Vol.2014-NL-216 No.10
Vol.2014-SLP-101 No.10
2014/5/22

条件付きロジスティック分布を用いた重み付き多タスク学習

濱口 拓男[1,a)]　新保 仁[1,b)]　松本 裕二[1,c)]

概要：NLP における多くの問題は，クラス分類として定式化される．Multi-Task Feature Learning(MTFL) は，クラス分類や回帰問題といったタスクを複数同時に学習することで，タスク全体の精度を改善する多タスク学習の一種である．しかし MTFL は全てのデータがどれか 1 つのタスクに所属している事を仮定しており，データがどのタスクに所属するかが不明瞭な場合や，複数のタスクに所属している場合には適用できなかった．本論文では，条件付きロジスティック分布という考えを用いることで，そのような状況でも MTFL を適用できる拡張手法を提案する．我々の方法はタスクの情報が無い場合でも，元々の MTFL の精度とほぼ同等の精度を実現する．

キーワード：多タスク学習，ロジスティック回帰，条件付きロジスティック分布，Multi-task Feature Learning

1. はじめに

1.1 研究の背景

テキストや画像などのデータが与えられた時，そのデータがどのクラスに所属するかを推定する問題はクラス分類と呼ばれる．自然言語処理においてもクラス分類は重要である．与えられた文の意味がポジティブなものかネガティブなものかを推定する問題や，ニュースの内容がスポーツや経済, 国際問題といったカテゴリィのどれに所属するかを推定する問題などはクラス分類の例である．他にも品詞タグ付けや固有表現抽出・感情推定など，多くの場面で用いられている．その為，クラス分類の精度改善は重要なテーマとなっており，様々なモデルが研究されている．

Argyriou らによって提案された Multi-Task Feature Learning [1](MTFL) は，そのようなクラス分類を扱うことのできるモデルである．MTFL は多タスク学習の一種であり，複数のタスクを組み合わせることで性能の向上を実現している．通常，クラス分類や回帰問題では，1 つのロジスティック回帰やリッジ回帰のモデルを用いて学習を行う．それに対し MTFL は，それらモデルを 1 つのタスクとして考え，複数のモデルを同時に学習する．この時，個々のタスクを独立して学習するのではなく，タスクのパラメータを正則化項で関係付けることによって，過学習を抑制しながらタスクに特化した特徴を捉えることができる．

多タスク学習には多くのモデルが存在する．共通する点は，複数のモデルを何らかの形で組み合わせて学習することで，性能の向上を期待している点である．例えば品詞タグ付けを 1 つのタスク，固有表現抽出を 1 つのタスクとして考えた場合，この 2 つを独立に学習するのではなく，各タスクのパラメータに何らからの相互依存関係をもたせたせると精度が向上すると期待するのが多タスク学習の基本的な姿勢である．

また MTFL との関係性を見出すことができる手法も存在する．例えばガウス過程に事前分布を考えた [7] や，t 過程を用いる [5]，タスク間の関係を学習できる [9]，行列値正規分布を仮定する [8]，タスクに潜在的なパラメータを考えた [6] などである．他にもニューラルネットに基づいたモデル [3] [4] も存在する．

1.2 研究の目的

前述のように，MTFL は複数のタスクを組み合わせる手法である．しかし MTFL はその手法の性質上，モデルを適用できない場合が考えられる．例えば推定したい出力が 1 つなのに対し，2 つ以上の出力結果が推定される場合である．

本論文では，まずこの点に関して MTFL の定義を振り返りながら考察を行う．次に条件付きロジスティック回帰を導入することで，複数の結果の重み付き和を計算できる手

[1] 奈良先端科学技術大学院大学
Nara Institute of Sience and Technology
[a)] takuo-h@is.naist.jp
[b)] shimbo@is.naist.jp
[c)] matsu@is.naist.jp

法を提案する．この手法を応用することにより，テスト時にタスクの情報が存在しない場合においても，既存の手法の精度を落とさず扱うことが可能になる．

本論文の構成を述べる．まず本章の残りでクラス分類の用語の整理と，数式や関数の定義を行う．次の 2 章ではロジスティック回帰を正則化を含めて振り返る．3 章ではタスクの定義に注目しながら MTFL に関して述べ，4 章では，MTFL で扱えない場合を事例を挙げながら考察を行う．そして 5 章では，4 章で指摘した点を解決する手法を提案する．6 章では提案手法と既存手法との比較実験を行った結果について述べる．そして最後の章で結論を述べた．

1.3 基本的な数式の表記

ここでは，本論文全体で用いる数式や確率分布の記法について定義をする．最初に簡単な定義として $\mathbb{R}$ を実数とし，自然数 T に対して $\mathbb{N}_T \coloneqq \{1,2,,,T\}$ とする．またベクトルと行列をボールド体で表す．

d 次元のベクトル $\mathbf{x}$ があった時，その i 番目の次元の要素を $\mathbf{x}_{[i]}$ とし，p ノルムを $\|\mathbf{x}\|_p = (\sum_{i=1}^{d} \|\mathbf{x}_{[i]}\|^p)^{\frac{1}{p}}$ と定義する．

ただし $|a|$ はスカラ a の絶対値である．特別に言及が無い場合は $\|\mathbf{x}\| = \|\mathbf{x}\|_2$ であるとする．同じ次元のベクトル $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^d$ に対し内積を $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{d} \mathbf{x}_{[i]} \mathbf{y}_{[i]}$ で定義する．

