情報学習理論

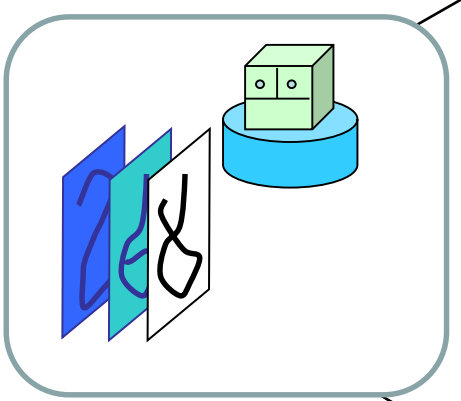
渡辺澄夫
東京工業大学

教師あり学習の枠組み

学習データ

$X_1, X_2, \cdots, X_n$

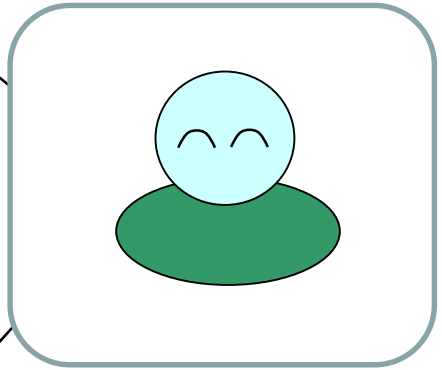
$Y_1, Y_2, \cdots, Y_n$



テストデータ

X

Y



情報源

学習モデル

復習：学習と汎化

学習誤差関数

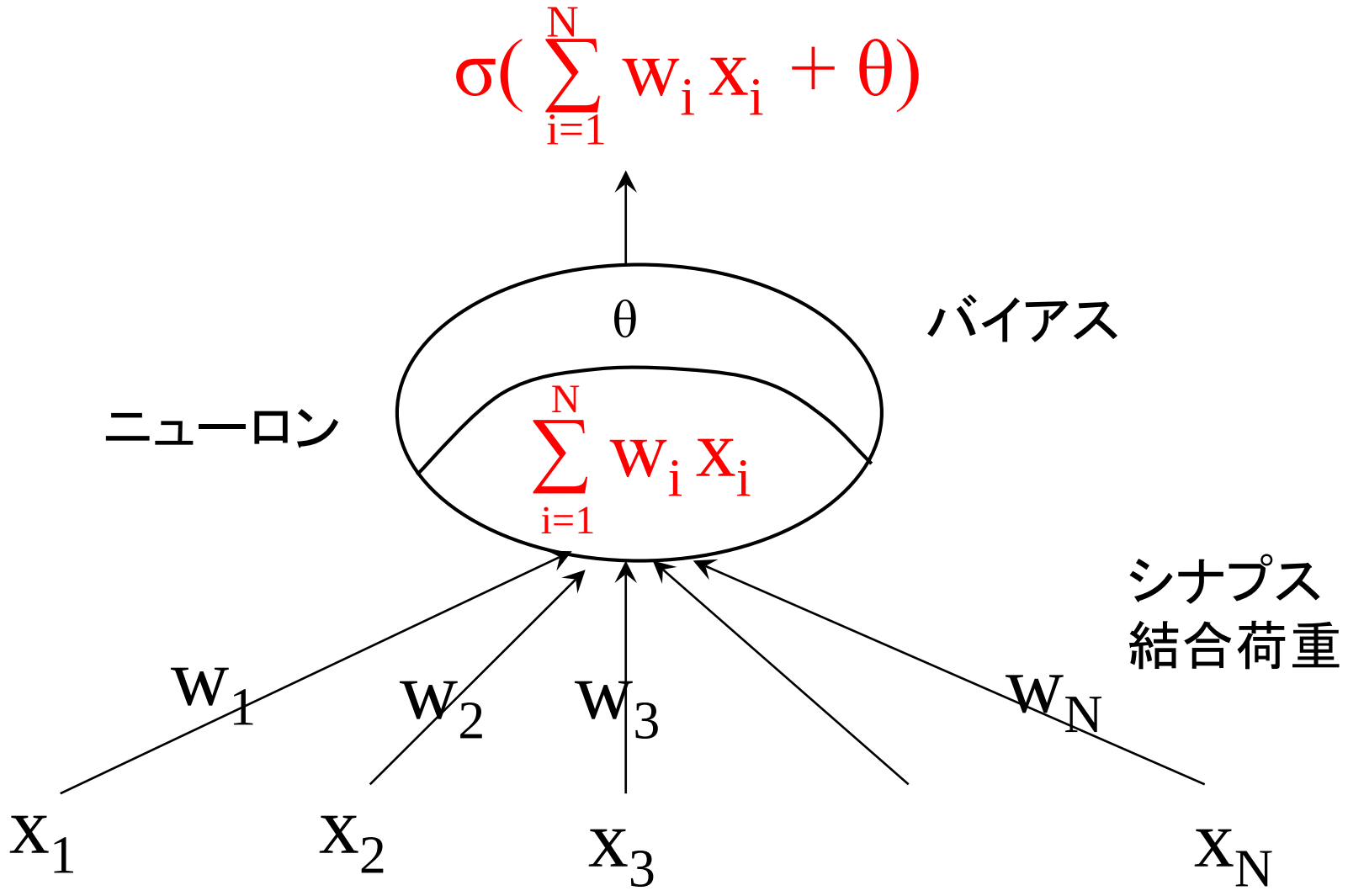
$$E(w) = (1/n) \sum_{i=1}^{n} (Y_i - f(X_i, w))^2$$

