解説

過負荷学習法[†]

野田 五十樹*

1. はじめに

神経回路網を応用する上でいつも問題となるのが、どのような構造の回路網を用いるかということである。なかでも、隠れ層にどれだけの数のユニットを用意するかは普遍的な問題となっている。

隠れ層のユニット数は神経回路網のいくつかの能力を決定する重要なファクターである。まず、このユニット数は回路網が実現できるタスクの複雑さに大きな影響を及ぼす。単純に言えば、複雑なタスクを実現するためには、十分多くの数のユニットを用意しなければならない。

また、隠れ層のユニット数は回路網の学習における汎化能力にも大きな影響を及ぼす。一般にコンパクトな(隠れユニット数が少なく、リンクの数が少ない)回路網の方が汎化能力に優れ、未知の入力に対し妥当な出力を得ることができる。

さらに、隠れ層のユニット数は回路網が獲得する内部表現にも大きな影響を及ぼす。隠れ層のユニットの活性パターンは回路網が実現しているタスクに必要な情報を何らかの形で表している。よって、回路網の学習は、学習すべきタスクに必要な情報の表現(内部表現)を獲得するプロセスとみなすことができる。そしてこの機能を利用して、性質の良くわからないデータの解析や特徴抽出を行なうことができる。この内部表現獲得においても、獲得される内部表現の質は隠れ層のユニット

数に依存しており、適切な数のユニットを用意する必要がある。

このように、神経回路網の学習においては、隠れ層のユニット数を慎重に選ばなければならない。言い替えば、学習すべきタスクとうまくバランスする数のユニットを用意しなければならない。本稿では、この学習すべきタスクと隠れ層のユニット数をバランスさせる方法の1つ、過負荷学習法^(9,10)を紹介する。

2. 過負荷学習法

2.1 従来の方法

学習すべきタスクと隠れ層のユニット数をバランスさせる方法としては、これまで、以下のような方法が提案されてきた。

- ・ 少ない数(例えば1つ)から徐々に隠れユニット数を増やしていき、タスクの達成度や情報量基準などで最適の隠れユニット数を求める⁽¹²⁾。
- ・ 多い目の隠れユニットを持つ回路網でタスクを学習し、学習後の隠れ層の活性度の変化やリンクの重みを解析してタスクの実現にあまり寄与していないユニットを順々に取り除いていく。
(プルーニング^(6,4,3))

これらの方法の難点は、学習を繰り返し行なわなければならない、タスクの達成度やリンクの重みの解析など複雑な操作を必要とすることである。

上であげた方法はいずれも、与えられたタスクに応じて隠れユニットの数を増減するという方法であった。次では、これと逆の考え方、つまり、タスクの方を調整して与えられた隠れユニット数とバランスさせることを考えよう。

[†] Overload Learning

* Isoki NODA

電子技術総合研究所

Electrotechnical Laboratory

2.2 タスクの統合と過負荷学習法

ここで、与えられたタスク(主タスク)と、それを学習すべき回路網(図1-(a))が与えられている場合を考えよう。この回路網には、主タスクに対し多い目の隠れユニットが与えられているとする。この回路網で主タスクを学習した場合、隠れユニットのうち一部は冗長になっている。この冗長なユニットが回路網の汎化能力に悪影響を及ぼす。さてここで、回路網に新しい入出力層(副入力層、副出力層)を加え、これらの層を使って主タスクとは別のタスク(副タスク)を学習させることを考える(図1-(b))。この際、主タスクと副タスクは同時に学習するものとする。この場合、図1-(a)の場合に冗長になっていた隠れユニットは副タスクの実現に使われるようになるだろう。そして、副タスクが適度に複雑であれば、隠れ層のユニットは無駄なく主タスク、副タスクに使われるようになる。特に、副タスクが主タスクとは完全に独立なものであれば、副タスクに使われているユニットは主タスクとは無関係になり、ないものと考えて差し支えない。よって、学習が終わった後、副タスクの部分を取り除けば、主タスクに対し、必要最小限の数の隠れユニットのみが有効に働く回路網を得ることができる。これは、前節で述べた、主タスクに使われているユニットの数と主タスクの複雑さがバランスしている状態になっているといえる。

