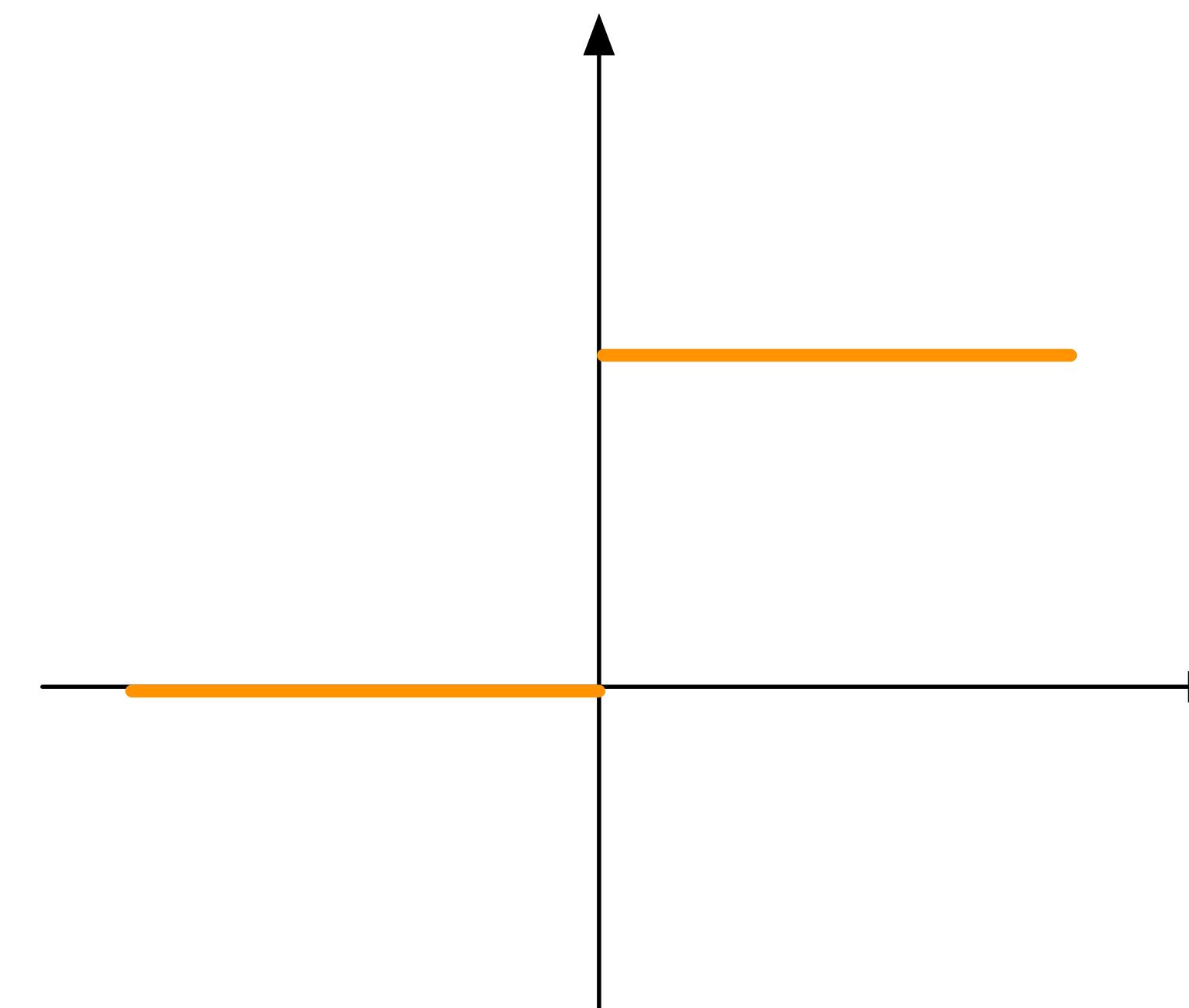
勾配消失問題への対策 (1)

ReLU 関数 (ランプ関数) の利用

ReLU関数



ReLU関数の導関数

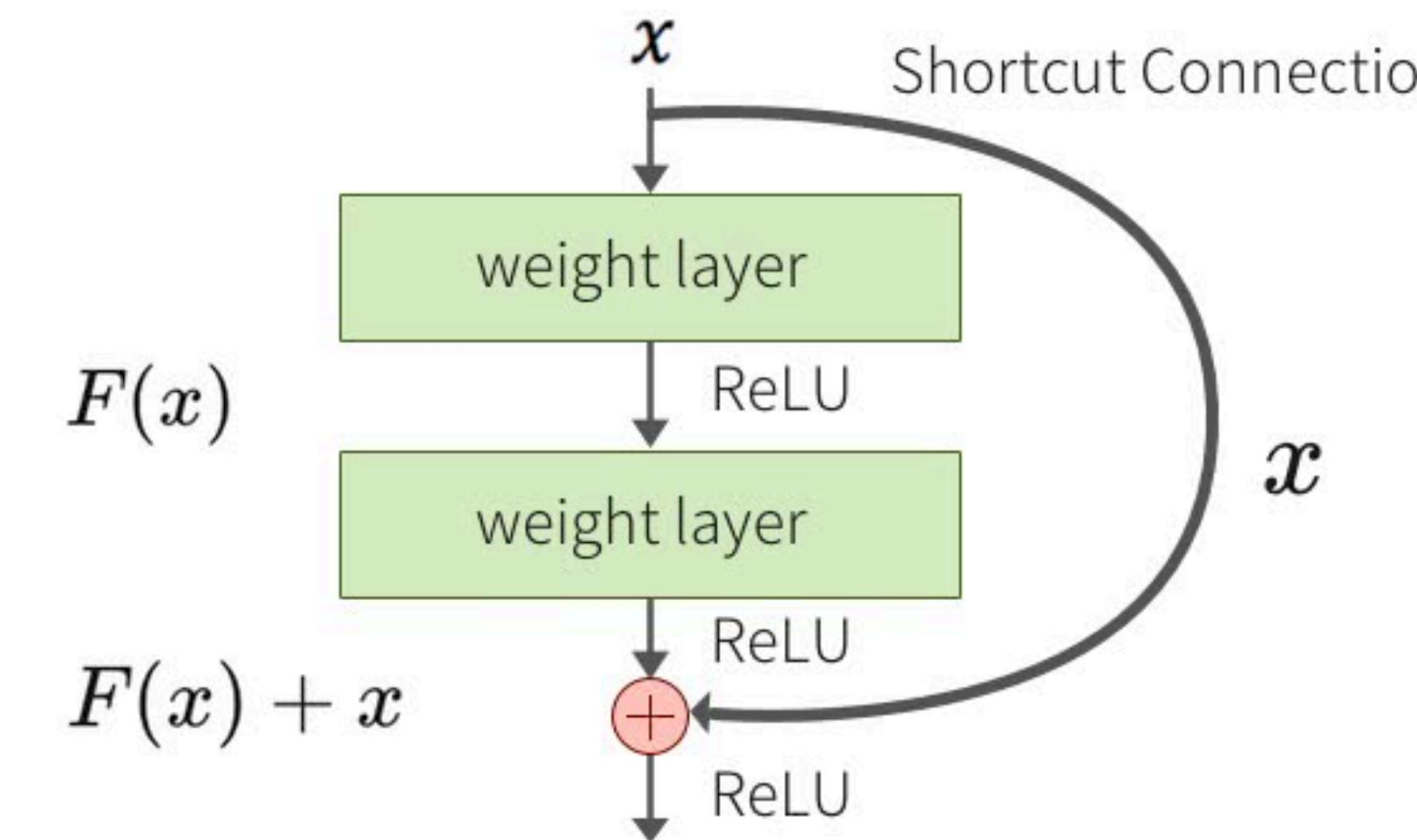


逆伝播フェーズにおける積の値が急速に小さくなることはなくなる。
→多層 (深層) の学習が容易になる。

勾配消失問題への対策(2)

