

## 入力ゲート付きニューラルネットワーク

村田, 純一

九州大学大学院システム情報科学研究科電気電子システム工学専攻

野田, 哲史

九州大学総合理工学研究科エネルギー変換工学専攻 : 修士課程

平澤, 宏太郎

九州大学大学院システム情報工学研究科電気電子システム工学専攻

<https://doi.org/10.15017/1524284>

---

出版情報 : 九州大学大学院システム情報科学紀要. 2 (2), pp.283-288, 1997-09-26. 九州大学大学院システム情報科学研究科

バージョン :

権利関係 :

# 入力ゲート付きニューラルネットワーク

村田 純一\*・野田 哲史\*\*・平澤宏太郎\*\*\*

## Neural Networks with Input Gates

Junichi MURATA, Tetsushi NODA and Kotaro HIRASAWA

(Received June 23, 1997)

**Abstract:** An architecture of multi layer neural networks is proposed. The networks are equipped with gates on their input channels to control flow of the input signals. A gate on an input channel opens and closes depending on the current values of the other input signals. The dependency is automatically determined based on training data so that the gate opens when the corresponding input is necessary and closed when not necessary. These gates give the networks a good generalization ability because they can eliminate harmful inputs. And they provides us with an insight into the input-output relationship since they can indicate which input is significant in what situation.

**Keywords:** Neural networks, Input gates, Input selection, Generalization ability, Non-black-box modeling

### 1. はじめに

従来、システムを制御する制御器は、制御対象（システム）を伝達関数、状態方程式などのモデルやステップ応答、周波数応答などの応答波形によって把握し、制御目的に応じて設計されてきた。

従来の制御において、「自動化」は一応実現されているかのように見えるが、それがうまく機能するのは固定された対象に対してのみである。システムが変化する場合は、人間が予期されるシステムの変化を想定しそれに基づいて制御器を変化させているにすぎない。すなわち、人間が予め想定していないシステムの変化には制御器は対処できない。

今後我々が取り扱う制御対象は、より複雑で変化の多いシステムになることが予想される。そのようなシステムにおいて、例えば予想できない事故などの特殊事情が発生した場合、従来の制御では不十分である。我々は諸々のシステムの変化に柔軟に対処できる制御器をつくる必要がある。これを実現するためには、制御器に人間の持つ獨創性、一般化能力、抽象化能力、洞察力といった高度な能力を持たせなくてはならない。これにより制御器が自分自身を設計、構築、改造することが可能となる。

本研究の最終目標は、このような自己形成機能を持つ制御器設計にあるが、それをめざす上でまず制御対象を表現するモデルの自動形成を行う。このモデルはシステムの挙動に関する有限のデータから構築されるものであ

るが、システムの一般的規則を表現したモデルでなければならぬ。さらに、その一般的規則が陽に具体的に解るようなモデルであるべきである。これにより、制御器自身が制御対象を理解し、制御対象の変化に対応できるようになるからである。

以上から、本論文では制御対象の同定を行う自動形成システムの構築を考える。すなわち、対象システムの観測データに基づいて、システムの一般的規則を内包したモデルを自動形成する。モデルには、一般性と正確さを兼備しさらに解釈可能であることが要求される。そのため、ある程度の一般化（汎化）能力を備え持つニューラルネットワークをモデル化の道具として用い、これに一般性と解釈可能性を付与するための改良を施した入力ゲート付きニューラルネットワークを提案する。

### 2. モデルの一般性・抽象性と入力情報

本論文では、観測データに基づいて対象の挙動を表し、かつ一般性、汎用性、正確さを満たすモデルを、ニューラルネットワークを用いて自動形成することを目的としている。しかし、ニューラルネットワークはそのままでは満足できるほどの汎化能力を持ってはいない。

最も単純で一般性の低いモデル化は観測された入力と出力の値の組を表に羅列することである。しかし、これでは観測データにない入力に対する出力は求められない。また、本来は不要な情報までも誤って入力として扱うことも避けられない。つまり、モデル作成に使用したデータ以外のことには対処できず、しかもデータ中の不要な情報に左右されやすく、汎化能力が無いということになる。

