数理解析研究所講究録
第1554巻 2007年 217-224

定数項の大きな線形しきい値関数に対する高速なオンライン学習

石橋 浩介 (Kosuke Ishibashi), 畑埜 晃平 (Kohei Hatano), 竹田 正幸 (Masayuki Takeda)

九州大学システム情報科学研究院情報理学部門
(Department of Informatics, Graduate School of Information Science and Electrical Engineering, Kyushu University)

概要

学習対象が大きな定数項を持つ関数に対する，高速なオンライン学習アルゴリズムを考える．そのような関数の例としては，k-節式や r-of-k しきい値関数が挙げられる．これらの関数はオンライン学習アルゴリズム Winnow によって，高速に学習可能であることが知られている．しかし，Winnow を用いて高速に学習するためには，学習対象のマージンや定数項といった情報をあらかじめ知っておく必要がある．だが，これらの情報は一般には未知である．そこで，本論文ではマージンや定数項のいずれかが既知の場合に学習可能な適応型 Winnow を提案し，またそれが従来の Winnow と同程度の速さで動作することを示す．

1 はじめに

線型しきい値関数のオンライン学習問題は，機械学習における最も基礎的な問題の1つである．オンライン学習は学習者と教師の逐次的なやりとりによって行われる．各試行 $t = 1, 2, \ldots,$ において，(i) 学習者は事例 $\boldsymbol{x}_t \in \mathcal{X} = \{-1, +1\}^N$ を教師から受け取り，(ii) 学習者の持つ予測関数に応じて，ラベル $\hat{y}_t \in \{-1, +1\}$ の予測を行う．次に (iii) 学習者は教師から真のラベル $y_t \in \{-1, +1\}$ を受け取り，必要に応じて予測関数を更新する．学習者の目的は，誤り回数を可能な限り小さくすることである．特に，線型しきい値関数のオンライン学習においては，各ラベル y は線型しきい値関数 $\mathrm{sign}(\boldsymbol{u} \cdot \boldsymbol{x} + b)$ $(\boldsymbol{u} \in \mathbb{R}^N, b \in \mathbb{R})$ によって決まると仮定する，ここで，$\mathrm{sign}(x)$ は符号を返す関数であり，$x \geq 0$ のとき $+1$, それ以外の場合 -1 を返す．

線型しきい値関数の学習はさかんに研究されており (例えば [1, 4, 7, 8, 10, 11, 12, 13])，特に代表的な学習アルゴリズムに，Perceptron [11, 12, 13] と Winnow [7] が挙げられる．共に線型しきい値関数 $\mathrm{sign}(\boldsymbol{w} \cdot \boldsymbol{x} + b')$ を予測関数として用いるが，主な違いとして，Perceptron は予測関数を $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \alpha y_t \boldsymbol{x}_{t,i}$ (α は定数) と加算的に更新するのに対して，Winnow は $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t \exp\{\alpha y_t \boldsymbol{x}_{t,i}\}$ と乗算的に更新を行う．ラベルを与える線型しきい値関数 (学習対象) がマージン γ_p^* を持つ場合，Perceptron は $O(1/\gamma_2^{*2})$, Winnow は $O(\ln N/\gamma_\infty^{*2})$ 回の誤り回数で学習することが可能である．ここでマージン γ_p^* $(p = 1, \ldots, \infty)$ とは，真の重みベクトル $\boldsymbol{u}$ と事例空間 $\mathcal{X}$ の任意の事例 $\boldsymbol{x}$ に対して，$\min_{\boldsymbol{x} \in \mathcal{X}} \frac{|\boldsymbol{u} \cdot \boldsymbol{x}|}{\|\boldsymbol{x}\|_p \|\boldsymbol{u}\|_q}$ (ただし，$1/p + 1/q = 1$) となる値のことであり，真の重みベクトルが成す超平面から最も近い事例までの距離を表す．ただし，距離の定義は各アルゴリズムによって異なり，Perceptron ならば2ノルム $\|\boldsymbol{x} - \boldsymbol{x}'\|_2 = \sqrt{\sum_{i=1}^N (x_i - x_i')^2}$，Winnow ならば ∞ ノルム $\|\boldsymbol{x} - \boldsymbol{x}'\|_\infty = \max_{i=1,\ldots,N} |x_i - x_i'|$ で定義される．

