

## カオス時系列予測問題に対する自己生成ニューラルネットワークのアンサンブル学習特性に関する研究

### A Study on Ensemble Learning Performance of Self-Generating Neural Networks for Chaotic Time Series Prediction

中原 正貴<sup>†</sup>  
Masaki Nakahara

井上 浩孝<sup>‡</sup>  
Hirotaka Inoue

#### 1. まえがき

近年、時系列予測を行うためのニューラルネットワークによる方法が提案されており、従来用いられている統計的手法の ARMA モデル [1] と比較して、良好な予測精度を得られることが報告されている [2]。ニューラルネットワークを用いた手法では高い予測性能が得られるが、研究者各自が各問題に対するニューラルネットワークの構成や学習係数等のパラメータを経験や勘によって事前に決定する必要がある。特定の問題に対して最適なネットワーク構造を決定することは難しく、大変煩わしい作業である。このような煩わしい状況をさけるため、自己生成ニューラルネットワーク (self-generating neural network: SGNN) がネットワーク設計の容易さのため注目を集めている。SGNN は Kohonen の自己組織化地図 (self-organizing maps: SOM) [3] を拡張したものであり、競合学習により自己生成ニューラル木 (self-generating neural tree: SGNT) を生成する。SGNT アルゴリズムは訓練データ集合から直接ニューラル木を自動的に生成するため、パラメータの設定は必要なく、我々は訓練データを与えてやるだけでよい。この SGNT アルゴリズムは基本的に階層型クラスタリングアルゴリズムであるため、分類やクラスタリング問題に関しては他のニューラルネットワークよりも優れた結果を示すことが知られている。我々の研究ではこの SGNT アルゴリズムを実変数写像問題である時系列予測に対して優れた予測を高速に行えることを示した [4]。本研究では、SGNN の高速な学習性能を効果的に活かし、予測精度を改善するため複数の SGNN によるアンサンブル学習を行い、使用する SGNN の数に対する予測精度と処理時間を検討する。

#### 2. 自己生成ニューラルネットワーク

SGNN は SOM と同様に競合学習を用いて SGNT に与えられた訓練データ集合  $D = \{(\mathbf{x}_i, y_i) | i = 1, \dots, N\}$  内のデータを動的に組み込む。ここで、 $\mathbf{x}_i \in \mathbb{R}^m$  は入力ベクトルを、 $y_i$  は期待出力を表す。

SGNT は根、節点、葉からなる木構造のモデルである。根は SGNT 内に 1 つだけ存在し、入力データが提示される部分である。葉は訓練データが格納され、未学習データに対する出力候補となる部分である。根と葉を結ぶ部分に、節点が存在する。この節点は SGNT の生成過程で自動的に生成される。任意の節点において、根の方向に結合している節点または根をその節点の親とし、葉の方向に結合している節点または葉をその節点の子とする。SGNT アルゴリズムで使用する諸定義を以下に示す。

- 訓練実例データベクトル:  $\mathbf{e}_i = (e_{i1}, e_{i2}, \dots, e_{im})$ 。
- ニューロン:  $n_j$ 。
- 重みベクトル:  $\mathbf{w}_j = (w_{j1}, w_{j2}, \dots, w_{jm})$ 。
- $n_j$  に含まれる葉の数:  $c_j$ 。
- 距離測度:  $d(\mathbf{e}_i, \mathbf{w}_j) = \sqrt{\sum_{k=1}^m (e_{ik} - w_{jk})^2}$ 。
- 勝者ニューロン:  $n_{win}$ 。

根、節点、葉にはニューロンが存在し、それぞれ  $m$  次元重みベクトルとそれらのニューロンを根とみなした部

<sup>†</sup> 呉工業高等専門学校専攻科機械電気工学専攻

<sup>‡</sup> 呉工業高等専門学校電気情報工学科

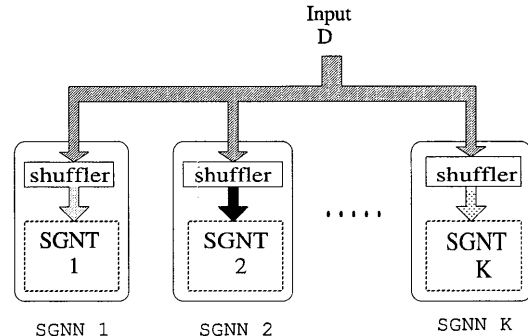


図 1 K 個の SGNT によるアンサンブル学習

分木に含まれる葉の数  $c_j$  を保持する。学習には  $N$  個の  $\mathbf{e}_i$  を使用する。競合学習による勝者となったニューロン  $n_{win}$  の重みベクトルは次のように更新する。

