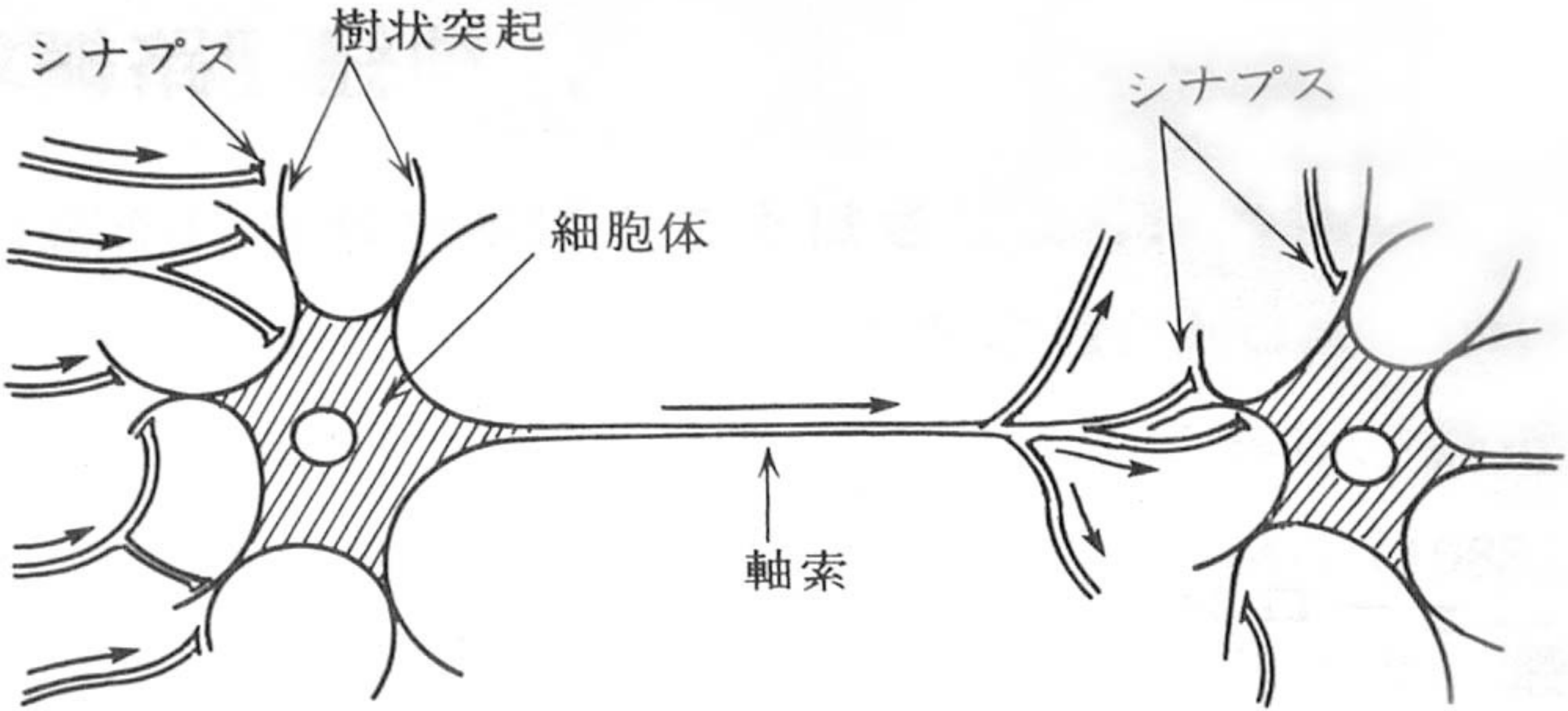
機械学習

神経回路網モデルと深層学習



- 神経回路網モデルとコネクショニズム，甘利俊一，東大出版会

ニューロンの性質

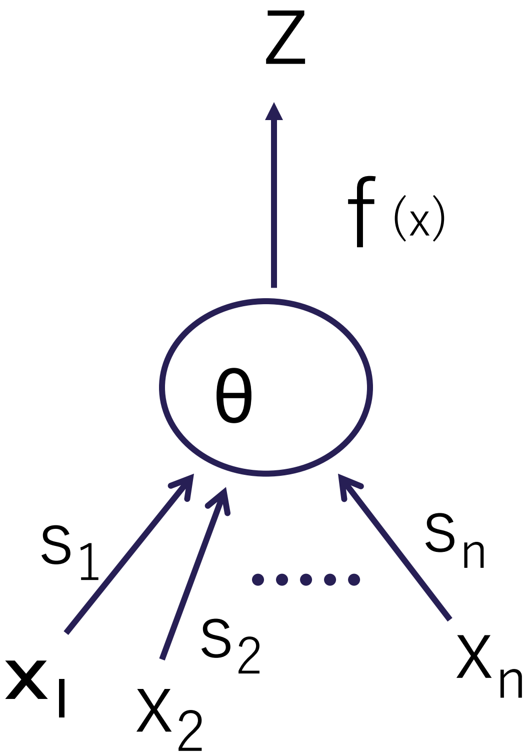
- 線型加算性:

  ニューロンは他のニューロンからの信号を重み付きで総和する。

- 非線型しきい値性:

  総和がしきい値を越えなければ何事も起こらず,越えればパルスを一つ出すという非線型の作用をする。

ニューロンの数理モデル



$x_1, x_2, \ldots, x_n$: 入力

z: 出力

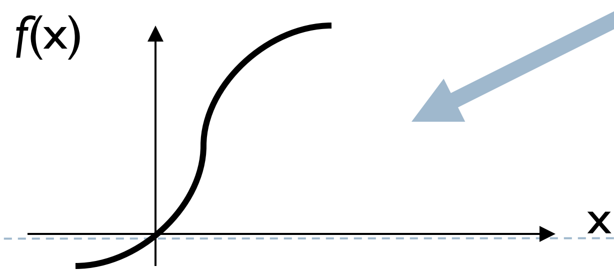
$s_1, s_2, \ldots, s_n$:それぞれの重み（シナプス効率）

θ: スレッシュホールド

$$\sum_i s_i x_i - \theta$$

により、入力の重み付総和が求められる.

