# Sparse Bilinear Logistic Regression

Jianing V. Shi[1,2]*, Yangyang Xu[3], and Richard G. Baraniuk[1]

[1] Department of Electrical and Computer Engineering, Rice University

[2] Department of Mathematics, UCLA

[3] Department of Computational and Applied Mathematics, Rice University

February 08, 2014

**Abstract**

In this paper, we introduce the concept of sparse bilinear logistic regression for decision problems involving explanatory variables that are two-dimensional matrices. Such problems are common in computer vision, brain-computer interfaces, style/content factorization, and parallel factor analysis. The underlying optimization problem is bi-convex; we study its solution and develop an efficient algorithm based on block coordinate descent. We provide a theoretical guarantee for global convergence and estimate the asymptotical convergence rate using the Kurdyka-Łojasiewicz inequality. A range of experiments with simulated and real data demonstrate that sparse bilinear logistic regression outperforms current techniques in several important applications.

## 1 Introduction

Logistic regression [1] has a long history in decision problems that arise in computer vision [2], bioinformatics [3], gene classification [4], and neural signal processing [5]. Recently sparsity has been introduced into logistic regression to combat the curse of dimensionality, by stipulating that only a subset of explanatory variables are informative about classification [6]. The indices of the non-zero weights correspond to features that are informative about classification, therefore leading to feature selection. Sparse logistic regression has many attractive properties such as robustness to noise and logarithmic sample complexity bounds [7].

In the classical form of logistic regression the explanatory variables are treated as i.i.d. vectors. However, in many real-world applications, the explanatory variables take the form of matrices. In image recognition tasks [8], each feature is an image. Visual recognition tasks

*Corresponding author's email address: jianing@math.ucla.edu

for video data often use a feature-based representation, such as the scale-invariant feature transform (SIFT) [9] or histogram of oriented gradient (HOG) [10], to construct features for each frame, resulting in histogram-time feature matrices. Brain-computer interfaces based on electroencephalography (EEG) make decisions about motor action [11] using channel-time matrices.

For these and other applications, bilinear logistic regression [12] extends logistic regression to explanatory variables that take two-dimensional matrix form. The resulting dimensionality reduction of the feature space in turn yields better generalization performance. In contrast to standard logistic regression, which collapses each feature matrix into a vector and learns a single weight vector, bilinear logistic regression learns weight factors along each dimension of the matrix to form the decision boundary. It has been shown that the unregularized bilinear logistic regression outperforms linear logistic regression in several applications, including brain-computer interface [12]. It has also been shown that in certain visual recognition tasks, a support vector machine (SVM) applied in the bilinear feature space outperforms an SVM applied in the standard linear feature space as well as an SVM applied to a dimensionality-reduced feature space using PCA [13].

Moreover, bilinear logistic regression has found application in style and content separation [14], which can improve the performance of object recognition tasks under various nuisance variables such as orientation, scale, and viewpoint. Bilinear logistic regression identifies subspace projections that factor out informative features and nuisance variables, thus leading to better generalization performance.

Finally, bilinear logistic regression reveals the contributions of different dimensions to classification performance, similarly to parallel factor analysis [15]. This leads to better interpretability of the resulting decision boundary.

In this paper, we introduce sparsity to the bilinear logistic regression model and demonstrate that it improves generalization performance in a range of classification problems. Our contributions are three-fold. First, we propose a sparse bilinear regression model that fuses the key ideas behind both sparse logistic regression and bilinear logistic regression. Second, we study the properties of the solution of the bilinear logistic regression problem. Third, we develop an efficient algorithm based on block coordinate descent for solving the sparse bilinear regression problem. Both the theoretical analysis and the numerical optimization are complicated by the bi-convex nature of the problem, since the solution may become stuck at a non-stationary point. In contrast to the conventional block coordinate descent method, we solve each subproblem using the proximal method, which significantly accelerates convergence. We also provide a theoretical guarantee for global convergence, and estimate the asymptotical convergence rate using a result based on the Kurdyka-Łojasiewicz inequality.

## 2 Sparse bilinear logistic regression

### 2.1 Problem Definition

We consider the following problem in this paper: Given $n$ sample-label pairs $\{(\mathbf{X}_i, y_i)\}_{i=1}^n$, where $\mathbf{X}_i \in \mathbb{R}^{s\times t}$ is an explanatory variable in the form of a matrix and $y_i \in \{-1, +1\}$ is a categorical dependent variable, we seek a decision boundary to separate these samples.

### 2.2 Prior Art

*Logistic Regression*

The basic form of logistic regression transforms each explanatory variable from a matrix to a vector, $\bar{\mathbf{x}}_i = vec(\mathbf{X}_i) \in \mathbb{R}^p$, where $p = st$. One seeks a hyperplane, defined as $\{\mathbf{x} : \mathbf{w}^\top \mathbf{x} + b = 0\}$, to separate these samples. For a new data sample $\bar{\mathbf{x}}_i$, its category can be predicted using a binomial model based on the margin $\mathbf{w}^\top \bar{\mathbf{x}}_i + b$. Figure 1 illustrates such an idea.

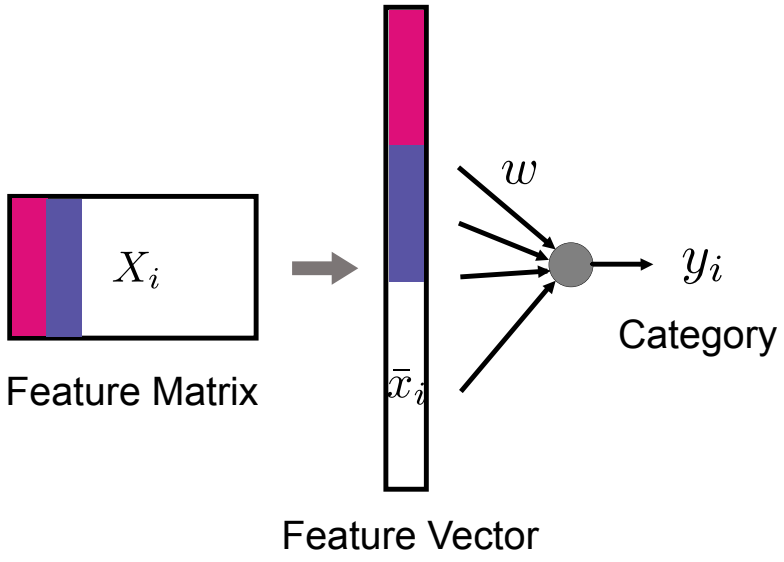


Figure 1: Illustration for logistic regression.

Essentially the logistic regression constructs a mapping from the feature vector $\bar{\mathbf{x}}_i$ to the label $y_i$,

$$\Psi^{LR} : \mathbf{w}^\top \bar{\mathbf{x}}_i + b \mapsto y_i.$$

Assuming the samples of both classes are i.i.d., the conditional probability for classifier label $y_i$ based on sample $\bar{\mathbf{x}}_i$, according to the logistic model, takes the form of

$$p(y_i|\bar{\mathbf{x}}_i, \mathbf{w}, b) = \frac{\exp[y_i(\mathbf{w}^\top \bar{\mathbf{x}}_i + b)]}{1 + \exp[y_i(\mathbf{w}^\top \bar{\mathbf{x}}_i + b)]}, \ i = 1, \cdots, n.$$

To perform the maximum likelihood estimation (MLE) of $\mathbf{w}$ and $b$, one can minimize the empirical loss function

$$\ell(\mathbf{w}, b) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + \exp[-y_i(\mathbf{w}^\top \bar{\mathbf{x}}_i + b)]\right). \tag{1}$$

*Sparse Logistic Regression*

We assume some sparsity promoting prior on $\mathbf{w}$, typically the Laplacian prior. The maximum a posteriori (MAP) estimate for sparse logistic regression can be derived,

$$\min_{\mathbf{w},b} \ell(\mathbf{w}, b) + \lambda\|\mathbf{w}\|_1, \tag{2}$$

where $\lambda$ is a regularization parameter.

*Bilinear Logistic Regression*

A key insight of bilinear logistic regression is to preserve the matrix structure of the explanatory variables. The decision boundary is constructed using a weight matrix $\mathbf{W}$, which is further factorized into $\mathbf{W} = \mathbf{U}\mathbf{V}^\top$ with two factors $\mathbf{U} \in \mathbb{R}^{s\times r}$ and $\mathbf{V} \in \mathbb{R}^{t\times r}$. Figure 2 illustrates the concept of bilinear logistic regression.

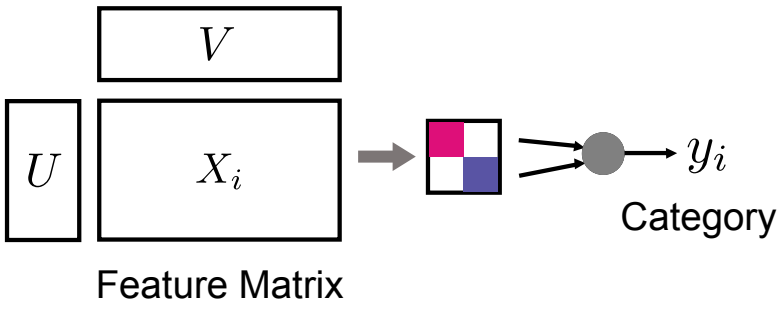


Figure 2: Illustration for bilinear logistic regression.

