機械学習・ディープラーニングのための
基礎数学講座　微分・線形代数 Day 3

SkillUP AI

配布物

- 1_slide：スライド教材が入ったフォルダ
  - diff_and_linalg_DAY3.pdf：このスライド
- revision_history.txt：改訂履歴

本講座の全体の内容

Day 1 ：微分基礎

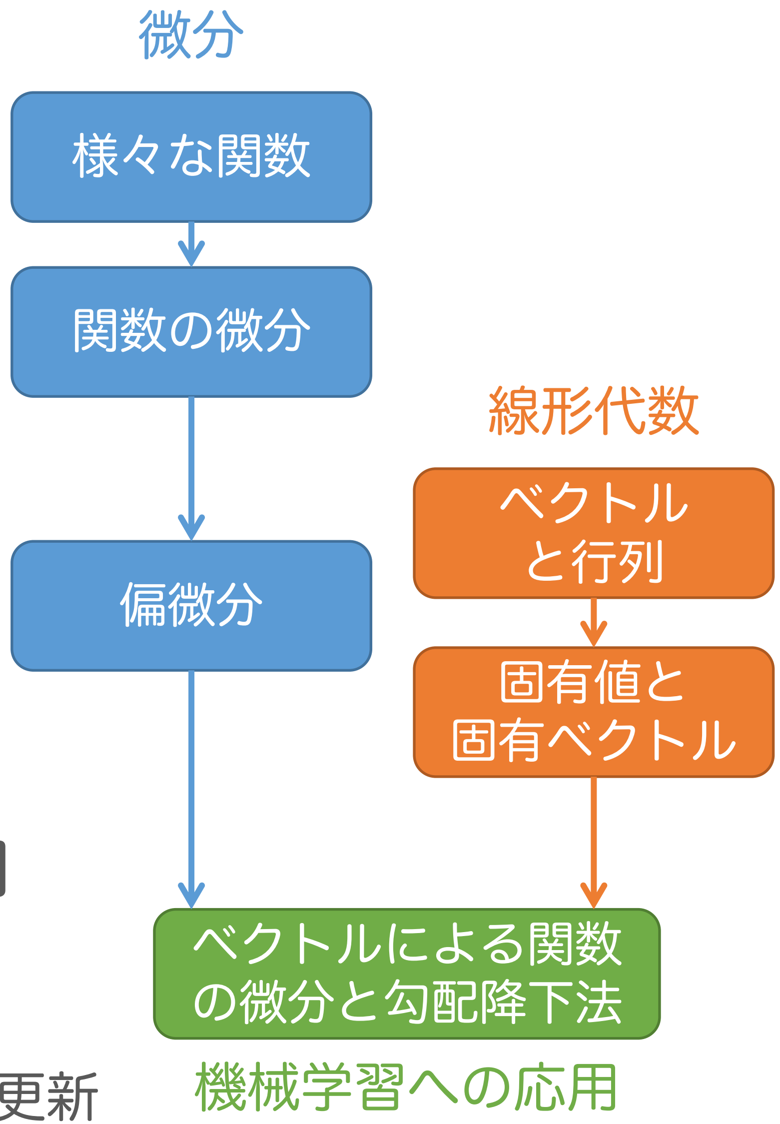
- 内容：様々な関数, 関数の微分
- 修了演習：シグモイド関数の計算グラフと逆伝播計算

Day 2：偏微分と線形代数基礎

- 内容：偏微分, ベクトルと行列, 固有値と固有ベクトル
- 修了演習：特異値分解

Day 3：微分・線形代数の機械学習/深層学習への応用

- 内容：ベクトルによる関数の微分, 勾配降下法
- 修了演習：最小二乗法・誤差逆伝播法 & 勾配法による重み更新



参考文献（DAY1～DAY3を通して）

- **増補改訂版 語りかける中学数学**
  - https://www.beret.co.jp/books/detail/459
  - 中学数学が怪しい方へ
- **ライブ講義　大学１年生のための数学入門**
  - https://bookclub.kodansha.co.jp/product?item=0000275978
  - 数学で出てくる記号の意味が怪しい方へ
- **数研講座シリーズ 大学教養 微分積分の基礎**
  - https://www.chart.co.jp/goods/item/sugaku/46983.php
- **数研講座シリーズ 大学教養 微分積分**
  - https://www.chart.co.jp/goods/item/sugaku/39940.php
- **チャート式シリーズ 大学教養 微分積分**
  - https://www.chart.co.jp/goods/item/sugaku/39952.php
  - 問題集形式

参考文献（DAY1～DAY3を通して）

- 数研講座シリーズ 大学教養 線形代数
  - https://www.chart.co.jp/goods/item/sugaku/39946.php
- チャート式シリーズ 大学教養 線形代数
  - https://www.chart.co.jp/goods/item/sugaku/44006.php
  - 問題集形式
- 最短コースでわかる ディープラーニングの数学
  - https://www.nikkeibp.co.jp/atclpubmkt/book/19/273470/
- ゼロから作るDeep Learning ―Pythonで学ぶディープラーニングの理論と実装
  - https://www.oreilly.co.jp/books/9784873117584/
  - 計算グラフの考え方を掴み、ディープラーニングの実装方法を理解できる
  - E資格対策として必読の一冊（※ この一冊で全範囲をカバーできるわけではない）
- データサイエンスのための数学
  - https://www.kspub.co.jp/book/detail/5169988.html

本講座でやること / やらないこと

- やること
  - 微分と線形代数分野の重要な概念・公式
  - 各種公式を使った問題演習
  - 機械学習 / 深層学習における上記概念・公式の利用方法の概要
- やらないこと
  - 紹介する公式等の厳密な証明
  - Python等を用いた実装方法
  - 機械学習 / 深層学習の各種手法の詳細な説明

講座に入る前に

- 青字・下線付きは URL リンク付き文字です
  - PDFビューワ上で該当箇所をクリックすると
    参考ページに遷移することができます
  - 例）スキルアップAI
    （スキルアップAIのトップページ https://www.skillupai.com/ へ遷移）

講座に入る前に

- 本講座では機械学習 / 深層学習を学ぶための土台となる内容を学習します
  そのため、目的意識を持って学び、アウトプットすることが重要です
- そこで、次の2点を必ず実施しましょう

1. **事前にスライドに目を通し、予習を行いましょう**
   - 漫然と目を通すだけでなく、どの部分を集中して聞くべきか自分の中で決めておきましょう
2. **各 DAY ごとに振り返り・言語化の時間を取りましょう**
   - 振り返り内容
     - この DAY で学んだ内容で参考になったことは？
     - 内容の簡単なサマリ
     - 重要な公式のまとめ など
   - 振り返りの結果は紙やテキストファイルにまとめましょう

Day 3

微分・線形代数の機械学習/深層学習への応用

目次

第１章：スカラー関数のベクトル微分

第２章：ベクトル関数のベクトル微分

微分の発展！機械学習では
ベクトルによる微分が登場します

第３章：勾配降下法

第４章：修了演習

- 最小二乗法
- 誤差逆伝播法 & 勾配法による重み更新

これまで学んだ知識を総動員し
機械学習・深層学習で
必須の技術を学びましょう

第１章

スカラー関数のベクトル微分

スカラー関数

スカラーとは？ → 値１つのこと

スカラー関数とは → 値を１つ返す関数

$$f(\boldsymbol{x}) = x_1^2 + x_2^2 \qquad \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$f(\boldsymbol{x})$ を $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ で微分するのが「ベクトルによる微分」

スカラー関数のベクトル微分

スカラー関数のベクトル微分

$$\frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}} = \left(\frac{\partial f(\boldsymbol{x})}{\partial x_1} \ \frac{\partial f(\boldsymbol{x})}{\partial x_2} \cdots \frac{\partial f(\boldsymbol{x})}{\partial x_d} \right)^{\mathrm{T}}$$

$\boldsymbol{x}$：d次元の縦ベクトル

$f(\boldsymbol{x})$：スカラー関数

$f(\boldsymbol{x})$ を $\boldsymbol{x}$ の各要素で微分して並べているだけ！

スカラー関数のベクトル微分

$f(x_1, x_2) = a_1 x_1 + a_2 x_2$ のとき

$$\frac{\partial f}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{pmatrix}^{\mathrm{T}} = \begin{pmatrix} \frac{\partial}{\partial x_1}(a_1 x_1 + a_2 x_2) & \frac{\partial}{\partial x_2}(a_1 x_1 + a_2 x_2) \end{pmatrix}^{\mathrm{T}}$$

$$= (a_1 \ \ a_2)^{\mathrm{T}} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$$

スカラー関数のベクトル微分

$\boldsymbol{a}, \boldsymbol{x}$ は d 次元の実数ベクトル Aは $d \times d$ 行列とすると

以下の公式が成り立つ

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{a}^{\mathrm{T}} \boldsymbol{x} = \boldsymbol{a} \qquad \frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{a} = \boldsymbol{a}$$

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{x} = 2\boldsymbol{x} \qquad \frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = \left(A + A^{\mathrm{T}}\right) \boldsymbol{x}$$

スカラー関数のベクトル微分

$\boldsymbol{a}, \boldsymbol{x}$ は d 次元の実数ベクトル Aは $d \times d$ 行列とすると

以下の公式が成り立つ

$$\boxed{\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{a}^{\mathrm{T}} \boldsymbol{x} = \boldsymbol{a}}$$

$$\boxed{\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{a} = \boldsymbol{a}}$$

$$\boxed{\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{x} = 2\boldsymbol{x}}$$

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = \left(A + A^{\mathrm{T}}\right) \boldsymbol{x}$$ 付録

赤枠３つを証明してみよう！

スカラー関数のベクトル微分

d 次元の行列・ベクトルを使って証明しても、実感が湧かないので

以降のページでは

$$\boldsymbol{a} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$$

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

とします

スカラー関数のベクトル微分

$\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} = (a_1 \quad a_2)\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = a_1x_1 + a_2x_2$より

$$\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} = \frac{\partial}{\partial \boldsymbol{x}}(a_1x_1 + a_2x_2) = \left(\frac{\partial}{\partial x_1}(a_1x_1 + a_2x_2) \quad \frac{\partial}{\partial x_2}(a_1x_1 + a_2x_2)\right)^{\mathrm{T}}$$

$$= (a_1 \quad a_2)^{\mathrm{T}} = \boldsymbol{a}$$

よって確かに $\dfrac{\partial}{\partial \boldsymbol{x}}\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} = \boldsymbol{a}$ である

スカラー関数のベクトル微分

$$\boldsymbol{x}^{\mathrm{T}}\boldsymbol{a} = \begin{pmatrix} x_1 & x_2 \end{pmatrix}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = a_1 x_1 + a_2 x_2$$ より

$$\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{a} = \frac{\partial}{\partial \boldsymbol{x}}(a_1 x_1 + a_2 x_2) = \begin{pmatrix} \frac{\partial}{\partial x_1}(a_1 x_1 + a_2 x_2) & \frac{\partial}{\partial x_2}(a_1 x_1 + a_2 x_2) \end{pmatrix}^{\mathrm{T}}$$

$$= \begin{pmatrix} a_1 & a_2 \end{pmatrix}^{\mathrm{T}} = \boldsymbol{a}$$

よって確かに $\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{a} = \boldsymbol{a}$ である

スカラー関数のベクトル微分

$$\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x} = \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1^2 + x_2^2$$より

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}}\boldsymbol{x} = \frac{\partial}{\partial \boldsymbol{x}}(x_1^2 + x_2^2) = \begin{pmatrix} \frac{\partial}{\partial x_1}(x_1^2 + x_2^2) & \frac{\partial}{\partial x_2}(x_1^2 + x_2^2) \end{pmatrix}^{\mathrm{T}}$$

$$= \begin{pmatrix} 2x_1 & 2x_2 \end{pmatrix}^{\mathrm{T}} = 2\boldsymbol{x}$$

よって確かに　$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}}\boldsymbol{x} = 2\boldsymbol{x}$　である

スカラー関数のベクトル微分

$$\frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}} = \left(\frac{\partial f(\boldsymbol{x})}{\partial x_1} \frac{\partial f(\boldsymbol{x})}{\partial x_2} \cdots \frac{\partial f(\boldsymbol{x})}{\partial x_d}\right)^{\mathrm{T}} \Rightarrow \nabla f(\boldsymbol{x})$$

と表記することもあります

$\nabla f(\boldsymbol{x})$ を 勾配ベクトル（グラディエント, gradient）と呼ぶ

∇：ベクトル作用素（ナブラ記号）

第１章：理解確認

（１）　$\boldsymbol{w} = \begin{pmatrix} w_0 \\ w_1 \end{pmatrix}$, $f(\boldsymbol{w}) = w_0 + w_1 x_1$ とする

$\frac{\partial}{\partial \boldsymbol{w}}(\, f(\boldsymbol{w}) - 10\,)^2$ を求めよ

（２）　$\boldsymbol{u} = \begin{pmatrix} x \\ y \end{pmatrix}$, $f(\boldsymbol{u}) = x^2 + y^2$とする

$\frac{\partial}{\partial \boldsymbol{u}} f(\boldsymbol{u}) = \boldsymbol{0}$ を満たす $\boldsymbol{u}$ の各要素の値を求めよ

第１章：理解確認

（１）

$$\frac{\partial}{\partial \boldsymbol{w}}(\,f(\boldsymbol{w}) - 10\,)^2 = \begin{pmatrix} \dfrac{\partial}{\partial w_0}(\,f(\boldsymbol{w}) - 10\,)^2 \\ \dfrac{\partial}{\partial w_1}(\,f(\boldsymbol{w}) - 10\,)^2 \end{pmatrix}$$

$$\frac{\partial}{\partial w_0}(\,f(\boldsymbol{w}) - 10\,)^2 = \frac{\partial}{\partial w_0}(w_0 + w_1 x_1 - 10)^2 = 2(w_0 + w_1 x_1 - 10) \cdot 1 = 2(w_0 + w_1 x_1 - 10)$$

$$\frac{\partial}{\partial w_1}(\,f(\boldsymbol{w}) - 10\,)^2 = \frac{\partial}{\partial w_1}(w_0 + w_1 x_1 - 10)^2 = 2(w_0 + w_1 x_1 - 10) \cdot x_1 = 2x_1(w_0 + w_1 x_1 - 10)$$

$$\frac{\partial}{\partial \boldsymbol{w}}(\,f(\boldsymbol{w}) - 10\,)^2 = \begin{pmatrix} 2(w_0 + w_1 x_1 - 10) \\ 2x_1(w_0 + w_1 x_1 - 10) \end{pmatrix}$$

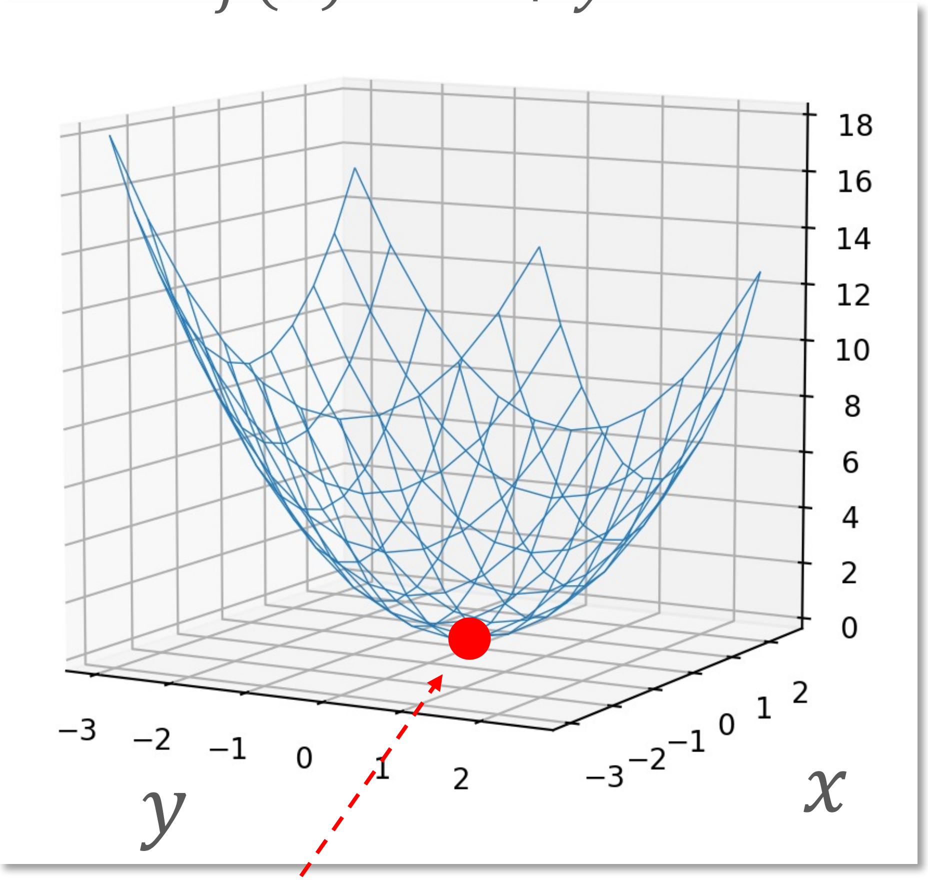
第１章：理解確認

（2）

$$\frac{\partial}{\partial \boldsymbol{u}} f(\boldsymbol{u}) = \begin{pmatrix} \frac{\partial}{\partial x}(x^2+y^2) \\ \frac{\partial}{\partial y}(x^2+y^2) \end{pmatrix} = \begin{pmatrix} 2x \\ 2y \end{pmatrix}$$