汎化能力を向上させるには、観測データの内外挿に

平成9年6月23日受付

\* 電気電子システム工学専攻

\*\* 総合理工学研究科エネルギー変換工学専攻修士課程

\*\*\* 電気電子システム工学専攻

よってデータに含まれていない入力に対しても出力を得ることができるようにすることと、不要な入力情報を取り除くことが必要である。このようにして得られたモデルは高い汎化能力を得ることができ、システムの入出力の構造についての知見を得ることもできる。

ニューラルネットワークにおいては、内外挿による汎化能力の向上は自ら備わっている機能であり、不要な入力を取り除く、言い換えれば必要な入力を選択するための方法もいくつか提案されている<sup>2),4)</sup>。しかし、これらの方法では、各入力情報が必要であるか否かは個々の入力に独立に決まるものとしていた。すなわち、ある入力は常に必要であるかあるいは常に不要であるかのいずれかであり、他の入力の値に応じて要不要が変化することは想定していない。しかし、現実には、例えば装置を自動運転している時には大きな意味を持つ自動操作信号が、手動操作の時にはシステム何も影響を与えない、など、入力の要不要は状況によって変化する場合も多い。

本論文では、入力の要不要が他の入力情報に依存する場合を考え、この依存関係を表現し、入力を用いるか否かを適切に切り替えることのできるニューラルネットワークとして、入力ゲート付きニューラルネットワークを提案する。

### 3. 入力ゲート付きニューラルネットワーク

入力情報の要不要が他の入力の値に依存するという状況は、各入力の伝達経路にゲートを設け、そのゲートの開閉を他の入力の値で決定することによって表現することができる。このような入力にゲートを持つニューラルネットワークを「入力ゲート付きニューラルネットワーク」と呼ぶことにする。

ゲートを持つニューラルネットワークとして、JordanらによるHierarchical Mixture of Experts (HMM)がある<sup>5)</sup>。しかしHMMでは、問題を細分化し、細分化された個々の小問題を取り扱うExpert networkの切り替えを行うためにゲートを利用しており、入力の要不要を判定するために使用するこの論文におけるゲートとは働き、取り扱いが異なっている。

Fig.1 に示すように、入力端とニューラルネットワーク本体の間の入力伝達経路上にゲート  $G$  がある。このゲートは、ゲートが設置されている伝達経路以外の伝達経路に流れる入力値を取り込み、その値に応じて開閉をする。例えば、ある時刻に各入力端に入力値  $u_1, \dots, u_N$  が入力されたとする。この中で第  $i$  番目の入力  $u_i$  が他入力との間にある相互関係から出力の記述に不要な場合、 $u_i$  のニューラルネットワーク部への入力を防ぐべくゲート  $G_i$  が閉じる。逆に必要な場合はゲート  $G_i$  は開く。すなわち、ゲート  $G_i$  内には  $u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_N$  の値に応じて  $u_i$  の要不要を判定し、この信号をニューラル

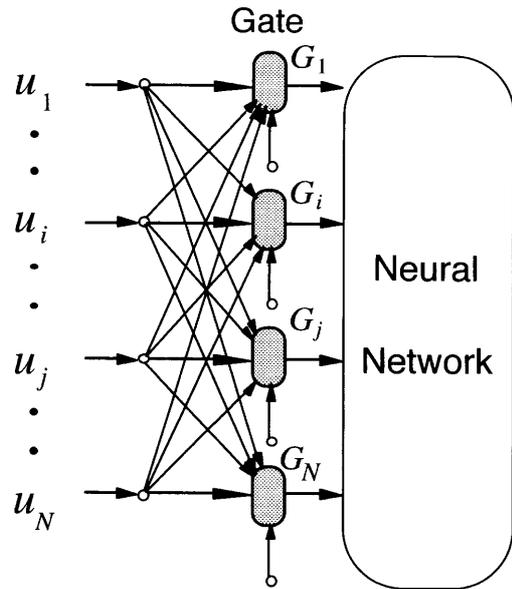


Fig.1 Neural network with input gates.

