情報システム工学科
伊達 章

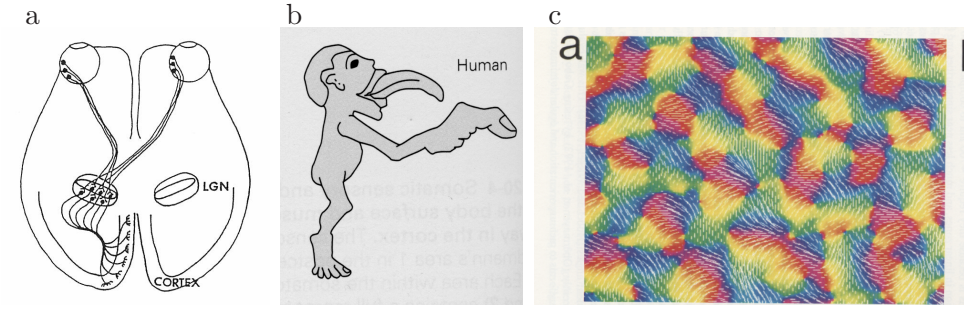
自己組織化モデル

- 基本事項：

  ・自己組織化マップ（SOM, Self-organizing Map）とは

  1. 大脳感覚野の機能地図が学習により形成されていく様子を説明する自己組織化モデル
     (a) ランダムな結合から，トポグラフィックマップが形成される（図 a）.
     (b) 使用頻度の高い領域ほど解像度がよくなる（図 b）.
     (c) 一部が壊れても，他の部分が代わりに働くようになる.
     (d) 高次元信号をうまく低次元に落として表現できる（図 c）.
  2. 教師なし学習．数学的には非線形の主成分分析（大学院生向けの話題）
  3. 興奮パターンのダイナミックスと結合係数のダイナミックスを融合したモデル．von der Malsburg が 1973 年に提案したモデルを Kohonen が単純化.
  4. やや複雑なモデルを簡単化したため，通常の神経回路モデルとは異なる.
  5. 動作アルゴリズムは単純．したがって，計算機シミュレーションも簡単.
  6. ただし，どうしてこれでうまくいくか理解することは簡単ではない.
  7. 工学的な応用範囲がとても広い

a b c



- 用語
  - 入力信号空間： 文字通り，入力信号 $\boldsymbol{x}$ の空間 (今日の話は 1 次元)
  - 参照ベクトル： 素子の結合係数と思って差し支えない．入力信号空間中を参照するという意味で参照ベクトルという.
  - 競合と協調．各入力 $\boldsymbol{x}$ に対し，出力層の 1 個の素子（勝者と呼ぶ）のみが興奮.
  - 近傍学習： となりの素子と協調して進む学習のこと．別名，おすそわけ学習とも言う.

- SOM の学習アルゴリズム

  高次元のデータを要約して，最も重要な 2 つの変数だけを取り出したい場合，2 次元に配列された素子からなる SOM を用いる．配列が 2 次元であることは本質的ではなく，必要に応じて 1 次元や 3 次元の配列を考えればよい．以下では 1 次元の SOM を利用する際の学習アルゴリズムを説明する．各素子は参照ベクトルと呼ばれる n 次元ベクトルをもつ．i 番目の素子の参照ベクトルを $\boldsymbol{m}_i$（$i = 1, \cdots, n$）とする．$\boldsymbol{m}_i$ の次元は SOM に与えられる入力信号 $\boldsymbol{x}$ の次元と同一である．入力信号空間が k 次元の場合，$\boldsymbol{m}_i = (m_{i1}, m_{i2}, \cdots, m_{ik})$ と書ける．以下に学習アルゴリズムを示す.

