

# SAPT

# 第6回知能化分科会

一色 浩  
(SAPT知能化分科会長)  
2018.07.07

SAPT電子会議室C

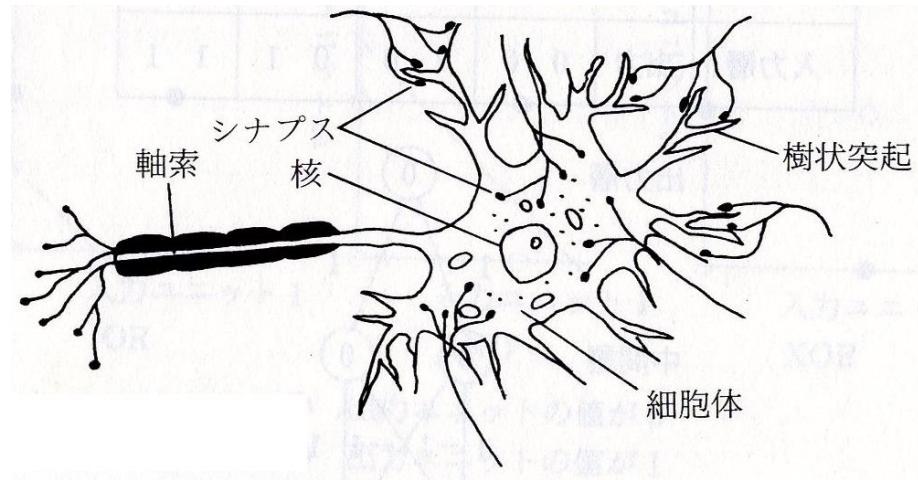
# 目 次

0. これまでのまとめ...深層学習とは
1. 瀧 雅人著「これならわかる深層学習」
2. 動物のニューロンの数の一覧
3. 植物の神経
4. 飲み会の代わり...自由討論

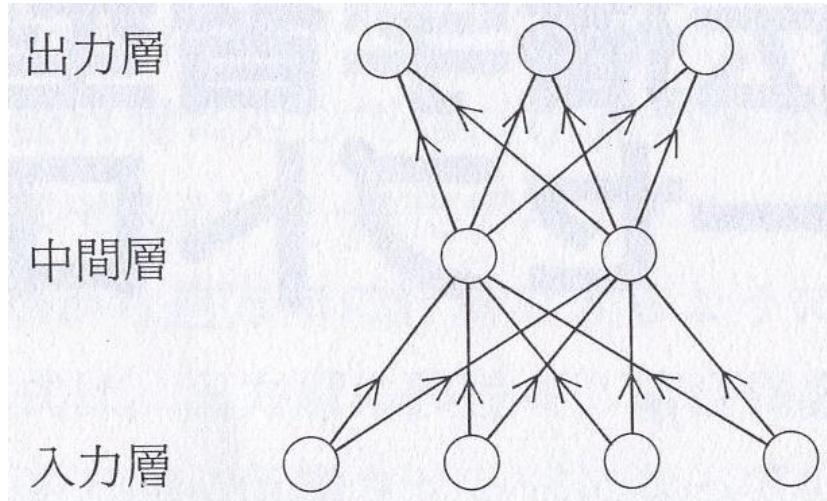
# これまでのまとめ...深層学習とは

## ニューロン

人間の脳には140億個のニューロンがあり、ネットワークを作っている



## 誤差逆伝播学習・階層ニューラル・ネットワーク(NN)



誤差(教師データと出力データの差)をゼロにする学習をする

最近NNが大きな話題

ブレークスルーのキーワードは**深層学習**

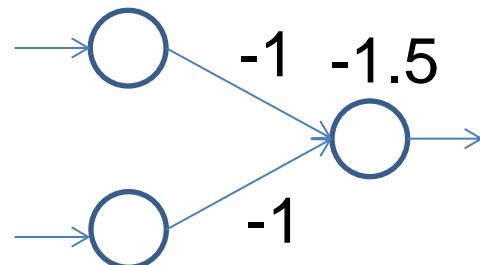
人工知能の中核技術になる勢い

これまでのNN

汎化能力(未学習のデータに正解する能力)が非常に低い

NNはいかなる問題も解けることが数学的に保証

NNはNAND回路を作れる。NAND回路を用いればいかなる論理回路も構成できる。



NAND回路

汎化能力を高めるために多層化すると、ローカル・ミニマムに捕まって学習を継続できない



NNの冬の時代

2006年にHintonらが深層ボルツマン・マシンに成功

2012年にHintonらが深層NNが画像認識で画期的な誤り率をたたき出す



深層NNの時代到来

# 深層学習とは何か？

## 従来のNNの問題点と対策

汎化能力が低い、しかし多層化できない

十分に学習できない

シグモイド以外の活性化関数の導入

例えばReLU(Rectified Linear Unit)

前処理ネットワークの導入

自己符号化器(Autoencoder)

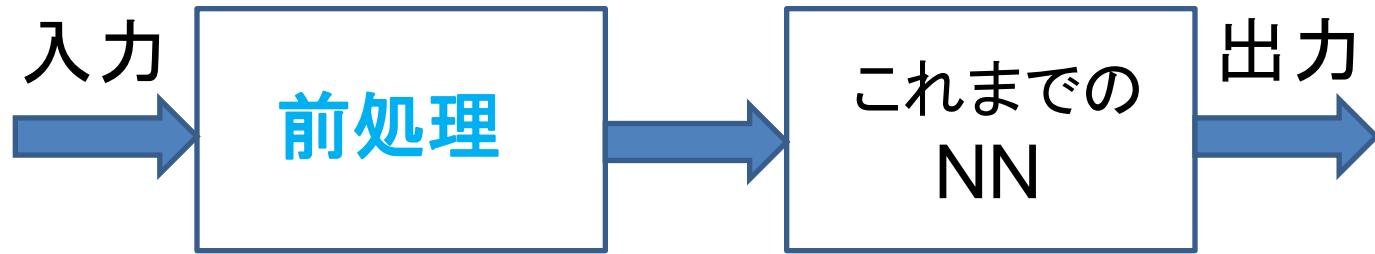
畳み込みNN(Conversation NN)

