

第13回講義「パターン認識」

[1] 次元圧縮

- 主成分分析と固有次元
- ニューラルネットによる次元圧縮

[2] 学習と汎化

- 問題の所在

[3] 最後に

次元圧縮とは

d -次元空間の x

$$x = (x_1, \dots, x_d)^t$$



$$d > M$$

M -次元空間の z

$$z = (z_1, \dots, z_M)^t$$

情報の損失を最小限に抑える

(1) 教師なし次元圧縮： → 主成分分析 (PCA)

入力空間 x_n の分布のみに着目する

(2) 教師あり次元圧縮：

分類のための教師データ t^n も考慮する

主成分分析 (1)

d 個の正規直交ベクトル $\{\mathbf{u}_i\}_{i=1}^d$ による \mathbf{x} の表現:

$$\mathbf{x} = \sum_{i=1}^d z_i \mathbf{u}_i \quad \text{where } \mathbf{u}_i^t \mathbf{u}_j = \delta_{ij}$$

z_i を求めると $\{\mathbf{u}_i\}_{i=1}^d$ の直交性より $z_i = \mathbf{u}_i^t \mathbf{x}$

M 次元分の $\{z_i\}_{i=1}^M$ のみを保持すると

$$\tilde{\mathbf{x}} = \sum_{i=1}^M z_i \mathbf{u}_i + \sum_{i=M+1}^d b_i \mathbf{u}_i \quad \text{where } b_i : \text{constants}$$

主成分分析 (2)

データセット $\{\mathbf{x}_n\}$ に対しては

$$\mathbf{x}_n = \sum_{i=1}^d z_i^n \mathbf{u}_i \quad \rightarrow \quad \tilde{\mathbf{x}}_n = \sum_{i=1}^M z_i^n \mathbf{u}_i + \sum_{i=M+1}^d b_i \mathbf{u}_i$$

近似の誤差は
$$\mathbf{x}_n - \tilde{\mathbf{x}}_n = \sum_{i=M+1}^d (z_i^n - b_i) \mathbf{u}_i$$

誤差の評価関数:

$$E_M = \frac{1}{2} \sum_{n=1}^N \|\mathbf{x}_n - \tilde{\mathbf{x}}_n\|^2 = \frac{1}{2} \sum_{n=1}^N \sum_{i=M+1}^d (z_i^n - b_i)^2$$

主成分分析 (3)

最適近似の条件:

$$E_M = \frac{1}{2} \sum_{n=1}^N \|\mathbf{x}_n - \tilde{\mathbf{x}}_n\|^2 = \frac{1}{2} \sum_{n=1}^N \sum_{i=M+1}^d (z_i^n - b_i)^2$$
$$\rightarrow \min \text{ for } \{\mathbf{u}_i, b_i\}_{i=M+1}^d$$



$$\frac{\partial E_M}{\partial b_i} = \sum_{n=1}^N (z_i^n - b_i) = 0$$

$$\therefore b_i = \frac{1}{N} \sum_{n=1}^N z_i^n = \mathbf{u}_i^t \bar{\mathbf{x}} \quad \text{where } \bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

主成分分析 (4)

最適近似の条件:

$$E_M = \frac{1}{2} \sum_{n=1}^N \sum_{i=M+1}^d (z_i^n - b_i)^2 = \frac{1}{2} \sum_{n=1}^N \sum_{i=M+1}^d \{ \mathbf{u}_i^t (\mathbf{x}_n - \bar{\mathbf{x}}) \}^2$$

$$\rightarrow \min \text{ for } \{ \mathbf{u}_i \}_{i=M+1}^d$$

書き換えると:

$$E_M = \frac{1}{2} \sum_{i=M+1}^d \mathbf{u}_i^t \Sigma \mathbf{u}_i \rightarrow \min \text{ for } \{ \mathbf{u}_i \}_{i=M+1}^d$$

$$\text{where } \Sigma \equiv \sum_n (\mathbf{x}_n - \bar{\mathbf{x}}) (\mathbf{x}_n - \bar{\mathbf{x}})^t$$

