

# 確率伝播法と条件付確率表の自 己組織化を行う大脳皮質モデル (ニューロコンピューティング研究会)

産業技術総合研究所

一杉裕志

2006-10-11

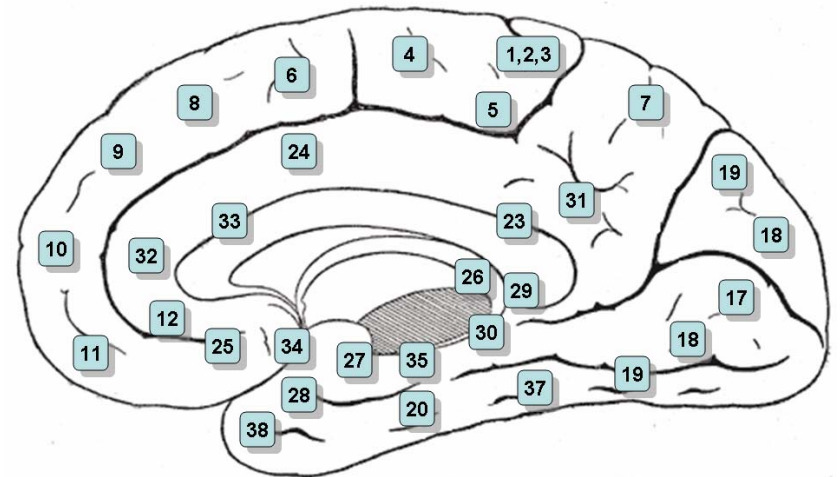
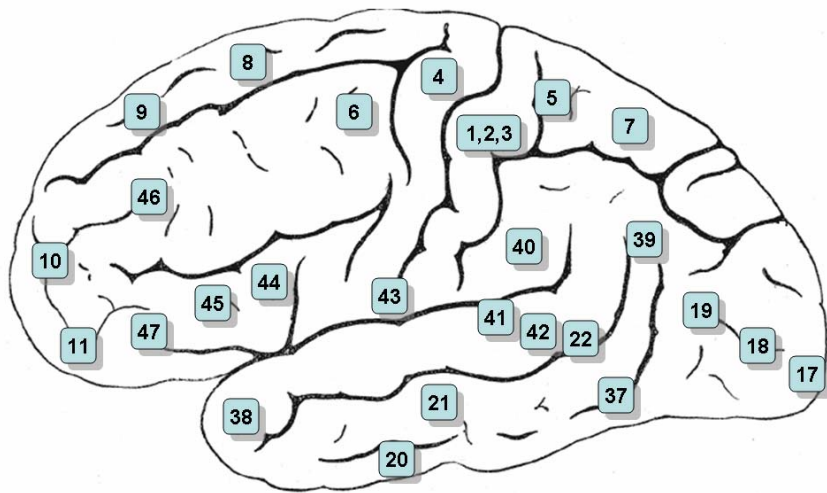
# 概要

- ベイジアンネットの条件付確率表を自己組織化マップで獲得し、確率伝播法を近似計算する神経回路モデル(BESOM モデル)について述べる。
- このモデルは大脳皮質の解剖学的構造とよく対応する。
- 今後このモデルを拡張していき、大脳皮質の主要な機能の計算機上での再現を目指す。
- (シミュレーションは計画中。)

背景

# ブロードマンの領野

- 大脳皮質の約50の領野
- 機能局在
- 下位領野と上位領野の双方向の結合



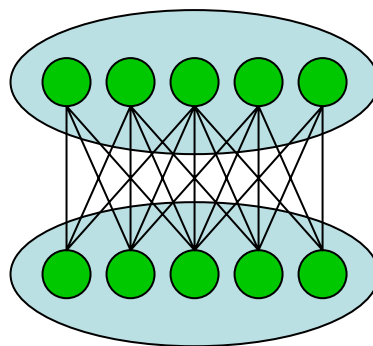
# 現状における大脳皮質の理解

- 大脳皮質の各領野の役割や、領野間接続の解剖学的構造はかなり分かってきてはいるが、機能を計算機上で再現できるほどは分かっていない。
- 自己組織化マップ(SOM)やベイジアンネットを用いた大脳皮質のモデルはあるが、両方の特徴を統合したものはない。

# 自己組織化マップ(SOM)とは

- 競合学習・近傍学習を特徴とする教師なし学習アルゴリズム。
- 高次元の特徴ベクトルを、低次元に圧縮。
- 視覚野の方位選択カラムのモデルを工学的に扱いやすくしたもの。

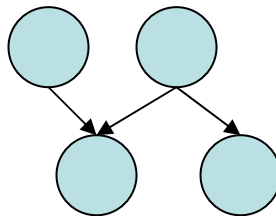
競合層



入力層

# ベイジアンネットとは

- 対象を、因果関係のある確率変数のグラフ構造として表現するモデル化手法。
- 一部の確率変数の観測値が与えられると、条件付確率表を使って、他の確率変数の値をベイズ推定できる。
- 確率伝播アルゴリズムという効率的な計算方法が使われる。
- ノイズに強いパターン認識。リアルタイム性。

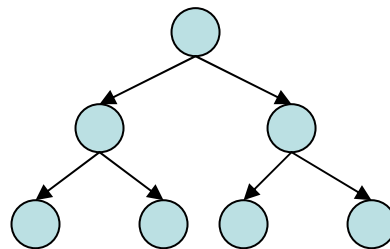


# SOMとベイジアンネットを融合し た大脳皮質のモデル

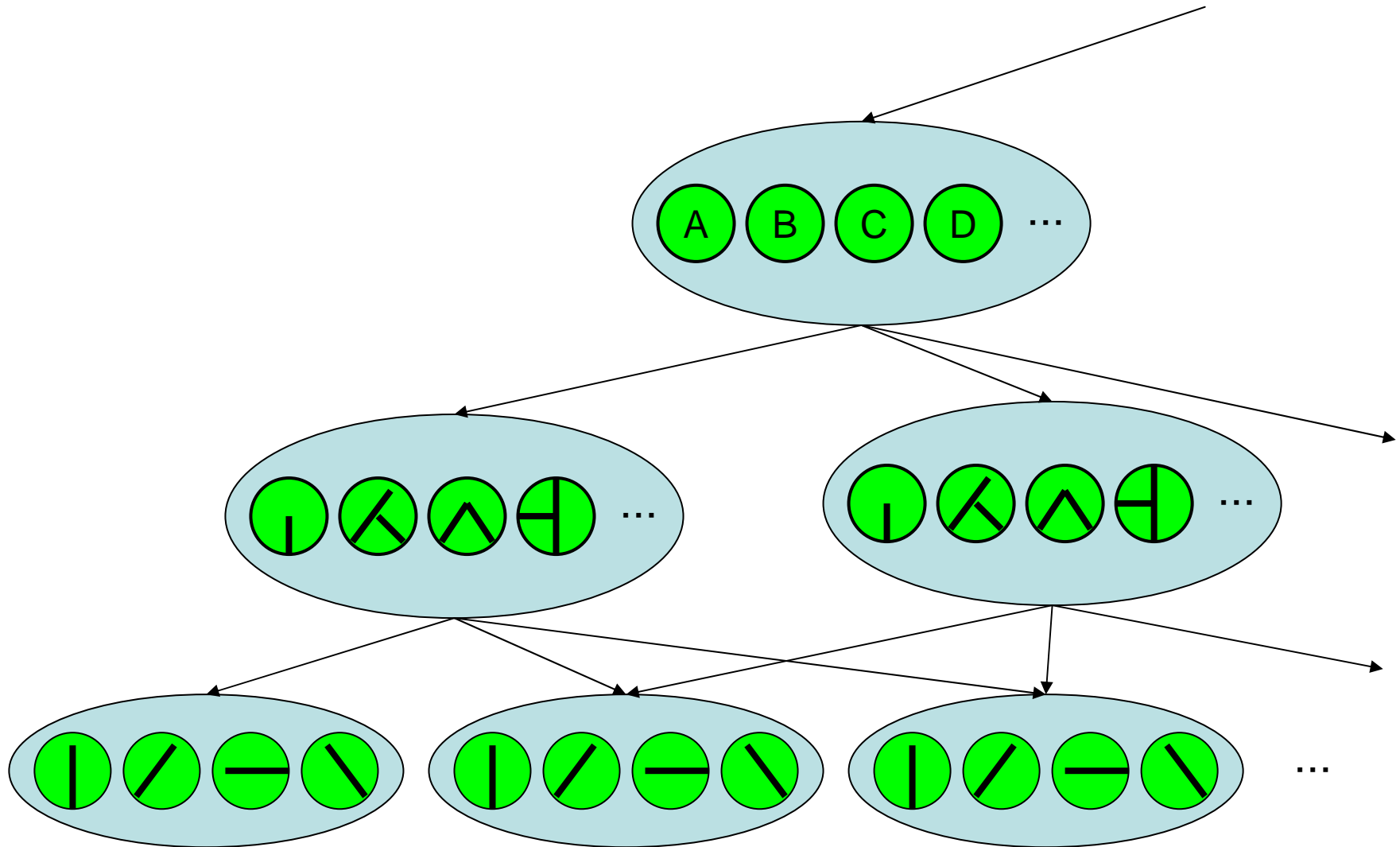


# BESOM の概要

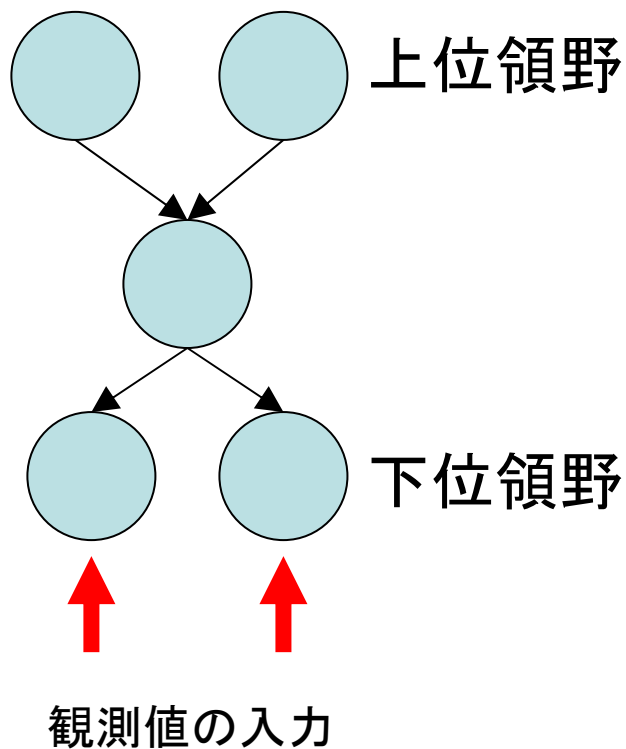
- 複数のSOMを双方向に結合したネットワーク(Bidirectional SOM)。ベイジアンネットの各ノードをSOMにしたものとも言える。
- SOMは、子ノードからの入力の組をクラスタリングする。学習結果が条件付確率表を表す。
- 観測値は最下端のノードのみ。
- 上にいくほど抽象度の高い情報を表現。