ネットワークに導くゲートの開閉を決定する機能が備わっており、これは学習により適切に機能するように決定される。

すべてのゲートが開いている状態から学習を開始すると、はじめはニューラルネットワークに入力されていた不要な入力も、ゲート部が学習を行い、入力値に応じてゲートが適切に開閉することによって、最終的にはモデル化に有効な入力情報のみが入力されるネットワークモデルとなる。

また、学習によって決定されたゲートの構造を調べることにより、どのような入力パターンの時にどの入力情報が不要か、ということを知ることができる。このことは、抽象的概念をネットワークモデルで表現するものであり、後の制御器設計に大きく役立つものである。

次に、ゲートの構造として2通りの構造を提案し、それぞれの学習アルゴリズムの説明を行う。

## 4. ゲートの構造と学習アルゴリズム

### 4.1 線形構造のゲート

線形構造のゲート  $G_i$  は、入力  $u_i$  にゲートの開度 (0 以上 1 以下の値) を乗じた値  $v_i$  をニューラルネットワークに伝えるもので、以下の式で表される。

$$v_i = G_i(u_i) = S(z_i) \cdot u_i \tag{1}$$

$$S(z_i) = \frac{1}{1 + \exp(-\alpha z_i)} \tag{2}$$

$$z_i = \sum_{j \neq i} \gamma_j^i u_j + \gamma_0^i \tag{3}$$

ここで、

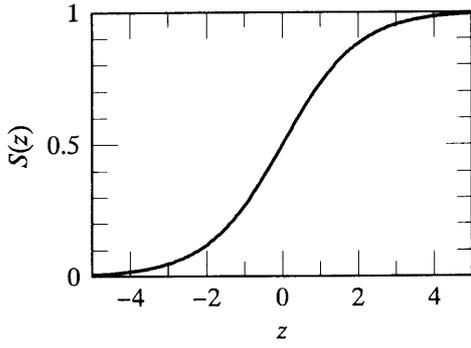


Fig.2 Switching function.

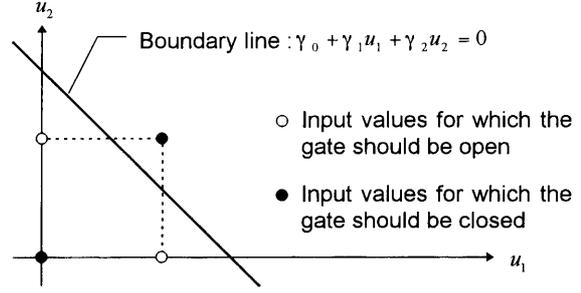


Fig.3 Limitation of linear structure gates.

$\gamma_j^i$ : 第j入力からゲート $G_i$ への重み係数  
 $S(\cdot)$ : スイッチング関数 (Fig.2参照)

である。

ゲート  $G_i$  の開度を表すスイッチング関数  $S(z)$  の値は、 $u_i$  以外の入力値と、それらの入力値とゲート  $G_i$  の間の重み係数 $\gamma_j^i$  とから決定される。ここで、 $z_i$ が、各入力値とゲート間の重み係数との線形和という構造をしているのが特徴である。以下に、この線形和の係数  $\gamma_j^i$  を学習によって求めるためのアルゴリズムを説明する。

ゲートの役割はネットワークが正しい出力を出す限りにおいて、できるだけネットワークに与える入力情報を少なくすることである。そこで、ゲートの係数 $\gamma_j^i$ の両方を、次の評価関数を最小化するように Back propagation (BP) 法により学習する。

まず、ネットワークが正しい出力を出しているかどうかを表す評価として、一般的な BP 法の評価関数と同じく、ネットワーク出力と教師信号の誤差平方和を用いる。次に、ネットワークに与える入力情報の量を表す評価として、ゲートの開度  $S(z_i)$  を用いる。これら2つの評価を用いて、ネットワークの学習における評価関数  $E$  を定義する。

$$E_1 : \text{出力と教師信号の間の誤差平方和} \quad (4)$$

$$E_2 = \frac{1}{2} \sum_i \{S(z_i)\}^2 \quad (5)$$

$$E = E_1 + \beta E_2 \quad (\beta > 0) \quad (6)$$

この評価関数を最小化するゲートの重み係数を BP 法を用いて求める。ネットワーク内の重み係数は、通常のニューラルネットワークと同様に誤差平方和 $E_1$  をBP法で最小化することにより学習を行う。ゲート部の $\gamma_j^i$ の更新式は以下の通りである。

$$\gamma_j^i \leftarrow \gamma_j^i - \eta \frac{\partial \dagger E}{\partial \dagger \gamma_j^i} \quad (7)$$

$$\frac{\partial \dagger E}{\partial \dagger \gamma_j^i} = \frac{\partial \dagger E}{\partial \dagger S(z_i)} S'(z_i) u_j \quad (8)$$

$$\frac{\partial \dagger E}{\partial \dagger S(z_i)} = \frac{\partial \dagger E_1}{\partial \dagger v_i} u_i + \beta S(z_i) \quad (9)$$

