知能情報システム

第8回 機械学習

教師あり学習 (その1)

秦野 亮

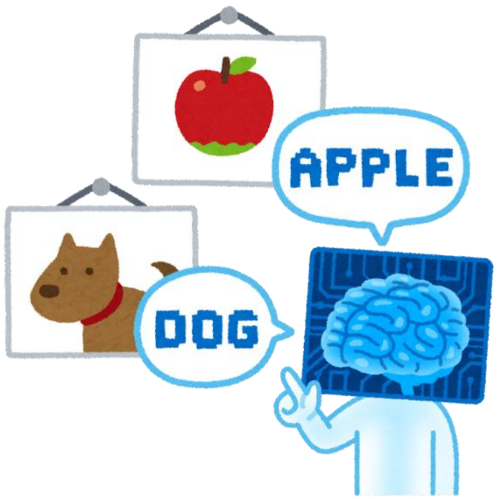
東京理科大学 経営システム工学科

2025

本日の講義内容

教師あり学習について学ぶ。

1. 教師あり学習の概要
2. 基本的な線形分類モデル



モデル関連の図表は以下の文献に依るものです：
S. Raschka, V. Mirjalili, Python Machine Learning 3rd Ed., Impress, 2020.

教師あり学習 再掲

- システム外部からの教師信号（ラベル）による教示によって学習を行う。

例：手書き数字の認識

学習フェーズ

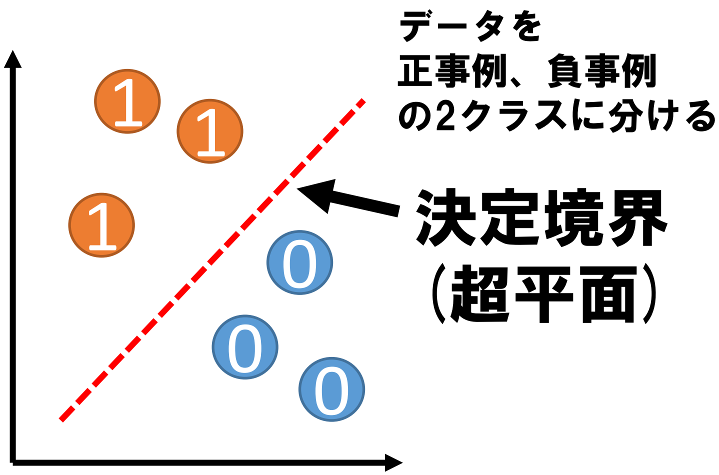
サンプル

ラベル　1　2　3　7　1　2　1　7

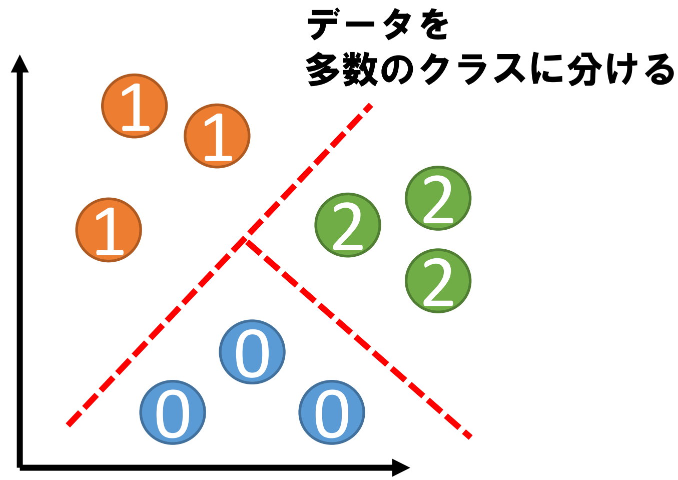
予測フェーズ

これは何でしょう？

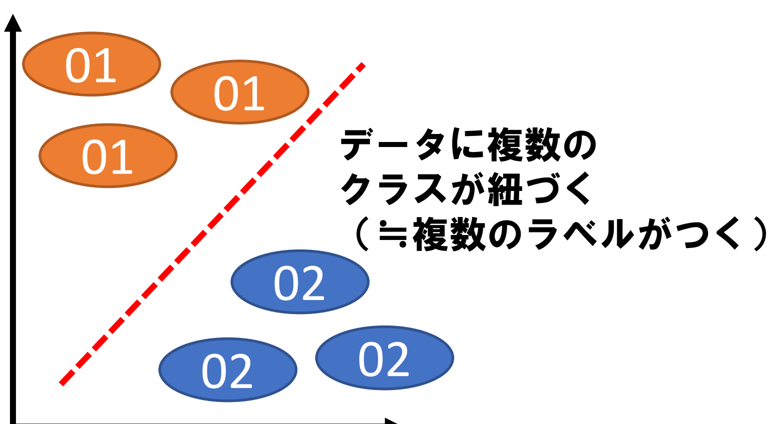
主な教師あり学習の分類



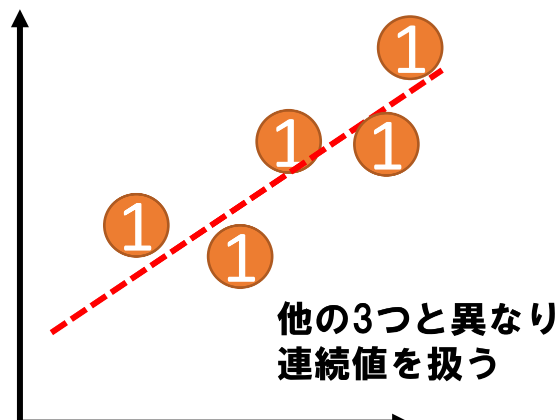
二値分類



多値分類



多ラベル分類



回帰

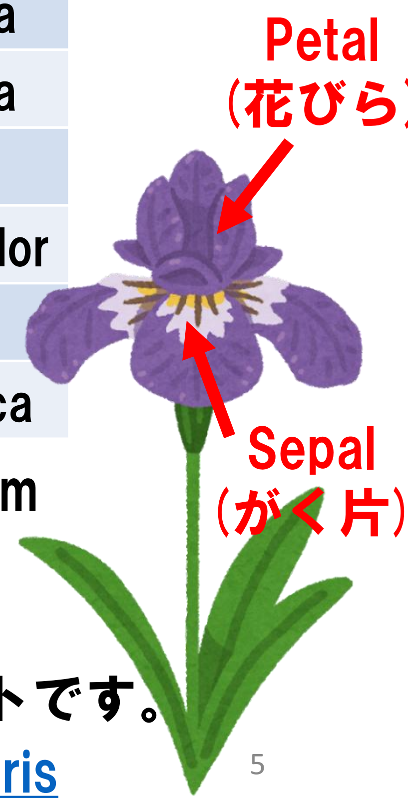
データセットの形式

特徴量（説明変数）　　ラベル（目的変数）

Sample No.	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
⋮	⋮	⋮	⋮	⋮	⋮
51	7.0	3.2	4.7	1.4	Versicolor
⋮	⋮	⋮	⋮	⋮	⋮
150	5.9	3.0	5.1	1.8	Virginica