図1 過負荷学習法

実際には、上で述べた主タスクと副タスクによる隠れユニットの使い分けはユニット毎には起こらず、隠れユニット全体の活性値パターンが張る高次元ベクトル空間の中の主成分(次元)毎に使い分けが起きる。しかし、回路網の実現能力や汎化能力の点から見れば、ユニットか成分かは問題ではなく、差し支えない。また、内部表現獲得の視点からでも、活性値パターンを主成分分析すれば、使われている成分を見つけ出すことは容易であり、さほど問題にはならない。

よって、上で述べた副タスクを統合して同時に学習する方法は、隠れ層のパターンの成分のうち、主タスクに有効に働く分を必要最小限にする方法とみなすことができる。この方法を回路網に余計な負荷(副タスク)を掛けて学習を行なわせることから、**過負荷学習法**と呼ぶ。

2.3 ランダムタスクを用いた過負荷学習法

実際には、副タスクは慎重に選ぶ必要がある。まず第1に、副タスクは十分複雑であり、主タスクと副タスクを合わせることで、回路網が過負荷な状態、すなわち、両タスクを実現するために必要な隠れユニット数が実際の隠れユニット数を上回る状態になる必要がある。副タスクが簡単過ぎる場合、冗長な隠れユニットが残ってしまい、上で述べたような効果が十分発揮されない。第2に、副タスクは主タスクとは独立であり、学習後に副タスクを切り離しても主タスクに影響が及ばないものを選ぶ必要がある。副タスクが主タスクと関連のあるものの場合、主タスクの出力が副タスクの入力に依存してしまう可能性があり、学習後の副タスクの切り離しがうまく行かない。第3に、主タスクの学習が副タスクの学習より優先されるようにコントロールできる必要がある。副タスクの学習が進み過ぎてしまうと、本来の主タスクの学習が十分達成されない可能性がある。

そこで、これらの条件を満たすため、以下のようなタスクを考える。

ランダムタスク

与えられたサイズ r (入力の次元数) に対し、 r

次元の一樣ランダムパターン*1を生成し、このパターンを入力とし、同じパターンを出力とするタスク。つまり、サイズ r のランダムパターンの恒等写像。

このランダムタスクは副タスクとして以下のような性質を持つ。

- ・入力パターンの各ユニットの値は独立に生成されるため、この恒等写像を実現するのに必要な隠れ層のサイズはランダムパターンのサイズ r に等しくなる。よって、 r を隠れ層のサイズより大きくとれば、回路網を確実に過負荷な状態にすることができる。
- ・入力パターンは独立にランダムに生成されるため、他のタスクとは必ず独立したタスクとなる。また、全ての要素を 0.5 にすることで平均パターンをつくり出すことができ、学習後、副タスクを容易に切り離すことができる。

さらに、主タスクの学習を優先させるという条件を満たすために、複合タスクを学習する際に最小化するコスト関数を、主タスクおよび副タスクにおける誤差の単純な和ではなく、以下の(1)式のように副タスクの誤差に十分小さい正の重み α を乗じた重み付き和とする。

このほか、ランダムタスクでは教師信号は入力信号と同じなので、特別に教師を用意する必要がないという利点もある。

具体的には、以下のような方法で学習を行なう。

- R.1 図 1-(b) のような回路網を考える。この回路網において 入力 層及び 出力 層は主タスクの入出力の層であり、副入力層、副出力層はランダムタスクの入出力の層である。なお、ランダムパターンのサイズ(r)は隠れ層のサイズの 2 倍とする。
- R.2 学習時においては、まず主タスクの入力パターンを入力層に入力し、出力の目標パターンを出力層に教師パターンとして与える。同時に、ランダムパターンを副入力層に入力し*2、

それと同じパターンを副出力層に教師パターンとして与える。これらの教師パターンを用いて回路網は以下の誤差関数を極小化するように誤差逆伝搬(BP)学習を行なう。

$$E = \langle |o_{\text{主出力}} - t_{\text{主出力}}|^2 \rangle + \alpha \langle |o_{\text{副出力}} - o_{\text{副入力}}|^2 \rangle \quad (1)$$