主成分分析 (5)

$$E_M = \frac{1}{2} \sum_{i=M+1}^d \mathbf{u}_i^t \Sigma \mathbf{u}_i \rightarrow \min \text{ for } \{\mathbf{u}_i\}_{i=M+1}^d$$

$$\text{where } \Sigma \equiv \sum_n (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^t$$

Lagrange の未定係数法:

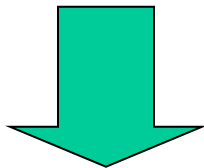
$$\hat{E}_M = \frac{1}{2} \sum_{i=M+1}^d \mathbf{u}_i^t \Sigma \mathbf{u}_i - \frac{1}{2} \sum_{i=M+1}^d \lambda_i (\mathbf{u}_i^t \mathbf{u}_i - 1)$$

$$\frac{\partial \hat{E}_M}{\partial \mathbf{u}_i} = \Sigma \mathbf{u}_i - \lambda_i \mathbf{u}_i = \mathbf{0} \quad \therefore \Sigma \mathbf{u}_i = \lambda_i \mathbf{u}_i$$

主成分分析 (6)

E_M の最小化:

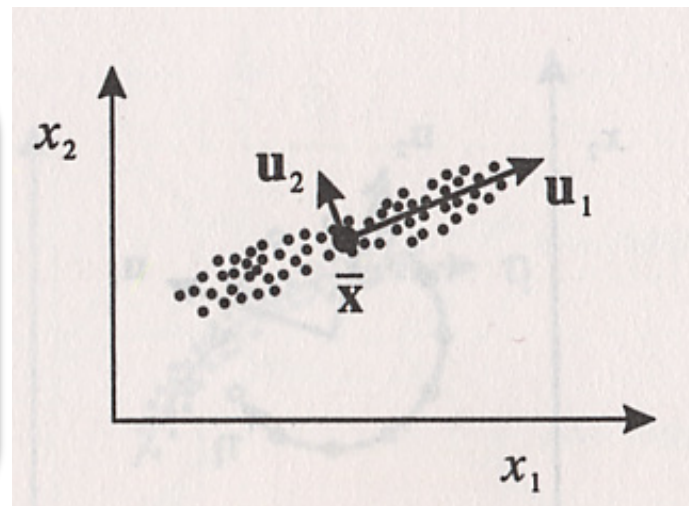
$$E_M = \frac{1}{2} \sum_{i=M+1}^d \mathbf{u}_i^t \Sigma \mathbf{u}_i = \frac{1}{2} \sum_{i=M+1}^d \mathbf{u}_i^t \lambda_i \mathbf{u}_i = \frac{1}{2} \sum_{i=M+1}^d \lambda_i$$



$$\Sigma \mathbf{u}_i = \lambda_i \mathbf{u}_i \quad (1 \leq i \leq d)$$

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{M+1} \geq \dots \geq \lambda_d$$