サンプル（インスタンス）

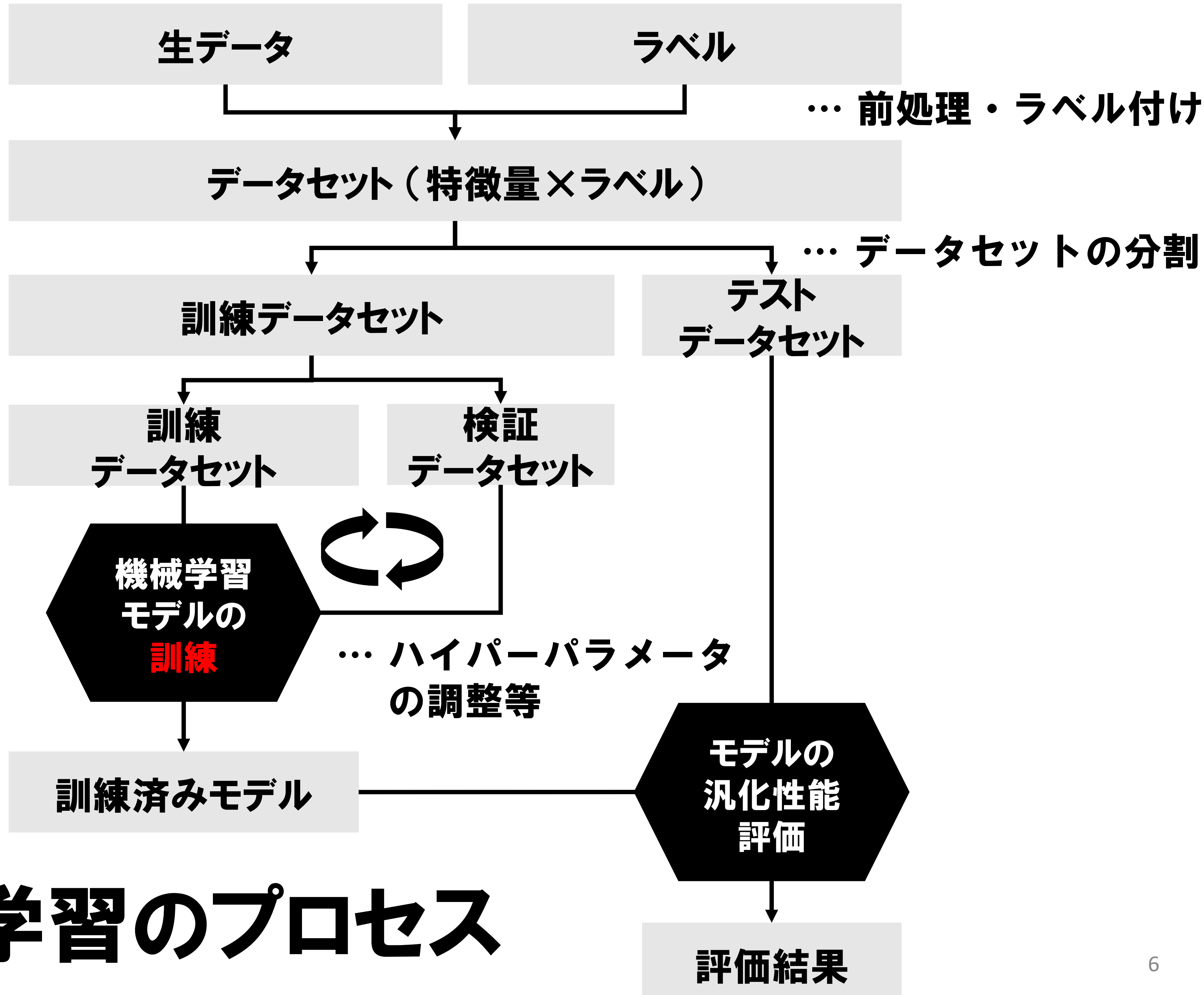
※width/lengthの**単位**はcm



上記の例は「あやめ（花菖蒲）」の分類に関するデータセットです。

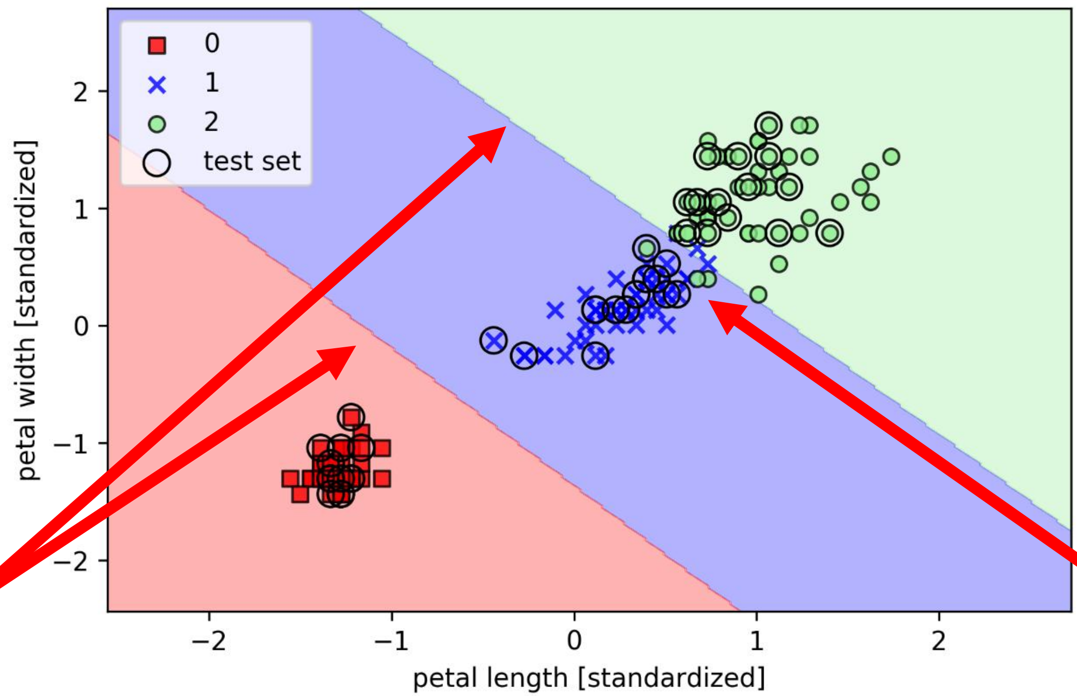
出典: Iris Dataset, https://archive.ics.uci.edu/ml/datasets/Iris

学習のプロセス


生データ
ラベル
… 前処理・ラベル付け
データセット（特徴量×ラベル）
… データセットの分割
訓練データセット
テスト
データセット
訓練
データセット
検証
データセット
機械学習
モデルの
訓練
… ハイパーパラメータ
の調整等
訓練済みモデル
モデルの
汎化性能
評価
評価結果

