F-014

Top-k SVM 学習のための双対座標上昇法

竹内 勇気 † 富井 和彦 † 佐藤 貴亮 † 大沼 由弥 † 加藤 毅 † § ¶
Yuki Takeuchi Kazuhiko Tomii Takaaki Sato Yuya Onuma Tsuyoshi Kato

1. はじめに

近年，多クラス分類のベンチマーキングにおいて，クラス数が年々増加し，これに伴い，**トップ k 誤分類率**という性能評価指標がよく用いられている．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．クラス間の重なりが大きな多クラス分類タスクがいや増すばかりの今日，トップ k 誤分類率は理にかなった多クラス分類器の性能指標と言えよう．

2015 年に Lapin ら [1] はトップ k 誤分類率に特化して設計された多クラス分類器の学習算法を提案した．Lapin ら [1] の発表により，トップ k 誤分類率を最適化する問題は解決されたかに見えたが，本研究の理論的解析により，実は解決されていなかったことを発見した．Lapin らが導出した双対問題は誤っていた．本論文では，Lapin らの誤りを訂正し，正しい学習アルゴリズムを提示する．

2. トップ k SVM と Lapin ら [1] の誤り

d 次元 m クラスの多クラス分類器は $d \times m$ 行列 $\boldsymbol{W} := [\boldsymbol{w}_1,\dots,\boldsymbol{w}_m]$ をパラメータに持ち，入力ベクトル $\boldsymbol{x} \in \mathbb{R}^d$ に対し，m 次元の予測スコア $\boldsymbol{s} = \boldsymbol{W}^\top \boldsymbol{x}$ を出力するとする．

トップ k 誤分類率に対応する損失として，**トップ k *0/1* 損失** [1] がある．この損失は，トップ k 分類成功すれば 0，トップ k 誤分類すれば 1 と定義されている．Lapin ら [1] はトップ k 0/1 損失の上限となる**凸上限サロゲート損失** $\Phi_{\mathrm{tk}}(\boldsymbol{s}; y) := \phi_{\mathrm{tk}}(\boldsymbol{s} - s_y \mathbf{1}_m; y)$ を導入した．ここで，$\phi_{\mathrm{tk}}(;y) : \mathbb{R}^m \to \mathbb{R}$ は

$$\phi_{\mathrm{tk}}(\boldsymbol{z}; y) := \frac{1}{k} \max\left(0, \sum_{j=1}^{k} (\boldsymbol{z} + \mathbf{1} - \boldsymbol{e}_y)_{[j]}\right) \quad (1)$$

とした．ただし，x_j は任意のベクトル $\boldsymbol{x}$ に対し，$\boldsymbol{x}$ の要素の大きいほうから j 番目の要素の値である．Lapin ら [1] は $\Phi_{\mathrm{tk}}(\cdot; y)$ を**トップ k ヒンジ損失**と名付けた．この損失を使った学習器は**トップ k *SVM***と呼ばれる．n 個の訓練用例題 $(\boldsymbol{x}_i, y_i) \in \mathbb{R}^d \times \mathcal{Y}$ $(i = 1,\dots,n)$ に対し，学習算法の目的関数は，正則化項を加えたトップ k ヒンジ損失の平均

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

で与えられる．λ は正則化パラメータである．Lapin ら [1] は，$P_{\mathrm{tk}}(\boldsymbol{W})$ の最小解を見つけるため，**確率的双対座標上昇法** (SDCA) を採用した．SDCA は，**主問題**を直接解く代わりに，**双対問題**を解いて，**双対変数**の最適解から主変数 $\boldsymbol{W}$ を復元する方法である．

凸上限サロゲート損失の**凸共役**は双対問題を導くうえで鍵となる．トップ kSVM の双対問題は，**双対目的関数**

$$D_{\mathrm{tk}}(\boldsymbol{A}) := -\frac{\lambda}{2}\|\boldsymbol{W}(\boldsymbol{A})\|_{\mathrm{F}}^2 - \frac{1}{n}\sum_{i=1}^{n} \Phi_{\mathrm{tk}}^*(-\boldsymbol{\alpha}_i\,;\, y_i) \quad (3)$$