学習し過ぎて過学習になる

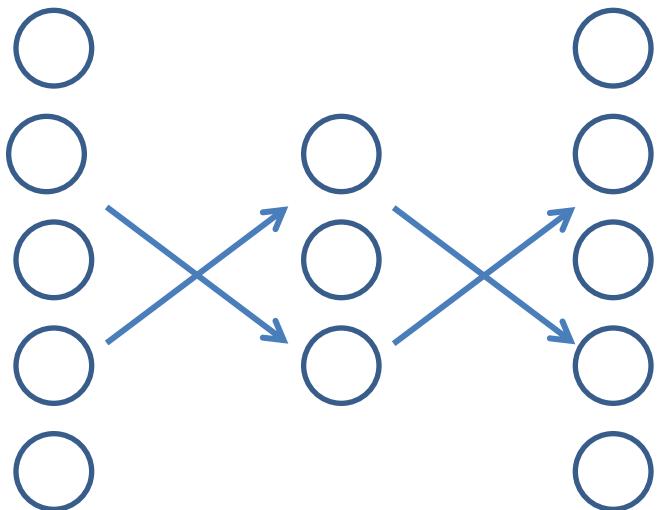
学習の制限

正則化

## 新しいアーキテクチャの導入：



## 自己符号化器 (Autoencoder)

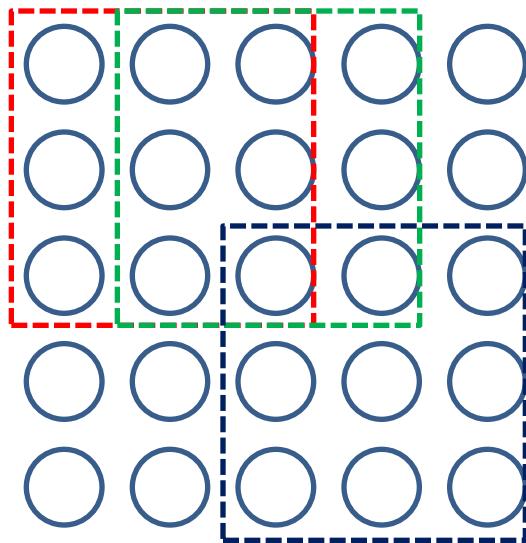


ニューロン数：  
入力層 = 出力層 > 中間層

学習：  
教師データ = 入力データ  
中間層に圧縮されたデータ

# 畳み込みNN(Convolution NN)

データを  
2次元的  
に考える



瀧 雅人, 「これなら分かる  
深層学習入門」, 講談社,  
2017, p. 159

$$26 = 5 \times 2 + 1 \times 0 + 0 \times (-2) \\ + 1 \times 0 + 6 \times 2 + 1 \times 0 \\ + 4 \times (-2) + 1 \times 0 + 6 \times 2$$

5 1 0 1 9

1 6 1 7 0

4 1 6 3 5      \*

1 8 0 1 0

2 2 8 0 4

2 0 -2

0 2 0

-2 0 2

26 6 -6

0 -4 8

24 -14 -4

## 正則化

過学習の防止：過剰な自由度を除く

重み減衰(Weight Decay)項を導入：

$$E_{WD}(\mathbf{w}) = E(\mathbf{w}) + \lambda \mathbf{w}^T \mathbf{w} = \text{minimum}$$

誤差関数  $E(\mathbf{w})$  だけでなく,  $\mathbf{w}^T \mathbf{w}$  も最小化されるので, 不要な重みは出来るだけゼロに近づく.

実際的な方法：訓練データを学習データと評価データに分け, 評価データが増加に転じた時点で学習をやめる.

性質のよく分かったものの学習には、従来からある機械学習が非常にうまく機能する。例えば、関数関係の把握には数学的な補間がうまくいく場合が多い。

複雑な現象や規則性を見いだせにくい現象（株価の変動）の学習には、従来からある機械学習がうまく機能しない。

複雑な画像認識などはニューロの独壇場である。

一般的に言って、どんな問題がニューロに適している、またはニューロでないと解けないのか？

こんなことを理解することが重要と考えられないか？

# こんな問題は一口で解ける？...地震の予測

地震年表には過去の大きな地震の膨大な記録がある。

## 東北大震災の前震、余震

年	月日	名称	マグニチュード
2011	0.09	三陸沖	7.3
2011	311	東北地方太平洋沖地震	9.1
2011	311	岩手県沖	7.4
2011	311	茨城県沖	7.6
2011	311	三陸沖	7.5
2011	312	長野県北部	6.7
2011	315	静岡県東部	6.4
2011	407	宮城県沖	7.2
2011	411	福島県浜通り	7
2011	412	福島県中通り	6.4
2011	630	長野県中部	5.4
2011	710	三陸沖	7.3
2012	314	三陸沖	6.9
2012	314	千葉県東方沖	6.1
2012	1207	三陸沖	7.3
2013	1026	福島県沖	7.1
2014	712	福島県沖	7
2016	1122	福島県沖	7.4
2016	1228	茨城県北部	6.3

## 地震の年表 (日本)

[https://ja.wikipedia.org/wiki/%E5%9C%80%E9%9C%87%E3%81%AE%E5%B9%B4%E8%A1%A8\\_\(%E6%97%A5%E6%9C%AC\)](https://ja.wikipedia.org/wiki/%E5%9C%80%E9%9C%87%E3%81%AE%E5%B9%B4%E8%A1%A8_(%E6%97%A5%E6%9C%AC))