Residual network の利用

ある層で求める最適な出力を学習するのではなく、層の入力を参照した残差関数を学習する、 という考え方に基づく。

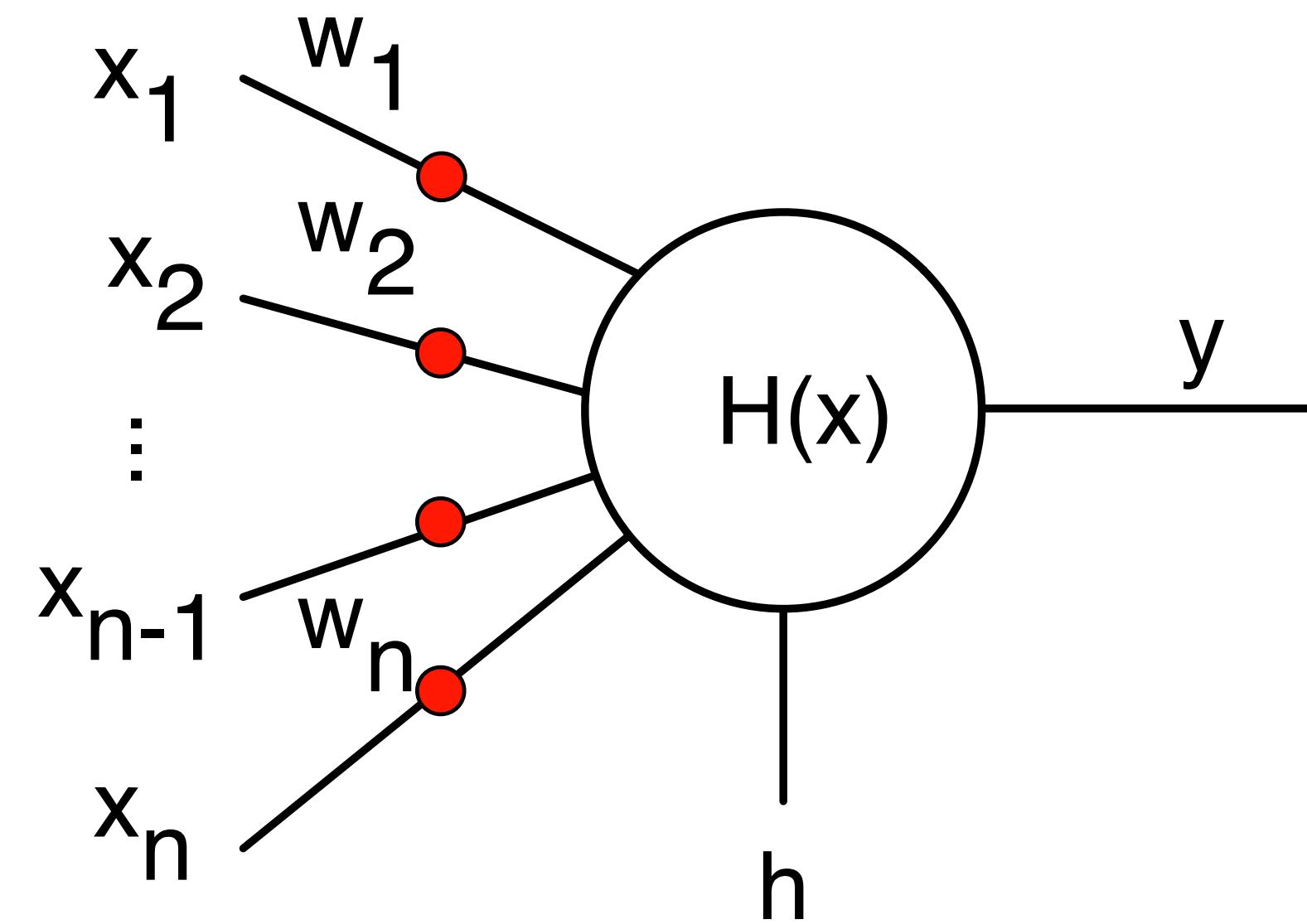
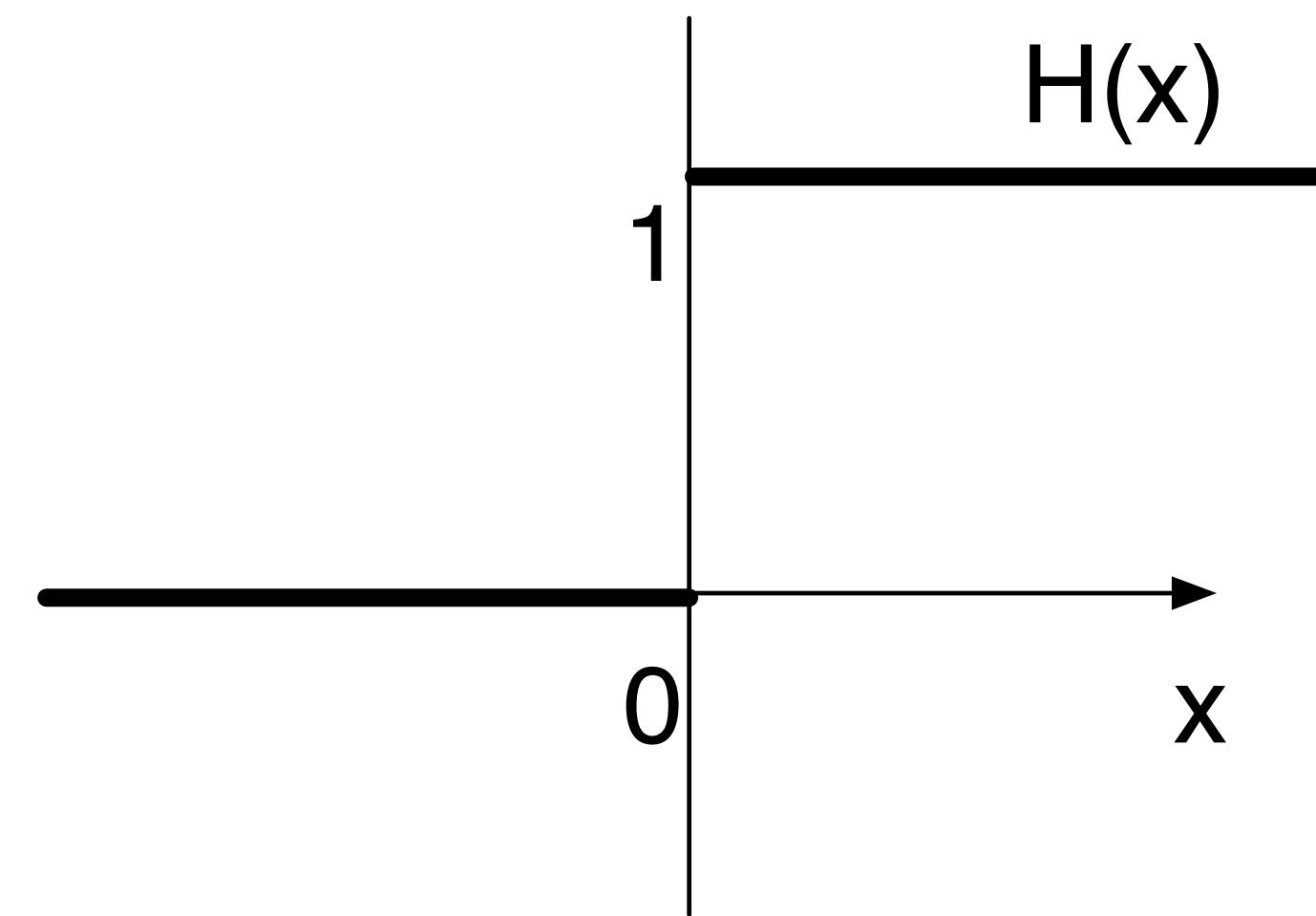