# BESOMによる階層的なパターン認識



# BESOM におけるSOMとベイジアンネットワークの構成要素の対応



脳	SOM	ベイジアンネットワーク
下位領野	入力層	子ノード
ハイパー コラム	マップ (競合層)	ノード(確 率変数)
V層錐体 細胞?	特徴、 ユニット	確率変数 の値
発火率	類似度	確率
シナプス 結合度	結合度	条件付確 率

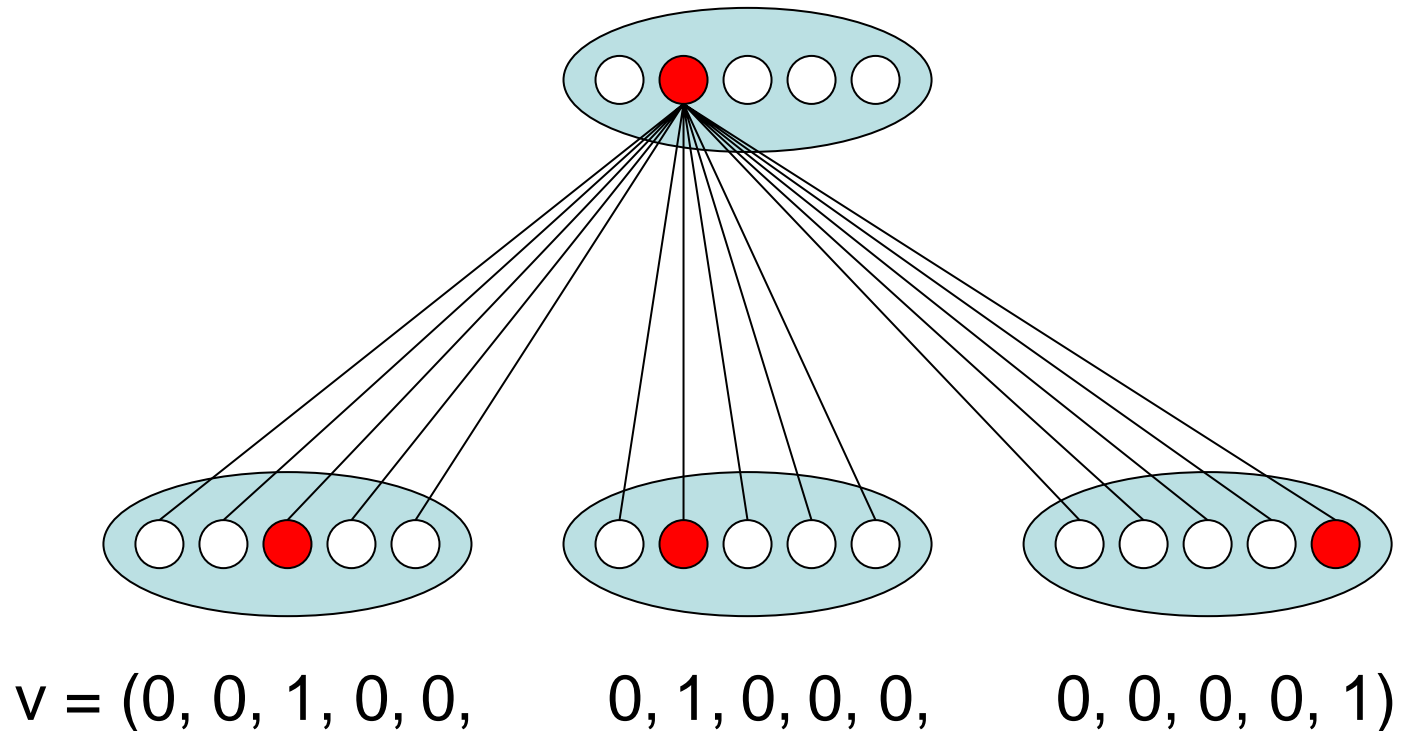
# 認識ステップと学習ステップ

- 次の2つのステップを交互に繰り返す
  - 認識ステップ: 現在の条件付確率表に基づき、近似確率伝播法で各ノードの値をベイズ推定する。
  - 学習ステップ: 各ノードの値のMAP推定結果を用いて条件付確率表を更新する。

# BESOM の学習ステップ

# 学習時の入力ベクトルの例

- 各子ノードの状態をMAP推定した結果を入力ベクトルとして受け取る。



# 勝者ユニットの学習則

- 結合ベクトルを入力ベクトルに近づける

$$w_{ij} = (1 - \alpha)w_{ij} + \alpha v_j$$

- 以下の条件を満たせば、 $w_{ij}$  は条件付き確率  $P(Y=y_j|X=x_i)$  になる。
  - 近傍半径が十分小さく無視できる。
  - 学習率  $\alpha$  が  $n$  回目の入力ベクトルに対し  $1/n$  となるようスケジューリングされている。

# BESOMの認識ステップ



# 確率伝播アルゴリズム [Pearl 1988]

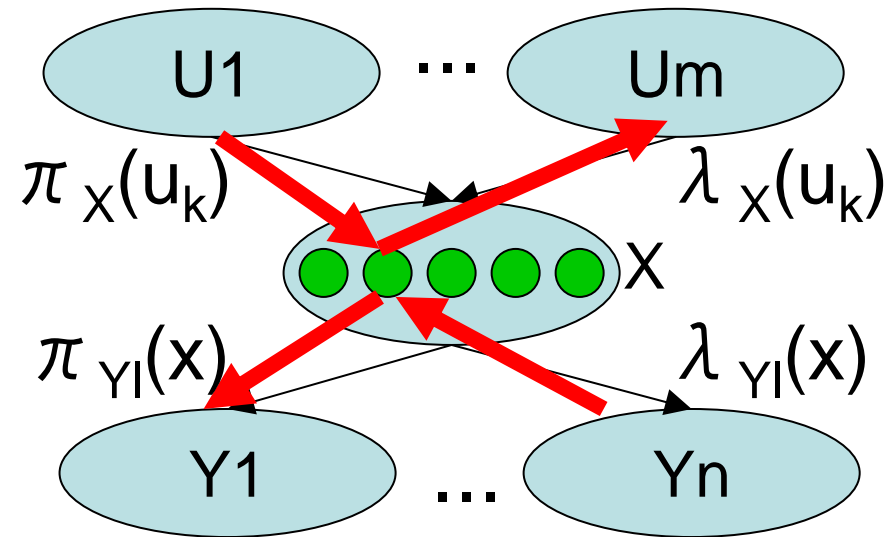
$$BEL(x) = \alpha \lambda(x) \pi(x)$$

$$\pi(x) = \sum_{u_1, \dots, u_m} P(x | u_1, \dots, u_m) \prod_k \pi_X(u_k)$$

$$\lambda(x) = \prod_l \lambda_{Y_l}(x)$$

$$\pi_{Y_l}(x) = \pi(x) \prod_{j \neq l} \lambda_{Y_j}(x)$$

$$\lambda_X(u_k) = \sum_x \lambda(x) \sum_{u_1, \dots, u_m / u_k} P(x | u_1, \dots, u_m) \prod_{i \neq k} \pi_X(u_i)$$



# 近似確率伝播アルゴリズムの導出

- 目的:

- 条件付確率表  $P(X|U_1, \dots, U_m)$  は親ノードの数  $m$  に対し指数的にサイズが増大してしまう。これを  $P(X|U_i)$  で近似したい。
- 神経回路網で実現可能なほどに計算を簡単にしたい。

- 方針:

- 条件付確率表の性質に着目。

- (Cf. noisy-OR [Pearl 1988]) 
$$P(X | U_1, \dots, U_m) \approx \sum_{i=1}^m P(X | U_i)$$

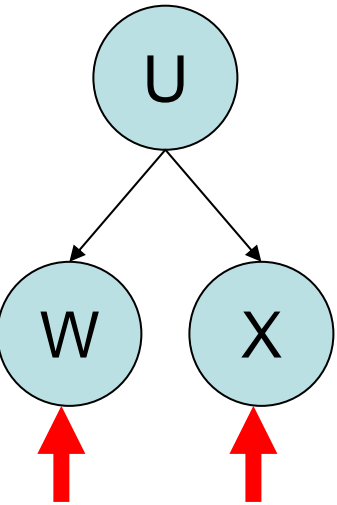
- $\pi$ 、 $\lambda$  メッセージに「送信相手からの情報を除外する」ことをやめる。

- 正規化  $\sum \pi_{Y_i}(x) = 1$

# $P(X|U_1, U_2, \dots, U_m)$ の近似

- 前提：
  - ノードがSOMであり入力の組を分類。
  - 最下端のノードのみが観測値。
  - SOMのユニット数は十分に多く、入力パターンをすべて分類しきれる。

# SOMが獲得する結合表の例

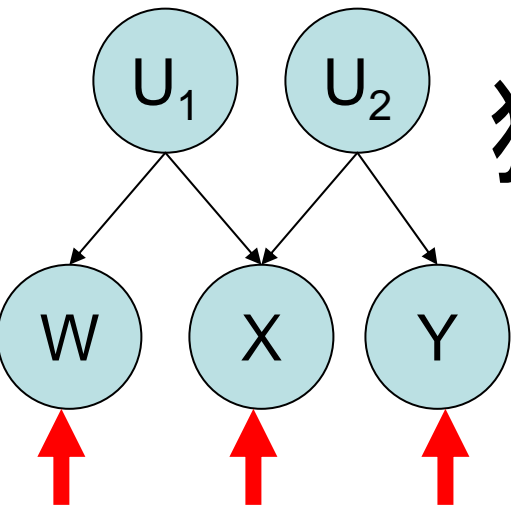


- UのSOMは特徴W、Xの組を重複のないように分類する。
- Uのあるユニットが勝者になったということは、高い確率で必要な特徴が入力されたことを意味する。

U	W	X
u1	w1	x1
u2	w1	x2
u3	w1	x3
u4	w1	x4
u5	w2	x1
u6	w2	x2
u7	w2	x3
u8	w2	x4
...		