$\frac{\partial}{\partial \boldsymbol{u}} f(\boldsymbol{u}) = \mathbf{0}$ を解くと、 $x = y = 0$

$f(\boldsymbol{u}) = x^2 + y^2$



$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

第２章

ベクトル関数のベクトル微分

ベクトル関数

ベクトル関数 → ベクトルを返す関数

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

行列はベクトル関数とも言える

$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$ を $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ で微分するのが「ベクトル関数のベクトルによる微分」

ベクトル関数のベクトル微分

ベクトル関数のベクトル微分

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_1}{\partial x_n} & \cdots & \frac{\partial y_m}{\partial x_n} \end{pmatrix}$$

$\boldsymbol{x}$：n 次元の縦ベクトル

$\boldsymbol{y}$：m 次元の縦ベクトル

「$\boldsymbol{y}$ の各要素」を「$\boldsymbol{x}$ の各要素」で微分して並べているだけ！

※ この行列の転置を定義とする場合もあります

ベクトル関数のベクトル微分

$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$のとき

2×3行列

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_2}{\partial x_1} & \frac{\partial y_3}{\partial x_1} \\ \frac{\partial y_1}{\partial x_2} & \frac{\partial y_2}{\partial x_2} & \frac{\partial y_3}{\partial x_2} \end{pmatrix}$$

ベクトル関数のベクトル微分

$\boldsymbol{x}$はn次元の実数ベクトル、Aは$n \times n$行列とすると、以下の公式が成り立つ

$$\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x}} = \begin{pmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{pmatrix} = I \qquad \frac{\partial}{\partial \boldsymbol{x}} A\boldsymbol{x} = A^{\mathrm{T}}$$

上記二つを導出してみましょう！

ベクトル関数のベクトル微分

$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ のとき

$$\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial x_1}{\partial x_1} & \frac{\partial x_2}{\partial x_1} \\ \frac{\partial x_1}{\partial x_2} & \frac{\partial x_2}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

よって確かに $\frac{\partial x}{\partial x} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I$ である

ベクトル関数のベクトル微分

$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$のとき　$A\boldsymbol{x} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 \\ a_{21}x_1 + a_{22}x_2 \end{pmatrix}$

$$\frac{\partial}{\partial \boldsymbol{x}} A\boldsymbol{x} = \frac{\partial}{\partial \boldsymbol{x}} \begin{pmatrix} a_{11}x_1 + a_{12}x_2 \\ a_{21}x_1 + a_{22}x_2 \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x_1}(a_{11}x_1 + a_{12}x_2) & \frac{\partial}{\partial x_1}(a_{21}x_1 + a_{22}x_2) \\ \frac{\partial}{\partial x_2}(a_{11}x_1 + a_{12}x_2) & \frac{\partial}{\partial x_2}(a_{21}x_1 + a_{22}x_2) \end{pmatrix}$$

$$= \begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix} = A^{\mathrm{T}}$$

よって確かに　$\frac{\partial}{\partial \boldsymbol{x}} A\boldsymbol{x} = A^{\mathrm{T}}$　である

補足）ヘッセ行列

$$\nabla\left(\nabla f(\boldsymbol{x})\right) = \nabla^2 f(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1^2} & \frac{\partial f}{\partial x_1 x_2} & \cdots & \frac{\partial f}{\partial x_1 x_n} \\ \frac{\partial f}{\partial x_2 x_1} & \frac{\partial f}{\partial x_2^2} & \cdots & \frac{\partial f}{\partial x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial x_n x_1} & \frac{\partial f}{\partial x_n x_2} & \cdots & \frac{\partial f}{\partial x_n^2} \end{pmatrix}$$

ニュートン法というパラメータ最適化手法で登場

第２章：理解確認

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad f(\boldsymbol{x}) = x_1^2 + 2x_1x_2 + 3x_2^2$$

（１）$\dfrac{\partial}{\partial \boldsymbol{x}} f(\boldsymbol{x})$

（２）$\dfrac{\partial}{\partial \boldsymbol{x}}\left(\dfrac{\partial}{\partial \boldsymbol{x}} f(\boldsymbol{x})\right)$

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad f(\boldsymbol{x}) = x_1^2 + 2x_1x_2 + 3x_2^2$$

（１） $\frac{\partial}{\partial x} f(\boldsymbol{x}) = \begin{pmatrix} 2x_1 + 2x_2 \\ 2x_1 + 6x_2 \end{pmatrix}$

（２） $\frac{\partial}{\partial x}\left(\frac{\partial}{\partial x} f(\boldsymbol{x})\right) = \begin{pmatrix} 2 & 2 \\ 2 & 6 \end{pmatrix}$

第３章

勾配降下法

勾配降下法

機械学習・深層学習における最重要アルゴリズム

<u>勾配降下法(Gradient Descent)</u>

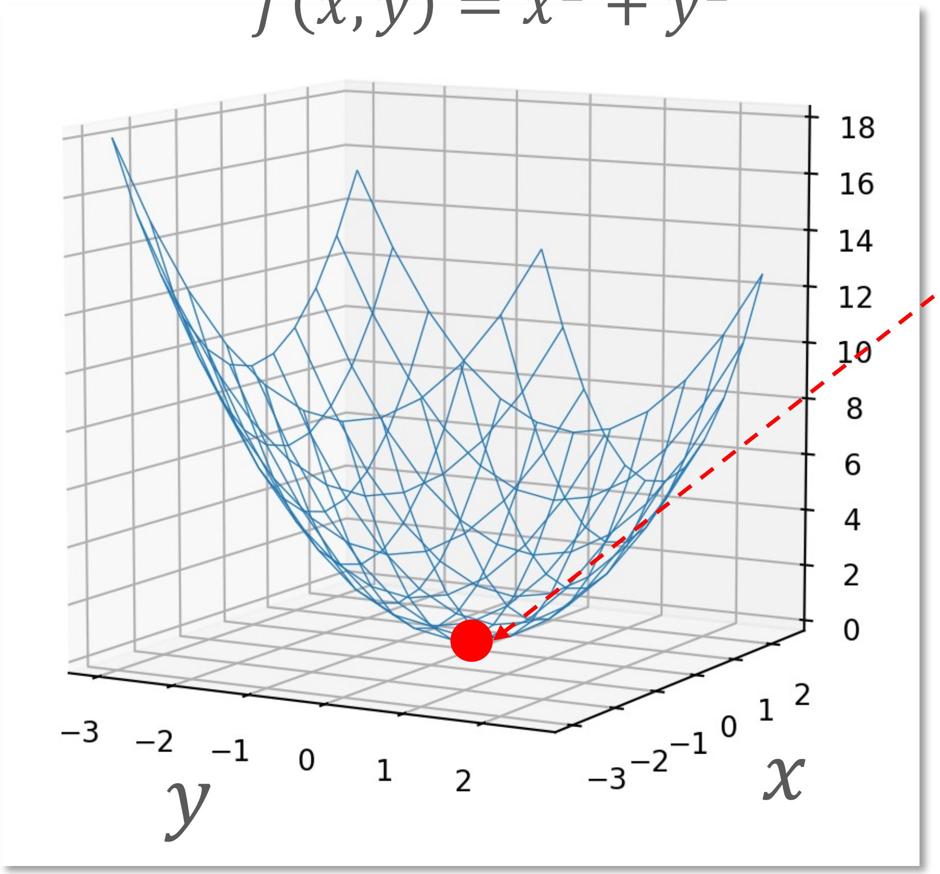
$$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \rho \left.\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}^{(t)}}\right|_{\boldsymbol{w}=\boldsymbol{w}^{(t)}}$$

関数 $E(\boldsymbol{w})$ に最小値 * を与えるパラメータ $\boldsymbol{w}$ を求めるアルゴリズム

* 厳密には極小値

勾配降下法

$f(x, y) = x^2 + y^2$



$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ を求めるためにはどうすれば良い？

$\nabla f(x, y) = \mathbf{0}$ を解く！

しかし、この方程式が解けない場合も多い！

関数に最小値を与える最適解が欲しいが
解析的には求められないという状況が
機械学習では頻発する

勾配降下法

ただ $\nabla f(x,y)=\mathbf{0}$ は解けなくても

$\nabla f(x,y)$ は求められる！

関数の勾配情報を元に最小値を与える解を探索できないだろうか？

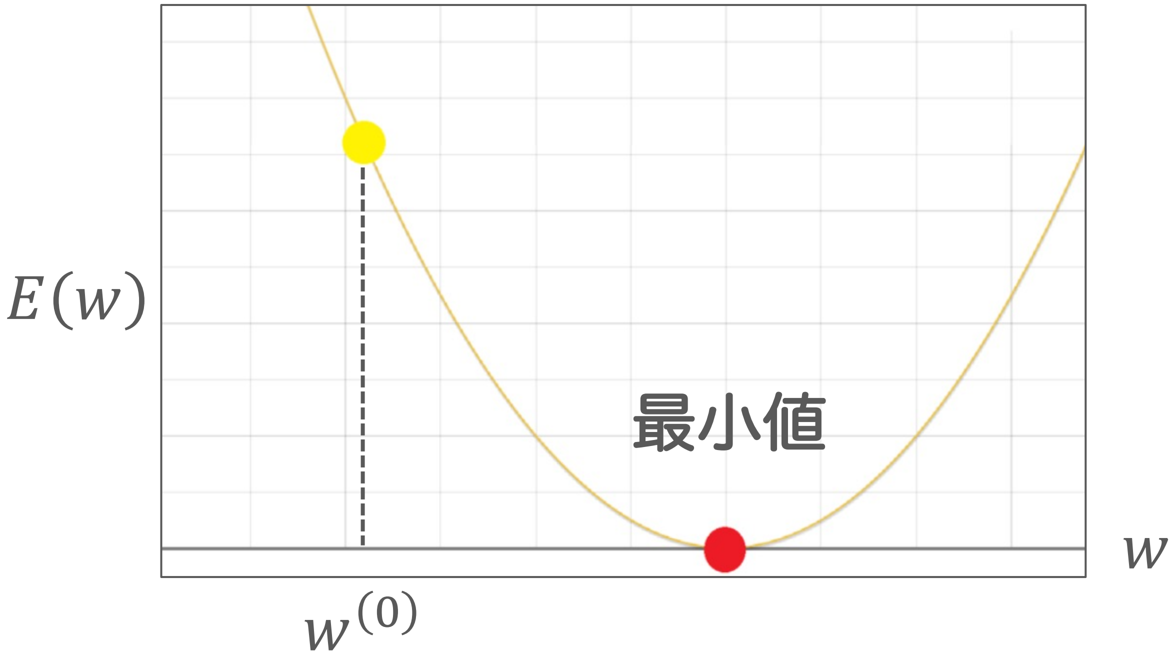
関数 $E(\boldsymbol{w})$ に最小値を与える解を $\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}}=\mathbf{0}$ を解かずして求めよう

