情報処理学会論文誌　数理モデル化と応用　Vol. 4　No. 3　84–93 (July 2011)

特徴の出現回数に応じたL_1正則化を実現する教師ありオンライン学習手法

大岩秀和[†1]　松島慎[†1]　中川裕志[†2]

オンライン学習（逐次学習）とは，訓練データを 1 つ受け取るたびに逐次的に学習を行う手法であり，大規模な訓練集合からの学習が効率化される．L_1 正則化とは，学習上不要なパラメータを零化する手法で，学習の高速化とメモリ効率の改善が期待される．2009 年に提案された FOBOS[7] は上記の 2 手法を組み合わせた，教師あり学習のための L_1 正則化付きオンライン学習手法である．しかし FOBOS では，特徴の出現回数が不均一な訓練集合では，低頻度の特徴が排除されやすい．FOBOS では特徴出現頻度やパラメータの累積更新幅とは無関係に全特徴に共通の零化を施すためである．しかし既存の L_1 正則化付きオンライン学習アルゴリズムでは，この性質は分析されてこなかった．本稿では，パラメータの累積更新幅に応じた L_1 正則化を構築する手法について述べるとともに，特徴の出現回数の情報を用いた L_1 正則化を導入した教師あり学習のためのオンライン学習手法（HF-FOBOS）を提案する．さらに，HF-FOBOS は既存手法と同等の計算コスト・収束速度でパラメータの累積更新幅に応じた L_1 正則化を実現する学習手法であることを確認する．また，HF-FOBOS と FOBOS に対して実問題に基づく実験を行い，出現頻度を利用した L_1 正則化が精度向上へ寄与することを示した．

L_1 Regularized Online Supervised Learning Using Feature Frequency

Hidekazu Oiwa,[†1] Shin Matsushima[†1]
and Hiroshi Nakagawa[†2]

Online learning is a method that updates parameters whenever it receives a single data. Online learning can learn efficiently from large data set. L_1 regularization is used for inducing sparsity into parameters and exclude unnecessary parameters. FOBOS[7] combines these two methods described above and presented a supervised online learning method with an efficient L_1 regularization. FOBOS has the property the parameters of low frequency features are zeros in a heterogeneous data set. However, this property is not analyzed enough in the field of online learning. In this paper, we presented a new online supervised learning method with L_1 regularization based on the number of occurrences of feature, named Heterogeneous Frequency FOBOS (HF-FOBOS). HF-FOBOS can solve optimization problems at same computational costs and convergence rate as FOBOS. Moreover, we examined the performance of our algorithms with classification tasks, and confirmed L_1 regularization based on the frequency of features improve accuracy.

1. はじめに

機械学習における教師あり学習では，入力データと出力データのペア事例を大量に集めた訓練集合を用いて，入力から出力への関数を学習する．教師あり学習は，株価予測やメールフィルタリング・画像認識等，様々な分野に応用されている．そのため，高精度かつ高速な教師あり学習のためのアルゴリズムが長年研究されており，現在にいたるまで様々な手法が提案されてきた．

2000 年頃からネットワーク技術やデータベース技術の急激な進歩により，訓練集合として利用可能なデータが爆発的に増大した．訓練集合の大規模化は，高精度な学習器を設計するうえで重要な役割を果たす．しかし大規模データを扱うには，現実的な計算時間で学習を終了させるために学習にかかる計算コストを考慮に入れることが重要である．また，訓練集合全体をメモリに一度に載せることが不可能な大規模なデータの場合，学習時に空間計算量の工夫も必要である．

オンライン学習は，データの大規模を考慮した学習手法の 1 つである．訓練集合全体が学習前に与えられるバッチ学習とは異なり，オンライン学習では，訓練集合中のデータが 1 つずつ逐次的に与えられ，そのたびに学習器を更新する．後に述べるようにオンライン学習は，空間計算量の削減が期待され，訓練集合が冗長なとき学習の高速化が可能になる．

また，学習器に汎化性能を持たせる正則化手法の中でも，推定精度の向上に寄与しないパラメータを零化させる L_1 正則化（Lasso 正則化）が注目されている．L_1 正則化では，学習に不要な特徴[⋆1]が排除され，高速かつ少ない作業領域での学習が実現できる．

†1 東京大学情報理工学系研究科
Graduate School of Informational Science and Technology, The University of Tokyo
†2 東京大学情報基盤センター
Information Technology Center, The University of Tokyo
⋆1 特徴とは，入力データをもとに構成される特徴ベクトル中の各次元のデータを指す．

そして，オンライン学習と L_1 正則化を同時に実現する Forward Backward Splitting（FOBOS）が文献 7) で提案された．FOBOS は，データが与えられるたびに学習器の精度を向上させるステップと L_1 正則化を施すステップを 1 回ずつ行う手法である．ただし，FOBOS の L_1 正則化では，全パラメータに対して共通の零化が施されている．そのため，低頻度の特徴に対応するパラメータは，学習への有用性と無関係に 0 になりやすい．そして，テキスト情報や画像情報を扱う場合，一般的に訓練集合中の特徴出現頻度は不均一である．

そこで，本稿では FOBOS を拡張した新たな L_1 正則化付きオンライン学習手法を提案する．提案手法は，特徴の出現回数の情報を用いた L_1 正則化を実現する．したがって，各特徴の出現回数が不均一な場合にも頻度情報に頑健な零化が可能になる．さらに，提案手法はパラメータ更新の計算コストが FOBOS と同程度であること，データ数の増加に応じてパラメータは最適解に収束することを示す．次に，実データに基づくテストデータを利用して提案手法と FOBOS の性能の比較と検証を行った．その結果，提案手法は全データセットで FOBOS よりも高い精度を達成し，予測の向上に出現頻度情報を利用することの重要性が示唆された．

本稿の構成は以下のとおりである．2 章では本稿で用いる記号の定義を行い，問題設定について解説する．3 章ではオンライン学習，4 章では L_1 正則化についてより詳細に議論する．5 章では，FOBOS のアルゴリズムとその性質について解説する．6 章では，特徴の出現回数の情報を用いた L_1 正則化を実現する手法を提案し，そのアルゴリズムと性質について述べる．7 章では，実データを用いた従来手法との比較実験を行い，その性能を比較する．最後に，8 章で結論を述べる．

2. 問題設定

本稿では，スカラーは小文字 λ，ベクトルは太字の小文字 $\mathbf{x}$，行列は太字の大文字 $\mathbf{X}$ で表記する．スカラーの絶対値は $|\lambda|$ で表記する．L_p ノルムは $\|\mathbf{v}\|_p$ と表記する．$\langle \mathbf{x}, \mathbf{y} \rangle$ はベクトル $\mathbf{x}$ と $\mathbf{y}$ の内積を表す．また $[a]_+$ は，

$$[a]_+ = \begin{cases} a & \text{if} \quad \mathrm{a} \geq 0 \\ 0 & \text{if} \quad \mathrm{a} < 0 \end{cases}$$

sign(a) は，sign(a) = a/|a|（ただし，$a = 0$ であれば sign(a) = 0）と表記される関数と定義する．本稿で使用する記号の定義は，**表 3** に列挙している．