真の重みベクトル $\boldsymbol{u}$ が疎な場合，Winnow の方が Perceptron よりも少ない誤り回数で学習できることが知られている [5]．例えば，k-節式や r-of-k しきい値関数等のブール関数の学習問題を考える．$\mathcal{X} = \{-1, +1\}^N$ 上の k-節式とは N 変数中にある k 個の変数 $x_1, \ldots, x_k$ についてそれらのうち少なくともどれか1つが $+1$ の値を取るとき，$+1$ を返し，それ以外の場合に -1 を返す関数である．k-節式は $f(\boldsymbol{x}) = \mathrm{sign}\{(1/k)x_1 +$

$\ldots, (1/k)x_k + 1 - 1/k\}$ という線型関数で表現できる．同様に，r-of-k しきい値関数は，ある k 個の変数のうち r 個以上が $+1$ ならば $+1$ を返す関数であり，$f(\boldsymbol{x}) = \mathrm{sign}\{(1/k)x_1 + \ldots, (1/k)x_k + 1 - (2r-1)/k\}$ で表される．これらの関数は上述のオンライン学習アルゴリズムで学習可能である．Perceptron を用いた場合この k-節式や r-of-k しきい値関数をいずれも，$O(kN)$ 回の誤り回数で学習する．一方 Winnow を用いると，それぞれ，$O(k \ln N)$，$O(kr \ln N)$ の誤り回数で学習可能なことが示されている [7]．つまり，$k << N$ の 場合には，Winnow が Perceptron よりも指数的に高速であることがわかる．また，r-of-k しきい値関数の例でもわかるように，Winnow は学習対象の関数の定数項が大きければ誤り回数がより少なくなる，という性質が見られる．しかし，Winnow において上記の誤り回数で学習を行うためには，マージン γ^* と定数項の値をあらかじめ知っておく必要がある．だが，これらは真の線形しきい値関数が分からなければ求められない値であり，実際の学習において現実的ではない．

関連研究に，ALMA [2] という p-norm アルゴリズム [3, 4] の拡張がある．p-norm アルゴリズムとは，Perceptron や Winnow のより一般的な形であり，$p = 2$ とすれば Perceptron，$p = O(\ln N)$ とすれば Winnow と同じ動作をする．この，ALMA は学習対象のマージン γ_p^* の値を知らなくても学習可能であり，例えば，$p = O(\ln N)$ の場合，その誤り回数は $O(\frac{\ln N}{\gamma_\infty^{*2}})$ となる．さらに，定数 $c(0 \leq c < 1)$ を設定すれば，誤り回数 $O(\frac{\ln N}{(1-c)^2\gamma_\infty^{*2}})$ で学習し，得られた重みベクトルは事例集合 $\mathcal{X}$ に対して $c\gamma^*$ 以上のマージンを持つことが知られている．しかし，ALMA には定数項が大きければ誤り回数がより少なくなる，という性質はない．

本研究では，マージンや定数項を知らなくても学習可能で，かつ定数項が大きければ誤り回数が少なくなるという性質を保持した，改良版 Winnow の構築を目指す．学習対象 が正の定数項を持ち，各事例 $\boldsymbol{x}_t$，ラベル y_t に対して，$y_t(\boldsymbol{u} \cdot \boldsymbol{x}_t + 1 - \delta^*) \geq \gamma^*$ (ただし，$\|\boldsymbol{u}\|_1 = 1$, $0 \leq \delta^* \leq 1$) を満たすと仮定する (負の定数項を持つ場合は定数項を $-(1-\delta)$ とおくと同様に議論出来る)．この仮定のもとで，まず，γ^*，δ^* が既知の場合に Winnow の誤り回数が $O(\frac{\delta^* \ln N}{\gamma^{*2}})$ 回である事を示す．例えば，r-of-k しきい値関数の学習において，そのマージンは $\gamma^* = \frac{1}{k}$，$\delta^* = \frac{r}{k}$ となるので，結果として $O(kr \ln N)$ で学習可能となる．次に，γ^* と δ^* のいずれか一方のみが既知である場合にも既知の場合と同じオーダーの誤り回数で済むことを示す．特に，δ^*, γ^* が既知，または δ^* のみが既知の場合に本提案手法は ALMA 同様，定数 c を設定すれば，誤り回数 $O(\frac{\delta^* \ln N}{(1-c)^2\gamma^{*2}})$ で学習し，得られた重みベクトルは事例集合 $\mathcal{X}$ に対して $c\gamma^*$ 以上のマージンを持つことができる．残念ながら，δ^*, γ^* 両方が未知の場合に $O(\frac{\delta^* \ln N}{(1-c)^2\gamma^{*2}})$ の誤り回数で学習可能かどうかは未解決問題である．

