# 先端メディアゼミナールⅡ

伊藤聡志

## 2章　ニューラルネット

# ニューラルネットとは

ニューラルネットとは、人間の脳内の神経細胞の仕組みを参考にして作られた「学習する数理モデル」であり、それまでの主流であった直列処理ではなく、並列処理による論理演算である。

近年では自動販売機などに使われている。

# 直列処理と並列処理の違い

直列処理

- 高速に計算することが可能
- 一部の経路が破壊されると運用できない
- ユニット数が増加すると実行時間も増加する

並列処理

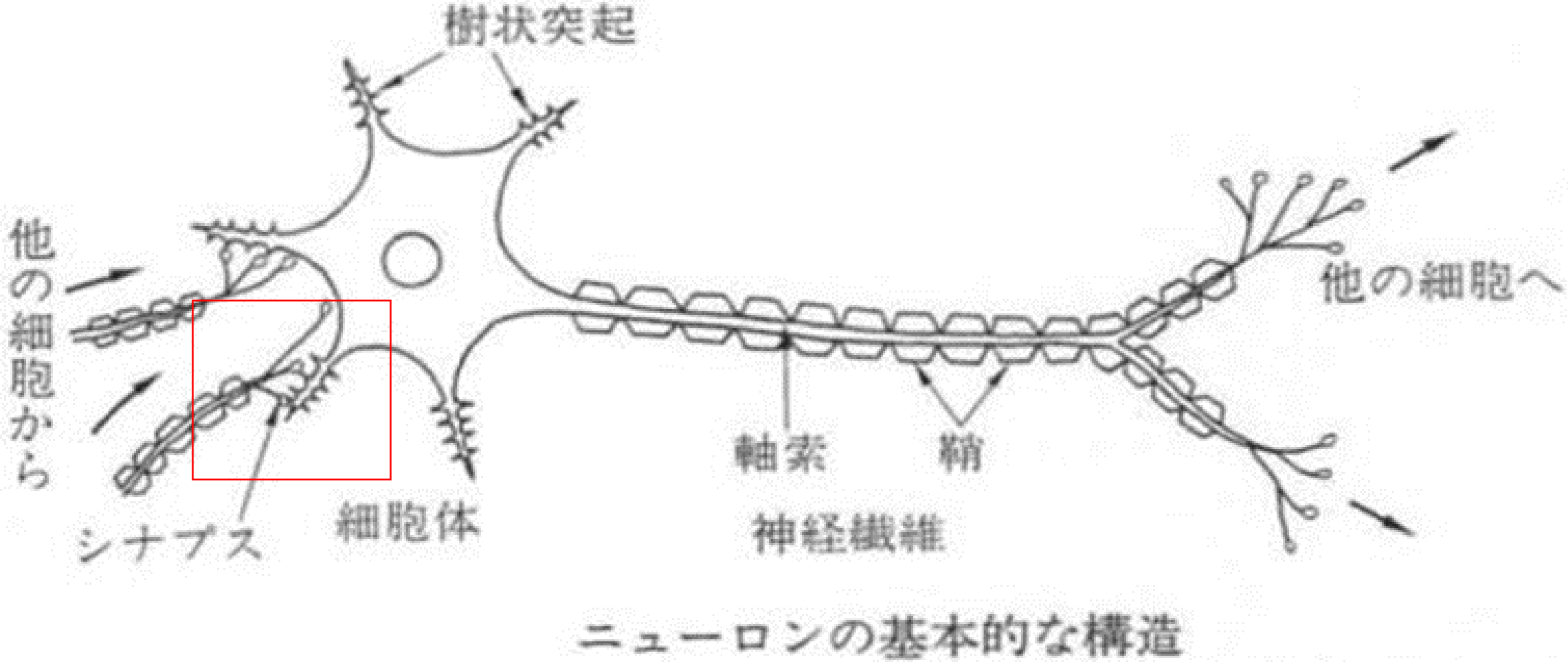
- 一つ一つのユニットの計算速度は遅い
- 一部の経路が破壊されても運用できる
- ユニット数が増加しても実行時間は増加しない