次元が $d \times d$ である単位行列と直交行列をそれぞれ, $\mathbb{I}_d$, O_d と表記する．次元数を表す d は曖昧性がない場合は省略することがある．正方行列 $\mathbf{A} \in \mathbb{R}^{d \times d}$ が与えられた時，その転置行列・逆行列・擬逆行列をそれぞれ $\mathbf{A}^{\mathrm{T}}$, $\mathbf{A}^{-1}$, $\mathbf{A}^{+}$ で表記する．またトレースを $\mathrm{tr}[\mathbf{A}]$ とし，$|\mathbf{A}|$ を行列式とする．

これらの表記を用いて行列におけるノルムを定義する．$\mathbf{a}_i \in \mathbb{R}^d$ である a_i を列ベクトルとして要素に持つ行列 $(\mathbf{a}_1, \mathbf{a}_2, ,, \mathbf{a}_k) = \mathbf{A} \in R^{d \times k}$ に対し $\|\mathbf{A}\|_{r,p} = (\sum_{i=1}^{k} \|\mathbf{a}_i\|_r^p)^{\frac{1}{p}}$ であると $\mathbf{A}$ の (r,p) ノルムとする．このノルムは MTFL で用いられる．

1.4 関数・確率分布・その他

シグモイド関数を $\sigma(x) = (1 + \exp(-x))^{-1}$ と定義する．自然対数を $\ln(x)$ とし，$\exp(x) = e^x$ であるとする．

変数を

$$\mathbf{x} \in \mathbb{R}^d \ , \ \boldsymbol{\mu} \in \mathbb{R}^d \ , \ \boldsymbol{\Gamma} \in \mathbb{R}^{d \times d}$$
$$\mathbf{X} \in \mathbb{R}^{d \times n} \ , \ \mathbf{M} \in \mathbb{R}^{d \times n} \ , \ \mathbf{A} \in \mathbb{R}^{n \times n} \ , \ \mathbf{B} \in \mathbb{R}^{d \times d}$$

に対し，多変量正規分布と行列値正規分布をそれぞれ以下のように定義する．

実数	$\mathbb{R}$
自然数の集合	$\mathbb{N}_T$
絶対値	$\|\mathbf{x}\|$
ベクトル	$\mathbf{x} = (\mathbf{x}_{[1]}, \mathbf{x}_{[2]}, , , \mathbf{x}_{[d]})$
p ノルム	$\|\mathbf{x}\|_p$
内積	$\langle \mathbf{x}, \mathbf{y} \rangle$
単位行列	$\mathbb{I}$
直交行列	O
転置行列	$\mathbf{A}^{\mathrm{T}}$
逆行列	$\mathbf{A}^{-1}$
擬逆行列	$\mathbf{A}^{+}$
トレース	$\mathrm{tr}[\mathbf{A}]$
行列式	$\|\mathbf{A}\|$
(r,p) ノルム	$\|\mathbf{A}\|_{r,p}$
シグモイド関数	$\sigma(x)$
自然対数	$\ln(x)$
多変量正規分布	$\mathcal{N}(\mathbf{x} \mid \mu, \boldsymbol{\Gamma})$
行列値正規分布	$\mathcal{MN}(\mathbf{X} \mid \mathbf{M}, \mathbf{A}, \mathbf{B})$

表 1 数式の表記

$$\begin{aligned}
\mathcal{N}(\mathbf{x} \mid \mu, \Gamma) &= \frac{1}{C_{\mathcal{N}}} \exp[-\frac{1}{2}(x-\mu)^T \Gamma^{-1}(x-\mu)] \\
&\quad s.t. \ C_{\mathcal{N}} = (2\pi)^{\frac{d}{2}} |\Gamma|^{\frac{1}{2}} \\
\mathcal{MN}(\mathbf{X} \mid \mathbf{M}, \mathbf{A}, \mathbf{B}) &= \frac{1}{C_{\mathcal{MN}}} \exp(-\frac{1}{2}\mathrm{tr}[\mathbf{A}^{-1}(\mathbf{X}-\mathbf{M})^T \mathbf{B}^{-1}(\mathbf{X}-\mathbf{M})]) \\
&\quad s.t. \ C_{\mathcal{MN}} = (2\pi)^{\frac{dn}{2}} |\mathbf{A}|^{\frac{d}{2}} |\mathbf{B}|^{\frac{n}{2}}
\end{aligned}$$

ただし $C_{\mathcal{N}}$, $C_{\mathcal{MN}}$ は正規化定数である．文脈から判断できる場合にはこれらの正規分布を単に正規分布と呼ぶ．

2. ロジスティクス回帰と正則化項

2.1 ロジスティック分布

入力を $\mathbf{x} \in \mathbb{R}^d$ とし，予測したいラベルを $y \in \{+1, -1\}$ とした場合，ロジステック分布は

$$\begin{aligned}
p(y = +1 \mid \mathbf{x}, \mathbf{w}) &= \frac{1}{C} \exp(\langle \mathbf{x}, \mathbf{w}_+ \rangle) = \sigma(\langle \mathbf{x}, \mathbf{w}_+ \rangle) \\
p(y = -1 \mid \mathbf{x}, \mathbf{w}) &= \frac{1}{C} \exp(\langle \mathbf{x}, \mathbf{w}_- \rangle) = \sigma(-\langle \mathbf{x}, \mathbf{w}_+ \rangle) \\
s.t. \ C &= \sum_{l \in \{-1,+1\}} \exp(\langle \mathbf{x}, \mathbf{w}_l \rangle)
\end{aligned}$$

