6M-02

Top-k SVM の学習アルゴリズム

富井 和彦 * 竹内 勇気 * 宇敷 卓哉 * 加藤 毅 * ‡ §
Kazuhiko Tomii Yuki Takeuchi Takuya Ushiki Tsuyoshi Kato

1. はじめに

コンピュータビジョン分野では，公開データセットの整備が進むにつれ，多クラス分類におけるクラス数は増大し，これに伴って，**トップ k 誤分類率** [2] による性能評価が広く用いられるようになった．トップ k 誤分類率は次のように定義される．ある 1 個の未知データが与えられたとする．多クラス分類器は各クラスの予測スコア $s_1, \dots, s_m$ を算出する．m 個の予測スコアのうち，最も大きな k 個のスコアに対応するクラスの中に真のクラス $y \in \mathcal{Y} := \{1, \dots, m\}$ が含まれていればトップ k 分類成功，含まれていなければトップ k 誤分類とみなす．トップ k 誤分類率は，このような手順で評価した場合の評価用例題における誤分類の割合である．

従来，多クラス分類器の学習には **Max ヒンジ損失**が広く用いられてきたが，これはトップ k 誤分類率を最適化するための損失関数ではなかった．これに対し，Lapin ら [2] はトップ k 誤分類率を最適化できるようにトップ k ヒンジ損失

$$\Phi(\boldsymbol{s}\,;\,y) := \frac{1}{k}\max\left(0, \sum_{p=1}^{k}(\boldsymbol{s} - s_y\mathbf{1} + \mathbf{1} - \boldsymbol{e}_y)_{[p]}\right)$$

を設計した．しかし，本研究では，Lapin らが導出した理論に重大な誤りがあるために最適解に到達できていないことを突き止めた [3].

本研究の貢献は，以下の通りである：

- 文献 [2] に展開されている理論が誤っているため，その論文に記載されているアルゴリズムではトップ k ＳＶＭの最適解は得られないことを見つけた (この発見は文献 [3] で報告済み).
- **ブロック座標 Frank-Wolfe(BCFW)** 法の枠組みを適用することで，トップ k ＳＶＭのための新しい最適化アルゴリズムを開発した．また，各反復における計算量は $O(m(d+\log m))$，収束率は $O(n+(1/\lambda\epsilon))$ であることを導いた．
- 実データを用いた数値実験により，Lapin らのコードでは最適解に到達できないが，BCFW 法では最適解に到達できること，また，ナイーブな確率的座標上昇法より BCFW 法のほうが倍以上高速に最適化できることを示す．

2. 確率的座標上昇法によるトップ k ＳＶＭの学習

トップ k ＳＶＭを含む多くの学習問題は次式で表される目的関数 $P(\boldsymbol{W})$ の最小化問題に帰着される：

$$P(\boldsymbol{W}) := \frac{\lambda}{2}\|\boldsymbol{W}\|_{\mathrm{F}}^2 + \frac{1}{n}\sum_{i=1}^{n}\Phi(\boldsymbol{W}^\top\boldsymbol{x}_i\,;\,y_i) \qquad (1)$$

但し，$(\boldsymbol{x}_i, y_i) \in \mathbb{R}^d \times \mathcal{Y}$ は第 i 訓練用例題とし，例題数を n とした．この Φ にトップ k ヒンジ損失を採用するとトップ k ＳＶＭになる．$P(\boldsymbol{W})$ の最小化問題の**双対問題**は

$$D(\boldsymbol{A}) := -\frac{\lambda}{2}\|\boldsymbol{W}(\boldsymbol{A})\|_{\mathrm{F}}^2 - \frac{1}{n}\sum_{i=1}^{n}\Phi^*(-\boldsymbol{\alpha}_i\,;\,y_i) \qquad (2)$$