$\partial \dagger E_1 / \partial \dagger v_i$  はニューラルネットワークの重み係数の学習における BP の後向き計算から求めることができる。

ここで、 $\gamma_j^i$ の初期値の絶対値は、 $z_i$ の絶対値が小さくなるように小さくとる。 $z_i$ の絶対値が大きくなると、スイッチング関数の飽和領域に陥り、学習が進まなくなるからである。

また、誤差  $E_1$  が大きくならない範囲で、ゲートの開度  $E_2$  を小さくすることを意図しているので、 $\beta$  は小さい正の数とする。しかし、 $\beta$  が過度に小さいとゲートが機能しないことになるので調整が必要である。

#### 4.2 デルタ関数を用いたゲート

ゲートの開度を決定する  $z_i$  が入力  $u_j$  の線形関数である場合、入力変数の張る空間は、ゲートが開く領域と閉じる領域に超平面で分離されることになる。したがって、Fig.3のようなゲートの開閉を実現したい時には、線形構造のゲートでは対応できない。もっときめ細かく分離しようとする、何らかの非線形のゲートを用いる必要がある。そこで、最も直接的な方法として、デルタ関数を入力値の一つ一つに配したゲートを考案する。

入力変数が有限個の離散値をとる場合、Fig.4のように入力値が取りうる点にデルタ関数を配置する。デルタ関数は

$$\delta(x_1, \dots, x_N) = \begin{cases} 1 & (x_1 = \dots = x_N = 0) \\ 0 & (\text{otherwise}) \end{cases} \quad (10)$$

で表される。

第  $j$  番目の入力  $u_j$  が取りうる値の集合を  $\{\phi_k^j\}$  とする

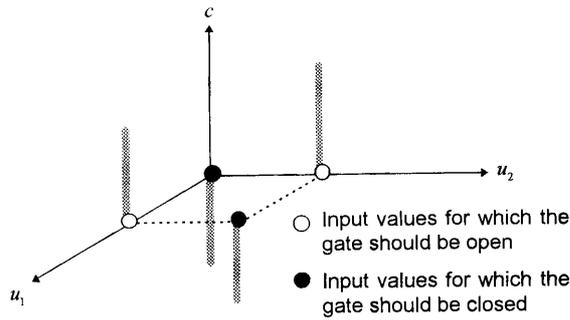


Fig.4 Allocation of delta functions.

とき、ゲートは次式に従って開閉するものとする。

$$v_i = G_i(u_i) = S(z_i) \cdot u_i \quad (11)$$

$$z_i = \sum_{k_1} \cdots \sum_{k_{i-1}} \sum_{k_{i+1}} \cdots \sum_{k_N} c_{k_1, \dots, k_N} \cdot \delta(u_1 - \phi_{k_1}^1, \dots, u_{k_N} - \phi_{k_N}^1) \quad (12)$$

とする。ここで、 $c_{k_1, \dots, k_N}$  は入力値が  $(\phi_{k_1}^1, \dots, \phi_{k_N}^1)$  の時の  $z_i$  の値に相当し、ゲートが開くべき入力値では  $c_{k_1, \dots, k_N}$  は大きく、逆にゲートが閉じるべき入力値では小さくなる。