本稿で対象とするのは，正則化項付きオンライン学習による教師あり学習である．教師あり学習の目的は，入力データ $\mathbf{x}$ と対応する出力データ y のペアが大量に含まれる訓練集合 $S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_T, y_T)\}$ を用いて，入力から出力への関数を設計することである．この関数は，入力ベクトルから決定される特徴ベクトル $\Phi(\mathbf{x}) \in \Re^n$ と重みベクトル $\mathbf{w} \in \Re^n$ の内積で定義される．したがって，出力データの予測値 $\hat{y}$ は $\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle$ で表す．

正則化項付きオンライン学習による教師あり学習では，入力と出力のペアが 1 組与えられるたびに重みベクトル $\mathbf{w}$ を逐次的に更新する．具体的には，次の手順で重みベクトルの更新が行われる．

(1)　t 番目の入力データ $\mathbf{x}_t$ を受け取る．
(2)　現在の重みベクトル $\mathbf{w}_t$ と特徴ベクトル $\Phi(\mathbf{x}_t)$ の内積を計算し，出力データの予測値 $\hat{y}_t$ を求める．
(3)　入力データ $\mathbf{x}_t$ に対応する真の出力データ y_t を受け取る．
(4)　予測値 $\hat{y}_t$ と真の値 y_t を用いて重みベクトルを $\mathbf{w}_{t+1}$ に更新する．
(5)　$t+1$ 番目のデータが存在する場合，(1) に戻る．

重みベクトルの更新基準は，損失関数 $\ell(\cdot) : \Re^n \to \Re_+$ と正則化項 $r(\cdot) : \Re^n \to \Re_+$ の和で定められる．損失関数 $\ell(\cdot)$ とは，重みベクトル $\mathbf{w}_t$ から導出される予測値 $\hat{y}_t$ が真の値 y_t から乖離しているほど，値が大きくなる関数である．損失関数には，二乗損失 $\ell_t(\mathbf{w}) = (y_t - \langle \mathbf{w}, \Phi(\mathbf{x}_t) \rangle)^2$，Hinge-Loss $\ell_t(\mathbf{w}) = [1 - y_t \langle \mathbf{w}, \Phi(\mathbf{x}_t) \rangle]_+$ 等が用いられる．正則化項は，重みベクトルの複雑さを表現する関数で平滑化の働きを持つ．正則化項には，L_1 ノルム $r(\mathbf{w}) = \lambda \|\mathbf{w}\|_1$ や L_2 ノルム $r(\mathbf{w}) = \lambda \|\mathbf{w}\|_2$ が用いられる．ここで，λ は損失関数と正則化項それぞれの学習への寄与度比を定めるスカラーである．

本稿で扱う損失関数は

$$\ell_t(\mathbf{w}) = \hat{\ell}_t(\langle \mathbf{w}, \Phi(\mathbf{x}_t) \rangle; y_t) = \hat{\ell}_t(\hat{y}_t; y_t) \tag{1}$$

で表現可能な関数に限定する．式 (1) が成立するとき損失関数の重みベクトルに関する勾配は，$\Phi(\mathbf{x}_t)$ のスカラー倍で表すことができる．さらに，損失関数は凸性を持つ関数に限定する．ここで，式 (2) を満たす関数 f を凸性を持つ関数と定義する．

$$\forall \mathbf{a}_1, \mathbf{a}_2 \in \Re^n \quad \forall \lambda \in [0, 1] \quad \lambda f(\mathbf{a}_1) + (1 - \lambda) f(\mathbf{a}_2) \geq f(\lambda \mathbf{a}_1 + (1 - \lambda) \mathbf{a}_2) \tag{2}$$

上であげた二乗損失や Hinge-Loss 等の損失関数はこれらの制約をすべて満たす．

正則化項付きオンライン学習における教師あり学習の目的は，学習過程で生じた損失関数と正則化項の総和を最小化することである．これを式で表すと，

$$\min_{\mathbf{w}_1,\mathbf{w}_2,\ldots} \sum_t \{\ell_t(\mathbf{w}_t) + r(\mathbf{w}_t)\} \tag{3}$$

となり，この値を最小化する最適な重みベクトル $\mathbf{w}_t^*$ の導出が目的となる．式 (3) は，各時点 t の重みベクトル $\mathbf{w}_t$ に関して，独立に $\ell_t(\mathbf{w}_t) + r(\mathbf{w}_t)$ を最小化することと同値である．つまり，各 t に関して $\ell_t(\cdot) + r(\cdot)$ の値が最小化される最適な重みベクトル $\mathbf{w}_t^*$ に更新する戦略を構築することがオンライン学習の目標である．

しかし，式 (3) から最適な戦略を設計することは不可能である．いかなる戦略を用いても式 (3) の損失関数 $\ell_t(\cdot)$ を更新後の重みベクトルに不適合な関数にすれば，式 (3) の上限を無限大へ発散させることが可能なためである．したがって，式 (3) の最小化問題からオンライン学習の戦略を評価することは非常に困難である．そこで，重みベクトル $\mathbf{w}_t$ の更新戦略の性能は Regret と呼ばれる概念で評価する．オンライン学習における T 個のデータを受け取った後の Regret は，式 (4) で定義される．

$$\begin{aligned}\mathrm{Regret}(T) &= \sum_{t=1}^{T} \{\ell_t(\mathbf{w}_t) + r(\mathbf{w}_t) - \ell_t(\mathbf{w}^*) - r(\mathbf{w}^*)\} \\ s.t. \quad \mathbf{w}^* &= \arg\min_{\mathbf{w}} \sum_{t=1}^{T} \{\ell_t(\mathbf{w}) + r(\mathbf{w})\}\end{aligned} \tag{4}$$

つまり，オンライン学習で生じた累積損失と訓練集合をすべて見た後に最適なパラメータを求めたときに生じた累積損失の差が Regret の定義である．

Regret 上限が $o(T)$ となるオンライン学習手法では，T の増加に応じて 1 データあたりの Regret 上限は減少し，0 に収束する．したがって，その手法で導出される重みベクトル $\mathbf{w}_t$ は，訓練集合全体に対する最適なパラメータ $\mathbf{w}^*$ への収束が保証される．さらに，Regret 上限が低いオーダで抑えられるならば，重みベクトルは高速に最適なパラメータへ収束することが示される．このことから，Regret 上限の最小化をオンライン学習の目的とおくことができる．

3. オンライン学習

オンライン学習とは入力と出力のペアが 1 組与えられるたびに逐次的に重みベクトル $\mathbf{w}$ を更新する手法である．したがって，オンライン学習では，単一のデータのみを用いてパラメータの更新が可能である．オンライン学習を用いる主な利点として，以下のことがあげられる．

- メモリ効率が良い．大規模な訓練集合から学習を行うとき，全データを一度にメモリ上に載せて最適なパラメータ $\mathbf{w}^*$ を求めることは，空間計算量の制約から現実的には非常に困難である．ただし，オンライン学習でパラメータの更新に必要とされるのは，単一データのみである．すなわち，訓練集合が大規模でも省メモリで学習が可能である．
- 再学習が容易である．一度最適な重みベクトルを設計した後に，新たな訓練集合の情報をパラメータに反映させることを再学習と定義する．バッチ学習では，過去のデータを含む全訓練集合を用いて，最適な重みベクトル $\mathbf{w}^*$ を再計算する必要がある．一方オンライン学習では，新しいデータのみからパラメータ更新が可能である．したがって，効率的な再学習が実現できる．
- 訓練集合が冗長なとき，学習の高速化が可能である．バッチ学習ではデータが冗長な場合も全訓練集合から最適な重みベクトル $\mathbf{w}$ を求める必要がある．一方，オンライン学習では逐次的に学習を行うため，以前と同質のデータを受け取ったときすでにそのデータに適合した学習器が構築されている場合が多く，学習が高速化される．