勾配降下法

※ 乱数を使ってランダムに決めるのが一般的

まずは適当に※ w の初回位置 $w^{(0)}$ を設定

$w^{(0)}$ の値をどのように更新すると、最小値に近付く？

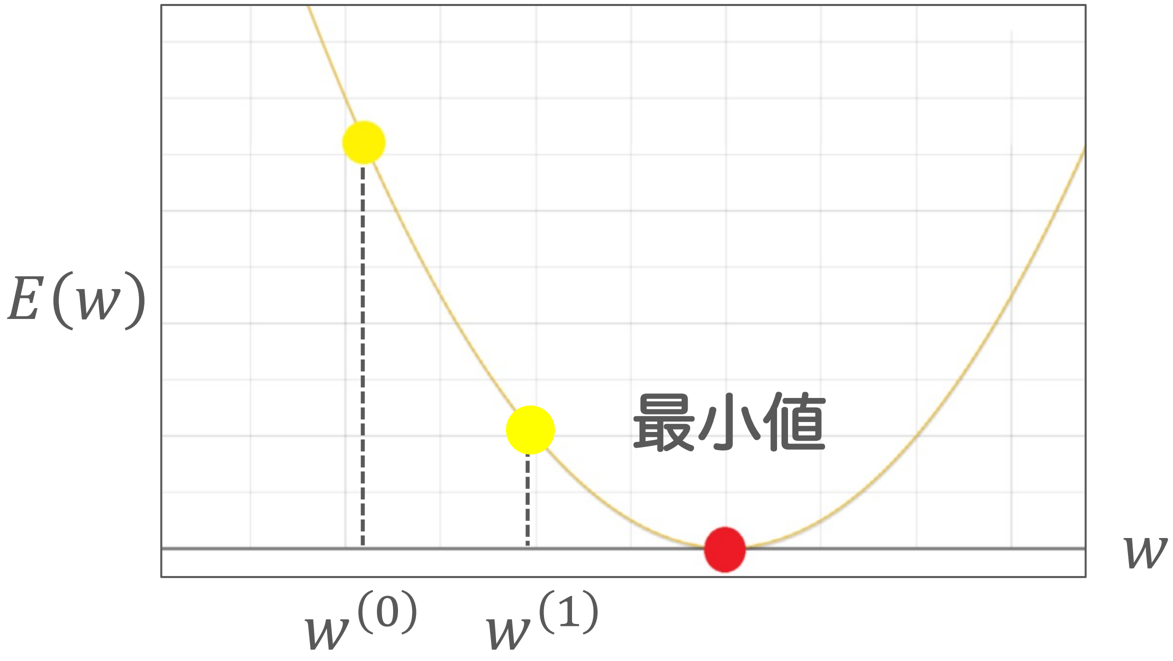


勾配降下法

※ 乱数を使ってランダムに決めるのが一般的

まずは適当に※ w の初回位置 $w^{(0)}$ を設定

$w^{(0)}$ の値をどのように更新すると、最小値に近付く？



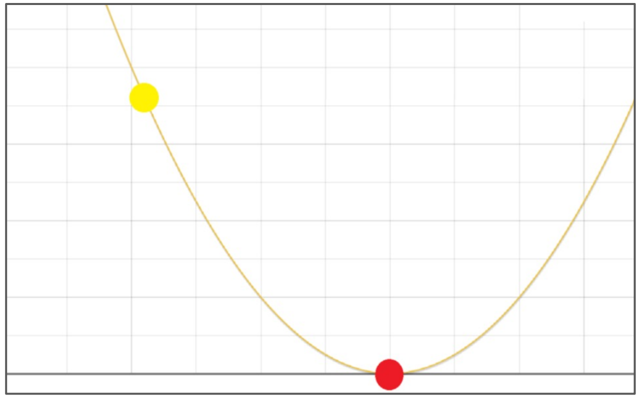
$$w^{(1)} \leftarrow w^{(0)} + \alpha$$

$w^{(0)}$ を大きくする

この更新を何度も繰り返せば最小値に辿り着きそう

勾配降下法

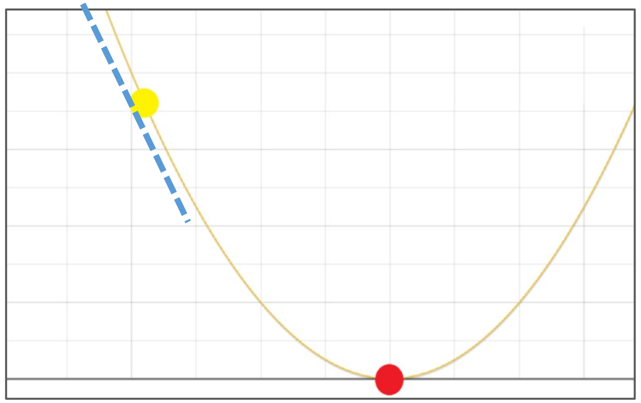
w を更新方法をルール化できれば、コンピュータを用いて最適解を探索できる



どのようなルールを組めば良い？

勾配降下法

w を更新方法をルール化できれば、コンピュータを用いて最適解を探索できる



どのようなルールを組めば良い？

傾きが負ならば、w を大きくする

傾きが正ならば、w を小さくする

勾配降下法

$$w^{(1)} \leftarrow w^{(0)} + \alpha$$

$w = w^{(0)}$ における $\frac{\partial E}{\partial w}$ の値が負ならば

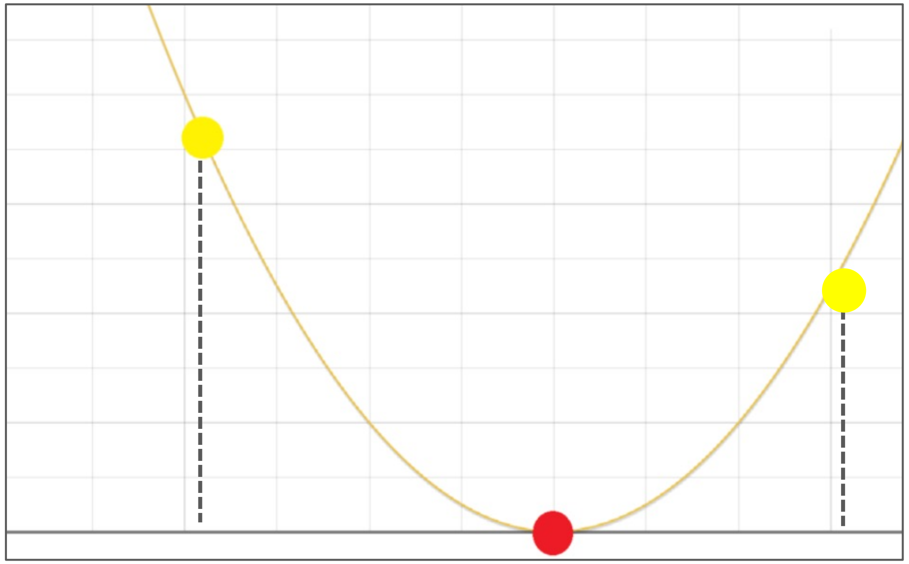
更新量 α は正の値とすればよい

$\alpha = -\frac{\partial E}{\partial w}$ としてしまえば、上手く更新できそう？

アイデアとしては良いのだが実用上上手くいかない

勾配降下法

$\frac{\partial E}{\partial w}$ の絶対値が大きすぎて、更新量が大きすぎる場合がある



$w^{(0)} \longrightarrow w^{(1)}$

$+\alpha$

$$w^{(1)} \leftarrow w^{(0)} + \alpha$$

$$\alpha = -\left.\frac{\partial E}{\partial w}\right|_{w=w^{(0)}}$$

更新量 $-\frac{\partial E}{\partial w}$ に係数をかけて更新量を抑える

勾配降下法

$0.1 \sim 0.001$ などの小さな値を勾配に掛けることで極端な更新を避ける

$$w^{(t+1)} \leftarrow w^{(t)} - \rho \left.\frac{\partial E}{\partial w}\right|_{w=w^{(t)}}$$

この更新則を繰り返せば最小値に辿り着きそう！

では最小値に辿り着いたことを

どのように検知する？

勾配降下法

0.1 ~ 0.001 などの小さな値を勾配に掛けることで極端な更新を避ける

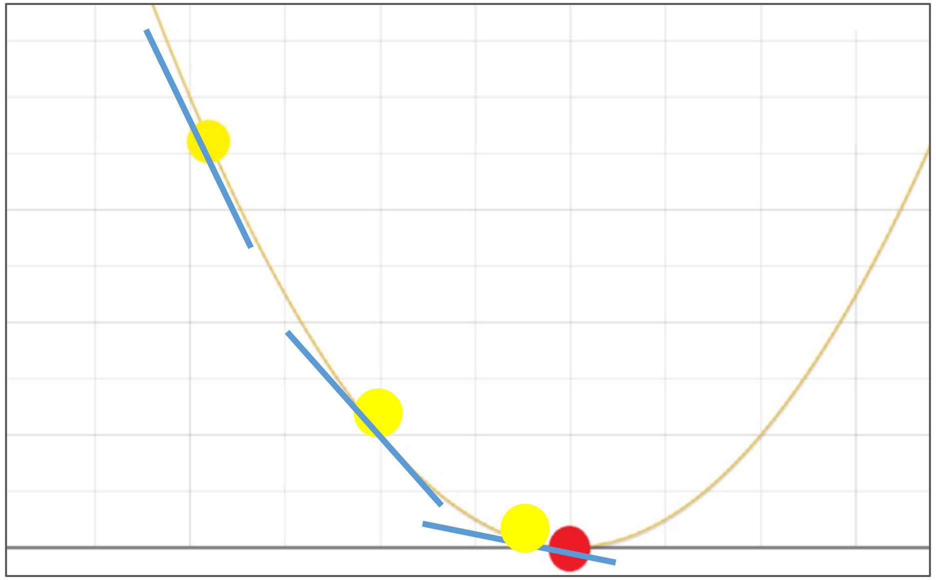
$$w^{(t+1)} \leftarrow w^{(t)} - \rho \left.\frac{\partial E}{\partial w}\right|_{w=w^{(t)}}$$

この更新則を繰り返せば最小値に辿り着きそう！

では最小値に辿り着いたことをどのように検知する？

$$\left.\frac{\partial E}{\partial w}\right|_{w=w^{(t)}} = 0$$

勾配降下法



実際には勾配の値がちょうど０となることはない
勾配の値が非常に小さな値となったら更新を停止

停止した時の w の値を関数に最小値を与える w であると見なす！

勾配降下法

機械学習・深層学習における最重要アルゴリズム

<u>勾配降下法(Gradient Descent)</u>

$$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \rho \left.\frac{\partial E}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\boldsymbol{w}^{(t)}}$$

※ ベクトルの更新に対応した一般的な表記

機械学習の文脈では係数 ρ は「学習率」と呼ばれる

第３章：理解確認

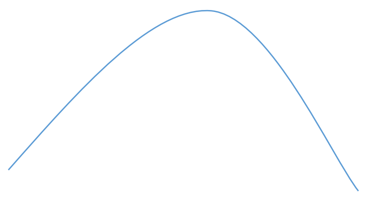
（１）上に凸な関数の最大値を探索する「勾配上昇法」の更新則を求めよ

（２）学習率 ρ の値が極端に小さい場合、更新にどのような影響が及ぶだろうか

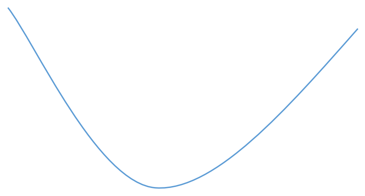
（３）実のところ、勾配降下法には最適解が必ず得られるという保証は無い

最適解が必ず得られるとは限らない関数を考えてみよ

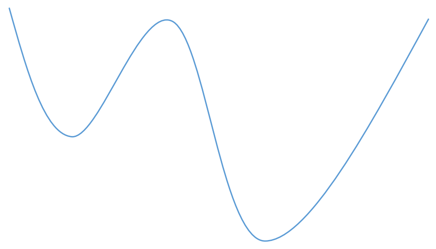
上に凸な関数



下に凸な関数



凸ではない（非凸な）関数



勾配降下法

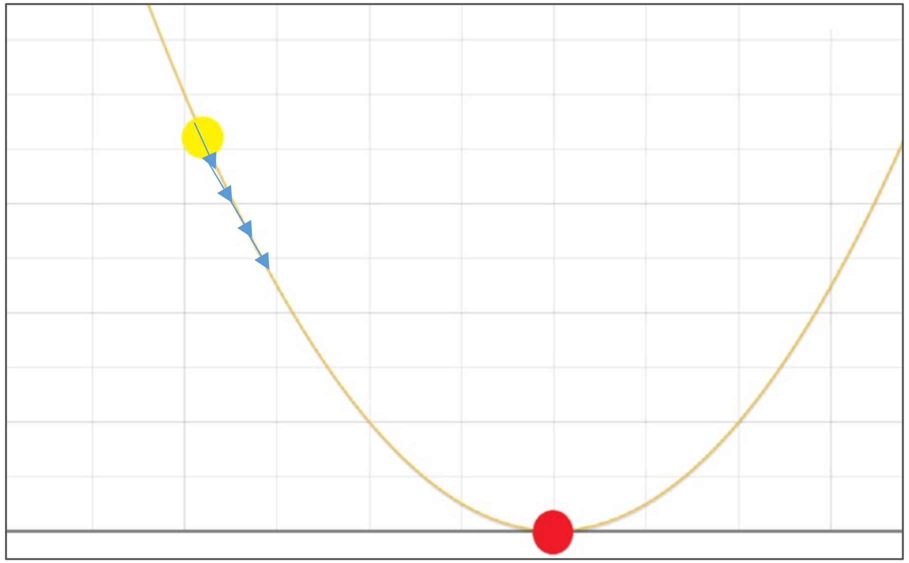
（１）降下法とは逆の動きをすればよい

$$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} + \rho \left.\frac{\partial E}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\boldsymbol{w}^{(t)}}$$

傾きが負ならば w を小さくする

傾きが正ならば w を大きくする

（２）



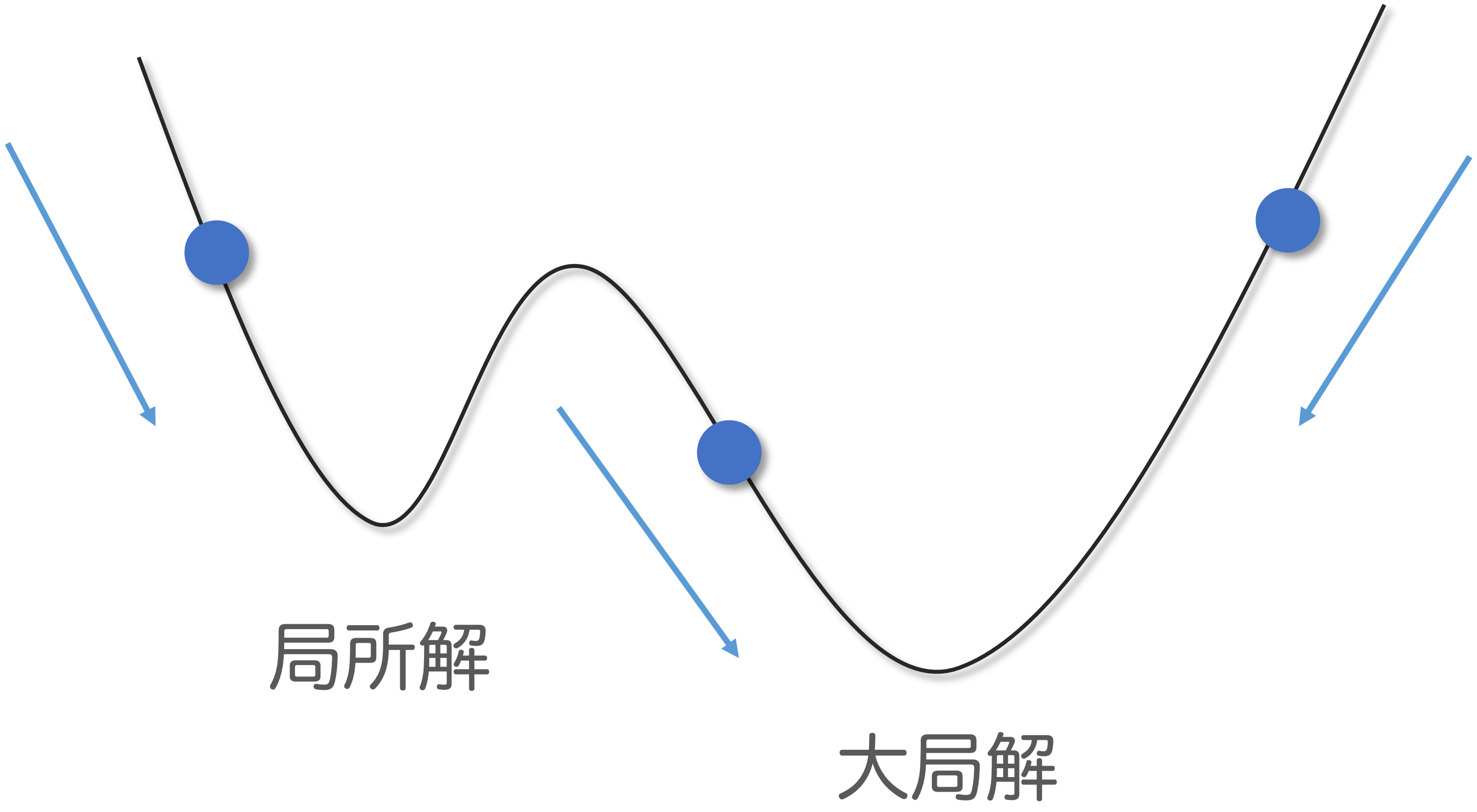
学習率が過度に小さいと

更新がなかなか進まず

最適解が得られるまでに時間がかかる

勾配降下法

（3）凸ではない関数の場合、初期値に応じて得られる最適解が異なる



第４章

修了演習

最小二乗法

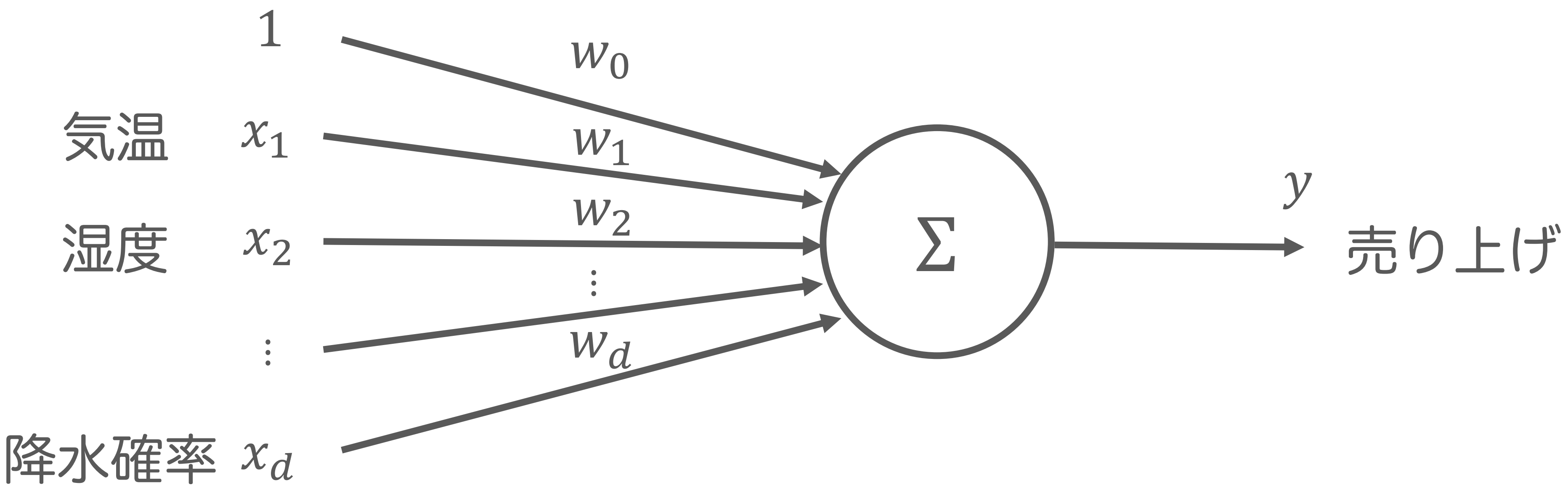
線形モデル

「気温，湿度，降水確率，および過去の売り上げデータ」などから
「ある商品の売上」を予測したい

この問題を線形モデルを用いて解決することを考えよう！

線形モデル

線形モデル



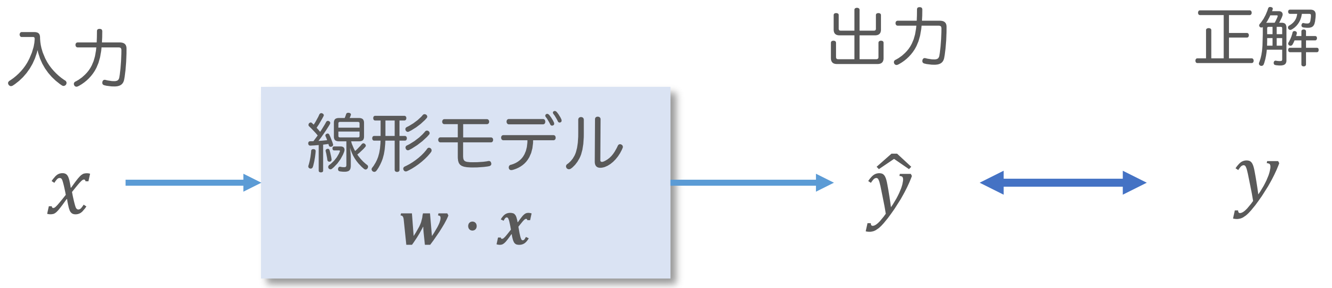
$$y = w_0 \cdot 1 + w_1 x_1 + w_2 x_2 + \cdots + w_d x_d = \boldsymbol{w} \cdot \boldsymbol{x}$$