2 準備

$\mathcal{X} = \{-1, +1\}^N$ を 事例空間 とする．また事例空間 $\mathcal{X}$ の要素を 事例 と呼び，事例と $\boldsymbol{x} \in \mathcal{X}$ と ラベル $y \in \{-1, +1\}$ の組 $(\boldsymbol{x}, y)$ を 例 と呼ぶ．次に学習対象の関数について述べる．各事例 $\boldsymbol{x}$ のラベル y は未知の線型しきい値関数によって $y = \mathrm{sign}(\boldsymbol{u} \cdot \boldsymbol{x} + 1 - \delta^*)$ と与えられると仮定する．ここで，重みベクトル $\boldsymbol{u} \in \mathbb{R}^N$ は N 次元の正数ベクトルとし，$\sum_i u_i = 1$, $u_i \geq 0 (i = 1, \ldots, N)$ を満たすとする．また，各例 $(\boldsymbol{x}, y)$ に対して $y(\boldsymbol{u} \cdot \boldsymbol{x} + 1 - \delta^*) \geq \gamma^*$ を満たすとする．さらに，以下のような仮定を設ける：(i) $\boldsymbol{x} = (1, 1, \ldots, 1)$ に対して $y = +1$, (ii) $\boldsymbol{x} = (-1, -1, \ldots, -1)$ に対して $y = -1$．もしも仮定 (i) または (ii) が満たされない場合，学習対象はそれぞれ -1, $+1$ を返す定数関数となってしまう．以上の仮定から，次の命題がただちに成り立つ．

命題 1. $\gamma^* \leq \delta^*$.

$\{1, \ldots, N\}$ 上の任意の確率分布 $\boldsymbol{p}, \boldsymbol{q}$ について，$\boldsymbol{p}$ と $\boldsymbol{q}$ の 相対エントロピー(Kullback-Leibler ダイバージェンスとも呼ばれる) を $\Delta(\boldsymbol{p}, \boldsymbol{q}) = \sum_{i=1}^N p_i \ln \frac{p_i}{q_i}$ と定義する．相対エントロピーは常に非負であり，$\Delta(p, q) = 0$ となるのは $p = q$ のときのみである．おおまかに言えば，相対エ

ントロピーは確率分布間の“距離のようなもの”だと見なせる．しかし，一般に相対エントロピーは非対称であり，三角不等式を満たさないため距離ではない．

3 Winnow アルゴリズム

3.1 マージンと定数項が既知の場合

まず，マージン γ^* と定数項 δ^* が両方分かっている場合について考える．マージン，定数項の予想値をそれぞれ γ, δ に固定した場合のアルゴリズム $\mathrm{Winnow}_{\gamma,\delta}$ を図 1 に示す．

以下の補題を示す．

補題 1. $0 \le \alpha \le (\ln 2)/2 \approx 0.35$ とすると，各 $t = 1, \ldots, T$ について

$$\ln Z_t \le \{-y_t(1-\delta) + c\gamma\}\alpha + 4\delta\alpha^2$$

である．

証明. まず，指数関数 e^x の凸性より，

$$e^{\alpha y_t x_{t,i}} \le \frac{e^{\alpha} - e^{-\alpha}}{2} y_t w_{t,i} + \frac{e^{\alpha} + e^{-\alpha}}{2},$$

が成り立つ．よって

$$\begin{aligned}
&\ln \sum_{t=1}^{n} w_{t,i} e^{\alpha y_t x_{t,i}} \\
&\le \ln \left\{ \frac{e^{\alpha} - e^{-\alpha}}{2} y_t \sum_{i=1}^{n} w_{t,i} + \frac{e^{\alpha} + e^{-\alpha}}{2} \right\} \\
&\le \ln \left\{ \frac{e^{\alpha} - e^{-\alpha}}{2} \{-y_t(1-\delta) + c\gamma\} \right. \\
&\qquad \left. + \frac{e^{\alpha} + e^{-\alpha}}{2} \right\} \\
&= \ln \left\{ \frac{1 - y_t(1-\delta) + c\gamma}{2} e^{\alpha} \right. \\
&\qquad \left. + \frac{1 + y_t(1-\delta) - c\gamma}{2} e^{-\alpha} \right\},
\end{aligned}$$

ここで，最後の不等式は $y_t(\boldsymbol{w}_t \cdot \boldsymbol{x}_t + 1 - \delta) < c\gamma$ であることから導ける．今，$g(a) = \ln(w_1 e^{a} + w_2 e^{-a})$ とおく ($w_1 + w_2 = 1$ and $w_1, w_2 \ge 0$ とする)．$g(a)$ のテイラー展開を計算すると，ある a' $(0 < a' < a)$ が存在して，

$$\begin{aligned}
g(a) &= g(0) + g'(0)a + \frac{1}{2} g''(a')a^2 \\
&= (w_1 - w_2)a + \frac{2w_1 w_2}{(w_1 e^{a'} + w_2 e^{-a'})^2} a^2 \\
&\le (w_1 - w_2)a + 2e^{2a} w_1 w_2 a^2 \qquad (1)
\end{aligned}$$