オンライン学習における最適化問題 (3) のうち，損失関数 $\ell_t(\cdot)$ と正則化項 $r(\cdot)$ が凸性を満たすものを逐次凸計画問題と呼ぶ．そして，逐次凸計画問題に対するオンライン学習手法として劣勾配法（Subgradient Method）[1] が提案されている．劣勾配法は，入力データと出力データのペアが 1 つ与えられるたびに損失関数と正則化項の劣勾配（Subgradient）を用いて，逐次的にパラメータを更新する手法である．ここで劣勾配とは，式 (5) を満足するベクトル $\mathbf{g} \in \Re^n$ を指す．

$$\forall \mathbf{y} \quad f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y} - \mathbf{x} \rangle \tag{5}$$

損失関数や正則化項に微分不可能な点が存在する場合であっても関数が凸であれば劣勾配は必ず存在する．劣勾配の集合を $\partial f(\mathbf{x})$ で表す．

劣勾配法は，損失関数と正則化項の和 $f_t(\mathbf{w}_t) = \ell_t(\mathbf{w}_t) + r(\mathbf{w}_t)$ の劣勾配集合 $\partial f_t(\mathbf{w}_t)$ から任意のベクトル $\mathbf{g}_t^f$ を用いて，式 (6) に従いパラメータを更新する手法である．

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{g}_t^f \quad s.t. \quad \mathbf{g}_t^f \in \partial f_t(\mathbf{w}_t) \tag{6}$$

ここで，$\eta_t \in \Re$ はパラメータの更新幅である．

式 (6) で示されているように劣勾配法では，$f_t(\cdot)$ の劣勾配と逆の方向に η_t の幅だけパラメータを更新している．したがって，劣勾配法は損失関数と正則化項の和が最小化される方向へ逐次的にパラメータを更新するアルゴリズムである．劣勾配法は Regret 上限が $o(T)$ で抑えられることが証明されている[12]．

その他の代表的なオンライン学習手法として，Perceptron [11]，Online Passive-Aggressive [3]，Confidence-Weighted Algorithms [4)–6)] 等が提案されている．

4. L_1 正則化

重みベクトルの推定では正則化が施されることが一般的である．正則化項を導入しない最適化問題を解くと訓練集合中のデータに過剰適合したパラメータが得られてしまい，新しいデータに対する予測精度が劣化するためである．このように，適切な汎化性能を持たない学習器が設計される問題を過学習（Over Fitting）と呼ぶ．過学習を抑止するため，平滑化の働きを持つ罰則項を最適化問題に導入する様々な手法が提案されてきた．

また，自然言語等の分野では，入力データの次元が数十万以上の実問題が多く存在する．膨大な次元数のデータからの学習は，計算コストも非常に大きくなる．そこで，L_1 正則化によるパラメータの零化の技術が正則化手法の援用として利用されている．L_1 正則化とは，最適化問題の正則化項に L_1 ノルムを導入したものである．

L_1 ノルムは，0 以外のいかなる点においても勾配の絶対値が一定である．したがって，L_1 ノルムを導入した最適化問題を解くと，損失関数の勾配がある値以下のパラメータはすべて 0 になる．このように，L_1 正則化は損失関数の最小化への寄与が小さいパラメータを排除する．パラメータを零化する性質は，正則化項に L_2 ノルムを導入する L_2 正則化では現れない．L_2 正則化ではパラメータの値が 0 に近づくと L_2 ノルムの勾配も 0 に収束することから，損失関数の勾配が L_2 ノルムの勾配よりも高速に 0 に収束しない限りパラメータの値は零化されないためである．

先に述べたように，L_1 正則化は学習上不要なパラメータを零化でき，重みベクトルは疎な形に変化する．このように，重みベクトルを疎化させる働きをスパース化と呼ぶ．重みベクトルが疎な形になれば，最適化時に考慮されるパラメータ次元数を圧縮できる．そのため，膨大な次元数のデータから学習を行うときに計算コストを削減することが可能になる．

5. Forward Backward Splitting（FOBOS）

オンライン学習と L_1 正則化を組み合わせるには，式 (3) の正則化項に L_1 正則化を導入し，劣勾配法を用いることが考えられる．ただし，この方法は，重みベクトルのスパース化の働きが損なわれる．上記の更新手法は，L_1 正則化による更新と損失関数による更新が同時に行われるため，L_1 ノルムの零化の働きが損失関数の劣勾配に妨害されるためである．

Forward Backward Splitting（以下，FOBOS）は上記の問題を解決し，オンライン学習と L_1 正則化をそれぞれの特性を損なうことなく組み合わせる学習手法を考案した．FOBOS では，L_1 ノルムによるスパース化を実現するため，重みベクトルの更新アルゴリズムを次の 2 ステップに分解している．

ステップ 1 は，損失最小化ステップである．ステップ 1 では，正則化項を除いた最適化問題を劣勾配法によって解く．

$$\mathbf{w}_{t+1/2} = \mathbf{w}_t - \eta_t \mathbf{g}_t^{\ell} \tag{7}$$

ここで，$\eta_t \in \Re_+$ は t 番目のデータにおける損失最小化ステップのステップ幅，$\mathbf{g}_t^{\ell} \in \partial \ell_t(\mathbf{w}_t)$ は，$\ell_t(\mathbf{w}_t)$ の任意の劣勾配である．ステップ幅 η_t は，t 番目のデータでの更新幅を調節する．

ステップ 2 は，L_1 正則化ステップである．ステップ 2 では，ステップ 1 で求めた重みベクトル $\mathbf{w}_{t+1/2}$ の変化に罰則を課すと同時に，L_1 ノルムで正則化を施す．

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} \left\{ \|\mathbf{w} - \mathbf{w}_{t+1/2}\|_2^2/2 + \eta_{t+1/2}\lambda\|\mathbf{w}\|_1 \right\} \tag{8}$$

$\eta_{t+1/2} \in \Re_+$ は L_1 正則化項のステップ幅である．$\eta_{t+1/2}$ は，L_1 正則化の寄与度を調整する．

ステップ 1 とステップ 2 からパラメータの第 j 成分に関する閉じた形の更新式が導出できる．

$$\begin{aligned} w_{t+1}^{(j)} &= \mathrm{sign}\left(\mathrm{w}_{\mathrm{t+1/2}}^{(\mathrm{j})}\right)\left[\left|\mathrm{w}_{\mathrm{t+1/2}}^{(\mathrm{j})}\right| - \eta_{\mathrm{t+1/2}}\lambda\right]_+ \\ &= \mathrm{sign}\left(\mathrm{w}_{\mathrm{t}}^{(\mathrm{j})} - \eta_{\mathrm{t}}\mathrm{g}_{\mathrm{t}}^{\ell,(\mathrm{j})}\right)\left[\left|\mathrm{w}_{\mathrm{t}}^{(\mathrm{j})} - \eta_{\mathrm{t}}\mathrm{g}_{\mathrm{t}}^{\ell,(\mathrm{j})}\right| - \eta_{\mathrm{t+1/2}}\lambda\right]_+ \end{aligned} \tag{9}$$