$$P(W=w2|U=u5) = P(X=x1|U=u5) \doteq 1$$

$$P(W \neq w2|U=u5) = P(X \neq x1|U=u5) \doteq 0$$



# 獲得される条件付確率表

- 親ノードの少なくとも1つが特徴  $x$  を必要とするとき、 $x$  が入力されたと推論すべき。

$$P(X = x_1 | U_1, U_2)$$

		U <sub>2</sub>													
U <sub>1</sub>					1			1							
					1			1							
		1	1	1	1	1	1	1	1	1	1	1	1	1	1
					1			1							
					1			1							
					1			1							
		1	1	1	1	1	1	1	1	1	1	1	1	1	1
					1			1							
					1			1							
					1			1							

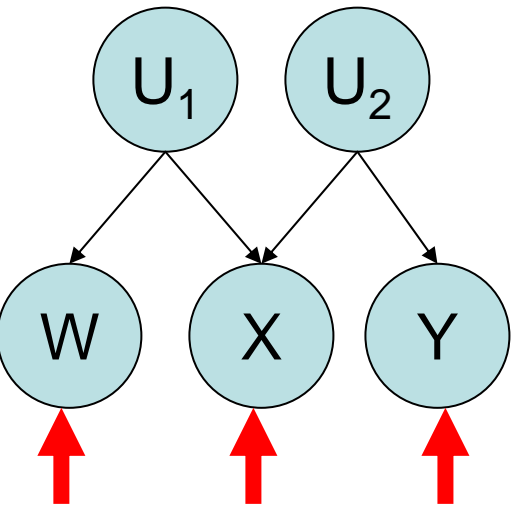
$$P(X = x_2 | U_1, U_2)$$

		U <sub>2</sub>													
U <sub>1</sub>												1			1
		1	1	1	1	1	1	1	1	1	1	1	1	1	1
												1			1
												1			1
												1			1
												1			1
		1	1	1	1	1	1	1	1	1	1	1	1	1	1
												1			1
												1			1
												1			1

...

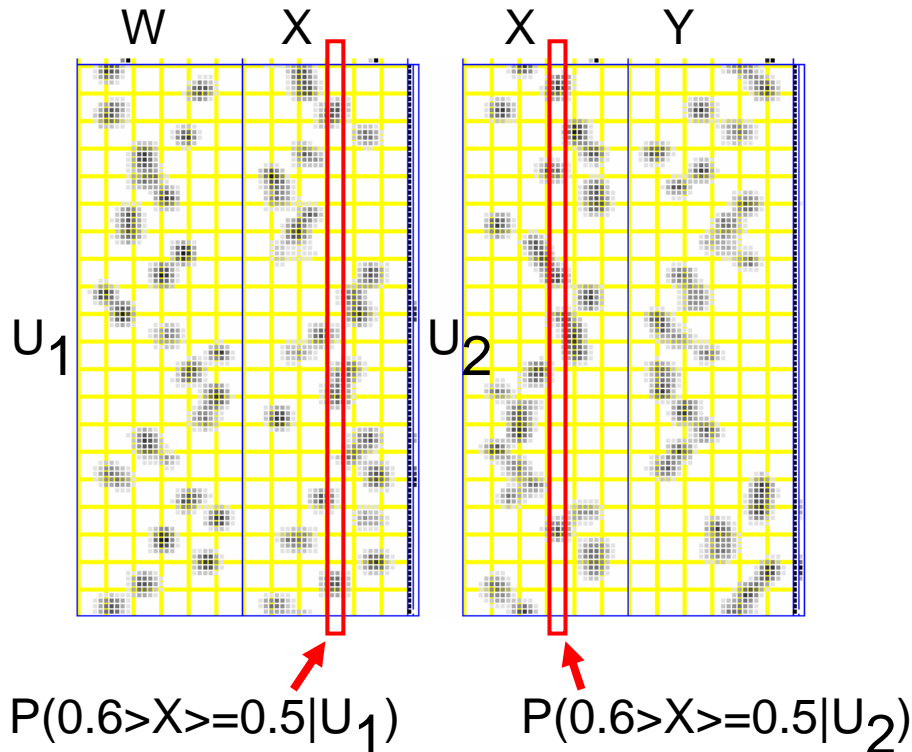
$$P(X | U_1, U_2) \approx P(X | U_1) + P(X | U_2)$$

# 実験結果

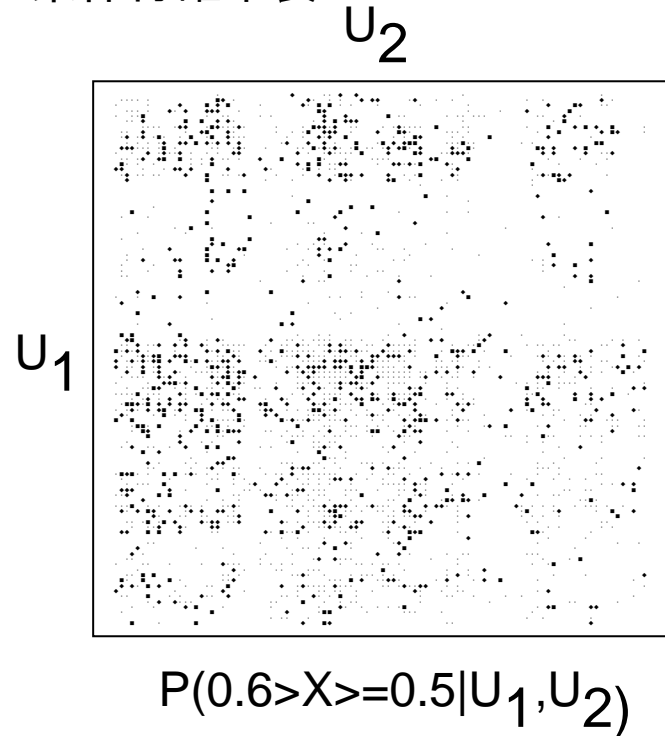


連続値  $[0,1)$  をメンバーシップ関数のようにコーディングして入力

3万ステップ目における2つのSOMの結合表

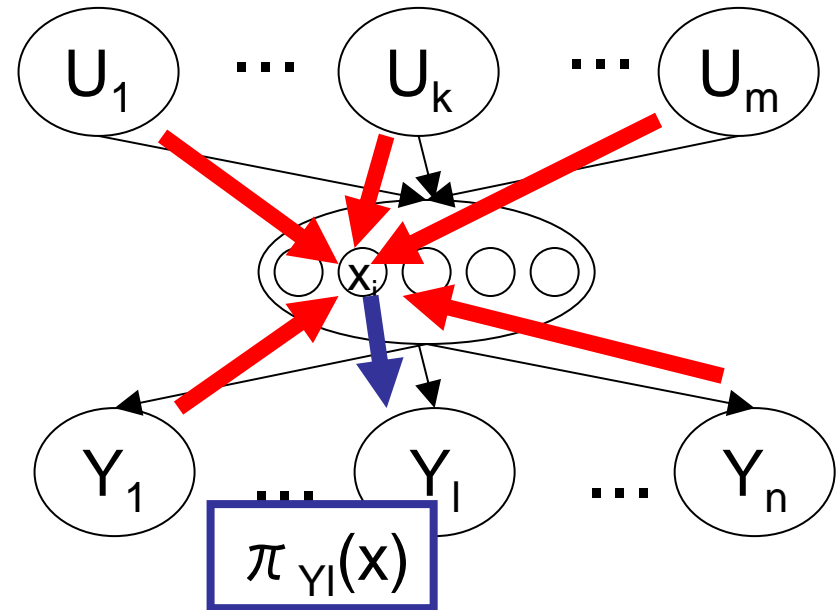
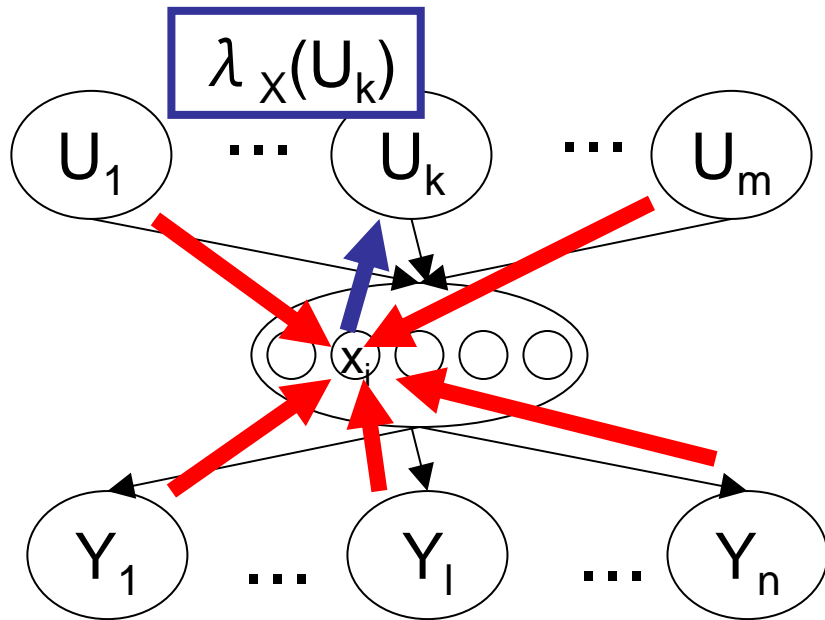