を最大化する双対変数 $\boldsymbol{A} := [\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_n] \in \mathbb{R}^{m\times n}$ を求める問題である．ただし，$\boldsymbol{W}(\boldsymbol{A}) := \frac{1}{n\lambda}\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{\alpha}_i^\top$ といた．$\Phi_{\mathrm{tk}}^*(\cdot; y_i)$ は $\Phi_{\mathrm{tk}}(\cdot; y_i)$ の凸共役である．(3) からわかるように，凸上限サロゲート損失の凸共役は正しく導出されなければ，誤った目的関数を最大化してしまうことになる．Lapin らは，トップ k ヒンジ損失の凸共役の導出にあたって，**トップ k 単体**

$$\Delta_{k,m} := \left\{\boldsymbol{x} \in \mathbb{R}_+^m \,|\, \|\boldsymbol{x}\|_1 \le 1,\, \boldsymbol{x} \le (\|\boldsymbol{x}\|_1/k)\,\mathbf{1}_m\right\}$$

を定義した．本研究では，トップ k ヒンジ損失の凸共役が次のようにあらわされることを新たに導出した：

$$\Phi_{\mathrm{tk}}^*(\boldsymbol{v}\,;\, y) = \begin{cases} v_y & \text{if } \langle \boldsymbol{v}, \mathbf{1}\rangle = 0 \text{ and} \\ & \quad \exists \beta \in \mathbb{R} \text{ s.t.} \\ & \quad\quad \boldsymbol{v} + (\beta - v_y)\boldsymbol{e}_y \in \Delta_{k,m} \\ +\infty & \text{o.w.} \end{cases}$$

この結果から，双対問題は

$$\begin{aligned} \max \quad & -\frac{\lambda}{2}\|\boldsymbol{W}(\boldsymbol{A})\|_{\mathrm{F}}^2 + \frac{1}{n}\sum_{i=1}^{n} \alpha_{y_i,i} \\ \text{wrt} \quad & \boldsymbol{A} = [\alpha_{j,i}]_{j,i} = [\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_n] \in \mathbb{R}^{m\times n}, \\ & \boldsymbol{\beta} = [\beta_1,\dots,\beta_n]^\top \in \mathbb{R}^n, \\ \text{subject to} \quad & \forall i, \quad \langle \boldsymbol{\alpha}_i, \mathbf{1}\rangle = 0, \\ & \quad (\alpha_{y_i,i} - \beta_i)\boldsymbol{e}_{y_i} - \boldsymbol{\alpha}_i \in \Delta_{k,m} \end{aligned}$$

のように導かれる．この双対問題の最適解を $\boldsymbol{A}_\star$ とすると，$P_{\mathrm{tk}}(\boldsymbol{W})$ の最小解は $\boldsymbol{W}(\boldsymbol{A}_\star)$ で与えられる．Lapin らは凸共役の導出の時点で誤っており，彼らの双対問題は $\boldsymbol{\beta} = \mathbf{0}$ に固定したものになっていた．よって，Lapin らが誤って導いた双対問題の実行可能領域は，正しい実行可能領域より狭いことになる（図 1(a)）．この図のように，最適解の集合と Lapin らの実行可能領域との共通集合がない場合，Lapin らの算法では最適解に到達することは不可能になる．

†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センター (CRANTS)
¶早稲田大学規範科学総合研究所 (IIRS)