ここで，ベクトルの第 i 成分は $x^{(i)}$ と表記している．式 (9) の導出手順は，付録 A.1 に記載する．式 (9) より，$w_t^{(j)}$ から直接 $w_{t+1}^{(j)}$ が導出可能であることが確認できる．また，

$$w_{t+1}^{(j)} = \begin{cases} 0 & \text{if } \ |\mathrm{w}_{\mathrm{t+1/2}}^{(\mathrm{j})}| \le \eta_{\mathrm{t+1/2}}\lambda \\ w_{t+1/2}^{(j)} - \eta_{t+1/2}\lambda & \text{if } \ \mathrm{w}_{\mathrm{t+1/2}}^{(\mathrm{j})} > \eta_{\mathrm{t+1/2}}\lambda \\ w_{t+1/2}^{(j)} + \eta_{t+1/2}\lambda & \text{if } \ \mathrm{w}_{\mathrm{t+1/2}}^{(\mathrm{j})} < -\eta_{\mathrm{t+1/2}}\lambda \end{cases} \tag{10}$$

より，$|w_{t+1/2}^{(j)}| \le \eta_{t+1/2}\lambda$ となるパラメータ $w_{t+1/2}^{(j)}$ は，ステップ 2 ですべて 0 になる．

損失関数や正則化項が以下の条件を満たすとき，適切なステップ幅を設定すれば，定理 5.1 から FOBOS における Regret 上限 (4) が $o(T)$ となることが証明できる．定理 5.1 の証明は付録 A.2 で行う．本稿の定理 5.1 は，文献 7) のものとは異なる．

定理 5.1　損失関数と正則化項が凸性を持ち，$\forall \mathbf{w}_t \ \ \|\mathbf{w}_t - \mathbf{w}^*\|_2 \le D$ と，$\|\partial \ell_t\| \le G$,

$\|\partial r\| \leq G$ が成立するとき，η_t を任意の $c > 0$ を用いて $\eta_t = \eta_{t+1/2} = c/\sqrt{t}$ と設定すると，式 (11) が成立し FOBOS の Regret は $O(\sqrt{T})$ で上から抑えられる．

$$R_{\ell+r}(T) \leq 2GD + \left(D^2/2c + 8G^2c\right)\sqrt{T} = O(\sqrt{T}) \tag{11}$$

ここで，任意の関数 f に関して，$\|\partial f\| = \sup_{\mathbf{g}\in\partial f(\mathbf{w})} \|\mathbf{g}\|_2$ と定義している．

定理 5.1 の条件は，一般的な損失関数，正則化項に対して成立することが知られている[9]．例として，L_1 正則化項は $\|\partial r(\mathbf{w})\|^2 \leq \lambda^2$ で上から抑えられる．

FOBOS の主な特徴として，以下の 2 点があげられる．

- 劣勾配法等の既存手法と同等の計算コストで重みベクトルの更新が行える．つまり，L_1 正則化を導入しないオンライン学習手法と同じ計算量のオーダで，毎回のパラメータ更新が可能である．
- 劣勾配法等の既存手法と同じく，Regret 上限は $o(T)$ になる．つまり FOBOS は，ステップ幅を適切に設定してやれば，一般的な条件を満たす任意の損失関数で最適解に収束することが保証されている．

6. Heterogeneous Frequency FOBOS（HF-FOBOS）

5 章では，L_1 正則化を導入したオンライン学習手法である FOBOS について説明した．しかし，1 章で指摘したとおり，FOBOS では全パラメータ共通の零化を施していることが式 (10) から分かる．したがって，低頻度の特徴に対応するパラメータは，更新累積値そのものが小さいために高頻度の特徴に比べ 0 になりやすい．

例として，訓練集合中に出現頻度が 1/2 の特徴 a と 1/100 の特徴 b が共存する場合を考える．このとき，特徴 b は，毎回のパラメータの更新幅が

$$\eta_t |g_t^{\ell,(b)}| \geq \lambda \sum_{s=t}^{t+100} \eta_{s+1/2}$$

を満たさなければ，零化が必ず発生する．一方，特徴 a は毎回のパラメータの更新幅が

$$\eta_t |g_t^{\ell,(a)}| \geq \lambda \sum_{s=t}^{t+1} \eta_{s+1/2}$$

ならば，零化は必ずしも発生しない．つまり FOBOS では，特徴間で劣勾配の値が同一であったとしても出現頻度に違いがあると，零化の結果が変化する危険性がある．

実問題の多くは，訓練集合中の特徴出現頻度が不均一である．一例として，単語の出現頻度はべき乗則に従うことが経験則から知られている．この性質はジップ則と呼ばれ，自然言語から生成された実問題の場合，各特徴の出現頻度はべき乗則に従うと予想される．上記の性質を持つ実問題に対して，FOBOS は学習に有用な特徴のみを残すようにスパース化が行われるとは限らない．

6.1 HF-FOBOS のアルゴリズム

本稿の提案手法（HF-FOBOS）では，L_1 正則化に各特徴の出現頻度を利用することで FOBOS の改良を行った．HF-FOBOS では，高頻度な特徴に対応するパラメータが L_1 正則化時に大きな値で重み付けされる．そのため，低頻度の特徴が零化されやすい性質が改善されるように設計されている．

HF-FOBOS は，ステップ 1 のアルゴリズムは FOBOS と同様である．そして，ステップ 1 での毎回の更新幅 $\eta_t \mathbf{g}_t^\ell$ を用いて，ステップ 2 を式 (12) に変更する．

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} \left\{ \|\mathbf{w} - \mathbf{w}_{t+1/2}\|_2^2/2 + \lambda\eta_{t+1/2}\|\mathbf{H}_t\mathbf{w}\|_1 \right\} \tag{12}$$

ここで，

$$\mathbf{H}_t = \begin{pmatrix} h_{t,norm}^{(1)} & 0 & \dots & 0 \\ 0 & h_{t,norm}^{(2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & h_{t,norm}^{(n)} \end{pmatrix} \quad s.t.\ \ h_{t,norm}^{(j)} = \sqrt[p]{\sum_{s=1}^{t} \left(\eta_s g_s^{\ell,(j)}\right)^p}$$

とする．つまり $h_{t,norm}^{(j)}$ は，ステップ 1 での特徴 j の毎回の更新幅を並べたベクトルに関して L_p ノルムを計算した値であり，$\mathbf{H}_t$ は，各特徴の $h_{t,norm}^{(j)}$ の値を対角項に並べた行列である．行列 $\mathbf{H}_t$ と重みベクトル $\mathbf{w}_t$ の積をとると，重みベクトルの各成分が，対応する $h_{t,norm}^{(j)}$ によって重み付けされる．高頻度の特徴 j は，$g_s^{\ell,(j)}$ が非零である割合も高いため，$h_{t,norm}^{(j)}$ の値も大きくなる．したがって，高頻度の特徴 j には強い零化が作用する．

ベクトル $h_{t,norm}^{(j)}$ のパラメータ p を変化させることで，$h_{t,norm}^{(j)}$ を L_1 ノルム，L_2 ノルム，L_3 ノルム等，様々な値に置き換えることができる．パラメータ p を変化させると，行列 $\mathbf{H}_t$ の対角項の値も変化する．各ノルムにおける $h_{t,norm}^{(j)}$ の変化を**図 1** に示す．図 1 は，左から右へ $h_{t,norm}^{(j)}$ の値が大きい順に並べている．縦軸は，各ノルムにおける特徴 j の $h_{t,norm}^{(j)}$ の値である．図 1 から L_1 ノルムは出現頻度の変化に最も影響を受けることが分かる．一方，L_∞ ノルムは出現頻度には影響を受けにくく，最も FOBOS に近い．

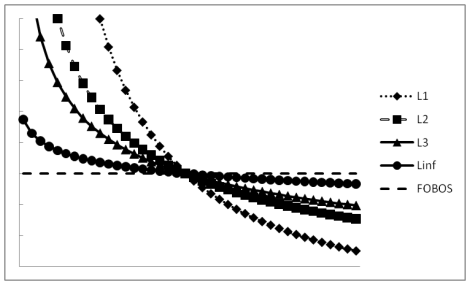

図 1　各ノルムにおける $h^{(j)}_{t,norm}$ の比較
Fig. 1　Comparison of the parameter of $h^{(j)}_{t,norm}$ when norms change.

