

## ニューラルネットツリーの汎化能力に関する一考察

林 博友<sup>†1</sup>会津大学 大学院  
コンピュータ理工学研究科  
d8092103@u-aizu.ac.jp趙 強福<sup>†2</sup>会津大学  
コンピュータ理工学部  
qf-zhao@u-aizu.ac.jp

## 1 はじめに

ニューラルネットツリー (NNTree: Neural Network Tree) とは、判別木とニューラルネットワーク (NN: Neural Network) のハイブリッドモデルである。その構造は図 1 に示すように、判別木の各中間ノードにそれぞれ NN を埋め込んだ形となる。各 NN は各中間ノードでの局所判別にのみ使用され、NNTree の判別結果は普通の判別木と同様に終端ノードに割当てられたクラスラベルによって決定される。NNTree は、一見判別木であるが、一意にモジュール型 NN に変換できるため NN でもある。

NNTree の利点の一つは、その構造が生成時に自動的に獲得されることである。従来の NN を考えると、適切なネットワーク構造を事前に決定することは難しく、事前知識がなければ、試行錯誤せざるを得なかった。学習を通して自動的にネットワーク構造を得る方法もいくつか提案されているものの、繰り返し再学習を必要とするため、多くの学習時間を必要とする欠点がある。

NNTree を使用すれば試行錯誤する必要はなく、ネットワークの構造をユーザーが決める必要もないので、簡単に利用できる。従来の多変量決定木の生成は NP 完全であるが、NNTree では発見的グループ分割法と誤差逆伝播 (BP) 法を組み合わせることで、NP 完全問題を回避し高速に NNTree を生成可能とした。

これまで我々は実験を通して NNTree が既存のいくつかのモデル (単変量決定木、単一モデルフル結合型 MLP (MLP: Multilayer Perceptron) 等) と比較し、NNTree の方が優れていることを示してきた。しかし、それらは古典的なモデルであり最新のモデルではなかった。本論文では、最も汎化能力が高いと言われているサポートベクターマシン (SVM: Support Vector Machine) とその性能を比較検討したい。

SVM はその内部構造から NN の一種であるとも言われているが、マージン最大化やカーネルトリックといった工夫を取り入れることで高い汎化能力を得ることを可能としている。しかし、高い汎化能力を実現するためには、大量のサポートベクターが必要となる。複雑

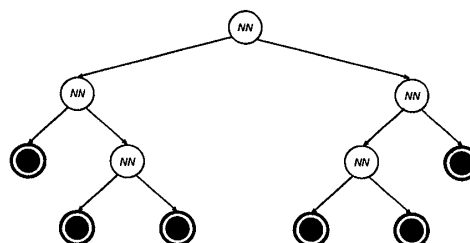


図 1: ニューラルネットツリーの例

な 2 クラスの問題であっても 1 個の SVM 解決してしまうため、複雑な判別境界を実現するために大量のサポートベクターを必要とする。

一方 NNTree の場合は、各終端ノードに割当てられたクラスの判別境界を考えると、それは各終端ノードまでのパス上に存在する NN の合成関数で表わされると考えられる。NNTree は各パス上の訓練データが十分に判別できるようになった時点でそのパスの成長 (中間ノードの追加) を停止する。つまり、終端ノードのクラスを判別できる判別境界が得られた時点で停止する。全体では複雑な判別境界であっても、より単純な判別境界の組合せで表現できるならば、そのシステム規模を小さくすることができる。これが NNTree と SVM の大きな相違点である。

## 2 実験結果

NNTree と SVM の性能を比較するために、カリフォルニア大学アーヴァイン校 (UCI) の公開データベースから得たデータベースを使い、実験を行った。使用したデータベースは Adult (adult), Credit Approval (crx), Dermatology (derma), Pima Indians Diabetes (diabetes), Glass Identification (glass), Internet Advertisements (internet), Iris (iris), Isolet (isolet), Letter Recognition (letter), Pen-based Recognition of Handwritten Digits (pendigits) の計 10 個である。括弧内の略称を表中及び以下の議論で使用する。

表 1 に各データベースのパラメーターを示す。すべての実験において、10-分割交差検定 (10-fold cross validation) をそれぞれ 5 回、計 50 回の試行を行った。実験に使用したコンピュータはサンワークステーションの Sun Ultra20 M2 モデル (CPU:AMD Opteron 1214, メモリ:1GB) である。

†Consideration about Generalization Ability of Neural Network Trees

<sup>†1</sup>Hiroto Hayashi, Graduate School of Computer Science and Engineering, The University of Aizu

<sup>†2</sup>Qiangfu Zhao, School of Computer Science and Engineering, The University of Aizu