を満たす．不等式 (1) に $w_1 = \frac{1 - y_t(1-\delta) + c\gamma}{2}$, $w_2 = \frac{1 + y_t(1-\delta) - c\gamma}{2}$ および $a = \alpha$ を代入すると (ここで，$\frac{1 - y_t(1-\delta) + c\gamma}{2}, \frac{1 + y_t(1-\delta) - c\gamma}{2} \ge 0$ が成り立つことは，$\delta \ge \gamma$, $\delta + \gamma \le 2$ から容易に確かめられる)，以下の不等式が得られる．

$$\begin{aligned}
&\ln \sum_{t=1}^{n} w_{t,i} e^{\alpha y_t x_{t,i}} \\
&\le \{-y_t(1-\delta) + c\gamma\}\alpha \\
&\quad + \frac{e^{2\alpha}}{2} \{2\delta - (\delta - y_t c\gamma)^2 - 2y_t c\gamma\}\alpha^2 \\
&\le \{-y_t(1-\delta) + c\gamma\}\alpha + 4\delta\alpha^2,
\end{aligned}$$

ここで最後の不等式は $-y_t c\gamma \le \delta$ より成り立つ．　(証明終わり)

次に，$\mathrm{Winnow}_{\gamma^*,\delta^*}$ について以下の定理が成り立つ．

定理 2. 例の列 $S = ((\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_T, y_T))$ が与えられたとする．$t = 1, \ldots, T$ のとき，$y_t(\boldsymbol{u} \cdot \boldsymbol{x} + 1 - \delta^*) \ge \gamma^*$ となるような $(\boldsymbol{u}, 1-\delta^*) \in \mathbb{R}^n \times \mathbb{R}$ が存在するならば，$\mathrm{Winnow}_{\gamma^*,\delta^*}$ の誤り回数は高々

$$m \le \frac{16\delta^* \ln n}{(1-c)^2 \gamma^{*2}}$$

となる．ただし，$\sum_i u_i = 1$ かつ $0 \le \delta^* \le 1$．加えて，m 回の間違いの後，$\mathrm{Winnow}_{\gamma^*,\delta^*}$ の仮説は $\boldsymbol{x}_t$ に対して少なくとも $c\gamma^*$ のマージンを持つ．

証明. 解析を単純化するため $\mathrm{Winnow}_{\gamma^*,\delta^*}$ は全ての $t = 1, 2, \ldots, m$ について予測を誤ると仮定する．$\boldsymbol{w}_t$ を更新した際の $\boldsymbol{u}$ と $\boldsymbol{w}_t$ 間の相対エントロピーの減少分を評価することにより証明を行う (これは，オンライン学習アルゴリズムの解析においては標準的な技法である．例えばサーベイ [6] を参照せよ)．まず，以下の不等式が成

Winnow$_{\gamma,\delta}$

1. $\boldsymbol{w}_1 = (1/n, \ldots, 1/n) \in \mathbb{R}^N$ とする.
2. $t = 1, \ldots, T$ について，以下の処理を繰り返す:
   (a) 事例 $\boldsymbol{x}_t \in \mathcal{X}$ を受け取る;
   (b) 予測 $\hat{y}_t = \mathrm{sign}(\boldsymbol{w}_t \cdot \boldsymbol{x}_t + 1 - \delta)$ を計算する;
   (c) ラベル $y_t \in \{-1, +1\}$ を受け取る;
   (d) もし間違っていたら (つまり $y_t(\boldsymbol{w}_t + 1 - \delta) \leq c\gamma$ ならば) 以下のように更新:

$$w_{t+1,i} = \frac{w_{t,i} e^{\alpha y_t x_{t,i}}}{Z_t}$$

   . ただし $\alpha = (1-c)\gamma/4\delta$ かつ $Z_t = \sum_{i=1}^{n} w_{t,i} e^{\alpha y_t x_{t,i}}$. 間違えなかった場合，$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t$ とする.

図 1: パラメータ γ と δ を固定した Winnow アルゴリズム

り立つ.