HF-FOBOS の更新式は，FOBOS と同様の手順で導出可能である．

$$\begin{aligned} w^{(j)}_{t+1} &= \mathrm{sign}\left(\mathrm{w}^{(\mathrm{j})}_{\mathrm{t+1/2}}\right)\left[\left|\mathrm{w}^{(\mathrm{j})}_{\mathrm{t+1/2}}\right| - \eta_{\mathrm{t+1/2}}\mathrm{h}^{(\mathrm{j})}_{\mathrm{t,norm}}\lambda\right]_+ \\ &= \mathrm{sign}\left(\mathrm{w}^{(\mathrm{j})}_{\mathrm{t}} - \eta_{\mathrm{t}}\mathrm{g}^{\ell,(\mathrm{j})}_{\mathrm{t}}\right)\left[\left|\mathrm{w}^{(\mathrm{j})}_{\mathrm{t}} - \eta_{\mathrm{t}}\mathrm{g}^{\ell,(\mathrm{j})}_{\mathrm{t}}\right| - \eta_{\mathrm{t+1/2}}\mathrm{h}^{(\mathrm{j})}_{\mathrm{t,norm}}\lambda\right]_+ \end{aligned} \tag{13}$$

この更新式から，HF-FOBOS のパラメータ更新は FOBOS のパラメータ更新と同じ計算量オーダで実現できることが分かる．

6.2　HF-FOBOS の Regret

HF-FOBOS は正則化項が $r(\mathbf{w}) = \lambda\|\mathbf{w}\|_1$ から $r_t(\mathbf{w}) = \lambda\|\mathbf{H}_t\mathbf{w}\|_1$ に変化している．正則化項を重みベクトル $\mathbf{w}$ で偏微分して L_2 ノルムをとると，

$$\|\partial r_t\| = \lambda\sqrt{\sum_{k=1}^{n}\left(h^{(k)}_{t,norm}\right)^2} \tag{14}$$

が導出される．このことから，補題 6.1 が示される．

補題 6.1　$\|\partial\ell_t\| \le G$ が満たされるとき，$\eta_t = \eta_{t+1/2} = c/\sqrt{t}$ かつ $h^{(k)}_{t,norm}$ を $p > 2$ の L_p ノルムに設定したとき，式 (15) を満たす定数 U が必ず存在する．

$$\lim_{t\to\infty}\|\partial r_t\| < U \tag{15}$$

証明の詳細は，付録 A.3 で示す．

一方 $p \le 2$ の場合は適当な定数 V を用いて，行列 $\mathbf{H}_t$ の対角成分を $H^{(k,k)}_t = \min(h^{(k)}_{t,norm}, V)$ と定義し直す．ここで行列 $\mathbf{H}$ の第 i，j 成分を $H^{(i,j)}$ と表している．すると，$\|\partial r_t\| \le \sqrt{n}\lambda V$ より，$p \le 2$ の場合でも $\lim_{t\to\infty}\|\partial r_t\|$ が適当な定数 U で上から抑えられる．$\|\partial r_t\|$ の上限がある定数で抑えられるとき，定理 5.1 と同様の議論を用いて Regret 上限を求めることが可能である．したがって，定理 6.1 が示される．

定理 6.1　$\mathbf{H}_t$ の対角成分 $h^{(k)}_{t,norm}$ を L_p ノルムに設定したとき，

$$H^{(k,k)}_t = \begin{cases} \min(h^{(k)}_{t,norm}, V) & \text{if}\ \ \mathrm{p} \le 2 \\ h^{(k)}_{t,norm} & \text{if}\ \ \mathrm{p} > 2 \end{cases}$$

と定義する．損失関数と正則化項が凸性を持ち，かつ $\forall\mathbf{w}_t\ \ \|\mathbf{w}_t - \mathbf{w}^*\|_2 \le D$，$\|\partial\ell_t\| \le U$，$\|\partial r_t\| \le U$ が成立し，η_t を任意の定数 $c > 0$ を用いて $\eta_t = \eta_{t+1/2} = c/\sqrt{t}$ と設定する．このとき，補題 6.1 と同様に式 (16) が導出される．

$$R_{\ell+r}(T) \le 2UD + \left(D^2/2c + 8U^2c\right)\sqrt{T} = O(\sqrt{T}) \tag{16}$$

ここで，$\|\partial f(\mathbf{w})\| = sup_{\mathbf{g}\in\partial f(\mathbf{w})}\|\mathbf{g}\|_2$ と定義している．したがって，HF-FOBOS の Regret 上限は $O(\sqrt{T})$ になる．

証明の手順は，付録 A.2 と同様．したがって，HF-FOBOS の Regret 上限は $o(T)$ である．

7. 実　　験

実問題に基づくテストデータを用いて，本稿の提案手法（HF-FOBOS）に関する性能評価を行った．

今回行った実験の手法について説明する．実験には，Amazon.com のデータセットである[2),⋆1] books，dvds，ニュース記事に基づく 20 NewsGroups[8]（news20）のサブセット[⋆2]，Reuters-21578[10]（reut20）[⋆3]を用いた．

books，dvds は，Amazon.com のレビュー記事からなる評価分類のデータセットである．各レビュー文章から特徴ベクトルを生成し，その文章が製品に対して positive な意見を述べているか否かを判定するタスクである．

news20 は約 20,000 件，20 カテゴリのニュース記事からなるデータセットである．ニュース記事から各単語の tf-idf 値を並べたベクトルを生成し，そのベクトルを用いてその記事が属するカテゴリを判別するタスクである．本研究では，news20 のうち，ob-2-1，sb-2-1，

⋆1 http://www.cs.jhu.edu/∼mdredze/datasets/sentiment
⋆2 http://mlg.ucd.ie/datasets
⋆3 http://www.daviddlewis.com/resources/testcollections/reuters21578

表 2　各分類手法の正識別率 [%]・零特徴率 [%] 反復回数：20 回
Table 2　Precision and sparseness rate of the experiments (number of iterations: 20).