Bilinear logistic regression constructs a new mapping from the feature matrix $\mathbf{X}_i$ to the label $y_i$,

$$\Psi^{BLR} : \mathrm{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b \mapsto y_i,$$

where $\mathrm{tr}(\mathbf{A}) = \sum_i a_{ii}$ for a square matrix $\mathbf{A}$. Under these settings, the empirical loss function in (1) becomes

$$\ell(\mathbf{U}, \mathbf{V}, b) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + \exp[-y_i(\mathrm{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)]\right). \tag{3}$$

### 2.3 Our New Model

*Sparse Bilinear Logistic Regression*

We assume some sparsity promoting priors on $\mathbf{U}$ and $\mathbf{V}$, and derive the MAP estimate for sparse bilinear logistic regression. The variational problem is

$$\min_{\mathbf{U},\mathbf{V},b} \ell(\mathbf{U},\mathbf{V},b) + r_1(\mathbf{U}) + r_2(\mathbf{V}), \tag{4}$$

where $r_1$ and $r_2$ are assumed to be convex functions incorporating the priors to promote structures on $\mathbf{U}$ and $\mathbf{V}$, respectively. Due to space limitation, in this paper we focus on the elastic net regularization term

$$r_1(\mathbf{U}) = \mu_1\|\mathbf{U}\|_1 + \frac{\mu_2}{2}\|\mathbf{U}\|_F^2, \tag{5a}$$

$$r_2(\mathbf{V}) = \nu_1\|\mathbf{V}\|_1 + \frac{\nu_2}{2}\|\mathbf{V}\|_F^2, \tag{5b}$$

where $\|\mathbf{U}\|_1 \triangleq \sum_{i,j}|u_{ij}|$. Depending on the applications, some other regularizers can be used. For example, one can use the total variation regularization, which we will explore in future work.

## 3 Numerical Algorithm to Solve (4)

### 3.1 Block Coordinate Descent

We propose efficient numerical algorithm to solve for the variational problem (4). It is based on the *block coordinate descent* method, which iteratively updates $(\mathbf{U}, b)$ with $\mathbf{V}$ fixed and then $(\mathbf{V}, b)$ with $\mathbf{U}$ fixed. The original flavor of block coordinate descent alternates between the following two subproblems:

$$(\mathbf{U}^k, \hat{b}^k) = \operatorname*{argmin}_{(\mathbf{U},b)} \ell(\mathbf{U},\mathbf{V}^{k-1},b) + r_1(\mathbf{U}), \tag{6a}$$

$$(\mathbf{V}^k, b^k) = \operatorname*{argmin}_{(\mathbf{V},b)} \ell(\mathbf{U}^k,\mathbf{V},b) + r_2(\mathbf{V}). \tag{6b}$$

The pseudocode for block coordinate descent is summarized in Algorithm 1.

Note that even though various optimization methods exist to solve each block, due to the nonlinear form of the empirical loss function $\ell(\cdot)$, solving each block accurately can be computationally expensive.

### 3.2 Block Coordinate Proximal Descent

In order to accelerate computation, we have chosen to solve each block using the proximal method. We call it *block coordinate proximal descent* method. Specifically, at iteration $k$,

**Algorithm 1** Block Coordinate Descent

**Input:** $\{\mathbf{X}_i, y_i\}_{i=1}^n$

**Initialization:** Choose $(\mathbf{U}^0, \mathbf{V}^0, b^0)$

**while** convergence criterion not met **do**

Compute $(\mathbf{U}^k, \hat{b}^k)$ by solving (6a)

Compute $(\mathbf{V}^k, b^k)$ by solving (6b)

Let $k = k + 1$

**end while**

we perform the following updates:

$$\begin{aligned}
\mathbf{U}^k =& \operatorname*{argmin}_{\mathbf{U}} \langle \nabla_{\mathbf{U}} \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \mathbf{U} - \mathbf{U}^{k-1} \rangle \\
&+ \frac{L_u^k}{2} \|\mathbf{U} - \mathbf{U}^{k-1}\|_F^2 + r_1(\mathbf{U}), \qquad (7a) \\
\hat{b}^k =& \operatorname*{argmin}_{b} \langle \nabla_b \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), b - b^{k-1} \rangle \\
&+ \frac{L_u^k}{2} (b - b^{k-1})^2, \qquad (7b) \\
\mathbf{V}^k =& \operatorname*{argmin}_{\mathbf{V}} \langle \nabla_{\mathbf{V}} \ell(\mathbf{U}^{k}, \mathbf{V}^{k-1}, \hat{b}^{k}), \mathbf{V} - \mathbf{V}^{k-1} \rangle \\
&+ \frac{L_v^k}{2} \|\mathbf{V} - \mathbf{V}^{k-1}\|_F^2 + r_2(\mathbf{V}), \qquad (7c) \\
b^k =& \operatorname*{argmin}_{b} \langle \nabla_b \ell(\mathbf{U}^{k}, \mathbf{V}^{k-1}, \hat{b}^{k}), b - \hat{b}^{k} \rangle \\
&+ \frac{L_v^k}{2} (b - \hat{b}^k)^2, \qquad (7d)
\end{aligned}$$

where $L_u^k$ and $L_v^k$ are stepsize parameters to be specified in Section 3.4. Note that we have decoupled $(\mathbf{U}, b)$-subproblem to (7a) and (7b) since the updates of $\mathbf{U}$ and $b$ are independent. Similarly, $(\mathbf{V}, b)$-subproblem has been decoupled to (7c) and (7d).

Denote the objective function of (4) as

$$F(\mathbf{U}, \mathbf{V}, b) \triangleq \ell(\mathbf{U}, \mathbf{V}, b) + r_1(\mathbf{U}) + r_2(\mathbf{V}).$$

Let $F^k \triangleq F(\mathbf{U}^k, \mathbf{V}^k, b^k)$ and $\mathbf{W}^k \triangleq (\mathbf{U}^k, \mathbf{V}^k, b^k)$. We define *convergence criterion* as $q^k \leq \epsilon$, where

$$q^k \triangleq \max \Big\{ \frac{\|\mathbf{W}^k - \mathbf{W}^{k-1}\|_F}{1 + \|\mathbf{W}^{k-1}\|_F}, \frac{|F^k - F^{k-1}|}{1 + F^{k-1}} \Big\}, \qquad (8)$$

and $\|\mathbf{W}\|_F^2 \triangleq \|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + |b|^2$.

The pseudocode for block coordinate proximal descent is summarized in Algorithm 2.

**Algorithm 2** Block Coordinate Proximal Descent

**Input:** $\{\mathbf{X}_i, y_i\}_{i=1}^n$

**Initialization:** Choose $(\mathbf{U}^0, \mathbf{V}^0, b^0)$

**while** convergence criterion not met **do**

Compute $(\mathbf{U}^k, \hat{b}^k)$ by (7a) and (7b)

Compute $(\mathbf{V}^k, b^k)$ by (7c) and (7d)

Let $k = k + 1$

**end while**

### 3.3 Solving the Subproblems

The $b$-subproblems (7b) and (7d) are simply gradient descent,

$$\hat{b}^k = b^{k-1} - \frac{1}{L_u^k} \nabla_b \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \tag{9a}$$

$$b^k = \hat{b}^k - \frac{1}{L_v^k} \nabla_b \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k). \tag{9b}$$

The $\mathbf{U}$-subproblem (7a) and $\mathbf{V}$-subproblem (7c) are both strongly convex and can be solved by various convex programming solvers. However, the algorithm may need to run a few iterations to converge, therefore it is important to solve them very efficiently.

The beauty of using the proximal method is its admission for closed-form solutions. More specifically, for elastic net regularization terms $r_1$ and $r_2$ defined as (5), both (7a) and (7c) admits closed form solutions,

$$\mathbf{U}^k = \mathcal{S}_{\tau_u} \left( \frac{L_u^k \mathbf{U}^{k-1} - \nabla_{\mathbf{U}} \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1})}{L_u^k + \mu_2} \right), \tag{10a}$$

$$\mathbf{V}^k = \mathcal{S}_{\tau_v} \left( \frac{L_v^k \mathbf{V}^{k-1} - \nabla_{\mathbf{V}} \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)}{L_v^k + \nu_2} \right), \tag{10b}$$

where $\tau_u = \frac{\mu_1}{L_u^k + \mu_2}$, $\tau_v = \frac{\nu_1}{L_v^k + \nu_2}$, and $\mathcal{S}_\tau(\cdot)$ is the component-wise shrinkage defined by

$$\big(\mathcal{S}_\tau(\mathbf{Z})\big)_{ij} = \begin{cases} z_{ij} - \tau, & \text{if } z_{ij} > \tau; \\ z_{ij} + \tau, & \text{if } z_{ij} < -\tau; \\ 0, & \text{if } |z_{ij}| \le \tau. \end{cases}$$

The proximal method leads to closed-form solution for each subproblem, and the entire algorithm only involves matrix-vector multiplication and component-wise shrinkage operator. Therefore our numerical algorithm is promised to be computationally efficient. We will corroborate such a statement using numerical experiments.