(証明終わり)

$$\begin{aligned}
&\Delta(\boldsymbol{u}, \boldsymbol{w}_t) - \Delta(\boldsymbol{u}, \boldsymbol{w}_{t+1}) \\
&\geq \alpha\{\gamma^* - y_t(1-\delta^*)\} - \ln Z_t.
\end{aligned}$$

補題 1 と α の定義より,

$$\begin{aligned}
&\Delta(\boldsymbol{u}, \boldsymbol{w}_t) - \Delta(\boldsymbol{u}, \boldsymbol{w}_{t+1}) \\
&\geq \alpha\{\gamma^* - y_t(1-\delta^*)\} \\
&\quad + \{y_t(1-\delta^*) - c\gamma^*\}\alpha - 4\delta^*\alpha^2 \\
&\geq (1-c)^2 \frac{\gamma^{*2}}{8\delta^*} - (1-c)^2 \frac{\gamma^{*2}}{16\delta^*} \\
&= (1-c)^2 \frac{\gamma^{*2}}{16\delta^*}.
\end{aligned}$$

$t = 1, \ldots, m$ の場合について相対エントロピーを足し合わせると,

$$\begin{aligned}
&\sum_{t=1}^{m} (\Delta(\boldsymbol{u}, \boldsymbol{w}_t) - \Delta(\boldsymbol{u}, \boldsymbol{w}_{t+1})) \\
&= \Delta(\boldsymbol{u}, \boldsymbol{w}_1) - \Delta(\boldsymbol{u}, \boldsymbol{w}_{m+1}) \\
&\geq \frac{m(1-c)^2\gamma^{*2}}{16\delta^*}.
\end{aligned}$$

したがって,

$$\begin{aligned}
m &\leq \frac{16\delta^* \Delta(\boldsymbol{u}, \boldsymbol{w}_1)}{(1-c)^2\gamma^{*2}} \\
&= \frac{16\delta^* \ln N}{(1-c)^2\gamma^{*2}}.
\end{aligned}$$

3.2 定数項が既知でマージンが未知の場合

次に，定数項 δ^* は分かっているが，マージン γ^* は分かっていない場合に適応した Winnow$_\delta$ を示す (図 2). Winnow$_{\delta^*}$ の誤り回数のオーダーは定数項，マージン共に分かっている場合の Winnow$_{\gamma^*,\delta^*}$ と同じである.

定理 3. 例の列 $S = ((\boldsymbol{x}_1, y_t), \ldots, (\boldsymbol{x}_T, y_T))$ が与えられたとする. $t = 1, \ldots, T$ のとき，$y_t(\boldsymbol{u} \cdot \boldsymbol{x} + 1 - \delta^*) \geq \gamma^*$ となるような $(\boldsymbol{u}, 1-\delta^*) \in \mathbb{R}^n \times \mathbb{R}$ が存在するならば，Winnow$_{\delta^*}$ の誤り回数は高々

$$m \leq \frac{256\delta^* \ln N}{3(1-c)^2\gamma^{*2}}$$

である. ただし，$\sum_i u_i = 1$ かつ $0 < \delta^* \leq 1$.

証明. $\gamma^* \geq \gamma_{n_0} \geq \frac{\gamma^*}{2}$ なる γ_{n_0} を考える. 定理 1 より，Winnow$_{\gamma_{n_0},\delta^*}$ の誤り回数は，高々 $\frac{16\delta^* \ln N}{(1-c)^2\gamma_{n_0}^2}$.
次に，$n = n_0$ となるまでの誤り回数を考える. $(\frac{1}{2})^{n_0} \leq \gamma^*$ より $n_0 \geq \log_2 \frac{1}{\gamma^*}$ なので，誤り回数は

$$m \leq \sum_{n=1}^{n_0-1} \frac{16\delta^* \ln N}{(1-c)^2\gamma_n^2} \leq \frac{64\delta^* \ln N}{3(1-c)^2\gamma^{*2}}$$

Winnow$_\delta$
begin
1. $n=1$ とする;
2. 以下の処理を繰り返す:
   (a) $\gamma_n=(\frac{1}{2})^n$ とする;
   (b) Winnow$_{\gamma_n,\delta}$ を実行する; もし Winnow$_{\gamma_n,\delta}$ が $\frac{16\delta\ln N}{(1-c)^2\gamma_n^2}$ 回以上間違えたなら Winnow$_{\gamma_n,\delta}$ の実行を中断し，$n=n+1$ とする; そうでないなら，終了する.
end.