ただし、 $o_{\text{主出力}}$ 、 $o_{\text{副出力}}$ 及び $o_{\text{副入力}}$ は各々出力層、副出力層及び副入力 層の活性パターンベクトルであり、 $t_{\text{主出力}}$ は主タスクの教師パターンである。また、 $\langle x \rangle$ は x の平均を、 $|x|$ はベクトル x のノルムを表し、 α は正の定数である。

- R.3 学習後は副出力層は隠れ層から切り離し、副入力層の全てのユニットの出力値を 0.5 に固定する。

3. 適用例

では、過負荷学習法の効果をいくつかの実験によって示してみよう。

3.1 空間的内部表現の獲得と汎化

タスクが空間に関するものの場合、内部表現としてはその空間の位相を反映したものであることが望ましいと考えられる。そこで、以下のように 3 次元空間に関するタスクを考え、それを過負荷学習法により回路網に学習させる実験を行なってみた。

まず、ある立方体を考え、その立方体の各頂点(apex)、辺(edge)、面(plane)に図 2 のようにラベルをふる。そして辺のラベルを入力し、その辺に接する 2 頂点、2 面のラベルを出力するというマッピングを考える。このマッピングの入出力は記号(ラベル)であるので、これを局所表現でパターンで表した図 3 のようなマッピングに変換する。このマッピングを回路網で学習させ、どのような

*1 各次元の値を $[0,1]$ の一樣分布で独立に生成したパターン。

*2 ランダムパターンは入力を行う毎に新たに生成する。

xx1, x0x, 1xx の反対側は xx0, x1x, 0xx

図2 ラベルづけされた立方体の辺、頂点、面

ただし、Plane層左より0xx、x0x、xx0、xx1、x1x、1xx、

Apex層左より000、001、010、011、100、101、110、111

図3 辺から面、頂点へのマッピング

内部表現を獲得するかみてみよう。

図3を見てわかるように、このマッピングは一見、元の立方体の3次元的な情報は失われているように見える。よって、これを単なる記号のマッピングとして学習してしまうと、空間的な関係を使った汎化を行なうことができないであろう。それに対し、学習により空間的な内部表現を作り上げることができれば、空間の連続性を使った汎化を行なうことができ、未知の入力に対しても正しく反応できると考えられる。

実際に図4のような回路網(Edge層12ユニット、隠れ層12ユニット、Plane層6ユニット、Apex層8ユニット、副入力、副出力層24ユニット)を用いて学習を行なわせた。ただし、汎化能力をテストするため、図3のうち、下線のついた部分の出力に関する教師信号は与えずに*3学習を行なった。

*3 教師信号を与えられない場合には、その出力ユニットで生じる誤差を学習のために用いない。すなわち、(1)式の誤差の計算に含めない。

図4 辺、面、頂点のマッピングを学習する回路網

(a) 過負荷学習法

(b) 従来の学習法

図5 辺、頂点、面のマッピングの学習結果

学習後の回路網の出力を図5に示す。この図の(a)は過負荷学習法を行なった回路網の結果であり、(b)は比較のため、副タスクの学習を行なわない従来の学習を行なった回路網の結果である。これによると、従来の学習の場合、教師信号が与えられなかった部分の出力に関してはでたらめな出力をしている。つまり、マッピングを単なる記号のマッピングとして学習したため、空間的な推論ができていないことになる。これに対し、過負荷学習法の場合は未知の入力に対し正しい出力をしている。これは、学習により空間的な関係を表す適

切な内部表現を獲得し、それに基づいて汎化が行なわれたと考えられる。

ここで、この学習において回路網が獲得した内部表現を調べてみよう。この内部表現を視覚化するため、学習後の回路網に各辺のラベルを入力した時の隠れ層のパターンを記録し、それらのパターンの主成分分析を行ない、各パターンの第1、第2、第3主成分をとってプロットした(図6-(a))。これを見ると分かるように、各辺のラベルが入力された時の隠れ層のパターンの配置は、隠れ層のパターン空間の中で元の立方体の各辺の中点の配置(図7)になっていることが分かる。それに対し、従来の学習の場合の隠れ層のパターンの配置は図6-(b)のようになっており、(a)の場合のような立方体との対応は見受けられない。