の最大化問題である．ただし，$\boldsymbol{\alpha}_i \in \mathbb{R}^m$ は双対変数 $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ の第 i 列であり，$\boldsymbol{W}(\boldsymbol{A}) := \boldsymbol{X}\boldsymbol{A}^\top/(\lambda n)$，$\boldsymbol{X} := [\boldsymbol{x}_1, \dots, \boldsymbol{x}_n]$ とした．$\Phi^*(\cdot\,;\,y_i)$ は $\Phi(\cdot\,;\,y_i)$ の**凸共役**である．$D(\boldsymbol{A})$ の最大化問題を解けば，$\boldsymbol{W}(\boldsymbol{A})$ により主問題の最適解を復元できる．Lapin ら [2] は，$D(\boldsymbol{A})$ の最大化のために**確率的座標上昇法**を採用した．即ち，各反復で $\boldsymbol{A}$ の列 $\boldsymbol{\alpha}_i$ を無作為に選び，

$$\Delta\boldsymbol{\alpha} \in \operatorname*{argmax}_{\Delta\boldsymbol{\alpha}\in\mathbb{R}^m} D(\boldsymbol{A} + \Delta\boldsymbol{\alpha}\boldsymbol{e}_i^\top) \qquad (3)$$

なる $\Delta\boldsymbol{\alpha}$ を求めて，$\boldsymbol{\alpha}_i \leftarrow \boldsymbol{\alpha}_i + \Delta\boldsymbol{\alpha}$ と更新していく反復法である．Lapin ら [2] の論文では，(3) を高速に求めるアルゴリズムを提案しており，もしもそのアルゴリズムが正しく (3) を満たす解を与えるなら，トップ k ＳＶＭは効率的に学習できる．

本研究では，トップ k ヒンジ損失の凸共役は $\forall\boldsymbol{v} \in \mathrm{dom}(\Phi^*(\cdot\,;\,y))$ に対して，$\Phi^*(\boldsymbol{v}\,;\,y) = v_y$ で与えられ，その**有効ドメイン** (effective domain) ¶は

$$\mathrm{dom}(\Phi^*(\cdot\,;\,y)) = \big\{\boldsymbol{v} \in \mathbb{R}^m \,\big|\, \langle\boldsymbol{v}, \mathbf{1}\rangle = 0, \quad \exists\beta \in \mathbb{R}\ \text{s.t.}\ \ \boldsymbol{v} + (\beta - v_y)\boldsymbol{e}_y \in \Delta_{k,m}\big\} \qquad (4)$$

となることを導いた [3]．一方，Lapin ら [2] は誤って有効ドメインを (4) より狭く導いたため，彼らのアルゴリズムは最適解に到達できないものになってしまっていた．

正しい有効ドメインで (3) を求めるには．**2 次計画問題**を解くことになる．これによって確かに最適解に到達できるようになるが，汎用ソルバーでは各反復の計算時間がかかりすぎてしまうため，スケーラビリティを損なってしまう．

3. ブロック座標 Frank-Wolfe(BCFW) 法の適用

本節では，$D(\boldsymbol{A})$ の最大化のために**ブロック座標 Frank-Wolfe(BCFW)** 法 [1] を適用する．BCFW 法は Algorithm 1 で表されるような反復法であり，各反復ごとに，**探索方向** $\boldsymbol{v}^{(t)} \in \mathbb{R}^m$ を求める問題と $\gamma_t \in \mathbb{R}$ を**直線探索**する問題を解く必要がある．

探索方向 $\boldsymbol{v}^{(t)} \in \mathbb{R}^m$ を求める問題は，$-\mathrm{dom}(\Phi^*(\cdot\,;\,y_i))$ は超多面体内であること，および目的関数が線形関数であることから，**線形計画問**

*群馬大学大学院理工学府
‡群馬大学次世代モビリティ社会実装研究センター (CRANTS)
§早稲田大学規範科学総合研究所 (IIRS)

¶拡張実数値関数 $f : \mathbb{R}^m \to \mathbb{R}\cup\{+\infty\}$ の有効ドメインは $\mathrm{dom}(f) := \{\boldsymbol{x} \in \mathbb{R}^m \,|\, f(\boldsymbol{x}) < +\infty\}$ と定義されている．