出典 https://deeppage.net/deep_learning/2016/11/30/resnet.html

直感的説明: 逆伝播フェーズにおいて、ショートカットパスに沿って絶対値の大きな勾配情報が流れる

パーセプトロンは勾配法による学習ができない

形式ニューロン

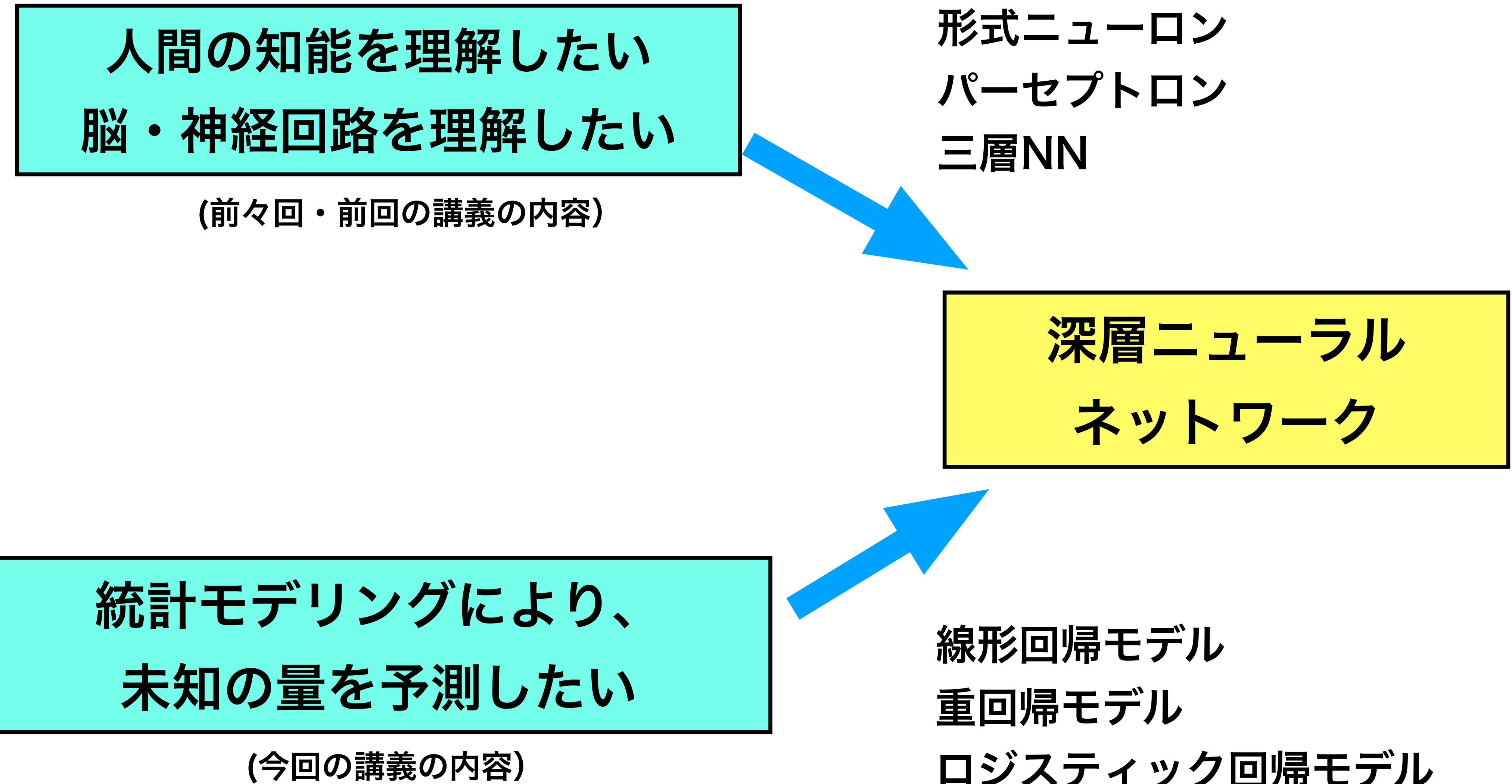


Q. なぜだろうか？

回帰問題における予測モデル



深層ニューラルネットに至る道



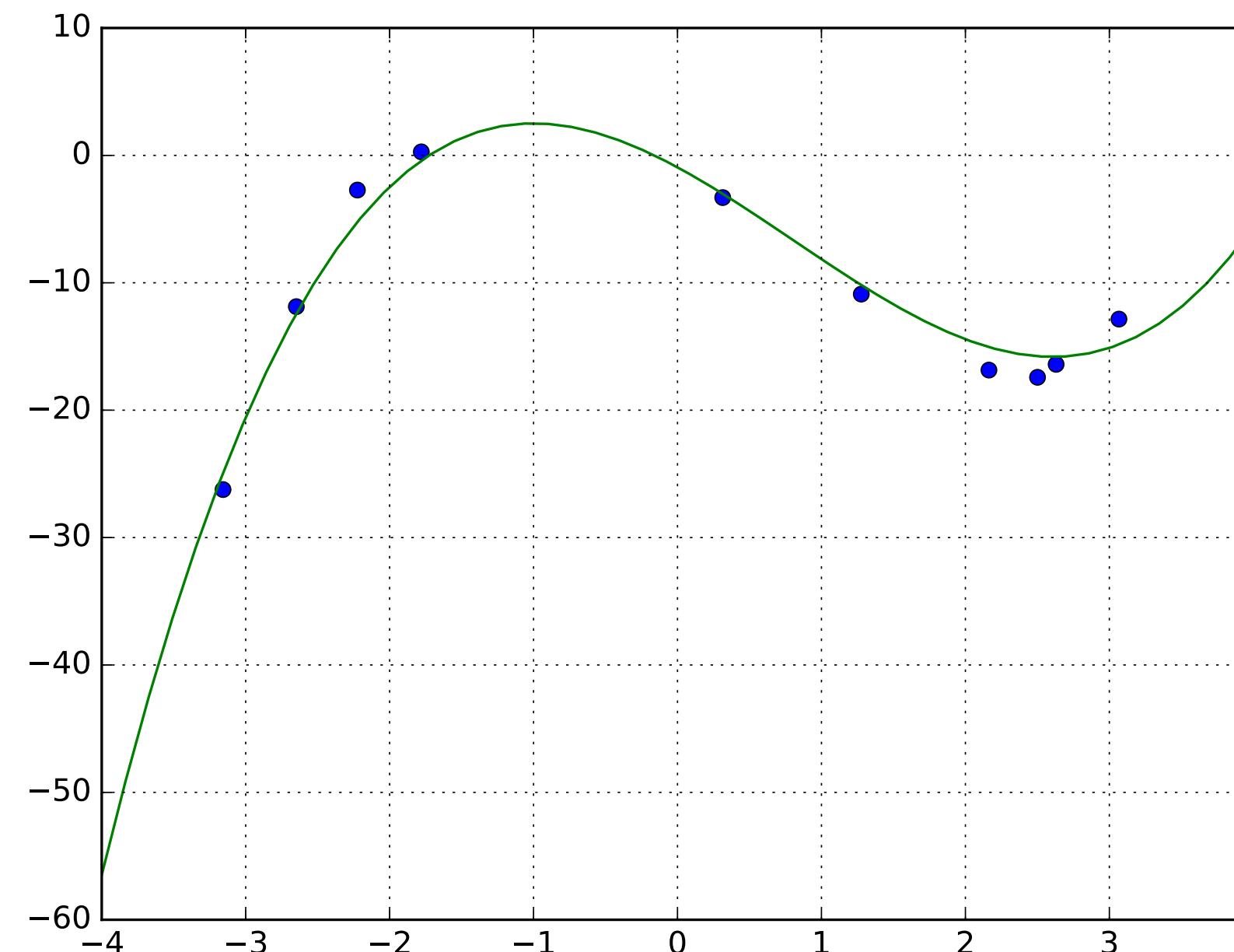
回帰問題

与えられたデータ点に対して、未観測点の値を推定したい

例：多項式回帰モデル

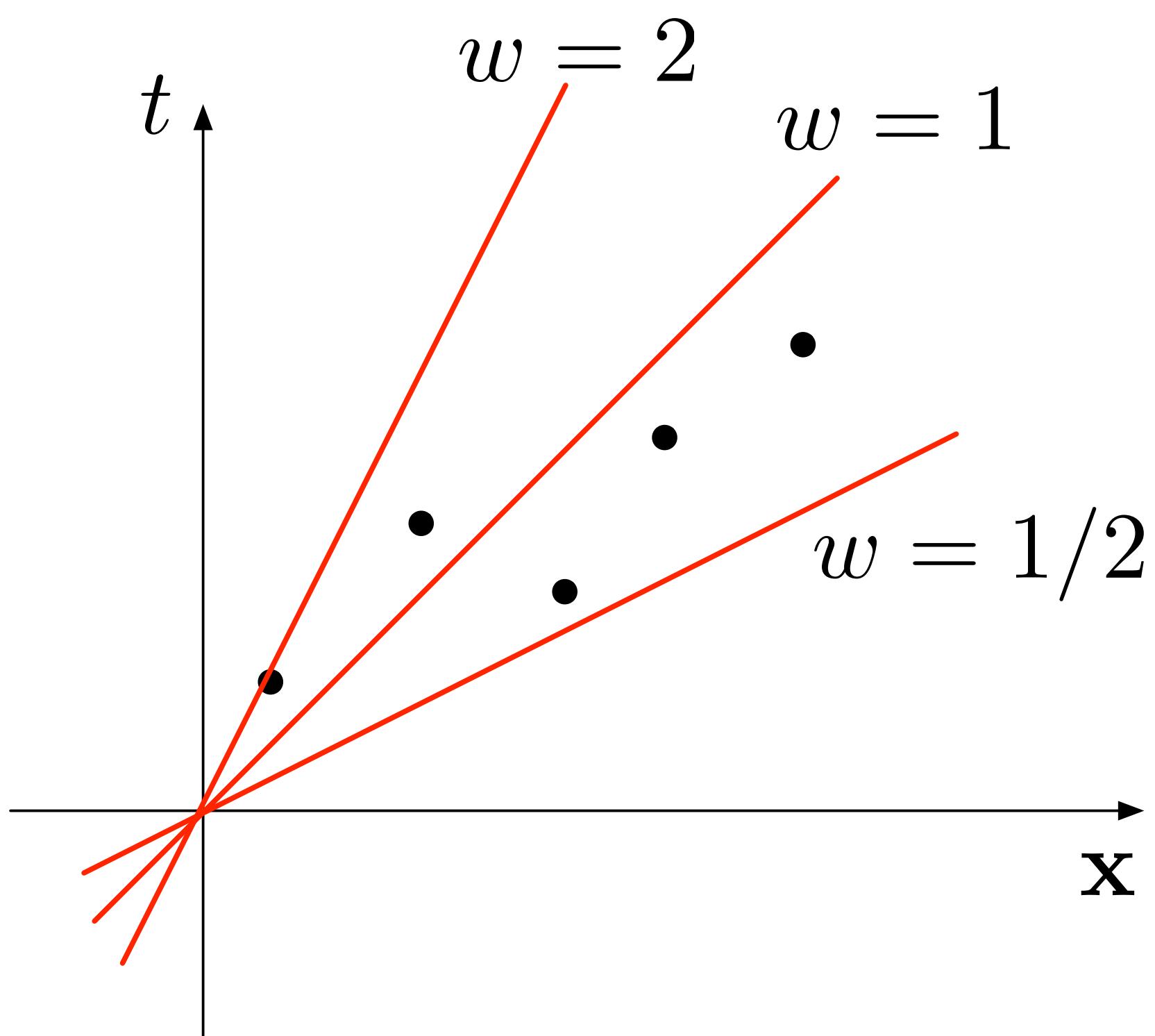
$$f_{\Theta}(x) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M$$

M をこの多項式モデルの次数と呼ぶ。



線形回帰の例

$$f_{\Theta}(x) = wx, \quad \Theta = \{w\}$$



最もデータにフィットする直線を選びたい

練習問題(1)

パラメトリック関数モデル(線形単回帰モデル)

$$f_{\Theta}(x) = wx, \quad \Theta = \{w\}$$

データ

$$\{(x_1 = 1, t_1 = 2.2), (x_2 = 2, t_2 = 3.8)\}$$

[練習問題]

(1) この場合の二乗誤差関数を書け。

(2) 二乗誤差関数の勾配 (導関数) を求めよ。

(3) 極値条件より、 w の値を決定せよ。

$$loss(y, \hat{y}) = \frac{1}{2} \sum_{i=1}^n |y_i - \hat{y}_i|^2$$

練習問題(1)解答

$$\begin{aligned}(1) \quad E(w) &= \frac{1}{2}(wx_1 - t_1)^2 + \frac{1}{2}(wx_2 - t_2)^2 \\&= \frac{1}{2}(w - 2.2)^2 + \frac{1}{2}(2w - 3.8)^2\end{aligned}$$