次の余震がいつ、  
どこで、どの規模  
で発生するか？

# 瀧 雅人著「これならわかる深層学習」

## 3 ニューラルネット

### 3.1 神経細胞のネットワーク

人間の脳には、1000億個以上の下図のような神経細胞（ニューロン）がネットワークを形成している。

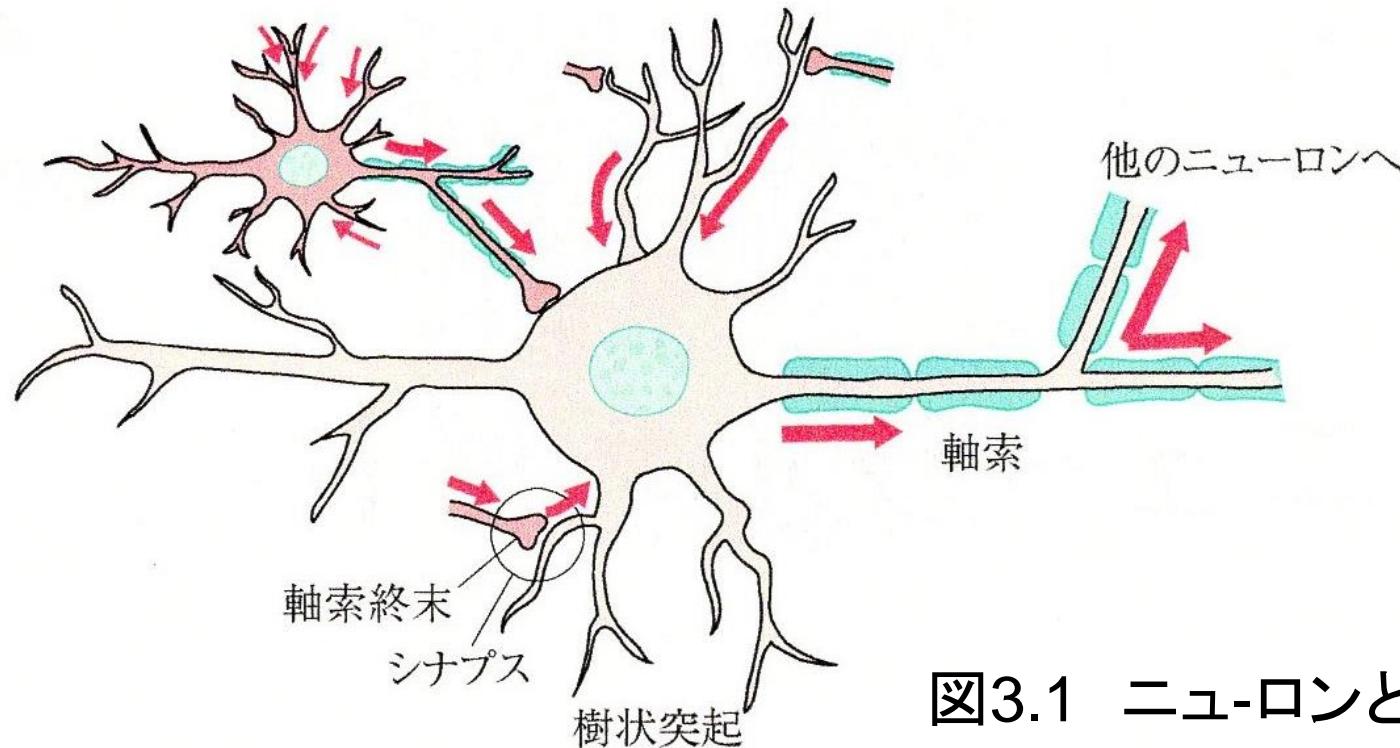


図3.1 ニューロンとシナプス

神経細胞の数は大脳で数百億個、小脳で1000億個、脳全体では千数百億個にもなる。

一つの神経細胞からは、信号を出力する長い「**軸索**」と、信号を入力する木の枝のように複雑に分岐した短い「**樹状突起**」が伸びている。

軸索終末と樹状突起との間には**シナプス**があり、信号が伝達される。

シナプスでは軸索を伝わってきた電気信号を受けると、**シナプス小胞**というカプセルに詰め込まれた**神経伝達物質**と呼ばれる化学物質を放出する。

樹状突起の**レセプタ**が受けると**電気信号**が細胞体（中央の膨らみ）に向けて出される。

細胞体では樹状突起から上がってきた全信号を合算し、それがしきい値を超えると軸索に向けて電気信号を発する。

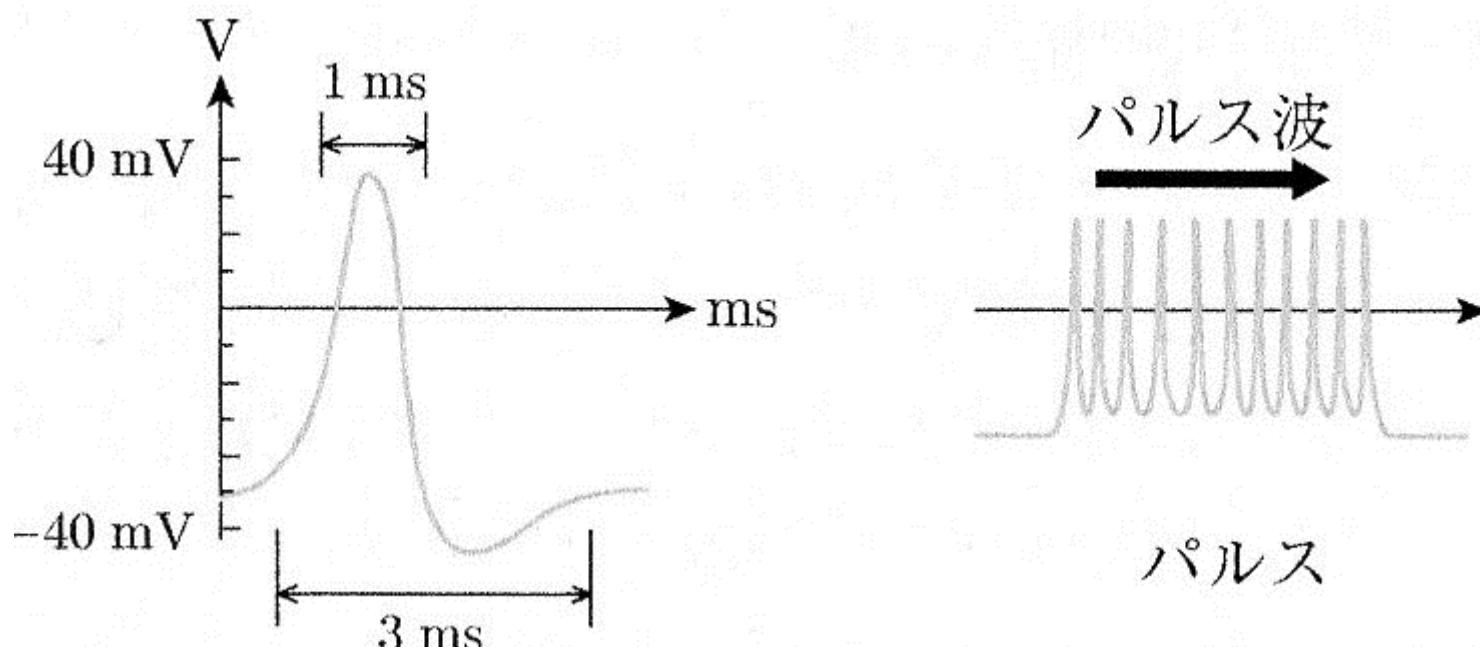


図3.2 活動電位と電気パルスの伝播

電気信号は連續パルスからなる。パルスの振幅は決まっており、信号の強さはパルスの密度である。

ニューロンの形状を明らかにしたのはイタリア人のカミッロ・ゴルジで、1873年に銀とクロムを用いた細胞の染色法を編み出して観察に成功する。

スペインのサンティアゴ・ラモン・イ・カハールは、シナプスでは軸索終末と樹状突起は直接つながっておらず、狭い隙間があることを主張した。ゴルジは継っていると考えた。光学顕微鏡では直接確認できないため見解が別れたが、結果的にはカハールが正しかった。

1906年にゴルジとカハールはノーベル賞を共同受賞したが、その時点でも二人は対立したままであった。

ニューロン同士が直接継っていない事には、信号処理上、深い意味があろう。

## 3.2 形式ニューロン

NNの起源は1943年のマカロックとピツの研究.