図 2: δ^* のみが既知の場合の Winnow アルゴリズム

となる. よって学習が終わるまでの総誤り回数は,

$$\frac{16\delta^*\ln N}{(1-c)^2\gamma_{n_0}^2}+\frac{64\delta^*\ln N}{3(1-c)^2\gamma^{*2}}=\frac{256\delta^*\ln N}{3(1-c)^2\gamma^{*2}}$$

となる. (証明終わり)

3.3 マージンが既知で定数項が未知の場合

学習対象のマージンが既知で定数項が未知の場合に適応した Winnow$_\gamma$ を提案し (図 4), Winnow$_{\gamma^*}$ の誤り回数の上限のオーダーが，両パラメータが既知の場合の Winnow$_{\gamma^*,\delta^*}$ と同じ事を示す. 残念ながら，提案する手法ではマージンを近似的に最大化する性質を持たない.

この手法では，$\{0,1\}^N$ の事例空間に対して設計された Littlestone のオリジナルの Winnow を $\{-1,+1\}^N$ の事例空間用に修正したもの ($\{0,1\}^N$-Winnow$_\eta$ と呼ぶ) をサブルーチンとして用いる. 図 3 に詳細を示す.

まず，$\{0,1\}^N$-Winnow$_\eta$ の誤り回数を評価する. $\tilde{x}_t, i=\frac{x_{t,i}+1}{2}$, $\tilde{w}_{t,i}=w_{1,i}e^{\sum_{j=1}^{t-1}2\alpha y_j\tilde{x}_j}$ とおく. このとき，以下の関係が成り立っている.

$$\begin{aligned}
&\boldsymbol{w}_t\cdot\boldsymbol{x}_t+1-\theta_t\geq 0\\
\iff\quad & 2\boldsymbol{w}_t\cdot\tilde{\boldsymbol{x}}_t\geq\theta_t\\
\iff\quad & 2\boldsymbol{w}_t\cdot\tilde{\boldsymbol{x}}_t\geq\frac{1\cdot e^{2\alpha\sum_{j=1}^{t-1}y_j}}{\prod_{j=1}^{t-1}Z_j}\\
\iff\quad & \tilde{\boldsymbol{w}}_t\cdot\tilde{\boldsymbol{x}}_t\geq\frac{1}{2}
\end{aligned}$$

同様に，$\boldsymbol{w}_t\cdot\boldsymbol{x}_t+1-\theta_t\leq 0\iff\tilde{\boldsymbol{w}}_t\cdot\tilde{\boldsymbol{x}}_t\leq\frac{1}{2}$ が成り立つ. また，$\tilde{\boldsymbol{w}}_{t+1}=\tilde{\boldsymbol{w}}_te^{2\alpha y_t\tilde{x}_i}$ を満たす. 一方，$\boldsymbol{u}$ に関しては，$y_t=+1$ のとき，$\boldsymbol{u}\cdot\tilde{\boldsymbol{x}}\geq\frac{\delta^*+\gamma^*}{2}$, $y_t=-1$ のとき，$\boldsymbol{u}\cdot\tilde{\boldsymbol{x}}\geq\frac{\delta^*-\gamma^*}{2}$ が成り立つ. 以上から，事例空間 $\{-1,+1\}^N$ 上のアルゴリズム $\{0,1\}^N$-Winnow$_\eta$ は 事例空間 $\{0,1\}^N$ 上で 重みベクトル $\tilde{\boldsymbol{w}}$ およびしきい値 1 を用いたアルゴリズム，つまり Littlestone のオリジナルの Winnow アルゴリズムと等価である. よって, Littlestone の結果 [7] を用いると以下が成り立つ.

定理 4 ([7]). 例の列 $S=((\boldsymbol{x}_1,y_t),\ldots,(\boldsymbol{x}_T,y_T))$ が与えられたとする. $\tilde{\boldsymbol{x}}\in\{0,1\}^N$, $y_t\in\{-1,1\}$, $t=1,\ldots,T$ のとき,

$$\begin{aligned}
&\boldsymbol{u}\cdot\tilde{\boldsymbol{x}}_t\geq 1 \qquad (y_t=1)\\
&\boldsymbol{u}\cdot\tilde{\boldsymbol{x}}_t\leq 1-\frac{2\gamma^*}{\delta^*+\gamma^*} \quad (y_t=-1)
\end{aligned}$$

となるような $(\boldsymbol{u},\eta)\in\mathbb{R}^N\times\mathbb{R}$ が存在するならば，$\{0,1\}^N$-Winnow$_\eta$ の誤り回数は高々

$$m\leq\frac{7\ln N}{\eta\gamma^*}$$

である. ただし，$\sum_i u_i=1$ かつ $\eta\leq\frac{2\gamma^*}{\delta^*+\gamma^*}$.