### 3.4 Selection of $L_u^k$ and $L_v^k$

To ensure the sequence generated by Algorithm 2 attains sufficient decrease in the objective function, $L_u^k$ is typically chosen as a Lipschitz constant of $\nabla_{(\mathbf{U},b)}\ell(\mathbf{U},\mathbf{V}^{k-1},b)$ with respect to $(\mathbf{U},b)$. More precisely, for all $(\mathbf{U},b)$ and $(\tilde{\mathbf{U}},\tilde{b})$, it holds that

$$\begin{aligned}&\|\nabla_{(\mathbf{U},b)}\ell(\mathbf{U},\mathbf{V}^{k-1},b)-\nabla_{(\mathbf{U},b)}\ell(\tilde{\mathbf{U}},\mathbf{V}^{k-1},\tilde{b})\|_F\\ \leq& L_u^k\|(\mathbf{U},b)-(\tilde{\mathbf{U}},\tilde{b})\|_F,,\end{aligned}$$

where $\|(\mathbf{U},b)\|_F := \sqrt{\|\mathbf{U}\|_F^2+b^2}$. Similarly, $L_v^k$ can be chosen as a Lipschitz constant of $\nabla_{(\mathbf{V},b)}\ell(\mathbf{U}^k,\mathbf{V},b)$ with respect to $(\mathbf{V},b)$. The next lemma shows that the two partial gradients $\nabla_{(\mathbf{U},b)}\ell(\mathbf{U},\mathbf{V},b)$ and $\nabla_{(\mathbf{V},b)}\ell(\mathbf{U},\mathbf{V},b)$ are Lipschitz continuous with constants dependent on $\mathbf{V}$ and $\mathbf{U}$ respectively.

**Lemma 3.1** *The partial gradients $\nabla_{(\mathbf{U},b)}\ell(\mathbf{U},\mathbf{V},b)$ and $\nabla_{(\mathbf{V},b)}\ell(\mathbf{U},\mathbf{V},b)$ are Lipschitz continuous with constants*

$$L_u = \frac{\sqrt{2}}{n}\sum_{i=1}^n\big(\|\mathbf{X}_i\mathbf{V}\|_F+1\big)^2, \tag{11a}$$

$$L_v = \frac{\sqrt{2}}{n}\sum_{i=1}^n\big(\|\mathbf{X}_i^\top\mathbf{U}\|_F+1\big)^2, \tag{11b}$$

**Proof.** By straightforward calculation, we have

$$\nabla_{\mathbf{U}}\ell(\mathbf{U},\mathbf{V},b) = -\frac{1}{n}\sum_{i=1}^n\Big(1+\exp\big[y_i\big(\mathrm{tr}(\mathbf{U}^\top\mathbf{X}_i\mathbf{V})+b\big)\big]\Big)^{-1}y_i\mathbf{X}_i\mathbf{V}, \tag{12a}$$

$$\nabla_{\mathbf{V}}\ell(\mathbf{U},\mathbf{V},b) = -\frac{1}{n}\sum_{i=1}^n\Big(1+\exp\big[y_i\big(\mathrm{tr}(\mathbf{U}^\top\mathbf{X}_i\mathbf{V})+b\big)\big]\Big)^{-1}y_i\mathbf{X}_i^\top\mathbf{U}, \tag{12b}$$

$$\nabla_b\ell(\mathbf{U},\mathbf{V},b) = -\frac{1}{n}\sum_{i=1}^n\Big(1+\exp\big[y_i\big(\mathrm{tr}(\mathbf{U}^\top\mathbf{X}_i\mathbf{V})+b\big)\big]\Big)^{-1}y_i. \tag{12c}$$

For any $(\mathbf{U},b)$ and $(\tilde{\mathbf{U}},\tilde{b})$, we have

$$\begin{aligned}&\|\nabla_{(\mathbf{U},b)}\ell(\mathbf{U},\mathbf{V},b)-\nabla_{(\mathbf{U},b)}\ell(\tilde{\mathbf{U}},\mathbf{V},\tilde{b})\|_F\\ \leq&\frac{1}{n}\sum_{i=1}^n\left|\Big(1+\exp\big[y_i\big(\mathrm{tr}(\mathbf{U}^\top\mathbf{X}_i\mathbf{V})+b\big)\big]\Big)^{-1}-\Big(1+\exp\big[y_i\big(\mathrm{tr}(\tilde{\mathbf{U}}^\top\mathbf{X}_i\mathbf{V})+\tilde{b}\big)\big]\Big)^{-1}\right|\big(\|\mathbf{X}_i\mathbf{V}\|_F+1\big)\\ \leq&\frac{1}{n}\sum_{i=1}^n\Big(\|\mathbf{U}-\tilde{\mathbf{U}}\|_F\|\mathbf{X}_i\mathbf{V}\|_F+|b-\tilde{b}|\Big)\big(\|\mathbf{X}_i\mathbf{V}\|_F+1\big)\\ \leq&\frac{1}{n}\sum_{i=1}^n\big(\|\mathbf{X}_i\mathbf{V}\|_F+1\big)^2\Big(\|\mathbf{U}-\tilde{\mathbf{U}}\|_F+|b-\tilde{b}|\Big)\\ \leq&\frac{\sqrt{2}}{n}\sum_{i=1}^n\big(\|\mathbf{X}_i\mathbf{V}\|_F+1\big)^2\|(\mathbf{U},b)-(\tilde{\mathbf{U}},\tilde{b})\|_F,\end{aligned}$$

where in the third inequality we have used the inequality

$$|(1+e^s)^{-1} - (1+e^q)^{-1}| \le |s-q|,$$

and the last inequality follows from

$$\|\mathbf{U}-\tilde{\mathbf{U}}\|_F + |b-\tilde{b}| \le \sqrt{2}\|(\mathbf{U},b)-(\tilde{\mathbf{U}},\tilde{b})\|_F$$

by Cauchy-Schwarz inequality. This completes the proof of (11a), and (11b) can be shown in the same way. ■

However, $L_u^k$ and $L_v^k$ chosen in such a manner may be too large and slow down the convergence. Therefore we have chosen to use an alternative and efficient way to dynamically update them. Specifically, we let

$$L_u^k = \max(L_{\min}, L_u^{k-1}\eta^{n_u^k}) \tag{13}$$

where $L_{\min} > 0$, $\eta > 1$, and $n_u^k \ge -1$ is the smallest integer such that

$$\begin{aligned}
&\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k) \\
\le\ &\ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}) \\
&+ \langle \nabla_{\mathbf{U}}\ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \mathbf{U}^k - \mathbf{U}^{k-1}\rangle \\
&+ \langle \nabla_b \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \hat{b}^k - b^{k-1}\rangle \\
&+ \frac{L_u^k}{2}\|\mathbf{U}^k - \mathbf{U}^{k-1}\|_F^2 + \frac{L_u^k}{2}(\hat{b}^k - b^{k-1})^2,
\end{aligned} \tag{14}$$

and let

$$L_v^k = \max(L_{\min}, L_v^{k-1}\eta^{n_v^k}), \tag{15}$$

where $n_v^k \ge -1$ is the smallest integer such that

$$\begin{aligned}
&\ell(\mathbf{U}^k, \mathbf{V}^k, b^k) \\
\le\ &\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k) \\
&+ \langle \nabla_{\mathbf{V}}\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k), \mathbf{V}^k - \mathbf{V}^{k-1}\rangle \\
&+ \langle \nabla_b \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k), b^k - \hat{b}^k\rangle \\
&+ \frac{L_v^k}{2}\|\mathbf{V}^k - \mathbf{V}^{k-1}\|_F^2 + \frac{L_v^k}{2}(b^k - \hat{b}^k)^2.
\end{aligned} \tag{16}$$

The inequalities (14) and (16) guarantee sufficient decrease of the objective and are required for convergence. If $L_u^k$ and $L_v^k$ are taken as Lipschitz constants of $\nabla_{(\mathbf{U},b)}\ell(\mathbf{U}, \mathbf{V}^{k-1}, b)$ and $\nabla_{(\mathbf{V},b)}\ell(\mathbf{U}^k, \mathbf{V}, b)$, then the two inequalities must hold. In our dynamical updating rule, note that in (13) and (15), we allow $n_u^k$ and $n_v^k$ to be negative, namely, $L_u^k$ and $L_v^k$ can be smaller than their previous values. Moreover, $n_u^k$ and $n_v^k$ must be finite if the sequence $\{(\mathbf{U}^k, \mathbf{V}^k)\}$ is bounded, and thus the updates in (13) and (15) are well-defined.

