

要約

動径基底関数ネットワーク（Radial Basis Function Network：RBFN）は、任意の非線形関数に対し、基底関数を足し合わせることで関数近似を効果的に行うことができるニューラルネットワークのうちの一つであり、様々な問題に適用されている。しかし、RBFNには冗長なニューロンの存在や、基底関数が不足していることによって関数近似ができない場合があるといった問題点がある。

そこで本研究では、このような問題に対して、ニューロン間の競合や、基底関数の複製を考慮したRBFNを提案する。そして、新たなアルゴリズムを機械学習へ適用し、さらなる効率化・改善を目的とする。

キーワード：シナプス結合荷重、シナプス可塑性方程式、競合動径基底関数ネットワーク、複製、学習の高速化

1 はじめに

関数近似問題やパターン識別に適したニューラルネットワークの一つにRBFNがある。RBFNは階層型ニューラルネットワークに比較してニューロンごとの局所的な学習が可能であるなどの優れた点を持つ。しかし、RBFNでは未知の非線形関数を近似するため、あらかじめ必要なニューロン数が不明であり冗長なニューロンを必要とする。一般に、ニューロンの増加は学習の遅延化や過学習の問題を生じることが知られている。

これらの問題を解決するために、適者生存型学習則に基づいたシナプス可塑性方程式を適用した、競合動径基底関数ネットワーク（Competitive RBFN：CRBFN）が提案されている[1]。CRBFNは、シナプス結合荷重間に競合を生じさせ、学習に必要なニューロンのみが自然に生き残るようになっており、冗長なニューロンの削減を図ることができる。しかし、CRBFNでは基底関数を追加する機能は無く、基底関数の数が足りない場合は、関数近似自体が不可能となる。そこで、CRBFNに基底関数を複製して追加する機能を加えた、複製・競合動径基底関数ネットワーク(Reproductive CRBFN：RC-RBFN)が提案されている[2]。

そこで本研究では、RC-RBFNが効率よく動径基底関数を削除あるいは追加するニューラルネットワークであることを示したのち、機械学習のアルゴリズムに適用して従来より学習が効率的に行われていることを示す。

2 複製・競合を考慮した動径基底関数ネットワーク

2.1 競合動径基底関数ネットワーク

ニューラルネットワークは大きく分けて、素子であるニューロン、それらを結合するシナプス、そして動作規則により構成される。なかでも、記憶にもっとも関係した情報処理は、シナプスにおいて行われているとされる。記憶には種々のものが考えられるが、本論文では短期記憶と長期記憶に着目し、短期記憶はニューロンの発火頻度、長期記憶は細胞膜の特性の変化により生じるものとする。シナプスの可塑性を記述する方程式は、これらの要因を含んだものとなっていなければならない。また、実際の生体では、シナプス結合の性質が興奮性、抑制性であるのかは送り出すニューロンにより決まる(Dale則)。さらに、微小な領域では成長や活動に必要な神経成長因子(Nerve Growth Factor：NGF)は競合によってシナプスに摂取される。