$$\begin{aligned}(2) \quad \frac{\partial E(w)}{\partial w} &= (w - 2.2) + 2(2w - 3.8) \\&= 5w - 9.8\end{aligned}$$

(3) $E(w)$ は w に関して二次関数であり、下に凸である。

したがって、「極値 = 最小値」が保証される。

$$5w - 9.8 = 0 \text{ を解いて、} w^* = 1.96$$

練習問題(2)

パラメトリック関数モデル(重回帰モデル)

$$f_{\Theta}(x_1, x_2) = w_1 x_1 + w_2 x_2, \quad \Theta = \{w_1, w_2\}$$

データ

$$\{((1,2),4), ((3,1),2)\}$$

[練習問題]

(1)誤差関数を求めよ。

(2)極値条件(勾配ベクトル=0)より、 w_1, w_2 の値を決定せよ。

練習問題(2)解答

(1) 誤差関数 $E(w_1, w_2)$ は

$$E(w_1, w_2) = \frac{1}{2}(w_1 + 2w_2 - 4)^2 + \frac{1}{2}(3w_1 + w_2 - 2)^2$$

と与えられる。この関数を w_1, w_2 でそれぞれ偏微分すると

$$\begin{aligned}\frac{\partial E}{\partial w_1} &= 10w_1 + 5w_2 - 10 \\ \frac{\partial E}{\partial w_2} &= 5w_1 + 5w_2 - 10\end{aligned}$$

(2)

$$\begin{aligned}\frac{\partial E}{\partial w_1} &= 10w_1 + 5w_2 - 10 = 0 \\ \frac{\partial E}{\partial w_2} &= 5w_1 + 5w_2 - 10 = 0\end{aligned}$$

を解くことにより $w_1^* = 0, w_2^* = 2$ を得る。

回帰予測はどんなところで使えるのか

例えば、興味のある対象について、過去のデータから将来の観測値を予測する、という使い方ある。

例 1: 株価予測

訓練集合 第 i 日の円ドル為替レート、第 i 日のX社の株価、
そのほか経済指標

→

予測目標 将来ある時点でのX社の株価

例 2: 自動車事故率(保険会社の視点から)