# 脳の神経細胞（１）

脳は約１４０億個のニューロン（神経細胞）から構成されており、個々のニューロンは約１万個の他のニューロンと結びついている。

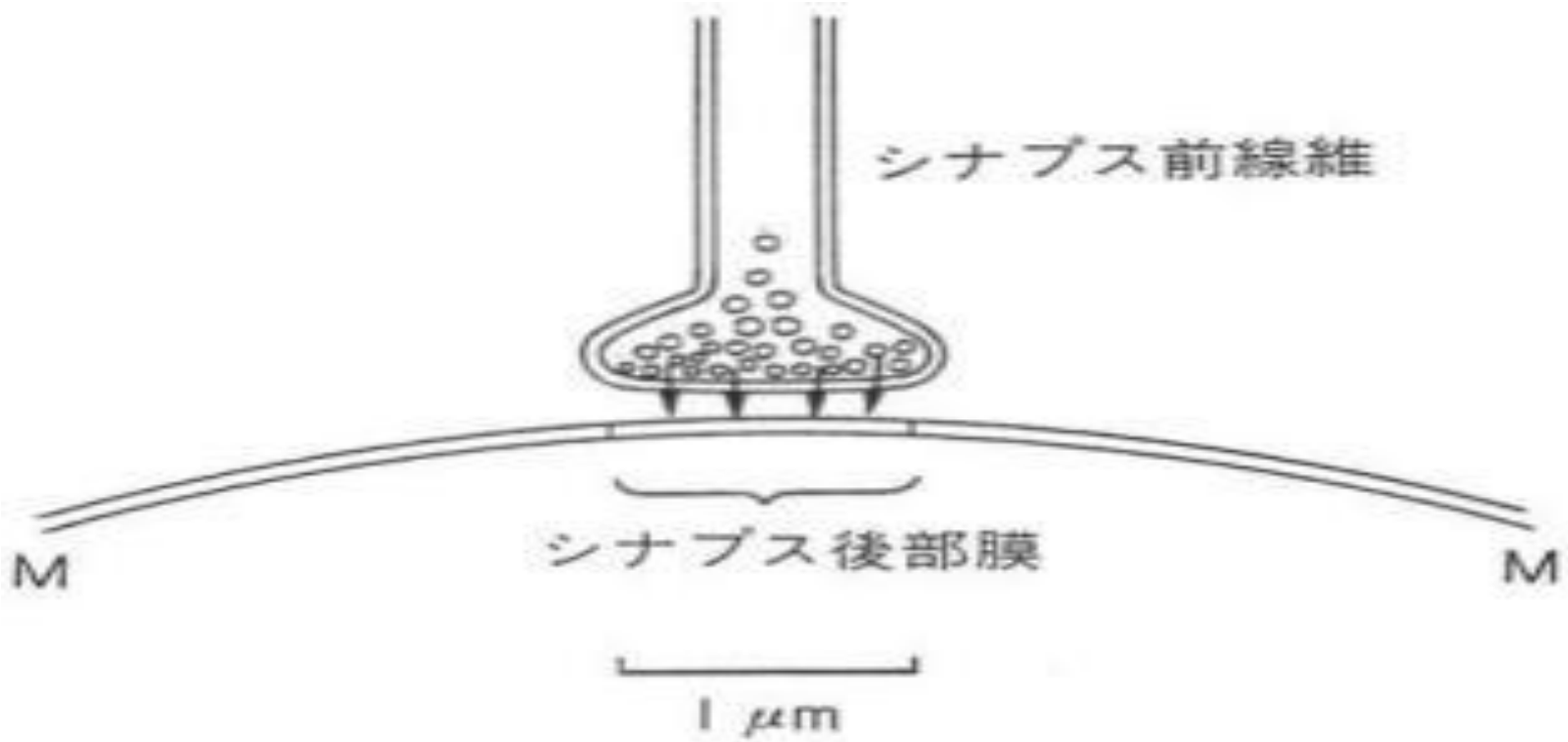
一つ一つのニューロンの処理速度は遅いが、情報が膨大なニューロンに伝わり並列処理されるため、脳全体では高速に情報を処理することができる。