汎化誤差関数

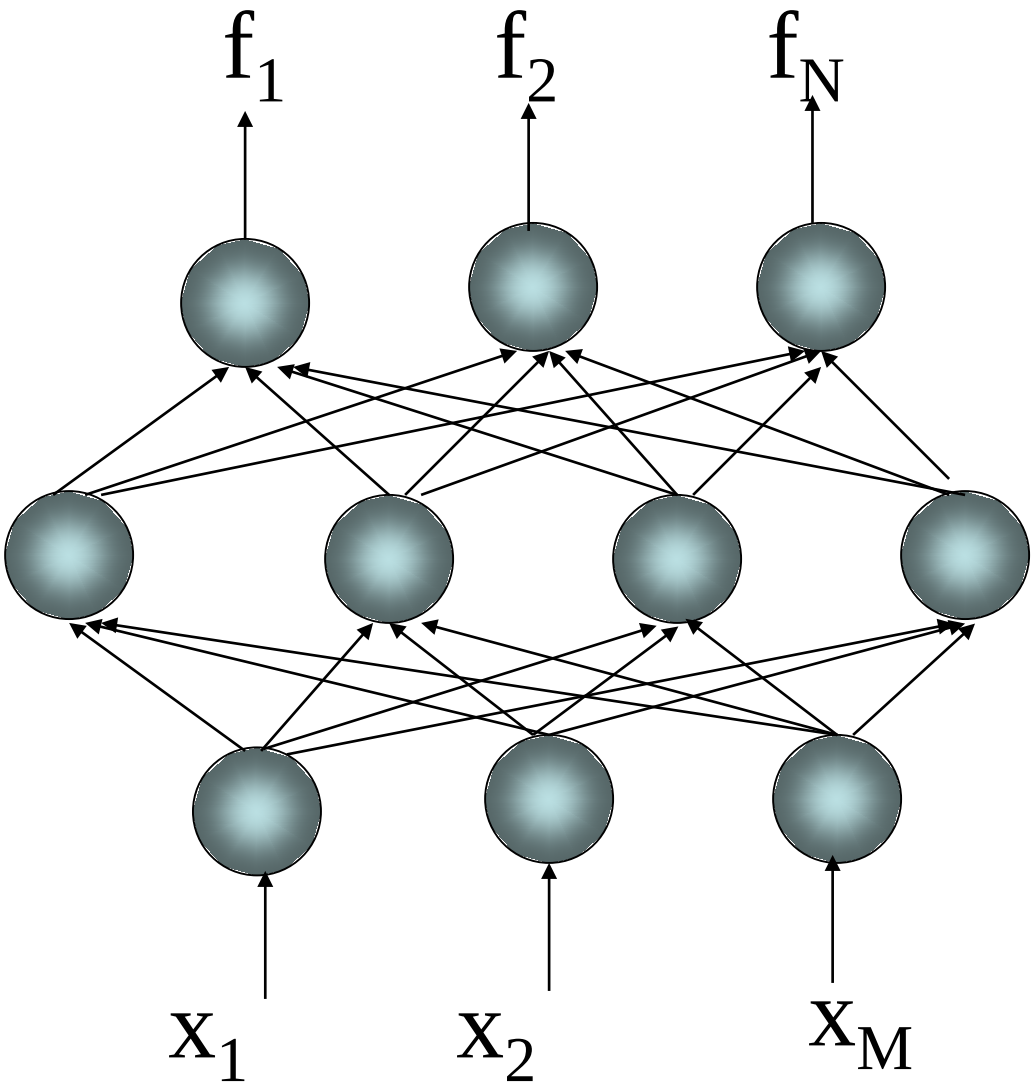
$$F(w) = \iint (y - f(x,w))^2 \; q(x,y) \; dxdy$$