うまく訓練できたモデルの例

前出のデータセットから、
以下の様な多値分類モデルを得ることができる。



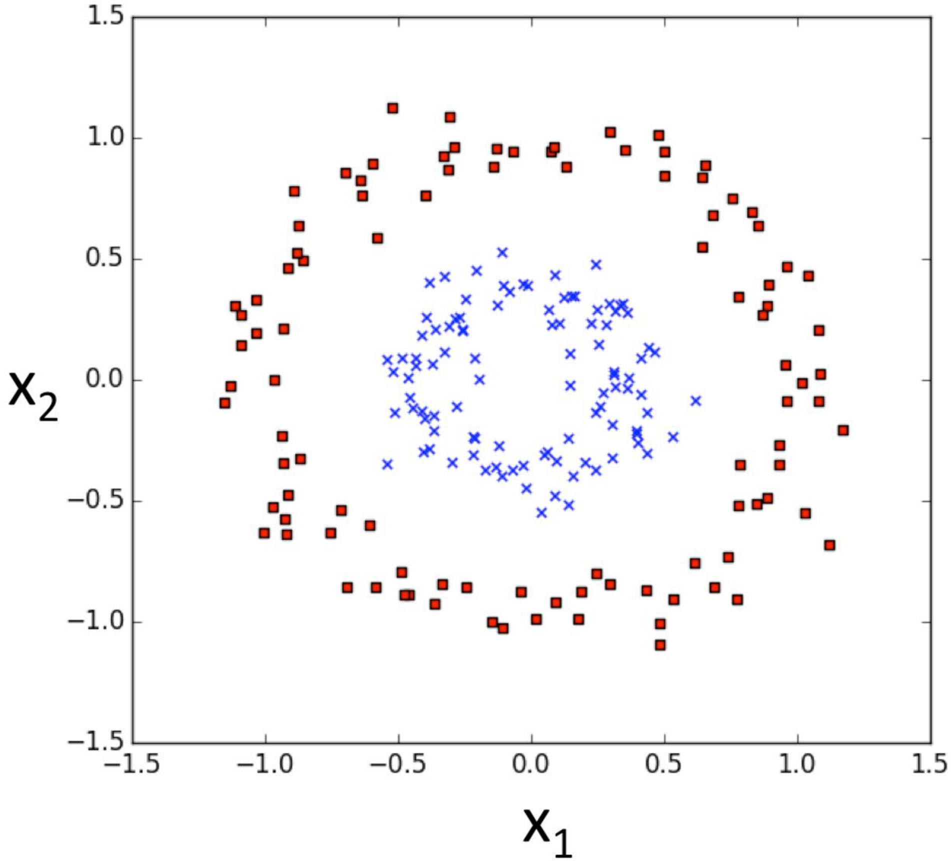
決定境界

幾つか
例外もある

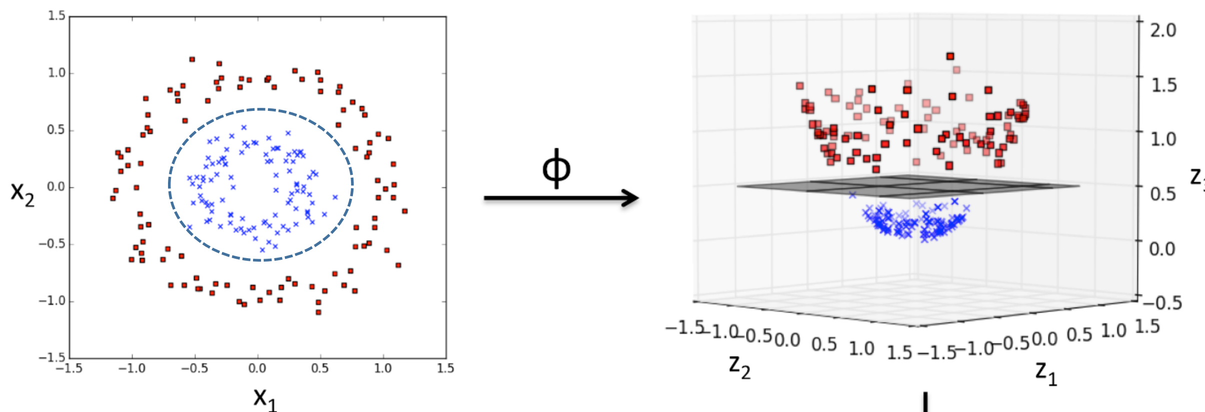
この様な分類の仕方を**線形**分離と呼ぶ。

Quiz: 訓練してみよう

線形分離できるでしょうか？

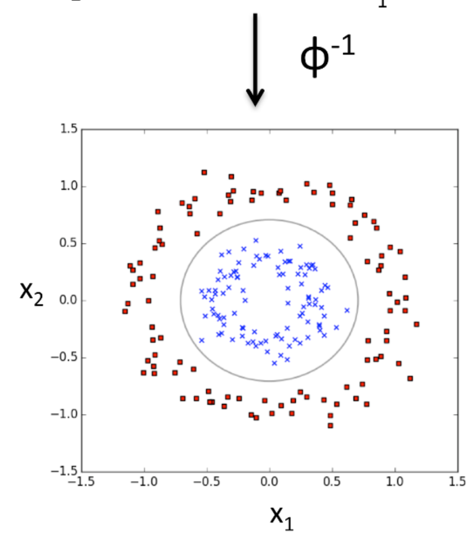


解答例



普通にやると**出来ません。**
決定境界が曲線を含む
≒**非線形分離**です。

なお、**特別**な射影関数ϕを
導入し、高次元空間に
データを写像すれば
そこで**線形分離出来る**
…こともあります。



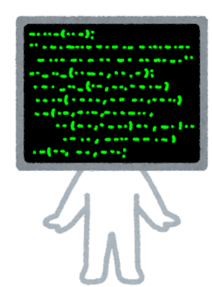
カーネル法(SVM)を用いた
分類の例です。

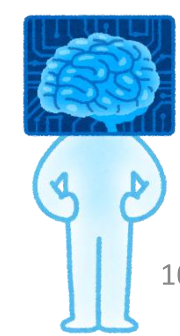
様々な教師あり学習のモデル

- パーセプトロン、ロジスティック回帰
- ナイーブベイズ分類器、隠れマルコフモデル
- サポートベクトルマシン、条件付き確率場
- 決定木、帰納論理プログラミング
- アンサンブル学習
  - Bugging、ランダムフォレスト、XGBoost、LightGBM
- （深層）ニューラルネットワーク etc.

他にもいっぱいあります。
用途に応じて使い分けます。

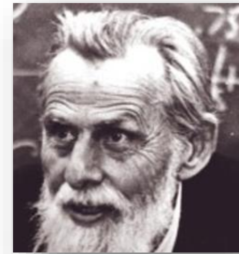
「決定境界の引き方」
が違うだけのかな？





Perceptron

パーセプトロン

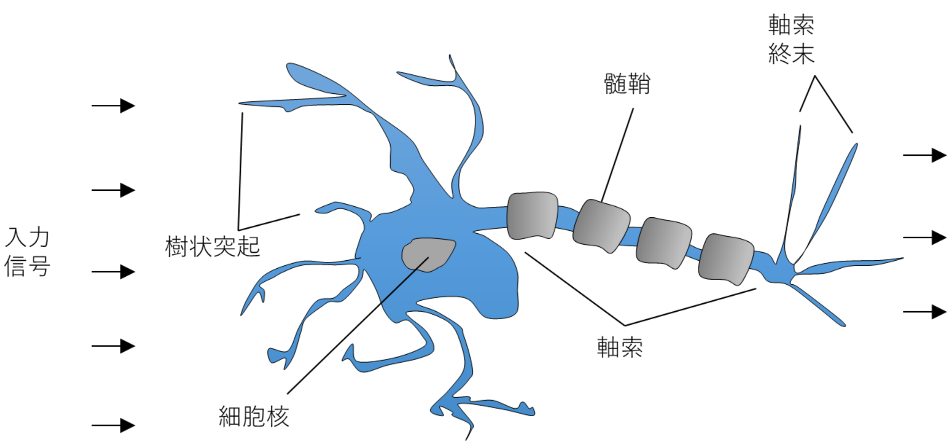


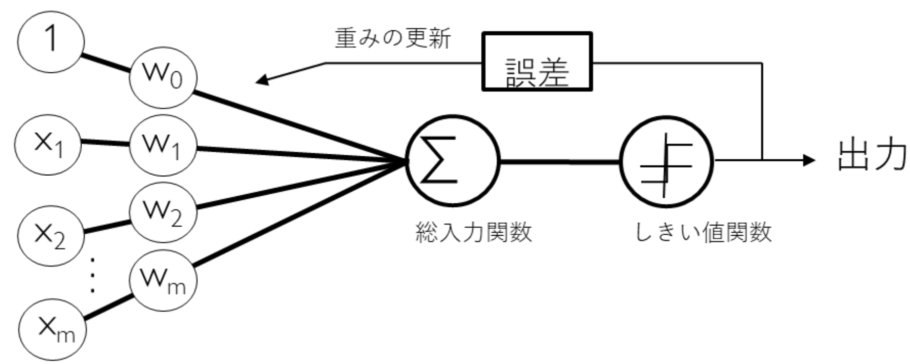
W. McCulloch

W. Pitts

F. Rosenblatt

- 人間の脳細胞の基本単位となる「ニューロン」の動作を模した機械学習モデル
- 1943年のW. McCulloch & W. Pittsらにより提案された「形式ニューロン」のアイデアに基づき、1958年にF. Rosenblattが学習規則を与えた。