# 脳の神経細胞（２）




ニューロンの基本的な構造

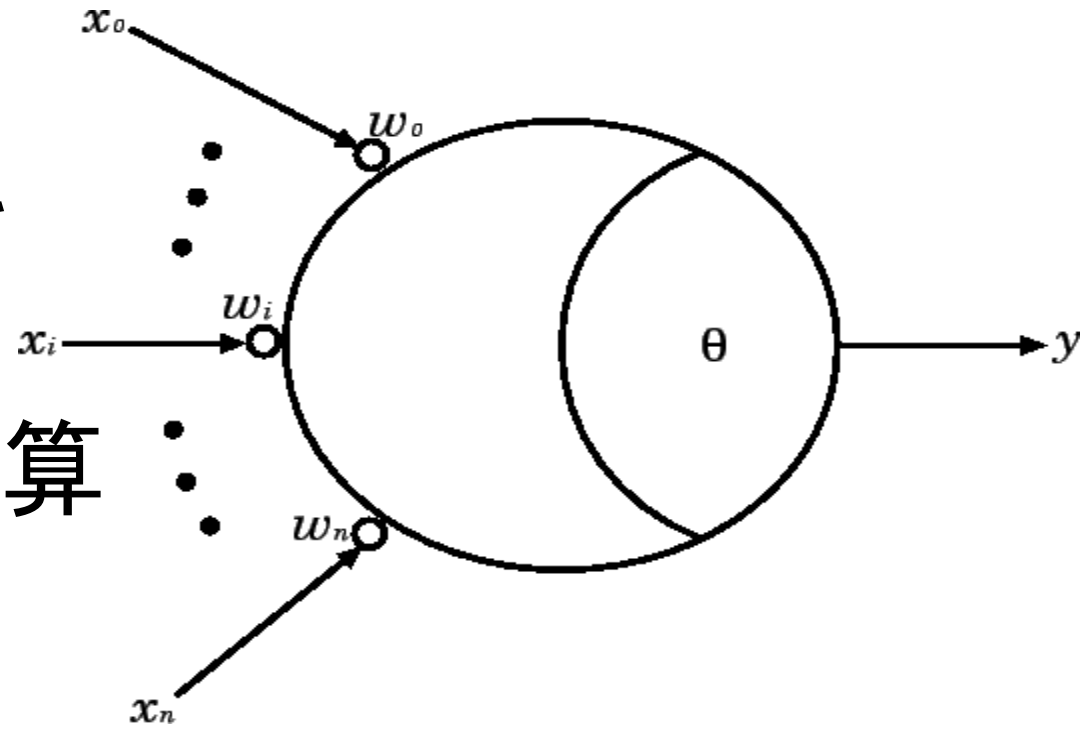
# 脳の神経細胞(3)




シナプス

# マッカロとピッツのモデル

マッカロとピッツは生体の神経細胞の働きを数理モデルで表現した。（1943年）
簡単な仕組みであるのに
OR,AND,NOR,NAND,XOR等を
簡単に表現でき、それらの
組み合わせで高度な論理演算
を表現できる。