で定義される確率分布である．ただし $\mathbf{w}_{+1}, \mathbf{w}_{-1}$ は $\mathbf{x}$ と同じ次元のベクトルであり，その添字のラベル毎に別のベクトルであるとする．2 クラスロジスティック分布における (x, y) の確率は $p(y \mid \mathbf{x}, \mathbf{w}) = \sigma(y\langle \mathbf{x}, \mathbf{w} \rangle)$ と書くことができる．

個々のインスタンスを $z_i = (\mathbf{x}_i, y_i)$ とし，また与えられたデータセット全体を $Z = (z_i)_{i=1}^N$ とする．この時，データ全体での尤度は，

$$p(Z \mid \mathbf{w}) = \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \mathbf{w})$$

と表すことが出来る．これらの定義は多クラスロジスティック分布に関しても同様にすることができる．

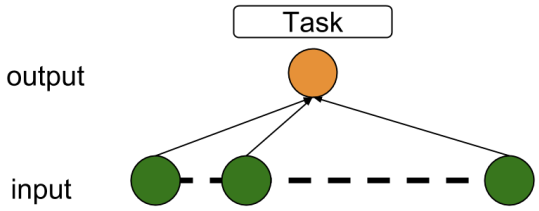

図 1 入力ベクトルを緑，出力のラベルをオレンジで表現したロジスティック回帰

2.2 ロジスティック回帰

2.2.1 誤差関数と正則化項

尤度の最大化 $\max_{\mathbf{w}} p(Z \mid \mathbf{w})$ を考えよう．ただし $\mathbf{w}$ は確率分布のパラメータである．また最大化したい関数を，目的関数と呼ぶことにする．

対数は単調増加である為，目的関数 $p(Z \mid \mathbf{w})$ の対数を取り，-1 をかけて

$$\arg\max_{\mathbf{w}} p(Z \mid \mathbf{w}) = \arg\min_{\mathbf{w}} -\ln[p(Z \mid \mathbf{w})]$$

と変形することができる．この時 $-\ln[p(Z \mid \mathbf{w})]$ を誤差関数と呼び $E(\mathbf{w})$ と定義する．前節で述べた 2 クラス分類においてロジスティクス分布を仮定すると，

$$E(\mathbf{w}) = -\ln [p(Z \mid \mathbf{w})] = -\ln \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \mathbf{w})$$
$$= \sum_{i=1}^{N} \ln [1 + \exp(-y_i \langle \mathbf{x}_i, \mathbf{w} \rangle)]$$

と具体的な誤差関数を導出することができる．このモデルをロジスティック回帰と呼ぶ．

尤度関数のみを用いるロジスティック回帰は，データに対して過学習する傾向にある．そこでパラメータ $\mathbf{w}$ に対して事前分布を考える事により，過学習を抑制することができる．

正規分布による事前分布の場合を考える．パラメータ $\mathbf{w}$ が正規分布 $\mathcal{N}(0, \frac{2}{\lambda}\mathbb{I})$ に従うとすれば，目的関数と誤差関数はそれぞれ

$$p(Z \mid \mathbf{w}, \lambda) = \prod_{i=1}^{N} \sigma(y_i \langle \mathbf{x}_i, \mathbf{w} \rangle) \frac{1}{C_{\mathcal{N}}} \exp(-\lambda \|\mathbf{w}\|^2)$$
$$E(\mathbf{w}) = \sum_{i=1}^{N} \ln(1 + \exp(-y_i \langle \mathbf{x}_i, \mathbf{w} \rangle)) + \lambda \|\mathbf{w}\|^2 + \ln[C_{\mathcal{N}}]$$

となる．ただし $C_{\mathcal{N}}$ は w に依存しない値であり，最小化には影響を与えない．この様に，誤差関数を

$$E(\mathbf{w}) = \sum_{i=1}^{N} loss(\mathbf{x}_i, y_i, \mathbf{w}) + \lambda Reg(\mathbf{w})$$

の形でかけるモデルが存在する．$loss$ を損失関数と呼び，Reg を正則化項と呼ぶ．今のモデルだと $loss(\mathbf{x}, y, \mathbf{w}) = \ln(1 + e^{-y\langle \mathbf{x}, \mathbf{w} \rangle})$ であり，$Reg(\mathbf{w}) = \|\mathbf{w}\|^2$ である．

分類問題における他の損失関数の選択肢としてはヒンジ損失 $loss = (\mathbf{x}, y, \mathbf{w}) = max(0, 1 - \langle \mathbf{x}, \mathbf{w} \rangle)$ が考えられる．これは SVM と関係した損失関数である．正則化項にも，他に例えばラプラス分布を事前分布として仮定した場合に導出される $Reg(\mathbf{w}) = \|\mathbf{w}\|_1$ などが存在する．特に $Reg(\mathbf{w}) = \|\mathbf{w}\|_2^2$ は L2 正則化，$Reg(\mathbf{w}) = \|\mathbf{w}\|_1$ は L1 正則化と呼ばれる．

本論文ではロジステック回帰を元にした手法を提案する．

3. Multi-Task Feature Learning

3.1 タスクの定義

Multi-Task Feature Learning におけるタスクの定義を述べよう．簡便化の為に各タスクにインデックスが割り当てられているとする．

> タスク
>
> t 番目のタスクに対し，データとして m 個のインスタンスの集合 $(\mathbf{x}_{t1}, y_{t1}), , (\mathbf{x}_{tm}, y_{tm}) \in (\mathbb{R}^d \times \mathbb{R})$ が与えられているとする．$\mathbf{x}_{ti}$ を入力として y_{ti} を予測する関数 f_t を学習することをタスクと呼ぶ．