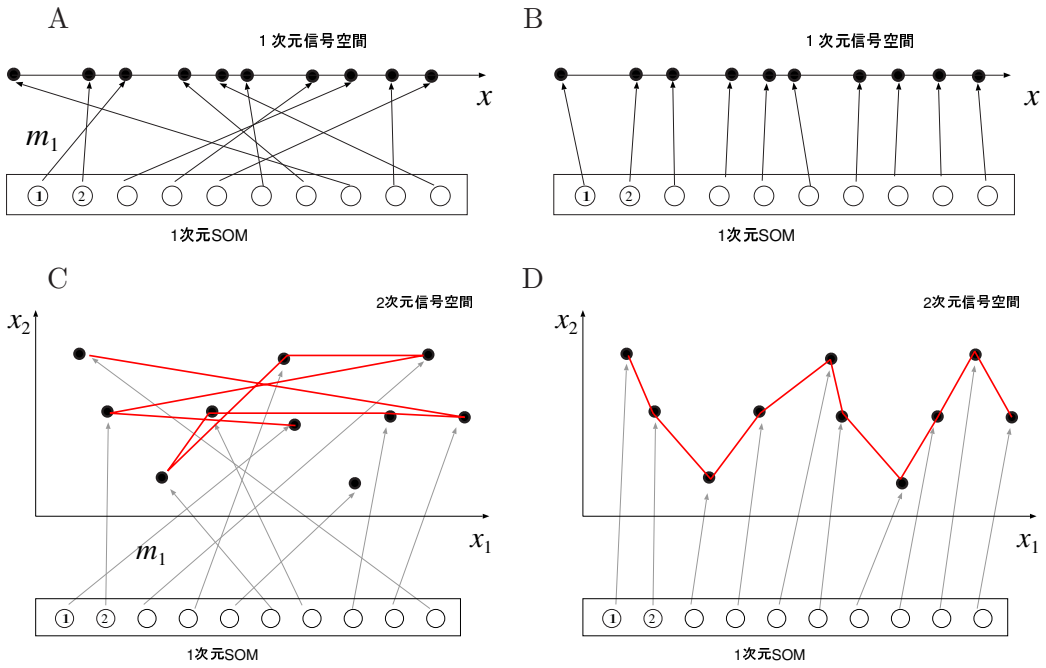


図 1: 1 次元の信号空間に 1 次元 SOM を適用する様子（A,B），および 2 次元の信号空間に 1 次元 SOM を適用する様子（C,D）．学習前（A,C）と学習後（B,D）．各素子の参照ベクトルは入力信号空間を参照する（各素子から信号空間に向けて矢印が書かれている）．参照ベクトルは，入力信号空間中において各素子が担当する領域の中心を指している．

1. 初期設定： 各参照ベクトル $\boldsymbol{m}_i$ に適当な乱数を割り当てる．
2. 外界の確率的構造（例えば一様分布）にしたがい入力信号 $\boldsymbol{x}$ を生成し，モデルに与える．
3. すべての素子の中で入力 $\boldsymbol{x}$ に最も近い参照ベクトルをもつ素子（以後，勝者と呼ぶ）

$$c = \underset{i}{\operatorname{argmin}} ||\boldsymbol{m}_i - \boldsymbol{x}|| \tag{1}$$

を求める（興奮パターンのダイナミクスに対応）．入力 $\boldsymbol{x}$ に対し，1 個の素子のみが興奮．

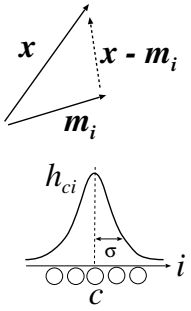
4. 以下の式にしたがい参照ベクトル $\boldsymbol{m}_i, i = 1, \cdots, n$ の学習をおこなう（下図）．

$$\Delta \boldsymbol{m}_i = \alpha h_{ci}(\boldsymbol{x} - \boldsymbol{m}_i) \tag{2}$$

$$h_{ci} = \exp -\frac{|c-i|^2}{2\sigma^2} \tag{3}$$

ここで h_{ci} は 素子の配列状で i 番目の素子と勝者である c 番目の素子との間の距離に依存して決まる関数，α は学習の強さを表す正の定数（結合係数のダイナミクスに対応）．

5. 2. から 4. を繰り返す．



学習は勝者の周囲でおこる．これを近傍学習といい h_{ci} を近傍関数という．上式で与えた近傍関数はガウス関数を使っていて，σ は近傍の広がりを決める正の定数である．式 (1) のノルムは，入力信号空間内の ユークリッド距離，式 (3) では 素子配列上の ユークリッド距離を計算しており対象が異なっていることに注意したい．2 次元や 3 次元に配列された SOM を用いる場合は，式 (3) がその配列の空間（2 次元もしくは 3 次元）での素子間の距離を計算することになる．