Algorithm 1 ブロック座標 Frank-Wolfe(BCFW) 法

begin
Let $\boldsymbol{A}^{(0)} \in \mathrm{dom}(-D)$;
for $t = 1, 2, \ldots$ **do**
  Pick $i \in \{1, \ldots, n\}$ at random;
  $\boldsymbol{v}^{(t)} \in \arg\max_{\boldsymbol{v} \in -\mathrm{dom}(\Phi^*(\cdot\,; y_i))} \left\langle \frac{\partial D(\boldsymbol{A}^{(t-1)})}{\partial \boldsymbol{\alpha}_i}, \boldsymbol{v} \right\rangle$;
  $\gamma_t := \underset{\gamma \in [0,1]}{\mathrm{argmax}}\, D\left(\boldsymbol{A}^{(t-1)} + \gamma(\boldsymbol{v}^{(t)} - \boldsymbol{\alpha}_i^{(t-1)})\boldsymbol{e}_i^\top\right)$;
  $\boldsymbol{A}^{(t)} := \boldsymbol{A}^{(t-1)} + \gamma_t(\boldsymbol{v}^{(t)} - \boldsymbol{\alpha}_i^{(t-1)})\boldsymbol{e}_i^\top$;
end for
end.

題になる．これに対し，著者らは，Φ がトップ k ヒンジ損失の場合，探索方向 $\boldsymbol{v}^{(t)} \in \mathbb{R}^m$ の問題の最適解集合は関数 $\Phi(\cdot\,; y_i)$ の $\boldsymbol{z}_i^{(t-1)} := \boldsymbol{W}(\boldsymbol{A}^{(t-1)})^\top \boldsymbol{x}_i$ における**劣微分** $\partial\Phi(\boldsymbol{z}_i^{(t-1)}\,; y_i)$ の符号を逆転した集合に一致する，すなわち，

$$\underset{\boldsymbol{v} \in -\mathrm{dom}(\Phi^*(\cdot\,; y_i))}{\mathrm{argmax}} \left\langle \frac{\partial D(\boldsymbol{A}^{(t-1)})}{\partial \boldsymbol{\alpha}_i}, \boldsymbol{v} \right\rangle = -\partial\Phi(\boldsymbol{z}_i^{(t-1)}\,; y_i)$$

を満たすことを発見した．

さらに，直線探索に関しても，区間 $[0,1]$ において 1 次元関数 $\gamma \mapsto D\left(\boldsymbol{A}^{(t-1)} + \gamma(\boldsymbol{v}^{(t)} - \boldsymbol{\alpha}_i^{(t-1)})\boldsymbol{e}_i^\top\right)$ を最大にする値は

$$\gamma_t = \mathrm{Clip}_{[0,1]}\left(\frac{\lambda n \left\langle \boldsymbol{e}_{y_i} - \boldsymbol{z}_i^{(t-1)}, \boldsymbol{v}^{(t)} - \boldsymbol{\alpha}_i^{(t-1)} \right\rangle}{\|\boldsymbol{x}_i\|^2 \|\boldsymbol{v}^{(t)} - \boldsymbol{\alpha}_i^{(t-1)}\|^2}\right)$$

のような閉形式で与えられることを見つけた．

よって，次の理論的結果を得た：

Theorem 1. トップ k ＳＶＭの双対目的関数 $D(\boldsymbol{A})$ を最大化するための BCFW 法の各反復は $O(m(d + \log m))$ で計算できる．

4. 収束解析

目的関数ギャップ (objective gap) $e(\boldsymbol{A}^{(t)}) := \max_{\boldsymbol{A}'} D(\boldsymbol{A}') - D(\boldsymbol{A}^{(t)})$ が ϵ 以下になるような解は ϵ **最適解**と呼ばれている．本研究では，3 節で述べた BCFW 法で，双対変数 $\boldsymbol{A}$ が双対問題の ϵ 最適解に到達する反復数の上限に関する理論的結果を得た．