	HF-FOBOS $p=1$	HF-FOBOS $p=2$	HF-FOBOS $p=3$	HF-FOBOS $p=\infty$	FOBOS
books	85.23[1.52] (34.52)	**85.52**[1.24] (48.26)	85.14[1.33] (49.58)	85.05[1.41] (69.39)	84.98[1.61] (48.28)
dvds	82.49[1.68] (37.46)	84.75[1.75] (59.72)	**85.03**[2.28] (63.74)	84.02[1.66] (67.19)	83.91[1.55] (79.57)
ob-2-1	97.00[1.73] (42.78)	**97.10**[1.14] (56.73)	96.90[1.87] (59.03)	96.80[1.94] (59.78)	96.40[1.96] (49.23)
sb-2-1	**98.90**[0.83] (60.13)	98.40[0.80] (70.32)	98.40[1.11] (71.99)	98.10[1.14] (72.69)	97.20[1.78] (84.25)
ob-8-1	92.25[1.14] (62.83)	**93.10**[1.41] (62.84)	93.00[1.29] (64.64)	91.45[1.33] (77.78)	90.63[1.64] (87.90)
sb-8-1	90.90[1.72] (68.26)	92.55[1.85] (68.49)	**93.78**[2.44] (70.23)	91.25[1.44] (83.53)	90.53[1.61] (67.46)
reut20	95.23[0.65] (89.11)	**96.04**[0.56] (90.38)	95.91[0.55] (90.21)	94.80[0.67] (91.05)	95.53[0.63] (89.29)

表 1　データセットの概要
Table 1　Abstract of datasets.

	データ数	特徴次元数	カテゴリ数
books	4,465	332,441	2
dvds	3,586	282,901	2
ob-2-1	1,000	5,942	2
sb-2-1	1,000	6,276	2
ob-8-1	4,000	13,890	8
sb-8-1	4,000	16,282	8
reut20	7,800	34,488	20

ob-8-1，sb-8-1 の 4 つのサブセットを用いた．サブセットごとに，分類したいカテゴリの数や記事の精度が異なる．サブセットの名前の意味は以下のとおりである．1 文字目のアルファベットは ‘o’ は ‘overlapped’，‘s’ は ‘separeted’ を意味している．‘o’ の方が記事の精度が低い．2 文字目のアルファベットは，クラス間のデータ数の不均一性を表している．‘b’ は ‘balanced’ を意味し，クラス間でデータ数は均等である．最後の数字は，サブセット中に登場するクラスの種類数を表している．

Reuters-21578（reut20）も news20 と同じくニュース記事からなるデータセットである．本研究では，このコーパスから 20 クラスの分類を作成し，使用した．

表 1 に，各データセットの特徴次元数やデータ数，ニュースカテゴリの数（クラス数）を示す．

損失関数には Hinge-Loss を使用した．ただし，3 クラス以上の分類問題の場合は FOBOS，HF-FOBOS の更新式をそのまま適用することはできない．そこで本実験では，文献 3) と同様の手法で 3 クラス以上の分類問題に適用した．ステップ幅 η_t は，Regret の条件を満たすように $\eta_t = \eta_{t+1/2} = 1/\sqrt{t}$ とした．また，$p=1$，2 の HF-FOBOS では，$V=500$ に設定した[⋆1]．さらに，FOBOS，HF-FOBOS ともに損失関数と正則化項のそれぞれの学習への寄与度を調整するパラメータ λ を設定する必要がある．本実験では，性能評価の際にはデータセットを 10 分割した交差検定法を用いて，各手法において精度が最も高くなるパラメータ λ の値を選択した．各試行では，20 回反復を行った．

実験では，HF-FOBOS のパラメータ $p=1$，2，3，∞ と FOBOS の 5 種類の手法で実験を行い，精度と重みベクトル中の 0 要素の割合を比較した．

実験の結果をまとめた表が **表 2** である．表 2 には，各データセットで，交差検定法で求めた最適な λ の値で実験したときの分類精度を示す．この数値が高いほど，高精度な学習器が設計されていることを表している．角括弧 [·] 内の数値は標準偏差を示す．括弧 (·) 内の数値は，各データセットの各手法で λ を固定したときの重みベクトル中の 0 要素の割合を示す．また，各データセットで最高精度を達成した値は**太字**で示す．

表 2 より，全データセットに対して，HF-FOBOS は FOBOS よりも高い精度を示していることが確認できる．特に，$p=2$，3 の HF-FOBOS は全データセットにおいて一様に FOBOS からの精度向上が示された．この結果から，特徴の出現頻度情報を用いた L_1 正則化は精度の向上に寄与することが示唆される．

8. ま と め

本稿では，特徴の出現頻度情報を利用した L_1 正則化付きオンライン学習のための新しい数理モデルを提案した．既存の L_1 正則化付きオンライン学習では，訓練集合中の特徴の出

⋆1 今回の実験では，$\mathbf{h}^{(j)}_{t,norm}$ の値は 500 を超えなかったため，V の値は結果に影響しない．

現頻度が不均一な場合，低頻度の特徴が優先的に零化される性質を持つ．そこで，提案手法である HF-FOBOS では，各特徴の累積更新幅を用いた L_1 正則化を導入することで，頻度の低い特徴が優先的に排除されなくなり，特徴出現頻度や各特徴がとりうる値が不均一なデータに対しても不要な特徴のみを排除する適切なスパース化を可能にした．さらに本稿では，実データに基づくタスクを利用して，HF-FOBOS と FOBOS の性能比較を行った．その結果，HF-FOBOS は全データセットに対して FOBOS よりも高い精度を達成することを確認した．

謝辞　本稿の作成にあたり有益な助言をいただいた東京大学情報基盤センターの佐藤一誠助教に感謝いたします．

参 考 文 献

1) Bertsekas, D.P.: *Nonlinear Programming*, 2nd edition, Athena Scientific (1999).
2) Blitzer, J., Dredze, M. and Pereira, F.: Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification, *Proc. ACL-07, the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, CZ, pp.440–447, Association for Computational Linguistics (2007).
3) Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S. and Singer, Y.: Online Passive-Aggressive Algorithms, *Journal of Machine Learning Research*, Vol.7, pp.551–585 (2006).
4) Crammer, K., Dredze, M. and Kulesza, A.: Multi-class confidence weighted algorithms, *EMNLP '09: Proc. 2009 Conference on Empirical Methods in Natural Language Processing*, Morristown, NJ, USA, pp.496–504, Association for Computational Linguistics (2009).
5) Crammer, K., Fern, M.D. and Pereira, O.: Exact convex confidence-weighted learning, *Advances in Neural Information Processing Systems 22* (2008).
6) Dredze, M. and Crammer, K.: Confidence-weighted linear classification, *ICML '08: Proc. 25th International Conference on Machine Learning*, pp.264–271, ACM (2008).
7) Duchi, J. and Singer, Y.: Efficient Online and Batch Learning Using Forward Backward Splitting, *Journal of Machine Learning Research*, Vol.10, pp.2899–2934 (2009).
8) Lang, K.: Newsweeder: Learning to filter netnews, *Proc. 12th International Conference on Machine Learning*, pp.331–339 (1995).
9) Langford, J., Li, L. and Zhang, T.: Sparse Online Learning via Truncated Gradient, *J. Mach. Learn. Res.*, Vol.10, pp.777–801 (2009).
10) Lewis, D.D.: Reuters-21578.
11) Rosenblatt, F.: The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, *Psychological Review*, Vol.65, No.6, pp.386–408 (1958).
12) Zinkevich, M.: Online convex programming and generalized infinitesimal gradient ascent, *ICML*, pp.928–936 (2003).

