ニューロコンピューティングと情報論的学習理論

赤穂昭太郎
産業技術総合研究所 脳神経情報研究部門

1 はじめに

1980 年代にニューロコンピュータのブームが訪れた。そのきっかけは、神経回路のモデルである多層パーセプトロンとその学習アルゴリズムであるバックプロパゲーションが提案（正確に言えば再発見）されたことによる。多層パーセプトロンは十分たくさんの中間層の素子数を用意してやれば任意の入出力関係を表現可能であることも示された。そこで多くの人は巨大なネットワークを作って環境の中に置いてやれば人間並みに学習ができ、さらに大げさに言えば脳の学習の基本原理もすべて解明されるかのような幻想を描いた。

だが実際には期待したようにはならず、ピーク時のようなブームは去っていった。しかし全く研究分野が廃れてしまったかというとそんなことはない。ニューラルネットもうまく使ってやれば非常に役に立つという実験的な事実は積み上がっており、いろいろなタイプのニューラルネットやそこから派生した数々のモデルについて、どのような性能をもっているかがさまざまな立場から研究されるようになった

ニューラルネットワークに特有の学問分野というのはなかったので、計算機科学、統計学、統計物理といった分野の研究者が参入してきてニューロコンピューティングという横断的な分野を生み出し現在に至っている。
ブームは去ったが中身の方は逆にどんどん面白くなってきている。

ニューロコンピューティングではいわゆる神経回路に留まらず、より一般化された形で「学習」という問題を扱うようになり、人工知能の一分野である機械学習との交流も著しい。これらの分野統合的な流れは情報論的学習理論に至る一つの系譜とも考えられる。

最近ではいわゆるニューラルネットと呼ばれるモデルに関する論文発表を見ることも少なくなり、あるニューロコンピューティングの国際会議では、ニューラルネットというキーワードは不採択論文のキーワードの最上位に来るという話である。しかしながら、本稿では、あえて神経素子モデルからスタートして、中間層の素子数選択という問題を中心に情報論的学習理論に至る系譜をたどることで、この分野でどのような研究が行なわれているかの一端を紹介したい。

2 ニューラルネットとは

まずニューラルネットとは何かを簡単に説明しておく。脳を構成する神経素子のモデルとして最も単純なのは、ほかのいくつかの素子からの入力信号 x_i が w_i という重みつきで加算され、その値 $\sum_i w_i x_i$ が、あるしきい値 h を超えると発火し (1 を出力), そうでなければ 0 を出力するという変換器 (線形しきい素子) である。神経素子は与えられた環境に応じて、重み w_i やしきい値 h といったパラメータを変化させることによって環境を「学習」する。もちろん、実際の神経細胞は複雑なダイナミクスをもち、そうしたモデルもいろいろと考えられているが、線形しきい素子だけでも十分複雑なことが起きるし、わかっていないことも多い。

さて、線形しきい素子をつなげばニューラルネットができるわけだが、フィードバックのない層状の回路にすればこれは単に入力から出力への変換回路として働く。一方、相互結合などを考え、フィードバックをもつ系と考えると、非線形のダイナミクスが入るので、面白い性質が山ほど出てくるのだが、そちらは統計物理が力を発揮する領域なので岡田氏による解説などを参照していた

だくことにして、本稿ではフィードバックがない場合についてだけ考える。

3　中間層の素子数を決める問題

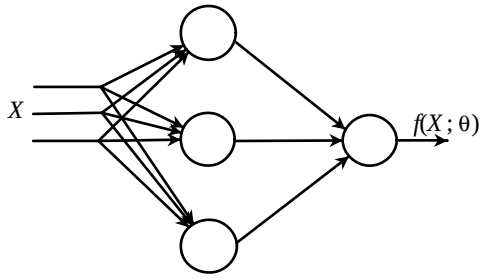

図 1: 3 層ニューラルネット

図１は層状の回路の例であり、入力をいったん中間層と呼ばれる層に属する素子群によって変換し、それをさらに出力層で変換したモデルで、3 層の (フィードフォワード型) ニューラルネットと呼ばれている。ここで、学習するパラメータ全体を θ で表し、与えられた入力ベクトル X に対する出力を $f(X;\theta)$ と書くことにする。以下話を単純にするために、2 クラスのパターン識別の問題を考える。例えば、食べ物の画像を入力としてそれがうまいかまずいかを判定するのが課題である。 この場合は入力の次元は画像の次元数、出力は２値出力の素子が１つだけである。このとき、中間層の素子の数は自由に選べることになるが、それではいくつに取るのが最適だろうかという疑問がわく。