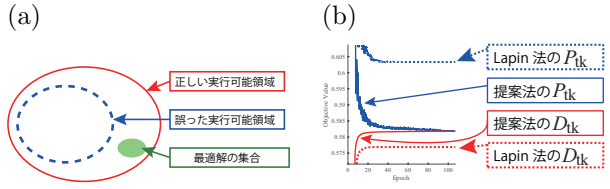

図 1: (a) Lapin ら [1] の実行可能領域．(b) 数値例．

3. 正しい SDCA 法

本研究でも，双対問題を SDCA 法で解くことにする．SDCA 法では，反復 t において，$\boldsymbol{A}$ に含まれる n 列から，無作為に選んだ第 i 列 $\boldsymbol{\alpha}_i$ のみ更新する．すなわち，第 $(t-1)$ 反復において，更新済みの双対変数の値を $\boldsymbol{A}^{(t-1)} = \left[\boldsymbol{\alpha}_1^{(t-1)}, \ldots, \boldsymbol{\alpha}_n^{(t-1)}\right]$ とすると，第 t 反復において $\boldsymbol{\alpha}_i^{(t)}$ を計算し，$\boldsymbol{A}^{(t)} = \boldsymbol{A}^{(t-1)} + (\boldsymbol{\alpha}_i^{(t)} - \boldsymbol{\alpha}_i^{(t-1)})\boldsymbol{e}_i^\top$ のように第 i 列を更新する．$\boldsymbol{\alpha}_i^{(t)}$ の計算は，部分問題

$$\begin{aligned} \min \quad & \frac{1}{2}\|\boldsymbol{\alpha}_i\|^2 + K_{i,i}^{-1} n\lambda \left\langle \boldsymbol{\alpha}_i, \bar{\boldsymbol{z}}_i^{(t)} - \boldsymbol{e}_{y_i} \right\rangle \\ \text{wrt} \quad & \boldsymbol{\alpha}_i \in \mathbb{R}^m, \quad \beta_i \in \mathbb{R}, \\ \text{subject to} \quad & \langle \boldsymbol{\alpha}_i, \mathbf{1} \rangle = 0, \quad (\alpha_{y_i,i} - \beta_i)\boldsymbol{e}_{y_i} - \boldsymbol{\alpha}_i \in \Delta_{k,m} \end{aligned}$$

で与えられる $(m+1)$ 変数の線形制約 2 次計画問題に帰着される．ここで，$K_{i,i} := \|\boldsymbol{x}_i\|^2$ とし，$\bar{\boldsymbol{z}}_i^{(t)} := \boldsymbol{W}(\boldsymbol{A}^{(t-1)} - \boldsymbol{\alpha}_i^{(t-1)}\boldsymbol{e}_i^\top)^\top \boldsymbol{x}_i$ とした．

4.Lapin の実装

Lapin は `libsdca` というプログラムを公開している．現在公開されているコード https://github.com/mlapin/libsdca/commit/fd5c1f1 を解析してみると，主目的関数において，トップ k ヒンジ損失 $\Phi_{\mathrm{tk}}(\cdot; y)$ ではなく，Lapin らが誤って導出したトップ k ヒンジ損失の凸共役の凸共役 $\Phi_{\mathrm{ptk}}(\cdot; y)$ が実装されていた．すなわち，$\boldsymbol{z}^{\setminus y}$ を $\boldsymbol{z} \in \mathbb{R}^m$ から第 y 要素だけを取り除いたベクトル，また，

$$\phi_{\mathrm{ptk}}(\boldsymbol{z}; y) := \frac{1}{k} \max\left(0, \sum_{j=1}^{k} (\boldsymbol{z}^{\setminus y} + \mathbf{1})_{[j]}\right)$$

として，$\Phi_{\mathrm{ptk}}(\boldsymbol{s}; y) := \phi_{\mathrm{ptk}}(\boldsymbol{s} - s_y\mathbf{1}; y)$ が計算され，これを使って，

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

の値が出力されるようになっている．本論文では，トップ k ヒンジ損失 $\Phi_{\mathrm{tk}}(\cdot; y)$ と区別するため，$\Phi_{\mathrm{ptk}}(\cdot; y)$ を **Lapin 損失**と呼ぶことにする．Lapin 損失は，

$$\forall \boldsymbol{s} \in \mathbb{R}^m, \forall y \in \mathcal{Y}, \quad \Phi_{\mathrm{tk}}(\boldsymbol{s}; y) \geq \Phi_{\mathrm{ptk}}(\boldsymbol{s}; y) \qquad (4)$$