$$w_{jk} \leftarrow w_{jk} + \frac{1}{c_j} (e_{ik} - w_{jk}), 1 \leq k \leq m.$$

#### 3. アンサンブル学習法

SGNN は高速学習と大規模問題への適用可能性の優れた特性があるが、分類精度はバックプロパゲーションのような教師あり学習が実装されるフィードフォワード型ネットワークに比べ劣る。そこで我々は SGNN の高速学習特性を利用し、与えられた訓練データからより高い分類精度を引き出すため、K 個の SGNT によるアンサンブル学習を考える。SGNT の構造は学習中に動的に変化する。SGNT アルゴリズムはすべての訓練データが加えられた後に木構造を決定する。異なる木構造を持つ SGNT は訓練データの入力順を入れ替えることで生成される (図 1)。

#### 4. カオス時系列予測

カオスのもつ初期値に対する敏感な依存性の影響により、カオス時系列予測は長期予測が不可能であるとされている。しかしながら、非線形ダイナミクスが同定できれば、短期予測は可能である。

時系列予測は一定時間おきに等間隔にとられた  $m$  個の時系列データ  $\mathbf{x}_t, \mathbf{x}_{t-r}, \dots, \mathbf{x}_{t-(m-1)r}$  を入力データとし、そのステップ先の時系列データ  $\mathbf{x}_{t+p}$  を予測する。

$$\mathbf{x}_{t+p} = f(\mathbf{x}_t, \mathbf{x}_{t-r}, \dots, \mathbf{x}_{t-(m-1)r})$$

ここで  $\mathbf{x}_{t+p}$  は時刻  $t+p$  における真の時系列データ  $\mathbf{x}_{t+p}$  の予測値を表す。時系列予測に関する SGNN の入出力変数は次のようになる。

$$\begin{aligned} \mathbf{x}_i &= (\mathbf{x}_t, \mathbf{x}_{t-r}, \dots, \mathbf{x}_{t-(m-1)r}), \\ y_i &= \mathbf{x}_{t+p}, \\ o_i &= \mathbf{x}_{t+p} \end{aligned}$$

#### 5. 実験結果

アンサンブルする SGNN の数の変化による SGNN の予測特性、処理時間を分析するため、Henon map, Ikeda map より生成されるカオス時系列予測に対する予測実験

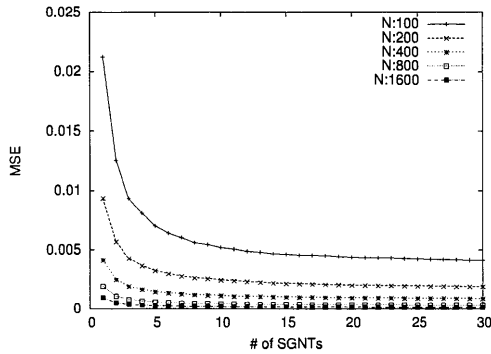


図2 Henon map に対する SGNT の数と MSE の関係

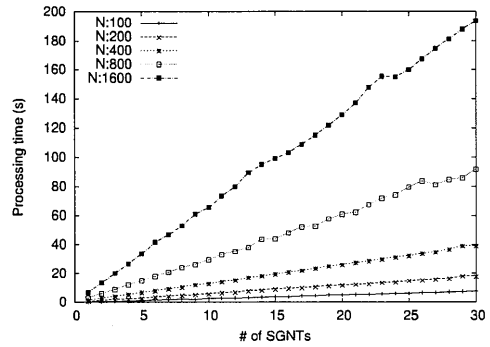


図3 Henon map に対する SGNT の数と処理時間 (s) の関係

を行った。処理時間は、100回の試行の訓練とテストに必要な総処理時間を表している。本研究では、数値実験にPC-AT互換機のパーソナルコンピュータ (CPU: Intel Pentium IV 2.40GHz, Memory: 512MB) を使用した。実際に使用したカオス時系列について次に記す。

- Henon map: Henon map は次式で与えられる2次元写像である。

$$\begin{aligned} x_{t+1} &= y_t + 1 - Ax_t^2, \\ y_{t+1} &= Bx_t \end{aligned}$$

各パラメータは  $A = 1.4$ ,  $B = 0.3$ ,  $x_0 = 0.3$ ,  $y_0 = 0.3$  である。今回、 $x$  の時系列を評価する。