神経生理学者で外科医のマカロックは、チューリングの影響を受けて、思考の計算モデルを作ろうと考えた。

神童のピツを共同研究者として、脳の論理処理の計算モデル(形式ニューロンと呼ばれる素子からなる回路)を考え出した。

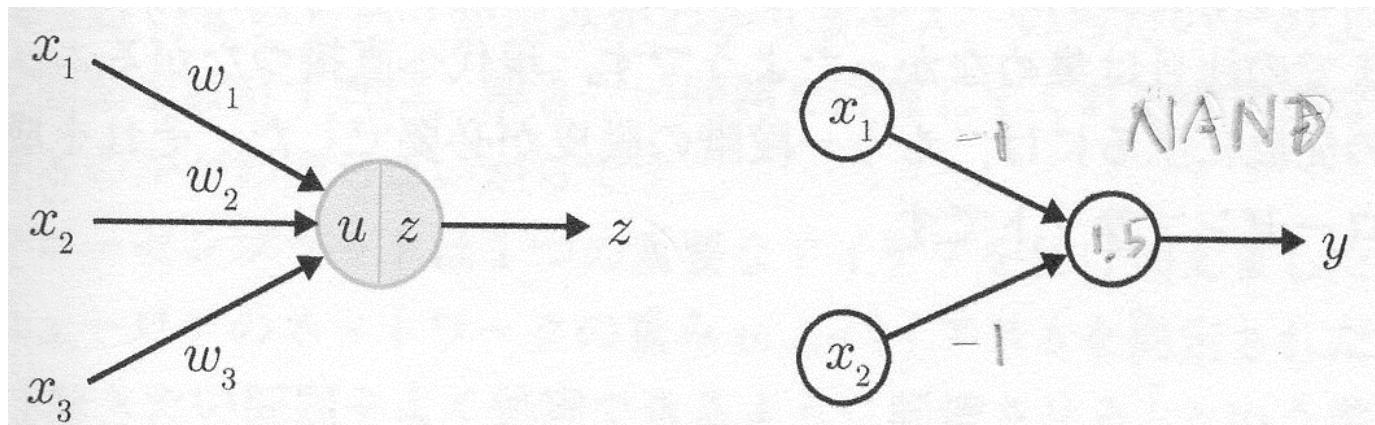


図3.3 形式ニューロン(左)と簡単なニューロン回路の例(右)

形式ニューロンには他の多数の形式ニューロンから0か1の信号  $x_i$  が入る。重みを  $w_i$  として総入力  $u$  を

$$u = \sum_i w_i x_i \quad (3.1)$$

とする。このニューロンの出力  $z$  を

$$z = \theta(u + b) = \theta\left(\sum_i w_i x_i + b\right) \quad (3.3)$$

とする。ここで  $\theta$  はヘビサイドの階段関数：

$$\theta(u + b) = \begin{cases} 1 & (u \geq -b) \\ 0 & (u < -b) \end{cases} \quad (3.2)$$

入力  $u$  を出力  $z$  に変換する関数を **活性化関数** という。

図3.3の右はNAND。NNでいかなる論理演算も可能。

### 3.3 パーセプトロン

前節ではパラメータ  $w$  や  $b$  は固定値であった.

ローゼンブラットはパラメータ  $w$  や  $b$  を「問題の解を与えるように訓練する」という学習の概念を導入した.

つまり、図3.4のようなNNに教師あり学習をさせる.

ローゼンブラットはこのようなNNをパーセプトロンと呼んだ.

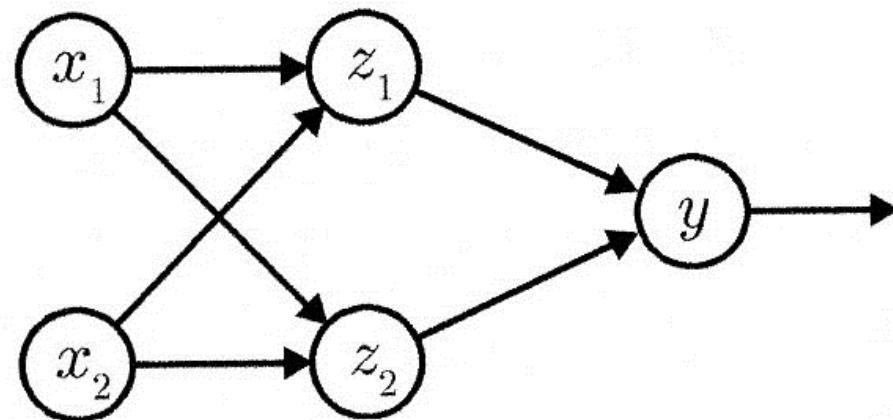


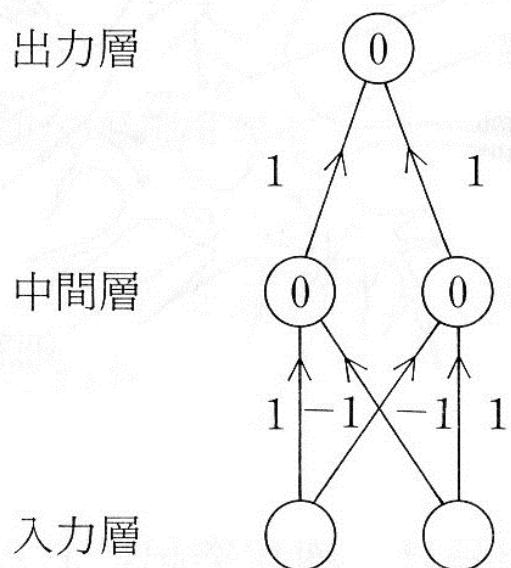
図3.4 3層からなるパーセプトロンの例

人工知能と呼ばれる研究分野を立ち上げたミンスキー（ローゼンブラットの高校の同級生）は、入力と出力からなる2層パーセプトロンではXORのような線形分離不可能問題は解けないと示した。

これが一因となり60年代の終わりに第1次ニューラルネットブームが終焉した。

3層パーセプトロンならばXORを解ける。

出力層	出力	0	1	1	0
	入力	0	1	1	0
中間層	出力	0 0	1 0	0 1	0 0
	入力	0 0	1 -1	-1 1	0 0
入力層	出力	0 0	1 0	0 1	1 1