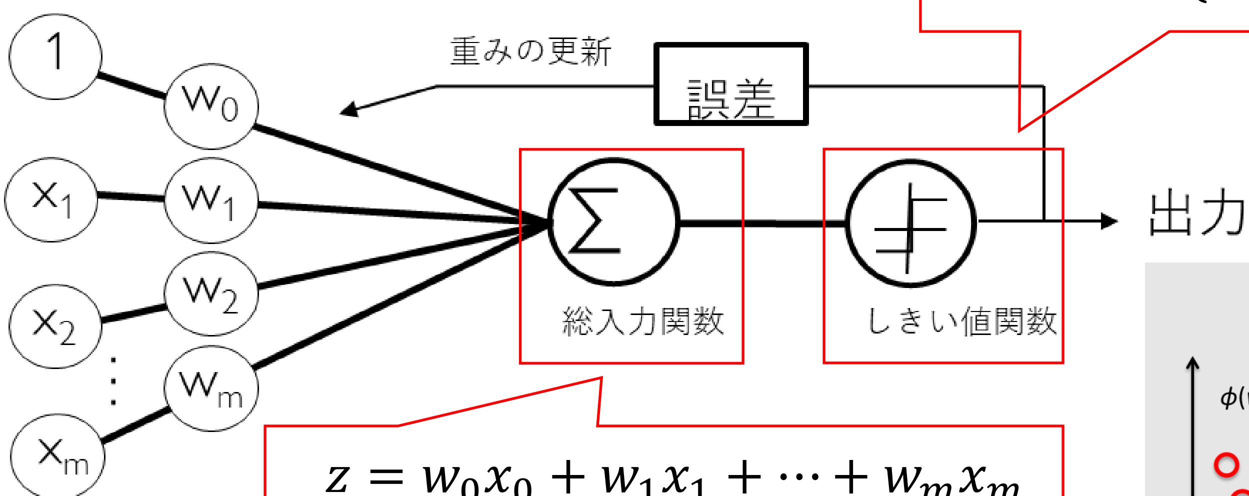
参考記事: https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

パーセプトロンのモデル

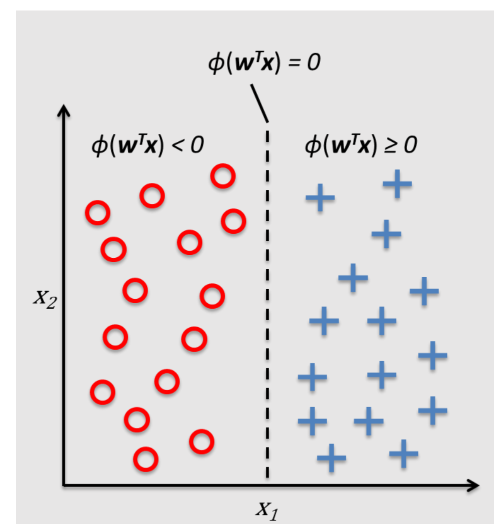
刺激（≒入力）によってニューロンが
発火する or しない（≒出力）
仕組みを模している。

単位ステップ関数

$$\phi(z) = \begin{cases} 1 & \text{if } z \geq 0, \\ -1 & \text{if } z < 0. \end{cases}$$



$$z = w_0 x_0 + w_1 x_1 + \cdots + w_m x_m$$
$$= \sum_{j=0}^{m} w_j x_j = \boldsymbol{w}^T \boldsymbol{x}$$



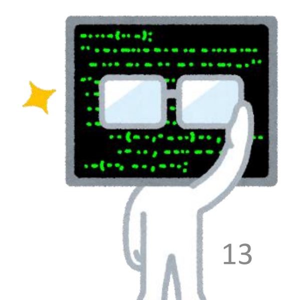
※上記の図ではモデル式導出の関係で $x_0 = 1$ としています。

パーセプトロンの学習規則

1. 重みベクトルwを初期化する（0 or 乱数）
2. 各訓練データx^iごとに以下の計算をする：
   1. 予測値$\widehat{y^i}$を計算する：
   $$\widehat{y^i} = \phi(z^i),$$
   ただし $z^i = \boldsymbol{w}^T \boldsymbol{x}^i$である。
   2. 重みベクトルw中の各重みw_jを更新する：
   $$w_j := w_j + \Delta w_j$$
   $$\Delta w_j = \eta \left(y^i - \widehat{y^i} \right) x_j^i$$

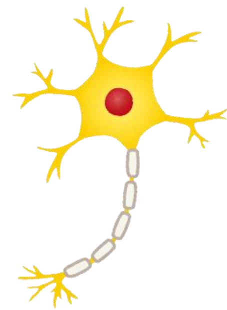
   ただし、ηは学習率（0～1の実数値）、
   y^iは訓練データx^iの真のラベル、
   $\widehat{y^i}$は訓練データx^iの予測値のラベルである。

要するに、試しに予測して駄目なら重みwを更新するということ。



ニューロン1粒の気持ち
になって学習してみよう。

手を動かしてみよう！

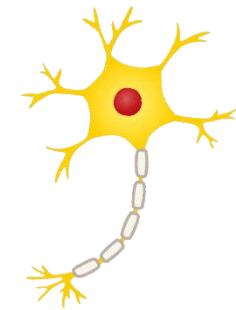


- いま、重み$w = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix}$のパーセプトロンのモデルが与えられている。
- このモデルを学習率$\eta = 0.5$で学習させたい。そこで訓練データ$x = \begin{bmatrix} 1 \\ 1 \\ 0.5 \end{bmatrix}$と真のラベル$y = 1$のペアを用いて学習を試みて、以下の予測値$\hat{y}$が得られた場合の重み$w$の更新を行いなさい。

（1） $\hat{y} = 1$
（2） $\hat{y} = -1$

変数の表記の意味や
添字に注意しよう。

解答例



与えられた条件下で以下を解けば良い。

$$w_j := w_j + \Delta w_j \quad (\text{今回は} j \in \{0,1,2\})$$
$$\Delta w_j = \eta(y - \hat{y})x_j$$