Theorem 2. 各学習用例題に対して $\|\boldsymbol{x}_i\| \leq 1$ とする．$t_0 := \max(0, \lceil n \log(\lambda n/8) \rceil)$ とすると，$\forall t \geq t_0$ に対して，目的関数ギャップは

$$e(\boldsymbol{A}^{(t)}) \leq 32\lambda^{-1}/(2n + t - t_0) \qquad (5)$$

で抑えられる．

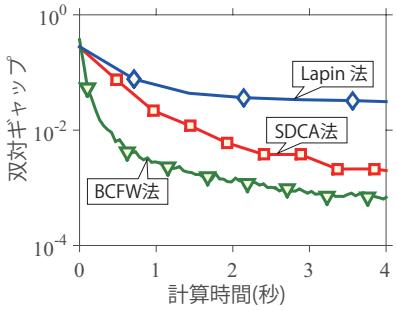

図 1: 3 手法の計算時間の比較.

Proof Sketch: トップ k 損失 $\Phi(\cdot\,; y)$ は，最大値ノルムに関してリプシッツ連続であり，そのリプシッツ係数は 2 であることを示すことができる．これより，任意の $s \in [0,1]$ に対して，不等式

$$e(\boldsymbol{A}^{(t)}) \leq (1 - s/n)e(\boldsymbol{A}^{(t-1)}) + 8s^2\lambda^{-1}n^{-2}$$

を得る．反復 $t = t_0$ に対しては $s = 1$ と置くことにより，(5) を得る．$t > t_0$ に対しては $s = 2n/(2n + t - t_0 - 1)$ とおいたとき反復 $(t-1)$ のとき (5) を満たすと仮定すると反復 t のときも (5) が成立するので，数学的帰納法により，題意を得る． □

よって，3 節で示した BCFW 法は**劣線形** (sub-linear) 収束することを示すことができた．

5. 数値実験

ベンチマーク用公開データセット 17 Flowers を使って，2 節の述べた確率的座標上昇法 (**SDCA 法**)，本論文で提案する BCFW 法，Lapin ら [2] のアルゴリズム (**Lapin 法**) の 3 種類の最適化アルゴリズムを比較した．Lapin 法には，Lapin が提供している Matlab コードを用いた．SDCA 法における各反復の 2 次計画問題は CPLEX で実装した．BCFW 法はすべて Matlab で実装した．図 1 は横軸を計算時間として**双対ギャップ** $e_{\mathrm{dg}}(\boldsymbol{A}) := P(\boldsymbol{W}(\boldsymbol{A})) - D(\boldsymbol{A})$ をプロットした．$\boldsymbol{A}$ がいかなる値でも双対ギャップは $e_{\mathrm{dg}}(\boldsymbol{A}) \geq e(\boldsymbol{A}) \geq 0$ を満たし，最適解では $e_{\mathrm{dg}}(\boldsymbol{A}) = 0$ となる．Lapin 法では $e_{\mathrm{dg}}(\boldsymbol{A}) = 0.028$ で停滞してしまっている．これは Lapin 法が誤った凸共役を用いたために，実行可能領域が狭く見積もられてしまい，その結果，最適解に至らなかったからである．SDCA 法では，双対ギャップが 10^{-2} に至るまで約 2.5 秒かかったのに対し，BCFW 法は約 0.5 秒で双対ギャップが 10^{-2} に達することができた．

謝辞： 本研究は JSPS 科研費 40401236 の助成を受けたものである．

参考文献

[1] Lacoste-Julien et al: Block-Coordinate Frank-Wolfe Optimization for Structural SVMs, *ICML*, pp. 53–61 (2013).
[2] Lapin et al.: Top-k Multiclass SVM, *NIPS 2015*, pp. 325–333 (2015).
[3] 竹内勇気，富井和彦，加藤毅ら：Top-k SVM 学習のための双対座標上昇法，FIT2017，第 2 分冊，pp. 269–270 (2017).