3層パーセプトロンによるXORの解

## 3.4 順伝播型ニューラルネットワークの構造

### 3.4.1 ユニットと順伝播型ニューラルネットワーク

図3.5に階層ニューラルネットワークを示す。パーセプトロンの入出力は0, 1であったがNNでは**実数値**。

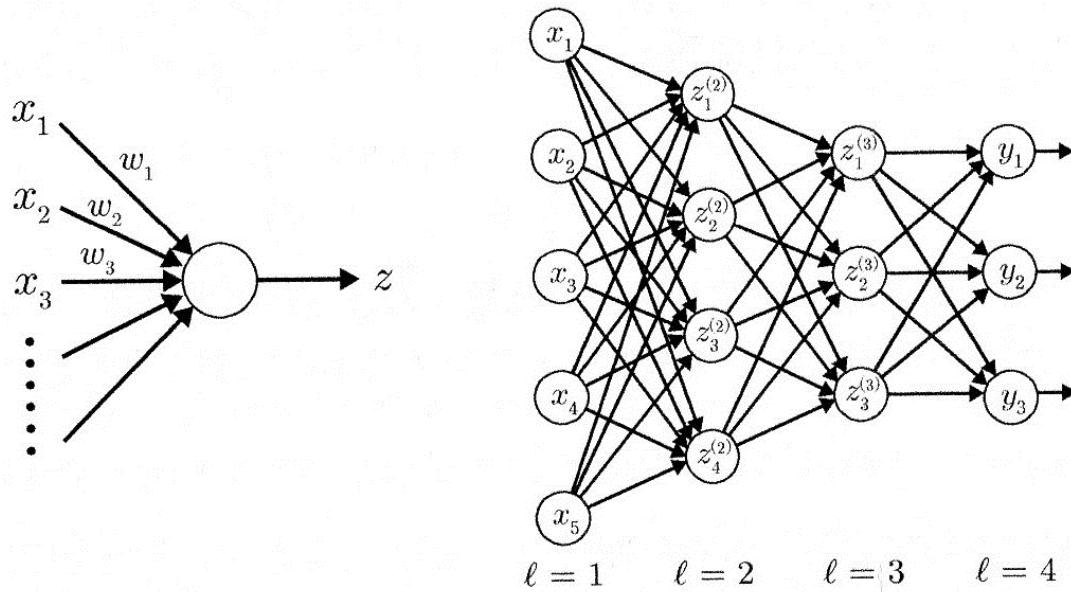


図3.5 左はユニットの構造。右は4層順伝播型NNの例。丸の変数は各ユニットの出力値を意味する。

出力と入力、入力と出力の関係はパーセプトロンと同じ。改めて書くと

$$u = \sum_i w_i x_i \quad (3.6)$$

$$z = f(u + b) = f\left(\sum_i w_i x_i + b\right) \quad (3.7)$$

パーセプトロンとの違いは入出力が実数になっていること、活性化関数には連続関数が使われる。

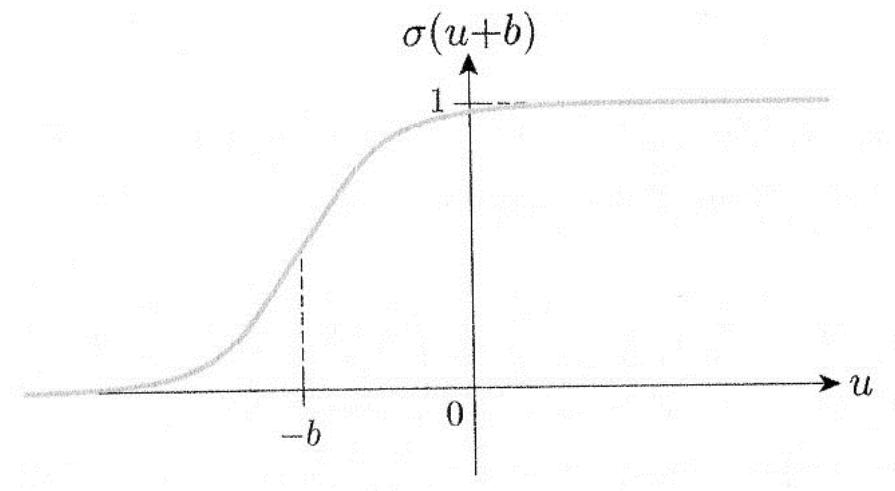


図3.6 シグモイド関数

## 3.4.2 入力層

第1層の出力は入力そのもの.

$$z_i^{(1)} = x_i, \quad \mathbf{z}^{(1)} = \mathbf{x} \quad (3.8)$$

## 3.4.3 中間層

第 $l$ 層の $j$ 番目のユニットを考える.

第 $l-1$ 層の $i$ 番目のユニットの出力  $z_i^{(l-1)}$  を入力とするので,

$$u_j^{(l)} = \sum_i w_{ji}^{(l)} z_i^{l-1} \quad (3.9)$$

したがって、出力は

$$z_j^{(l)} = f^{(l)} \left( u_j^{(l)} + b_j^{(l)} \right) = f^{(l)} \left( \sum_i w_{ji}^{(l)} z_i^{l-1} + b_j^{(l)} \right) \quad (3.11)$$

重み行列:  $\mathbf{W}^{(l)} = \begin{pmatrix} w_{11}^{(l)} & w_{12}^{(l)} & \dots \\ w_{21}^{(l)} & w_{22}^{(l)} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$  (3.12)

を導入すると

$$\mathbf{u}^{(l)} = \mathbf{W}^{(l)} \mathbf{z}^{(l-1)}, \quad \mathbf{z}^{(l)} = f^{(l)} \left( \mathbf{u}^{(l)} + \mathbf{b}^{(l)} \right) \quad (3.13)$$

バイアスを重みに含めて考えることができる.  
 すべての中間層に常に1を出力するユニットを加える.  
 これを  $i = 0$  というラベルで呼ぶことにすると

$$z_0^{(l)} = 1 \quad (3.15)$$

## 変更点

(3.12)式は

$$\mathbf{W}^{(l)} = \begin{pmatrix} w_{10}^{(l)} & w_{11}^{(l)} & w_{12}^{(l)} & \cdots \\ w_{20}^{(l)} & w_{21}^{(l)} & w_{22}^{(l)} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

(3.13)式は

$$\mathbf{u}^{(l)} = \mathbf{W}^{(l)} \mathbf{z}^{(l-1)}, \quad \mathbf{z}^{(l)} = f^{(l)}(\mathbf{u}^{(l)}) \quad (3.17)$$

### 3.4.4 出力層

$L$  層からなるNNの出力は最後の中間層である第  $L-1$  層の出力  $\mathbf{h} = \mathbf{z}^{(L-1)}$  を受けて,  $\hat{\mathbf{y}} = \mathbf{z}^{(L)}$  を出力する.

$$\hat{\mathbf{y}} = \mathbf{z}^{(L)} = f^{(L)}(\mathbf{u}^{(L)}), \quad \mathbf{u}^{(L)} = \mathbf{W}^{(L)} \mathbf{h} \quad (3.18)$$

最終中間層出力  $\mathbf{h}$  のことを入力  $\mathbf{x}$  の **深層表現** という.