## 4 Convergence Analysis

We now establish the global convergence of our algorithm, as well as estimate its asymptotic convergence rate.

**Assumption 4.1** *Assume the objective function $F$ is lower bounded and the problem* (4) *has at least one stationary point. In addition, assume the sequence $\{\mathbf{W}^k\}$ is bounded.*

**Remark 4.1** *According to* (11)*, $L_u^k, L_v^k$ must be bounded if $\{\mathbf{W}^k\}$ is bounded. In addition, for the regularization terms, $r_1$ set by* (5a) *and $r_2$ taken as* (5b)*, then $F$ is lower bounded by* zero*, and* (4) *has at least one solution.*

**Theorem 4.1 (Subsequence Convergence)** *Under Assumption 4.1, let $\{\mathbf{W}^k\}$ be the sequence generated from Algorithm 2. Then any limit point $\bar{\mathbf{W}}$ of $\{\mathbf{W}^k\}$ is a stationary point of* (4)*.*

**Proof.** From Lemma 2.3 of [16], we have

$$F(\mathbf{W}^{k-1}) - F(\mathbf{U}^k, \hat{b}^k, \mathbf{V}^{k-1}) \geq \frac{L_u^k}{2}\big(\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2\big),$$

and

$$F(\mathbf{U}^k, \hat{b}^k, \mathbf{V}^{k-1}) - F(\mathbf{W}^k) \geq \frac{L_v^k}{2}\big(\|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |\hat{b}^k - b^k|^2\big).$$

Assume $\min(L_u^k, L_v^k) \geq L_{\min}$ for all $k$. Summing up the above two inequality gives

$$F(\mathbf{W}^{k-1}) - F(\mathbf{W}^k) \geq \frac{L_{\min}}{2}\big(\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + \|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2 + |\hat{b}^k - b^k|^2\big), \quad (17)$$

which yields

$$F(\mathbf{W}^0) - F(\mathbf{W}^N) \geq \sum_{k=1}^{N}\big(\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + \|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2 + |\hat{b}^k - b^k|^2\big).$$

Letting $N \to \infty$ and observing $F \geq 0$, we have

$$\sum_{k=1}^{\infty}\big(\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + \|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2 + |\hat{b}^k - b^k|^2\big) \leq \infty.$$

Hence, $\mathbf{W}^k - \mathbf{W}^{k-1} \to \mathbf{0}$.

Let $\bar{\mathbf{W}}$ be a limit point. Hence, there exists a subsequence $\{\mathbf{W}^k\}_{k\in\mathcal{K}}$ converging to $\bar{\mathbf{W}}$. Passing to another subsequence, we can assume that $\{L_u^k\}_{k\in\mathcal{K}}$ and $\{L_v^k\}_{k\in\mathcal{K}}$ converge to $\bar{L}_u$ and $\bar{L}_v$ respectively. Note that $\{\mathbf{W}^{k-1}\}_{k\in\mathcal{K}}$ also converges to $\bar{\mathbf{W}}$ and $\{\hat{b}^k\}_{k\in\mathcal{K}} \to \bar{b}$. Letting $k \in \mathcal{K}$ and $k \to \infty$ in (7a), we have

$$\bar{\mathbf{U}} = \operatorname*{argmin}_{\mathbf{U}} \langle \nabla_{\mathbf{U}}\ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}), \mathbf{U} - \bar{\mathbf{U}}\rangle + \frac{\bar{L}_u}{2}\|\mathbf{U} - \bar{\mathbf{U}}\|_F^2 + r_1(\mathbf{U}),$$

which implies $\mathbf{0} \in \nabla_{\mathbf{U}}\ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}) + \partial r_1(\bar{\mathbf{U}})$. Similarly, one can show $\mathbf{0} \in \nabla_{\mathbf{V}}\ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}) + \partial r_2(\bar{\mathbf{V}})$ and $\nabla_b \ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}) = 0$. Hence, $\bar{\mathbf{W}}$ is a critical point. ■

In order to establish global convergence, we utilize Kurdyka-Łojasiewicz inequality [17–19] defined below.

**Definition 4.1 (Kurdyka-Łojasiewicz Inequality)** *A function $F$ is said to satisfy the Kurdyka-Łojasiewicz inequality at point $\bar{\mathbf{W}}$, if there exists $\theta \in [0, 1)$ such that*

$$\frac{|F(\mathbf{W}) - F(\bar{\mathbf{W}})|^\theta}{dist(\mathbf{0}, \partial F(\mathbf{W}))} \tag{18}$$

*is bounded for any $\mathbf{W}$ near $\bar{\mathbf{W}}$, where $\partial F(\mathbf{W})$ is the limiting subdifferential [20] of $F$ at $\mathbf{W}$, and $dist(\mathbf{0}, \partial F(\mathbf{W})) \triangleq \min\{\|\mathbf{Y}\|_F : \mathbf{Y} \in \partial F(\mathbf{W})\}$.*

**Theorem 4.2 (Global Convergence)** *Suppose Assumption 4.1 holds and $F$ satisfies the Kurdyka-Łojasiewicz inequality at a limit point $\bar{\mathbf{W}}$ of $\{\mathbf{W}^k\}$. Then $\mathbf{W}^k$ converges to $\bar{\mathbf{W}}$.*

**Proof.** The boundedness of $\{\mathbf{W}^k\}$ implies that all intermediate points are bounded. Hence, there exists a constant $L_{\max}$ such that $L_u^k, L_v^k \leq L_{\max}$ for all $k$, and also there is a constant $L_G$ such that for all $k$

$$\|\nabla_{\mathbf{U}}\ell(\mathbf{W}^k) - \nabla_{\mathbf{U}}\ell(\mathbf{W}^{k-1})\|_F \leq L_G\|\mathbf{W}^k - \mathbf{W}^{k-1}\|_F, \tag{19a}$$

$$\|\nabla_{\mathbf{V}}\ell(\mathbf{W}^k) - \nabla_{\mathbf{V}}\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F \leq L_G\|\mathbf{W}^k - (\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F, \tag{19b}$$

$$\|\nabla_b\ell(\mathbf{W}^k) - \nabla_b\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F \leq L_G\|\mathbf{W}^k - (\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F. \tag{19c}$$

Let $\bar{\mathbf{W}}$ be a limit point of $\{\mathbf{W}^k\}$ and assume $F$ satisfies KL-inequality within $\mathbb{B}_\rho(\bar{\mathbf{W}}) \triangleq \{\mathbf{W} : \|\mathbf{W} - \bar{\mathbf{W}}\|_F \leq \rho\}$, namely, there exists constants $0 \leq \theta < 1$ and $C > 0$ such that

$$\frac{|F(\mathbf{W}) - F(\bar{\mathbf{W}})|^\theta}{\mathrm{dist}(\mathbf{0}, \partial F(\mathbf{W}))} \leq C, \ \forall \mathbf{W} \in \mathbb{B}_\rho(\bar{\mathbf{W}}). \tag{20}$$

Noting $\mathbf{W}^k - \mathbf{W}^{k-1} \to \mathbf{0}$, $|b^k - \hat{b}^k| \to 0$, and the continuity of $\phi(s) = s^{1-\theta}$, we can take sufficiently large $k_0$ such that

$$2\|\mathbf{W}^{k_0} - \mathbf{W}^{k_0+1}\|_F + \|\bar{\mathbf{W}} - \mathbf{W}^{k_0}\|_F + |b^{k_0+1} - \hat{b}^{k_0+1}| + \frac{1}{\tilde{C}^2}\phi(F(\mathbf{W}^{k_0}) - F(\bar{\mathbf{W}})) \leq \rho, \tag{21}$$

where $\tilde{C} = \sqrt{\frac{(1-\theta)L_{\min}}{8C\cdot(3L_G+2L_{\max})}}$. Without loss of generality, we assume $k_0 = 0$ (i.e., take $\mathbf{W}^{k_0}$ as starting point), since the convergence of $\{\mathbf{W}^k\}_{k\geq 0}$ is equivalent to that of $\{\mathbf{W}^k\}_{k\geq k_0}$. In addition, we denote $F_k = F(\mathbf{W}^k) - F(\bar{\mathbf{W}})$ and note $F_k \geq 0$ from the non-increasing monotonicity of $\{F(\mathbf{W}^k)\}$.