個々のタスクに与えられたインスタンスの集合を $Z_t = \{z_{ti}\}_{ti=t1}^{tm}$ と表記する．

多クラス分類の場合，予測する出力はラベルかベクトルで表現される．その為，MTFL におけるタスクの定義である出力 y_{ti} が実数値であるという定義には当てはまらない．そこで本論文では多クラス分類の出力も考慮して，y_{ti} はベクトルでも良いものとする．この場合もインスタンスは $z_{ti} = (\mathbf{x}_{ti}, y_{ti})$ として表記する．

留意するべき事として，この定義はタスクに所属するインスタンスに制約を設けていない点が挙げられる．つまり $z_k \in Z_{t_1}$ かつ $z_k \in Z_{t_2}$ であるようなインスタンス $z_k = (\mathbf{x}_k, y_k)$ が存在しても問題がない．この時，MTFL のタスクでは，1 つのラベル y_k に対し複数の値 $f_{t1}(\mathbf{x}_k)$ と $f_{t2}(\mathbf{x}_k)$ が予測されることになる．次の段落では，MTFL の仮定を押さえながらモデルを導出することによって，同一のデータに対し，複数の異なる予測が生じることを防げないモデルとなっていることを確認する．

3.2 モデルの仮定と導出

MTFL は，学習したい関数 f_t が，ある関数 $h_i : \mathbb{R}^d \to \mathbb{R}$ の線型和から構成される事を仮定する．数式で書けば，

$$f_t(\mathbf{x}) = \sum_{i=1}^{d} a_{it} h_i(\mathbf{x})$$

である．h_i が $\mathbf{x}$ の次元である d 個存在する点に注意され

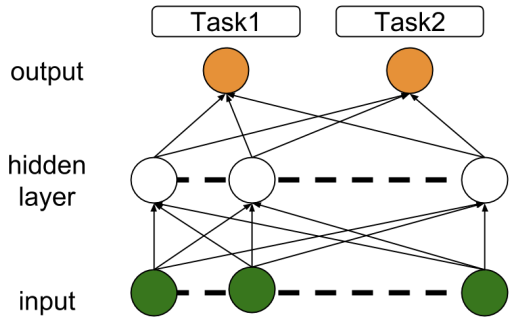

図 2 Multi Task Feature Learning の構造．活性化関数が線形でノードの数が入力ベクトルの事件と同じ二層の特殊なニューラルネットとして定義される．

たい．この構造を示したものが図 2 である．これは活性化関数に恒等関数を用いた二層のニューラルネットの構造をしており，ニューラルネットを用いた multitask learning [3] はこの構造と関係した形になっている．

次に MTFL は，$h_i(\mathbf{x}) = \langle \mathbf{x}, \mathbf{u}_i \rangle$ であることを仮定する．ここで $\mathbf{u}_i$ は直交行列の列ベクトルであり，$(\mathbf{u}_1, \mathbf{u}_2, , , \mathbf{u}_d) = \mathbf{U} \in O_d$ であるとする．この時，$\mathbf{a}_t = (a_{1t}, a_{2t}, , , a_{dt})^T$，$\mathbf{w}_t = \sum_i a_{it}\mathbf{u}_i = \mathbf{U}\mathbf{a}_t$ とすれば

$$f_t(\mathbf{x}) = \sum_{i=1}^{d} a_{it} h_i(\mathbf{x}) = \sum_{i=1}^{d} a_{it} \langle \mathbf{x}, \mathbf{u}_t \rangle = \langle \mathbf{x}, \sum_{i=1}^{d} a_{it}\, \mathbf{u}_t \rangle = \langle \mathbf{x}, \mathbf{w}_t \rangle$$

と書くことができる．

3.3 MTFL における目的関数

MTFL の目的関数は以下の様に定義されている．

$$\mathcal{E}(\mathbf{A}, \mathbf{U}) = \sum_{t=1}^{T} \sum_{i=1}^{m} loss(y_{ti}, \langle \mathbf{a}_t, \mathbf{U}^T \mathbf{x}_{ti} \rangle) + \lambda \|\mathbf{A}\|_{2,1}^2$$

ただし $\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, , , \mathbf{a}_t)$ とする．この目的関数は

$$\begin{aligned} \mathbb{R}(\mathbf{W}, \mathbf{D}) &= \sum_{t=1}^{T} \sum_{i=1}^{m} loss(y_{ti}, \langle \mathbf{x}_{ti}, \mathbf{w}_t \rangle) + \lambda \mathrm{tr}[\mathbf{W}^T \mathbf{D}^+ \mathbf{W}] \\ s.t.\ \ \mathrm{tr}[\mathbf{D}] &= 1 \quad range(\mathbf{W}) \subseteq range(\mathbf{D}) \end{aligned}$$

の最適化と等価である．この導出は MTFL [1] で示されている．

パラメータ D の最適化は W から閉じた形で導出することができる為，効率的に計算をすることができる．このように，MTFL は各タスクが 1 つの値を予測しており，複数の異なる予測結果を扱うことを想定していないモデルであることが分かる．

3.4 確率的な解釈

ロジスティック回帰にロジスティック分布が存在したように，MTFL にも確率的な解釈を与えることができる．この定式化は Zhang らが A Convex Formulation for Learning Task Relationships in Multi-Task Learning [9] で提案した Multi-Task Relationship Learning(MTRL) に基づく方法である．

MTRL ではタスクの定義を MTFL と同様に，T 個のタスクに対し，タスク $t \in \mathbb{N}_T$ にインスタンスの集合 Z_t が存在するという定義をしている．

各インスタンス $(\mathbf{x}_{ti}, y_{ti})$ の尤度関数が

$$y_{ti} \sim \mathcal{N}(y_{ti} \mid \langle \mathbf{w}_t, \mathbf{x}_{ti} \rangle, \epsilon_1)$$