このゲート構造において、観測データを用いて学習するのは、 $c_{k_1, \dots, k_N}$  である。学習に用いる評価関数は線形構造のゲートと同じく、

$$E_1 : \text{出力と教師信号の間の誤差平方和} \quad (13)$$

$$E_2 = \frac{1}{2} \sum_{j \neq i} \{S(z_i)\}^2 \quad (14)$$

$$E = E_1 + \beta E_2 \quad (\beta > 0) \quad (15)$$

とする。

$c_{k_1, \dots, k_N}$  の更新式は次のようになる。

$$c_{k_1, \dots, k_N} \leftarrow c_{k_1, \dots, k_N} - \eta \frac{\partial^\dagger E}{\partial^\dagger c_{k_1, \dots, k_N}} \quad (16)$$

$$\frac{\partial^\dagger E}{\partial^\dagger c_{k_1, \dots, k_N}} = \frac{\partial^\dagger E}{\partial^\dagger S(z_i)} S'(z_i) \cdot \delta(u_1 - \phi_{k_1}^1, \dots, u_{k_N} - \phi_{k_N}^1) \quad (17)$$

$$\frac{\partial^\dagger E}{\partial^\dagger S(z_i)} = \frac{\partial^\dagger E_1}{\partial^\dagger v_i} u_i + \beta S(z_i) \quad (18)$$

### 4.3 学習手順

ここでの学習は、誤差が大きくなりえない限りにおいて、ゲートを可能な限り閉じることを目的としている。そこで Fig.5 に示すように、まずゲートを開いた状態でニューラルネットワーク部を学習し (フェーズ1)、次に、この学習で達成した誤差を保ちつつゲートが閉じるようにゲート部のみの学習を行う (フェーズ2)。

また、ゲート部の学習を行う際、Fig.6 に示すようにスイッチング関数の傾き  $\alpha$  を最初小さくしておき、学習が

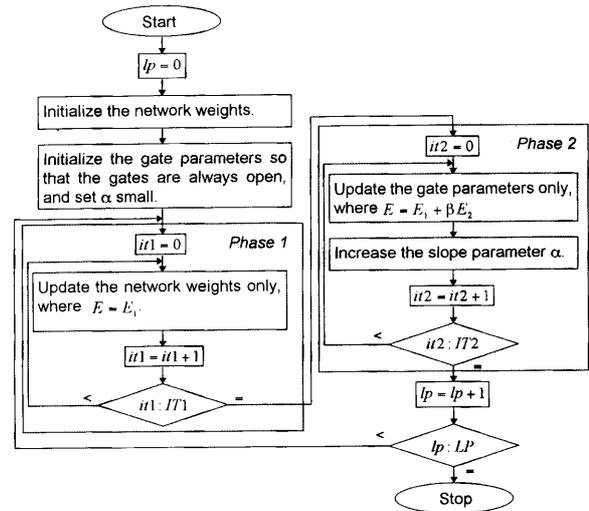


Fig.5 Two-phase learning of neural networks with input gates.

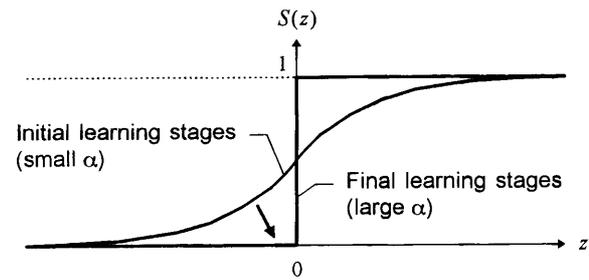


Fig.6 Slope of switching function.

進むにつれて大きくしていく。これにより学習の初期段階では、スイッチング関数の飽和領域に陥ることなくゲート構造の  $\gamma$  や  $c$  の学習が進み、また学習の終盤では、ゲートの開閉が明確になるという効果を得ることができる。

### 5. シミュレーション

入力ゲート付きニューラルネットワークの効果を確認するために、ネズミの行動を理解しようとするネコの問題を例題として、シミュレーションを行った。

ネズミは自分を中心とする  $3 \times 3$  マスの領域 (Fig.7) 内に存在するもの (オブジェクト) に対して反応する。この例題では、ネコ、食べ物、暗い所の3種類のオブジェクトを出現させる。ネズミはこれらのオブジェクトに対して、以下のルールに従って行動するものとする。

1. ネコがいればその反対方向に逃げる。
2. 食べ物または暗い所があればそこに行く。
3. 複数のオブジェクトが存在する場合は、次の優先順位にしたがって優先順位の最も高いオブジェクトに

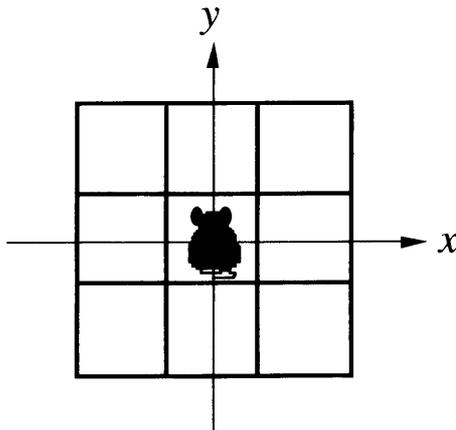


Fig.7 Coordinates around mouse.