From (7), we have

$$-\nabla_{\mathbf{U}}\ell(\mathbf{W}^{k-1})+\nabla_{\mathbf{U}}\ell(\mathbf{W}^{k})-L_u^k(\mathbf{U}^k-\mathbf{U}^{k-1})\in\partial r_1(\mathbf{U}^k)+\nabla_{\mathbf{U}}\ell(\mathbf{W}^k), \tag{22a}$$

$$-\nabla_{\mathbf{V}}\ell(\mathbf{U}^k,\mathbf{V}^{k-1},\hat{b}^k)+\nabla_{\mathbf{V}}\ell(\mathbf{W}^{k})-L_v^k(\mathbf{V}^k-\mathbf{V}^{k-1})\in\partial r_2(\mathbf{V}^k)+\nabla_{\mathbf{V}}\ell(\mathbf{W}^k), \tag{22b}$$

$$-\nabla_{b}\ell(\mathbf{U}^k,\mathbf{V}^{k-1},\hat{b}^k)+\nabla_{b}\ell(\mathbf{W}^{k})-L_v^k(b^k-\hat{b}^k)=\nabla_{b}\ell(\mathbf{W}^k). \tag{22c}$$

Hence,

$$\begin{aligned}
&\operatorname{dist}(\mathbf{0},\partial F(\mathbf{W}^k))\\
\le&\|\nabla_{\mathbf{U}}\ell(\mathbf{W}^k)-\nabla_{\mathbf{U}}\ell(\mathbf{W}^{k-1})\|_F+L_u^k\|\mathbf{U}^k-\mathbf{U}^{k-1}\|_F+\|\nabla_{\mathbf{V}}\ell(\mathbf{W}^k)-\nabla_{\mathbf{V}}\ell(\mathbf{U}^k,\mathbf{V}^{k-1},\hat{b}^k)\|_F\\
&+L_v^k\|\mathbf{V}^k-\mathbf{V}^{k-1}\|_F+\|\nabla_b\ell(\mathbf{W}^k)-\nabla_b\ell(\mathbf{U}^k,\mathbf{V}^{k-1},\hat{b}^k)\|_F+L_v^k|b^k-\hat{b}^k|\\
\le&(3L_G+2L_{\max})\big(\|\mathbf{W}^k-\mathbf{W}^{k-1}\|_F+|b^k-\hat{b}^k|\big).
\end{aligned} \tag{23}$$

Note (17) implies

$$F_k-F_{k+1}\ge\frac{L_{\min}}{4}\big(\|\mathbf{W}^{k+1}-\mathbf{W}^k\|_F^2+|b^{k+1}-\hat{b}^{k+1}|^2\big).$$

Assume $\mathbf{W}^k\in\mathbb{B}_\rho(\bar{\mathbf{W}})$ for $0\le k\le N$. We go to show $\mathbf{W}^{N+1}\in\mathbb{B}_\rho(\bar{\mathbf{W}})$. By the concavity of $\phi(s)=s^{1-\theta}$ and KL-inequality (20), we have

$$\phi(F_k)-\phi(F_{k+1})\ge\phi'(F_k)(F_k-F_{k+1})\ge\frac{(1-\theta)L_{\min}\big(\|\mathbf{W}^{k+1}-\mathbf{W}^k\|_F^2+|b^{k+1}-\hat{b}^{k+1}|^2\big)}{4C\cdot(3L_G+2L_{\max})\big(\|\mathbf{W}^k-\mathbf{W}^{k-1}\|_F+|b^k-\hat{b}^k|\big)}, \tag{24}$$

which together with Cauchy-Schwart inequality gives

$$\tilde{C}\big(\|\mathbf{W}^k-\mathbf{W}^{k+1}\|_F+|b^{k+1}-\hat{b}^{k+1}|\big)\le\frac{\tilde{C}}{2}\big(\|\mathbf{W}^{k-1}-\mathbf{W}^k\|_F+|b^k-\hat{b}^k|\big)+\frac{1}{2\tilde{C}}\big(\phi(F_k)-\phi(F_{k+1})\big). \tag{25}$$

Summing up the above inequality gives

$$\frac{\tilde{C}}{2}\sum_{k=1}^{N}\big(\|\mathbf{W}^k-\mathbf{W}^{k+1}\|_F+|b^{k+1}-\hat{b}^{k+1}|\big)\le\frac{\tilde{C}}{2}\big(\|\mathbf{W}^0-\mathbf{W}^1\|_F+|b^1-\hat{b}^1|\big)+\frac{1}{2\tilde{C}}\big(\phi(F_0)-\phi(F_{N+1})\big). \tag{26}$$

Hence,

$$\begin{aligned}
\|\mathbf{W}^{N+1}-\bar{\mathbf{W}}\|_F\le&\sum_{k=1}^{N}\|\mathbf{W}^k-\mathbf{W}^{k+1}\|_F+\|\mathbf{W}^0-\mathbf{W}^1\|_F+\|\bar{\mathbf{W}}-\mathbf{W}^0\|_F\\
\le&2\|\mathbf{W}^0-\mathbf{W}^1\|_F+\|\bar{\mathbf{W}}-\mathbf{W}^0\|_F+|b^1-\hat{b}^1|+\frac{1}{\tilde{C}^2}\phi(F_0)\le\rho,
\end{aligned}$$

where the last inequality is from (21). Hence, $\mathbf{W}^{N+1}\in\mathbb{B}_\rho(\bar{\mathbf{W}})$, and by induction, $\mathbf{W}^k\in\mathbb{B}_\rho(\bar{\mathbf{W}})$ for all $k$. Therefore, (26) holds for all $N$. Letting $N\to\infty$ in (26) yields

$$\sum_{k=1}^{\infty}\|\mathbf{W}^k-\mathbf{W}^{k+1}\|_F<\infty.$$

Hence, $\{\mathbf{W}^k\}$ is a Cauchy sequence and thus converges to the limit point $\bar{\mathbf{W}}$. ■

**Remark 4.2** *Note that the logistic function $\ell$ is real analytic. If $r_1$ and $r_2$ are taken as in (5), then they are semi-algebraic functions [21], and, according to [22], $F$ satisfies the Kurdyka-Łojasiewicz inequality at every point.*

**Theorem 4.3 (Convergence Rate)** *Depending on $\theta$ in (18), we have the following convergence rates:*

1. *If $\theta = 0$, then $\mathbf{W}^k$ converges to $\bar{\mathbf{W}}$ in finite iterations;*
2. *If $\theta \in (0, \frac{1}{2}]$, then $\mathbf{W}^k$ converges to $\bar{\mathbf{W}}$ at least linearly, i.e., $\|\mathbf{W}^k - \bar{\mathbf{W}}\|_F \leq C\tau^k$ for some positive constants $C$ and $\tau < 1$;*
3. *If $\theta \in (\frac{1}{2}, 1)$, then $\mathbf{W}^k$ converges to $\bar{\mathbf{W}}$ at least sublinearly. Specifically, $\|\mathbf{W}^k - \bar{\mathbf{W}}\|_F \leq Ck^{-\frac{1-\theta}{2\theta-1}}$ for some constant $C > 0$.*

**Proof.** We estimate the convergence rates for different $\theta$ in (20).

**Case 1:** $\theta = 0$. We claim $\mathbf{W}^k$ converges to $\bar{\mathbf{W}}$ in finite iterations, i.e., there is $k_0$ such that $\mathbf{W}^k = \bar{\mathbf{W}}$ for all $k \geq k_0$. Otherwise, $F(\mathbf{W}^k) > F(\bar{\mathbf{W}})$ for all $k$ since if $F(\mathbf{W}^{k_0}) = F(\bar{\mathbf{W}})$ then $\mathbf{W}^k = \bar{\mathbf{W}}$ for all $k \geq k_0$. By KL-inequality (20), we have $C \cdot \mathrm{dist}(\mathbf{0}, \partial F(\mathbf{W}^k)) \geq 1$ for all $k$. However, (22) indicates $\mathrm{dist}(\mathbf{0}, \partial F(\mathbf{W}^k)) \to 0$ as $k \to \infty$. Therefore, if $\theta = 0$, then $\mathbf{W}^k$ converges to $\bar{\mathbf{W}}$ in finite iterations.

**Case 2:** $\theta \in (0, \frac{1}{2}]$**.** Denote $S_N = \sum_{k=N}^{\infty} \left(\|\mathbf{W}^k - \mathbf{W}^{k+1}\|_F + |b^{k+1} - \hat{b}^{k+1}|\right)$. Note that (25) holds for all $k$. Summing (25) over $k$ gives $S_N \leq S_{N-1} - S_N + \frac{1}{2\tilde{C}^2} F_N^{1-\theta}$. By (20) and (23), we have

$$F_N^{1-\theta} = (F_N^{\theta})^{\frac{1-\theta}{\theta}} \leq \left(C \cdot (3L_G + 2L_{\max})\right)^{\frac{1-\theta}{\theta}} (S_{N-1} - S_N)^{\frac{1-\theta}{\theta}}.$$

Hence,

$$S_N \leq S_{N-1} - S_N + \hat{C}(S_{N-1} - S_N)^{\frac{1-\theta}{\theta}}, \tag{27}$$

where $\hat{C} = \frac{1}{2\tilde{C}^2}\left(C \cdot (3L_G + 2L_{\max})\right)^{\frac{1-\theta}{\theta}}$. Note that $S_{N-1} - S_N \leq 1$ as $N$ is sufficiently large, and also $\frac{1-\theta}{\theta} \geq 1$ when $\theta \in (0, \frac{1}{2}]$. Therefore, $(S_{N-1} - S_N)^{\frac{1-\theta}{\theta}} \leq S_{N-1} - S_N$, and thus (27) implies $S_N \leq (1 + \hat{C})(S_{N-1} - S_N)$. Hence, $S_N \leq \frac{1+\hat{C}}{2+\hat{C}} S_{N-1} \leq \left(\frac{1+\hat{C}}{2+\hat{C}}\right)^N S_0$. Notting that $\|\mathbf{W}^N - \bar{\mathbf{W}}\|_F \leq S_N$, we have

$$\|\mathbf{W}^N - \bar{\mathbf{W}}\|_F \leq \left(\frac{1+\hat{C}}{2+\hat{C}}\right)^N S_0.$$