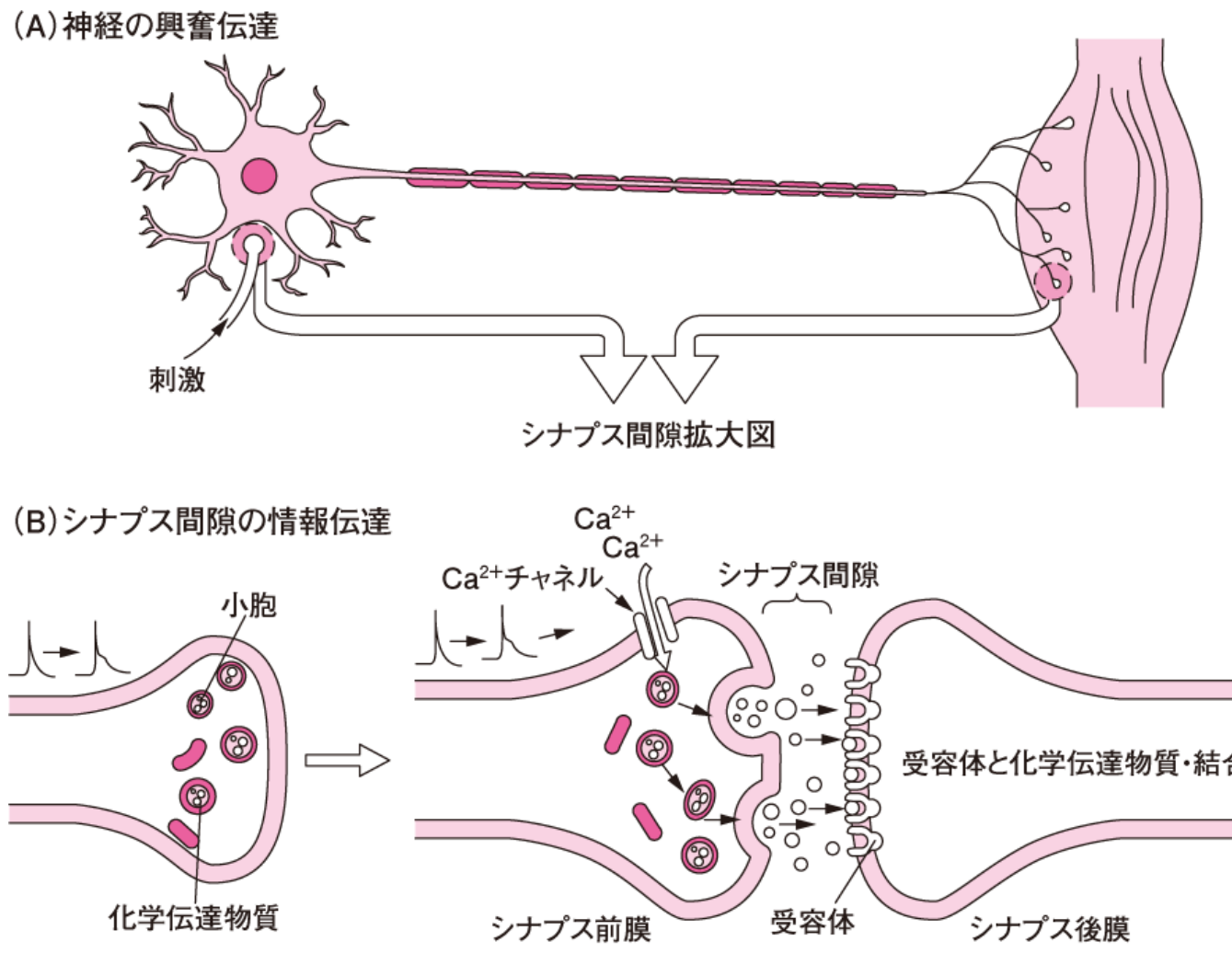


図2 微小領域におけるNGF摂取の様子

これらの事実もシナプス可塑性のモデル化において重要な要因であると考えられる。そこで、発火頻度や膜の特性変化を生じる物質の時間変化と、生理学的拘束条件であるDale則や微小な領域での競合を考慮したシナプス間感度の大きさの時間変化は以下の方程式に従うものとする(図2参照)。

Dale則を考慮したシナプス可塑性方程式

$$\begin{aligned} \frac{dw_{ik}^j}{dt} &= (\alpha_{ik}^j - \sum_h \gamma_{ik}^{jh} w_{ik}^h) w_{ik}^j \\ \alpha_{ik}^j &= \int_{x \in B_{ik}} \eta_{ik}(x) \xi_{ik}^j(x) dx & \gamma_{ik}^{jh} &= \int_{x \in B_{ik}} \xi_{ik}^j(x) \xi_{ik}^h(x) dx \\ \xi_{ik}^j(x) &= \mu_i \exp \left\{ -\frac{1}{2} \left(\frac{x - m^j}{\sigma^j} \right)^2 \right\} \end{aligned}$$

$$\begin{aligned} \alpha_{ik}^j &: \text{内的自然増加率} & \gamma_{ik}^{jh} &: \text{第} j, h \text{ニューロン間の競合の効果} \\ \xi_{ik}^j &: \text{第} j \text{ニューロンの微小領域} B_{jk} \text{におけるシナプス前終末発火頻度} & \eta_{ik} &: \xi_{ik}^j \text{が作用する第} i \text{ニューロンの細胞膜におけるシナプス後発火頻度} \end{aligned}$$

図3 シナプス可塑性方程式

このような、競合を考慮したシナプス可塑性方程式に従うRBFNがCRBFNである。

2.2 ターミナルアトラクタと基底関数の複製

先ほど述べたCRBFNでは、競合に負けたシナプス結合荷重は平衡状態で0になるが、平衡解への漸近は指数関数的に行われるので原理的には有限時間で平衡状態へ到達することはできない。