Table 1 Possible inputs to neural network.

Input number	Object	Attribute	Value
1	Cat	Existence	0 or 1
2		x-coordinate	-1, 0 or 1
3		y-coordinate	-1, 0 or 1
4	Food	Existence	0 or 1
5		x-coordinate	-1, 0 or 1
6		y-coordinate	-1, 0 or 1
7	Dark place	Existence	0 or 1
8		x-coordinate	-1, 0 or 1
9		y-coordinate	-1, 0 or 1

反応する：ネコ > 食べ物 > 暗い所。

4. ネズミは後方の物には反応しない。

ネズミの回りに三つのオブジェクトを配する組合せは、一つのオブジェクトについて9マスのいずれかに有るもしくは無い、の10通りの可能性があるので、 $10^3 = 1000$ 通りの組合せがある。このうちの一部の組合せに関するデータに基づいた学習により、ネズミの行動をモデル化する。

モデルへの全入力情報は、ネコ、食べ物、暗い所それぞれの有無、及び存在する場合の  $x, y$  座標の計九つ (Table 1) である。

ネズミの行動の観測データから、ネズミの行動ルールをモデル化することがシミュレーションの目的である。特に上記のルールのうち3番目のルールは、ある入力情報の要不要が他の入力情報に依存して変化することを示している。例えば、食べ物の有無の情報はネコがいない時には必要であるが、ネコがいる時には不要である。このような依存関係が入力ゲート付きニューラルネットワークによって発見できるかどうかを検証する。

Table 2 Gate opening (1000 training data).

Existence			Gate opening		
Cat	Food	Dark place	$G_1$	$G_4$	$G_7$
0	0	0	0.864	0.503	0.583
0	0	1	0.578	0.939	0.583
0	1	0	0.582	0.503	<u>0.004</u>
0	1	1	0.231	0.939	<u>0.004</u>
1	0	0	0.864	<u>0.000</u>	<u>0.000</u>
1	0	1	0.578	<u>0.000</u>	<u>0.000</u>
1	1	0	0.582	<u>0.000</u>	<u>0.000</u>
1	1	1	0.231	<u>0.000</u>	<u>0.000</u>

このシミュレーションにおいては、特に各オブジェクトの有無を表す入力の要不要の依存関係に着目しているため、これに対応する第1, 4, 7入力ノードにゲートを設ける。また、この例題では線形構造のゲートで十分であるため、これを用いるものとする。

ニューラルネットワークは入力層(9ノード)、第一中間層(9ノード)、第二中間層(9ノード)および出力層(2ノード)を持つ4層構造のものである。また、フェーズ1の学習回数  $IT1$  を50000回、フェーズ2の学習回数  $IT2$  を30000回としてシミュレーションを行った。

学習されたゲートが正しく開閉するかどうかを検証するために、ネコ、食べ物、暗い所の有無の組み合わせ  $2^3 = 8$  通りのパターンに対する学習済みゲートの開閉を調べた。

1000通りの組合せの全てを用いて学習を行ったネットワークが、これら検証用の8通りの場合にどのようにゲートを開閉するかを Table 2 に示す。表中の数字はゲートの開度を表し、1がゲートが完全に開いていることを、0が完全に閉じていることを示す。表中で、本来ゲートが閉じるべきケースについては下線を施している。

Table 2 から、ネコがいる場合は食べ物と暗い所の有無に関するゲートは閉じるなど、学習後のネットワークのゲートは期待された通りに開閉していることが分かる。これから、ネズミの行動ルール中の反応すべきオブジェクトの優先順位が、入力ゲート付きニューラルネットワークモデルによって正しく獲得されたことが確認される。