**Case 3:** $\theta \in (\frac{1}{2}, 1)$**.** Note $\frac{1-\theta}{\theta} < 1$. Hence, (27) implies

$$S_N \leq (1 + \hat{C})(S_{N-1} - S_N)^{\frac{1-\theta}{\theta}}.$$

Through the same argument in the proof of Theorem 2 of [23], we can show

$$S_N \leq c \cdot N^{-\frac{1-\theta}{2\theta-1}},$$

for some constant $c$. This completes the proof. ■

**Remark 4.3** *Note that the value of $\theta$ depends not only on $F$ but also on $\bar{\mathbf{W}}$. The paper [22] gives estimates for different classes of functions. Since the limit point is not known ahead, we cannot estimate $\theta$. However, our numerical results in Section 7 indicate that our algorithm converges asympototically superlinearly and thus $\theta$ should be less than $\frac{1}{2}$ for our tests.*

# 5 Statistical Analysis

# 6 Extensions to multi-class model

We can further generalize binary-class bilinear logistic regression (B-BLR) to multi-class bilinear logistic regression (M-BLR), which assumes each sample $\{\mathbf{x}_i\}$ to belong to $(m+1)$ classes and label $y_i \in \{1, 2, \cdots, m+1\}$. M-LR aims at finding $(m+1)$ hyperplanes $\{\mathbf{x} : \mathbf{w}_c^\top \mathbf{x} + b_c = 0\}_{c=1}^{m+1}$ to seperate these samples. According to the logistic model, the conditional probability for $y_i$ based on sample $\mathbf{x}_i$ is

$$P(y_i = c|\mathbf{x}_i, \mathbf{w}, \mathbf{b}) = \frac{\exp[\mathbf{w}_c^\top \mathbf{x}_i + b_c]}{\sum_{j=1}^{m+1} \exp[\mathbf{w}_j^\top \mathbf{x}_i + b_j]},\ c = 1, \cdots, m+1. \tag{28}$$

Because of the normalization condition $\sum_{c=1}^{m+1} P(y_i = c|\mathbf{x}_i, \mathbf{w}, \mathbf{b}) = 1$, one $(\mathbf{w}_c, b_c)$ needs not be estimated. Without loss of generality, we set $(\mathbf{w}_{m+1}, b_{m+1})$ to *zero*. Let $y_{ic} = 1$ if $y_i = c$ and $y_{ic} = 0$ otherwise. Then (28) becomes

$$P(y_i|\mathbf{x}_i, \mathbf{w}, \mathbf{b}) = \frac{\exp[\sum_{c=1}^{m} y_{ic}(\mathbf{w}_c^\top \mathbf{x}_i + b_c)]}{1 + \sum_{c=1}^{m} \exp[\mathbf{w}_c^\top \mathbf{x}_i + b_c]}.$$

The average negative log-likelihood function is

$$\mathcal{L}(\mathbf{w}, \mathbf{b}) = -\frac{1}{n}\sum_{i=1}^{n} \log P(y_i|\mathbf{x}_i, \mathbf{w}, \mathbf{b}) = \frac{1}{n}\sum_{i=1}^{n}\left(\log\left(1 + \sum_{c=1}^{m} \exp[\mathbf{w}_c^\top \mathbf{x}_i + b_c]\right) - \sum_{c=1}^{m} y_{ic}(\mathbf{w}_c^\top \mathbf{x}_i + b_c)\right) \tag{29}$$

To perform MLE for $(\mathbf{w}, \mathbf{b})$, one can minimize $\mathcal{L}(\mathbf{w}, \mathbf{b})$. Under the setting of BLR, namely, each sample is a matrix and each weight $\mathbf{w}_c$ has the form of $\mathbf{U}_c\mathbf{V}_c^\top$, the loss function in (29) becomes

$$\mathcal{L}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}, \mathbf{b}) = \frac{1}{n}\sum_{i=1}^{n}\left(\log\left(1 + \sum_{c=1}^{m} \exp[\mathrm{tr}(\mathbf{U}_c^\top \mathbf{X}_i \mathbf{V}_c) + b_c]\right) - \sum_{c=1}^{m} y_{ic}(\mathrm{tr}(\mathbf{U}_c^\top \mathbf{X}_i \mathbf{V}_c) + b_c)\right), \tag{30}$$

and (4) can be generalized to the regularized multi-class BLR (M-BLR)

$$\min_{\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}} \mathcal{L}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) + R_1(\boldsymbol{\mathcal{U}}) + R_2(\boldsymbol{\mathcal{V}}), \tag{31}$$

where $\boldsymbol{\mathcal{U}} = (\mathbf{U}_1, \cdots, \mathbf{U}_m), \boldsymbol{\mathcal{V}} = (\mathbf{V}_1, \cdots, \mathbf{V}_m)$ with $\mathbf{U}_c \in \mathbb{R}^{S\times K}$ and $\mathbf{V}_c \in \mathbb{R}^{T\times K}$ for each class $c$, and $R_1$ and $R_2$ are used to promote priori structures on $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}$, respectively.

The algorithm for solving (31) can be derived in a similar way as that for (4). We alternatively update $(\boldsymbol{\mathcal{U}}, \mathbf{b})$ and $(\boldsymbol{\mathcal{V}}, \mathbf{b})$ by

$$\boldsymbol{\mathcal{U}}^k = \operatorname*{argmin}_{\boldsymbol{\mathcal{U}}} \langle \nabla_{\boldsymbol{\mathcal{U}}} \mathcal{L}(\boldsymbol{\mathcal{U}}^{k-1}, \boldsymbol{\mathcal{V}}^{k-1}, \mathbf{b}^{k-1}), \boldsymbol{\mathcal{U}} - \boldsymbol{\mathcal{U}}^{k-1}\rangle + \frac{\gamma_u^k}{2}\|\boldsymbol{\mathcal{U}} - \boldsymbol{\mathcal{U}}^{k-1}\|_F^2 + R_1(\boldsymbol{\mathcal{U}}), \tag{32a}$$

$$\hat{\mathbf{b}}^k = \operatorname*{argmin}_{\mathbf{b}} \langle \nabla_{\mathbf{b}} \mathcal{L}(\boldsymbol{\mathcal{U}}^{k-1}, \boldsymbol{\mathcal{V}}^{k-1}, \mathbf{b}^{k-1}), \mathbf{b} - \mathbf{b}^{k-1}\rangle + \frac{\gamma_u^k}{2}\|\mathbf{b} - \mathbf{b}^{k-1}\|_2^2, \tag{32b}$$

$$\boldsymbol{\mathcal{V}}^k = \operatorname*{argmin}_{\boldsymbol{\mathcal{V}}} \langle \nabla_{\boldsymbol{\mathcal{V}}} \mathcal{L}(\boldsymbol{\mathcal{U}}^{k}, \boldsymbol{\mathcal{V}}^{k-1}, \hat{\mathbf{b}}^k), \boldsymbol{\mathcal{V}} - \boldsymbol{\mathcal{V}}^{k-1}\rangle + \frac{\gamma_v^k}{2}\|\boldsymbol{\mathcal{V}} - \boldsymbol{\mathcal{V}}^{k-1}\|_F^2 + R_2(\boldsymbol{\mathcal{V}}), \tag{32c}$$

$$\mathbf{b}^k = \operatorname*{argmin}_{\mathbf{b}} \langle \nabla_{\mathbf{b}} \mathcal{L}(\boldsymbol{\mathcal{U}}^{k}, \boldsymbol{\mathcal{V}}^{k-1}, \hat{\mathbf{b}}^k), \mathbf{b} - \hat{\mathbf{b}}^k\rangle + \frac{\gamma_v^k}{2}\|\mathbf{b} - \hat{\mathbf{b}}^k\|_2^2. \tag{32d}$$

The pseudocode is shown in Algorithm 3.