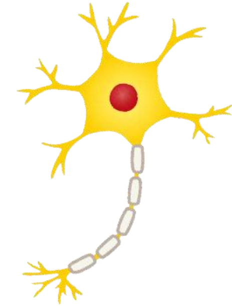
$$(1)\, \Delta w_0 = 0.5(1-1)\mathbf{1} = 0$$
$$\Delta w_1 = 0.5(1-1)1 = 0$$
$$\Delta w_2 = 0.5(1-1)0.5 = 0$$

よって、更新後の重みは以下のようになる:

$$\boldsymbol{w} := \boldsymbol{w} + \boldsymbol{\Delta w} = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix}.$$

すなわち、$\boldsymbol{w}$は学習前と同じ(更新されない)。

ニューロンだって
大変なんやで。



解答例（続き）

与えられた条件下で以下を解けば良い。

$$w_j := w_j + \Delta w_j \quad (\text{今回は} j \in \{0,1,2\})$$
$$\Delta w_j = \eta(y - \hat{y})x_j$$

(2) $\Delta w_0 = 0.5(1-(-1))1 = 0.5 \times 2 \times 1 = 1$

$\Delta w_1 = 0.5(1-(-1))1 = 0.5 \times 2 \times 1 = 1$

$\Delta w_2 = 0.5(1-(-1))0.5 = 0.5 \times 2 \times 0.5 = 0.5$

よって、更新後の重みは以下のようになる：

$$\mathbf{w} := \mathbf{w} + \Delta\mathbf{w} = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 1 \\ 1.5 \\ 1.5 \end{bmatrix}.$$

すなわち、学習により重みが変わった。

解答例 (2) に基づくラベルの予測

いま、$w=\begin{bmatrix}1\\1.5\\1.5\end{bmatrix}$の状態まで学習を進めた。

この状態のモデルを用いて、

真のラベルが$y=1$となるサンプルデータ $x'=\begin{bmatrix}1\\-1\\-1\end{bmatrix}$

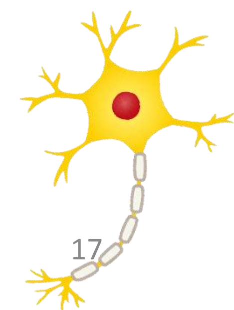
の予測値$\hat{y}$は以下のように計算する:

$$\hat{y}=\phi(\boldsymbol{w}^T\boldsymbol{x}')=\phi\left(\begin{bmatrix}1 & 1.5 & 1.5\end{bmatrix}\begin{bmatrix}1\\-1\\-1\end{bmatrix}\right)$$