定理 5. 例の列 $S=((\boldsymbol{x}_1,y_t),\ldots,(\boldsymbol{x}_T,y_T))$ が与えられたとする. $t=1,\ldots,T$ のとき，$y_t(\boldsymbol{u}\cdot\boldsymbol{x}+1-\delta)\geq\gamma^*$ となるような $(\boldsymbol{u},1-\delta^*)\in\mathbb{R}^N\times\mathbb{R}$ が存在するならば，Winnow$_{\gamma^*}$ の誤り回数は高々

$$m\leq\frac{14(\delta^*+\gamma^*)\ln N}{\gamma^{*2}}$$

である. ただし，$\sum_i u_i=1$ かつ $0<\delta^*\leq 1$.

$\{0,1\}^N$-**Winnow**$_\eta$

begin

1. $\boldsymbol{w}_1 = (1/n, \ldots, 1/n) \in \mathbb{R}^N$, $\theta_1 = 1$ and $m = 1$;
2. $t = 1, \ldots, T$ について, 以下の処理を繰り返す:
   (a) 事例 $\boldsymbol{x}_t \in \mathcal{X}$ を受け取る;
   (b) 予測 $\hat{y}_t = \mathrm{sign}(\boldsymbol{w}_t \cdot \boldsymbol{x}_t + 1 - \theta_t)$ を計算する;
   (c) ラベル $y_t \in \{-1, +1\}$ を受け取る;
   (d) $y_t(\boldsymbol{w}_t \cdot \boldsymbol{x}_t + 1 - \theta_t) < 0$ ならば, 以下のように更新:

$$w_{t+1,i} = \frac{w_{t,i} e^{\alpha y_t x_{t,i}}}{Z_t},$$

$$\theta_{t+1} = \frac{\theta_t e^{-y_t \alpha}}{Z_t},$$

   ただし, $\alpha = \eta/4$, $Z_t = \sum_{i=1}^n w_{t,i} e^{\alpha y_t x_{t,i}}$. 間違えなかった場合, $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t$ and $\theta_{t+1} = \theta_t$ とする.

end.

図 3: Littlestone 版 Winnow アルゴリズム (事例空間 $\{-1, +1\}^N$ 用に修正)

証明. $\eta^* = \frac{2\gamma^*}{\delta^* + \gamma^*}$ とし, $\eta^* \geq \eta_{n_0} \geq \frac{\eta^*}{2}$ なる η_{n_0} を考える. 定理 4 より, $\{0,1\}^N$-Winnow$_{\eta_{n_0}}$ の誤り回数は, 高々 $\frac{7 \ln N}{\eta_{n_0} \gamma^*} \leq \frac{14 \ln N}{\eta^* \gamma^*}$ である.
次に, $n = n_0$ となるまでの誤り回数を考える. $(\frac{1}{2})^{n_0} \leq \eta^*$ より $n_0 \geq \log_2 \frac{1}{\eta^*}$ なので, 誤り回数は

$$m \leq \Sigma_{n=1}^{n_0 - 1} \frac{7 \ln N}{\eta_n \gamma^*} \leq \frac{14 \ln N}{\eta^* \gamma^*}$$

となる. よって学習が終わるまでの総誤り回数は,

$$\frac{14 \ln N}{\eta^* \gamma^*} + \frac{14 \ln N}{\eta^* \gamma^*} \leq \frac{28 \ln N}{\eta^* \gamma^*} = \frac{14(\delta^* + \gamma^*) \ln N}{\gamma^{*2}}$$

となる. (証明終わり)

4 その他関連研究

その他の関連研究として [1, 8, 10] が挙げられる. [1] はエキスパート統合アルゴリズム Weighted Majority [9] と p-ノルムアルゴリズムについて, パラメータの自動化を行っている.

また, [8] はパラメータを自動化した Winnow アルゴリズムを提案している. これは, $x_{t,i} \in \{0,1\}$ が 0 のときと, 1 のときで別々の重み $w_{t,i}^+$, $w_{t,i}^-$ を定義して学習を行う. 学習時にはマージンといった値を必要としないが, 誤り回数は $O(\frac{1}{\gamma^2} \ln \frac{N}{\gamma})$ となる.

一方, [10] では, 今まで受け取った例すべてに対して定数項 $1 - \delta^*$ で γ^* 以上のマージンを持ち, 最もエントロピーが大きくなる 正規化重みベクトル $\boldsymbol{w}$ を用いることにより $O(\delta^* \ln N / \gamma^{*2})$ の誤り回数で仮説がマージン γ^* の重みベクトルに収束する事を示している (本論文の手法とは彼らの手法とは異なる). これは, 我々の手法と同じ誤り回数のオーダーであるが, γ^* と δ^* 両方が既知の場合しか議論されていない. 我々が示したようなどちらかが未知の場合について, 同じ誤り回数のオーダーとなるかは分からない.