↑マッカロとピッツのモデル

# 並列処理の冬の時代

１９４３年にマッカロとピッツによって卓抜なアイデアが出されたにもかかわらず、並列処理による論理演算はしばらく注目されなかった。

１９５８年にローゼンブラットによって「パーセプトロン」（ニューラルネットの初期モデル）が提案される。

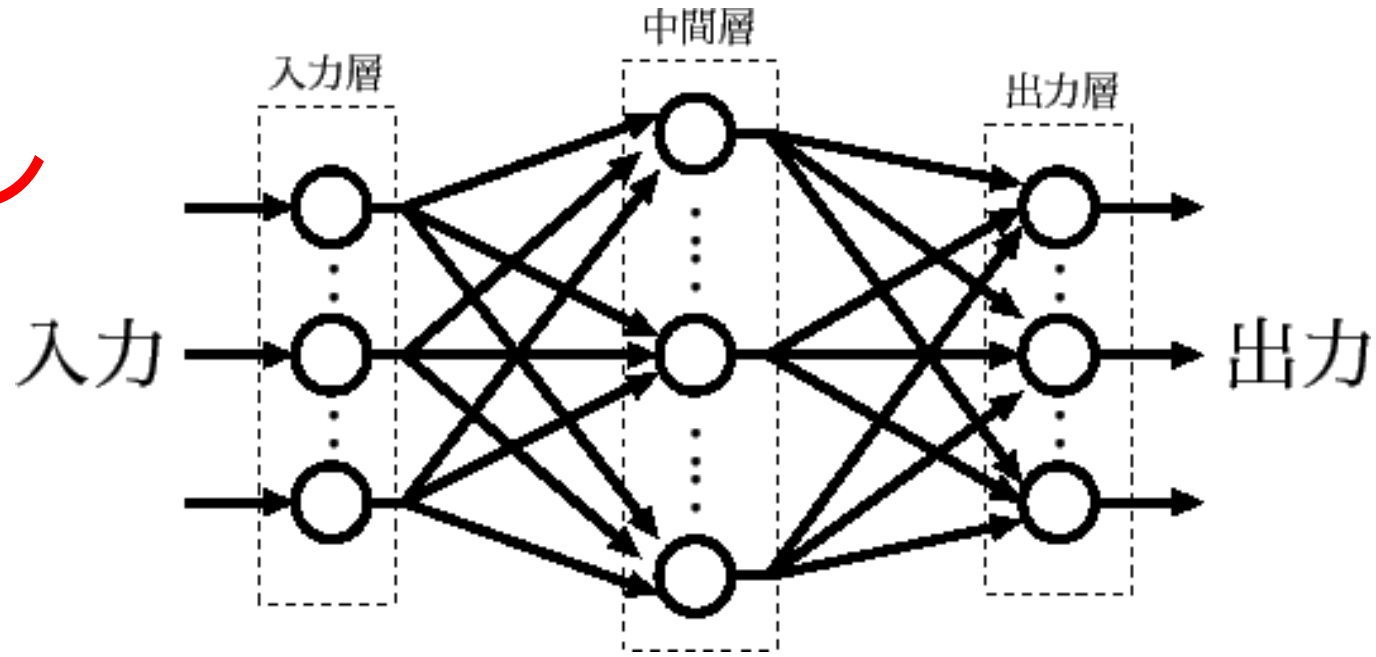
しかし、１９６９年にミンスキーとパパートによって、パーセプトロンは単純な課題しか解けないことが示されてしまう。