Karhunen-Loève transformation



固有次元の検出

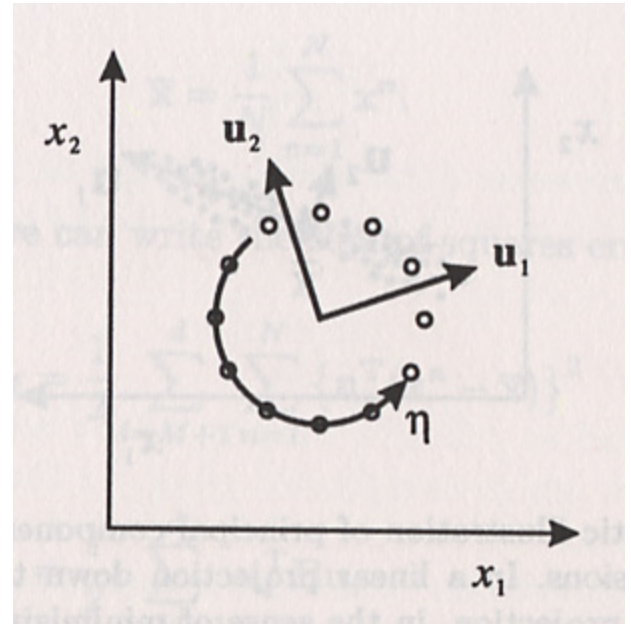
d 次元データの主成分分析



もし最初の k 個の固有値が残り $d - k$ 個よりはるかに大

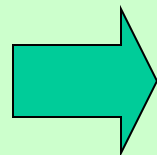


k が固有次元



非線形の場合は？

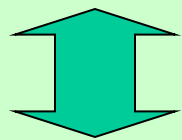
主成分分析は
線形変換



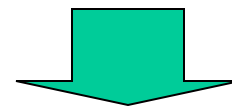
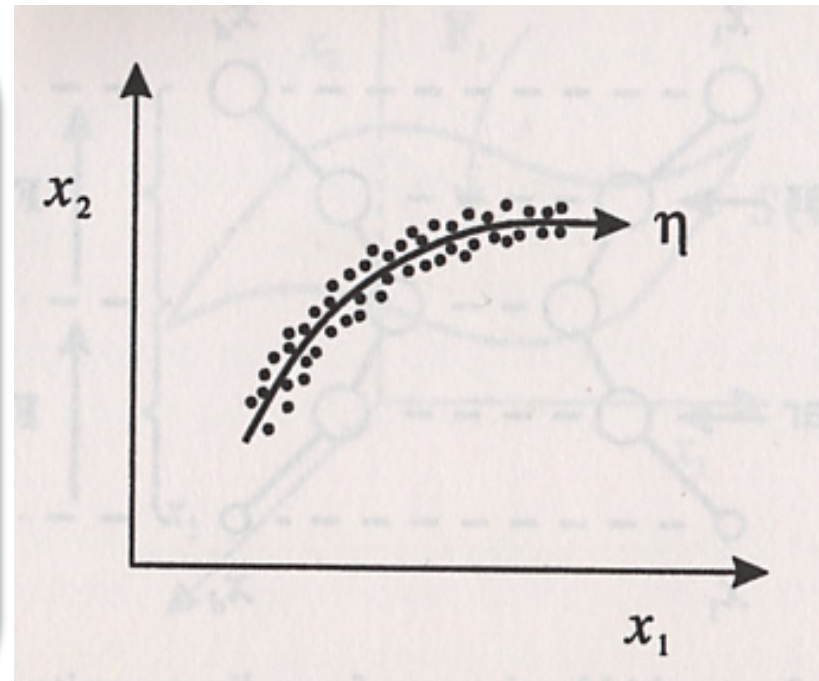
非線形の相関は
検出できない！

固有次元の基本的考え方

d 次元データの固有次元が k である



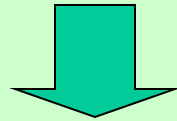
データが k 次元の部分空間に分布する



右上図のように雑音を重ねると分布は2次元になるしかし、固有次元は η の1次元と捉えてよい

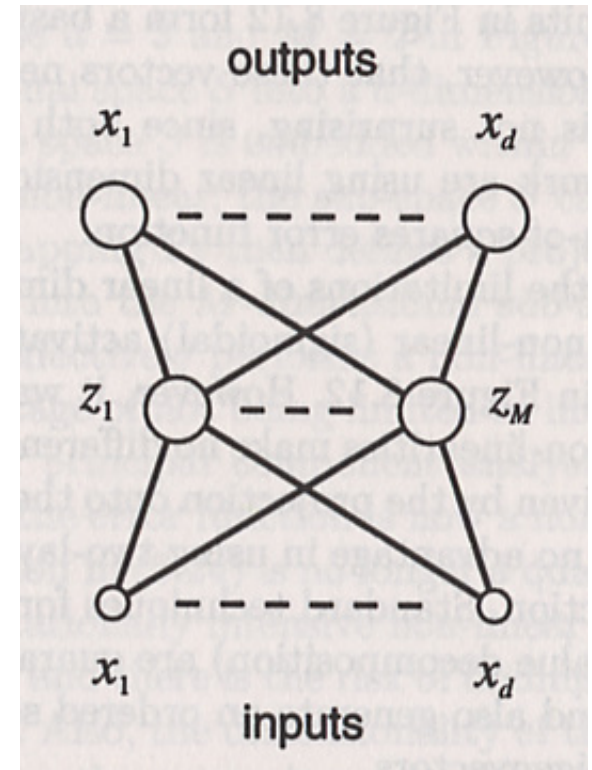
ニューラルネットによる次元圧縮 (1)