表 1: データベースのパラメータ

	Number of examples ( $N_i$ )	Number of features ( $N_d$ )	Number of classes ( $N_c$ )
adult	48842	14	2
crx	690	15	2
dermatology	366	34	6
diabetes	760	2	8
glass	214	9	6
internet	3279	1558	2
iris	150	4	3
isolet	7797	617	26
letter	20000	16	26
pen-based	10992	16	10

本論文で使用した NNTree 生成アルゴリズムは、文献 [1] で我々が提案したアルゴリズムであり、現在最も性能が良い NNTree が得られる。SVM は DAGDAG (Directed Acyclic Graph) SVM [2] を使用する。SVM そのものは本来 2 クラス分類器であるため、多クラス問題を解決するためには、何らかの工夫が必要である。そのため、1-v-r (One-Versus-Rest) SVM や DAG SVM、Max Wins 法等いくつかの方法が提案されている。今回使用するのは、その中でも性能が良いとされる DAG SVM と比較する。使用したカーネルは RBF カーネルである。なお、SVM の実装には LIBSVM [3] を利用した。

ここで実験で使用したパラメータを示す。まず、NNTree のパラメータは次の通りである。(1) BP に使う学習率=0.5, (2) 学習回数 (エポック数) =1,000, (3) NN の入力数= $N_d$  (特徴数), (4) 中間ニューロン数=4, (5) 出力ニューロン数=1, (6) 閾値調整範囲=0.5± $\alpha$ 。SVM のパラメータは、すべてデフォルトの値を使用しており、RBF カーネルのパラメータ  $\gamma = 1/N_d$  である。

実験結果を表 2 に示す。“Test error”はテストデータに対するエラー率 (%), “System size”は総ニューロン数、または総サポートベクター数, “Training time”は学習に要した時間である。

表 2 を見ると分かるように、認識率が良かったデータベース数は 5 対 5 であるが、isolet, letter, pendigits の 3 つのデータベースで SVM と NNTree の認識率に大きな差がある。これは今後の課題である。システム規模を比べると、すべてのデータベースで NNTree の方が小さく、約 3 倍~約 100 倍の差が生じている。認識率で勝っている 5 つのデータベースでは NNTree を利用した方がリソースを節約できる。また、わずかな差で負けているデータベースでは、仮にその差が許容範囲であればシステム規模が小さい NNTree の方が良いと考えられる。

表 2: 実験結果

	Method	Test error	System size	Training time
adult	NNTree	<b>14.81</b>	<b>257.4</b>	378.48
	SVM	15.16	15081.2	<b>314.63</b>
crx	NNTree	16.06	<b>26.0</b>	2.89
	SVM	<b>15.10</b>	212.6	<b>0.06</b>
derma	NNTree	<b>2.94</b>	<b>21.8</b>	0.04
	SVM	3.22	225.5	<b>0.02</b>
diabetes	NNTree	24.97	<b>48.2</b>	3.04
	SVM	<b>23.39</b>	354.3	<b>0.06</b>
glass	NNTree	<b>30.86</b>	<b>74.2</b>	0.83
	SVM	32.67	279.8	<b>0.01</b>
internet	NNTree	<b>2.54</b>	<b>10.0</b>	<b>18.31</b>
	SVM	2.98	424.4	21.77
iris	NNTree	<b>3.47</b>	<b>8.0</b>	0.12
	SVM	3.60	25.0	<b>0.01</b>
isolet	NNTree	8.16	<b>148.8</b>	199.00
	SVM	<b>2.67</b>	15555.2	<b>57.63</b>
letter	NNTree	15.71	<b>585.4</b>	142.61
	SVM	<b>9.14</b>	31235.7	<b>15.96</b>
pendigits	NNTree	2.37	<b>57.4</b>	4.03
	SVM	<b>0.45</b>	1346.4	<b>1.32</b>

### 3 終わりに

本論文では、これまで我々が検討してきた NNTree モデルと汎化能力が高いと言われている SVM モデルを比較検討した。5 つのデータベースで NNTree の方が認識率が良く、すべてのデータベースに対して NNTree の方がシステム規模が小さかったことを示した。明らかに NNTree の認識率が悪いデータベースもあり、今後いかに NNTree の汎化能力を向上できるかが課題である。同程度の汎化能力が得られればより少ないリソースで利用できる NNTree の有用性が高くなると考えられる。

### 参考文献

- [1] Hayashi, H. and Zhao, Q. F.: Improvement of the Neural Network Trees through Fine-Tuning of the Threshold of Each Internal Node, *Neural Information Processing*, vol. 5863, pp. 657-666 (2009).
- [2] Platt, J. C., Cristianini, N., and Shawe-Taylor, J.: Large margin DAGs for multiclass classification, *Advances in neural information processing systems*, vol. 12, no. 3, pp. 547-553 (2000).
- [3] Chang, Chih-Chung, and Lin, Chih-Jen: LIBSVM: a library for support vector machines, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>