であるとし，各タスクのパラメータ w_l と，それを列ベクトルに並べた行列 W に対し，パラメータ $\mathbf{W} = (\mathbf{w}_1, , , \mathbf{w}_T)$ に対し次の事前分布を仮定している．

$$\begin{aligned} \mathbf{W} &\sim \prod_{i=1}^{T} \mathcal{N}(\mathbf{w}_i \mid 0, \epsilon_2 \mathbb{I}) \mathcal{MN}(\mathbf{W} \mid 0, \mathbb{I}, \Omega) \\ &= \mathcal{MN}(\mathbf{W} \mid 0, \mathbb{I}, \epsilon_2 \mathbb{I}) \mathcal{MN}(\mathbf{W} \mid 0, \mathbb{I}, \Omega) \end{aligned}$$

最終的な誤差関数は

$$\begin{aligned} E(\mathbf{W}, \Omega) &= -\ln \prod_{t=1}^{T} \prod_{i=1}^{tm} p(y_{ti} \mid \mathbf{x}_{ti}, \mathbf{w}_t \epsilon_1) p(\mathbf{W} \mid \Omega, \epsilon_2) \\ &= \sum_{t=1}^{T} \sum_{i=1}^{tm} \frac{\epsilon_1}{2} (y_{ti} - \langle \mathbf{w}_t, \mathbf{x}_{ti} \rangle)^2 + \mathrm{tr}[\mathbf{W}(\frac{\epsilon_2}{2}\mathbb{I} + \Omega^{-1})\mathbf{W}^T] \\ &\quad + \ln(C_{\mathcal{MN}}) \end{aligned}$$

となる．この誤差関数に対し，正規化定数の代わりに $\mathrm{tr}[\Omega] = 1$ と制約をおいたものが MTRL である．2 乗損失を損失関数に書き換えれば

$$\begin{aligned} E(\mathbf{W}, \Omega) &= \sum_{t=1}^{T} \sum_{i=1}^{tm} loss(y_{ti}, \mathbf{x}_{ti}, \mathbf{w}_t) + \lambda_1 \mathrm{tr}[\mathbf{W}(\lambda_2 \mathbb{I} + \Omega^{-1})\mathbf{W}^T] + C \\ &\quad s.t.\ \ \mathrm{tr}[\Omega] = 1 \end{aligned}$$

である．ただし $\lambda_1 = \frac{1}{\epsilon_1}, \lambda_2 = \frac{\epsilon_2}{\epsilon_1}$ である．

より一般に行列 $\mathbf{W}$ に対し行列値ガウス分布を仮定すると

$$\begin{aligned} E(\mathbf{W}, \mathbf{A}, \mathbf{B}) &= \sum_{t=1}^{T} \sum_{i=1}^{tm} loss(y_{ti}, \mathbf{x}_{ti}, \mathbf{w}_t) + \lambda \mathrm{tr}[\mathbf{A}^{-1}\mathbf{W}^T\mathbf{B}^{-1}\mathbf{W}] \\ &\quad s.t.\ \ \mathrm{tr}[\mathbf{A}] = 1\ ,\ \mathrm{tr}[\mathbf{B}] = 1 \end{aligned}$$

というモデルを得ることができる．MTRL の λ_2 は λ_1 と合わせてチューニングすることで，タスク間の独立性をコントロールすることが可能である．しかし，本論文では単純な場合として，$\lambda_2 = 0$ を仮定する．

このモデルは $\mathbf{A}, \mathbf{B}$ の扱いにより，モデルとしての差異が発生する．$\mathbf{B} = \mathbb{I}$ の場合には MTFL に関係したモデルになり，$\mathbf{A} = \mathbb{I}$ の場合には MTRL に関係したモデルになる．$\mathbf{A} = \mathbb{I}$ かつ $\mathbf{B} = \mathbb{I}$ の場合は MTFL での呼び方にならって，Single Task Learning(STL) と呼ぶことにする．これは全て

のタスクを個別に学習するモデルと関係している．

$\mathbf{A} \neq \mathbb{I}$ かつ $\mathbf{B} \neq \mathbb{I}$ のモデルに関しては，Zhang らが [8] で詳しく議論を行なっている．ただその最適化は MTFL や MTRL と比べ複雑であり本論文では扱わない．

4. タスクの曖昧性

4.1 タスクの定義とタスクラベル

MTFL と MTRL でのタスクの定義は，T 個のタスクに対しインスタンスの集合 Z_t が存在し，各インスタンスの集合に対して出力を予測する事であった．これは，出力を予測する関数がハイパーパラメータと最適化する変数を除いて決まっている時，タスクはインスタンスの集合によって特徴付けられることを意味する．

MTFL と MTRL 両方の実験では，何らかの基準によってデータセット全体を幾つかのインスタンスの部分集合 Z_t に分割している．その基準に 1-of-K のラベル情報を用いているため，結果として複数の予測結果を扱う場合が発生することがない．これはインスタンスに通常の x,y とは別にラベル l が存在することを意味する．明示的に書けば，$z = (x,y,l)$ ということになる．このように，タスクの分割に用いられているラベル l を特別にタスクラベルと呼ぶことにする．このタスクラベル l に応じてデータセットを分割し，個々の部分集合を用いた学習が，MTFL によるタスクに該当する．

4.2 具体的なタスクの事例

ここでは，Amazon におけるレビューを例に，どのようなタスクが構成可能かを考察する．Amazon のレビューを元にしたデータセットは実際に MTRL で評価に用いられており，本論文での提案手法の評価にも用いている．