そこで、あるシステムが最終的に安定した「学習結果」や「最適解」に収束する、というターミナルアトラクタ(Terminal Attractor)の概念を適用して、与えられた時刻 t^* で平衡解へ収束するように修正されたシナプス可塑性方程式を導出する。まず、動的システムの安定性を解析するために使われる数学的なツールであるLyapunov関数とその時間変化、そして与えられた時刻 t^* で平衡解へ収束するように修正されたシナプス可塑性方程式を以下のように定義する(図3参照)。

Lyapunov関数とその時間変化

$$\begin{aligned} V(\mathbf{w}_{ik}) &= \frac{1}{2} \int_{x \in B_{ik}} \{ \eta_{ik}(x) - s_{ik}(x) \}^2 dx, & \frac{dV(\mathbf{w}_{ik})}{dt} &= -\frac{V(\mathbf{w}_{ik}^0)^R V(\mathbf{w}_{ik})^{\frac{1}{r}}}{Rt^*} \\ \frac{dw_{ik}^j}{dt} &= \frac{\left(\alpha_{ik}^j - \sum_{h=1}^{N_{ik}} \gamma_{ik}^{jh} w_{ik}^h \right) w_{ik}^j}{\sum_{j=1}^{N_{ik}} w_{ik}^j \left(\alpha_{ik}^j - \sum_{h=1}^{N_{ik}} \gamma_{ik}^{jh} w_{ik}^h \right)^2} \times \frac{V(\mathbf{w}_{ik}^0)^R V(\mathbf{w}_{ik})^{\frac{1}{r}}}{Rt^*} \\ \mathbf{w}_{ik} &\equiv [w_{ik}^1, w_{ik}^2, \dots, w_{ik}^{N_{ik}}]^T \in R^{N_{ik}} & \mathbf{w}_{ik}^0 &: \mathbf{w}_{ik} \text{の初期値} \\ r &: \text{任意の奇数} & R &: R = \frac{r-1}{r} & s_{ik}(x) &= \sum_{j=1}^{N_{ik}} w_{ik}^j \xi_{ik}^j(x) \end{aligned}$$

図4 Lyapunov関数とその時間変化

ここまでで、シナプス結合荷重間の競合を考慮したシナプス可塑性方程式を利用することにより学習の効率化を図る手法が提案されてきた。ところが、CRBFNは冗長な動径基底関数を削除する能力をもつものの、必要とされる動径基底関数を追加する能力は備えていない。ニューラルネットワークに関数近似を行うために必要な数のニューロンが存在しない場合は、関数近似を行うこと自体が不可能となる。

そこで、新しい動径基底関数を追加する能力を備えたニューラルネットワークとしてRC-RBFNが提案されている。このRC-RBFNはシナプス可塑性方程式に関する考察から得られるものであり、必要な動径基底関数を効率的に追加することができる。簡単のため以降はとある i と k についてのみ考えるものとし、添え字の i, k は省略する。

ここで、動径基底関数の複製アルゴリズムに論じるにあたって自由エネルギーを以下のように導出する(図4参照)。自由エネルギーはシステムのエネルギーとエントロピー(無秩序さ)のバランスを表す量であり、機械学習においては自由エネルギーの最小化を通じてパラメータの推定を行う。

自由エネルギーの導出

$$\begin{aligned} & \text{累積2乗誤差関数の最小化} & E(m^j) &= \frac{1}{2} \sum_i E(x_i, m^j) = \frac{1}{2} \sum_i \{ \eta(x_i) - s(x_i) \}^2 & \Rightarrow & \text{各入力ベクトル} x_i \text{ごとに} E(x_i, m^j) \text{を最小化} \\ & x_i \text{の条件付き確率密度関数} & p_\beta(x_i | m_{[i]}^j) &= Z_\beta^{-1}(m^j) \exp \{ -\beta E(x_i, m_{[i]}^j) \} & m_{[i]}^j &: x_i \text{に依存した平均ベクトル} \\ & & Z_\beta(m^j) &= Z_\beta(m^j) = -\sum_i \exp \{ -\beta E(x_i, m^j) \} \\ & p_\beta(x_i | m_{[i]}^j) \text{の特性} & & & & \\ & \text{確率の正規化と、} E(x_i, m^j) \text{の条件付き期待値} (E(m^j))_\beta = \sum_i p_\beta(x_i | m_{[i]}^j) E(x_i, m_{[i]}^j) \text{が一定になるという} & & & & \\ & \text{二つの制約のもとで、エントロピー} -S_\beta(m^j) = -\frac{1}{\beta} \sum_i p_\beta(x_i | m_{[i]}^j) \log p_\beta(x_i | m_{[i]}^j) \text{を最大化する。} & & & & \\ & \text{このとき、自由エネルギーは} F_\beta(m^j) = -\frac{1}{\beta} \log Z_\beta(m^j) \text{で定義され、} & & & & \\ & S_\beta(m^j) = -F_\beta(m^j) + \beta (E(m^j))_\beta \text{と表すことができる。} & & & & \end{aligned}$$