訓練集合 契約者の年齢、過去の事故歴、車種→

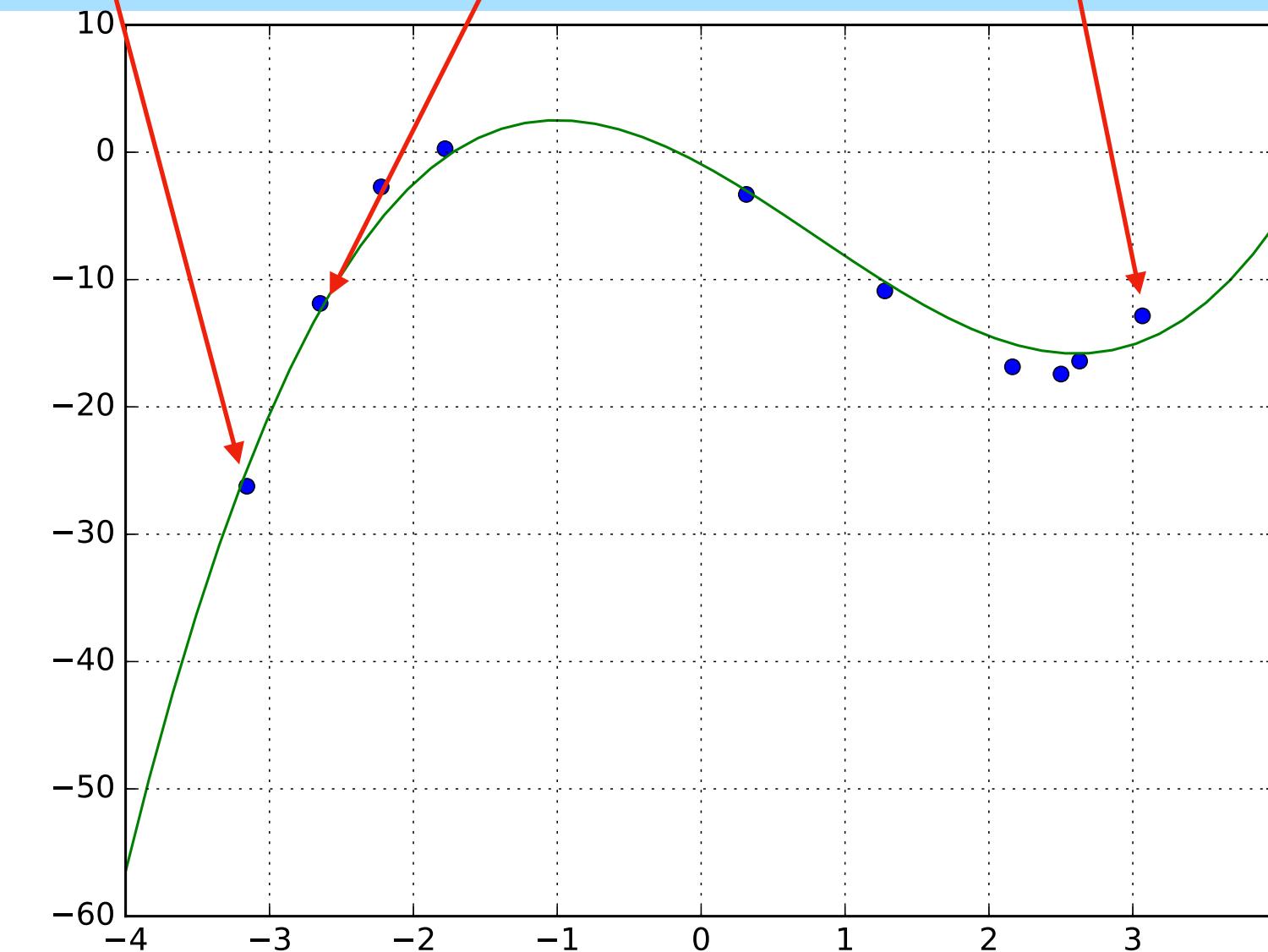
予測目標 将来のある人の事故率

多項式モデルのパラメータ最適化(1)

多項式モデル

$$f_{\Theta}(x) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M$$

データセット $\{(x_1, y_1), (x_2, y_2), \dots, (x_T, y_T)\}$



3次関数モデルによるフィッティング

多項式モデルのパラメータ最適化(2)

$$X = \begin{pmatrix} 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_T & x_T^2 & x_T^3 \end{pmatrix} \quad w = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{pmatrix} \quad Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \end{pmatrix}$$

最小二乗法によるパラメータ最適化

$\|Y - Xw\|^2$ を最小化するようにパラメータwを定める

ここで

$$a = (a_1, a_2, \dots, a_n) \quad \|a\| = \sqrt{a_1^2 + a_2^2 + \cdots a_n^2}$$

多項式モデルのパラメータ最適化(3)

(1) 解析的解法(微分して導関数をゼロと置く)

$$\hat{w} = (X^T X)^{-1} X^T Y$$

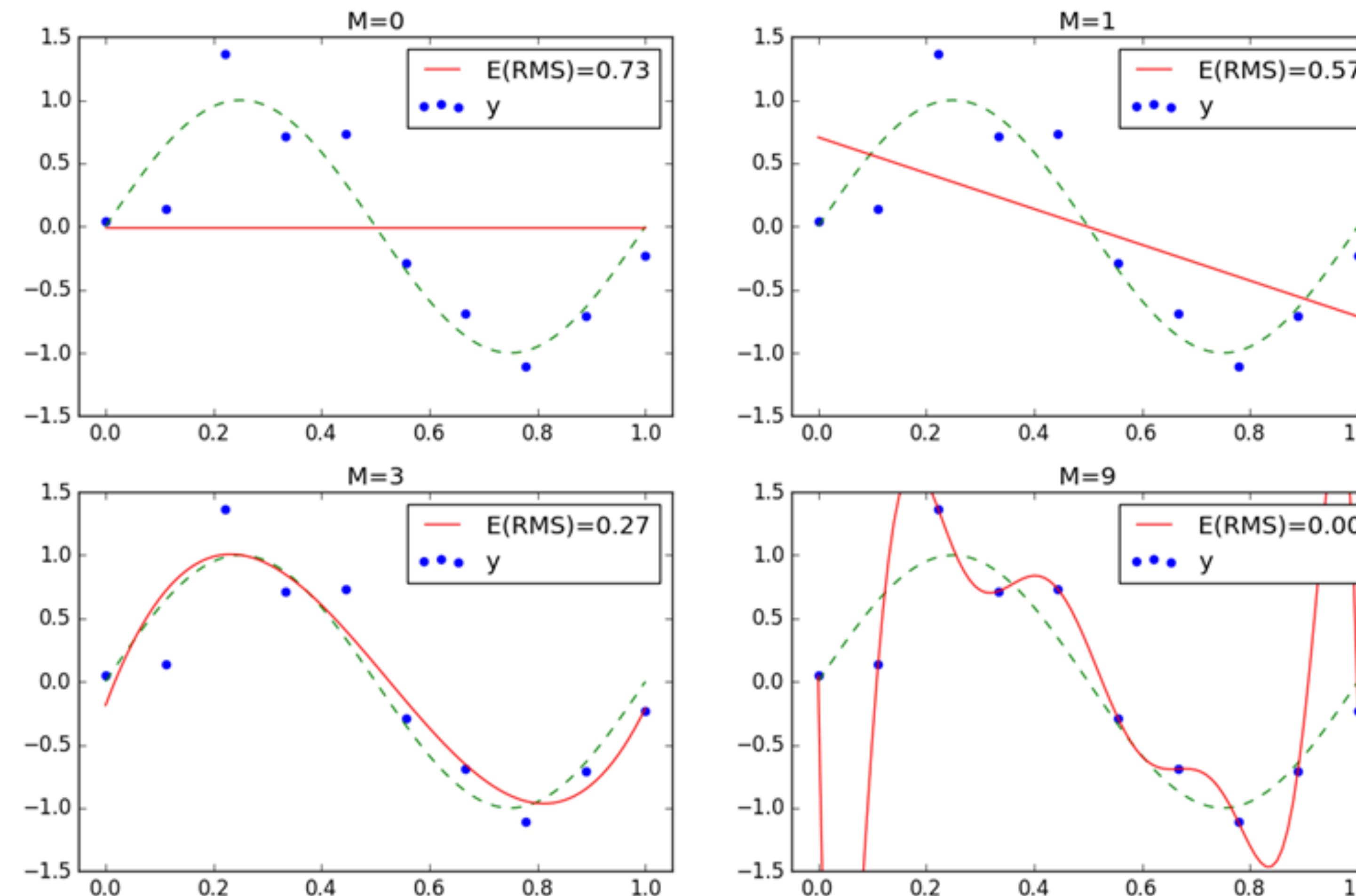
(2) 勾配法を利用する ($\|Y - Xw\|^2$ を最小化)

通常は (1) の解析的手法を利用すればよい。ロジスティック回帰、より一般の一般線形モデルでは解析的手法が利用できないので、その場合は勾配法を利用する

次数の選択(1)

- ・次数が小さすぎる→表現力が限られるため予測精度が十分でない可能性がある
- ・次数が大きすぎる→過学習の恐れが出てくる
- ・適切な次数を選択することが重要→次数選択問題（モデル選択問題の一種）
- ・テスト誤差を基準に選択するのがひとつ的方法
- ・赤池情報量基準(AIC)などの情報量基準をベースとして次数選択(やモデル選択)を行うこともできる

次数の選択 (2)



cited from <http://sonickun.hatenablog.com/entry/2016/07/18/191656>

さまざまな回帰モデル(1)

正弦波重ね合わせモデル

$$f_{\Theta}(x) = w_0 + w_1 \sin(\omega x) + w_2 \sin(2\omega x) + \cdots + w_M \sin(M\omega x),$$

周期関数のモデリングに向いている

線形重回帰モデル

$$f_{\Theta}(x) = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_M x_M, \quad x = (x_1, \dots, x_M) \in \mathbb{R}^n$$