できるだけ識別能力の高いモデルを作ろうと思えば、中間層の素子の数はできるだけ多いほうがいいように思える。　ところが、ニューラルネットは環境から得られる情報、上の例で言えば、いくつかの食べ物の画像とそれらがうまいかまずいかという訓練データだけが与えられ、それに基づいて学習を行なわねばならない。学習の結果として、訓練データだけに正しく答えるだけではだめで、未知の食べ物についてもきちんと判定を下して欲しい。このような、有限個のものだけから背後の構造をどこまで推定できるかという能力を汎化能力といい、90 年代の学習理論研究の多くが、汎化能力に関する研究であるといってもよい。

中間層の素子の数と汎化能力の間には果たしてどのような関係があるのだろうか。それは次の節で詳しく述べることにするが、本来モデルの選択に関わるのは汎化能力だけではない。 中間層の素子数を増やしていけばそれだけ素子資源を消費することになるし、学習にかかる時間も素子数によって変動すると考えられる。このような領域・時間計算量は問題によって制約を受けることもあり、モデルの選択に影響する。例えば明日の天気予報をするために学習結果が出るのが明後日では困ることになる。ただし、ニューラルネットのように学習にどれだけ時間がかかるのかを見積もるのが難しいモデルでは、モデルの選択に直接反映させるのは現在のところ困難である。従って、本稿では主として汎化能力だけに焦点を当てて考える。

4　学習における汎化能力

学習というのは、環境の一部分の情報である訓練データだけを用いて全体を推定しようという一種の逆問題なので、汎化能力を定量的に評価してやるためには環境やそこからデータを生成するメカニズムについて前提知識を入れる必要がある。前提知識の入れ方にはいろいろなレベルがあり、扱う問題によって違ってくるが、情報論的学習理論に関わるすべての分野で共通しているのは、訓練データが未知の確率分布から生成されるというものである。　これによって、汎化能力を理論的かつ定量的に評価することが格段に易しくなる。ここでは簡単のためデータはすべて同じ（未知の）確率分布 $P(X,Y)$ から独立して生成されるとする。学習に使えるのはそのように生成された N 組の訓練データ $(X_1,Y_1),\ldots,(X_N,Y_N)$ だけである。

ニューラルネットの場合、中間層の素子数を増やせば増やすほど複雑な入出力関係を表現できるので、与えられた訓練データにうまく適合することができる。ところがその分訓練データにだけ適合し、新規データにはうまく適合できない入出力関係を学習してしまう可能性も

高くなる。とすると直感的には、あまり多すぎもせず少なすぎもしない中間素子数というのがあると考えられるが、果たして本当にそうなのかを理論的に明らかにする必要があるし、またそれを決めるための規準が必要となる。詳細についてはすでに優れた解説 [4, 6] があるので、ここではそれらの流れを大雑把に紹介するに留める。

5 学習の目的

まず、汎化能力を評価するためにはそもそも学習が何を目的とするかを定めなければならない。通常は入力 X をパラメータ θ で処理したときに実は正解出力が Y だったとしたときの損失関数 $Q(X, Y, \theta)$ を定め、それをできるだけ小さくするという方針で学習が行なわれる。前にあげた識別の例で言えば、正しくうまいかまずいかを判定できれば 0 そうでなければ 1 とする 0-1 ロス

$$Q(X, Y, \theta) = (Y - f(X; \theta))^2$$

を用いるのが最も単純である。 汎化能力という観点から本当に見つけたいものは、未知のデータに関する損失関数の期待値（期待損失）

$$Q_{\exp}(\theta) = E_{X,Y}[Q(X, Y, \theta)]$$

を最小にするパラメータ θ^* である。　しかしながら、実際に学習できるのは、訓練データに関する損失（経験損失）

$$Q_{\mathrm{emp}}(\theta) = \frac{1}{N} \sum_{i=1}^{n} Q(X_i, Y_i, \theta)$$

を最小にするパラメータ $\hat{\theta}$ を見つけることだけである。汎化能力の理論では期待損失の最小値 $Q_{\exp}(\theta^*)$ と経験損失の最小値 $Q_{\mathrm{emp}}(\hat{\theta})$ がどれくらいずれているのかをなんらかの意味で評価することが主眼となる。