さらに出力関数f(x)でフィルタする



$$z = \mathrm{f}\left(\sum_i s_i x_i - \theta\right)$$

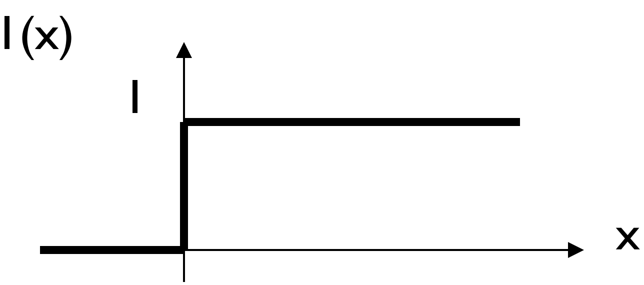
一つのニューロンにおける学習

- 競合学習

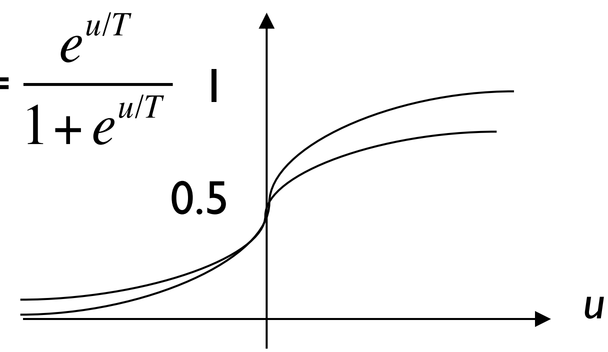
$$\sum_{i=1}^{n} s_i = 一定$$

一つのシナプス効率が上がれば他は下がる。

- 出力関数



$$f(u) = \frac{e^{u/T}}{1+e^{u/T}}$$



学習方程式（一つのニューロン）

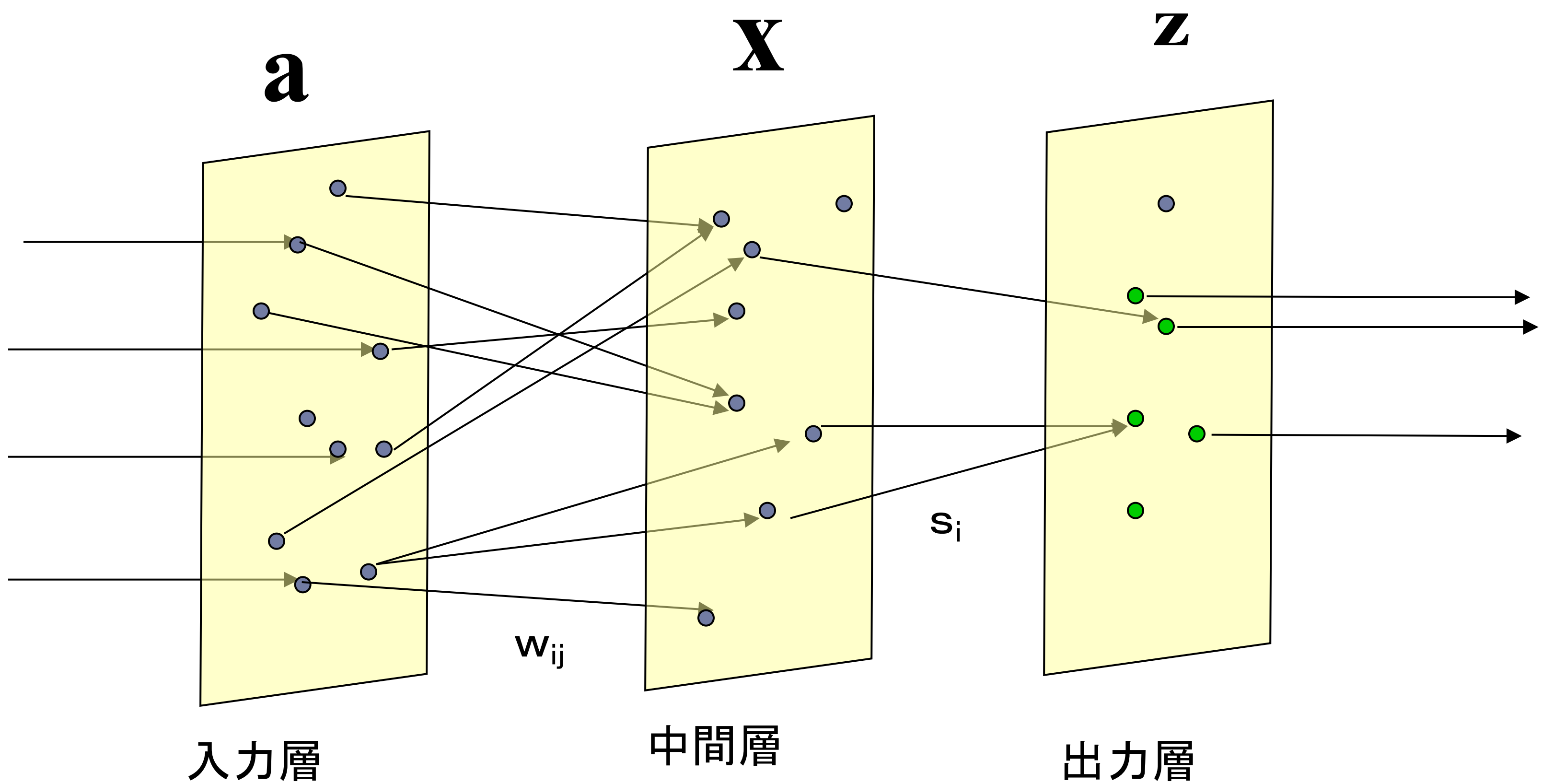
$$\underline{\tau' \frac{ds_i}{dt}} = -s_i + \underline{crx_i}$$