ここで重要な点は、タスクの表面上の表現である図3のマッピングには、もとのモデルが3次元であることが明示的に表されていないことである。すなわち、過負荷学習法の効果により隠れ層のパターン表現から冗長な成分をなくすことで図3のマッピングに内在する3次元で表される関係を抜きだし、それに適した内部表現を学習により獲得したと考えられる。そして、未知の出力についてもその表現に基づいて正しい値を推定できたと見ることができる。

3.2 特徴抽出とセンサーフュージョン

3.2.1 砂時計型回路網と恒等写像学習

では次に、内部表現の獲得、特に特徴抽出に焦点を当ててみよう。特徴抽出を行なう回路網としては、砂時計型回路網^(7,8)が良く用いられるが、この回路網においても隠れ層のユニットの数をどのように決めるかが問題となる。そこで、過負荷学習法をこの砂時計型回路網に適用してみよう。

図8のような回路網を考える。この回路網のうち、主入力層、主出力層は主タスク(目的のパターンの恒等写像)の学習を行ない、副入力層、副出力層は副タスク(ランダムタスク)の学習を行なう。そして、回路網は全体として以下のコストを極小化するようBP学習を行なう。

図6 各辺の隠れ層における内部表現

図7 辺の中点の配置

図8 砂時計型回路網への適用

$$E = \langle |o_{\text{主出力}} - o_{\text{主入力}}|^2 \rangle + \alpha \langle |o_{\text{副出力}} - o_{\text{副入力}}|^2 \rangle \quad (2)$$

ただし、 $o_{\text{主出力}}$ 、 $o_{\text{主入力}}$ 、 $o_{\text{副出力}}$ および $o_{\text{副入力}}$ は各々、主出力層、主入力層、副出力層、副入力層の活性パターンである。前節で述べているように、この回路網を用いれば内部表現層における主タスク(目的の恒等写像)に使われる主成分の数は必要な大きさに抑えられ、簡潔な内部表現が得られると考えられる。

3.2.2 ステレオ視差画像の特徴抽出

この回路網の効果を試すため、ステレオ視差のある視覚パターンの恒等写像の学習を行なった。

学習させる恒等写像の入力パターンとしては図9に示すようなものを考える。すなわち、2組のユニット列(長さ10)を考え(両眼の網膜に対応)、長さ l の像が左目では p 、右目では $p+d$ の位置に映っているとする。そして、位置 p 、長さ l 、ずれ d (奥行き)のパラメータを各々 $p=0\sim 9$ 、 $l=1\sim 4$ 、 $d=-2\sim 2$ の範囲で変化させ、パターン対を生成する。このパターン対を1つにまとめ、恒等写像の値とする。これを図10に示す回路網を用いて学習を行なわせた。ただし、各層のユニット数は主入出力層は20、副入出力層は40、内部表現層は20、各隠れ層は40である。

学習後、副タスクを取り除いて、上の実験と同じく内部表現層のパターンの主成分分析を行なった。その結果、回路網は主タスクを実現するために、内部表現層のパターン空間の3つの主成分のみを用いていることがわかった。つまり、回路網は視差のあるパターン情報の内部表現として3次元の表現を獲得していることがわかる。この次元

図9 ステレオ視差パターン

図10 ステレオ視差パターン用回路網

数はパターンを生成する際の独立パラメータ(p, l, d)の数に一致する。さらに、各視差パターンに対する内部表現を解析すると、図11、図12のようになる。図11は、像の長さ(l)別に内部表現のパターン空間内での位置をプロットしたものを示している。これを見るとわかるように、パラメータ l および p は内部表現では極座標に近い形で表現されていることがわかる。すなわち、パラメータ l は極座標の半径として、パラメータ p は角度として表現されている。一方、図12は、パラメータ l を3に固定した場合の、各パターンの内部表現をパラメータ d ごとにプロットしたものを示している。これを見るとわかるように、このずれのパラメータ d は解析しやすい形では表現されていないことがわかる。

以上のように、過負荷学習法は砂時計型回路網による恒等写像学習にも容易に適用することができる。そしてその結果、与えられたパターンの自由度に応じた数の主成分を持つ内部表現を獲得できることがわかる。ただし、パラメータ d の解析

図11 長さ及び位置の内部表現

図12 ずれの内部表現($l=3$)