図5 自由エネルギーの導出

このような自由エネルギーは、データのクラスタリングのための手法であるメルティング[4]においても同様に定義されている。メルティングとは、 $m = x$ かつ β が ∞ である初期状態から、徐々に β を0へ近づけていきながら、パラメータ m を自由エネルギー $F_\beta(m)$ の最急降下方向に更新していくものである。その結果、パラメータ m は徐々に同じ値をとりはじめ、最終的に一つの値に収束する。

次に、RC-RBFNにおける動径基底関数を複製する位置の決定法を図6のように定める。

複製する位置の決定法

$$\begin{aligned} & \text{RC-RBFNでの} m^j \text{の更新則} & \Delta_\beta m^j &= \sum_i p_\beta(x_i | m_{[i]}^j) \Delta m_{[i]}^j & \Delta m_{[i]}^j &= -\varepsilon \frac{\partial E(x_i, m_{[i]}^j)}{\partial m_{[i]}^j} \quad \varepsilon: \text{正の定数} \\ & \beta = 0 \text{で、初期の状態が} m_{[i]}^j = m^j(\forall i) \text{の場合} & & & & \\ & & \Delta_\beta m^j &= \sum_i p_0(x_i | m_{[i]}^j) \Delta m_{[i]}^j = \Delta m^j & & \\ & \text{であり、RC-RBFNのパラメータ} m^j \text{の更新則が従来のRBFNのパラメータ} m^j \text{の更新則そのものになっている。} & & & & \\ & \text{このとき、} \sigma^j \text{を固定したまま} \beta \rightarrow \infty \text{にすると、} \Delta_\infty m^j = 0 \text{とするパラメータ} m_{[i]}^j \text{は} & & & & \\ & \sum \exp \{ -\beta E(x_i, m_{[i]}^j) \} \xi^j(x_i, m_{[i]}^j) (x_i - m_{[i]}^j) \{ \eta(x_i) - s(x_i, m_{[i]}^j) \} = 0 & & & & \\ & \text{を満たし、} x_i(\forall i) \text{を含む任意の値となることがわかる。つまり、教師入力信号がパラメータ} m^j \text{の収束点として検出されることとなる。} & & & & \end{aligned}$$

図6 複製する位置の決定法

以上のことから、動径基底関数の複製を考慮したRC-RBFNの学習則は以下のように提案される。

RC-RBFNの学習則

$$\begin{aligned} & \text{STEP1: シナプス結合荷重} w^j, \text{パラメータ} m^j, \sigma^j \text{を各更新則により更新。} \\ & \text{STEP2: 累積2乗誤差関数} E(m^j) \text{が} E(m^j) \approx 0 \text{になったら学習終了。ある正数} \varepsilon \text{より大きな値に収束したならSTEP3へいく。} \\ & \text{STEP3: 学習収束時に得られているすべての動径基底関数について、} \beta \text{を0から徐々に大きくしていきながらパラメータ} m^j \text{を更新する。} \\ & \text{STEP4: 分岐により} \Delta_\beta m^j = 0 \text{となる点が増えたとき、第} j \text{動径基底関数を第} p \text{動径基底関数として複製する。そのとき、シナプス結合荷重} w^p, \text{パラメータ} \sigma^p \text{並びにパラメータ} m^p \text{は形質としてもとの第} j \text{動径基底関数のものを引き継ぎ、パラメータ} m^p \text{は新たに増えた点とする。STEP1へ戻る。} \end{aligned}$$