学習の真の目的は汎化誤差を小さくすることである

復習：神経素子


$\sigma(\sum_{i=1}^{N} w_i x_i + \theta)$
θ
バイアス
ニューロン
$\sum_{i=1}^{N} w_i x_i$
シナプス
結合荷重
w_1
w_2
w_3
w_N
x_1
x_2
x_3
x_N

復習：三層パーセプトロン



故郷から旅立つ

関数近似能力（1）

入江文平，舟橋賢一，サイベンコ (1987)

どんな連続関数も、
中間ユニットを十分たくさんとって
パラメータを適切に定めれば
3層パーセプトロンによって近似できる。

※多項式でも、三角関数でも、できる。

※「Weierstrassの多項式近似定理」

※関数空間のトポロジーは、同値でないものがたくさんあって、「近似できる」という意味にもいろいろなものがある。

関数近似能力（2）

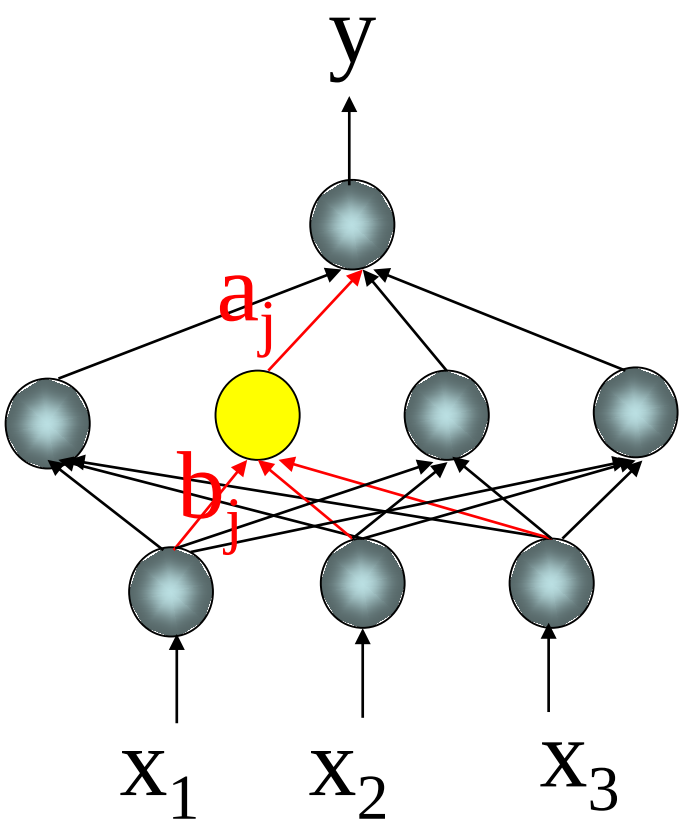
バロン(1993)