- τ' : 定数
- c : 定数（学習の効率）
- r : 学習信号（いつシナプスの効率を変えるのかの条件
- x_i : 入力

・ほっておけば、ゆっくり減衰する。

・ある条件が満たされるとき、その入力の強さに比例して増える。

学習パーセプトロン



$$x_i = \mathrm{f}\left(\sum_j w_{ij} a_j\right) \qquad z_k = \mathrm{f}\left(\sum_i s_{ki} x_i\right)$$

学習（結合効率の更新）

- 入力信号 $\mathbf{a}$ に対して望ましい出力を $\mathbf{y}_{\mathbf{d}}(\mathbf{a})$ とする。

- 損失関数（θはS_{ki}とw_{ij}のこと）

$$l(\mathbf{a},\theta) = \frac{1}{2}\left|\mathbf{z} - \mathbf{y}_{\mathbf{d}}(\mathbf{a})\right|^2$$

出力
（パーセプトロンの出力層の値）

誤差の絶対値の二乗（1/2は気にしない）

（つづき1）

- 学習信号が逆向きに流れることから、バックプロパゲーションと呼ばれる。
- 最急降下法（Steepest descent method）で解を求める。

$$\frac{\partial l}{\partial s_i} = \left\{z_i - y_{di}(\mathbf{a})\right\} \mathrm{f}'\left(\sum s_j x_j\right) x_i$$

$$= \left\{z_i - y_{di}(\mathbf{a})\right\} \mathrm{f}'\left(\mathbf{s} \bullet \mathbf{x}\right) x_i$$

- s_iを更新する（cは定数）。

$$s_i \longrightarrow s_i - c r_i x_i$$

ただし、 $r_i = (z_i - y_{di})\mathrm{f}'(\mathbf{s} \bullet \mathbf{x})$
（学習信号）

(つづき2)

$$\frac{\partial l}{\partial w_{ij}} = \frac{\partial l}{\partial x_i}\frac{\partial x_i}{\partial w_{ij}}$$

$$= \sum_k (z_i - y_{di}(\mathbf{a}))\frac{\partial z_k}{\partial x_i}\frac{\partial x_i}{\partial w_{ij}}$$

$$= \sum_k (z_i - y_{di}(\mathbf{a}))\mathrm{f}'\left(\sum_j s_{kj}x_j\right)s_{ki}\frac{\partial x_i}{\partial w_{ij}}$$