5 まとめと今後の課題

本研究では, γ^* または δ^* のうち, どちらか一方が未知の場合における Winnow アルゴリズ

Winnow$_\gamma$
begin
1. $n=1$ とする;
2. 以下の処理を繰り返す:
   (a) $\eta_n=(\frac{1}{2})^n$ とする;
   (b) $\{0,1\}^N$-Winnow$_{\eta_n}$ を実行する; もし $\{0,1\}^N$-Winnow$_{\eta_n}$ が $\frac{7\ln N}{\eta_n\gamma}$ 回以上間違えたなら $\{0,1\}^N$-Winnow$_{\eta_n}$ の実行を中断し, $n=n+1$ とする; そうでないなら, 終了する.

end.

図 4: γ^* のみが既知の場合の Winnow アルゴリズム

ムのパラメータ自動設定手法を提案した. 特に, γ^* のみが未知の場合において, 定数 c を設定すれば $O(\frac{\delta^*\ln N}{(c-1)^2\gamma^{*2}})$ の誤り回数で学習し, 得られた仮説が事例集合 $\mathcal{X}$ に対して $c\gamma^*$ 以上のマージンを持つことを示した.

しかし, 残念ながら提案手法では両方のパラメータを自動設定することは現時点で達成できていない. 例えば単純に γ^* と δ^* の両方を探索する場合を考えよう: $\gamma_n=(\frac{1}{2})^n$ と $\delta_m=(\frac{1}{2})^m$ とし, Winnow$_{\gamma_n,\delta_m}$ が $O(\delta_n\ln N/\gamma_n^2)$ の誤り回数で終了する, つまり $\gamma_{n_0}<\gamma^*$, $\delta_{m_0}<\delta^*$ となるような n と m を探すとする. このとき, $\gamma^*\le\delta^*$ より, $n\ge m$ なる n, m をすべて調べると, $n=n_0=\lceil\log 1/\gamma^*\rceil$, $m=m_0=\lceil\log 1/\delta^*\rceil$ になるまでの総誤り回数は, $\sum_{n=1}^{n_0}(\sum_{m=1}^{n}O(\frac{\delta_m\ln N}{\gamma_n^2}))=O(\frac{\ln N}{\gamma^{*2}})$ となり, $O(\frac{\delta^*\ln N}{\gamma^{*2}})$ とはならない. 両方のパラメータが未知の場合に, $O(\frac{\delta^*\ln N}{\gamma^{*2}})$ の誤り回数で学習可能かどうかは未解決問題である.

また, 今回の手法では Winnow をリセットしながら実行しているが, リセットにより, 以前に得られた情報は失われてしまう. また, 事例が線型分離不可能な場合にはリセットし続けるためノイズに対して頑健ではない. したがって, リセットなしでパラメータの自動設定を行うことも課題である.

参考文献

[1] Peter Auer, Nicoló Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64:48–75, 2002.

[2] Claudio Gentile. A new approximate maximal margin classification algorithm. *Journal of Machine Learning Research*, 2:213–242, 2001.

[3] Claudio Gentile. The robustness of the p-norm algorithms. *Machine Learning*, 53(3):265–299, 2003.

[4] Adam J. Grove, Nick Littlestone, and Dale Schuurmans. General convergence results for linear discriminant updates. In *Proceedings of the tenth anual conference of Computational learning theory*, pages 171–183, 1997.

[5] J. Kivinen, M. K. Warmuth, and P. Auer. The percepotron algorithm versus winnow: linear versus logarithmic mistake bounds when few input variables are relevant. *Artificial Intelligence*, 97(1-2):325–343, 1997.

[6] Jyrki Kivinen. Online learning of linear classifiers. In *Advanced lectures on machine learning*, pages 235–257, 2003.

[7] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new

linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[8] Nick Littlestone and Chris Mesterharm. An apobayesian ralative of winnow. In *Advances in Neural Information Processing Systems 9*, pages 204–210, 1997.

[9] Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

[10] Philip M. Long and Xinyu Wu. Mistake bounds for maximum entropy discrimination. In *Advances in Neural Information Processing Systems 17*, pages 833–840, 2004.

[11] M. L. Minsky and S. A. Papert. *Perceptrons*. MIT Press, 1969.

[12] A. B. Novikoff. On convergence proofs on perceptrons. In *Symposium on the Mathematical Theory of Automata*, volume 12, pages 615–622. Polytechnic Institute of Brooklyn, 1962.

[13] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1959.