### 3.4.5 関数

順伝播型NNは入力  $\mathbf{x}$  に各層での処理を順時加えるものであるから入力  $\mathbf{x}$  の次のような関数を与えている。

$$\hat{\mathbf{y}} = f^{(L)} \left( \mathbf{W}^{(L)} f^{(L-1)} \left( \mathbf{W}^{(L-1)} f^{(L-2)} \left( \dots \mathbf{W}^{(2)} f^{(1)}(\mathbf{x}) \right) \right) \right) \quad (3.19)$$

### 3.5 ニューラルネットによる機械学習

学習という概念は以下のようにまとめられよう。

出力  $y$  はパラメータ  $W, b$  の関数である：

$$y = y(x; W^{(2)}, \dots, W^{(L)}, b^{(2)}, \dots, b^{(L)}) \quad (3.20)$$

パラメータをまとめて  $w$  と書くと,  $w$  を適当な値にとることにより出力を与えられたデータに合わせることができる。

その手順は機械学習における通常の教師あり学習と同じである。

訓練データ集合  $D = \{(x_n, y_n)\}_{n=1, \dots, N}$  を用意し, 訓練データ  $x_n$  を与えた時の出力  $y(x_n, w)$  と目標値  $y_n$  とのズレを出来るだけ小さくする。

ズレを測る誤差関数  $E(\mathbf{w})$  の選び方は、考えるタスクと出力層の構造に依存する。誤差関数の最小化は

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} E(\mathbf{w}) \quad (3.21)$$

最終中間層出力  $\mathbf{h}$  のことを入力  $\mathbf{x}$  の **深層表現**という。

$$\mathbf{h} = \mathbf{z}^{(L-1)}(\mathbf{x}; \mathbf{w}^{(2)}, \dots, \mathbf{w}^{(L)}) \quad (3.22)$$

### 3.5.1 回帰

表現  $\mathbf{h}$  について **線形回帰** をさせたいならば、出力の活性化関数を恒等写像にする：

$$\mathbf{y} = \mathbf{W}^{(L)}\mathbf{h} + \mathbf{b}^{(L)} \quad (3.23)$$

出力をデータでフィッティングするならば、通常平均二乗誤差を最小化する。

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N (\mathbf{y}(\mathbf{x}_n; \mathbf{w}) - \mathbf{y}_n)^2, \quad \mathbf{w}^* = \arg \min_{\mathbf{w}} E(\mathbf{w}) \quad (3.25)$$

これが通常の回帰と違うのは、**回帰関数のパラメータだけではなく、表現を決める中間層のパラメータも同時に学習していること**である。

### 3.5.2 2値分類

回帰と並んで重要なタスクは分類である。2つのクラスに対応するラベル  $y$  の値は0, 1の二値とする。

$y$  が  $\mathbf{h} = \mathbf{z}^{(L-1)}$  のロジスティック回帰を与えるように設計。

訓練データ  $D = \{(\mathbf{x}_n, y_n)\}_{n=1,\dots,N}$  の標的値  $y_n$  は0か1であるが、ロジスティック回帰ではこの2値変数そのものではなくて、 $y$  が1である確率  $\hat{y} = P(y = 1 | x)$  を推定する。

なぜならば、2値変数  $y$  の従うベルヌーイ分布のもとで、

$$E_{P(y|\mathbf{x})}[y | \mathbf{x}] = \sum_{y=0,1} y P(y | \mathbf{x}) = P(y = 1 | \mathbf{x})$$

したがってロジスティック回帰を行うには、出力層のユニットが一つで、その出力値が  $P(y = 1 | \mathbf{x})$  を推定するようなものを考えるのが自然である。

式(2.58)に合わせて、以下のようにする：

$$y(\mathbf{x}; \mathbf{w}) = P(y = 1 | \mathbf{x}; \mathbf{w}) = \sigma\left(\sum_i w_i^{(L)} h_i\right) \quad (3.26)$$

### 3.5.3 多クラス分類

省略

## 3.6 活性化関数

マカラックとピツが導入したヘビサイドのステップ  
関数は、今ではほとんど用いられていない。

残念なことに隠れ層の活性化関数を一意に決める  
一般的な判断基準はいまだに存在しない。

### 3.6.1 シグモイド関数とその仲間

1980年代以降は微分可能な関数が使われている。

よく知られている例は、**ステップ関数を滑らかにしたシグモイド関数**である：

$$\sigma(u) = \frac{1}{1 + e^{-u}} = \frac{1}{1 + e^{-u}} \quad (2.57)$$

この関数には、次の好都合な関係がある：

$$\sigma'(u) = \sigma(u)(1 - \sigma(u))$$

シグモイド関数の値域は  $0 \leq \sigma(u) \leq 1$ ，次の双曲線正接関数は  $-1 \leq f(u) \leq 1$  である：

$$f(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}} \quad (3.36)$$

実用の場面では、中間層に関しては、シグモイドよりも、こちらが好まれた。

双曲線正接関数を区分線形近似したハード双曲線正接関数が用いられることがある：

$$f(u) = \begin{cases} 1 & (u \geq 1) \\ u & (-1 < u < 1) \\ -1 & (u \leq -1) \end{cases} \quad (3.37)$$

### 3.6.2 正規化線形関数

現在では、**ステップ関数ばかりかシグモイド関数や双曲線正接関数などもそれほど用いられない**。理由は後出。学習をスムースに進行させるには**正規化線形関数**(rectified linear function)を用いると良いことが分かっている。

## 正規化線形関数：

$$f(u) = \max(0, u) = \begin{cases} u & (u > 0) \\ 0 & (u \leq 0) \end{cases} \quad (3.38)$$

この活性化関数を持つユニットをReLU (rectified linear unit) と呼ぶ(図3.7).

このバージョンにリークリReLUがある(図3.7右).

$$f(u) = \alpha \min(0, u) + \max(0, u) \quad (3.39)$$

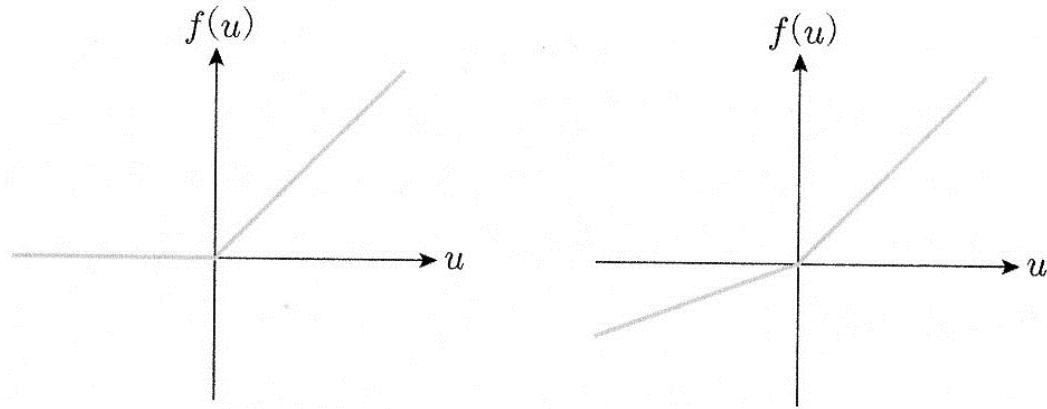


図3.7 左がReLU,  
右がPReLU

$\alpha$ を固定値とするのがリークィReLUで、計算で決まるパラメータとするのがパラメトリックReLUである。

### 3.6.3 マックスアウト

パラメトリックReLUでは $\alpha$ を計算で決める。活性化関数も学習するプローチである。

これをさらに推し進め、一般的な関数形を学習できる区分線形関数を導入できる。

このようなもので最近よく使われるものがマックスアウトである。

以下省略

### 3.7 なぜ深層学習とすることが重要なのか

複雑な学習には多くのパラメータが必要.