0から3万ステップまでの統計による  
条件付確率表



# メッセージ送信相手からの情報

- 送信相手からの情報を除外している。

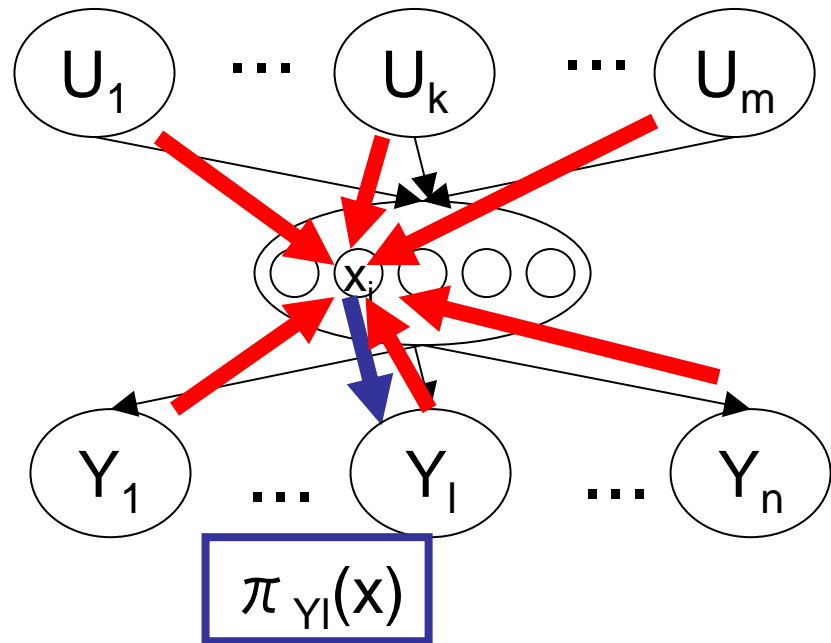


- 近似アルゴリズムでは、親ノード、子ノードの数が十分に多いと仮定し、除外するのをやめる。

# 例： $\pi_{Y_l}(x)$ の近似

- ノード  $Y_l$  に送るメッセージ  $\pi_{Y_l}(x)$  に、 $Y_l$  からの情報  $\lambda_{Y_l}(x)$  も含める。

$$\begin{aligned}\pi_{Y_l}(x) &= \pi(x) \prod_{j \neq l} \lambda_{Y_j}(x) \\ &\approx \pi(x) \prod_j \lambda_{Y_j}(x) \\ &= \lambda(x) \pi(x)\end{aligned}$$





# 近似確率伝播アルゴリズム

$$\lambda_{Y_l}^{t+1}(x) = Z_{Y_l}^t + \sum_{y_l} \lambda^t(y_l) P(y_l | x)$$

$$\lambda^{t+1}(x) = \prod_{l=1}^n \lambda_{Y_l}^t(x)$$

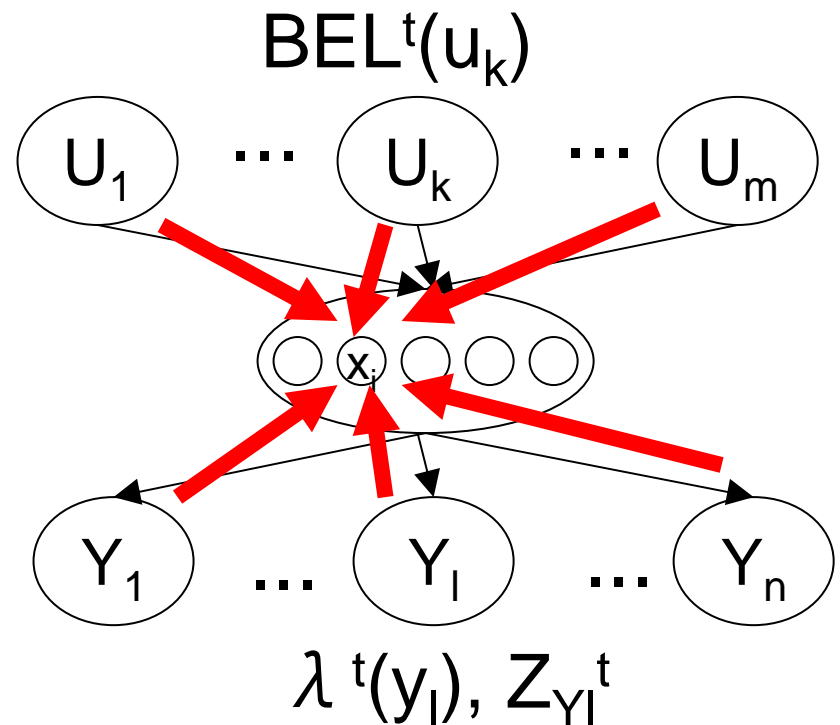
$$\kappa_{U_k}^{t+1}(x) = \sum_{u_k} P(x | u_k) BEL^t(u_k)$$

$$\pi^{t+1}(x) = \sum_{k=1}^m \kappa_{U_k}^{t+1}(x)$$

$$p^{t+1}(x) = \lambda^{t+1}(x) \pi^{t+1}(x)$$

$$Z_X^{t+1} = \sum_x p^{t+1}(x)$$

$$BEL^{t+1}(x) = p^{t+1}(x) / Z_X^{t+1}$$



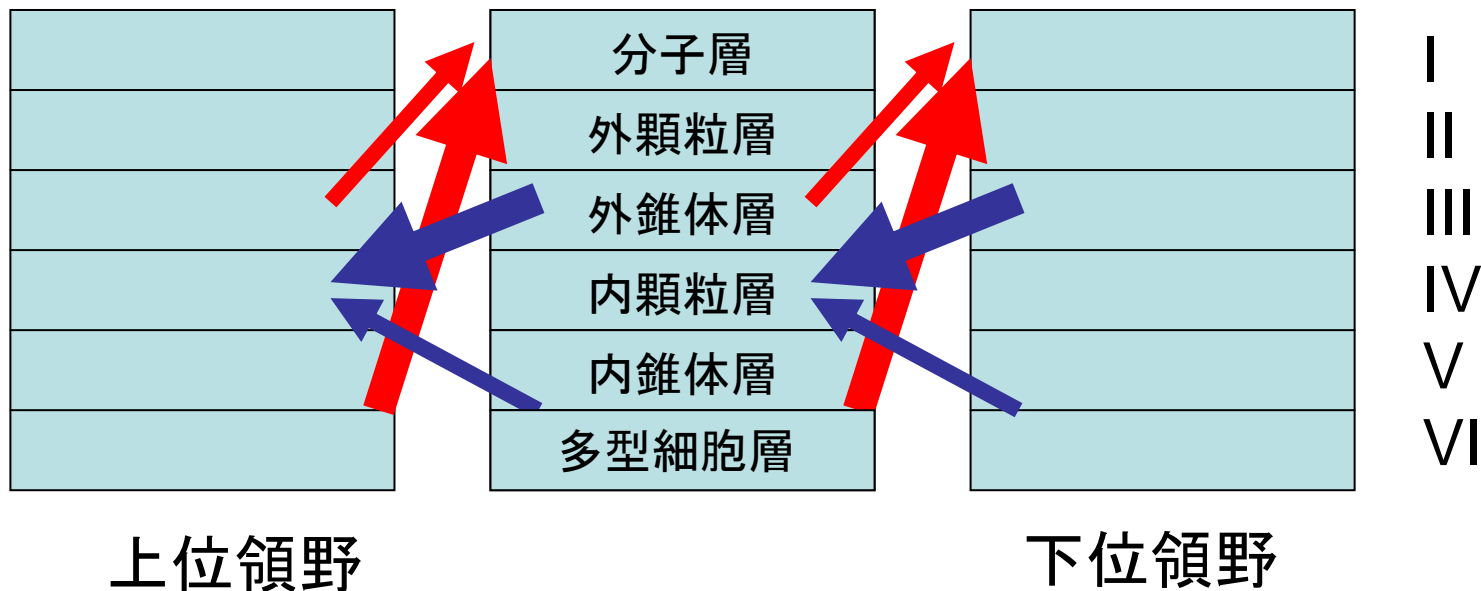
# 記憶量、計算量

- 学習時の入力ベクトルはスパースなので条件付確率表もスパースになる。それを利用して、記憶量・計算量のオーダーを下げることができる。
- 条件付確率表の表現に必要な記憶量は、ノード内のユニット数を  $s$  とすると  $O(s)$
- 計算量は並列計算の場合  $O(\log s)$
- 記憶量・計算量の観点からは大脳皮質のモデルの候補として問題ない。

# 大脳皮質の6層構造との対応

# 解剖学的知見 [Pandya and Yeterian 1985]

- 前方への結合は主にIII層からIV層、後方への結合は主にV・VI層からI層へ。
- V層からIV層への前方への結合もある。
- III層からI層への後方への結合も少しある。