結果のように、成分数が最少化されても必ずしも人間が期待するような表現にならないこともある。

3.2.3 センサーフュージョン

上の実験のように、学習している自由度がわかるだけではあまり面白くない。そこで、この実験にすこし改良を加え、ロボットのセンサーフュージョンにあたる実験を試みよう。

まず、上の実験で用いたステレオ視差パターンにパラメータ d に関する補助的な情報を示すビットを付け加えたパターンを考える。このビットはパラメータ d が -1 あるいは -2 の時のみオンになるものである。これはたとえば近くにある物体はタッチセンサーなど視覚以外からの補助的情報が得られるということを想定している。このパターンを用いて、前の実験と同様の回路網で恒等写像の学習を行なわせた。

学習後の内部表現パターンの主成分分析の結果、この設定の場合も実験4の場合と同様、内部表現は3主成分で表現されていることが分かった。さらに、この場合の図12に相当する表現を調べると、図13のようなものが得られる。これを見るとわかるように、前の実験では交差していた線が整列していることがわかる。これは、パラメータ d が内部表現の中で独立した次元として表わされていることを示している。ここで重要な点は、この実験で導入したビット情報はずれの情報を2段階(ビットのオン/オフ)のみで表現しているにも関わらず、得られた内部表現ではもとの5段階($d=-2\sim+2$)の情報が全て整列されていることである。

図13 補助データがあった場合のずれの内部表現 ($l=3$)

この実験の意義は、ステレオ画像とタッチセンサーという異種の入力から、統合された簡潔な表現を獲得しているという点にある。この統合的表現の獲得により、以下のようなことが期待できる。

- ・複数のセンサー入力を統合することで、各センサーの値に重畳する雑音などに対し頑健な表現が得られる。
 - ・多数のタイプの入力を統合することで、もとの情報源の構造をよく反映した表現が得られる。
- また、過負荷学習法を用いた恒等写像学習では、複雑な前処理などを経ずに、自動的に表現が獲得されることも重要である。今後、ロボットに搭載されるセンサーの種類は増加すると考えられるが、ここで用いた方法を用いれば、それらのセンサーを区別なく容易に統合していくことが可能になるであろう。

3.3 再帰結合回路網への適用

前節での例では、過負荷学習法を5層の回路網に適用した。このように過負荷学習法は回路網の構造や学習法にあまり依存せずに適用できるという利点がある。ここではこの利点の典型例として、再帰結合回路網への適用を示そう。

再帰結合回路網にはいろいろなタイプのものがあるが、ここでは Elman⁽²⁾により提案された単純再帰結合回路網を取り上げる。この回路網は記号処理における有限オートマトンと類似した構造を持っており、逐次処理や時系列処理を実現、学習、解析するモデルとして良く用いられる。この場合、学習後に得られた隠れ層の活性パターンの遷移を解析する必要があるが、隠れ層のユニットが多過ぎるとこの解析はかなり困難なものとなる。例えば、Cleeremans らの研究^(1,5)で用いられた Reber 文法を獲得する実験を隠れユニットを30個用いて行なうと、学習後の活性パターンの分布として図16-(b)のような結果が得られる。この図ではもとの Reber 文法の各状態に相当するクラスターをある程度見ることはできるが、これらのクラスターはかなり複雑に絡み合っており、自動的に解析することはかなり難しい。ここで、単純再帰結

合回路網に過負荷学習法を適用した回路網(図14)に同じタスクを行なわせると、学習後図16-(a)のような活性パターンの分布が得られる。ここでは、(b)の場合に比べかなり少ない数の点しかプロットしていないように見えるが、実際には同じ数の点をプロットしており、クラスタがほとんど1点に収束してしまっているのがこのように見える。これは、過負荷学習法の効果によりパターン遷移の表現が簡潔され、それによりこのような簡潔な表現が得られたと見ることができる。

図14 過負荷学習法を適用した単純再帰結合神経回路網

図15 Reber文法

図16 Reber文法学習後の各状態のパターンの分布：
(グラフの軸はパターンの第1、第2、第5主成分)