3層ニューラルネットは、
多項式や三角関数よりも
遥かに優れた近似効率を持つ。

※　神経回路網は、多項式や三角関数とは違うのか？
という問いかけがなされてきた。関数近似の意味では
その違いが示されている。

※　実は、多項式や三角関数でも、目標関数に合わせて
基底関数を選ぶことができれば同じオーダーの精度になる。
基底関数を最適化できるかどうかが本質的な違いである。

関数近似能力(3)



$$F(x) = \sum_{j=1}^{H} a_j \, f(b_j, x)$$

単純パーセプトロン、SVM 、
カーネルマシンでは $\{a_j\}$だけが
パラメータ

3層パーセプトロン、深層学習
では$\{ a_j, \ b_j \}$ の両方がパラメータ

$\{b_j\}$ は関数空間(関数の空間)を最適化し、
$\{a_j\}$は空間内での最適化を行っている。

◎ カーネルマシンの提案は、小川英光(1993)、SVM(1995)など

関数近似能力(4)

従来のモデル: (M=dim x)

どのように$\{b_j\}$を選んでも、ある関数 g とある H が存在して

$$\min_{\{a_j\}} \| g(x) - \sum_{j=1}^{H} a_j f(b_j, x) \|^2 \geqq C_1(g) / H^{2/M}$$

階層モデル: 任意の関数 g と任意の H に対して

$$\min_{\{a_j,b_j\}} \| g(x) - \sum_{j=1}^{H} a_j f(b_j, x) \|^2 \leqq C_2(g) / H$$

「次元の呪い」 は解かれた

Curse of dimensionality

画像・音声・自然言語は超高次元空間上にある。

人工知能を作るということは、超高次元空間上の関数または
確率密度関数を作るということである。

ものすごくたくさんのデータがあるように見える場合でも
超高次元空間上ではスパースでしかない。

例：10^{23} 個のデータがあっても 1000×1000画素の画像空間では
一次元あたり $(10^{23})^{1/(10^6)}$ ＝1.00005 個しかデータがない。

次元の呪い 「人工知能が関数または確率密度関数で表せた
としても、超高次元空間では学習によって作ることは不可能だろう」

と思われてきた。しかし
階層型学習モデルによって、この課題は解決された・・・。

注意： 学習モデルの能力とは

だが・・・ しかし・・・

任意の関数が近似できることは、どんな問題でも解決できることではない。

関数近似能力

どれだけ、たくさんの関数を近似できるか？

一般的には　両立しない

汎化能力

少ない個数の例から、未知のものに対して
どれだけ正確な予測ができるか？

◎ 関数近似能力については解明された。

◎ 汎化能力についても解析が進みつつある。

問1

確率分布 $q(x)$ と $q(y|x)$ が与えられたとき
次の二乗誤差 $E(g)$ を最小にする関数 $g(x)$ と
$E(g)$の最小値を $q(x)$ と $q(y|x)$ を用いて表せ。