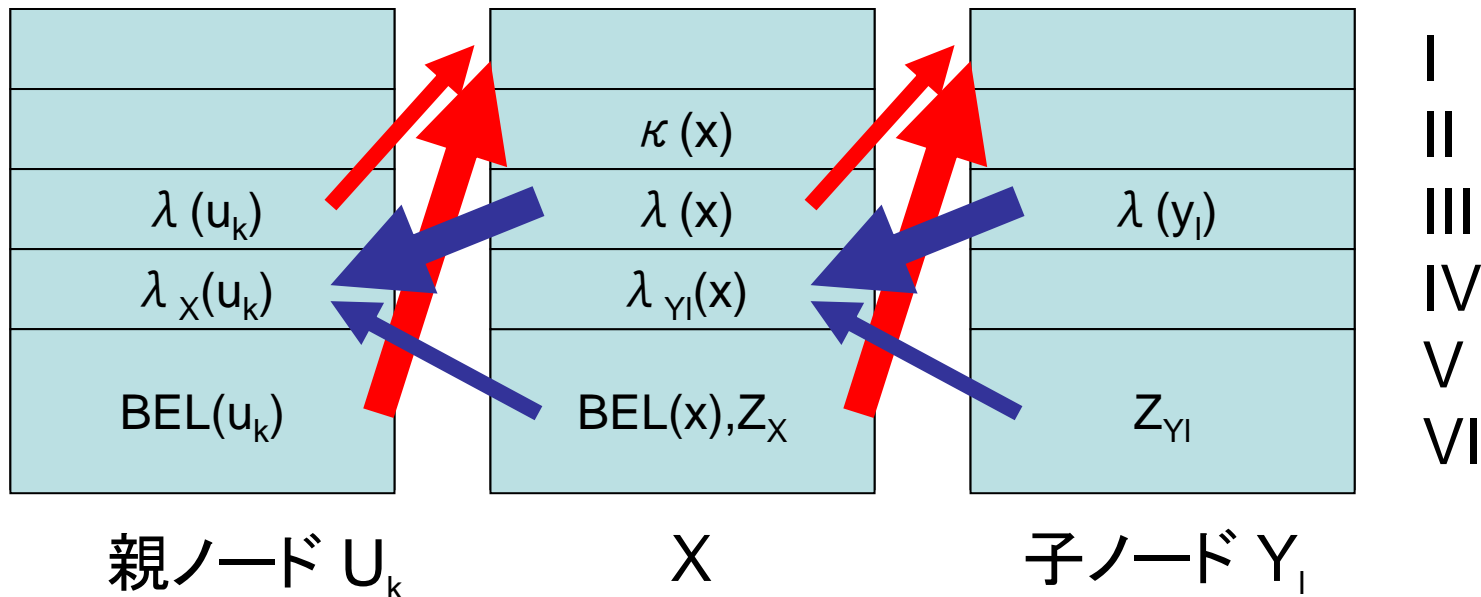


# 領野間結合の不思議な点

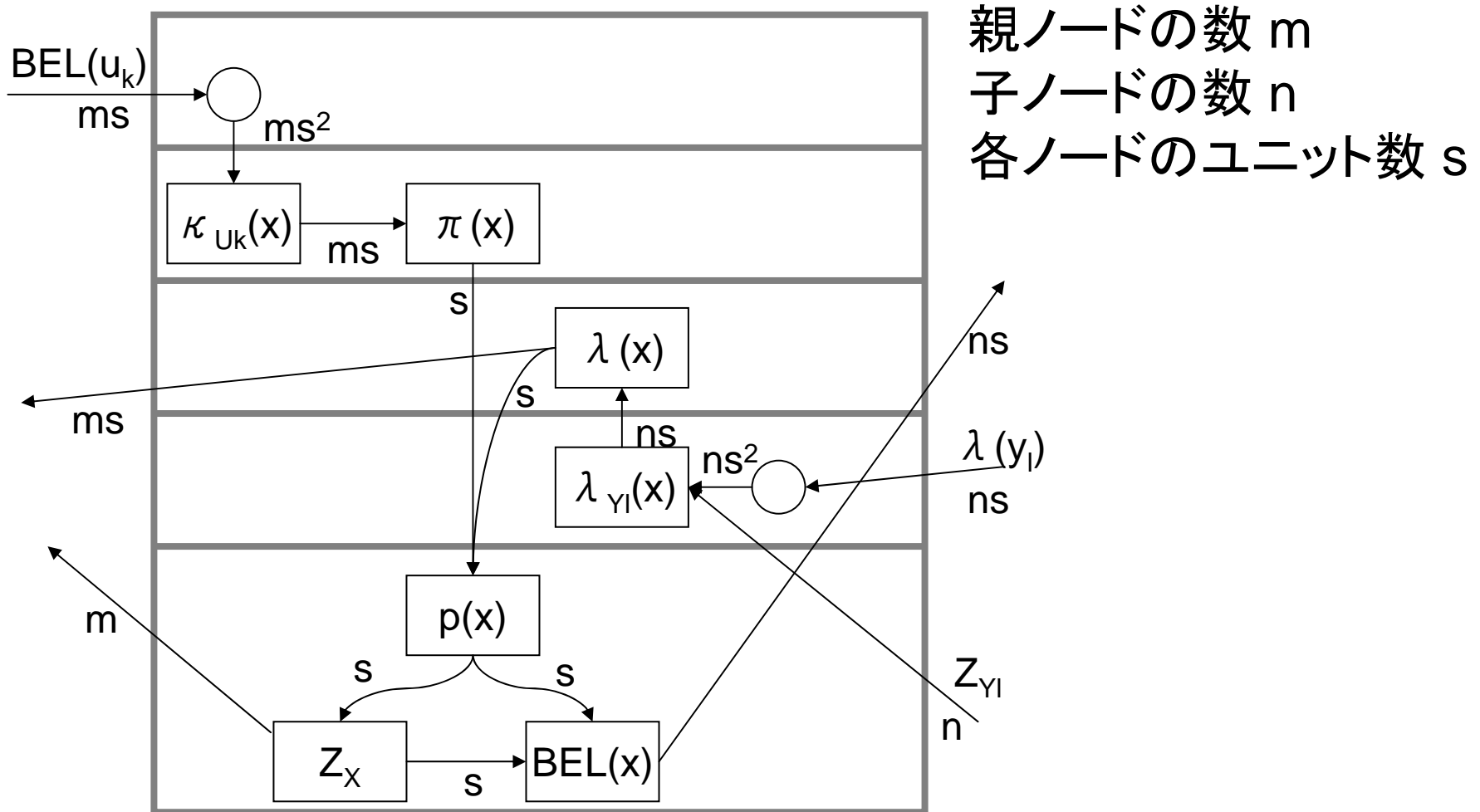
- カラム内では主にIV層→II・III層→V層と情報処理が進むと言われている。実際、運動野では、V層から脊髄に出力される。
- にもかかわらず、情報処理の最終結果ではない、途中結果のIII層の情報が上位領野に送られ、最終結果のV層の情報は下位領野に戻されるという、**とても不思議な構造**をしている。

# 6層への変数の割り当て

- 素直に対応が付き、偶然とは考えられない。
  - (III層からI層への後方への結合は謎。)

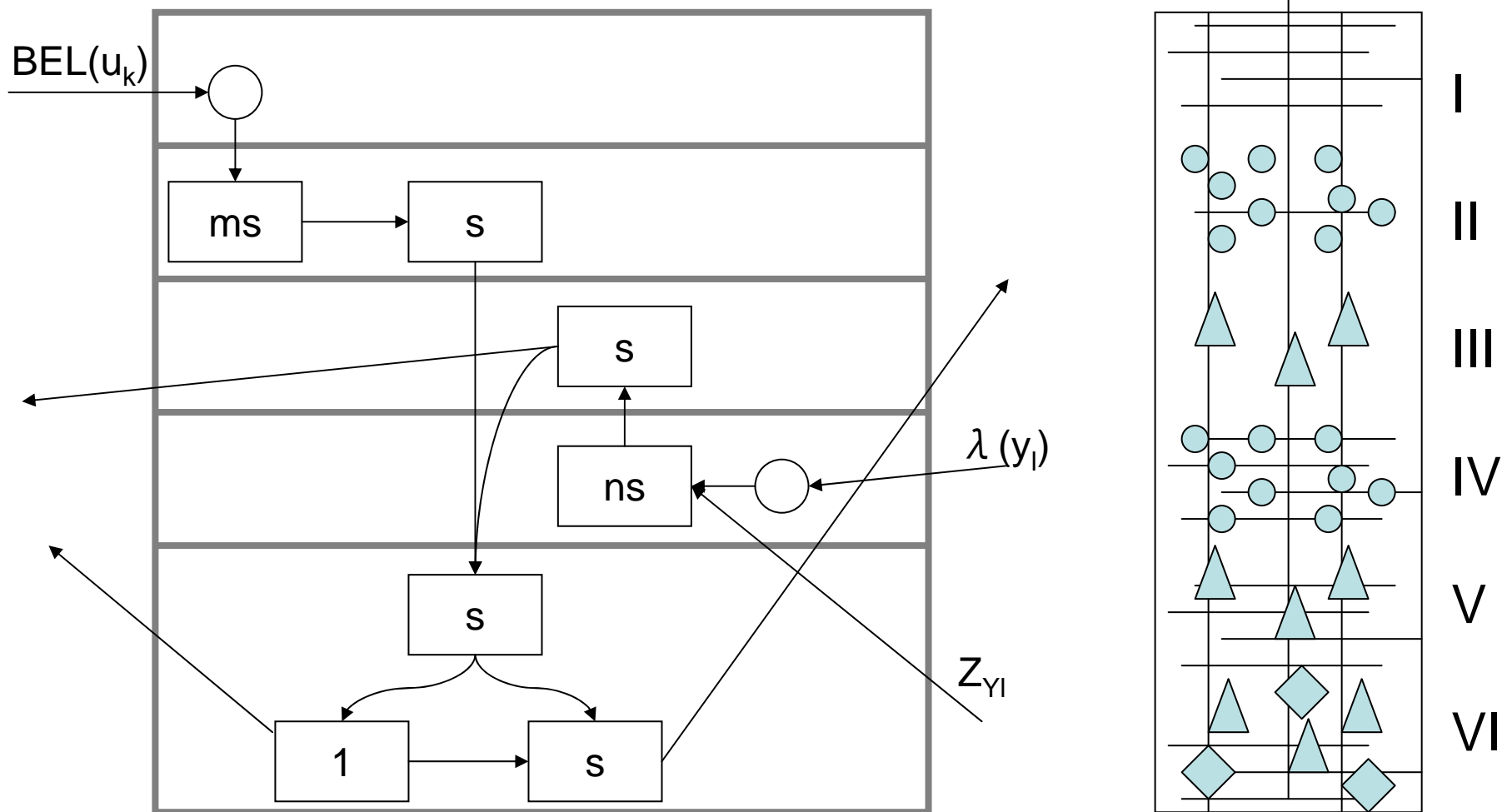


# ハイパーコラム内の配線量



# ハイパーコラム内のニューロン数

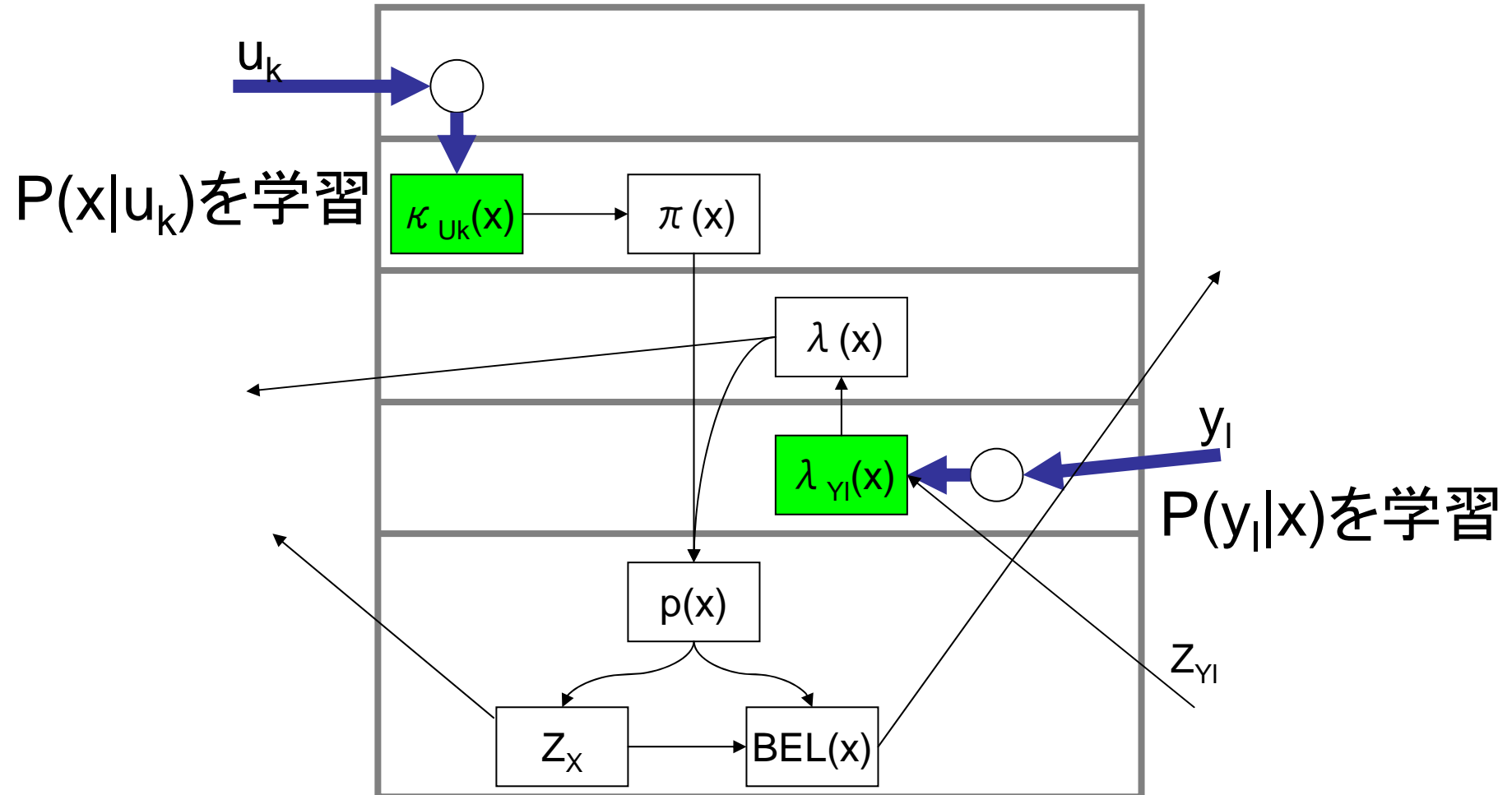
多くのニューロンを必要とする変数が顆粒層に対応、  
入力信号の分配がI層、IV層の接線線維に対応