$$= \left(\sum_k r_k s_k\right)\mathrm{f}'\left(\sum_m w_{im}a_m\right)a_j$$

$$= \tilde{r}_i a_j$$

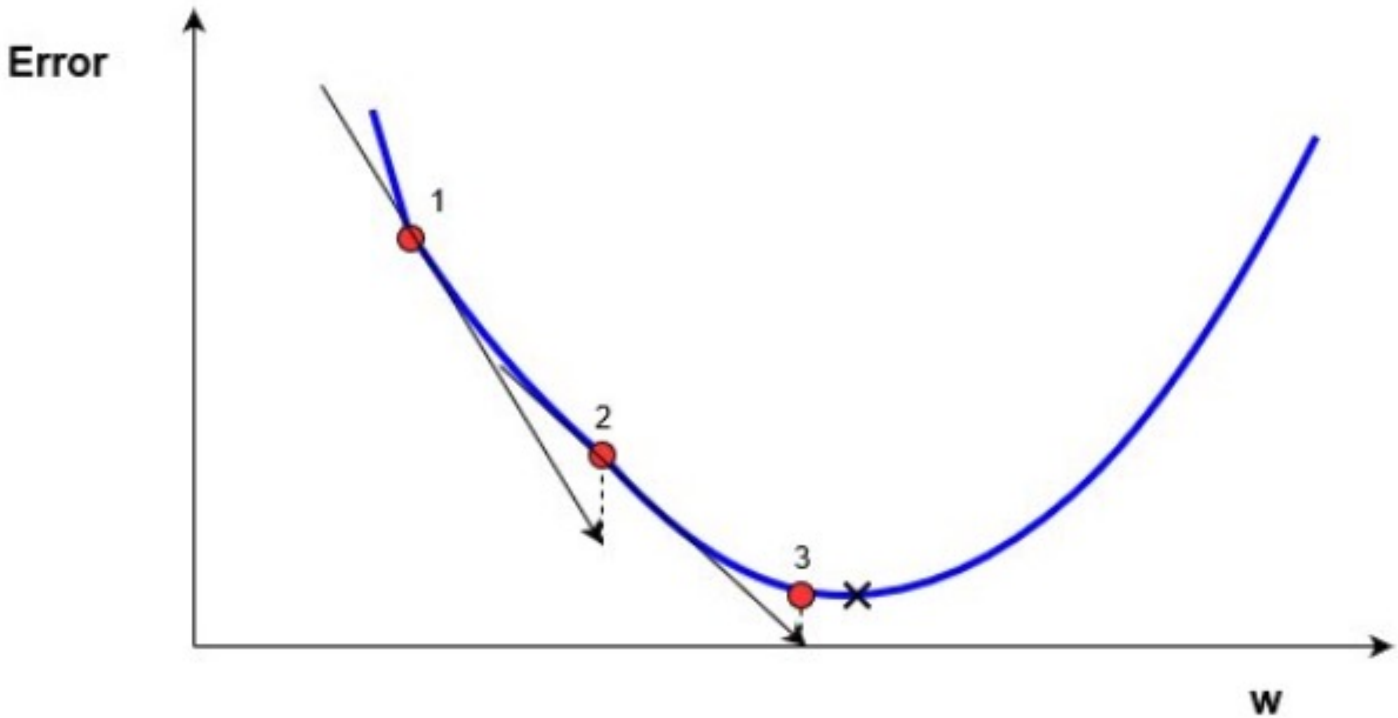
ただし、 $\tilde{r}_i = (\sum_k r_k s_k)\mathrm{f}'(\mathbf{w}_i \bullet \mathbf{a})$