レビューには多くの情報が存在するが，クラス分類における情報として典型的なものに，レビューのテキストと商品に対する評価が存在する．評価は 1，2，3，4，5 の内の 1 つの値を取る．この 1 と 2 をネガティブ，4 と 5 をポジティブとし，これを y と表記する．またレビューのテキスト情報を，Bag-of-Words 等の手法でベクトル化したものを $\mathbf{x}$ とする．この $\mathbf{x}$ を入力とし出力を y とすると，通常の 2 クラス分類を考えることができる．

Amazon においてレビューが対象にする商品は，Book や Music のような幾つかのカテゴリィに分類される．このカテゴリィが $1-ok-K_1$ のラベルであり，具体的なカテゴリィを l_1 と表記すると，インスタンスの情報は $z_i = (\mathbf{x}_i, y_i, l_{1i})$ となる．レビュー全体のインスタンスを Z_{total} とし，Book のラベルを持っているインスタンスの集合を Z_{book} とする．このように，カテゴリィ情報を用いて Z_{total} を分割したデータの部分集合 Z_{l_1} は，MTFL におけるタスクのデータとして用いることができる．このようにタスクラベルが $1-of-K_1$ である場合には出力される結果は 1 つであり，MTFL を問題なく適用することができる．

4.3 MTFL で扱えない場合

しかし，訓練データのドメインと評価データのドメインが違う場合など，タスクラベルが取得できない場合も存在する．またデータのドメインが同じ場合であっても，タスクラベルを取得できない場合も考えられる．例えば，レビュアーの情報は必ずしも利用できるわけではなく，場合によっては匿名で書かれている場合も存在する．このようなタスクラベルを利用できない場合に MTFL は用いることができない．

またインスタンスが複数のタスクに所属する場合も考えられる．これは例えばインスタンスにタスクラベルが 2 つ以上ある場合が考えられる．本に著者とジャンルのタスクラベルがついている場合，そのレビューは著者をタスクラベルとした分類器と，ジャンルをタスクラベルとした分類器が形成できる．この時，1 つのインスタンスに対し，2 の結果が予測されてしまい，MTFL の枠組みでは扱うことができない．

5. 提案手法:重み付き多タスク学習

前章で見てきたように，MTFL によるタスクの構成では，扱えない場合が存在した．本章は，条件付きロジスティック分布を定義することによって weighted Multi-Task Learning(wMTL) を提案する．この提案手法は，複数の出力を適切に重み付けをして，１つの予測結果にすることができる方法である．応用として，テスト時にタスクラベルが存在しない場合においても，既存の手法の精度を落とさず扱うことができる．

5.1 条件付きロジスティック分布

この節では，条件付きロジスティック分布を定義する．この定義は本論文によるもので，一般的な定義でないことに注意されたい．

MTFL に倣い，個々のインスタンスが多ラベルを持っている事を仮定する．例えばインスタンスが入力 $\mathbf{x} \in \mathbb{R}^d$ と 2 種類のラベル (y,l) を持っているとする．ただし $y \in Y$, $l \in L$ とする．

条件付きロジスティック分布を以下のように定義する

$$p(y \mid \mathbf{x}, l, \mathbf{w}) = \frac{1}{C_l} \exp[\langle \mathbf{x}, \mathbf{w}_{(y,l)} \rangle]$$
$$s.t. \quad C_l = \sum_{y' \in Y} \exp[\langle \mathbf{x}, \mathbf{w}_{(y',l)} \rangle]$$

この条件付きロジスティック分布を用いれば，通常のロジスティック分布は以下の様に展開することができる．

$$
\begin{aligned}
p(y \mid \mathbf{x}, \mathbf{w}) &= \sum_{l \in L} p(y, l \mid \mathbf{x}, \mathbf{w}) \\
&= \sum_{l \in L} p(l \mid \mathbf{x}, \mathbf{w}) p(y \mid \mathbf{x}, l, \mathbf{w})
\end{aligned}
$$

$p(y \mid \mathbf{x}, \mathbf{w})$ はデータがクラス l に割り当てられる確率であり，この式はその確率を重みとして $p(y \mid \mathbf{x}, l, \mathbf{w})$ を足し合わせた形になっている．

特に $p(l \mid \mathbf{x}, \mathbf{w})$ と $p(y \mid \mathbf{x}, l, \mathbf{w})$ がそれぞれロジスティック分布と条件付きロジスティック分布の場合には，

$$
\begin{aligned}
p(y \mid \mathbf{x}, \mathbf{w}) &= \sum_{l \in L} p(l \mid \mathbf{x}, \mathbf{w}) p(y \mid \mathbf{x}, l, \mathbf{w}) \\
&= \sum_{l \in L} \frac{\exp[\langle \mathbf{x}, \mathbf{w}_l \rangle]}{\sum_{a \in L} \exp[\langle \mathbf{x}, \mathbf{w}_a \rangle]} \frac{\exp[\langle \mathbf{x}, \mathbf{w}_{(y,l)} \rangle]}{\sum_{b \in Y} \exp[\langle \mathbf{x}, \mathbf{w}_{(b,l)} \rangle]} \\
&= \sum_{l \in L} \frac{1}{C} \exp[\langle \mathbf{x}, \mathbf{w}_l \rangle + \langle \mathbf{x}, \mathbf{w}_{(y,l)} \rangle] \\
s.t. \quad C &= \sum_{a \in L} \sum_{b \in Y} \exp[\langle \mathbf{x}, \mathbf{w}_a \rangle + \langle \mathbf{x}, \mathbf{w}_{(b,a)} \rangle]
\end{aligned}
$$