付　　録

A.1　FOBOS の更新式の導出

式 (8) は重みベクトルの各次元に対して独立なので，更新式は式 (17) のように，次元ごとに分解可能である．

$$w_{t+1}^{(j)} = \arg\min_w \left\{ (w - w_{t+1/2}^{(j)})^2/2 + \eta_{t+1/2}\lambda|w| \right\} \quad (j = 1, 2, \ldots, n) \tag{17}$$

式 (17) は $w_{t+1/2}^{(j)}$ の符号が逆転すると $w_{t+1}^{(j)}$ も符号が逆転するため，$w_{t+1/2}^{(j)} \geq 0$ と仮定してよい．また，$w_{t+1/2}^{(j)} w_{t+1}^{(j)} < 0$ のとき，

$$\begin{aligned}(w_{t+1/2}^{(j)})^2/2 &< (w_{t+1/2}^{(j)})^2/2 + (w_{t+1}^{(j)})^2/2 - w_{t+1/2}^{(j)} w_{t+1}^{(j)} \\ &< (w_{t+1}^{(j)} - w_{t+1/2}^{(j)})^2/2 + \eta_{t+1/2}\lambda|w_{t+1}^{(j)}|\end{aligned} \tag{18}$$

より，式 (17) と矛盾する．したがって，$w_{t+1}^{(j)} \geq 0$ としても一般性を失わない．この条件を用いてラグランジュ未定乗数法を適用すると，

$$L(w, \alpha) = (w - w_{t+1/2}^{(j)})^2/2 + \eta_{t+1/2}\lambda w - \alpha w \tag{19}$$

となる．ここで，α はラグランジュ乗数である．これを解くと，式 (9) が導出される．

A.2　FOBOS，HF-FOBOS の Regret 上限の証明

FOBOS，HF-FOBOS の Regret 上限を導く下準備として，まず補題 A.2.1 を導く．補題 A.2.1 は，以降 Regret 上限の証明手順を議論する際に必要となる．

補題 A.2.1　損失関数と正則化項が凸性を持ち，さらに式 (20) を満たすとき，

$$\|\partial\ell(\mathbf{w})\|^2 \leq G^2, \|\partial r(\mathbf{w})\|^2 \leq G^2 \tag{20}$$

$\eta_{t+1} \leq \eta_{t+1/2} \leq \eta_t$ と $\eta_t \leq 2\eta_{t+1}$ が成立するようにステップ幅を設定すれば，式 (21) が成立する．

$$\begin{aligned}\forall \mathbf{w}^* \quad \exists c \leq 5 \quad & 2\eta_t\ell(\mathbf{w}_t) - 2\eta_t\ell(\mathbf{w}^*) + 2\eta_{t+1/2}r(\mathbf{w}_{t+1}) - 2\eta_{t+1/2}r(\mathbf{w}^*) \\ & \leq \|\mathbf{w}_t - \mathbf{w}^*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}^*\|_2^2 + 8\eta_t\eta_{t+1/2}G^2\end{aligned} \tag{21}$$

補題 A.2.1 の証明を行う．はじめに，$\ell(\cdot)$ が凸であるという条件から，任意の劣勾配

$\mathbf{g}_t^\ell \in \partial\ell(\mathbf{w}_t)$ に関して，次の式が成立する．

$$\ell(\mathbf{w}^*) \geq \ell(\mathbf{w}_t) + \langle \mathbf{g}_t^\ell, \mathbf{w}^* - \mathbf{w}_t \rangle \quad \Longrightarrow \quad -\langle \mathbf{g}_t^\ell, \mathbf{w}_t - \mathbf{w}^* \rangle \leq \ell(\mathbf{w}^*) - \ell(\mathbf{w}_t) \tag{22}$$

正則化項 $r(\cdot)$ に関しても，同様の条件式が成立する．ここで，正則化項の任意の劣勾配を $\mathbf{g}_t^r$ と表記する．

Cauchy-Shwartz の不等式と式 (20) から，

$$\begin{aligned}
\langle \mathbf{g}_{t+1}^r, \mathbf{w}_{t+1} - \mathbf{w}_t \rangle &= \langle \mathbf{g}_{t+1}^r, \mathbf{w}_{t+1/2} - \mathbf{w}_t - \eta_{t+1/2}\mathbf{g}_{t+1}^r \rangle \\
&= \langle \mathbf{g}_{t+1}^r, -\eta_t \mathbf{g}_t^\ell - \eta_{t+1/2}\mathbf{g}_{t+1}^r \rangle \\
&\leq \|\mathbf{g}_{t+1}^r\|_2 \|\eta_t \mathbf{g}_t^\ell + \eta_{t+1/2}\mathbf{g}_{t+1}^r\|_2 \\
&\leq \eta_{t+1/2}\|\mathbf{g}_{t+1}^r\|_2^2 + \eta_t \|\mathbf{g}_{t+1}^r\|_2 \|\mathbf{g}_t^\ell\|_2 \\
&\leq (\eta_{t+1/2} + \eta_t) G^2
\end{aligned} \tag{23}$$

が成立する．ここで 1 行目の式変形では，式 (8) を重みベクトルで偏微分した際に導出される式，$\mathbf{w}_{t+1} = \mathbf{w}_{t+1/2} - \eta_{t+1/2}\mathbf{g}_{t+1}^r$ を用いている．式 (23) を用いて，$\mathbf{w}^*$ と $\mathbf{w}_{t+1}$ の差の上限を求め，さらに $\ell(\mathbf{w}_t) + r(\mathbf{w}_t) - \ell(\mathbf{w}^*) - r(\mathbf{w}^*)$ の上限を導出する．まず，$\mathbf{w}^*$ と $\mathbf{w}_{t+1}$ の差の L_2 ノルムは以下のように展開できる．

$$\begin{aligned}
\|\mathbf{w}_{t+1} - \mathbf{w}^*\|_2^2 &= \|\mathbf{w}_t - (\eta_t \mathbf{g}_t^\ell + \eta_{t+1/2}\mathbf{g}_{t+1}^r) - \mathbf{w}^*\|_2^2 \\
&= \|\mathbf{w}_t - \mathbf{w}^*\|_2^2 - 2\left(\eta_t \langle \mathbf{g}_t^\ell, \mathbf{w}_t - \mathbf{w}^* \rangle + \eta_{t+1/2}\langle \mathbf{g}_{t+1}^r, \mathbf{w}_t - \mathbf{w}^* \rangle\right) \\
&\quad + \|\eta_t \mathbf{g}_t^\ell + \eta_{t+1/2}\mathbf{g}_{t+1}^r\|_2^2 \\
&= \|\mathbf{w}_t - \mathbf{w}^*\|_2^2 - 2\eta_t \langle \mathbf{g}_t^\ell, \mathbf{w}_t - \mathbf{w}^* \rangle + \|\eta_t \mathbf{g}_t^\ell + \eta_{t+1/2}\mathbf{g}_{t+1}^r\|_2^2 \\
&\quad - 2\eta_{t+1/2}\left(\langle \mathbf{g}_{t+1}^r, \mathbf{w}_{t+1} - \mathbf{w}^* \rangle - \langle \mathbf{g}_{t+1}^r, \mathbf{w}_{t+1} - \mathbf{w}_t \rangle\right)
\end{aligned} \tag{24}$$