自己想起型の
ニューラルネットで学習



誤差関数:

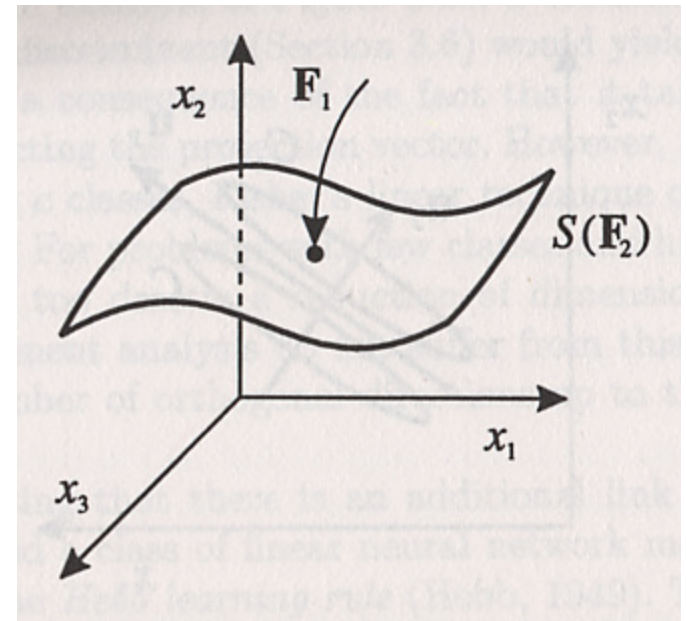
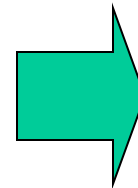
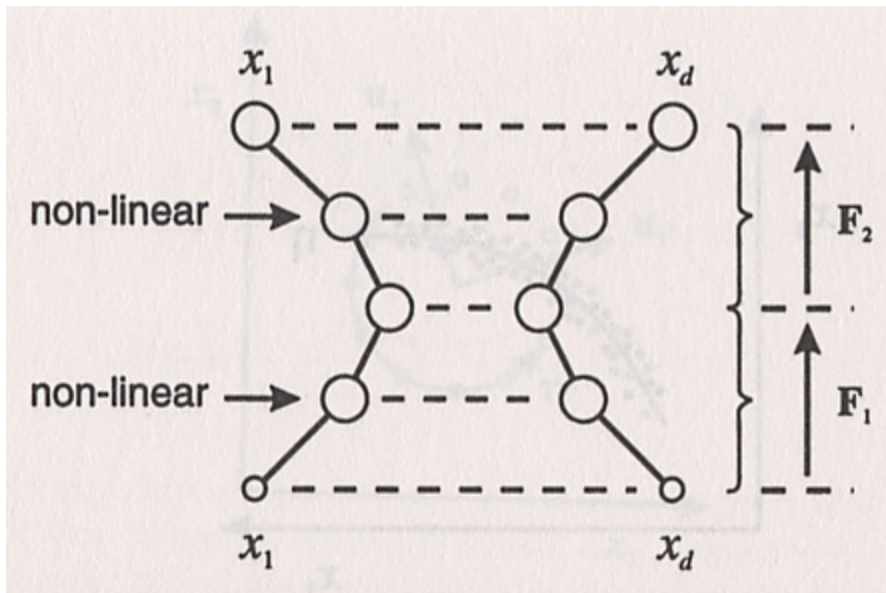
$$E = \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^d \{y_k(\mathbf{x}_n) - x_{n,k}\}^2$$



隠れユニットの $g(a)$ が *linear* か *sigmoidal* かに関わらず, 主成分分析から決定される M 次元部分空間と一致

ニューラルネットによる次元圧縮 (2)