間違え度合いの定量化

考えるべき問題

売り上げを正しく予測する良い $\boldsymbol{w}$ をどうやって定めれば良いか？

x	y	$\hat{y}$
x_1	y_1	$\hat{y}_1$
x_2	y_2	$\hat{y}_2$
$\vdots$	$\vdots$	$\vdots$

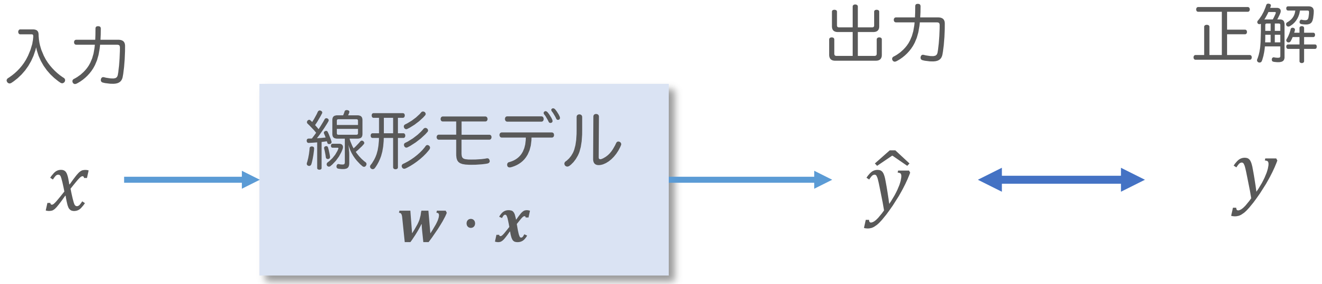


$\hat{y}$ の"間違い度合い"を定量化して

その間違いが最小の $\boldsymbol{w}$ を求めれば良い

間違え度合いの定量化

間違い度合いを表す指標を考えてみよう！



間違え度合いの定量化

間違い度合いを表す指標を考えてみよう！

入力 x → 線形モデル $\boldsymbol{w} \cdot \boldsymbol{x}$ → 出力 $\hat{y}$ ↔ 正解 y

絶対誤差

$$E = \frac{1}{N}\sum_{n=1}^{N}|y_n - \hat{y}_n|$$

二乗誤差

$$E = \frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2$$

微分できない点が
現れないため
二乗誤差の方が
取り扱いしやすい

補足：シグマ計算

シグマ記号（Σ）は数列の総和を表す

$\sum x_i$のように「上付き・下付き文字なし」の場合には x_i 全ての和を意味する

例）データセット：（$1,2,3,4$）

$x_1 = 1,\ x_2 = 2\ ,\ x_3 = 3,\ x_4 = 4$ とおく

$$\sum x_i = x_1 + x_2 + x_3 + x_4 = 10$$

つまり、二乗誤差の式は全データの誤差の二乗を足し合わせ、

データ件数で割ったもの（＝平均を取ったもの）と解釈できる

$$E = \frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2 = \frac{1}{N}\left(\underbrace{(y_1 - \hat{y}_1)^2}_{\text{1件目の正解 } y_1 \text{ と予測結果 } \hat{y}_1 \text{ の差の二乗}} + (y_2 - \hat{y}_2)^2 + \cdots + (y_N - \hat{y}_N)^2\right)$$

最小二乗法

線形モデルの場合、二乗誤差はどのように表現される？

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}(\quad ? \quad)^2$$

最小二乗法

線形モデルの場合、二乗誤差はどのように表現される？

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}(y_n - \boldsymbol{w}\cdot\boldsymbol{x}_n)^2$$

二乗誤差関数 $E(\boldsymbol{w})$ を最小にするような $\boldsymbol{w}$ を求めれば良い！

この手法を最小二乗法と呼ぶ

最小二乗法

線形モデルに対する二乗誤差は凸関数となる

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}(y_n - \boldsymbol{w}\cdot\boldsymbol{x}_n)^2$$

$E(\boldsymbol{w})$ を最小化する $\boldsymbol{w}$ を求めるには、どのような手法を使えば良い？

最小二乗法

$E(\boldsymbol{w})$ を最小化する $\boldsymbol{w}$ を求める

↓

1. $\frac{\partial}{\partial \boldsymbol{w}} E(\boldsymbol{w}) = \boldsymbol{0}$ を解析的に解く！
2. 勾配降下法を用いて最適解を探索する！

線形モデルの場合には方法１でも解が得られることが知られている

↑モデルが変わると大体は解が得られないので方法１は汎用的な方法ではない

最小二乗法

正規方程式

$$\boldsymbol{w} = \left(\Phi^{\mathrm{T}}\Phi\right)^{-1}\Phi^{\mathrm{T}}\boldsymbol{y}$$

$$\Phi = \begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} \\ \boldsymbol{x}_2^{\mathrm{T}} \\ \boldsymbol{x}_3^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_N^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ 1 & x_{31} & x_{32} & \cdots & x_{3d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{Nd} \end{pmatrix} \qquad \boldsymbol{w} = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_d \end{pmatrix} \qquad \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{pmatrix}$$

$\frac{\partial}{\partial \boldsymbol{w}} E(\boldsymbol{w}) = \boldsymbol{0}$ を解くと得られる
（導出は複雑．気になる方は付録を参照）

最小二乗法

$y = \boldsymbol{w} \cdot \boldsymbol{x} = w_0 + w_1 x$ という線形モデルを考える

二乗誤差関数$E(\boldsymbol{w})$ について以下の問いに答えよ

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\bigl(y_n - (w_0 + w_1 x_n)\bigr)^2 = \frac{1}{N}\sum_{n=1}^{N}(y_n - w_0 - w_1 x_n)^2$$

（1） $N = 2$ のときの w_0^2 と w_1^2 の係数を求めよ

（2） $\frac{\partial}{\partial \boldsymbol{w}} E(\boldsymbol{w})$ を求めよ

最小二乗法

（１） $E(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{2}(y_n - w_0 - w_1 x_n)^2$

w_0^2 の係数は 1

w_1^2 の係数は $\frac{1}{2}(x_1^2 + x_2^2)$

$$= \frac{1}{2}\{(y_1 - w_0 - w_1x_1)^2 + (y_2 - w_0 - w_1x_2)^2\}$$

$$= \frac{1}{2}\left\{\left(y_1 - (w_0 + w_1x_1)\right)^2 + \left(y_2 - (w_0 + w_1x_2)\right)^2\right\}$$

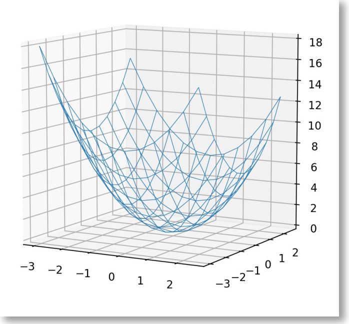
$$= \frac{1}{2}\{(y_1^2 - 2y_1(w_0 + w_1x_1) + (w_0 + w_1x_1)^2) + (y_2^2 - 2y_2(w_0 + w_1x_2) + (w_0 + w_1x_2)^2)\}$$

$$= \frac{1}{2}\{(y_1^2 - 2y_1w_0 - 2y_1w_1x_1 + w_0^2 + 2w_0w_1x_1 + w_1^2x_1^2) + (y_2^2 - 2y_2w_0 - 2y_2w_1x_2 + w_0^2 + 2w_0w_1x_2 + w_1^2x_2^2)\}$$

$$= \frac{1}{2}\{2w_0^2 + (x_1^2 + x_2^2)w_1^2 + (2x_1 + 2x_2)w_0w_1 + (-2y_1 - 2y_2)w_0 + (-2y_1x_1 - 2y_2x_2)w_1 + y_1^2 + y_2^2\}$$

$$= w_0^2 + \frac{1}{2}(x_1^2 + x_2^2)w_1^2 + (x_1 + x_2)w_0w_1 + (-y_1 - y_2)w_0 + (-y_1x_1 - y_2x_2)w_1 + \frac{1}{2}(y_1^2 + y_2^2)$$

最小二乗法



- 得られた結果を w_0 についてまとめると

$$E(\boldsymbol{w}) = w_0^2 + \frac{1}{2}(x_1^2 + x_2^2)w_1^2 + (x_1 + x_2)w_0w_1 + (-y_1 - y_2)w_0 + (-y_1x_1 - y_2x_2)w_1 + \frac{1}{2}(y_1^2 + y_2^2)$$

$$= w_0^2 + (w_1x_1 + w_1x_2 - y_1 - y_2)w_0 + \left(\frac{1}{2}(x_1^2 + x_2^2)w_1^2 + (-y_1x_1 + -y_2x_2)w_1 + \frac{1}{2}(y_1^2 + y_2^2)\right)$$

w_0 に関する二次関数！w_0^2 の係数は正なので、形状は下に凸

- 同じように、得られた結果を w_1 についてまとめると

$$E(\boldsymbol{w}) = w_0^2 + \frac{1}{2}(x_1^2 + x_2^2)w_1^2 + (x_1 + x_2)w_0w_1 + (-y_1 - y_2)w_0 + (-y_1x_1 - y_2x_2)w_1 + \frac{1}{2}(y_1^2 + y_2^2)$$

$$= \frac{1}{2}(x_1^2 + x_2^2)w_1^2 + (w_0x_1 + w_0x_2 - y_1x_1 + -y_2x_2)w_1 + \left(w_0^2 + (-y_1 - y_2)w_0 + \frac{1}{2}(y_1^2 + y_2^2)\right)$$

w_1 に関する二次関数！w_1^2 の係数は正なので、形状は下に凸

つまり、二乗誤差は下凸な関数 ⇒ 得られた解は大局解であることが保証

最小二乗法

（2）

$$\frac{\partial}{\partial \boldsymbol{w}} E(\boldsymbol{w}) = \begin{pmatrix} \dfrac{\partial}{\partial w_0} E(\boldsymbol{w}) \\ \dfrac{\partial}{\partial w_1} E(\boldsymbol{w}) \end{pmatrix}$$

$$\frac{\partial}{\partial w_0}\left\{\frac{1}{N}\sum_{n=1}^{N}(y_n - w_0 - w_1 x_n)^2\right\} = \frac{1}{N}\sum_{n=1}^{N}\frac{\partial}{\partial w_0}(y_n - w_0 - w_1 x_n)^2 = \frac{1}{N}\sum_{n=1}^{N}-2(y_n - w_0 - w_1 x_n)$$

$$\frac{\partial}{\partial w_1}\left\{\frac{1}{N}\sum_{n=1}^{N}(y_n - w_0 - w_1 x_n)^2\right\} = \frac{1}{N}\sum_{n=1}^{N}\frac{\partial}{\partial w_1}(y_n - w_0 - w_1 x_n)^2 = \frac{1}{N}\sum_{n=1}^{N}-2(y_n - w_0 - w_1 x_n)x_n$$

誤差逆伝播法

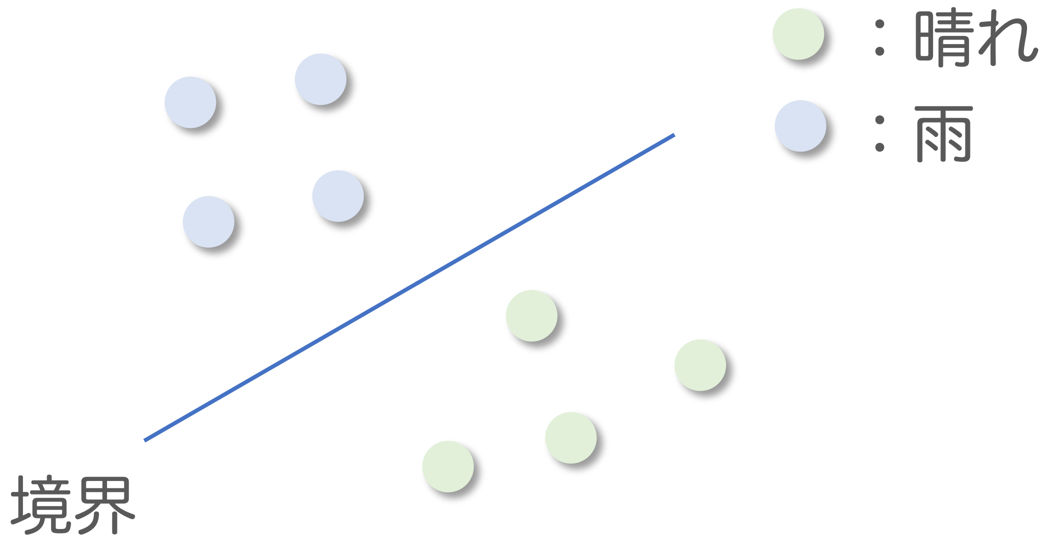
勾配法による重み更新

線形モデル

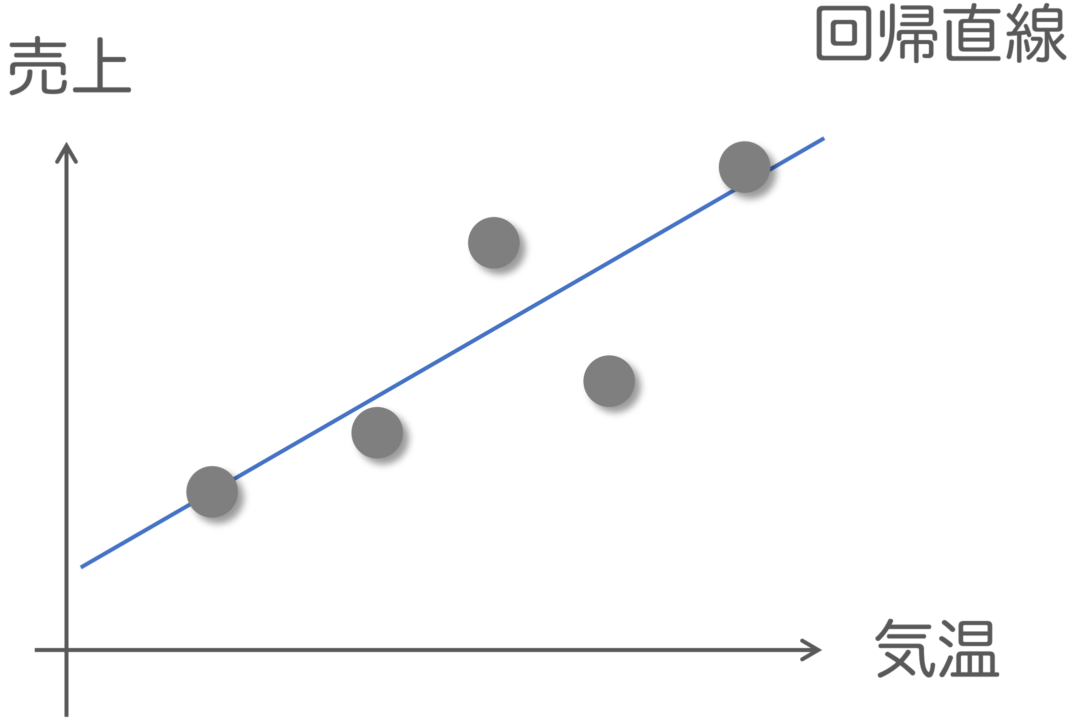
線形モデルは $\boldsymbol{x}$ と $\boldsymbol{w}$ の内積というシンプルな構造ではありながら