5.1 PAC による規準

汎化誤差を評価する一つの枠組みは「PAC (probabilistically approximately correct) 学習」と呼ばれるもので、その名が示すとおり、確率的にだいたい正しい汎化をしたいという考え方に基づく。　具体的には、期待損失と経験損失の差がある値以上になる確率を評価する。この確率は一種の大数の法則により上から押さえることができる。そこから例えば、確率 $1 - \delta$ 以上で

$$Q_{\exp}(\theta^*) < Q_{\mathrm{emp}}(\hat{\theta}) + G(C_f, \delta, N)$$

という形の不等式が成り立つことが示される。ここで C_f は、ニューラルネットを一つ決めたときに入出力関数がどれぐらい表現能力をもっているかという関数クラスの複雑度をあらわす指標であり、様々なバリエーションがあるが、最も基本的なものは VC (Vapnik-Chervonenkis) 次元と呼ばれもので、関数クラスだけから決まる値である。

中間層の素子数に応じた C_f がわかれば、上記の不等式の右辺を最小にするようなモデルを選ぶことで最適な素子数を選ぶことができると考えることができる。ただし、ここで注意しなければならないのは、この不等式は確率 $1 - \delta$「以上」で成り立つ、つまり、右辺の値は $1 - \delta$ 確率点に対するかなり粗い上限に過ぎないということである。そのため、必要以上に小さい関数クラスが選ばれることが多く、現在では直接その目的で用いられることは少ない。

PAC 学習では確率を荒っぽく不等式で押さえていくため、結果として出てくる確率値そのものは実際の値に比べてかなり粗くなってしまうのは致し方ない。それでも、不等式の押さえ方を工夫したり、C_f として VC 次元の代わりに訓練データに依存した複雑度を用いたりして不等式の評価を改善する試みは計算論的学習理論と呼ばれる分野を中心に精力的に行なわれている。

5.2 AIC と MDL

PAC では期待損失と経験損失の差の確率分布を評価しようとしたためモデルを選ぶ規準も粗いものしか得られなかった。一方、複数のモデルから一つを選ぶという規準で、PAC 学習などよりもよく知られているのは AIC (赤池の情報量規準) や MDL (記述長最小化原理) であろう。これらは、それぞれ統計学と情報理論の分野で導出されたモデル選択のための規準である。

AIC も MDL もその出発点は学習を統計的推定の問題として扱うところにある。PAC 学習では (X, Y) の分

布 $P(X,Y)$ とニューラルネットのパラメータ θ とは期待損失の計算で陰に結ばれていただけであるが、統計的推定ではパラメータ θ によって規定される確率モデル $p(X,Y;\theta)$ を考え、$P(X,Y)$ を $p(X,Y;\theta)$ で推定するという形式をとる。

この枠組みでの基本的な推定法として最尤推定法を考えよう。すなわち与えられた訓練データを $p(X,Y;\theta)$ に代入して得られる値を θ のもっともらしさ (尤度) と見て、それを最大にするような θ を見つけるというものである。この場合、損失関数として負の対数尤度 $-\log p(X,Y;\theta)$ を取れば損失関数最小化問題に帰着させることができる。

さて、確率モデルとして正則条件を満たすような素直なものを取れば、最尤推定では訓練データの数が多くなるにつれてパラメータの推定値 $\hat{\theta}$ の分布を正規分布とみなして評価できる。　この事実と、経験損失 $Q_{\rm emp}(\hat{\theta})$ を θ^* の周りで Taylor 展開した式とを用いて期待損失と経験損失の差の期待値が評価できる。従って、経験損失にその差を足してやれば期待損失の期待値がでるので、それをモデル選択基準として使ってやろうというのが AIC の基本的な考え方である。もし、未知の分布 $P(X,Y)$ がモデルの分布 $p(X,Y;\theta)$ に含まれていると仮定できれば、その値はパラメータ数に比例し、よく知られた AIC が導出される。

一方 MDL では $P(X,Y)$ で生成されたデータを伝送するときの符号長を考える。その際に、$P(X,Y)$ を確率モデル $p(X,Y;\theta)$ で最尤推定した上で、推定パラメータと $p(X,Y;\hat{\theta})$ で符号化したデータを伝送する 2 段階符号化というのを行なう。この符号化の符号長を最小にするモデルを選ぶのが MDL である。MDL と漸近的に等価な規準は統計的推定の立場からはベイズ推定からも導くことができるので、統計の分野では BIC (Bayes 情報量規準) として参照されることもある。