$$E(g) = \int\int (y-g(x))^2 \, q(y|x) \, q(x) \, dx \, dy$$

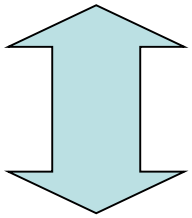
目的に合わせて相応しいモデルを利用する

汎化能力とモデルの設計

テストデータについての誤差（汎化誤差）を小さくできる能力を汎化能力という。

汎化能力の向上のための二つの考えかた

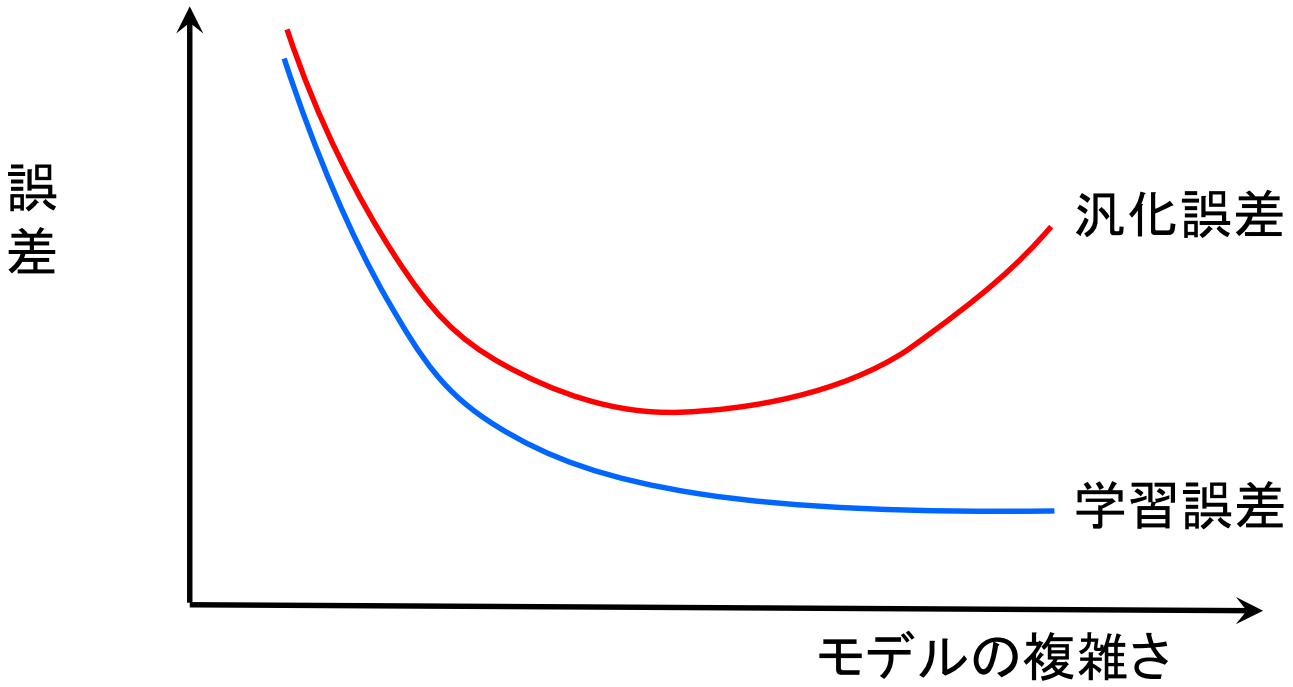
できるだけ単純なモデルを用いる



複雑なモデルを用いて学習方法を工夫する

方法(1)「最小二乗法＋モデル選択」

モデルが複雑であるほど 「汎化誤差ー学習誤差」 は
大きくなる(オーバーフィット)。



最小二乗法は「汎化誤差ー学習誤差」が大きいので、小さなモデルを
使わなくてはならない。このため神経回路網の長所が活かせない。

最尤法は1920頃 Fisher 等に支持されたが最良の方法でないことがわかった。

方法(2)「正則化」

学習誤差関数に正則化項を加えて最小化 (Tikhonov)

$$H(w) = \sum_{i=1}^{n} (Y_i - f(X_i, w))^2 + R(w)$$

例　λ>0 (ハイパーパラメータ)として

Ridge 項　$R(w) = \lambda \Sigma |w_j|^2$

Lasso 項　$R(w) = \lambda \Sigma |w_j|$

汎化能力が向上するかどうかは、真の分布、学習モデル、正則化項、ハイパーパラメータによって異なる。どのようにλを最適化すると良いのだろうか。

◎　Ridge : 大昔からあるので最初の研究は不明。たぶんTikhonov。
◎　Lasso : 石川眞澄(1990)、Tibshirani (1996) など。

方法(3)「ベイズ予測」

パラメータを確率分布にして平均関数で予測する

$$H(w) = \sum_{i=1}^{n} (Y_i - f(X_i, w))^2 + R(w)$$

$$p(w) \propto \exp(-H(w)) \quad : \text{事後分布}$$

$$f^*(x) = \int f(x,w)\, p(w)\, dw \quad : \text{ベイズ予測}$$

ベイズ法あるいはアンサンブル学習によって予測精度は向上するが
事後分布に従うパラメータ生成の演算量が大きかった。しかし近年、
ベイズ法もコンピュータで実現できるようになり、高精度な予測が
必要な問題を中心に使われるようになっている。

ベイズ法は1760頃からあるが広く使われるようになったのは1985頃から。

汎化能力の探求

◎　神経回路網の最小二乗推定における汎化誤差は、従来の統計学の方法では解析できない（萩原克幸・戸田尚宏・臼井支朗, 1992）。

最小二乗法はオーバーフィットが大きい。

最小二乗法を用いて汎化誤差を小さくするには小さなモデルを適用する必要がある。また AIC, BIC では神経回路網を評価できない。

◎　神経回路網にベイズ法を適用したときの汎化誤差は、新しい数学的方法によって解析できる(2001)。

ベイズ法ではオーバーフィットが小さい。

ベイズ法では大きめのモデルを用いてもオーバーフィットが小さく関数近似能力を活用することができ、汎化誤差を小さくできる。
神経回路網にも適用できる評価法がある(WAIC,WBIC)。