機械学習の主要用途である「回帰」「分類」の両方に応用可能

例）天気予報・売り上げ予測



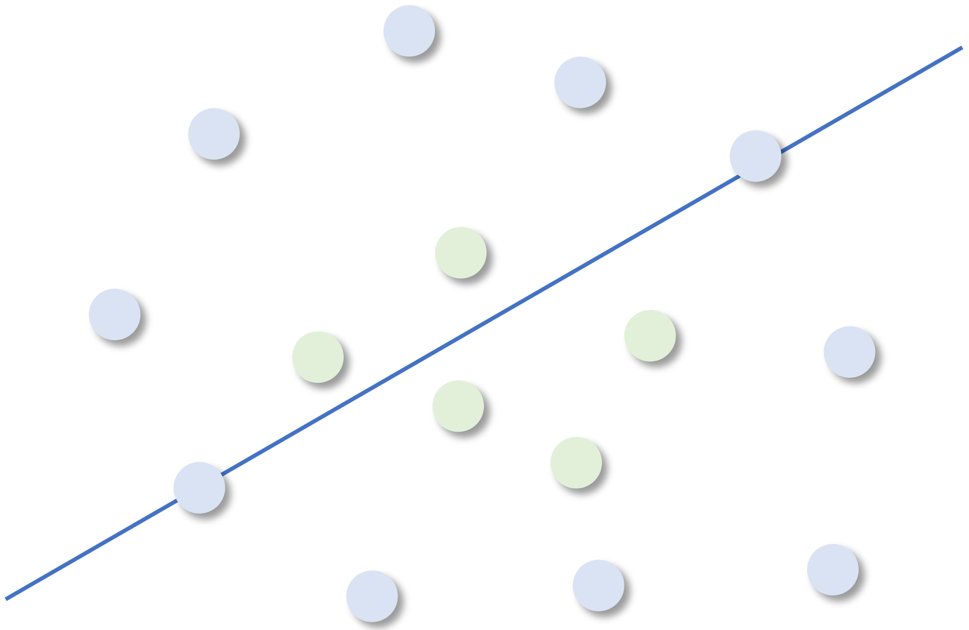
分類は線形モデルを拡張したロジスティック回帰を用いて可能



線形モデル

ただシンプルが故に

データに含まれる複雑な構造（特徴）は捉えることができない



：晴れ

：雨

直線的な特徴の取り方しかできない

線形モデル

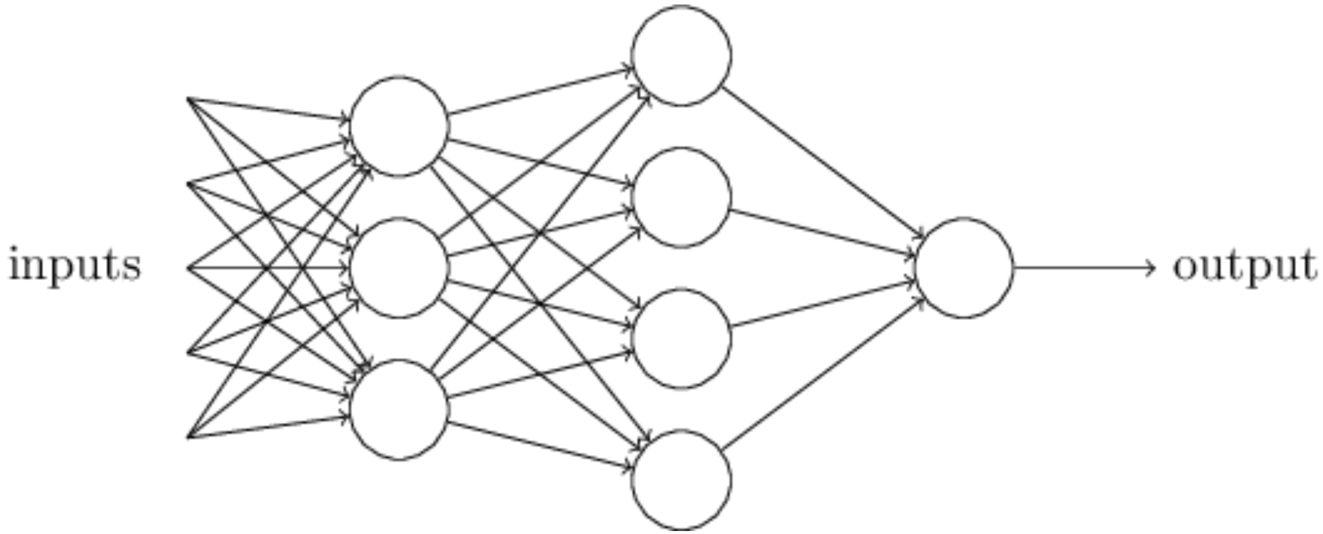
線形モデル単体では複雑な特徴を獲得できない

ならば、線形モデルを直列・並列に連結するとどうだろうか？

線形モデル

線形モデル単体では複雑な特徴を獲得できない

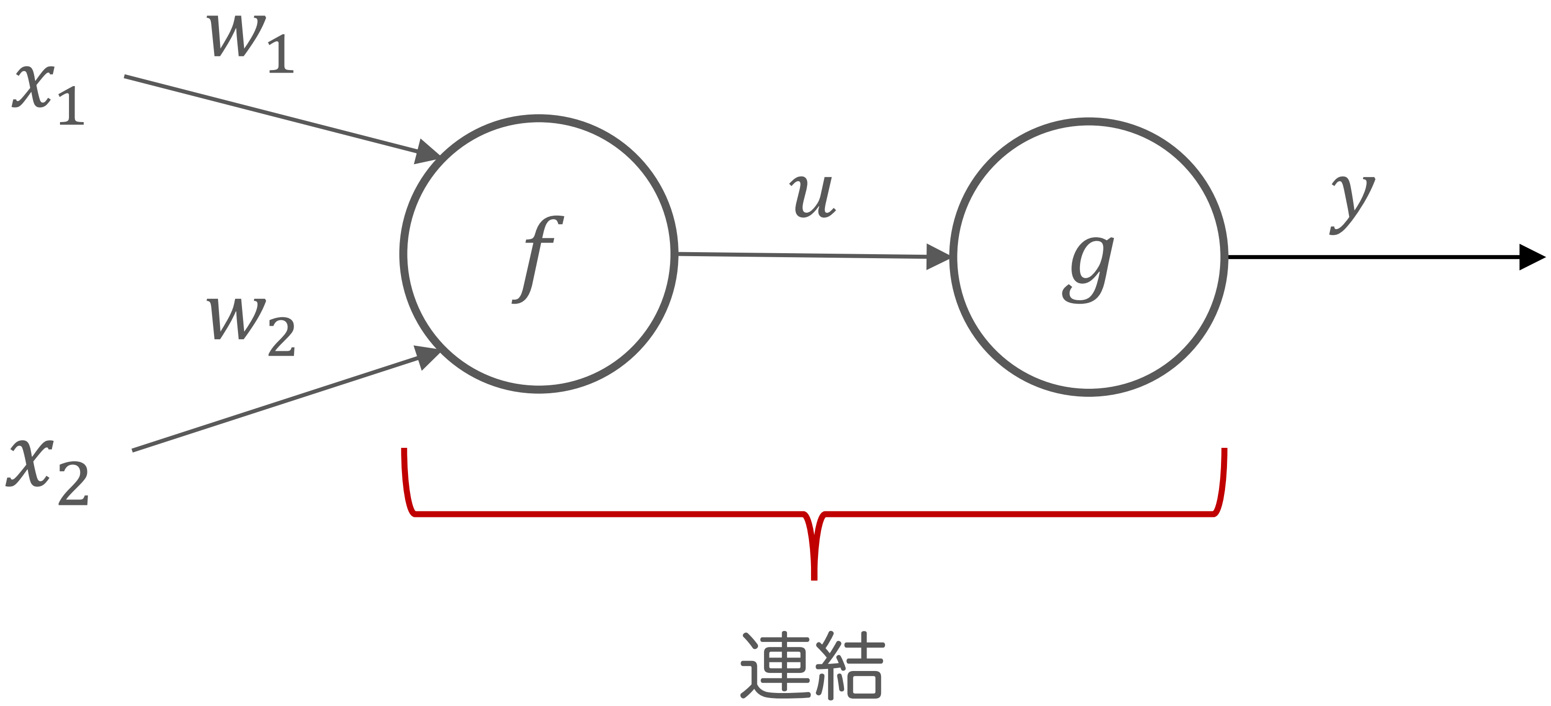
ならば、線形モデルを直列・並列に連結するとどうだろうか？



右方向に数を増やすとディープニューラルネットワークと言われる

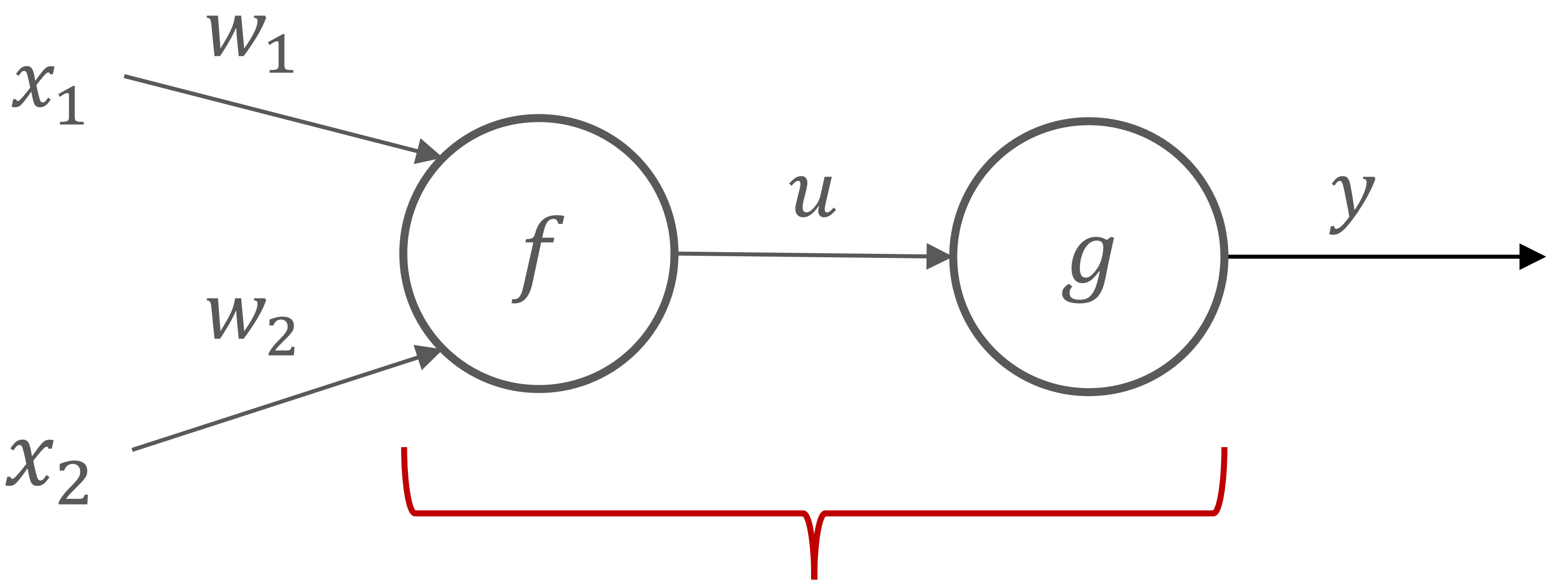
ニューラルネットワーク

「連結」という操作は数学的にどのように表現できるだろうか？



ニューラルネットワーク

「連結」という操作は数学的にどのように表現できるだろうか？



関数の合成

$$y = g(u) = g\left(f(x_1 w_1 + x_2 w_2)\right)$$

ニューラルネットワークは線形モデルを多数合成したもの

ただしその「合成」には一工夫が必要

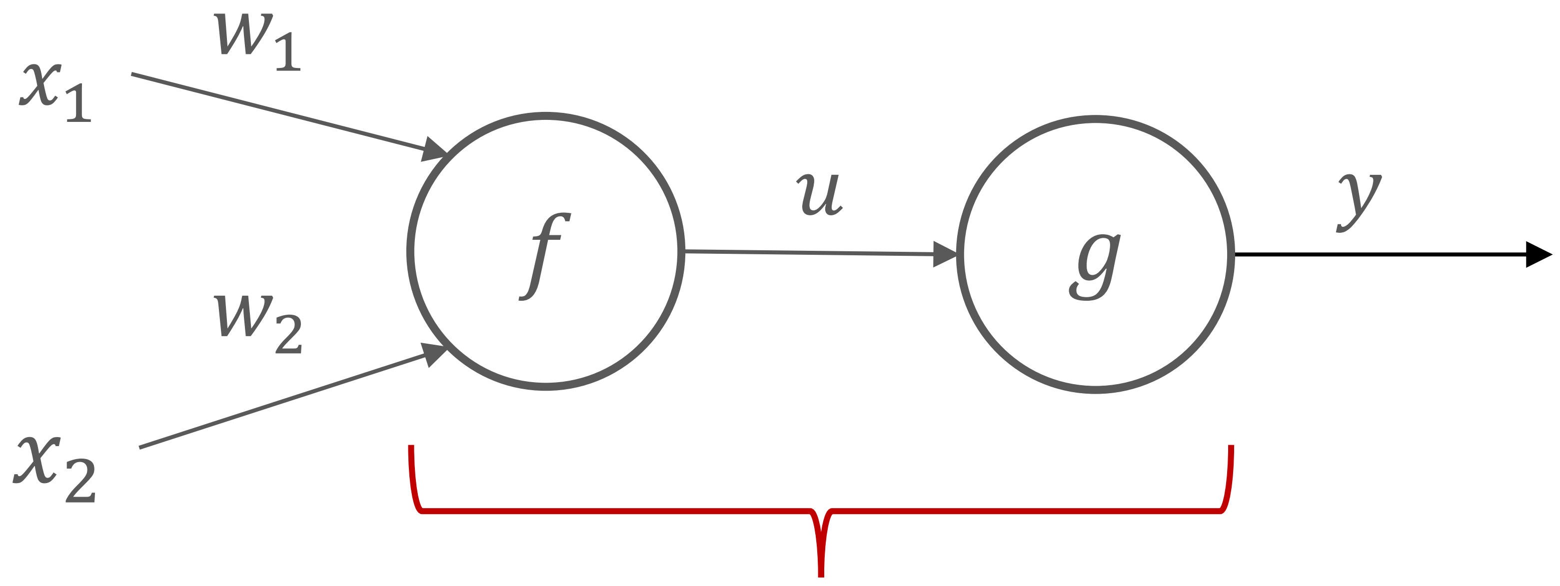
試しに以下のような変換を用いて関数の合成をしてみよう

$$f(x) = x$$

$$g(x) = x$$

恒等関数と呼ぶ

ニューラルネットワーク



関数の合成

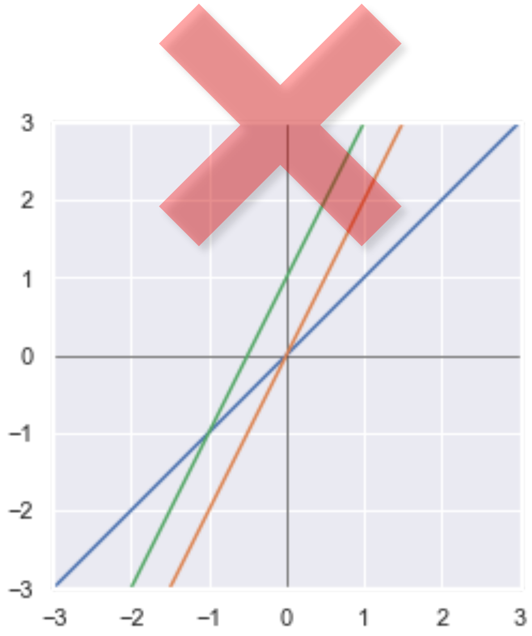
$$y = g(u) = u = f(x_1 w_1 + x_2 w_2) = x_1 w_1 + x_2 w_2$$