PAC, AIC, MDL は、モデル選択基準として図るものさしが異なっており、これは目的に応じて使い分けるべきで、一概にどれがいいということは言えない。 さらに、AIC や MDL では PAC に比べていろいろな仮定を付加しているが、それらを除いていく一般化もいろいろ研究されており、必要に応じて使える状況になってきている。 ただし、一般化していくと複雑な形になったり、PAC のように上限などの形でしか表現できなくなっていくので、やたらと一般化されたものを使えばよいというものではないことに注意する必要がある。

5.3　中間層の素子は少ないほどよいか

さて、PAC にしろ、AIC や MDL にしろ、基本的にはモデル選択基準は訓練データに関する誤差にモデルの複雑さに関わる項が加わった形をしている。すると中間層の素子数を選ぶ際には、複雑さは中間層の素子数の増加に伴って増えていくので訓練データに関する誤差が同程度なら、より素子数を少なくしたほうがよいと考えられる。このような規準は「けちの原理」とか呼ばれることもあるし、“Simple is best” という格言にも合致するものであり、モデル選択における標準的な原理として認知されている。

ただしここでは必ずしもそうではないという話をする。一つは特に AIC や MDL の場合に関係する特異性に起因する問題である。これに関しての詳細は甘利氏の解説を参照していただきたいが、階層型のニューラルネットのように階層構造を持つモデルでは中間層の素子数が冗長に存在する場合には AIC や MDL で仮定している正則条件が成り立たなくなるため不思議なことがいろいろと起きる。筆者らは、正規混合分布という階層モデルにおいて、パラメータ数は増えるのに汎化能力が逆に向上するという、上の「標準原理」に反する場合があることを示した (Akaho et al., 2000)。

もう一つは、関数のクラスを制限する方法は中間層の数だけではないので、中間層をいくら増やしても大丈夫という場合がある。その例としてまずベイズ推定を挙げよう。　ベイズ推定ではパラメータ θ も確率変数と考え、あらかじめどの様な分布に従うか (事前分布) がわかっているとして、訓練データが与えられたという条件下での θ の分布 (事後分布) を用いてパラメータに関する推論をする。事前分布についてわかっていれば事後分布は本当に θ が従う分布になっている。また、適当な事前分布 (無情報事前分布というものもある) を使ったとしても、パラメータの取りうる範囲に確率的な重み付けをしているので、事前分布が結果的に AIC や MDL で

行なっているような罰金項の役目をしてくれるため、たくさんの素子数を使ったモデルでも汎化能力がそれほど問題にならなくなる。　ベイズ的モデル選択についての詳細は上田氏の解説を参照されたい。

本稿では中間層の素子数をたくさん使っても大丈夫というもう一つの例としてサポートベクターマシンという識別のための学習モデルを取り上げる。

6　サポートベクターマシン (SVM)

サポートベクターマシン (SVM) ではまず入力を非常に大きい空間へ (非線形に) 写像し、その上で線形関数の識別を行なうと考える。これは中間層の素子数が非常に大きい 3 層ニューラルネットと似ていて、違いは入力から中間層の出力への変換は学習せず固定されていることと、必ずしもニューラルネットで用いられるような線形しきい素子でなくてもよいことである。ただし、非常に大きい空間への写像は領域計算量が大きく、最初のほうにも書いたように学習モデルとしては問題があるため、SVM ではカーネル法といわれるトリックを使う。その説明をするためにまず、入力 X に対する i 番目の中間層の素子の出力への変換を $\phi_i(X)$ と書くと、SVM では中間層から出力への重み係数が $\sum_j a_j\phi_i(X_j)$ という訓練データの変換値の重みつきの和になることが示せるので、最終的な出力素子への入力は $\sum_j a_j \sum_i \phi_i(X_j)\phi_i(X)$ という形に書ける。関数解析で出てくる Mercer の定理というのを使うと、正定値であるような対称関数 $k(U,V)$ を持ってくれれば必ず $\sum_i \phi_i(U)\phi_i(V)$ なる ϕ_i が存在することが言えるので、ϕ_i を計算しなくても $k(U,V)$ (カーネル関数という) を使えば一撃にして大きい空間の内積が計算できてしまうことになる。そのかわり簡単に計算できる $k(U,V)$ というのを先に決めてしまうので ϕ_i は存在することだけがわかっているだけという不思議なことになる。　これがトリックと呼ばれるゆえんである。