（学習信号）

- w_{ij}を更新する。

$$w_{ij} \longrightarrow w_{ij} - c\tilde{r}_i a_j$$

解の推定（直感的、1次元）

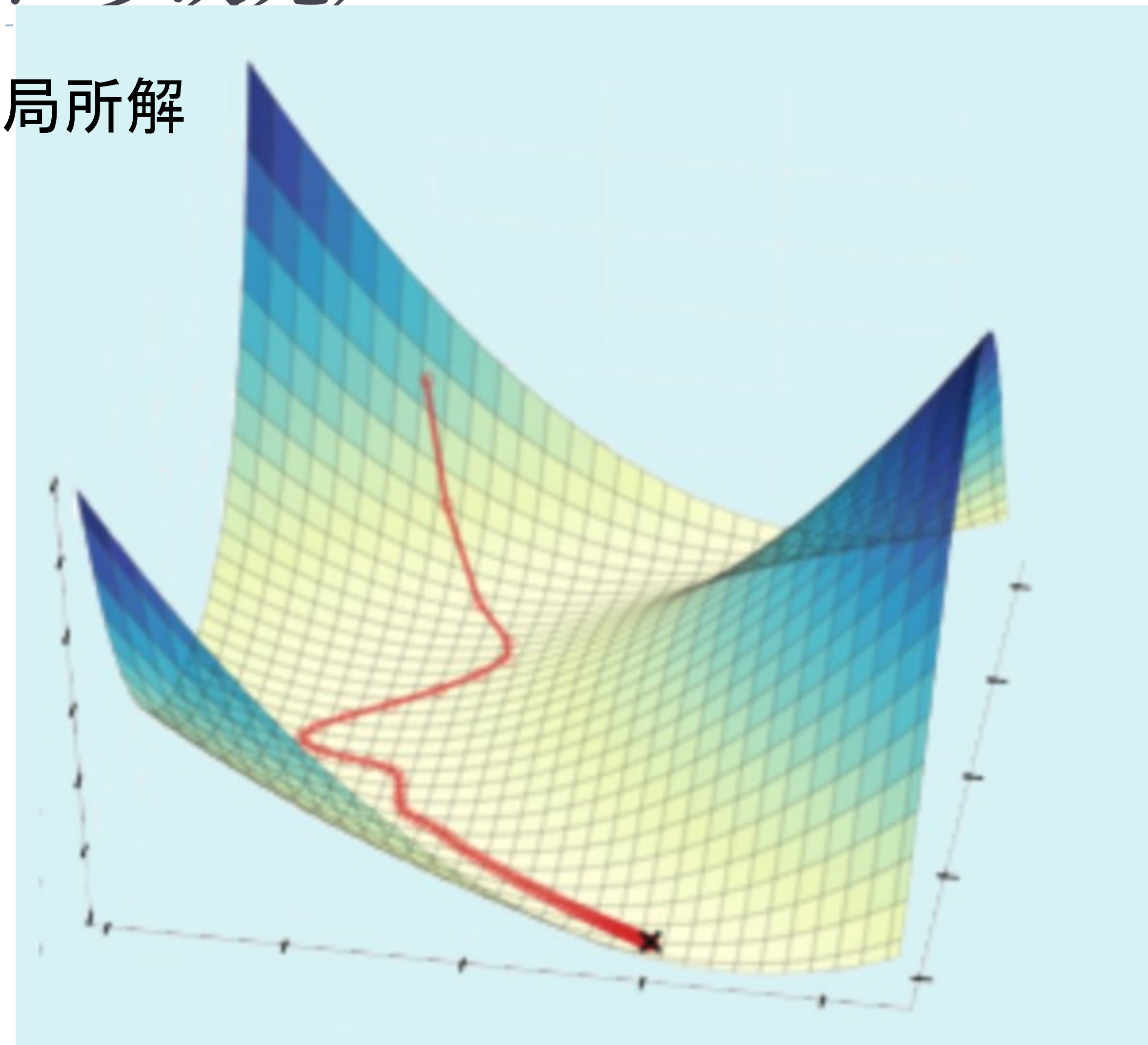


Simple, one-dimensional gradient descent

http://adventuresinmachinelearning.com/stochastic-gradient-descent/

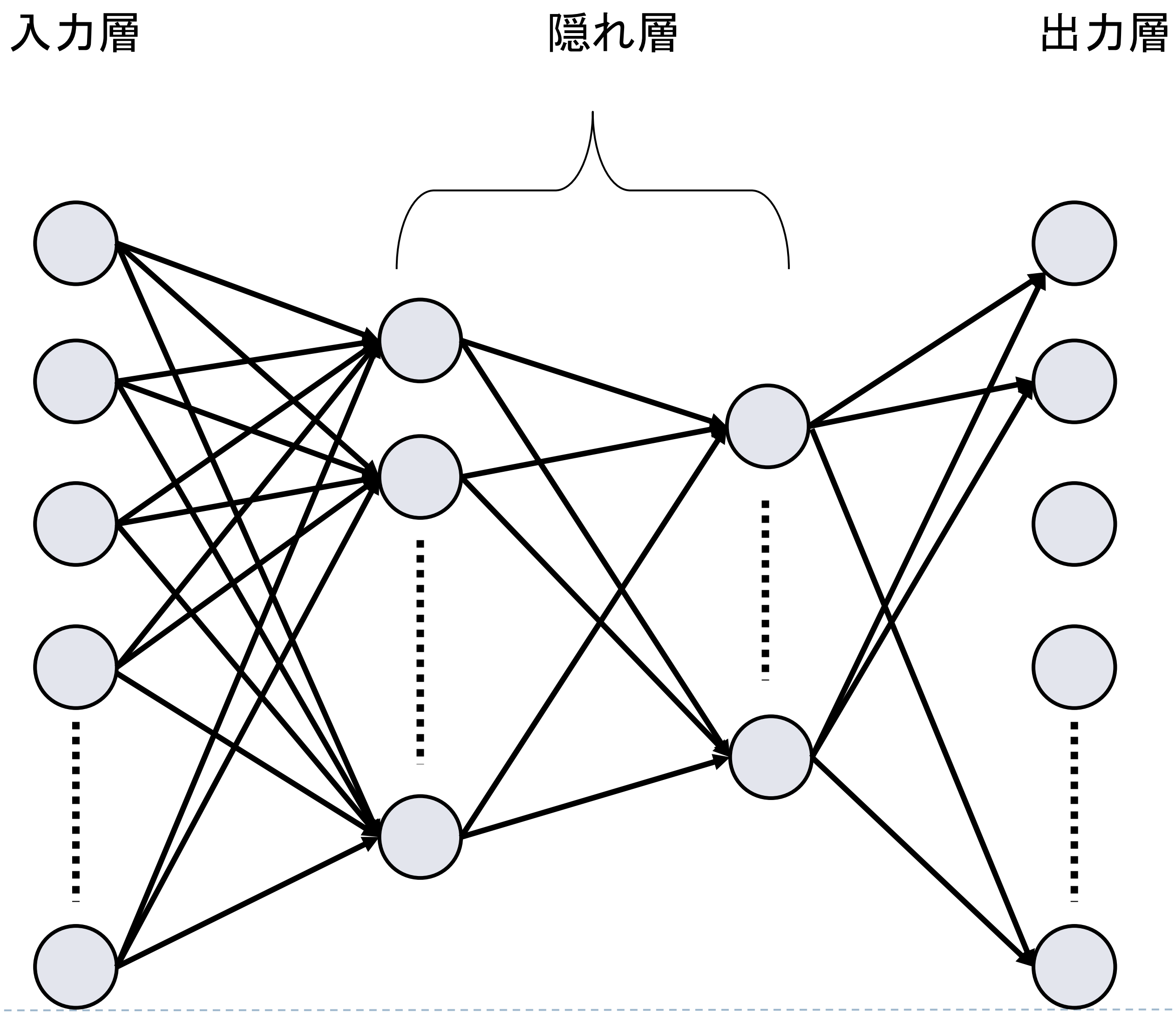
解の推定（直感的、多次元）

- （超）多次元空間での局所解の発見
- 超＝数百　！？



http://dsdeepdive.blogspot.com/2016/03/optimizations-of-gradient-descent.html

多重パーセプトロン



まとめ

- 概要
  - 特徴ベクトル
  - 訓練データ
  - 概要
- 評価
  - 再現率、適合率
  - オープンテスト、クローズテスト
  - 交差検定
- いくつか紹介
  - 決定木学習
  - 回帰分析
  - SVM
- 多クラス分類
- 実例 : SVM^{light}

課題

下の表を見て、天候W、温度T、湿度H、強風Sから「開催」、「中止」を決定する式を求めよ。
条件判定、論理演算（＜、＞、＝等）が少ないものがより良い解とする。

天候	温度（°F）	湿度（%）	強風	クラス
晴れ	75	70	真	開催
晴れ	80	90	真	中止
晴れ	85	85	偽	中止
晴れ	72	95	偽	中止
晴れ	69	70	偽	開催
曇り	72	90	真	開催
曇り	83	78	偽	開催
曇り	64	65	真	開催
曇り	81	75	偽	開催
雨	71	80	真	中止
雨	65	70	真	中止
雨	75	80	偽	開催
雨	68	80	偽	開催
雨	70	96	偽	開催

課題

- gain(「湿度が75%以下」) を求めよ。

多クラス分類（その２）

- pairwise法
  - クラスC_0, C_1, …, C_{n-1}から任意の2つのクラスを選ぶ全ての組み合わせ

    $$(C_0, C_1), \ldots, (C_i, C_j), \ldots$$

    に対して、

    それぞれの判定器

    $$\ldots, SVM_{ij}, \ldots$$

    を作る。
  - 多数決で一番多く判定されたクラスを採用する。