(;´Д｀)トホホ…

$$=\phi(1-1.5-1.5)=\phi(-2)=-1$$

つまり、$y\neq\hat{y}$となり予測失敗である。

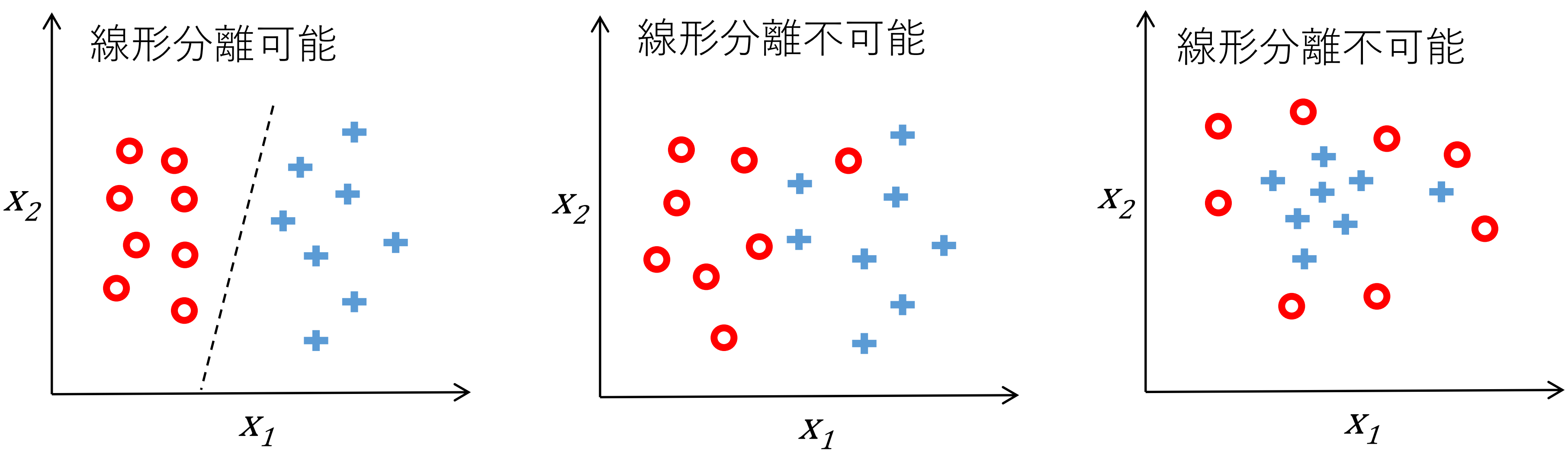


学習手続きの収束について

学習手続きが収束するのは、

データセットが2つのクラスに線形分離可能かつ

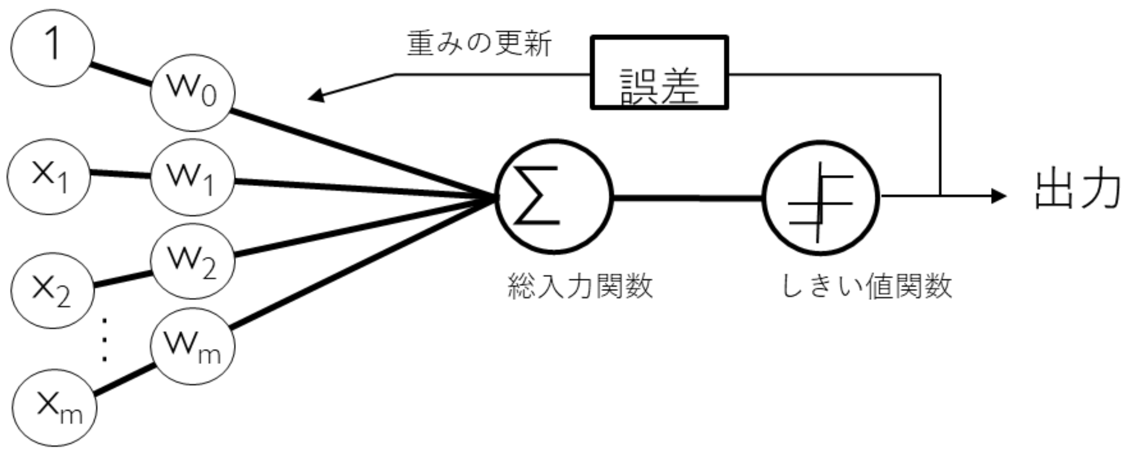
学習率が十分小さい場合のみ。



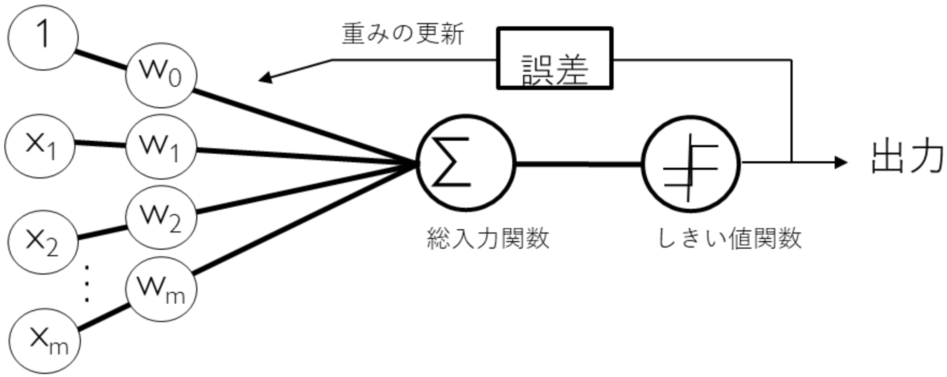
実用上、訓練の最大回数(エポック)や誤分類の最大回数を
学習の終了条件に含めて学習手続きが終了することを保証する。