このように過負荷学習法は再帰結合を持つ回路網にもなんの変更もなく適用することができる。再帰結合回路網では隠れ層のパターンが次の時刻の回路網の入力となるため、プルーニングのように隠れ層を人為的に操作してしまう方法は適用することが難しい。それに対し過負荷学習法では、隠れ層のパターン表現の簡単化は通常の学習を通じて行なわれるので、自動的に簡潔な表現を獲得することができる。再帰結合回路網以外にも、本方法は Boltzman machine や連続時間の処理を行なう回路網など、回路網のモデルや学習法などに限定されず適用することができると考えられる。

4. おわりに

多層神経回路網の隠れユニットの数と回路網が学習するタスクの複雑さをバランスする手法の1つ、過負荷学習法について解説した。この方法を用いれば、隠れ層に形成される内部表現が簡潔になり、その結果、学習の汎化能力を改善したり、学習したタスクの性質の解析を容易にしたりする効果がある。

本方法の特徴は、手順が単純であり、さまざまな構造の回路網や学習法に適用できることである。この方法では回路上に新たな層を加える必要はないので、広範囲の回路網のモデルや学習法に適用することができる。さらに、この単純さはハードウェア上に本方法を実現する上でも有利な点であると考えられる。

本方法の定量的な解析についてはまだ十分行なわれていない⁽¹¹⁾。大規模な応用に適用する上では、この解析は必要となってくるであろう。

参 考 文 献

- (1) Axel Cleeremans, David Servan-Schreiber, and James L. McClelland. Finite state automata and simple recurrent networks. *Neural Computation*, Vol.1, pp.372-381, 1989.
- (2) Jeffrey L. Elman. Finding structure in time.

- Technical Report CRL-TR-8801, Center for Research in Language, University of California, San Diego, April 1988.
- (3) Christian W. Omlin and C. Lee Giles. Pruning recurrent neural networks for improved generalization performance. Technical Report No. 93-6, Computer Science Department, Rensselaer Polytechnic Institute, Troy, N.T., April 1993.
- (4) Russell Reed. Pruning algorithms — a survey. *IEEE Trans. on Neural Networks*, Vol.4, No.5, pp.740-747, Sep. 1993.
- (5) D. Servan-Schreiber, A. Cleeremans, and J. L. McClelland. Graded state machines: The representation of temporal contingencies in simple recurrent networks. *Machine Learning*, Vol.7, No.2-3, pp.57-90, Sep. 1991.
- (6) J. Sietsma and R.J.F. Dow. Neural net pruning — why and how. In *Proc. of IEEE International Conference on Neural Networks*, pp.I-325--I-333, Jul. 1988.
- (7) 臼井支朗, 中内茂樹, 中野正恵. 多層ニューラルネット恒等写像モデルによるマンセル色空間の特徴解析. 技術研究報告 NC89-40, 電子情報通信学会, 12 1989.
- (8) 入江文平, 川人光男. 多層パーセプトロンによる内部表現の獲得 — テーブルルックアップ法と違うのか? —. 技術研究報告 NC89-15, 電子情報通信学会, 1989.
- (9) 野田五十樹. 過負荷学習法. 技術研究報告 NC93-28, 電子情報通信学会, 7月 1993.
- (10) 野田五十樹. 過負荷学習を用いた恒等写像学習による内部表現獲得. 神経回路学会第5回全国大会講演論文集, pp.80-81, 11月 1994.
- (11) 野田五十樹. 恒等写像学習における過負荷学習法の性質 — 線形回路網の場合 —. 技術研究報告 NC95-113, 電子情報通信学会, 2月 1996.
- (12) 和田安弘, 川人光男. 新しい情報量基準による汎化能力を持った神経回路網モデルの選択. 技術研究報告 NC90-20, 電子情報通信学会, 7 1990.

(1997年6月24日 受付)

[問い合わせ先]

〒305 つくば市梅園1-1-4
 電子技術総合研究所 情報科学部
 野田 五十樹
 TEL : 0298-54-5846
 FAX : 0298-51-9743
 E-mail: noda@etl.go.jp

—— 著 者 紹 介 ——

野田 五十樹 (のだ いそき)

電子技術総合研究所 情報科学部

1987年 京都大学工学部電気工学第2学科卒業、1992年 同大学院博士課程修了。同年4月 通商産業省工業技術院電子技術総合研究所入所。現在、主任研究員。神経回路網、記号・パターン処理の統合およびマルチエージェントシステムなどの研究に従事。博士(工学)。