複数の要因変数がある場合に向いている

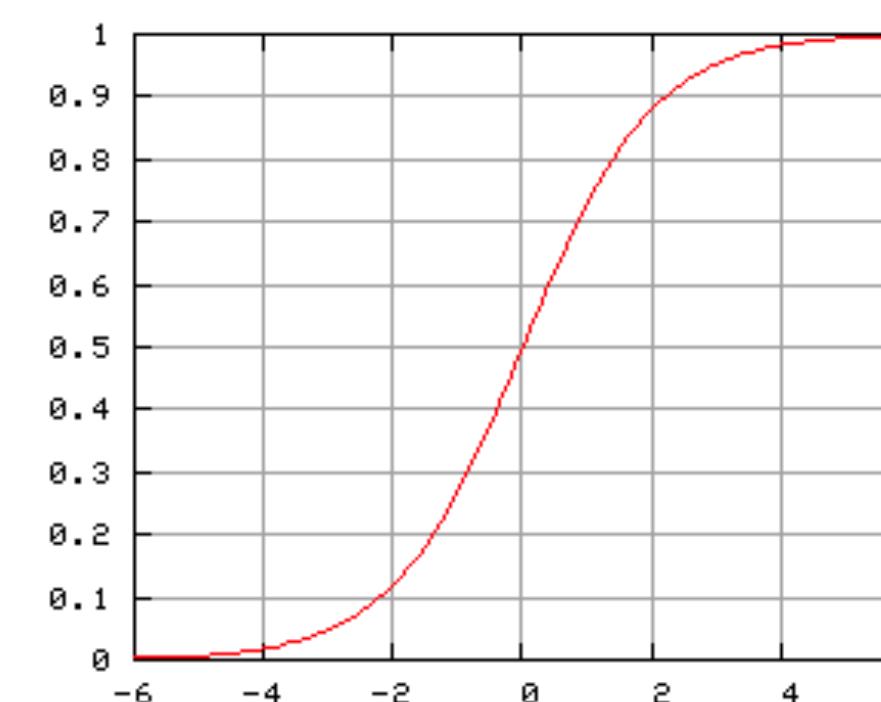
これらは**基底関数**が多項式モデルの場合と違うだけなので、多項式モデルの場合と同様にして、最適パラメータを計算することができる

さまざまな回帰モデル(2)

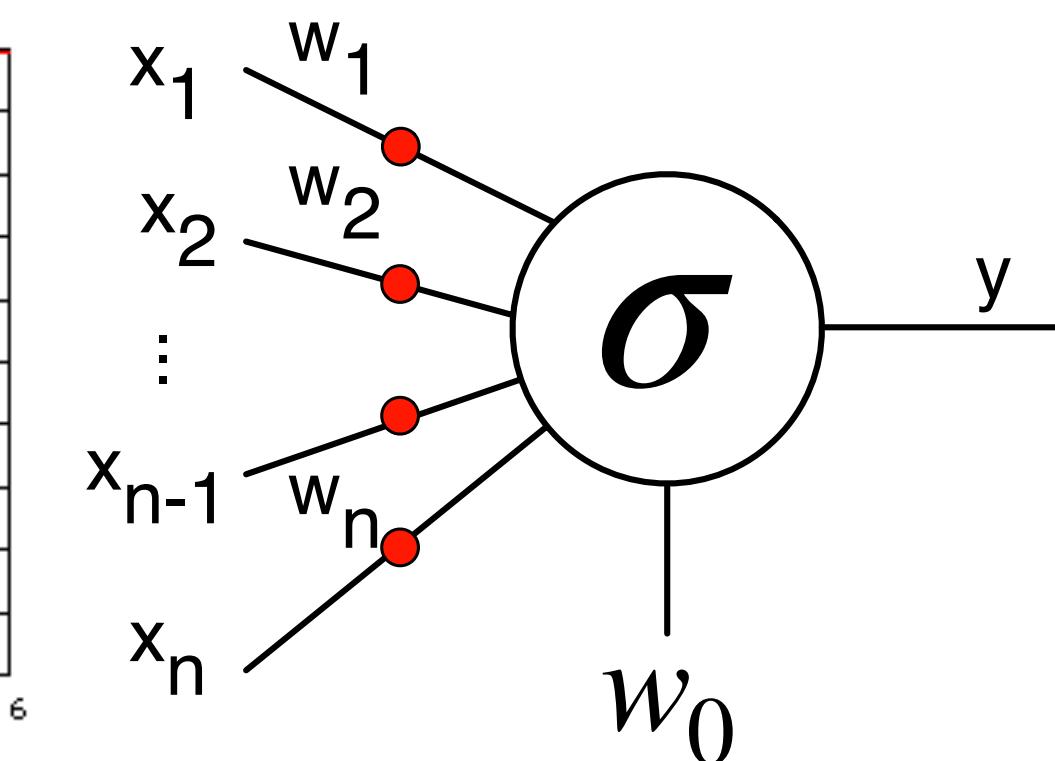
ロジスティック回帰モデル

$$f_{\Theta}(x) = \sigma(w_0 + w_1x_1 + w_2x_2 + \cdots + w_Mx_M), \quad x = (x_1, \dots, x_M) \in \mathbb{R}^n$$

ここで $\sigma(x) = 1/(1 + \exp(-x))$ はシグモイド関数



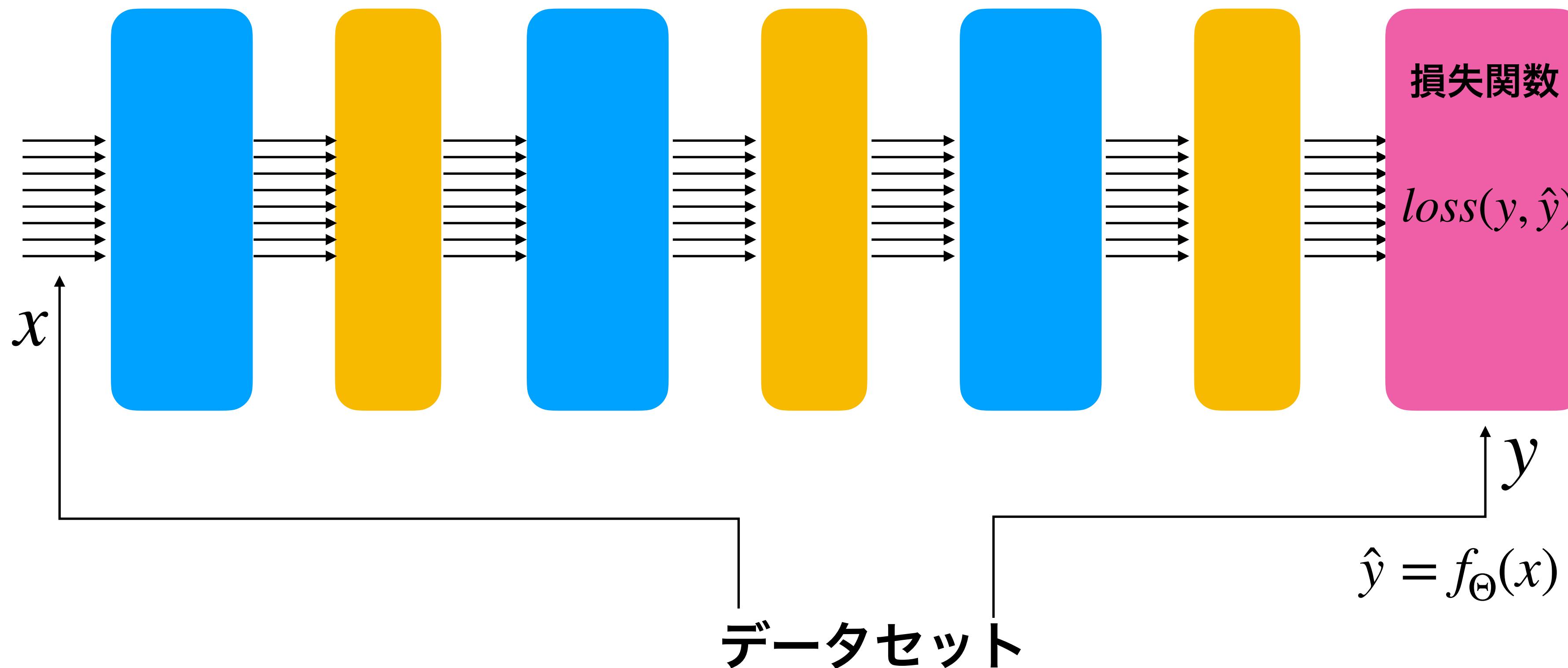
出典 <https://ja.wikipedia.org/wiki/シグモイド関数>



観測データが確率であったり、ゼロ・イチのデータの場合によく利用される。单層ニューラルネットワークと見ることもできる

さまざまな回帰モデル(3)