**Algorithm 3** Alternating proximal gradient method for (31)

**Input:** training data $\{\mathbf{X}_i, y_i\}_{i=1}^n$ with $y_i \in \{1, \cdots, m+1\}$;
**Initialization:** choose starting points $\boldsymbol{\mathcal{U}}^{-1} = \boldsymbol{\mathcal{U}}^0, \boldsymbol{\mathcal{V}}^{-1} = \boldsymbol{\mathcal{V}}^0$ and $\mathbf{b}^{-1} = \mathbf{b}^0$.
**for** $k = 1, 2, \cdots$ **do**
  Update $(\boldsymbol{\mathcal{U}}, \mathbf{b})$ to $(\boldsymbol{\mathcal{U}}^k, \hat{\mathbf{b}}^k)$ by (32a) and (32b);
  Update $(\boldsymbol{\mathcal{V}}, \mathbf{b})$ to $(\boldsymbol{\mathcal{U}}^k, \mathbf{b}^k)$ by (32c) and (32d).
  **if** Some stopping criterion is satisfied **then**
    Stop and output $(\boldsymbol{\mathcal{U}}^k, \boldsymbol{\mathcal{V}}^k, \mathbf{b}^k)$.
  **end if**
**end for**

We choose $\gamma_u^k$ and $\gamma_u^k$ in a similar way as $L_u^k$ and $L_v^k$ in (13) and (15). As long as $\{\boldsymbol{\mathcal{U}}^k\}$ and $\{\boldsymbol{\mathcal{V}}^k\}$ are bounded, $\gamma_u^k$ and $\gamma_u^k$ are finite due to the following lemma. Hence, the selection of $\gamma_u^k$ and $\gamma_u^k$ are well defined.

**Lemma 6.1** *The partial gradients $\nabla_{(\boldsymbol{\mathcal{U}},\mathbf{b})}\mathcal{L}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b})$ and $\nabla_{(\boldsymbol{\mathcal{V}},\mathbf{b})}\mathcal{L}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b})$ are Lipschitz continuous with constants*

$$\gamma_u = \frac{\sqrt{2}m}{n}\sum_{c=1}^{m}\sum_{i=1}^{n}\big(\|\mathbf{X}_i\mathbf{V}_c\|_F + 1\big)\big(\max_j \|\mathbf{X}_i\mathbf{V}_j\|_F + 1\big), \tag{33a}$$

$$\gamma_v = \frac{\sqrt{2}m}{n}\sum_{c=1}^{m}\sum_{i=1}^{n}\big(\|\mathbf{X}_i^\top\mathbf{U}_c\|_F + 1\big)\big(\max_j \|\mathbf{X}_i^\top\mathbf{U}_j\|_F + 1\big). \tag{33b}$$

**Proof.** By straightforward calculation, we have

$$\begin{aligned}
\nabla_{\mathbf{U}_c}\mathcal{L}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) =& \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\exp[\mathrm{tr}(\mathbf{U}_c^\top\mathbf{X}_i\mathbf{V}_c)+b_c]}{1+\sum_{j=1}^{m}\exp[\mathrm{tr}(\mathbf{U}_j^\top\mathbf{X}_i\mathbf{V}_j)+b_j]}\mathbf{X}_i\mathbf{V}_c - y_{ic}\mathbf{X}_i\mathbf{V}_c\right),\\
\nabla_{\mathbf{V}_c}\mathcal{L}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) =& \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\exp[\mathrm{tr}(\mathbf{U}_c^\top\mathbf{X}_i\mathbf{V}_c)+b_c]}{1+\sum_{j=1}^{m}\exp[\mathrm{tr}(\mathbf{U}_j^\top\mathbf{X}_i\mathbf{V}_j)+b_j]}\mathbf{X}_i^\top\mathbf{U}_c - y_{ic}\mathbf{X}_i^\top\mathbf{U}_c\right),\\
\nabla_{b_c}\mathcal{L}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) =& \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\exp[\mathrm{tr}(\mathbf{U}_c^\top\mathbf{X}_i\mathbf{V}_c)+b_c]}{1+\sum_{j=1}^{m}\exp[\mathrm{tr}(\mathbf{U}_j^\top\mathbf{X}_i\mathbf{V}_j)+b_j]} - y_{ic}\right).
\end{aligned}$$

Hence, for any $(\boldsymbol{\mathcal{U}},\mathbf{b})$ and $(\tilde{\boldsymbol{\mathcal{U}}},\tilde{\mathbf{b}})$,

$$\begin{aligned}
&\|\nabla_{\mathbf{U}_c}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) - \nabla_{\mathbf{U}_c}(\tilde{\boldsymbol{\mathcal{U}}},\boldsymbol{\mathcal{V}},\tilde{\mathbf{b}})\|_F\\
\le&\frac{1}{n}\sum_{i=1}^{n}\left|\frac{\exp[\mathrm{tr}(\mathbf{U}_c^\top\mathbf{X}_i\mathbf{V}_c)+b_c]}{1+\sum_{j=1}^{m}\exp[\mathrm{tr}(\mathbf{U}_j^\top\mathbf{X}_i\mathbf{V}_j)+b_j]} - \frac{\exp[\mathrm{tr}(\tilde{\mathbf{U}}_c^\top\mathbf{X}_i\mathbf{V}_c)+\tilde{b}_c]}{1+\sum_{j=1}^{m}\exp[\mathrm{tr}(\tilde{\mathbf{U}}_j^\top\mathbf{X}_i\mathbf{V}_j)+\tilde{b}_j]}\right|\|\mathbf{X}_i\mathbf{V}_c\|_F\\
\le&\frac{\sqrt{m}}{n}\sum_{i=1}^{n}\left\|\left[\left(\mathrm{tr}(\mathbf{U}_j^\top\mathbf{X}_i\mathbf{V}_j)+b_j\right) - \left(\mathrm{tr}(\tilde{\mathbf{U}}_j^\top\mathbf{X}_i\mathbf{V}_j)+\tilde{b}_j\right)\right]_{1\le j\le m}\right\|_2\|\mathbf{X}_i\mathbf{V}_c\|_F\\
\le&\frac{\sqrt{m}}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\|\mathbf{U}_j-\tilde{\mathbf{U}}_j\|_F\|\mathbf{X}_i\mathbf{V}_j\|_F + |b_j-\tilde{b}_j|\right)\|\mathbf{X}_i\mathbf{V}_c\|_F\\
\le&\frac{\sqrt{2}m}{n}\left(\sum_{i=1}^{n}\|\mathbf{X}_i\mathbf{V}_c\|_F\big(\max_j\|\mathbf{X}_i\mathbf{V}_j\|_F+1\big)\right)\left\|(\boldsymbol{\mathcal{U}},\mathbf{b})-(\tilde{\boldsymbol{\mathcal{U}}},\tilde{\mathbf{b}})\right\|_F,
\end{aligned}$$

where in the second inequality we have used

$$\left|\frac{\exp(s_c)}{1+\sum_{j=1}^{m}\exp(s_j)} - \frac{\exp(q_c)}{1+\sum_{j=1}^{m}\exp(q_j)}\right| \le \sqrt{m}\|\mathbf{s}-\mathbf{q}\|_2,$$

and the last inequality uses

$$\sum_{j=1}^{m}\left(\|\mathbf{U}_j-\tilde{\mathbf{U}}_j\|_F + |b_j-\tilde{b}_j|\right) \le \sqrt{2m}\left\|(\boldsymbol{\mathcal{U}},\mathbf{b})-(\tilde{\boldsymbol{\mathcal{U}}},\tilde{\mathbf{b}})\right\|_F.$$

Similarly, we have

$$|\nabla_{b_c}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) - \nabla_{b_c}(\tilde{\boldsymbol{\mathcal{U}}},\boldsymbol{\mathcal{V}},\tilde{\mathbf{b}})| \le \frac{\sqrt{2}m}{n}\left(\sum_{i=1}^{n}\big(\max_j\|\mathbf{X}_i\mathbf{V}_j\|_F+1\big)\right)\left\|(\boldsymbol{\mathcal{U}},\mathbf{b})-(\tilde{\boldsymbol{\mathcal{U}}},\tilde{\mathbf{b}})\right\|_F$$

Noting $\nabla_{(\boldsymbol{\mathcal{U}},\mathbf{b})} = (\nabla_{\mathbf{U}_1},\cdots,\nabla_{\mathbf{U}_m},\nabla_{b_1},\cdots,\nabla_{b_m})$ gives

$$\begin{aligned}
&\|\nabla_{(\boldsymbol{\mathcal{U}},\mathbf{b})}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) - \nabla_{(\boldsymbol{\mathcal{U}},\mathbf{b})}(\tilde{\boldsymbol{\mathcal{U}}},\boldsymbol{\mathcal{V}},\tilde{\mathbf{b}})\|_F\\
\le&\sum_{c=1}^{m}\left(\|\nabla_{\mathbf{U}_c}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) - \nabla_{\mathbf{U}_c}(\tilde{\boldsymbol{\mathcal{U}}},\boldsymbol{\mathcal{V}},\tilde{\mathbf{b}})\|_F + |\nabla_{b_c}(\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{V}},\mathbf{b}) - \nabla_{b_c}(\tilde{\boldsymbol{\mathcal{U}}},\boldsymbol{\mathcal{V}},\tilde{\mathbf{b}})|\right)\\
\le&\sum_{c=1}^{m}\frac{\sqrt{2}m}{n}\left(\sum_{i=1}^{n}\big(\|\mathbf{X}_i\mathbf{V}_c\|_F+1\big)\big(\max_j\|\mathbf{X}_i\mathbf{V}_j\|_F+1\big)\right)\left\|(\boldsymbol{\mathcal{U}},\mathbf{b})-(\tilde{\boldsymbol{\mathcal{U}}},\tilde{\mathbf{b}})\right\|_F.
\end{aligned}$$

This completes the proof of (33a), and (33b) can be shown in the same way. ■

We will take $R_1(\boldsymbol{\mathcal{U}}) = \sum_{c=1}^{m} r_1(\mathbf{U}_c)$ and $R_2(\boldsymbol{\mathcal{V}}) = \sum_{c=1}^{m} r_2(\mathbf{V}_c)$, where $r_1$ and $r_2$ are the same as those in (4). Note that each subproblem in (32) can be decoupled into $m$ independent problems, and they can be solved by the same method as discussed in section 3.3. We do not repeat it here.