神経回路網は召喚魔法？

ニューラルネットワークをさらに多層にしたものが深層学習。

深層学習は、画像・音声・自然言語のように対象が複雑になればなるほど、問題が大規模で難しくなればなるほど実力を発揮するが、適切に使うにはノウハウが必要になる。
このため「深層学習は召喚魔法（？）」という意見もある。

- ◎ ネットワーク構造の決め方
- ◎ パラメータ初期値の設計法
- ◎ 正則化項の作り方
- ◎ 最急降下法のとめ方
- ◎ アンサンブルの作り方　など

召喚士 募集中

深層学習を
自由自在に操り
真に強い敵と
戦える才能を持つ
人を探しています。

「誰でも使えるようになってほしい」という要望が多数。

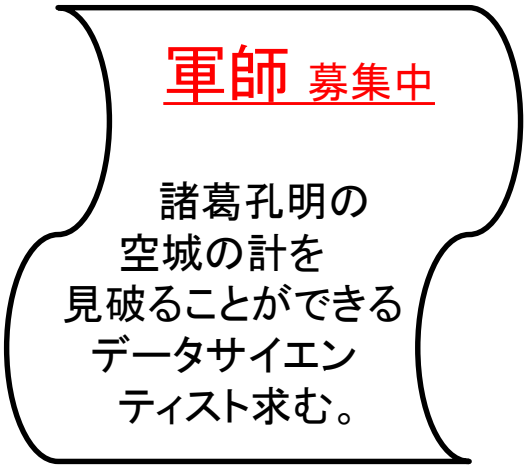
複雑なモデルが必要でないこともある

簡単な問題なら「神経回路網＋ベイズ」を使わなくても
「線形回帰＋最尤」で解決することも多いので、そのような
場合にまで無理して神経回路網を使う必要はありません。

高度な学習モデルを使いこなすことができる一方で
目的に合わせて簡単な方法も選べるのが真の上級者。

最強の方法は、それに相応しい
真に困難な問題に適用しましょう。

※　難しい問題とは：　変数間の関係を
簡単な方程式で書けないもの。



学習システムの応用(1)

神経回路網に限らず、いろいろな学習システムがある。
サポートベクトルマシン、隠れマルコフモデル、混合正規分布、
ボルツマンマシン、ベイズネットワーク、・・・、深層学習。

様々なことに応用されている。
音声画像の処理・認識、ロボット制御、医療情報処理、
ひとのモデリング、自動運転、経済予測、消費者解析、・・・。

応用をする場合、実務の知識と経験が必要です。
どのモデル・方法が適切かは現実をよく考えて決めます。

データ解析のための基本ツールも知っておきましょう。

学習システムの応用(2)