さて、ϕ_i の内積をカーネルで置き換えた出力素子への入力 $\sum_j a_j k(X_j,X)$ の形を見ると先ほどとは違って、$k(X_j,X)$ を中間層の j 番目の出力とするような 3 層のニューラルネットに近い形になることがわかるであろう。特に、$k(U,V) = 1/(1+\exp(-U\cdot V))$ という正定値カーネル関数を使うと、中間層のしきい値関数を 0, 1 の間でなめらかにつないだもの (シグモイド関数) になっていて、これはバックプロパゲーションなどで用いられるものと同じものである。$k(U,V)$ で置き換えた式でも見かけ上訓練サンプルと同じ個数の和の形になっているが、実は a_j の多くは 0 になることが多く、0 でない a_j に対する少数の和が実質的に残る。従って、領域計算量は解消され、これが SVM の売りの一つになっている。

前の節で述べた議論からすると、このままでは非常に高い次元の空間での識別であるから、汎化能力は全く劣ったものだということになる。それを防ぐために SVM では ϕ_i で変換された入力を単に識別するだけではなくてマージンと呼ばれる量ができるだけ大きくなるように最適化される。マージンというのは線形識別をする境界面と訓練データの間の距離で、直感的には境界面からできるだけ訓練データが離れていたほうが多少訓練データにノイズが加わっても間違えないわけだから汎化能力は高いだろうという予想はたつ。　これは PAC 学習の観点から理論的に示されていて、マージンを最大にする識別関数の VC 次元はマージンの大きさに反比例し、次元には依存しないことが示されている。つまり、いくら高い次元の空間に写像しようとも、期待損失の上限値である PAC 学習のモデル選択基準は発散しない保証を与えている。　粗い評価しかできない PAC であるが、この場合はモデル選択には直接使われず、SVM の性能保証の傍証として役に立っている。

7　むすび

ニューラルネットを題材として情報論的学習理論の一つの主要なテーマであるモデル選択について概略を紹介してきた。ニューロコンピューティングの分野はもともと脳に倣った情報処理機構を実現したい、あるいは脳の情報処理の原理を解明したいという動機から生じた分野である。本稿ではこの点にはあまり触れられなかったが、脳情報処理に関する情報論的な取り組みの最新動向 (独立成分分析など) については文献 [2, 3, 5] などが参考になる。

さて、モデル選択の問題は当然ながらニューラルネッ

トに限らず、構造の自由度のあるモデルでは必ず起きる問題であり、研究の量も莫大なので本稿で示したのは非常に単純で古典的といってよいものに過ぎない。また、本稿では主として汎化能力の観点からよいモデルというのを考察したが、モデルのよさは学習アルゴリズムや計算量によっても規定される複雑な問題である。３層ニューラルネットとバックプロパゲーション、SVM と２次計画法、隠れマルコフモデルと EM アルゴリズム、グラフィカルモデルとビリーフプロパゲーション等の例を挙げればわかるように、モデルと効率的なアルゴリズムの組み合わせがあることは重要である。さらに、実際の応用問題からもモデルの構造や学習アルゴリズムの速さに対する要請が出てくる。このようにモデルと学習アルゴリズム、さらには応用問題までもが完全には切り離せないところに情報論的学習理論研究の面白さと難しさがある。

情報論的学習理論自体、個々の特定の分野とそれを統合する一般論というトレードオフの側面がある。例えばベイズの枠組みなどはかなりの範囲をカバーする一般的なものであるが、単にそれで記述できたというだけではあまり実りがない。本当に面白くて生産的なのは、個々の分野で培われた技を駆使して導いた結果である。 ただし、既存の分野におけるやり方を辿るだけではなく、そのような技を生み出すことのできる新たな分野の創造に結びつくようなことがあればすばらしいと思う。

参考文献

[1] S. Akaho and H.J.Kappen: Nonmonotonic generalization bias of Gaussian Mixture Models, *Neural Computation*, 12(6), 1411–1426 (2000)

[2] 甘利ほか: 脳情報数理科学の展開、別冊・数理科学、サイエンス社 (2002)

[3] 甘利、村田 (編著): 独立成分分析、サイエンス社 (2002)

[4] 麻生: 情報論的学習理論、人工知能学会誌, 16(2), 287–299 (2001)

[5] 石井、前田: 脳における予測と推定の仕組み、Computer Today, No. 110, 16–23 (2002)

[6] 山西: 情報論的学習理論の展望、情報処理, 42(1), 9–15 (2001)