- Ikeda map: Ikeda map は次式で与えられる2次元写像である。

$$\begin{aligned} x_{t+1} &= 1 + \mu(x_t \cos a - y_t \cos a), \\ y_{t+1} &= \mu(x_t \sin a + y_t \cos a), \\ a &= 0.4 - 6.0 / (1 + x_t^2 + y_t^2), \end{aligned}$$

ここで、 $\mu = 0.9$ ,  $x_0 = 0.3$ ,  $y_0 = 0.3$  である。今回、 $x$  の時系列を評価する。

(1) Henon map

図2にHenon mapに対するSGNTの数とMSEの関係を示す。図3に処理時間を示す。ここで図中横軸はアンサンブルに使用するSGNNの数である。図2より、予測精度MSEは訓練データの数 $N$ とアンサンブルに使用するSGNTの数に依存しており、訓練データ、及びSGNTの数を増やせば増やすほど、予測精度が向上することがわかる。訓練データ数が800,1600までになると、ほぼ同じ予測結果になっている。処理時間を比べると、図3より、訓練データの数 $N$ とSGNTの数の両者が増加するほど、処理時間が増加し、比例関係が見られる。 $N$ が1600にもなると処理時間が3分以上かかってしまうものがあつた。

(2) Ikeda map

図4にIkeda mapの実験結果を示す。図5に処理時間を示す。Ikeda mapの結果はHenon mapの結果と似た結果となり、訓練データ、及びSGNTの数を増やせば増やすほど、予測精度が向上した。処理時間を比べると、Henon mapの結果と同様に訓練データの数 $N$ とSGNTの数の両者が増加するほど、処理時間が増加し、比例関係が見られる。

6. むすび

実験結果より、訓練データ数とSGNTの数を大きくすることによって、高い予測精度が得られることがわかった。訓練データ数とSGNTの数を大きくするにつれて、処理時間が増加してしまい訓練データ数1600, SGNTの数30のとき、約200秒かかった。訓練データ数1600と

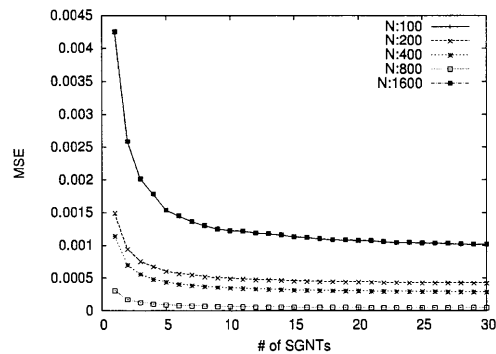


図4 Ikeda map に対する SGNT の数と MSE の関係

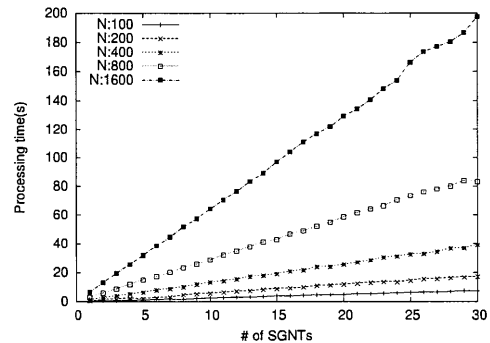


図5 Ikeda map に対する SGNT の数と処理時間 (s) の関係

800では、比較的高い予測精度が得られていたので、処理時間を考えれば本研究では訓練データ数が800のときが最適である。今後、より高い予測精度を得るには訓練データ数とSGNT数を大きくする必要があり、必ず処理時間が増加してしまう。そこで処理時間を少なくするために並列計算処理が今後望まれる。更に、時々刻々と変化するデータに対する予測特性を検討する予定である。

参考文献

- [1] B. Choi, "ARMA Model Identification," Springer Series in Statistics, Springer-Verlag, New York, 1992.
- [2] J. T. Connor and R. D. Martin, "Recurrent neural networks and robust time series prediction," IEEE Transactions on Neural Networks, vol. 5, no. 2, pp. 240-254, Mar. 1994.
- [3] T. Kohonen, "Self-Organizing Maps," Springer-Verlag, Berlin, 1995.
- [4] 井上浩孝, 成久洋之, "時系列予測に対する自己生成ニューラルネットワークの性能特性に関する一検討," 電子情報通信学会技術研究報告, CAS99-90, Nov. 9-10 1999.