学習理論・統計学・人工知能は、応用の世界から見たら
専門的知識や洞察力がいらない「道具」であることが望ましい。

しかし、実際は、学習することや推測することについての
広い知識と深い理解が応用の場面でこそ必要になる。
（ → 成功・不成功は担当者の能力に強く依存する）

このため、学習理論・統計学・人工知能の応用を
実務で行うには「ひと」が必要です。仕事名は、研究開発、
データ分析、経営企画、ソフトウエア工学、いろいろです。

最近ではデータの次元や量が非常に大きくなり、広い領域で
「データを解析するひと」が必要とされるようになってきました。

次週予告:「大都会SVMへ」

ボルツマンマシン

深層学習

ニューラルネットワーク

自己組織化

教師なし学習

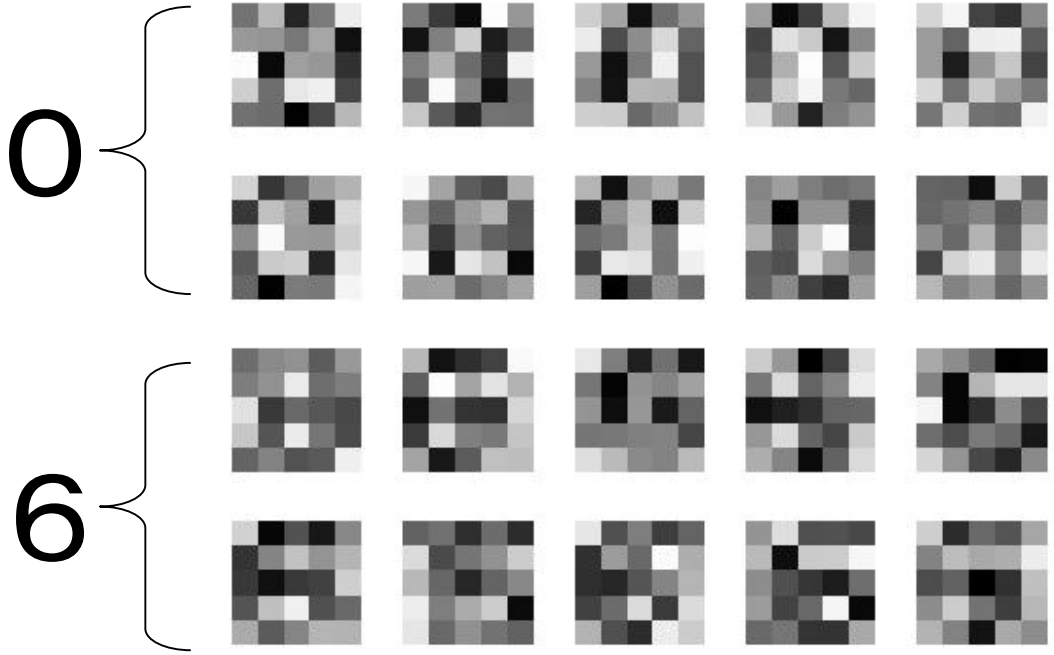
競合学習

教師あり学習

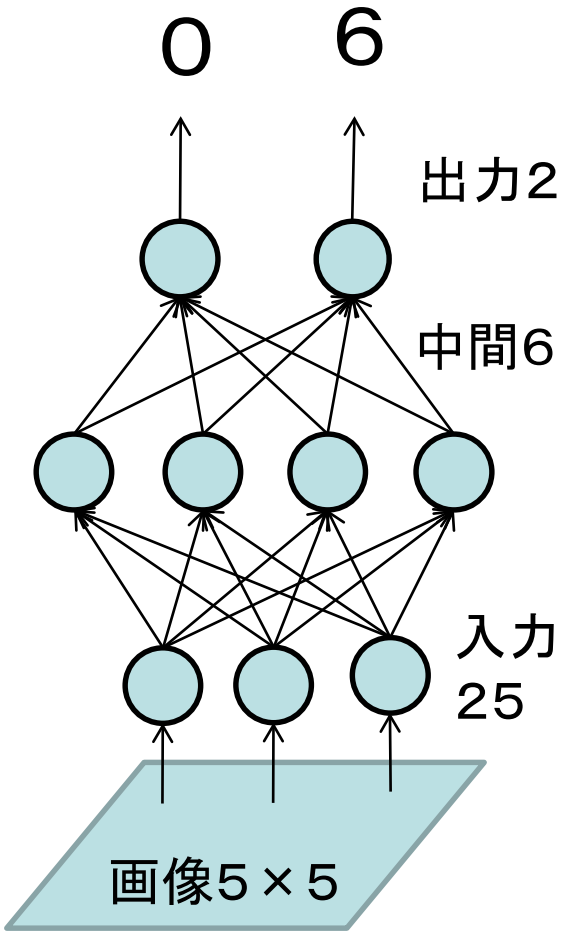
サポートベクタマシン

問題2

文字識別の問題　５×５
学習データ400個
テストデータ400個



ニューラルネット
入力25　中間6　出力2



Ridge と Lasso で学習誤り数、汎化誤り数、
結合荷重の様子を観察してみましょう。

問題2

	学習時の誤り個数	テストの誤り個数	結合荷重の様子
二乗誤差			
RIDGE			
LASSO			