これでは線形モデルと変わらない

ニューラルネットワーク

f, g 全てを恒等関数のような“単純”な関数にしては意味がない

特に中間層の関数 f には **複雑な関数** を取り入れる必要がある



$f(x)$

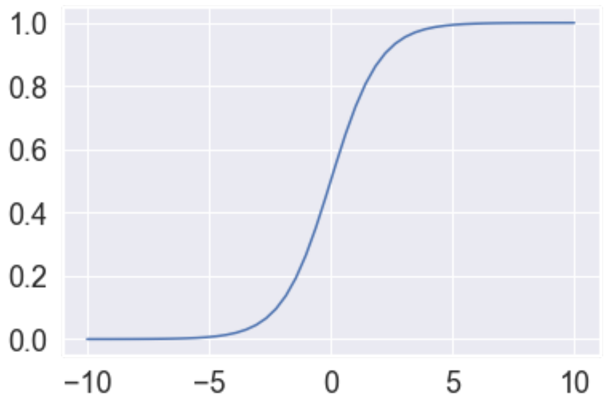
x

$f(x) = x$

$f(x) = 2x$

$f(x) = 2x + 1$

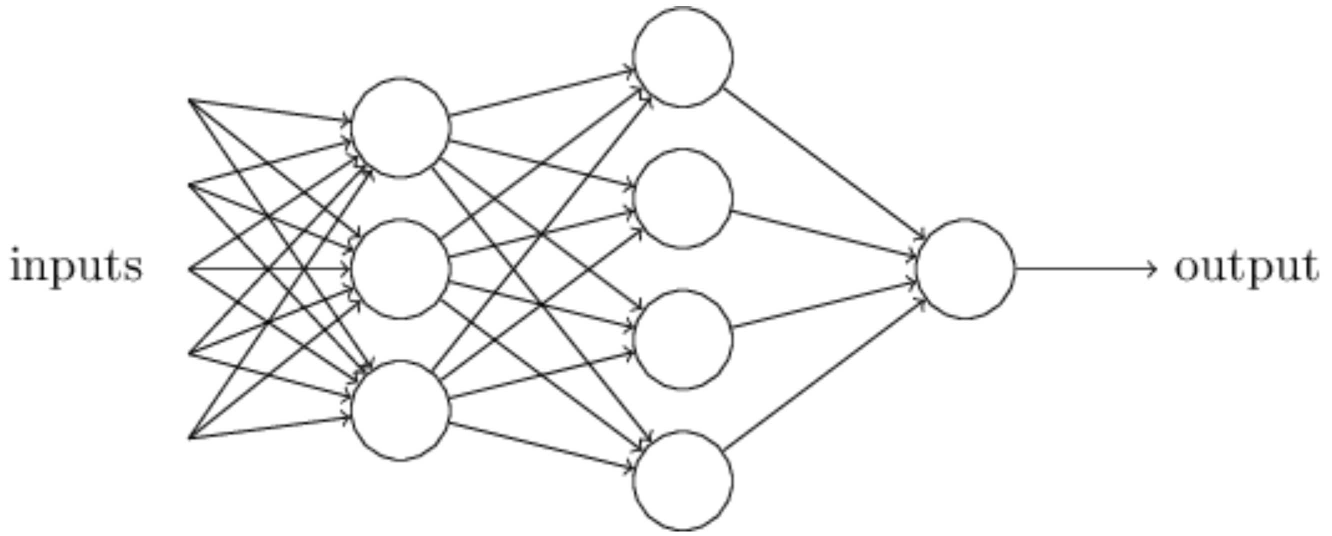
シグモイド関数



非線形関数と呼ぶ

ニューラルネットワーク

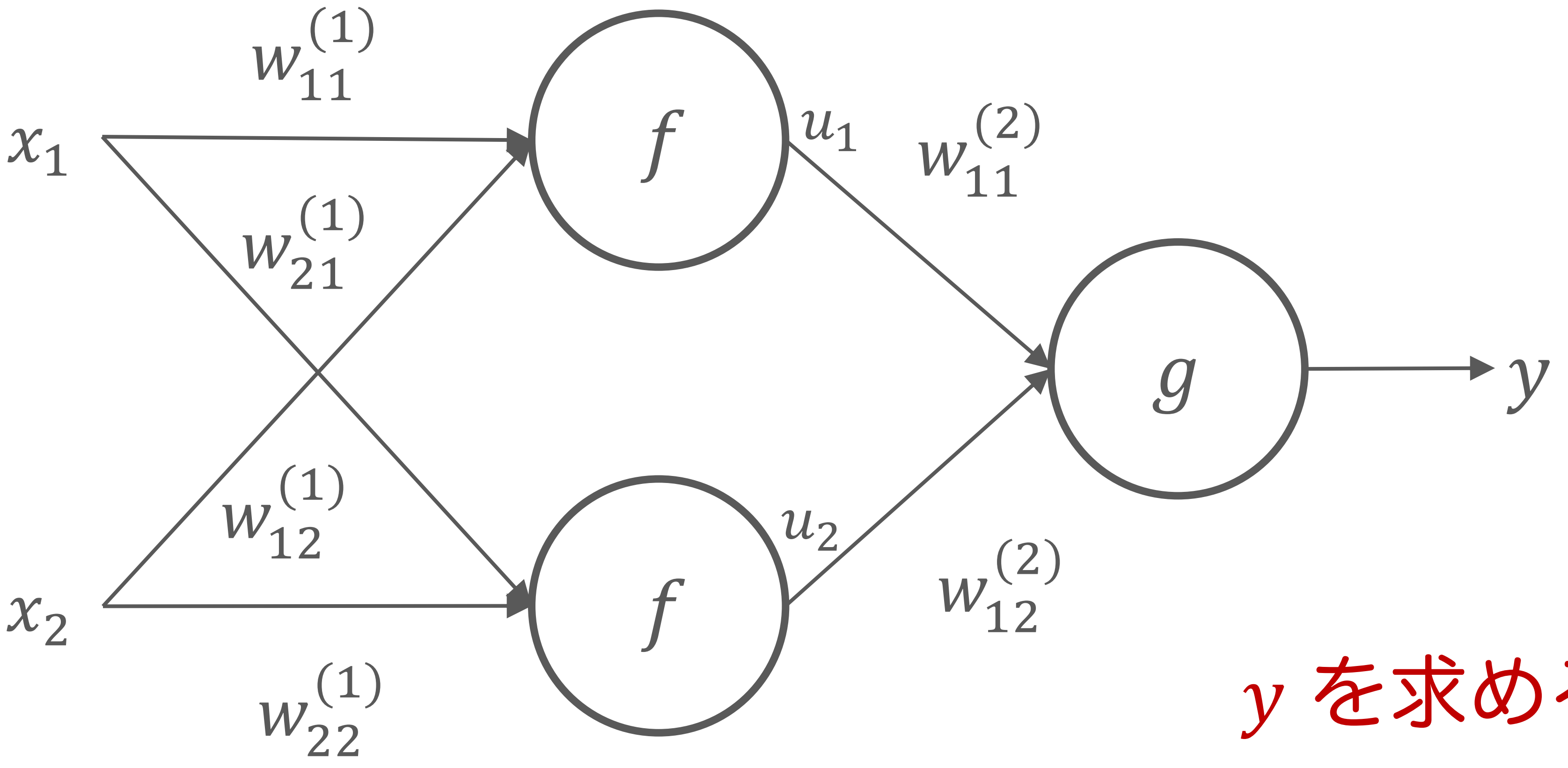
ニューラルネットワーク



線形モデル ＋ 非線形関数 ＋ 線形モデル ＋ 非線形関数 ＋・・・

複雑なデータ構造を捉えられるようになる

次はニューラルネットワークの学習（パラメータ最適化）に考えていこう



y を求めると？

※ u_1 と u_2 はそれぞれ f を通った後の値

ニューラルネットワーク

$$u_1 = f\left(w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2\right)$$

$$u_2 = f\left(w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2\right)$$

$$y = g\left(w_{11}^{(2)} u_1 + w_{12}^{(2)} u_2\right)$$

$$\Rightarrow \quad y = g\left(w_{11}^{(2)} f\left(w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2\right) + w_{12}^{(2)} f\left(w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2\right)\right)$$

誤差関数を二乗誤差に指定すると？

$$\Downarrow$$

ニューラルネットワーク

$$u_1 = f\left(w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2\right)$$

$$u_2 = f\left(w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2\right)$$

$$y = g\left(w_{11}^{(2)} u_1 + w_{12}^{(2)} u_2\right)$$

$$\Rightarrow \quad y = g\left(w_{11}^{(2)} f\left(w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2\right) + w_{12}^{(2)} f\left(w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2\right)\right)$$

誤差関数を二乗誤差に指定すると？

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(y_n - g\left(w_{11}^{(2)} f\left(w_{11}^{(1)} x_{n1} + w_{12}^{(1)} x_{n2}\right) + w_{12}^{(2)} f\left(w_{21}^{(1)} x_{n1} + w_{22}^{(1)} x_{n2}\right)\right)\right)^2$$

※ x_{n1} と x_{n2} は それぞれ n 番目のデータの 1次元目 と 2次元目 を表す

ニューラルネットワーク

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(y_n - g\left(w_{11}^{(2)} f\left(w_{11}^{(1)} x_{n1} + w_{12}^{(1)} x_{n2}\right) + w_{12}^{(2)} f\left(w_{21}^{(1)} x_{n1} + w_{22}^{(1)} x_{n2}\right)\right)\right)^2$$

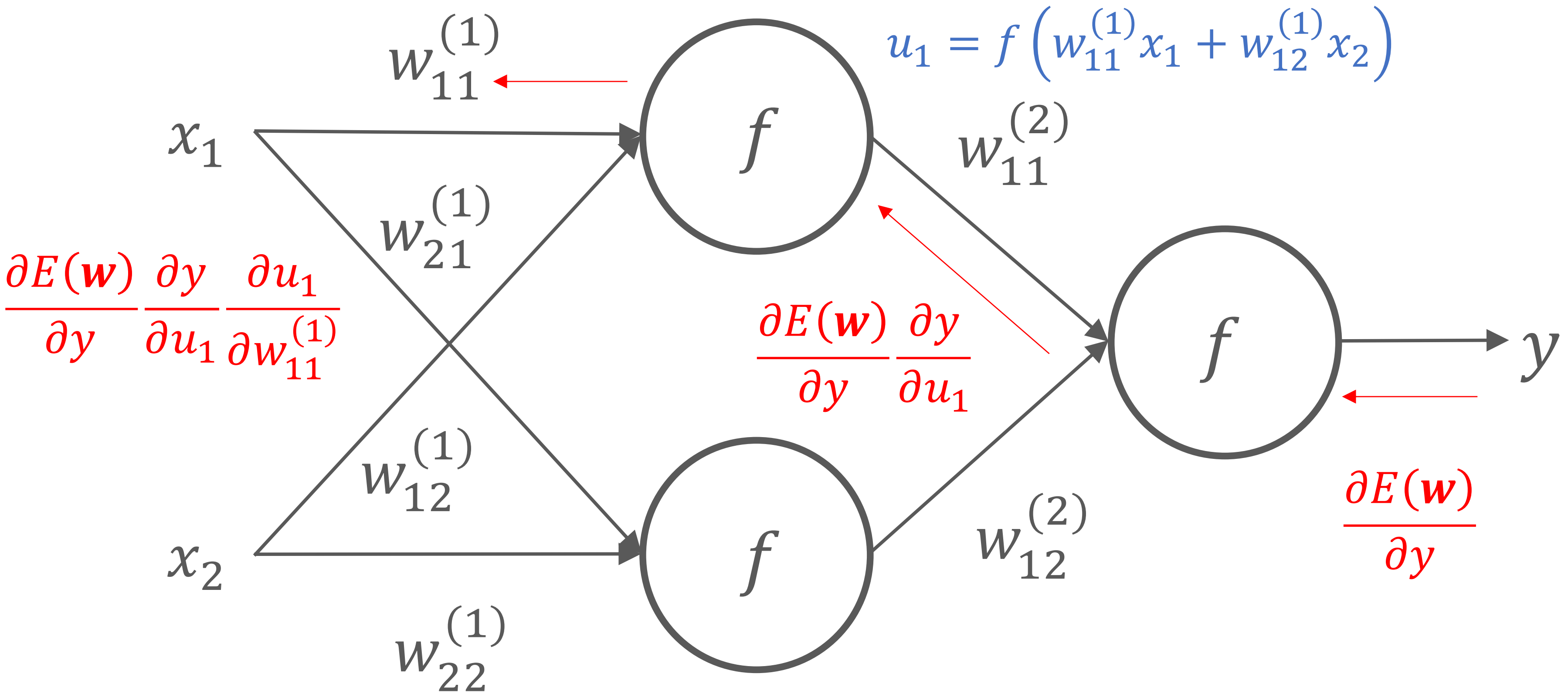
$E(\boldsymbol{w})$ を最小にする $\boldsymbol{w}$ を勾配降下法を用いて求めればよいが…

$\frac{\partial}{\partial \boldsymbol{w}} E(\boldsymbol{w})$ を求めるコストが非常に高い

偏微分の連鎖律を用いて効率的良く求める

ニューラルネットワーク

$$\frac{\partial E(\boldsymbol{w})}{\partial w_{11}^{(1)}} = \frac{\partial E(\boldsymbol{w})}{\partial y}\frac{\partial y}{\partial u_1}\frac{\partial u_1}{\partial w_{11}^{(1)}}$$



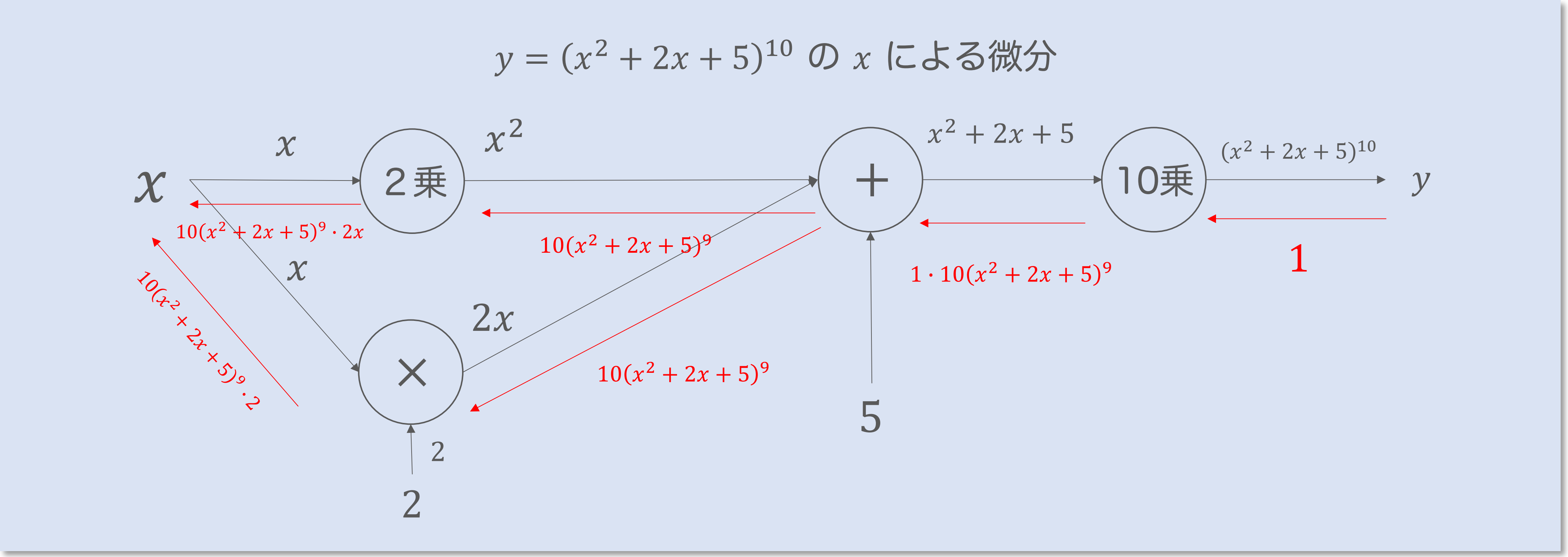
勾配降下法を用いて重み更新をすれば良い

誤差逆伝播法

偏微分の連鎖律を用いて誤差情報を逆伝播させ
重み更新に必要な勾配情報を求める手法

同じような操作をDay1の時に行なった

計算グラフによる合成関数の微分



ニューロンノード部分をさらに細かく分割して行なっていると言える

誤差逆伝播法 & 勾配法による重み更新

（1） 次のページにあるニューラルネットワーク（図1）を計算グラフで表せ

ノードは以下の4種類を用いよ

$\times$ 　 $+$ 　 sig 　 I

（2） $\dfrac{\partial}{\partial w_{22}^{(1)}} E(\boldsymbol{w})$ を求めよ（図2を参照せよ）

（3） 勾配降下法による $w_{22}^{(1)}$ に対する重みの更新則を求めよ

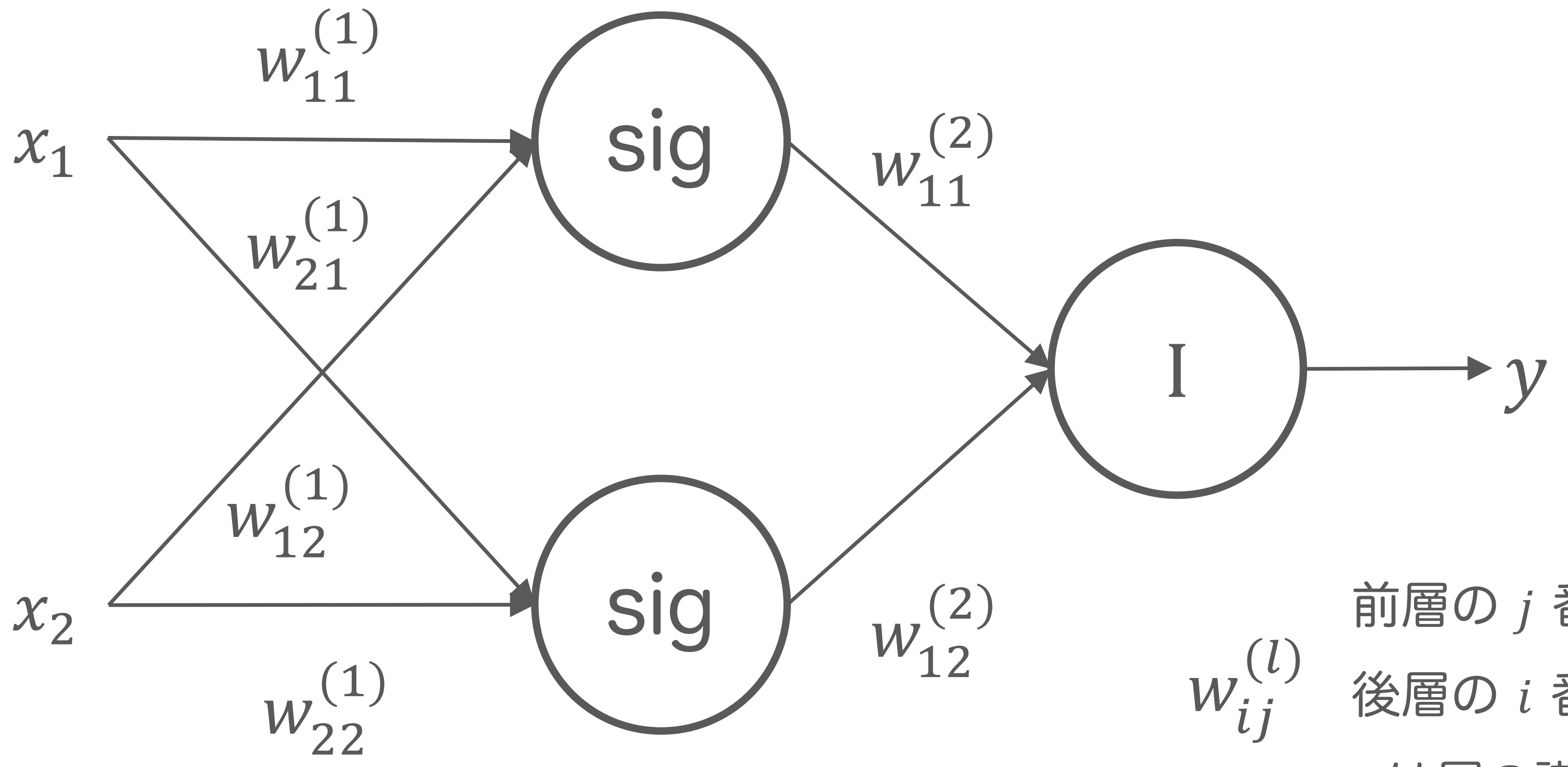
（4） 層の数が非常に大きい場合、誤差逆伝播法による重み更新は上手くいかなくなる