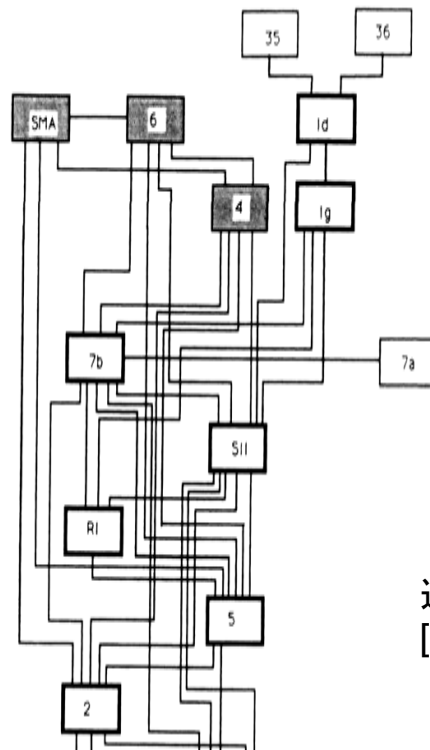
# 学習ステップ

- 親・子ノードからMAP推定結果の情報が送られてくれば、条件付確率表を学習可能。



# 同じ階層の領野間の接続の問題

- 大脳皮質では、上下関係のない、同じ階層の領野間の接続もある。
  - ベイジアンネットの枠組みを超えた通信路？

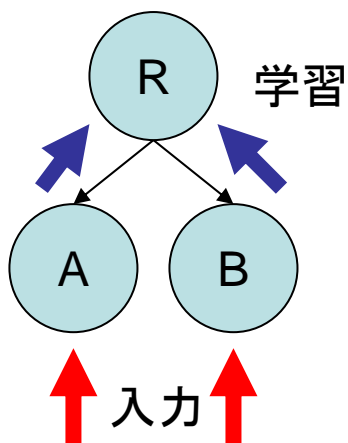


運動野周辺の結線図  
[Felleman, Essen 1991]

# BESOM と強化学習

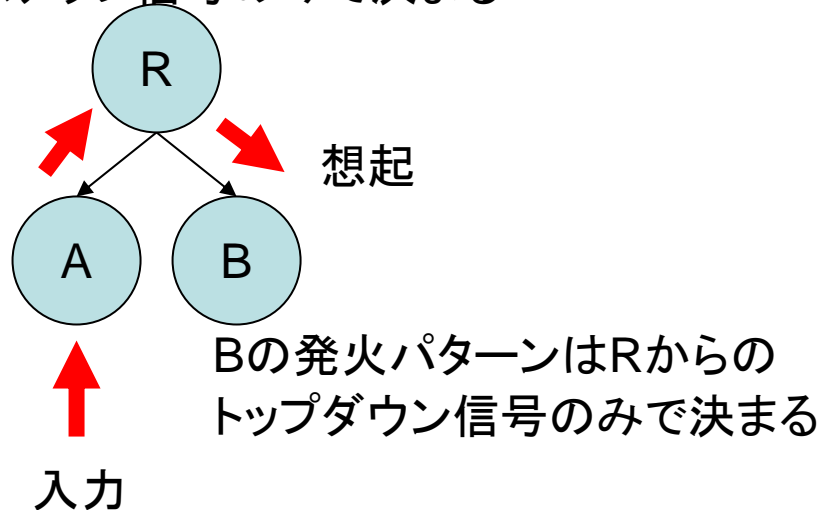
# 本モデルによる連想記憶

- 学習: ノードRのSOMがAとBからの入力の組を学習



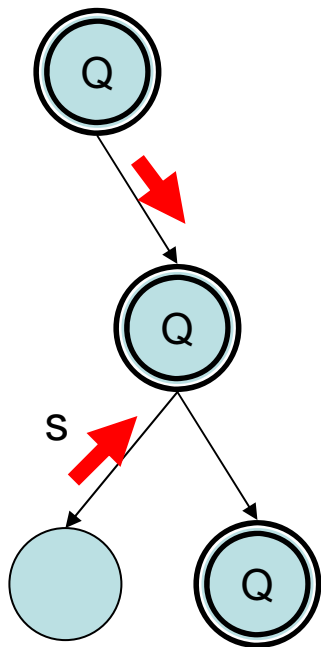
- 想起: Bへの入力を遮断すると、Aの発火パターンから連想された発火パターンがB上に想起