と書くことができる．以下ではこのロジスティック分布を用いる．

5.2 重み付き多タスク学習

前述の条件付きロジスティック回帰を用いて wMTL を定義する．データが $Z = (z_i)_{i=1}^N = (\mathbf{x}_i, y_i, l_i)_{i=1}^N$ であるとする．この時，wMTL の目的関数を

$$
\begin{aligned}
E(\mathbf{W}, \mathbf{A}, \mathbf{B}) &= -\ln[\prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \mathbf{W}) p(\mathbf{W} \mid \mathbf{A}, \mathbf{B})] \\
&= \sum_{i=1}^{N} -\ln[\sum_{l \in L} \frac{1}{C} \exp[\langle \mathbf{x}_i, \mathbf{w}_l \rangle + \langle \mathbf{x}_i, \mathbf{w}_{(y_i,l)} \rangle]] \\
&\quad + \lambda \mathrm{tr}[\mathbf{A}^{-1} \mathbf{W} \mathbf{B}^{-1} \mathbf{W}^T] \\
s.t. \quad & \mathrm{tr}[\mathbf{A}] = 1 \ , \ \mathrm{tr}[\mathbf{B}] = 1
\end{aligned}
$$

とする．ただし C は条件付きロジスティック分布で定義した Z であるとする．$\mathbf{A} = \mathbb{I}$ の場合が MTFL に，$\mathbf{B} = \mathbb{I}$ の場合が MTRL に，$\mathbf{A} = \mathbb{I}, \mathbf{B} = \mathbb{I}$ の場合が STL に関係している．提案手法の実験では，これら MTFL,MTRL,STL に wMTL を用いたモデルを使用している．

この目的関数の最適化は，2 段階に分けられる．まずそれぞれのラベルの予測をタスクとして，通常の MTFL や MTRL に適用しその結果を初期値とする．次に wMTL の目的関数で最適化を行うが．しかし，多くの場合通常の学習で獲得された初期値は十分に良い値であり，wMTL での最適化を省略しても余り問題は無い．

6. 評価実験

本実験では，タスクラベルが無い場合に wMTL がどのような影響を受けるかを，実データを用いて評価する．また比較の為に，タスクラベルを用いれる状態での MTFL や MTRL との比較を行う．

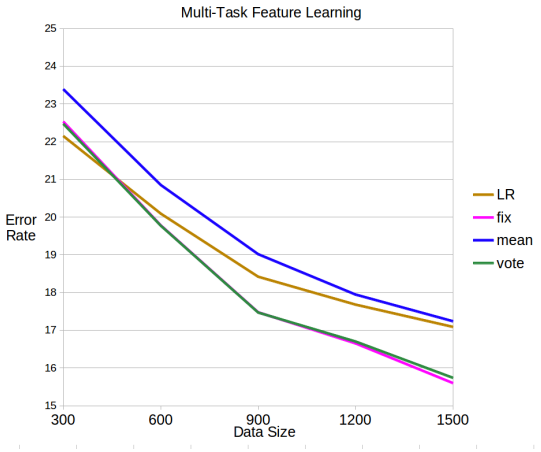

図 3 Multi-Task Feature Learning

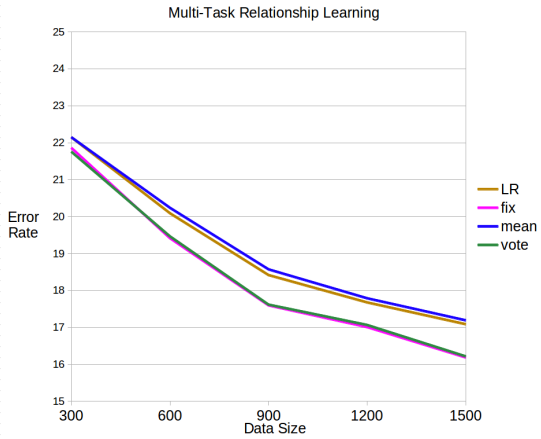

図 4 Multi-Task Relationship Learning

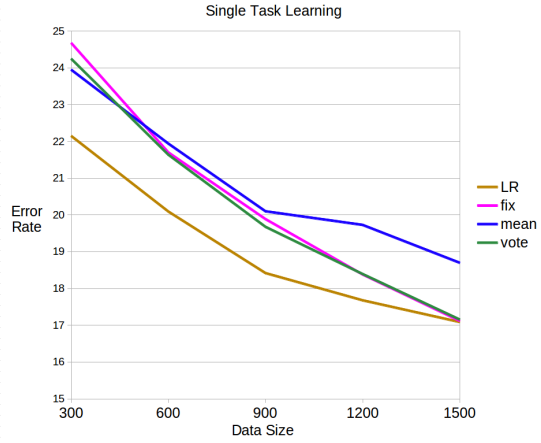

図 5 Single Task Learning

6.1 使用したデータセット

本論文の評価実験には，Blitzer らによる Multi-Domain Sentiment Dataset [2] を使用した．このデータセットは，Amazon のレビューを素材としたものであり，個々のインスタンスはレビューのテキスト情報をベクトル化した $\mathbf{x}$ と，レビューの評価がポジティブなものかネガティブなものかの情報 $y \in \{+,-\}$ から構成される．

またデータセットは，そのレビューされた商品のカテゴリィに基づき，Book,DVDs,Electronics,kitchen の 4 つのクラスに分割されている．このカテゴリィの情報を l とした時，インスタンスは $z = (\mathbf{x}, y, l)$ と表記することができる．このことにより，多ラベルの情報が存在することが分かる．それぞれのカテゴリィには 3000,2000,2000,2000 個のインスタンスが存在し，テキスト情報 $\mathbf{x}$ の素性次元は 473856 次元のベクトルで表現される．