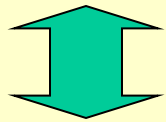
sigmoidal ユニットを持つ隠れ層を追加して、非線形の次元圧縮を実現



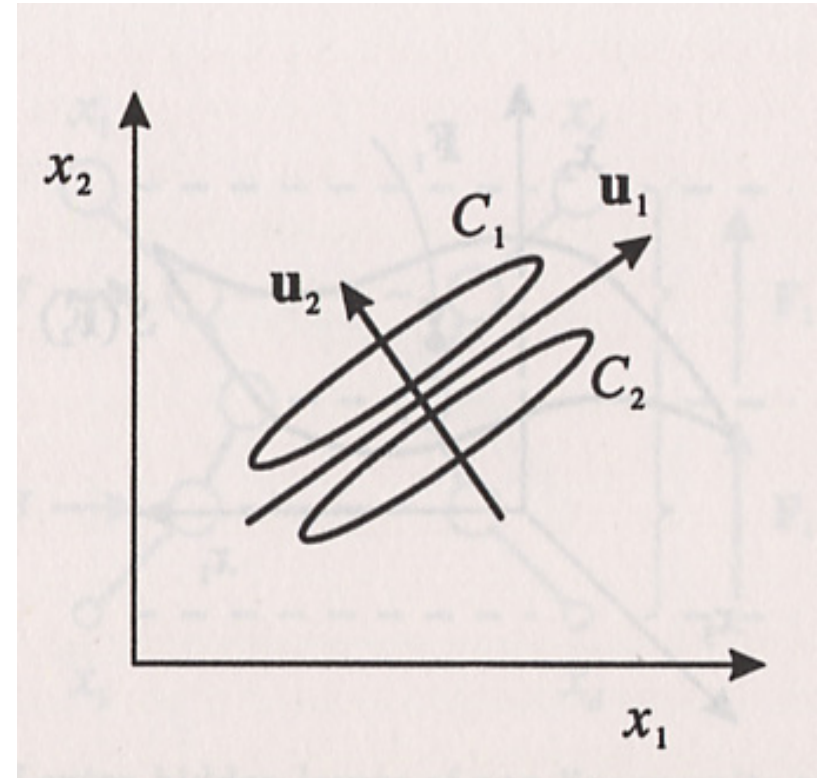
部分空間 S は超平面でなく、超曲面となる！

教師なし次元圧縮の限界

右図に対して
主成分分析で得られる
 u_1 は分類能力なし



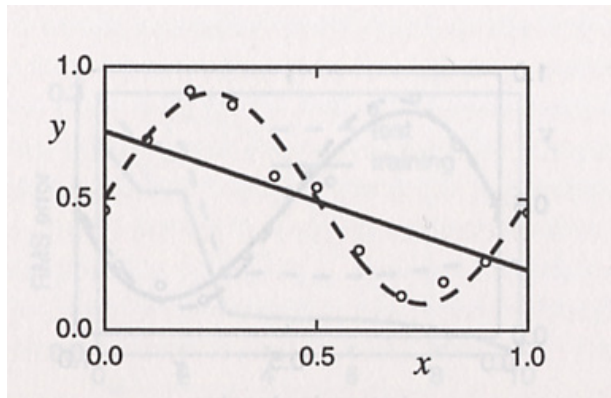
識別を考慮した Fisher's
linear discriminant は
 u_2 を与える



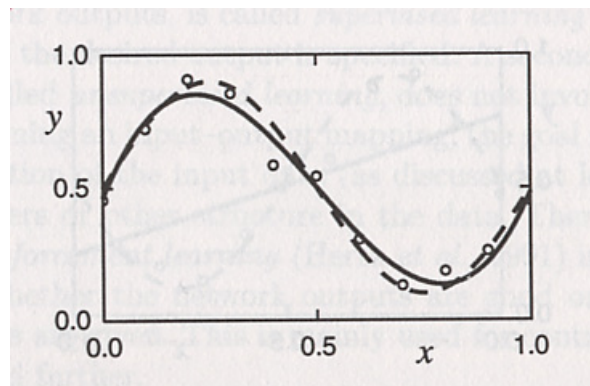
ただし, PCA は任意次元 $M < d$ への圧縮可能
一方, Fisher's technique は $c - 1$ 次元の選択のみ