Rの発火パターンはAからのボトムアップ信号のみで決まる



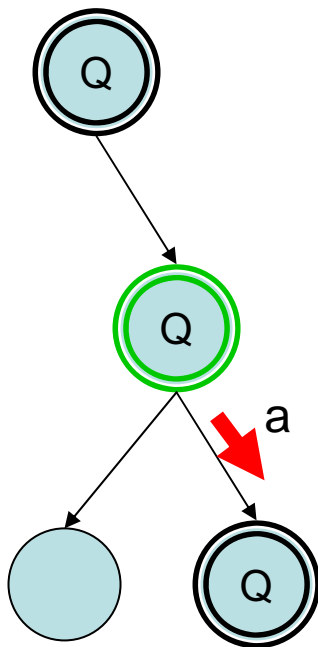
# Qノードの基本動作のサイクル

## 1. 状態認識



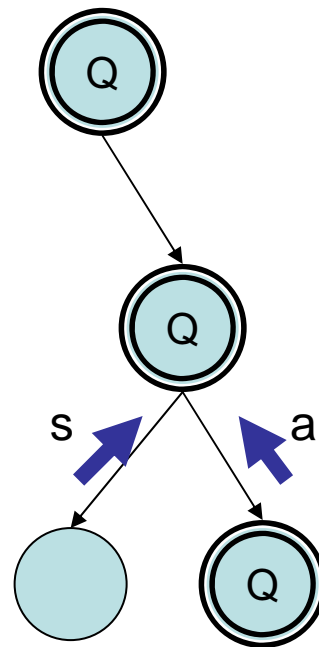
自分が置かれている  
状況をベイズ推定

## 2. 行動選択



選択肢の中から報酬期  
待値最大の行動を選択、  
子Qノードに命令

## 3. 状態行動対学習



$(s,a)$  の組を  
SOM が学習

# 提案モデルと解剖学的構造との対応

大脳皮質のQノードで各状態  $s$  の事後確率を推定

線条体の出力が  
累積報酬の期待値  
を表現

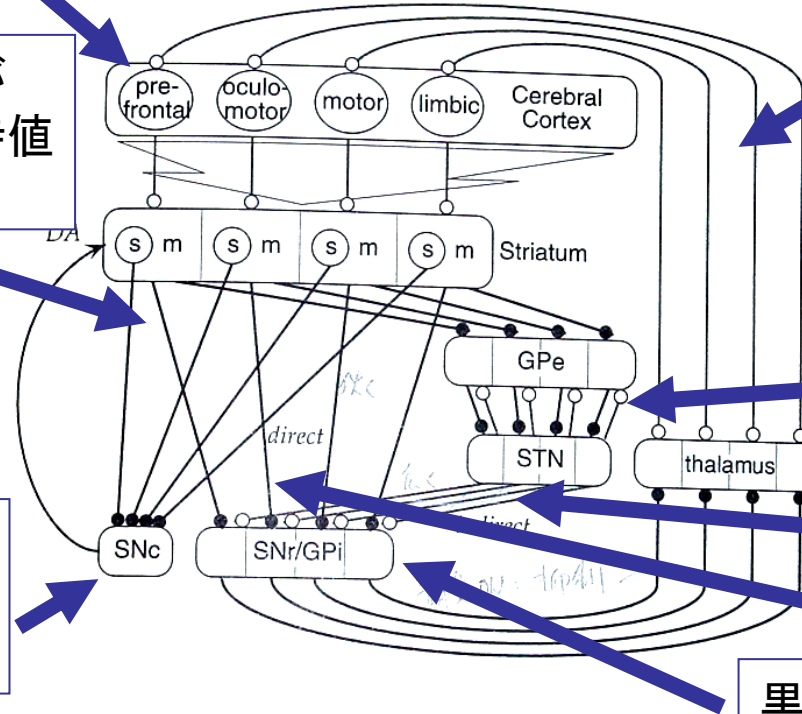
選択された行動が  
視床経由で皮質に返される

行動選択を定期的に  
リセットする回路？  
Cf. コホネン「自己組織化マップ」  
(改訂版)図4.2

間接路：興奮性の広い投射

直接路：抑制性の狭い投射

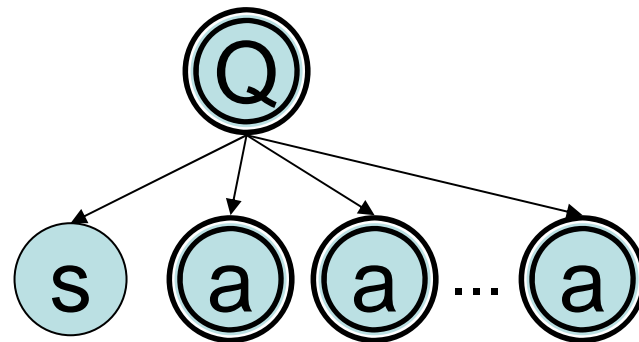
黒質網様部、淡蒼球内節で  
近傍内での最大の期待値を持つ  
行動を選択



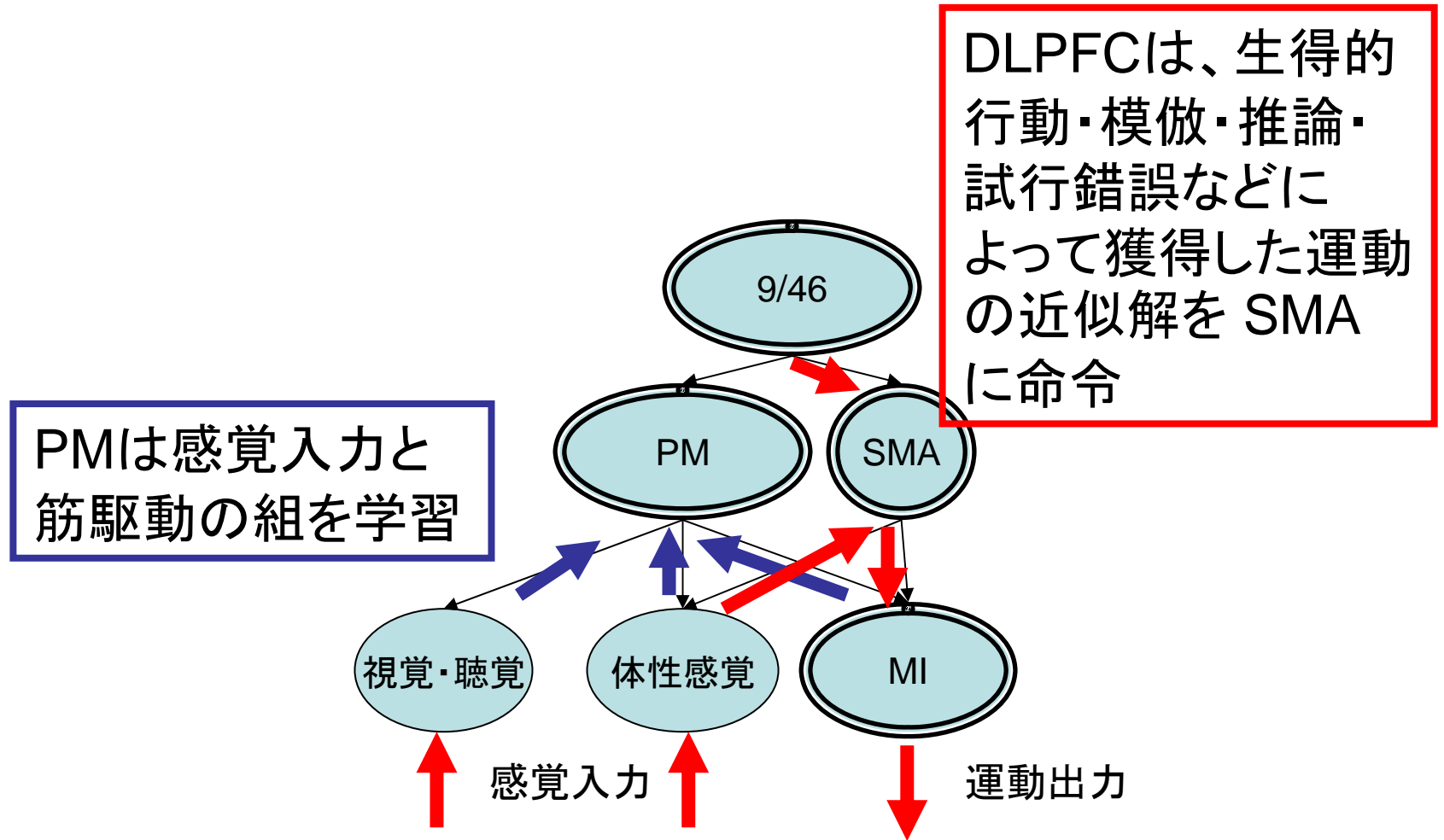
黒質緻密部が  
TD誤差を計算、  
線条体のQ値を更新

# SOMで状態行動対を学習する 計算論的な利点

- 記憶容量。
  - 使われない状態行動対は自然に忘却。
- 近傍学習による汎化能力。
- 次元圧縮。
  - 1階層につき10次元とすれば3階層で1000次元。  
(人間の運動の自由度は200程度)



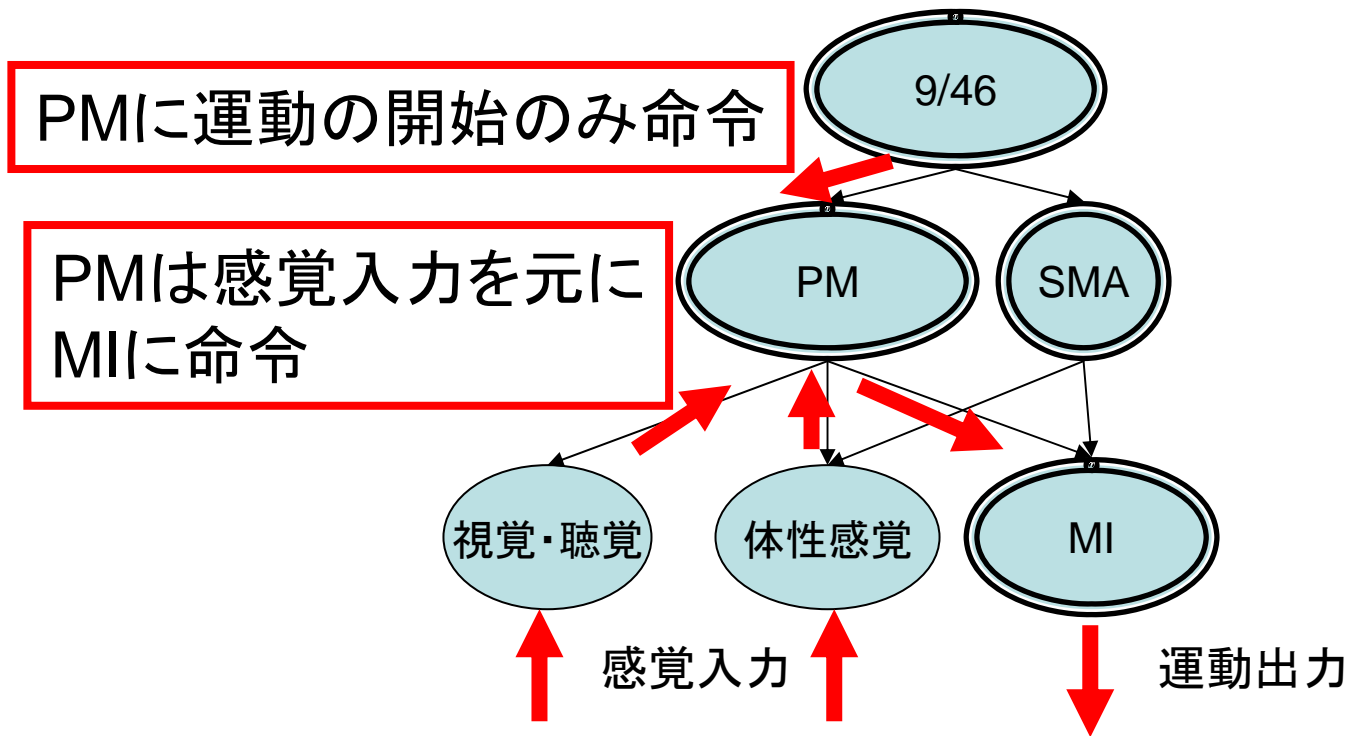
# 運動野の提案モデル： 運動学習の初期





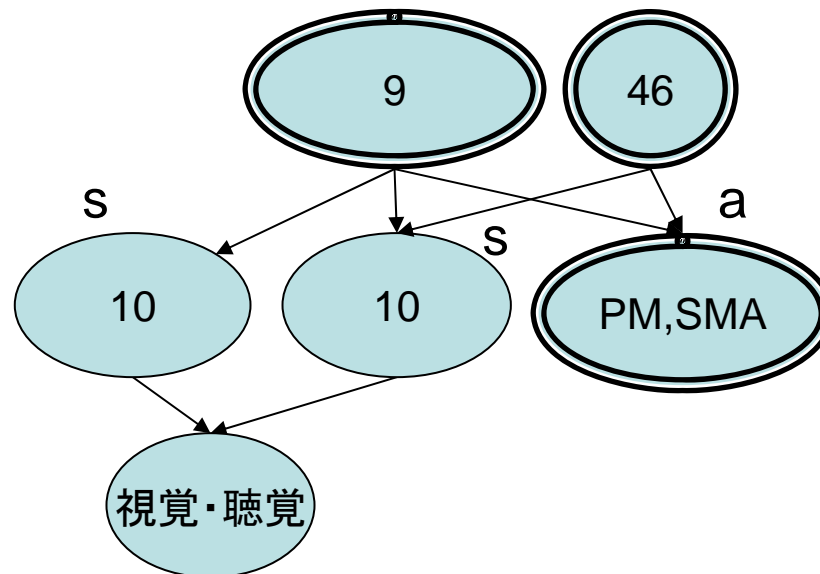
# 運動学習後

- PMは、DLPFCから具体的な命令がなくても、感覚入力を元にMIに自動的に命令が出せる。
- 以後もPMは強化学習で運動を最適化。



# 前頭前野背外側部による 状態変化予測機構の提案モデル

- $(s, s', a)$ テーブル: 9野?
- 状態  $s$ : 10野?
- 状態  $s$  から行動  $a$  を選択: 46野?