なお、学習用観測データ数が700, 500, 300の場合の結果を Table 3, Table 4, Table 5 に示す。

学習用観測データ数が700, 500の時は、表中のゲートが本来閉じるべき10箇所中8箇所まで正しくゲートを閉じることに成功した。これは、ネコが存在する時には食べ物、暗い所の情報はいづれも必要ない、というルールを獲得したことに相当する。

学習用観測データ数が300の時は、表の10箇所中6箇所

**Table 3** Gate opening (700 training data).

Existence			Gate opening		
Cat	Food	Dark place	$G_1$	$G_4$	$G_7$
0	0	0	0.690	0.475	0.500
0	0	1	0.500	0.450	0.500
0	1	0	0.475	0.475	<u>0.525</u>
0	1	1	0.289	0.450	<u>0.525</u>
1	0	0	0.690	<u>0.000</u>	<u>0.000</u>
1	0	1	0.500	<u>0.000</u>	<u>0.000</u>
1	1	0	0.475	<u>0.000</u>	<u>0.000</u>
1	1	1	0.289	<u>0.000</u>	<u>0.000</u>

**Table 4** Gate opening (500 training data).

Existence			Gate opening		
Cat	Food	Dark place	$G_1$	$G_4$	$G_7$
0	0	0	0.949	0.354	0.367
0	0	1	0.762	0.358	0.367
0	1	0	0.735	0.354	<u>0.341</u>
0	1	1	0.324	0.358	<u>0.341</u>
1	0	0	0.949	<u>0.009</u>	<u>0.008</u>
1	0	1	0.762	<u>0.009</u>	<u>0.008</u>
1	1	0	0.735	<u>0.009</u>	<u>0.007</u>
1	1	1	0.324	<u>0.009</u>	<u>0.007</u>

**Table 5** Gate opening (300 training data).

Existence			Gate opening		
Cat	Food	Dark place	$G_1$	$G_4$	$G_7$
0	0	0	0.513	0.844	0.997
0	0	1	0.332	1.000	0.997
0	1	0	0.252	0.844	<u>0.263</u>
0	1	1	0.137	1.000	<u>0.263</u>
1	0	0	0.513	<u>0.000</u>	<u>0.338</u>
1	0	1	0.332	<u>0.006</u>	<u>0.338</u>
1	1	0	0.252	<u>0.000</u>	<u>0.001</u>
1	1	1	0.137	<u>0.006</u>	<u>0.001</u>

まで正しくゲートを閉じることに成功した。これは、ネコが存在する時には食べ物の情報は必要ない、ネコと食べ物の両方が存在する時には暗い所の情報は必要ない、というルールに相当する。

## 6. ま と め

本論文では、階層型ニューラルネットワークの入力にゲートを設けた入力ゲート付きニューラルネットワークを提案した。これは、入力として必要な信号を、他の信号との相互依存関係も含めて表現できるものであり、入力信号の要不要を決定するゲートの機能は学習によって自動的に決定される。したがって、必要な信号のみを入力として受け入れることが可能となり、学習時の状況とは異なる状況でも良好な性能を発揮する能力（一般性、汎化能力）を向上させ、さらに、入出力データの間の構造的な関係に関する知見をも与える。

システムのモデル作成を行う人工的な例題に適用することにより、データ数が十分あれば、入力の要不要が適切に判断され、対象システムの特徴を読み取ることが可能であることが確認された。

データ数が少ない場合への対処、および、さまざまなシステムのモデル化に汎用的に用いることのできるゲートの構造の開発が今後の課題である。

## 参 考 文 献

- 1) 野田, 村田, 平澤: 第 14 回計測自動制御学会九州支部学術講演会予稿集, 221 (1995).
- 2) 野田, 村田, 平澤, 阿部: 平成 8 年電気学会電子・情報・システム部門大会予稿集, 531 (1996).
- 3) 野田, 村田, 平澤: 第 6 回インテリジェントシステム・シンポジウム予稿集, 127 (1996).
- 4) R. Battiti: Using Mutual Information for Selecting Features in Supervised Neural Net Learning, *IEEE Trans on Neural Networks*, **5**, 4, 537/550 (1994).
- 5) M.J. Jordan and R.A. Jacobs: Hierarchical Mixtures of Experts and the EM algorithms, *Proc. of 1993 International Joint Conference on Neural Networks*, **2**, 1339/1344 (1993).