学習と汎化の問題－回帰

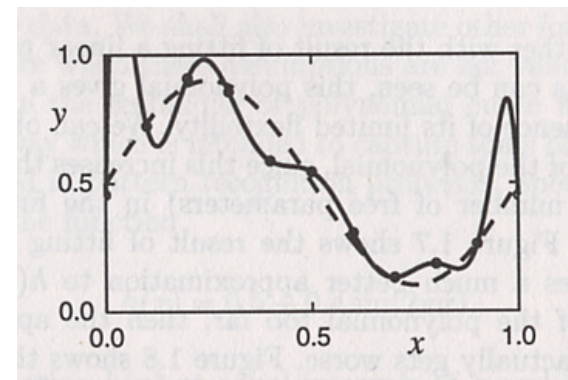
多項式による回帰: $y(x) = w_0 + w_1x + \dots + w_Mx^M$



$M = 1$

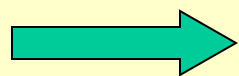


$M = 3$



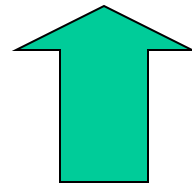
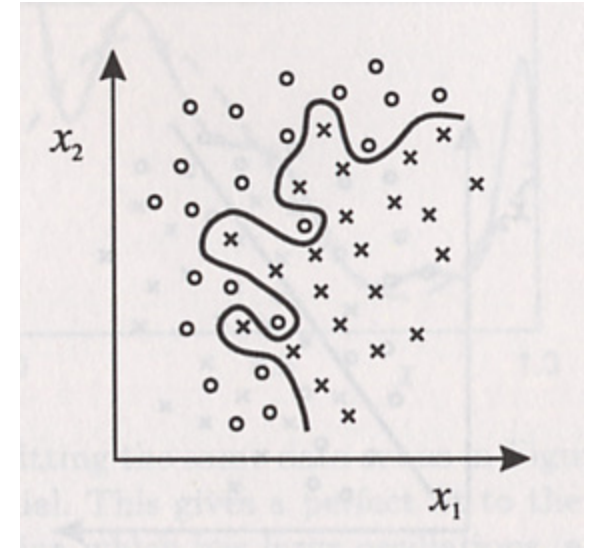
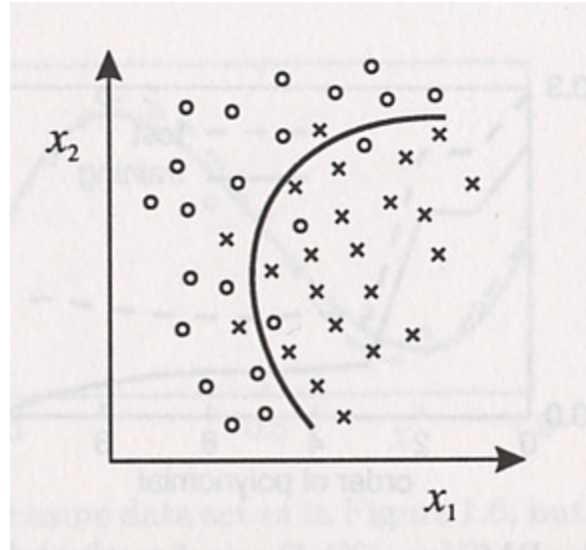
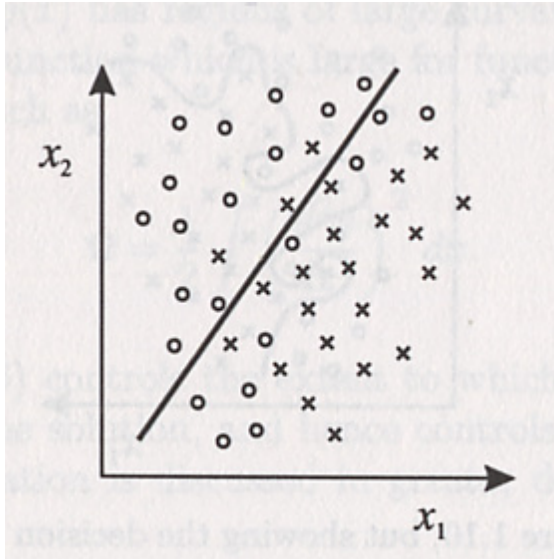
$M = 10$

汎化とは、学習データの正確な表現を求めることでなく、データの発生過程の統計的モデルを構築すること



モデルの複雑さはどう決定したらよいのか

学習と汎化の問題－分類



Occam's razor – William of Occam (1285–1349)