パラメータを増やすことは一つの層内で層を広げても  
も、層を多層化することでも可能. どちらが有利？

入出力間の複雑な関係を作るのに、層を広げる効果  
は足し算的、多層化は掛け算的. 多層化が有利.

多層化するのが深層学習.

中間層が $d_0$ 個のニューロンからなる $L$ 層NNにおいて、出力が  
表現できる関数の複雑さは  $(d/d_0)^{Ld_0} (d)^{d_0}$  で計られる.

多層化することにより局所的最小値に囚われ難くなり、  
学習が成功し易くなる可能性も指摘されている.

多層化すると、層が上がるごとに抽象度が進み、高度な学習が可能と考えられる。

ただし、深層化の謎を本当に解き明かすには、今後の理論的進展を待たねばならない。.

# 動物のニューロンの数の一覧

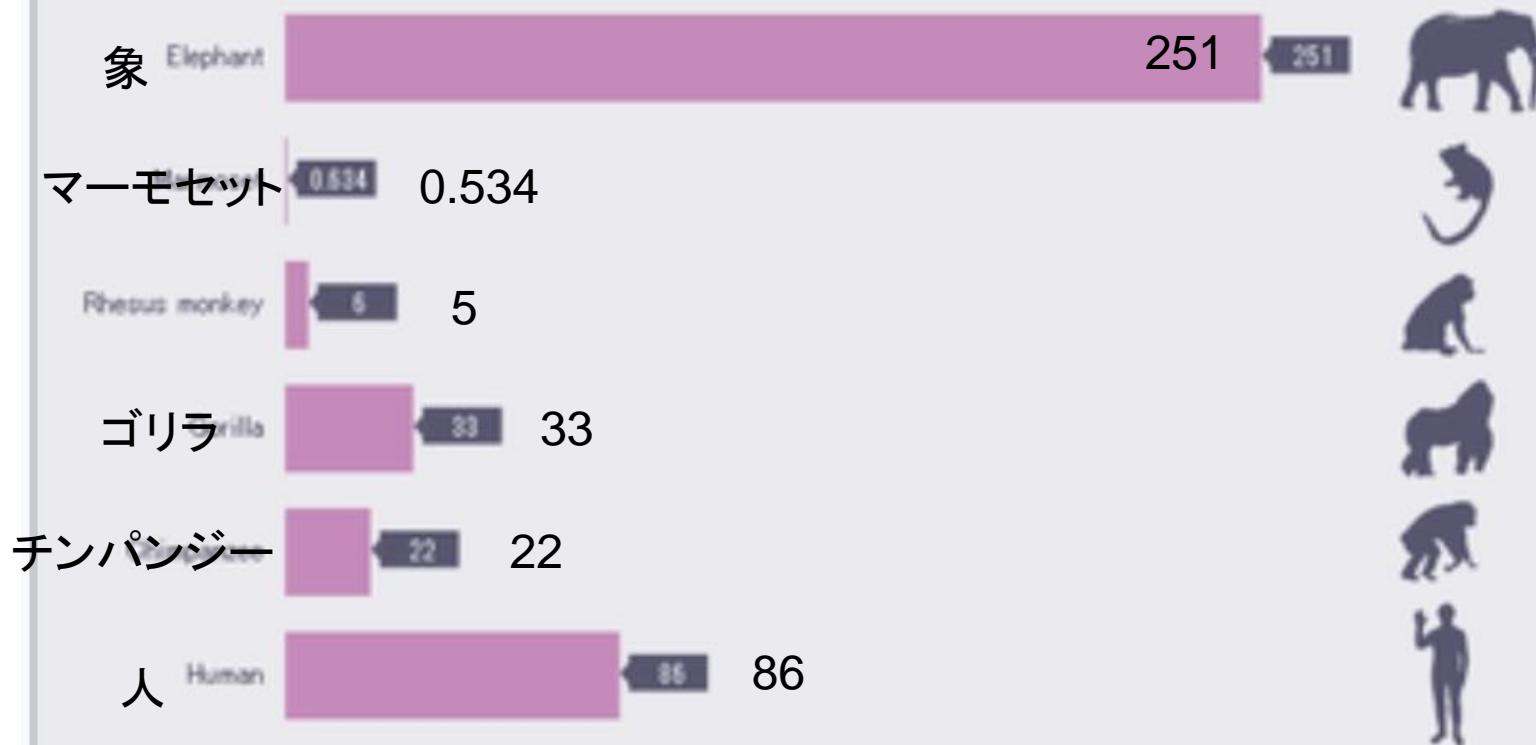
Wikipedia 動物のニューロンの数の一覧

<https://ja.wikipedia.org/wiki/%E5%8B%95%E7%89%A9%E3%81%AE%E3%83%8B%E3%83%A5%E3%83%BC%E3%83%AD%E3%83%B3%E3%81%AE%E6%95%B0%E3%81%AE%E4%B8%80%E8%A6%A7>

ニューロンは、その環境からの刺激を感じし、それに応じて振る舞うことができるよう、動物の神経系に情報を伝達する細胞。**必ずしも全ての動物がニューロンを持っているわけではなく、たとえばセンモウヒラムシや海綿動物は完全に神経細胞を欠いている。**