Quiz: パーセプトロンのモデルの拡張

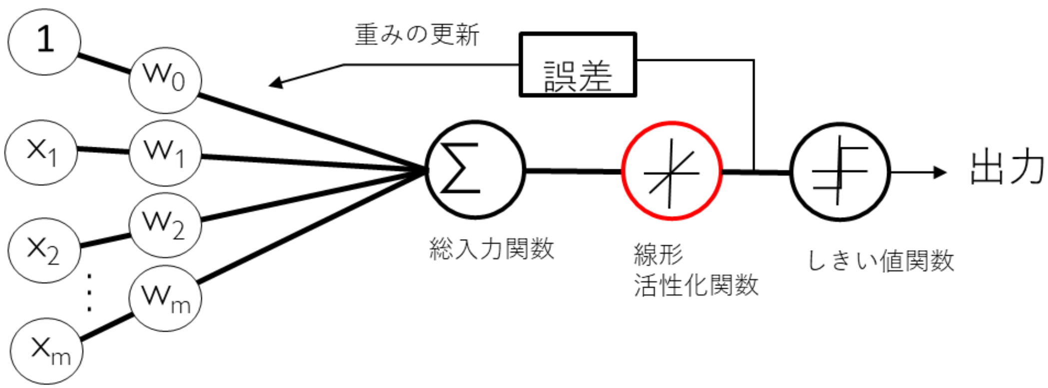
皆さんだったら、以下のパーセプトロンのモデルを どのように改良・拡張しますか？



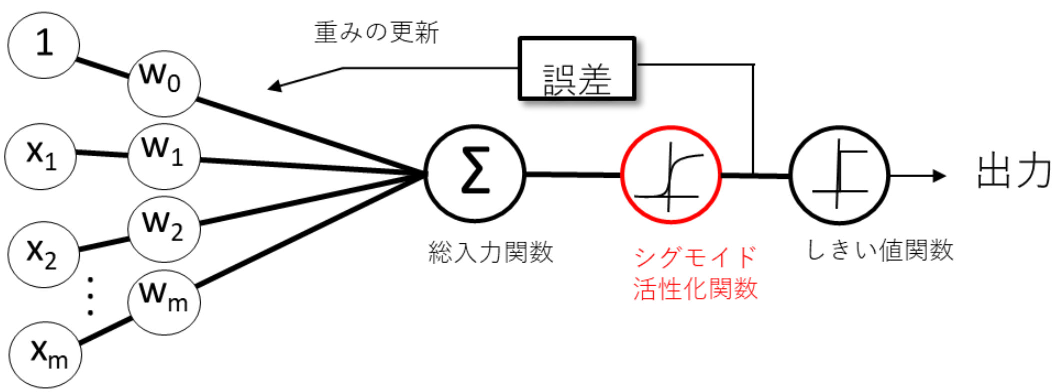
パーセプトロンを拡張したモデル



… パーセプトロン



… ADALINE

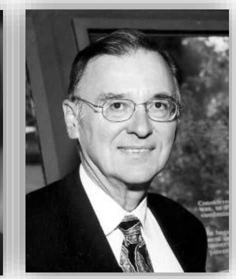


… ロジスティック回帰

Adaptive Linear NEuron

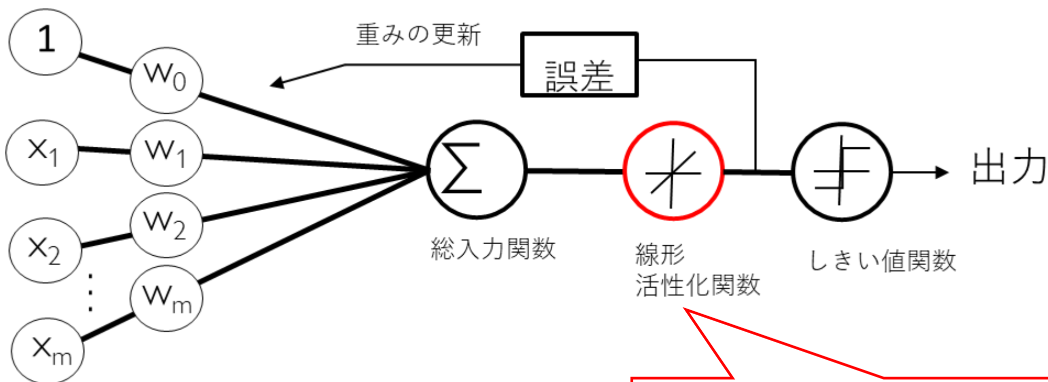
ADALINE

B. Widrow



M. Hoff

学習時に、訓練データx^iの真のクラスラベルy^iと
線形活性化関数からの出力値$\phi(z^i)$を比較する。



線形活性化関数

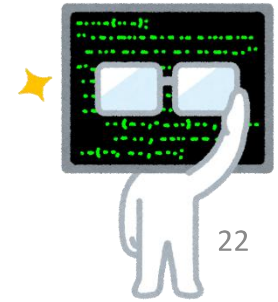
$$\phi(z) = z$$

要するに、恒等関数で
出力は連続値となる。
(単なる-1 or 1ではない)

ADALINEの学習規則

1. 重みベクトルwを初期化する（0 or 乱数）
2. 各訓練データx^iごとに以下の計算をする：
   1. 予測値$\widehat{y^i}$を計算する：
      $$\widehat{y^i} = \phi(z^i),$$
      ただし $z^i = \boldsymbol{w}^T \boldsymbol{x}^i$である。
   2. 重みベクトルwを更新する：