図7 RC-RBFNの学習則

3 提案手法

本研究で提案するシステムの概要について説明する。本研究では学習アルゴリズムとして、RC-RBFNを提案し、実際に基底関数の複製や冗長なニューロンが削除されていることを確認した。この学習アルゴリズムを既存の機械学習に適用することで、処理速度の向上を図る。

4 数値実験並びに考察

数値実験では、図8の設定で実験を行った。

数値実験の設定

$$\begin{aligned} & \bullet \text{基底関数の数: } 10 \\ & \bullet \text{教師信号: } \eta(x) = 4N(-13, 2) + 2N(-1, 1) + 3(8, 4) \\ & \text{ただし、} N(m, \Sigma) \text{は平均} m, \text{分散} \Sigma \text{の正規密度関数} \\ & \bullet \text{シナプス結合荷重} w \text{の初期値:} \\ & [w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}] = [0.1, 0.4, 0.7, 1.0, 1.3, 1.6, 1.9, 2.2, 2.5, 2.8] \\ & \bullet \text{平均位置} m \text{の初期値:} \\ & [m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9, m_{10}] = [-15, -7, -2.5, 13, -12, -3, 7, 8, 16] \\ & \bullet \text{分散} \Sigma \text{の初期値:} \\ & [\Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4, \Sigma_5, \Sigma_6, \Sigma_7, \Sigma_8, \Sigma_9, \Sigma_{10}] = [6, 10, 3, 5, 1, 13, 9, 11, 4, 2] \end{aligned}$$

図8 数値実験の設定

初めに、CRBFNが与えられた教師信号に対して近似できていることを示す(図9参照)。学習回数は1000回とした。

CRBFNによる学習の結果

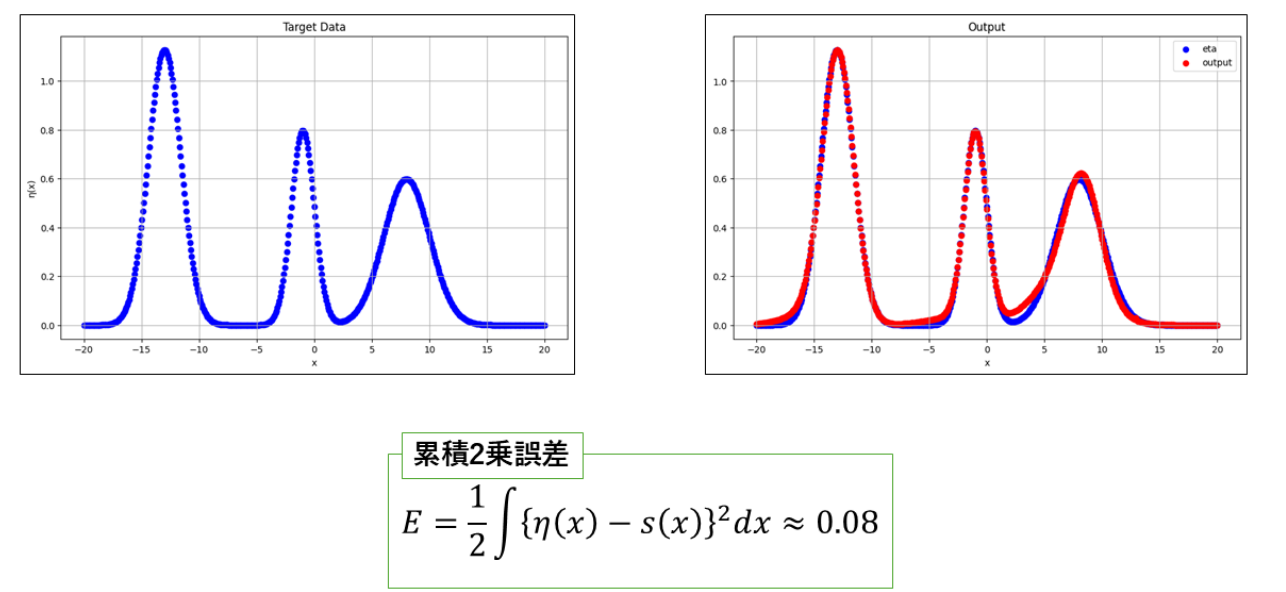


図9 CRBFNによる近似結果

学習回数1000回では完全に教師信号と一致させるまでは至らなかったが、シナプス結合荷重などのパラメータが収束していなかったため、学習回数をさらに増やすことで累積2乗誤差の値を減らすことが可能であると考えられる。