の性質があり，このため，$\forall \boldsymbol{W} \in \mathbb{R}^{d\times m}$ に対して，$P_{\mathrm{tk}}(\boldsymbol{W}) \geq P_{\mathrm{ptk}}(\boldsymbol{W})$ となる．ここからいえることは，

表 1: 平均 Top-k 誤分類率 (%)．

Caltech 101	$k=2$	$k=3$	$k=4$	$k=5$
トップ-k ヒンジ	**31.82**	**25.93**	**22.88**	**20.69**
Lapin 損失	31.83	25.96	22.91	20.73
Coli 100	$k=2$	$k=3$	$k=4$	$k=5$
トップ-k ヒンジ	**1.14**	**0.80**	**0.61**	**0.48**
Lapin 損失	1.17	0.82	**0.61**	0.49
Oxford	$k=2$	$k=3$	$k=4$	$k=5$
トップ-k ヒンジ	6.98	**5.14**	**4.11**	**3.36**
Lapin 損失	**6.97**	**5.14**	4.12	**3.36**

Lapin の実装はトップ k SVM の目的関数 $P_{\mathrm{tk}}(\boldsymbol{W})$ ではなく，$P_{\mathrm{ptk}}(\boldsymbol{W})$ を SDCA 法で解いてしまっていることになる．しかし，$\Phi_{\mathrm{ptk}}(\boldsymbol{s}; y)$ はすでにトップ k 0/1 損失の凸上限サロゲート損失ではないため，Lapin の実装は大義を喪失している．

5. 数値実験

最適性の確認　図 1(b) に，提案法と Lapin の実装の数値例を示す．提案法が生成する $\{\boldsymbol{A}^{(t)}\}_{t=1}^T$ から計算される $P_{\mathrm{tk}}(\boldsymbol{W}(\boldsymbol{A}^{(t)}))$ および $D_{\mathrm{tk}}(\boldsymbol{A}^{(t)})$ を実線でプロットし，Lapin の実装が生成する $\{\boldsymbol{A}^{(t)}\}_{t=1}^T$ から計算される $P_{\mathrm{tk}}(\boldsymbol{W}(\boldsymbol{A}^{(t)}))$ および $D_{\mathrm{tk}}(\boldsymbol{A}^{(t)})$ を点線でプロットした．提案法の**双対ギャップ** $(P_{\mathrm{tk}}(\boldsymbol{W}(\boldsymbol{A}^{(t)})) - D_{\mathrm{tk}}(\boldsymbol{A}^{(t)}))$ は 0 に収束していることから，最適解に収束したことが確認できる．一方，Lapin 実装は双対ギャップを大きく残したまま停滞してしまっている．これは最適解に至らなかったことを意味する．これは，最適解は $\boldsymbol{\beta} \neq \mathbf{0}$ ではないためであり，最適解の集合が Lapin らの実行可能領域との共通集合に入らないことを示唆しており，それ故に Lapin らの理論の誤りが深刻であることを証拠づけている．

パターン認識性能　3 個の画像データセット Caltech 101, Coli 100, Oxford を使って，Lapin 損失と比較した．それぞれで，訓練用/評価用を無作為に分割して得られるトップ k 誤分類率を計算した．これを 3 回繰り返してえられる 3 個のトップ k 誤分類率の平均を表 1 に示す．トップ k ヒンジ損失を正確に計算した提案法は，Lapin 損失と competitive な汎化性能を得た．

6. 結論

Lapin ら [1] は，トップ k ヒンジ損失を提案し，これに基づいた学習算法を提示した．本研究では，その学習算法に理論的な誤りがあることを発見し，実際に数値実験でも最適解に至らないことを示した．本論文では，誤りを訂正した新たな算法を示し，その算法は数値実験により最適解に到達できることを確認した．

謝辞：本研究は JSPS 科研費 26249075, 40401236 の助成を受けたものである．

参考文献

[1] M. Lapin et al. Top-k multiclass SVM. In *NIPS 2015*, pages 325–333, 2015.