その理由を考察せよ

ヒント：層が深くなるにつれて勾配はどのようになる？

誤差逆伝播法 & 勾配法による重み更新

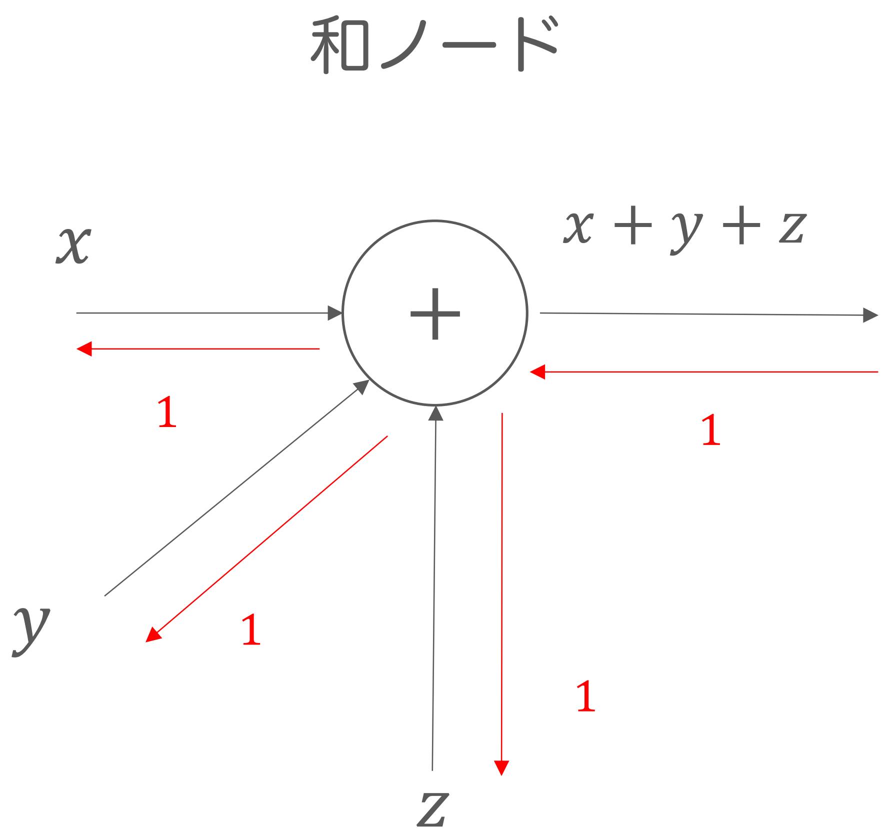
図１：活性化関数にシグモイド関数を設定したニューラルネットワーク

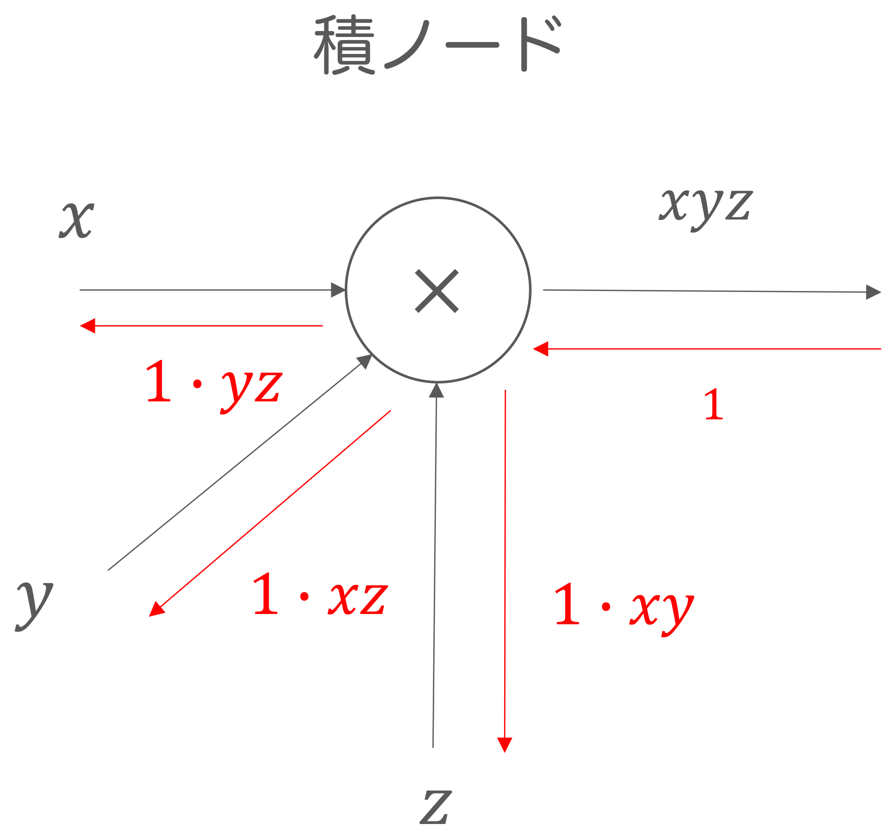


$w_{ij}^{(l)}$ 前層の j 番目のニューロンから後層の i 番目のニューロンへの重み

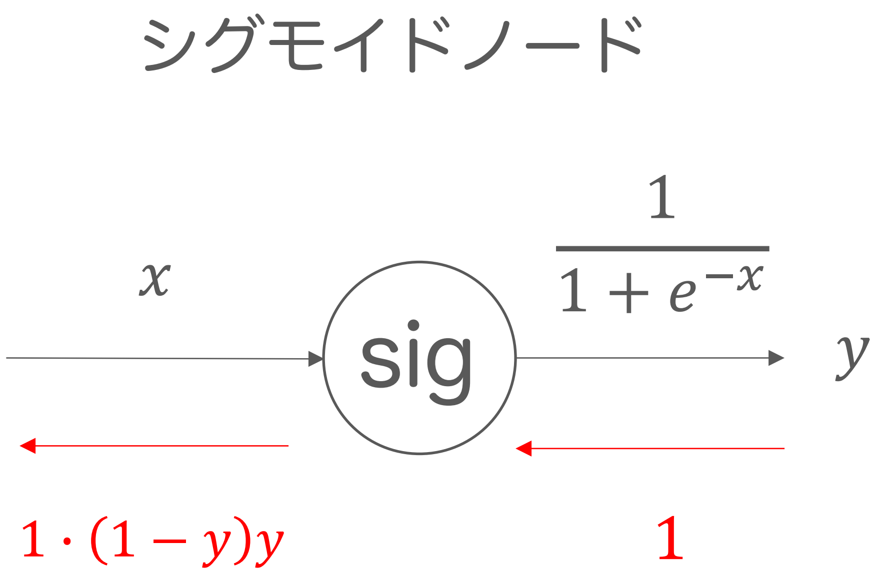
l は層の識別番号

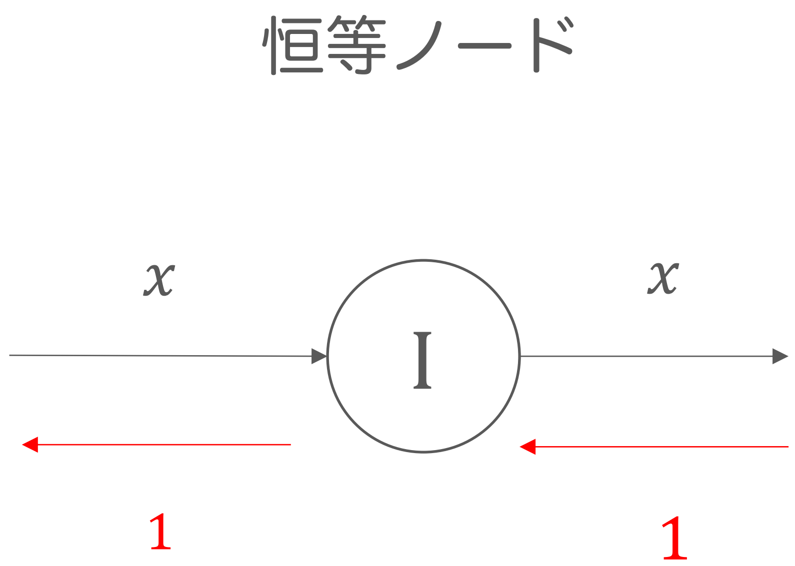
誤差逆伝播法 & 勾配法による重み更新





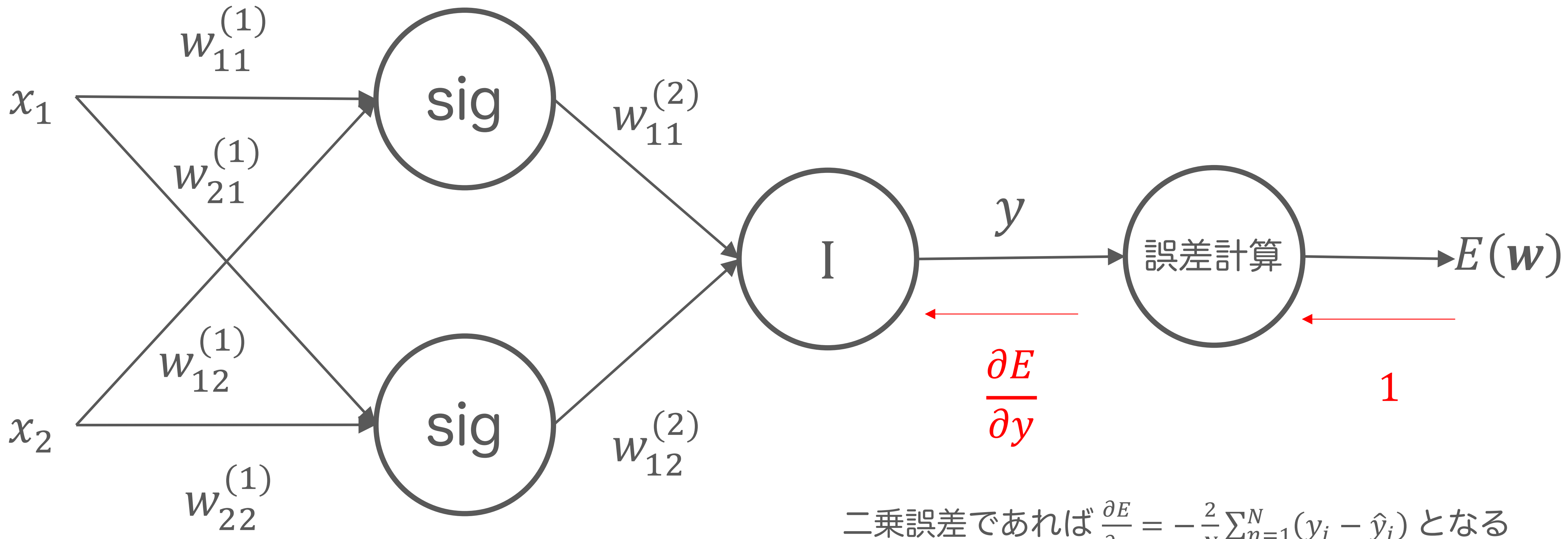
誤差逆伝播法 & 勾配法による重み更新





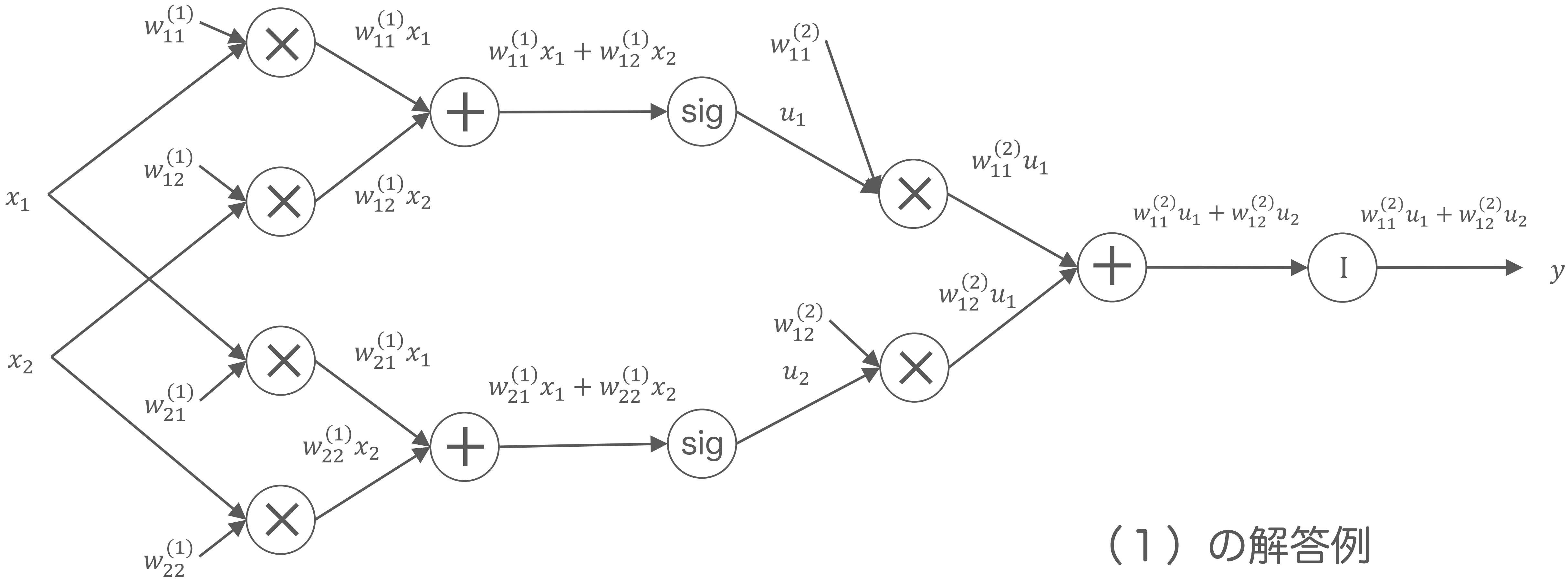
誤差逆伝播法 & 勾配法による重み更新

図２：誤差計算部分の逆伝播



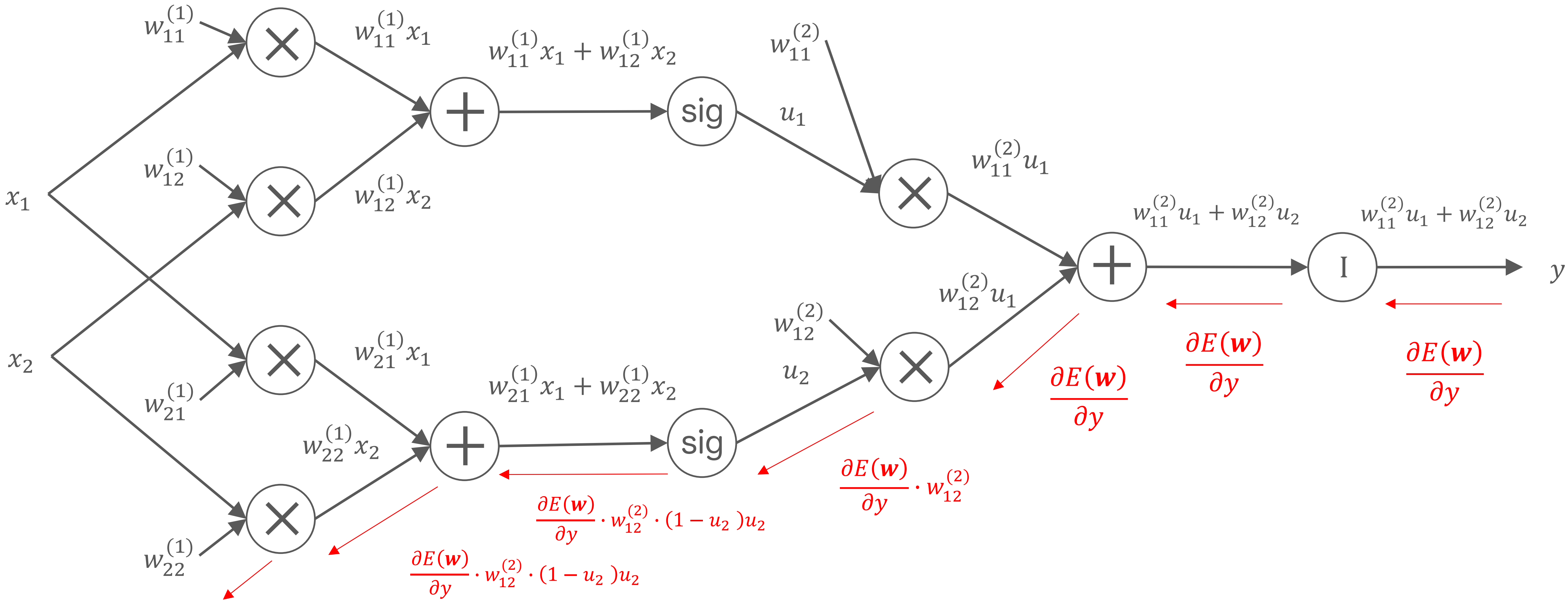
二乗誤差であれば $\frac{\partial E}{\partial y} = -\frac{2}{N}\sum_{n=1}^{N}(y_i - \hat{y}_i)$ となる