# 階層型ネットワークモデル（1）

1986年、ラメルハート、ヒントン、ウィリアムズによって階層型ネットワークモデルと逆伝播学習が提案される。

現代パーセプトロン

ともいう。




↑階層型ネットワークモデル

# 階層型ネットワークモデル（2）

マッカロとピッツのモデルではユニット内部の信号の変換に線形閾値関数を用いていたが、階層型ネットワークモデルではシグモイド関数が用いられている。

線形閾値関数は不連続関数であるためシステム全体を微分できないが、シグモイド関数は連続関数であるため微分が可能である。

脳の情報処理プロセスは離散的であるが、微分が可能だと統計学における最適化が利用できる。

# 逆伝播学習

逆伝播学習とは、階層型ネットワークモデルの学習アルゴリズムとして最も使われているものである。

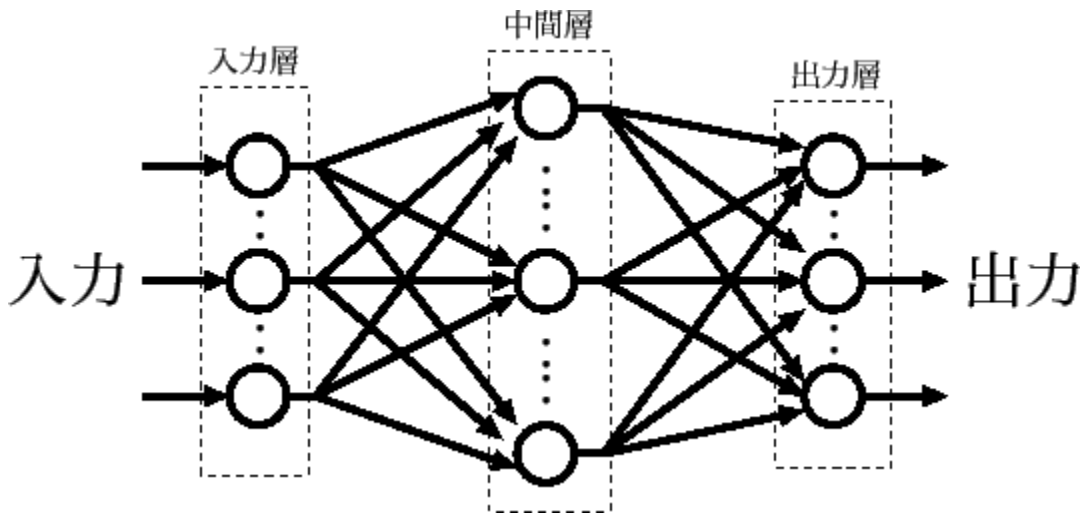
逆伝播法はバックプロパゲーション法、一般化デルタルールとも呼ばれる。

ニューラルネットに教師刺激（正しい反応）を与え、出力と教師刺激の誤差を計算することによって学習をさせる方法。

# ネットワーク・トポロジー

ネットワークの構造のことをトポロジーと呼ぶ。

トポロジーは層の数、各層内のユニット数で表現される。例えば下の図は、層の数：3、各層内のユニット数



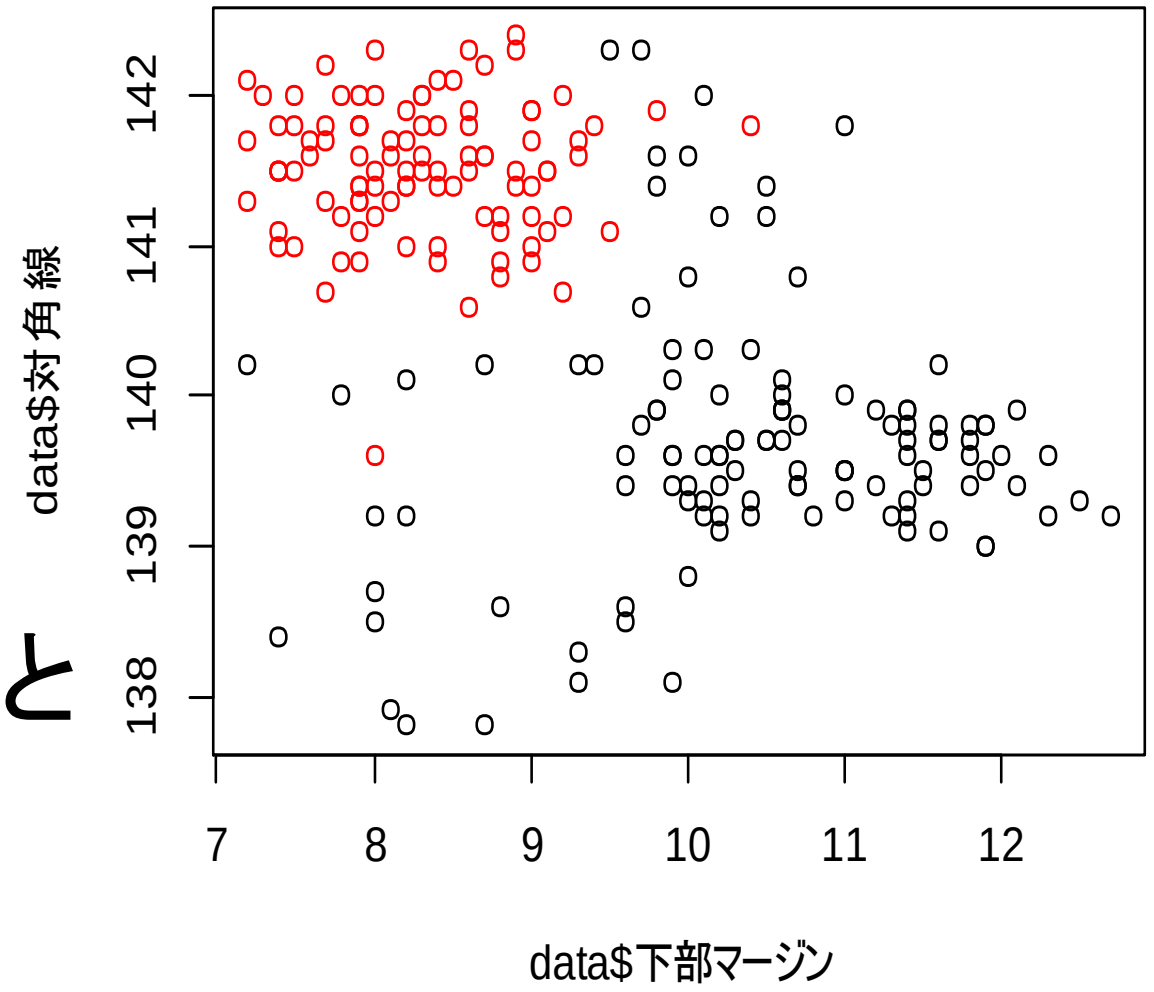

順に：3，3，3　より、

（a=3,b1=3,b2=3,b3=3）

と表される。

# 演習1

お札.csvを読み込み、
真札と偽札の散布図を
求めよ。
真札の点と偽札の点の
区別がつくようにすること



# 演習１　解き方の例

①散布図の枠だけを作る

plot(data$下部マージン,data$対角線,type=“n”)

②”points”で偽札の点だけを書く

points(data$下部マージン[data$真偽==
“偽札”],data$対角線[data$真偽==“偽札”])

③真札の点を色を変えて書く

points(data$下部マージン[data$真偽==
“真札”],data$対角線[data$真偽==“真札”],col=2)

# Rによるニューラルネット

## 偽札データの分析

# パッケージ

Rは関数とデータを機能別に分類してパッケージ（ライブラリ）という形にまとめている。

関数 library() を実行するとどのようなパッケージが用意されてるか知ることができる。

library(パッケージ名)で呼び出す。

初めから入っているパッケージ（標準パッケージ）以外にも、拡張パッケージをインストールすることもできる。

標準パッケージの例

base　Rの基本パッケージ

nnet　ニューラルネット用のパッケージ

# 演習2

教科書P70～P74を読んでRでニューラルネットを使ってみよう。

何行目の予測が間違っているか答えよ。

# 演習２　解説（１）

> getwd()