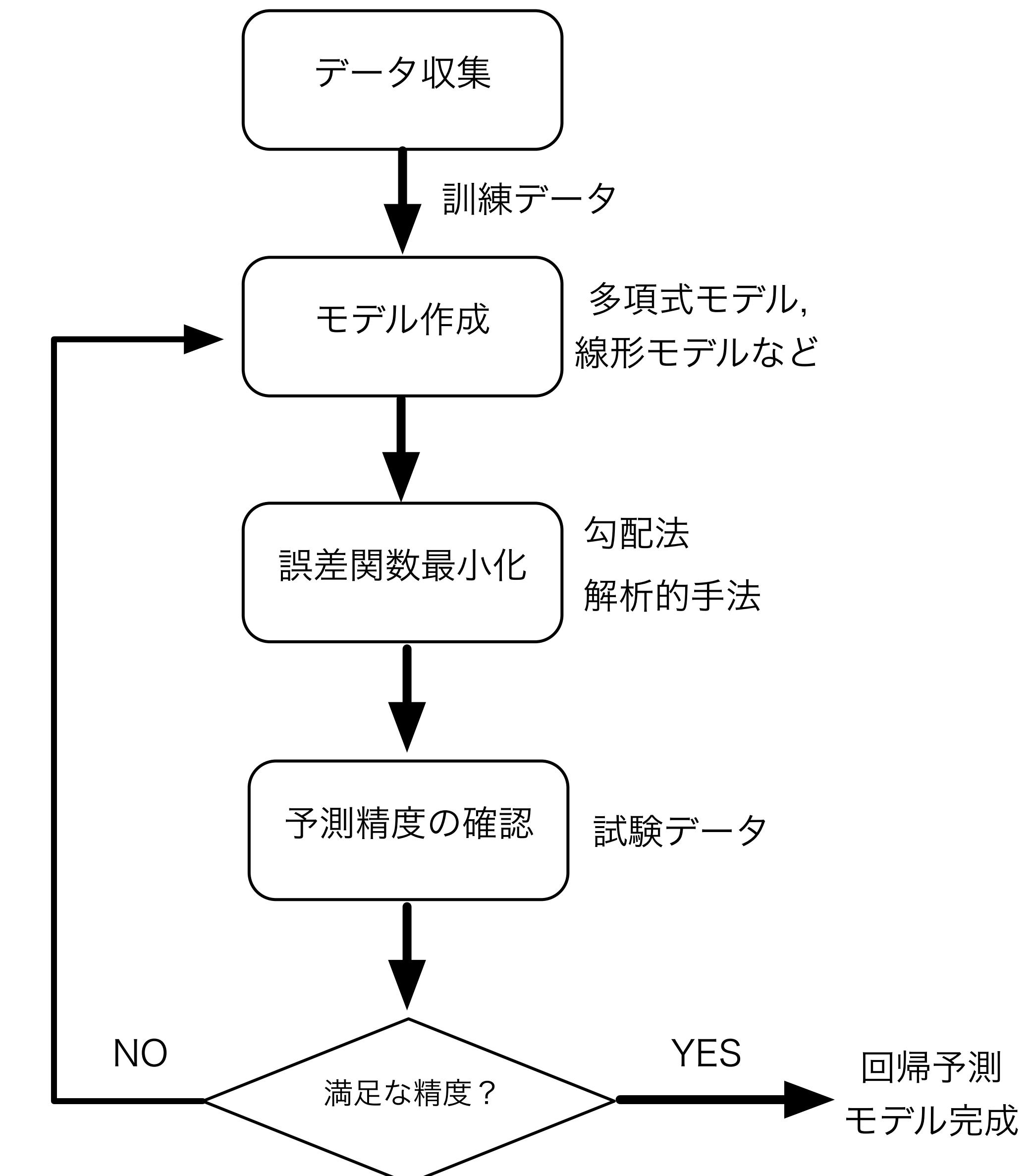
$$\Theta = \{W_1, b_1, W_2, b_2, \dots\}$$



深層NNも回帰問題に自然に利用ができる: 多層非線形関数に基づく回帰モデル

統計モデリングに基づくデータ解析フロー

- ・データセットを収集
- ・目標の推定精度(テスト誤差)を設定(ゴール設定)
- ・回帰モデルを選択
- ・誤差関数最小化により、パラメータ θ を学習
- ・テストデータに基づき、推定精度の評価を行う



データ解析フローの補足

- ・「この問題にはこの予測モデル」という公式はない→データ解析者の**創意工夫が重要**
- ・パラメータ学習においては、小さいモデルの場合は解析的手法が利用できる。そうでない場合は勾配法や凸最適化技法を利用する。
- ・訓練誤差が最小のモデルが必ずしも良いモデルとは限らないことを常に留意する(過学習の可能性)。**予測問題 ≠ 最適化問題(損失値)**
- ・汎化誤差(テスト誤差) の評価しつつ、モデルの逐次改良(次数の調節も含む) を進める
- ・テストデータの使い回しは危険(テストデータへの過適合を生じる) であることに留意が必要。データが少ないとには**クロスバリデーション**を利用する。

統計モデリングについて一歩進んだ知識を得るために



実践的データ解析の手法が
学べる



階層的統計モデルに基づく
データ解析の手法が学べる



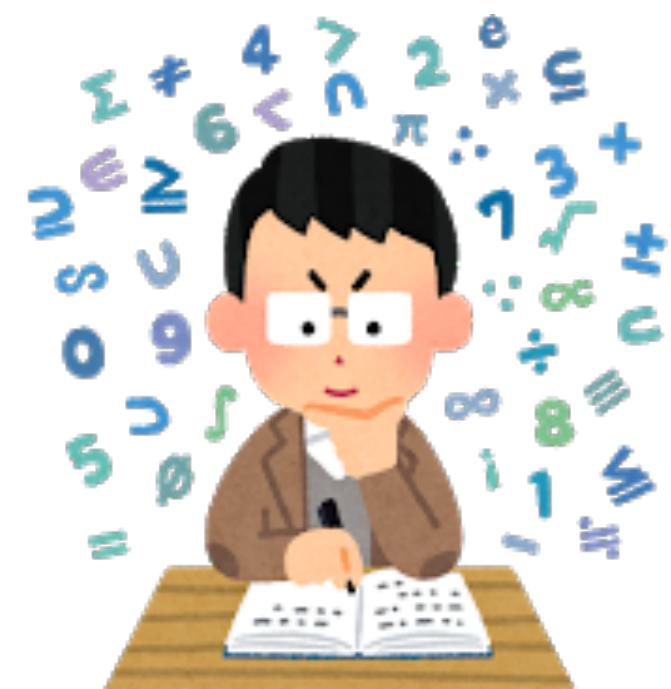
MCMCを利用したベイズ
統計モデリングが学べる

本講義のまとめ

- 誤差逆伝播法の原理
- 回帰問題における予測モデル

Q&A

- Q:自分のキャリアアップのために機械学習技術に興味はあるんですが、勉強のコツってありますか？
A:次ページ以降参照



深層学習の「学習のコツ」

- まずは一冊の入門書で概要を掴む
- 世間の評判や用途などを考えて、フレームワークを選択
(TensorFlow, PyTorchの2択)
- 簡単なプログラムを作成（最初はコードをgithubから拾ってきてまず写経でもよい）実行してみる
- 自分の専門分野で利用できるコードを書いてみるとモチベーションを保ちやすい(自分の専門性 + 深層学習)→分野への貢献につながるかも

実践が大事

- 深層学習分野は実践（コーディング）がとても大事
- フルスクラッチで書くのは結構大変だが、人の書いたソースコード（PyTorch, TensorFlow, Keras）がいっぱいネットにはあるので、最初はそれを有り難く使わせていただく
- 真似ることからスタート
- 人のコードを読むことも有用→ネット上の記事(Qiita)やgithubもみてみよう