誤差逆伝播法 & 勾配法による重み更新



（１）の解答例

誤差逆伝播法 & 勾配法による重み更新



$$\frac{\partial E(\boldsymbol{w})}{\partial y} \cdot w_{12}^{(2)} \cdot (1 - u_2)u_2 \cdot x_2$$

（2）の解答

誤差逆伝播法 & 勾配法による重み更新

（3） $w_{22}^{(1)(t+1)} \leftarrow w_{22}^{(1)(t)} - \eta \dfrac{\partial E(\boldsymbol{w})}{\partial y} \cdot w_{12}^{(2)} \cdot (1-u_2)u_2 \cdot x_2$ $\qquad u_2 = \dfrac{1}{1+e^{-w_{12}^{(1)}x_1 - w_{22}^{(1)}x_2}}$

（4） $\dfrac{\partial}{\partial \boldsymbol{w}}E(\boldsymbol{w})$ の値が０に近くなり更新が途絶えてしまうから

誤差逆伝播法を用いると $\dfrac{\partial}{\partial \boldsymbol{w}}E(\boldsymbol{w})$ が「掛け算の連鎖」で表現されることに注目せよ

小数を数多く掛けた場合には値が０に消失してしまう可能性がある

特にシグモイド関数による逆伝播項 $(1-u_2)u_2$ は

最大でも 0.25 しかとれないため、この影響が大きい

この「勾配消失問題」を解消するために様々な工夫が提案されている

例）シグモイド関数の代わりにReLU関数を用いる

終わりに

機械学習・ディープラーニングのための微分・線形代数

Day 1 ：微分基礎

- 内容：様々な関数, 関数の微分
- 修了演習：シグモイド関数の計算グラフと逆伝播計算

Day 2：線形代数基礎

- 内容：偏微分, ベクトル・行列, 固有値・固有ベクトル
- 修了演習：固有値分解

Day 3：微分・線形代数の機械学習/深層学習への応用

- 内容：ベクトルによる関数の微分, 勾配降下法
- 修了演習：最小二乗法・誤差逆伝播法 & 勾配法による重み更新

機械学習・ディープラーニングのための微分・線形代数

本講座はこれにて終了となります

Day1 ～ Day３ お疲れ様でした！

本講座で身に着けた微分・線形代数の知識は

機械学習/深層学習の習得に多いに役立つこと間違いなし！

またその他のデータ分析手法でも大活躍です

ぜひこれからも学習を続けてください！

付録

スカラー関数のベクトル微分

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = \left(A + A^{\mathrm{T}}\right) \boldsymbol{x}$$

$$\boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$= \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} a_{11}x_1 + a_{12}x_2 \\ a_{21}x_1 + a_{22}x_2 \end{pmatrix} = a_{11}x_1^2 + a_{21}x_1x_2 + a_{12}x_1x_2 + a_{22}x_2^2$$

スカラー関数のベクトル微分

$$\boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = a_{11}x_1^2 + a_{21}x_1x_2 + a_{12}x_1x_2 + a_{22}x_2^2$$

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = \frac{\partial}{\partial \boldsymbol{x}} (a_{11}x_1^2 + a_{21}x_1x_2 + a_{12}x_1x_2 + a_{22}x_2^2)$$

$$= \left(\frac{\partial}{\partial x_1} (a_{11}x_1^2 + a_{21}x_1x_2 + a_{12}x_1x_2 + a_{22}x_2^2) \quad \frac{\partial}{\partial x_2} (a_{11}x_1^2 + a_{21}x_1x_2 + a_{12}x_1x_2 + a_{22}x_2^2) \right)^{\mathrm{T}}$$

$$= (2a_{11}x_1 + (a_{21} + a_{12})x_2 \quad (a_{21} + a_{12})x_1 + 2a_{22}x_2)^{\mathrm{T}}$$

$$= \begin{pmatrix} 2a_{11}x_1 + (a_{11} + a_{21})x_2 \\ (a_{21} + a_{12})x_1 + 2a_{22}x_2 \end{pmatrix}$$

スカラー関数のベクトル微分

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = \begin{pmatrix} 2a_{11}x_1 + (a_{12} + a_{21})x_2 \\ (a_{21} + a_{12})x_1 + 2a_{22}x_2 \end{pmatrix}$$

$$= \begin{pmatrix} 2a_{11} & a_{12} + a_{21} \\ a_{21} + a_{12} & 2a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$= \left\{ \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} + \begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix} \right\} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = (A + A^{\mathrm{T}})\boldsymbol{x}$$

よって確かに $\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} = (A + A^{\mathrm{T}})\boldsymbol{x}$ である

最小二乗法の理論

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\{f(\boldsymbol{x}_n, \boldsymbol{w}) - y_n\}^2$$

$$= \frac{1}{N}\{f(\boldsymbol{x}_1, \boldsymbol{w}) - y_1\}^2 + \frac{1}{N}\{f(\boldsymbol{x}_2, \boldsymbol{w}) - y_2\}^2 + \cdots + \frac{1}{N}\{f(\boldsymbol{x}_N, \boldsymbol{w}) - y_N\}^2$$

$$\frac{\partial E(\boldsymbol{w})}{\partial w_1} = \frac{\partial}{\partial w_1}\frac{1}{N}\{f(\boldsymbol{x}_1, \boldsymbol{w}) - y_1\}^2 + \frac{\partial}{\partial w_1}\frac{1}{N}\{f(\boldsymbol{x}_2, \boldsymbol{w}) - y_2\}^2 + \cdots + \frac{\partial}{\partial w_1}\frac{1}{N}\{f(\boldsymbol{x}_N, \boldsymbol{w}) - y_N\}^2$$

全データに対する誤差$E(\boldsymbol{w})$は、各データに対する誤差を足し合わせたとみることができる。誤差の微分も、やはり各データに対する誤差の微分の総和になる

最小二乗法の理論

$$\frac{\partial}{\partial w_1}\frac{1}{N}\{f(\boldsymbol{x}_n, \boldsymbol{w}) - y_n\}^2 = \frac{1}{N}\frac{\partial}{\partial w_1}\{w_0 + w_1 x_{n1} + w_2 x_{n2} + \cdots + w_d x_{nd} - y_n\}^2$$

$z_n = w_0 + w_1 x_{n1} + w_2 x_{n2} + \cdots + w_d x_{nd} - y_n$とすると合成関数の微分のルールより

$$\frac{1}{N}\frac{\partial}{\partial w_1}{z_n}^2 = \frac{1}{N}\frac{\partial {z_n}^2}{\partial z_n}\frac{\partial z_n}{\partial w_1} = \frac{1}{N}\frac{\partial {z_n}^2}{\partial z_n}\frac{\partial}{\partial w_1}(w_0 + w_1 x_{n1} + w_2 x_{n2} + \cdots + w_d x_{nd} - y_n)$$

$$= \frac{1}{N} \cdot 2z_n \cdot x_{n1}$$

$$= \frac{2}{N} z_n x_{n1} = \frac{2}{N}(w_0 + w_1 x_{n1} + w_2 x_{n2} + \cdots + w_d x_{nd} - y_n) x_{n1}$$

最小二乗法の理論

$$\frac{\partial}{\partial w_1}\frac{1}{N}\{f(\boldsymbol{x}_n,\boldsymbol{w})-y_n\}^2=\frac{2}{N}(w_0+w_1x_{n1}+w_2x_{n2}+\cdots+w_dx_{nd}-y_n)x_{n1}=\frac{2}{N}z_nx_{n1}$$

$$\frac{\partial E(\boldsymbol{w})}{\partial w_1}=\frac{\partial}{\partial w_1}\frac{1}{N}\sum_{n=1}^{N}\{f(\boldsymbol{x}_n,\boldsymbol{w})-y_n\}^2=\sum_{n=1}^{N}\frac{\partial}{\partial w_1}\frac{1}{N}\{f(\boldsymbol{x}_n,\boldsymbol{w})-y_n\}^2$$

$$=\frac{2}{N}\sum_{n=1}^{N}z_nx_{n1}$$

ここで$z_n=w_0+w_1x_{n1}+w_2x_{n2}+\cdots+w_dx_{nd}-y_n$

最小二乗法の理論

w_0に対する偏微分は、
$x_{n0}=1$として考えればよい

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = \left(\frac{\partial E(\boldsymbol{w})}{\partial w_0}, \frac{\partial E(\boldsymbol{w})}{\partial w_1}, \cdots, \frac{\partial E(\boldsymbol{w})}{\partial w_d}\right)^{\mathrm{T}} = \frac{2}{N}\left(\sum_{n=1}^{N} z_n, \sum_{n=1}^{N} z_n x_{n1}, \cdots, \sum_{n=1}^{N} z_n x_{nd}\right)^{\mathrm{T}}$$

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0 \iff \frac{2}{N}\left(\sum_{n=1}^{N} z_n, \sum_{n=1}^{N} z_n x_{n1}, \cdots, \sum_{n=1}^{N} z_n x_{nd}\right)^{\mathrm{T}} = \mathbf{0}$$

$$\iff \left(\sum_{n=1}^{N} z_n, \sum_{n=1}^{N} z_n x_{n1}, \cdots, \sum_{n=1}^{N} z_n x_{nd}\right)^{\mathrm{T}} = \mathbf{0}$$

両辺を $\frac{N}{2}$ 倍

$$z_n = (w_0 + w_1 x_{n1} + w_2 x_{n2} + \cdots + w_d x_{nd} - y_n)$$

最小二乗法の理論

$$\left(\sum_{n=1}^{N} z_n\,,\sum_{n=1}^{N} z_n x_{n1}\,,\cdots,\sum_{n=1}^{N} z_n x_{nd}\right)^{\mathrm{T}} = \mathbf{0}$$

$$\Leftrightarrow \begin{pmatrix} z_1 + z_2 + \cdots + z_N \\ z_1 x_{11} + z_2 x_{21} + \cdots + z_N x_{N1} \\ z_1 x_{12} + z_2 x_{22} + \cdots + z_N x_{N2} \\ \vdots \\ z_1 x_{1d} + z_2 x_{2d} + \cdots + z_N x_{Nd} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_{11} & x_{21} & x_{31} & \cdots & x_{N1} \\ x_{12} & x_{22} & x_{32} & \cdots & x_{N2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1d} & x_{2d} & x_{3d} & \cdots & x_{Nd} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

最小二乗法の理論

$$\left(\sum_{n=1}^{N} z_n, \sum_{n=1}^{N} z_n x_{n1}, \cdots, \sum_{n=1}^{N} z_n x_{nd}\right)^{\mathrm{T}} = \mathbf{0}$$

$$\Longleftrightarrow \quad \begin{pmatrix} z_1 + z_2 + \cdots + z_N \\ z_1 x_{11} + z_2 x_{21} + \cdots + z_N x_{N1} \\ z_1 x_{12} + z_2 x_{22} + \cdots + z_N x_{N2} \\ \vdots \\ z_1 x_{1d} + z_2 x_{2d} + \cdots + z_N x_{Nd} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

内積

$$\Longleftrightarrow \quad \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_{11} & x_{21} & x_{31} & \cdots & x_{N1} \\ x_{12} & x_{22} & x_{32} & \cdots & x_{N2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1d} & x_{2d} & x_{3d} & \cdots & x_{Nd} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

最小二乗法の理論

$$\left(\sum_{n=1}^{N} z_n, \sum_{n=1}^{N} z_n x_{n1}, \cdots, \sum_{n=1}^{N} z_n x_{nd}\right)^{\mathrm{T}} = \mathbf{0}$$

$$\Leftrightarrow \begin{pmatrix} z_1 + z_2 + \cdots + z_N \\ z_1 x_{11} + z_2 x_{21} + \cdots + z_N x_{N1} \\ z_1 x_{12} + z_2 x_{22} + \cdots + z_N x_{N2} \\ \vdots \\ z_1 x_{1d} + z_2 x_{2d} + \cdots + z_N x_{Nd} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

内積

$$\Leftrightarrow \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_{11} & x_{21} & x_{31} & \cdots & x_{N1} \\ x_{12} & x_{22} & x_{32} & \cdots & x_{N2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1d} & x_{2d} & x_{3d} & \cdots & x_{Nd} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

最小二乗法の理論

$$\begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_{11} & x_{21} & x_{31} & \cdots & x_{N1} \\ x_{12} & x_{22} & x_{32} & \cdots & x_{N2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1d} & x_{2d} & x_{3d} & \cdots & x_{Nd} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ 1 & x_{31} & x_{32} & \cdots & x_{3d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{Nd} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \Phi^{\mathrm{T}} \boldsymbol{z} = \boldsymbol{0}$$

最小二乗法の理論

$$\Phi^{\mathrm{T}}\boldsymbol{z} = \Phi^{\mathrm{T}}\begin{pmatrix} w_0 + w_1 x_{11} + w_2 x_{12} + \cdots + w_d x_{1d} - y_1 \\ w_0 + w_1 x_{21} + w_2 x_{22} + \cdots + w_d x_{2d} - y_2 \\ w_0 + w_1 x_{31} + w_2 x_{32} + \cdots + w_d x_{3d} - y_3 \\ \vdots \\ w_0 + w_1 x_{N1} + w_2 x_{N2} + \cdots + w_d x_{Nd} - y_N \end{pmatrix}$$

$$= \Phi^{\mathrm{T}}\left\{\begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ 1 & x_{31} & x_{32} & \cdots & x_{3d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{Nd} \end{pmatrix}\begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_d \end{pmatrix} - \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{pmatrix}\right\}$$

$$= \Phi^{\mathrm{T}}(\Phi\boldsymbol{w} - \boldsymbol{y})$$

最小二乗法の理論

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0 \iff \Phi^{\mathrm{T}}(\Phi \boldsymbol{w} - \boldsymbol{y}) = \boldsymbol{0}$$

$$\Phi = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ 1 & x_{31} & x_{32} & \cdots & x_{3d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{Nd} \end{pmatrix}, \boldsymbol{w} = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_d \end{pmatrix}, \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{pmatrix}$$

最小二乗法の理論

結局

$$\Phi^{\mathrm{T}}(\Phi\boldsymbol{w} - \boldsymbol{y}) = \boldsymbol{0}$$

を解けばよい。

$\Phi^{\mathrm{T}}\Phi\boldsymbol{w} - \Phi^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{0}$より、$\Phi^{\mathrm{T}}\Phi\boldsymbol{w} = \Phi^{\mathrm{T}}\boldsymbol{y}$となり

$$\boldsymbol{w} = \left(\Phi^{\mathrm{T}}\Phi\right)^{-1}\Phi^{\mathrm{T}}\boldsymbol{y}$$

ここでは$\Phi^{\mathrm{T}}\Phi$が逆行列を持つとした

を得る。

この式を正規方程式と呼ぶ。