第 3 項の上限は以下の式で求められる．

$$\begin{aligned}
\|\eta_t \mathbf{g}_t^\ell + \eta_{t+1/2}\mathbf{g}_{t+1}^r\|_2^2 &= \eta_t^2 \|\mathbf{g}_t^\ell\|_2^2 + 2\eta_t \eta_{t+1/2}\langle \mathbf{g}_t^\ell, \mathbf{g}_{t+1}^r \rangle + \eta_{t+1/2}^2 \|\mathbf{g}_{t+1}^r\|_2^2 \\
&\leq 4\eta_t^2 G^2
\end{aligned} \tag{25}$$

式 (24) の上限は，式 (22)，(23)，(25) を用いて，導出可能である．

$$\begin{aligned}
\|\mathbf{w}_{t+1} - \mathbf{w}^*\|_2^2 &\leq \|\mathbf{w}_t - \mathbf{w}^*\|_2^2 - 2\eta_t \langle \mathbf{g}_t^\ell, \mathbf{w}_t - \mathbf{w}^* \rangle - 2\eta_{t+1/2}\langle \mathbf{g}_{t+1}^r, \mathbf{w}_{t+1} - \mathbf{w}^* \rangle \\
&\quad + \|\eta_t \mathbf{g}_t^\ell + \eta_{t+1/2}\mathbf{g}_{t+1}^r\|_2^2 + 4\eta_{t+1/2}\eta_t G^2 \\
&\leq \|\mathbf{w}_t - \mathbf{w}^*\|_2^2 + 2\eta_t(\ell(\mathbf{w}^*) - \ell(\mathbf{w}_t)) \\
&\quad + 2\eta_{t+1/2}(r(\mathbf{w}^*) - r(\mathbf{w}_{t+1})) + 8\eta_t^2 G^2
\end{aligned} \tag{26}$$

式 (26) より，補題 A.2.1 が示された．

補題 A.2.1 を用いて，FOBOS，HF-FOBOS における Regret の上限を求める．逐次凸計画に対する Regret 分析は文献 12) が有名であり，本稿でも同様の手順によって FOBOS, HF-FOBOS の Regret 上限を求められる．

$\|\partial\ell_t\|$ と $\|\partial r\|$ の上限が G で抑えられるならば，補題 A.2.1 より $\eta_t = \eta_{t+1/2}$ のとき，

$$\begin{aligned}
&\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w}^*) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}^*) \\
&\quad \leq \frac{1}{2\eta_t}\left(\|\mathbf{w}_t - \mathbf{w}^*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}^*\|_2^2\right) + 4G^2\eta_t
\end{aligned} \tag{27}$$

t について 1 から T までの総和をとると，$r(\mathbf{w}) \leq r(\mathbf{0}) + G\|\mathbf{w}\|_2 \leq 2GD$ の条件から，

$$\begin{aligned}
R_{\ell+r}(T) &= \sum_{t=1}^T (\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w}^*) + r(\mathbf{w}_t) - r(\mathbf{w}^*)) + r(\mathbf{w}_{T+1}) - r(\mathbf{w}_1) \\
&\leq 2GD + \sum_{t=1}^T \frac{1}{2\eta_t}\left(\|\mathbf{w}_t - \mathbf{w}^*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}^*\|_2^2\right) + 4G^2 \sum_{t=1}^T \eta_t \\
&\leq 2GD + \frac{D^2}{2\eta_1} + \frac{D^2}{2}\sum_{t=2}^T \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + 4G^2 \sum_{t=1}^T \eta_t \\
&\leq 2GD + \frac{D^2}{2\eta_T} + 4G^2 \sum_{t=1}^T \eta_t
\end{aligned} \tag{28}$$

が導出される．ここで，3 番目の式変形では $\|\mathbf{w}_t - \mathbf{w}^*\|_2 \leq D$ を用いている．$\eta_t = c/\sqrt{t}$ とすれば $\sum_{t=1}^T \eta_t \leq 2c\sqrt{T}$ より，Regret の上限は $O(\sqrt{T})$ で抑えられることが示される．これで，定理 5.1 が証明された．

A.3　補題 6.1 の証明

$h_{t,norm}^{(k)}$ が L_p ノルムのとき，$\|\partial\ell_t\| \leq G$ が満たされるならば

$$\forall t, k \quad |g_t^{\ell,(k)}| \leq \|\mathbf{g}_t^\ell\|_2 \leq \|\partial\ell_t\| \leq G$$

より，

$$h_{T,norm}^{(k)} \leq \|G\boldsymbol{\eta}_T\|_p = G\|\boldsymbol{\eta}_T\|_p \tag{29}$$

が導出できる．ここで，$\boldsymbol{\eta}_T = (\eta_1, \eta_2, \ldots, \eta_T)$ である．$\|\boldsymbol{\eta}_T\|_p$ についてみると，

表 3　記号の定義
Table 3　Notation.

a	スカラー
$\mathbf{a}$	ベクトル
$\mathbf{A}$	行列
$a^{(i)}$	ベクトルの第 i 成分
$A^{(i,j)}$	行列の第 i,j 成分
$\lvert\lambda\rvert$	スカラーの絶対値
$\lVert\mathbf{a}\rVert_p$	L_p ノルム
$\langle\mathbf{a},\mathbf{b}\rangle$	ベクトルの内積
S	訓練集合
T	データ数
n	特徴次元数
$\mathbf{x}$	入力データ
y	出力データ
$\Phi(\mathbf{x})$	特徴ベクトル
$\mathbf{w}$	重みベクトル
$\hat{y}$	予測値
$\ell(\cdot)$	損失関数
$r(\cdot)$	正則化項
η_t	ステップ幅
λ	寄与度パラメータ
$\mathbf{g}$	劣勾配

$$\|\boldsymbol{\eta}_T\|_p = \sqrt[p]{\sum_{t=1}^{T}(\eta_t)^p} \tag{30}$$

である．$\eta_t = c/\sqrt{t}$ を代入すると，

$$\|\boldsymbol{\eta}_T\|_p = c\sqrt[p]{\sum_{t=1}^{T} t^{-p/2}} \tag{31}$$

が導出される．$\sum_{t=1}^{T} t^{-p/2}$ の部分はゼータ関数になっているため，$-p/2 < -1$ であれば，$\sum_{t=1}^{T} t^{-p/2}$ は上限値を持つ．この上限値を仮に d とおくと，$\|\boldsymbol{\eta}_T\|_p = cd^{1/p}$ となるため，

$$\|\partial r_t(\mathbf{w})\| \leq \lambda G\sqrt{\sum_{k=1}^{n}(cd^{1/p})^2} \leq U \tag{32}$$

となる定数 U が存在する．したがって，補題 6.1 が示される．一方で $p \leq 2$ の場合，ゼータ関数は無限大に発散するため，上限を示すことはできない．

(平成 23 年 2 月 7 日受付)
(平成 23 年 3 月 24 日再受付)
(平成 23 年 3 月 25 日採録)

大岩　秀和（学生会員）
2010 年東京大学工学部卒業．2010 年より東京大学大学院情報理工学系研究科修士課程に在籍中．

松島　　慎
2008 年東京大学工学部卒業．2010 年より東京大学大学院情報理工学系研究科博士課程在籍．2011 年より日本学術振興会特別研究員（DC2）．

中川　裕志（正会員）
1975 年東京大学工学部卒業．1980 年東京大学大学院博士課程修了．工学博士．1980 年より横浜国立大学勤務．1999 年より東京大学情報基盤センター教授．人工知能，自然言語処理，統計的機械学習の研究に従事．