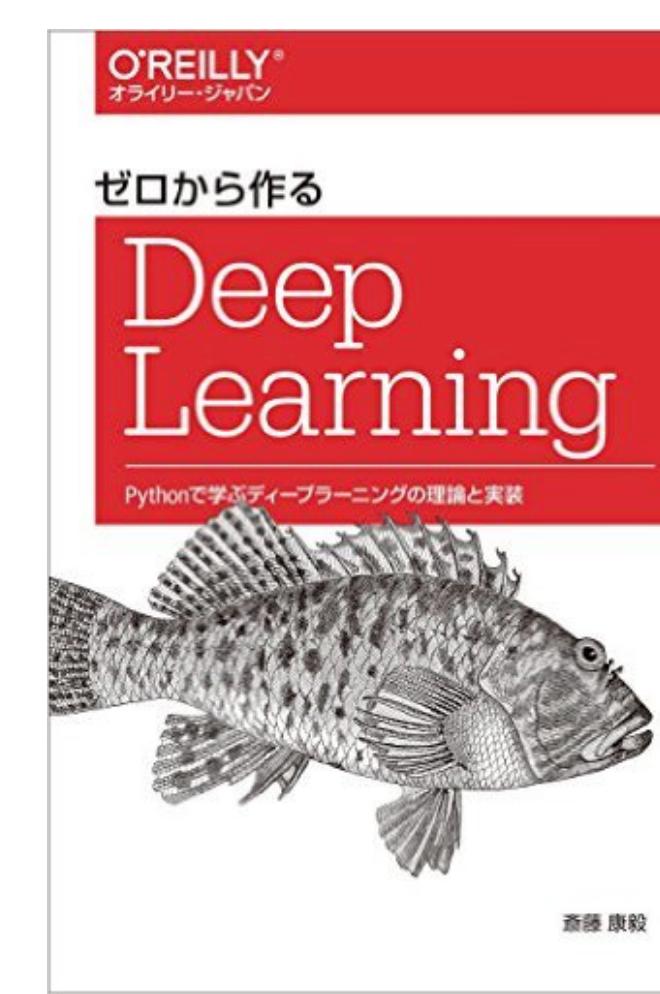


深層学習をもう一歩深く学ぶために

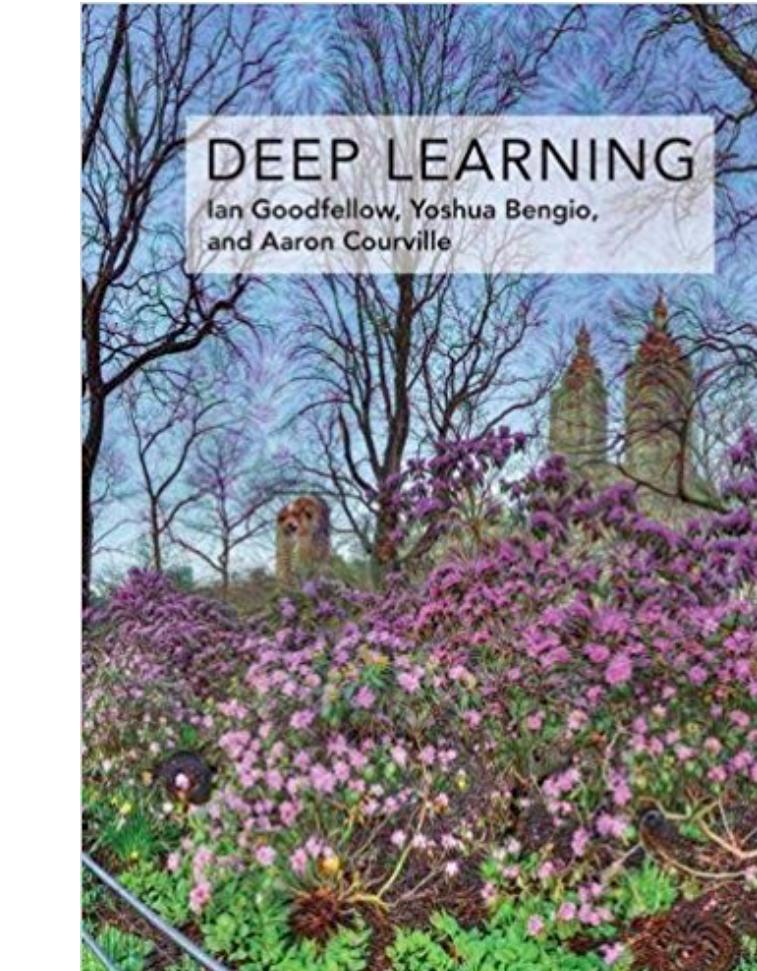
深層学習の概要を
手早く学べる



フルスクラッチ実装
により細部が学べる



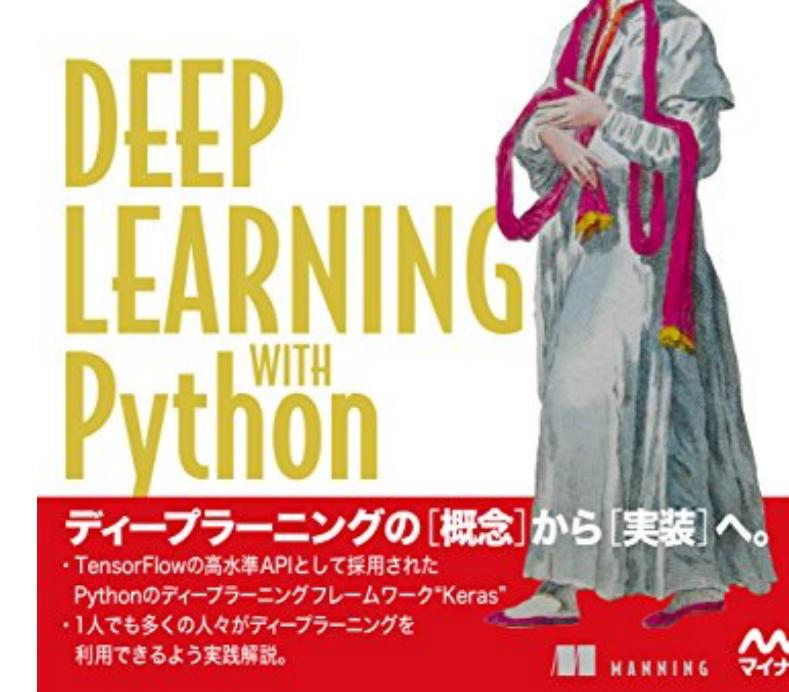
手元においておくと
よい



深層学習への実践的
入門書

**PythonとKerasによる
ディープラーニング**

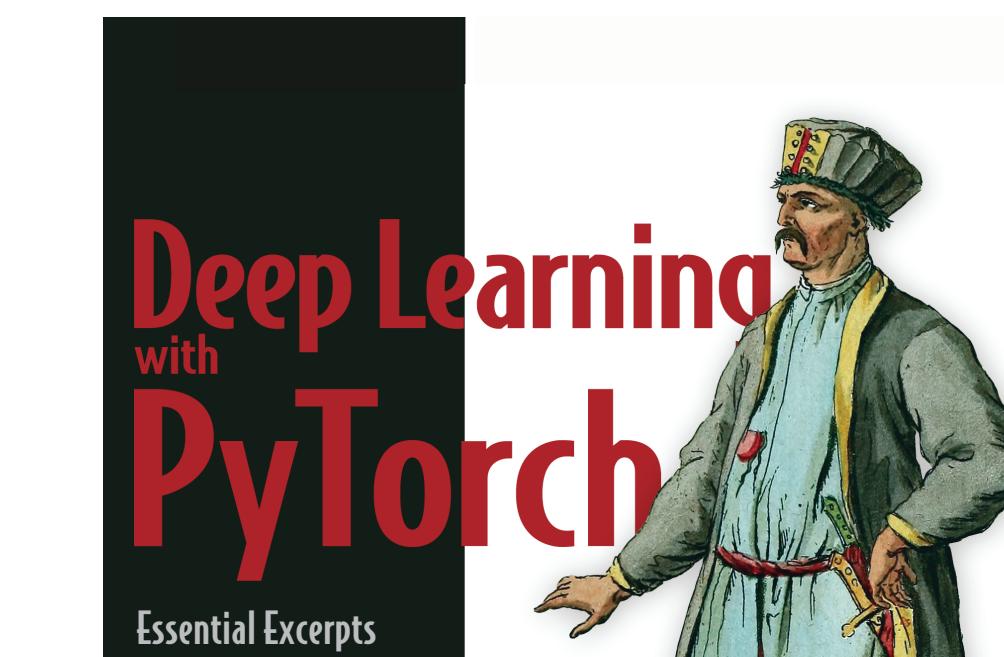
François Chollet [著] 株式会社クイーブ [訳]
巣籠悠輔 [監訳]



中級向け。入門書を終えたあとに
読んで実行するとかなり力がつく



入門向け。最近、日本語版がでました。



中間まとめ

- 深層学習は、情報工学分野のみならず**工学の多くの分野**に多大な影響を与えるつつある
- 深層構造のニューラルネットワークの学習には大量の質の良いデータが必要→今後**データの重要性**がより高まる
- 「数学の力+プログラミング力+(応用分野の)**専門力**」が必要
- 比較的気楽にコードを書くことができる(PyTorch + Google Colab)。
- 各自の専門領域 + 機械学習 → 新発見の可能性