> setwd("../Dropbox/Share/DL02045/Program/chap2/")

>library(nnet)

　↑パッケージ nnetの呼び出し

>data=read.csv("お札.csv",header=T)

↑１行目は列名であることを示す

# 演習２　解説（２）

> 下部マージン=(data$下部マージン - min(data$下部マージン))/(max(data$下部マージン)　- min(data$下部マージン))

>対角線=(data$対角線 - min(data$対角線))/(max(data$対角線)　- min(data$対角線))

　　↑値を０～１の範囲にする

>入力層=cbind(下部マージン,対角線)

　　↑新たに行列を作る

> 出力層=class.ind(data$真偽)

　　↑真偽を判断する行列に変換する

# 演習2　解説（3）

> head(出力層,n=3)

　　個先頭から3行分のデータを抜き出す

> tail(出力層,n=3)

　　個末尾から3行分のデータを抜き出す

　　さっきの class.indで真札の時は真札に1、偽札に0、偽札の時は

　　偽札に1、真札に0が表示されるようになっている。

# 演習２　解説（４）

> set.seed(3)

　　↑学習した結果をそろえるためにシード値を決める

> ネットワーク学習=nnet(入力層,出力層,size=3,rang=0.3,maxit=2000)

　　↑学習の仕方を関数 nnet()で決める

　　size:中間層のユニット数

　　rang:重みの初期値の範囲

　　maxit:学習の最大くりかえし数

# 演習２　解説（５）

>ネットワーク適用=predict(ネットワーク学習,入力層)

　　↑ネットワークの適用をしている

> 適用結果=round(ネットワーク適用,digits=3)

　　↑見やすくするために関数 round()で小数点以下第３桁で丸める

ここまでやったら> head(適用結果,n=215)でデータ全体を見る

０．５より大きい方が予想の答え

正しくできていると間違いが3つ見つかるはず（精度９８．６％）

# まとめと感想

脳の仕組みを参考にして作られた学習する数理モデル「ニューラルネット」

パッケージ（ライブラリ）を用いることによってRはかなり多くのことが出来るようになる。

今まで人間しかできなかった「なんとなく」を機械にも出来るようにしたニューラルネットは間違いなくこの先様々な場面で重要視される。

# 宿題

①演習２で求めた結果を使い、演習１で求めた散布図のどの３点の予測が間違っているか求めよ。

②学習最大くりかえし回数を５０にした結果を求めよ。

（発展）組み込みデータフレーム irisのデータを読み込み、ニューラルネットを用いて Sepal.Length. Sepal.Width. Petal.Length. Petal.Width.の４つの値から Speciesを予想しなさい。