次にシナプス結合荷重 w の時間変化にターミナルアトラクタを適用し、適用前の100回目の結果に指定した学習回数(今回は50回とした)でシナプス結合荷重が収束していることを示す。また、適用前と適用後と比較して同様の結果を得られていることも併せて示す。

TA適用前後の結果

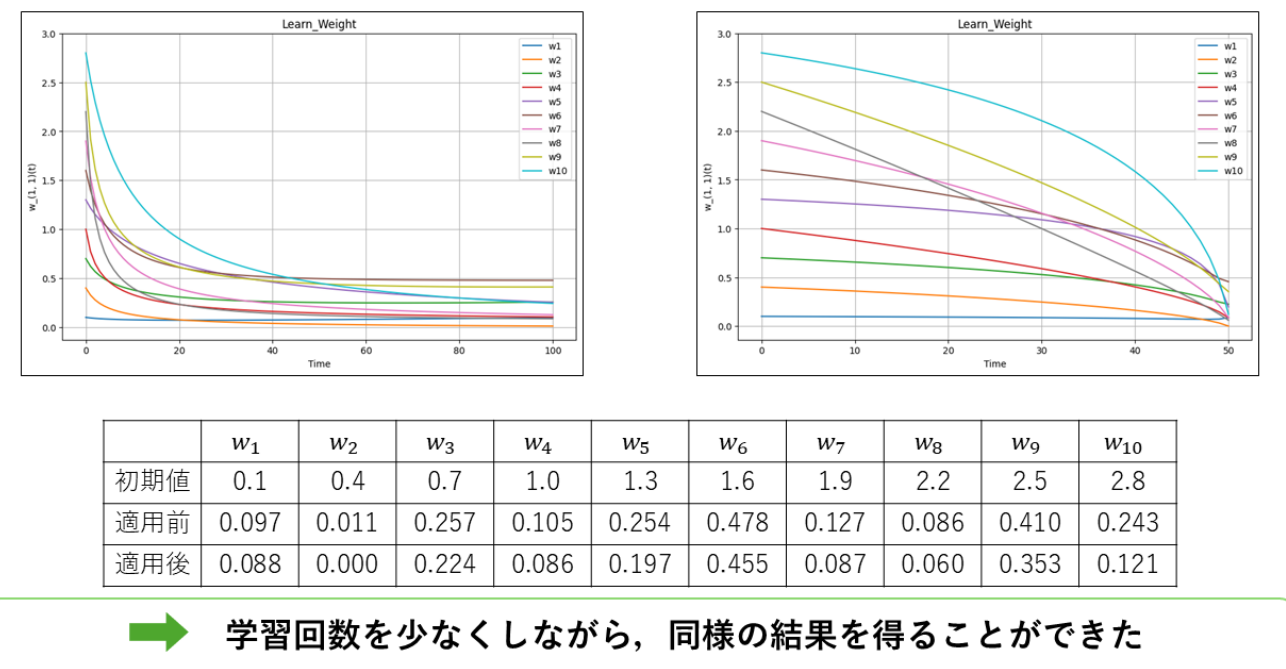


図8 TA適用前・適用後のシナプス結合荷重
最後に，平均 m の更新により基底関数が複製されていることを示す．

5 おわりに

本研究では，冗長なニューロンの削除，教師信号を近似するために必要な基底関数の追加を行う CRBFN や RC-

RBFN を提案した．今後の課題として，実際の機械学習の学習アルゴリズムに適用することと，負の教師信号に対しても従来と同様の近似を行えるようにすることを考えている．

参考文献

[1] 奥原 浩之, 尾崎 俊治, “ 適者生存型学習則を適用した競合動径基底関数ネット

ワーク ”, 電子情報通信学会論文誌, pp. 3191-3199, 1997

[2] 奥原 浩之, 佐々木 浩二, 尾崎 俊治, “ 環境の変化に適応できる複製・競合動径基底関数ネットワーク ”, 電子情報通信学会論文誌, pp. 941-951, 1999

[3] 奥原 浩之, 尾崎 俊治, “ Dale 則を考慮したシナプス可塑性方程式の解析 ”, システム制御情報学会論文誌, pp. 718-720, 1995

[4] Wong, Y.-. fai . Clustering Data by Melting. Neural Computation 1993, 5 (1), 89–104. <https://doi.org/10.1162/neco.1993.5.1.89>.