今後

# 今後シミュレーションで示す予定

- 学習・認識モデルとしての妥当性
  - 学習ステップ、認識ステップの収束性
  - 汎用性
  - 認識精度
- 様々な認知科学的現象の説明
  - 新規性の検出
  - 馴化、順応

# 今後

- 現在のモデルに足りない点
  - 選択的注意の機構。(視床、前部帯状回)
  - 短期記憶の機構。
  - 再帰的な行動の獲得・実行機構。
  - 宣言的知識の記憶・推論機構。
  - 複数のSOMによるアンサンブル学習。
  - …
- これらの機能を徐々に追加しつつ、簡単な環境下で動作するロボットのシミュレーションを行いたい。

# まとめ

- SOMの双方向ネットワークを仮定して導かれた近似確率伝播アルゴリズムが大脳皮質の解剖学的構造とよく一致し、運動野の定性的機能もよく説明できることを示した。
- 大脳皮質の主要な機能を理解するモデルであり、今後知能の高いロボットの実現に向けて大きく進展するかもしれない。

# 補足説明スライド

# 目指すゴールと方法論

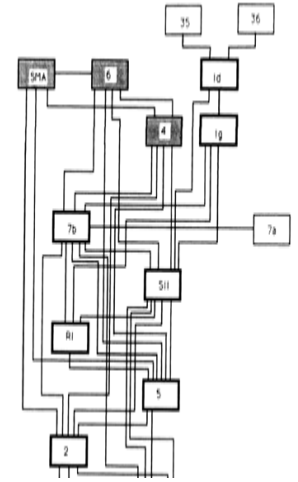
- 人間のように高度に知的なロボットを作る。
- そのための最短かつ確実なアプローチは、まずは脳というシステムをリバーースエンジニアリングして、アーキテクチャのヒントを得ることであると考える。
- アーキテクチャの正しさを確認するために、常に神経科学的知見との整合性を意識する。



# 大脳皮質のモデルとしてのSOM

- 構造的類似性
  - コラム構造、スパース性、ヘブ則による学習
- 機能的類似性
  - 自己組織化によるクラスタリング(教師なし学習)
- 計算論的利点
  - 階層的SOMによる次元圧縮、近傍学習による汎化
- モデルとしての問題点:
  - 「双方向階層的SOM」はあまり研究されていない。

# 大脳皮質のモデルとしての ベイジアンネットワーク



運動野周辺の結線図  
[Felleman, Essen 1991]

- 構造的類似性
  - 有向グラフ、双方向の結合
- 機能的類似性
  - ベイズ推定(ノイズに強い認識、曖昧な判断)、リアルタイム性
- 計算論的利点
  - 事前知識をグラフ構造として表現可能
- モデルとしての問題点:
  - 計算量・条件付確率表のサイズのオーダー、条件付確率表の獲得方法が不明

# SOMの結合度は条件付確率を表す

- ユニット  $x_i$  がMAP推定結果になった回数を  $n$ 、そのうち入力  $v_j$  もMAP推定結果であった回数を  $m^n$ 、その時の結合度を  $w_{ij}^n$ 、入力ベクトルの要素を  $v_j^n = 0$  or  $1$  とする。
- 以下のように帰納的に  $w_{ij}^n = m^n / n$  を示せる。

$$w_{ij}^n = (1 - \alpha)w_{ij}^{n-1} + \alpha v_j^n$$

ただし、

$$= ((n-1)w_{ij}^{n-1} + v_j^n) / n$$

$$\alpha = 1/n$$

$$= ((n-1)m^{n-1} / (n-1) + v_j^n) / n$$

$$m^n = \sum_{i=1}^n v_j^i$$

$$= (m^{n-1} + v_j^n) / n$$

$$w_{ij}^1 = v_j^1 = m^1 / 1$$

$$= m^n / n$$

# $\pi(x)$ の近似

- 仮定:  $\sum \pi_X(u_i)=1$  に正規化済み

$$\pi(x)$$

$$= \sum_{u_1, \dots, u_m} P(x | u_1, \dots, u_m) \prod_i \pi_X(u_i)$$

$$\approx \sum_{u_1, \dots, u_m} \left( \sum_k P(x | u_k) \right) \prod_i \pi_X(u_i) \quad (\text{CPTの近似})$$

$$= \sum_{u_1, \dots, u_m} \sum_k P(x | u_k) \prod_i \pi_X(u_i)$$

$$= \sum_k \sum_{u_k} P(x | u_k) \pi_X(u_k) \sum_{u_1, \dots, u_m / u_k} \prod_{i \neq k} \pi_X(u_i)$$

$$= \sum_k \sum_{u_k} P(x | u_k) \pi_X(u_k) \quad (\text{正規化の仮定より})$$

# $\lambda_X(u_k)$ の近似

$$\sum_{u_1, \dots, u_m / u_k} P(x | u_1, \dots, u_m) \prod_{i \neq k} \pi_X(u_i)$$

$$\approx \sum_{u_1, \dots, u_m / u_k} (\sum_{j \neq k} P(x | u_j) + P(x | u_k)) \prod_{i \neq k} \pi_X(u_i) \quad (\text{CPTの近似})$$

$$= \sum_{u_1, \dots, u_m / u_k} \sum_{j \neq k} P(x | u_j) \prod_{i \neq k} \pi_X(u_i) + P(x | u_k) \sum_{u_1, \dots, u_m / u_k} \prod_{i \neq k} \pi_X(u_i)$$

$$\approx \sum_{u_1, \dots, u_m} \sum_j P(x | u_j) \prod_i \pi_X(u_i) + P(x | u_k) \quad (\text{相手の情報を含める、正規化の仮定より})$$

$$= \pi(x) + P(x | u_k)$$

$$\lambda_X(u_k) = \sum_x \lambda(x) \sum_{u_1, \dots, u_m / u_k} P(x | u_1, \dots, u_m) \prod_{i \neq k} \pi_X(u_i)$$

$$\approx \sum_x \lambda(x) (\pi(x) + P(x | u_k)) \quad (\text{上記式を代入})$$

$$= \sum_x \lambda(x) \pi(x) + \sum_x \lambda(x) P(x | u_k)$$

# 計算機上での計算量・記憶量の 見積もり

# 実用化の可能性

- もし大脳皮質の機能が本アルゴリズムで十分よく近似できるのならば(パルス単位の演算や複雑な非線形演算の影響が支配的でないので)、脳の機能を計算機上で再現するのに必要な記憶量・計算量は**意外にも少ない**ことになる。
  - さらに、計算論的アルゴリズムと扱うデータの統計的性質が決まれば、それを利用した最適化ができ、素朴なニューロン単位のシミュレーションよりも記憶量・計算量が大幅に削減できる。

# 記憶量・計算量の見積もり

- 1万個のシナプスを持った100億個のニューロンは、1シナプス1byteとして、100Tbytesで表現可能。
- 近似確率伝播アルゴリズム1ステップの実行に必要な演算は約400T回。1ステップ10msec とすると 40000TFlops 程度必要。
  - しかし、発火率10%のスパース性の利用やメッセージ伝播のスケジューリングを工夫して最適化すれば 100TFlops 程度になるかもしれない。



# 10～20年後のハードウェアコスト

- 現在のパソコンは 1GB・1GFlops 程度。
- 早ければ10～20年後には100GB・100GFlops × 1000台、10億円程度のクラス・コンピュータで、人間の脳と同規模のシミュレーションが実現可能。
- 機能特化により、100～1000万円程度のコンピュータで、ある程度複雑な作業をロボットができるようになれば、十分市場が開ける可能性がある。

# 先行研究

# 先行研究: SAM [Fukushima 1987]

- SAM (Selective Attention Model) は、画像認識装置ネオコグニトロン[Fukushima 1980]にトップダウンの回路を追加したもの。
  - 競合学習するノードを**双方向**のネットワークで結合。
- 選択的注意、ノイズに強い認識など、人間の視覚の特徴を再現。
  - 視覚野の妥当なモデル。
  - **他の領野も同様のモデルで説明できると考えるのは妥当と考える。**

# SAM との比較

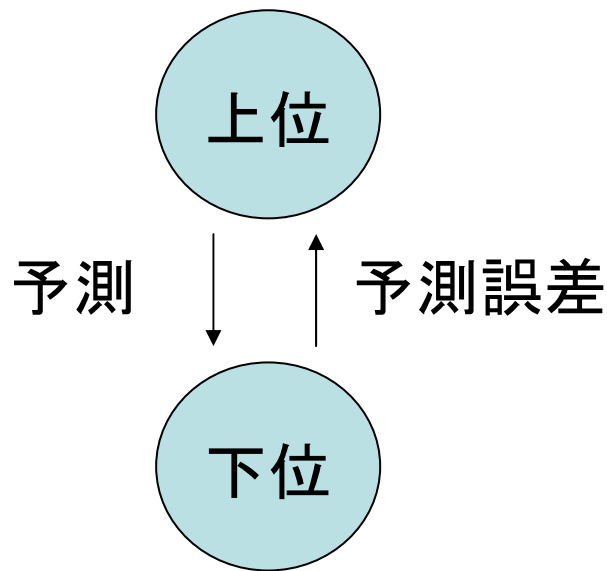
- 競合学習で特徴を分類するノードが階層的かつ双方向で結合するという基本構造は BESOM と同じ。
- SAM は、欠損した文字を補完したイメージが最下層に想起される。BESOM では補完された線を認識することはない。むしろ欠損部分が強調される。

A

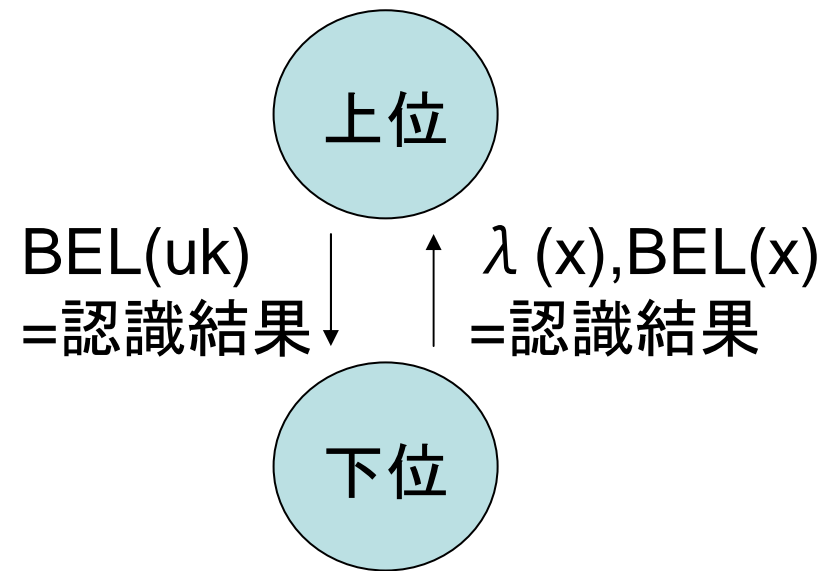
# 先行研究: predictive coding model

[Rao and Ballard 1999]

- 双方向の信号の意味が BESOM と違う。



predictive coding model



BESOM model