      $$\boldsymbol{w} := \boldsymbol{w} + \Delta \boldsymbol{w}$$
      $$\Delta w_j = \eta \sum_i \left(y^i - \widehat{y^i}\right) x_j^i$$
      ただし、ηは学習率（0～1の実数値）、
      y^iは訓練データx^iの真のラベル、
      $\widehat{y^i}$は訓練データx^iの予測値のラベルである。

訓練データごとに小刻みに更新するのではなく、訓練データ全体で更新する。



勾配降下法

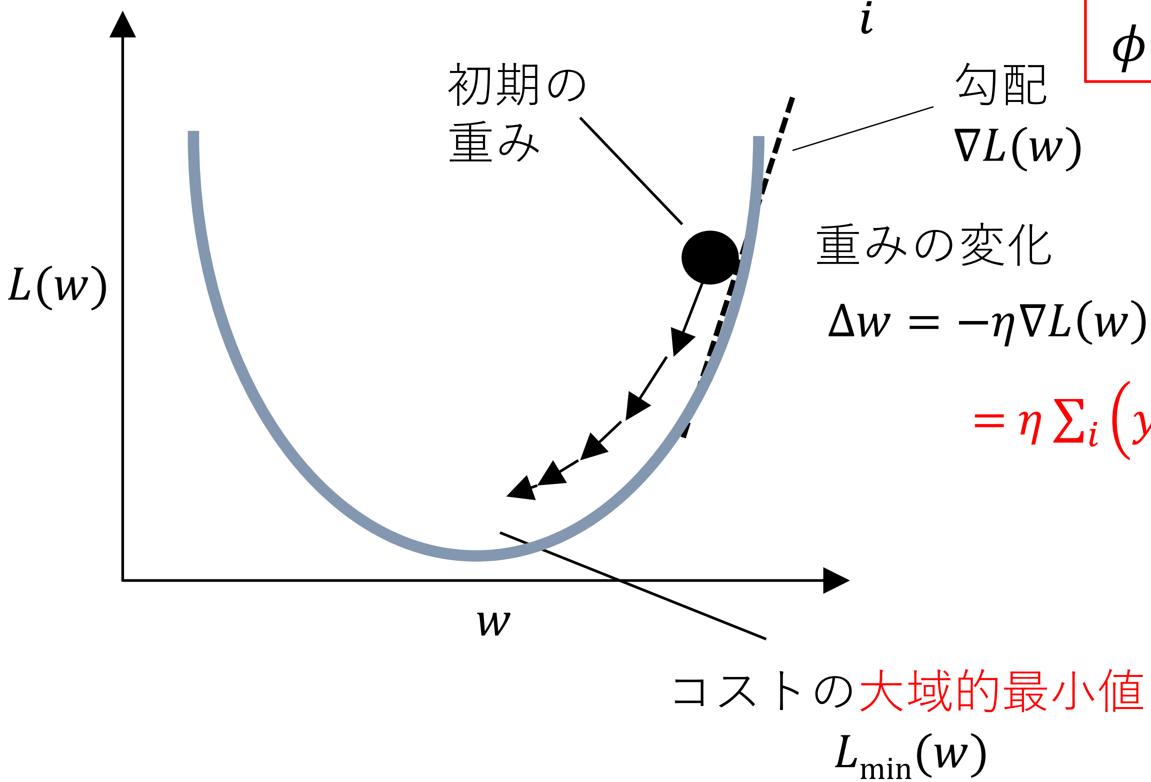
パーセプトロンと違って
損失関数は微分可能＆凸関数なので
損失を最小化する重みを発見できる！

Mean-squared Error

ADALINEは以下の損失関数L（平均二乗誤差）を
目的関数とする勾配降下法によって学習をしている：

$$L(\boldsymbol{w}, b) = \frac{1}{2n}\sum_{i}^{n}(y^i - \widehat{y^i})^2$$

$$\boxed{\phi(z^i) = z^i}$$



重みの変化

$$\Delta w = -\eta \nabla L(w) = -\eta \frac{\delta L}{\delta w_j}$$

$$= \eta \sum_i \left(y^i - \widehat{y^i}\right) x_j^i$$

学習率の設定が良くないと…

大域的最小値を超えてしまい、収束しづらくなる。