6.2 実験設定

タスクラベルが存在しない状態での予測として，全ての分類器の予測結果を平均したものと，本論文の提案手法である wMTL による予測を行った．また比較するために，タスクラベルが与えらた通常の状態での性能も評価した．使用した多タスク学習の手法は，MTFL,MTRL,STL である．

具体的な実験設定としては，学習用に各カテゴリィから $300, 600, 900, 1200, 1500$ 個のデータをランダムにサンプルし，同様に評価用に各カテゴリィから 500 個のデータをサンプルして学習と評価を行った．またハイパーパラメータである λ は $\lambda = 10^{\alpha}$ とした時，$\alpha = -2$ から $\alpha = -\frac{17}{4}$ まで $\frac{1}{4}$ 刻みで，合計 9 個の λ を用いた．この試行を 1 回とした場合，合計で 20 回試行を行った．

6.3 実験結果

前述の実験を行い，結果を図としてプロットした．図 3 は MTFL を用いた結果を，図 4 は MTRL を用いた結果を，図 5 は STL を用いた結果を記載した．これらの図全ての縦軸は誤答率であり，値をパーセンテージに基づいて表示している．横軸はデータサイズであり，各グラフの折れ線は手法に応じて色付けされている．MTFL:fix，MTFL:mean，MTFL:vote は，fix が与えられたタスクラベルを用いた場合を，mean が全ての学習器の予測を平均した場合を，vote が提案手法である wMTL を用いた場合を表している．またハイパーパラメータは前述の 9 個の内，最も良いものを用いた場合を記載している．

グラフには比較対象として，タスクラベルを用いない通常の L2 ロジスティック回帰による学習結果を LR として記載している．この記載方法は他の MTRL や STL においても，同様である．

MTFL と MTRL は，適切にパラメータが与えられた場合には，LR より精度が良い事がわかる．データ数が少ない場合には，MTFL と MTRL の予測精度は，LR によるものと大差無いが，データ量が増えるとその差は開いていき，MTFL においては 1%以上の差が出る結果となっている．また独立してタスクを学習する STL では，LR に大きく劣る結果になっている．そして図が示すように，wMTL を用いた場合はタスクラベルが存在しない場合でも，既存手法の精度とほぼ同程度の精度を実現できていることが分かる．

7. おわりに

7.1 本論文の成果と考察

本論文ではまず，MTFL におけるタスクの定義を見直すことで，その定式化では扱えない場合が存在することを指摘した．また応用上，そのような問題設定が自然に考えられることを，事例を交えながら説明をした．そして条件付きロジスティック回帰を導入することで，確率的な妥当性を与えることができるモデルである，wMTL を提案した．最後に，提案手法を応用することでテスト時にタスクラベルが存在しない場合でも，通常と同等の精度が期待できることを実験により確認した．

今回提案した手法は MTFL や MTRL に限らず，他の多タスク学習の幾つかにも適用が可能である．このことは多タスク学習の応用先をより広いものにすることができる．例えば，Amazon のレビューデータで学習した結果を楽天のビューに適用するような事が可能である．

wMTL はロジスティック回帰を条件付きで展開することで実現される．この方法は，カテゴリィのラベルに対して更に適用することができる．例えば Book の下に SF，ミステリ，サスペンス，ファンタジーという細かいカテゴリィが存在した時，この情報をタスクラベルとして条件付き展開することが可能である．このことは階層的なモデルを作れることを意味している．しかし，どのようなクラスをタスクラベルとして用いたら効果的なのか，どのような順番で用いれば有用なのかということは，研究の余地がある課題である．

参考文献

[1] Argyriou, A., Evgeniou, T. and Pontil, M.: Multi-task feature learning, *Advances in Neural Information Processing Systems 19* (2007).
[2] Blitzer, J., Dredze, M. and Pereira, F.: Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, *Association for Computational Linguistics*, Vol. 7, pp. 440–447 (2007).
[3] Caruana, R.: Multitask learning, *Machine learning*, Vol. 28, No. 1, pp. 41–75 (1997).
[4] Collobert, R. and Weston, J.: A unified architecture for natural language processing: deep neural networks with multitask learning, *Proceedings of the 25th International Conference on Machine Learning (ICML-08)*, pp. 160–167 (2008).
[5] Gong, P., Ye, J. and Zhang, C.: Robust Multi-Task Fea-

ture Learning., *KDD: proceedings/International Conference on Knowledge Discovery & Data Mining. International Conference on Knowledge Discovery & Data Mining*, Vol. 2012, pp. 895–903 (2012).

[6] Kumar, A. and Daume, H.: Learning Task Grouping and Overlap in Multi-task Learning, *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pp. 1383–1390 (2012).

[7] Yu, K., Tresp, V. and Schwaighofer, A.: Learning Gaussian processes from multiple tasks, *Proceedings of the 22nd International Conference on Machine Learning (ICML-05)*, pp. 1012–1019 (2005).

[8] Zhang, Y. and Schneider, J. G.: Learning Multiple Tasks with a Sparse Matrix-Normal Penalty, *Advances in Neural Information Processing Systems*, pp. 2550–2558 (2010).

[9] Zhang, Y. and Yeung, D.-Y.: A Convex Formulation for Learning Task Relationships in Multi-Task Learning, *Proceedings of the 26th Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-10)*, Corvallis, Oregon, AUAI Press, pp. 733–742 (2010).