ニューロンは、**脊椎動物の脳や、昆虫の神経節**のような構造を形成する。

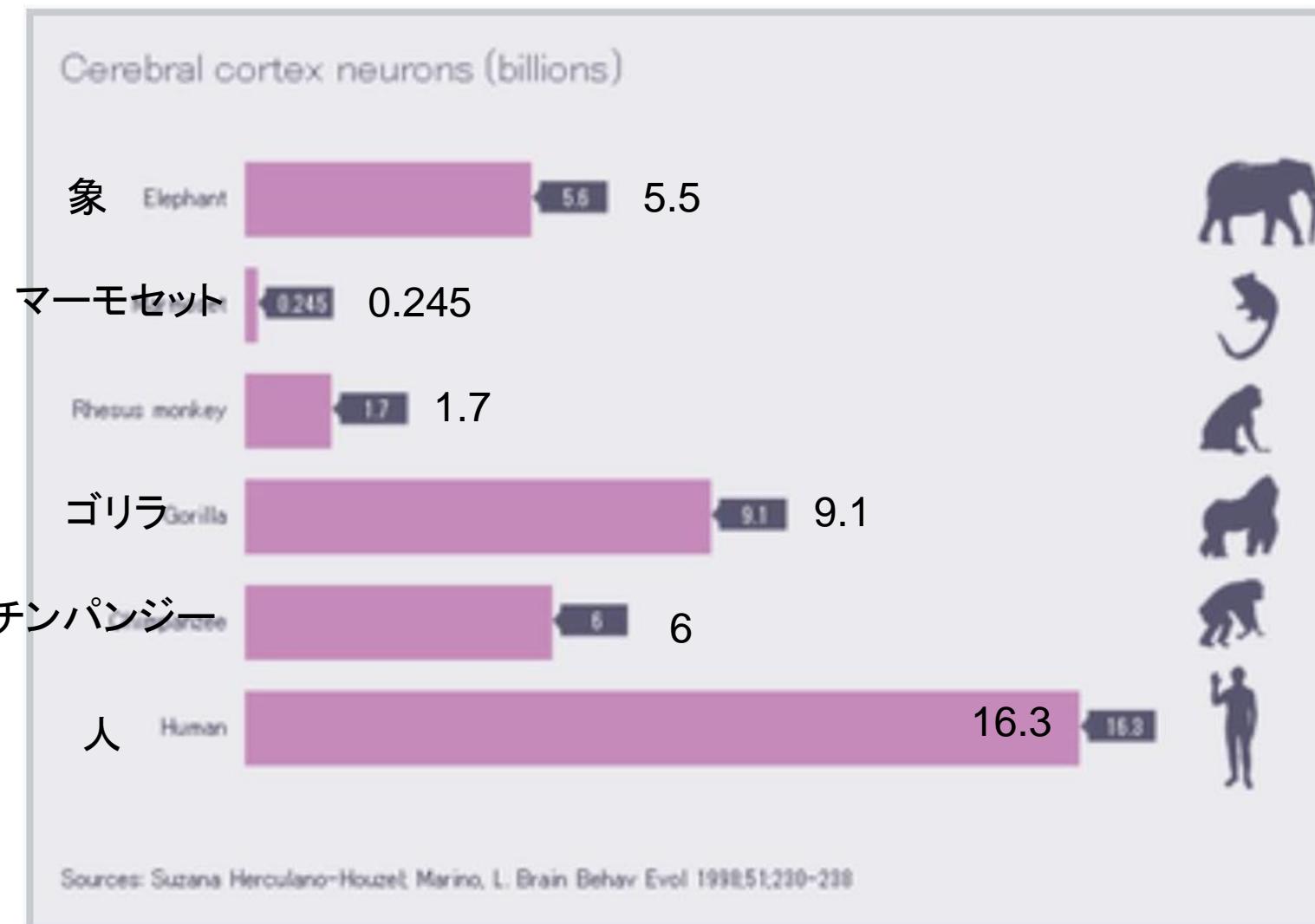
### Brain neurons (billions)



Sources: Suzana Herculano-Houzel; Marino, L. Brain Behav Evol 1998;51:230-238

神経系全体での神経細胞(単位10億)

# 哺乳類のみが大脳皮質を持っている。



大脳皮質の神経細胞(単位10億)

# 植物の神経

植物は太陽の光、気温、湿度という自然環境の中で、自分と子孫のために生きている。

岩石や金属などの無機物も自然環境の中で変化するが、植物の生命活動とは全く異なる。

無機物の変化は自然そのものであるが、植物の変化は与えられた環境の中から最大のメリットを引き出す意図がある。

この意図は個と集団の繁栄を求めるものであり、動物と変わらない。

意図の実現を図ることは一種の知的行動であるから、動物の場合と同じではないが、情報を伝達する仕組み、すなわち神経があると考えられよう。

## 道徳的動物日記

アメリカのVICEというwebページに掲載された、植物学者のダニエル・チャモヴィット氏へのインタビュー記事

<http://davitrice.hatenadiary.jp/entry/2016/01/23/203008>

Q: 私は、オジギソウと呼ばれる植物が、明らかに何かを感じている様子を撮影した動画を見ていて。誰かの手が触れると、オジギソウの葉っぱが閉じていきます…

A: はい、オジギソウは触れられたことを感じています(Feel)。気付いている(Aware)と言ってもいいでしょう。しかし、オジギソウは触れられていることを気にしません(doesn't Care)。葉っぱが切り落とされている時には、葉っぱはそのことを知っていますし、反応もします。しかし、その反応は「なんてことだ！また同じことが起こったら、僕はどうなるんだろう？」というように複雑なものではありません。

A:アブラムシが葉っぱ<sup>。</sup>を攻撃する時、**植物に電気信号が誘発され**、身を守ることを開始するために**葉から葉へと信号が送られる**ことが知られています。この電気信号の伝播の仕方は、神経系で電気信号が伝播する仕方と非常によく似ています。そして、植物は神経系なしで電気信号を送ることができます。ここで覚えてもらいたいことは、神経系は情報を処理する方法の一つではあっても、唯一のものではない、ということです。

**大分怪しいですが、以下のような考え方もあります。**

In Deep

<https://indeep.jp/conscious-and-intelligent-of-plants/>

「植物は人間にさわられると、植物の成長をコントロールする『ジャスモン酸エステル』という植物ホルモンを多く分泌して、より、たくましく成長する」

ということがわかったというのですが、思えば、この、「人間に触れられれば触れられるほど植物は強くなる」なんていうのも、なかなかよくできたメカニズムだと思うのですが、これは平たくいようと、たとえば、農作物でも家庭の植物などでも、

「手間をかけてあげればあげるほど、植物は丈夫で美しく成長する」

カナダ・ブリティッシュコロンビア大学の生態学教授スザンヌ・シマード( Suzanne Simard )氏は、BBC に、「植物たちを扱う際、尊敬の念を持って接すると、彼らは明らかに有情(非常の反対 / 愛憎の心があること)を持ちます」と述べた。

# 飲み会の代わり...自由討論

分科会運営方法に関するコメント

最近の世相、時代の潮流

年寄りは無条件にモノづくりは良いもの、楽しい  
ものと信じている？若者の考え方？

本田宗一郎や松下幸之助の生き方に共感？

夢：遠隔地音楽アンサンブル

時間遅れをどう克服するか？

音楽合奏に関する鍋島さんの経験

ユニソンからアンサンブルへ

ノイズキャンセラーが悪さをする？

A, Bの2信号があると、一方が信号、他方  
がノイズになり、抑えられる。

夢：遠隔地音楽アンサンブル

ヤマハNETDUETTO（宮竹さんからの情報）

現行の通信回線でよい。

各パートの時間遅れを極力小さくする。

ダウンロードの仕方

NETDUETTOラボ <http://netduetto.net/manual/>

7月4日（水）に前田，澤井，鍋島の諸氏と一緒に  
WiFiではなくて，有線接続で実験しました。

音の強弱，テンポが安定して，ようやく合奏出来る  
可能性が出てきました。遅れ時間80ms。

新井先生から，オーディオ・インターフェースの仕組みのご報告をいただけるのを期待しています。