[データとの食い違い] + [モデルの複雑さ] → 最小

正則化と汎化能力

汎化能力: 学習データへの over-fitting でなく
未知データに対する予測能力

正則化:

$$\tilde{E} = E + \nu \Omega$$

Ω : 正則化のペナルティ関数
→ 写像の滑らかさを評価

Ω の例:

$$\Omega = \frac{1}{2} \sum_i w_i^2$$

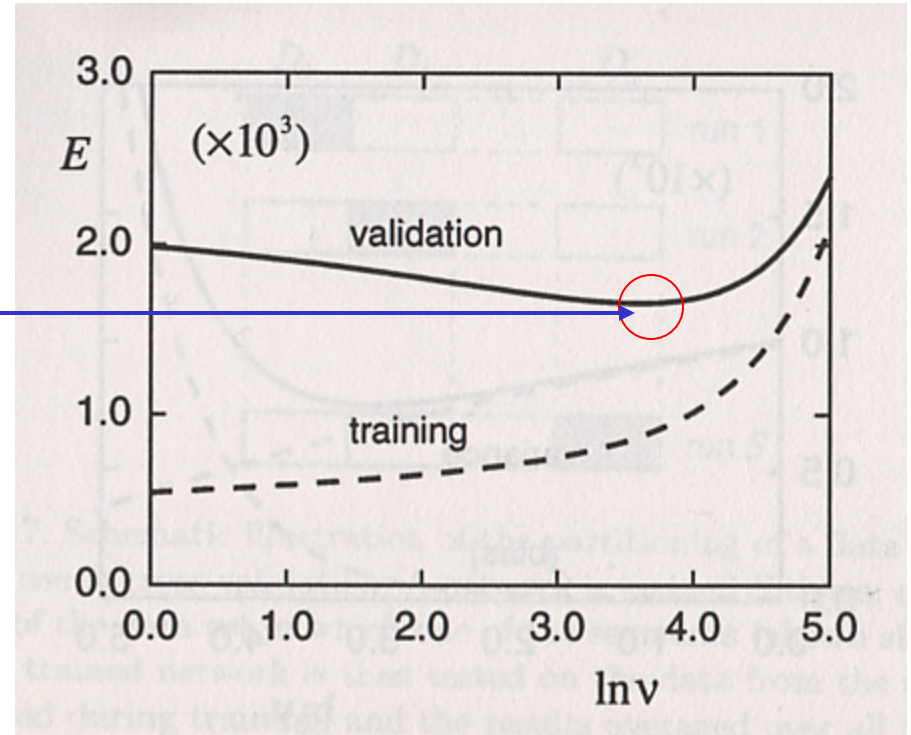
$$\Omega = \frac{1}{2} \sum_{n=1}^N \sum_{i=1}^d \sum_{k=1}^c \left(\frac{\partial^2 y_k^n}{\partial x_i^2} \right)^2$$

汎化能力の評価

正則化

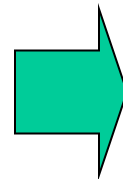
$$\tilde{E} = E + \nu\Omega$$

ν の選択



validation set の作成:

- [1] hold-out method
学習と評価に2分割
- [2] cross-validation
 S 分割して循環で評価
- [3] leave-one-out method



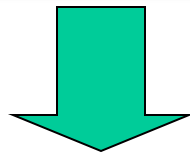
最後に独立なデータ
test set で評価

モデルの複雑さと汎化能力

validation set を用いずに汎化能力を評価する
→ モデルの複雑さそのものを評価

予測誤差:

$$PE = \text{training error} + \text{complexity term}$$



$$PE = \frac{2E}{N} + \frac{2W}{N} \sigma^2$$

E : 2乗和エラー

N : 学習データ数

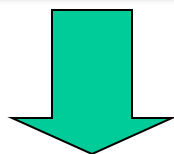
W : モデルのパラメータ数

σ^2 : データノイズの分散

最後に

パターン認識の面白さ:

人間と同じように概念を獲得し、パターンを認識できる能力を計算機で実現できる！



なかなか容易ではない

- [1] 生成モデル vs. 識別モデル
- [2] 学習データの有限性
- [3] 特徴抽出の問題
- [4] 教師なし学習の問題

→ 実用的/理論的に *most challenging*