# 7 Numerical Results

## 7.1 Implementation

Since the variational problem (4) is non-convex, the starting point is significant for both the solution quality and convergence speed of our algorithm. Throughout our tests, we simply set $b^0 = 0$ and chose $(\mathbf{U}^0, \mathbf{V}^0)$ as follows.

Let $\mathbf{X}^{av} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{X}_i$. Then set $\mathbf{U}^0$ to the negative of the first $r$ left singular vectors and $\mathbf{V}^0$ to the first $r$ right singular vectors of $\mathbf{X}^{av}$ corresponding to its first $r$ largest singular values.

The intuition of choosing such $(\mathbf{U}^0, \mathbf{V}^0)$ is that it is one minimizer of $\frac{1}{n}\sum_{i=1}^{n} \mathrm{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V})$, which is exactly the first-order Taylor expansion of $\ell(\mathbf{U}, \mathbf{V}, 0)$ at the origin, under constraints $\mathbf{U}^\top\mathbf{U} = \mathbf{I}$ and $\mathbf{V}^\top\mathbf{V} = \mathbf{I}$. Unless specified, the algorithms were terminated if they ran over 500 iterations or the relative error $q^k \leq 10^{-3}$.

## 7.2 Scalability

In order to demonstrate the computational benefit of proximal method, we compared Algorithm 2 with Algorithm 1 on randomly generated data. Each data point[1] in class "+1" was generated by MATLAB command `randn(s,t)+1` and each one in class "-1" by `randn(s,t)-1`. The sample size was fixed to $n = 100$, and the dimensions were kept by $s = t$ with $s$ varying among $\{50, 100, 250, 500, 750, 1000\}$. We tested two sets of parameters for the scalability test. We ran each algorithm with one set of parameters for 5 times with different random data.

Table 1 shows the average running time and the median number of iterations. From the table, we see that both Algorithm 1 and Algorithm 2 are scalable to large-scale dataset and converge within the given tolerance after quite a few iterations. The per-iteration running time increases almost linearly with respect to the data size. In addition, Algorithm 2 is much faster than Algorithm 1 in terms of running time. Note the degree of speedup depends on the parameters. In the first testing, where $\ell_2$ regularization dominates ($\mu_1 = \nu_1 = 0.1$, $\mu_2 = \nu_2 = 1$), Algorithm 2 is twice as fast as Algorithm 1. In the second testing, where

[1] We use synthetic data simply for scalability and speed test. For other numerical experiments, we use real-world datasets.

$\ell_1$ regularization dominates ($\mu_1 = \nu_1 = 0.1$, $\mu_2 = \nu_2 = 0$), Algorithm 2 is about 20 times faster than Algorithm 1.

Table 1: Scalability and comparison of Algorithm 1 and Algorithm 2. Shown are the average running time and median number of iterations.

| | Algorithm 1 | | Algorithm 2 | |
|---|---|---|---|---|
| $\mu_1 = \nu_1 = 0.1$, $\mu_2 = \nu_2 = 1$ | | | | |
| $(s, t)$ | time (sec.) | iter | time (sec.) | iter |
| $(50, 50)$ | 0.79 | 5 | **0.03** | 9 |
| $(100, 100)$ | 1.13 | 6 | **0.06** | 11 |
| $(250, 250)$ | 3.89 | 6 | **0.56** | 31 |
| $(500, 500)$ | 9.96 | 5 | **1.80** | 4 |
| $(750, 750)$ | 18.60 | 7 | **4.04** | 4 |
| $(1000, 1000)$ | 16.25 | 3 | **7.92** | 4 |
| $\mu_1 = \nu_1 = 0.1$, $\mu_2 = \nu_2 = 0$ | | | | |
| $(s, t)$ | time (sec.) | iter | time (sec.) | iter |
| $(50, 50)$ | 6.87 | 17 | **0.37** | 282 |
| $(100, 100)$ | 14.39 | 29 | **0.38** | 47 |
| $(250, 250)$ | 21.73 | 8 | **3.49** | 28 |
| $(500, 500)$ | 78.32 | 7 | **4.07** | 11 |
| $(750, 750)$ | 129.23 | 8 | **4.31** | 4 |
| $(1000, 1000)$ | 218.49 | 9 | **8.19** | 4 |

### 7.3 Convergence Behavior

We ran Algorithm 2 up to 600 iterations for the unregularized model ($\mu_1 = \nu_1 = \mu_2 = \nu_2 = 0$), and $10^4$ iterations for the regularized model where we set $\mu_1 = \nu_1 = 0.01$ and $\mu_2 = \nu_2 = 0.5$. For both models, $r = 1$ was used. The last iterate was used as $\mathbf{W}^*$. The dataset is described in Section 6.1.1.

Figure 3 shows the convergence behavior of Algorithm 2 for solving (4) with different regularization terms. From the figure, we see that our algorithm converges pretty fast and the difference $\|\mathbf{W}^k - \mathbf{W}^*\|_F$ appears to decrease linearly at first and superlinearly eventually.

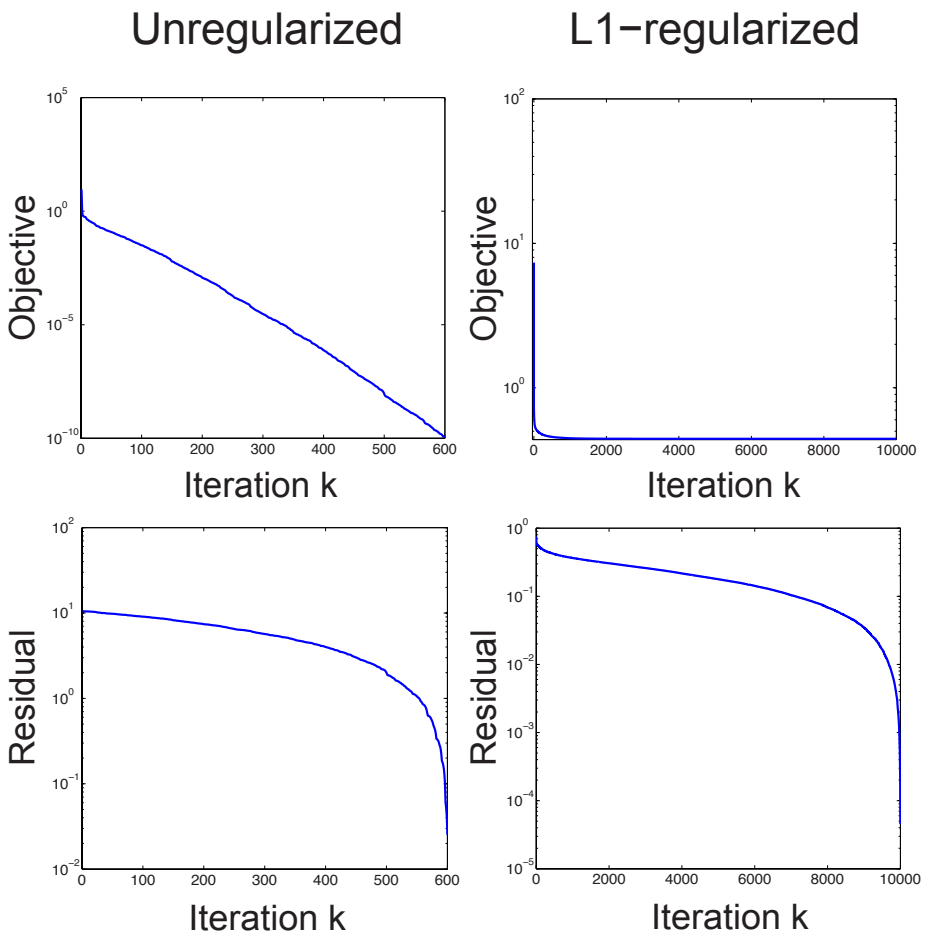


Figure 3: Convergence behavior for solving (4) using block coordinate proximal descent method. Top panel plots the objective function as a function of iteration. Bottom panel plots the residual $\|\mathbf{W}^k - \mathbf{W}^*\|_F$ as a function of iteration.

# 8 Applications

We apply sparse bilinear logistic regression to several real-world applications, and compare its generalization performance with logistic regression, sparse logistic regression and bilinear logistic regression. We also extend the sparse bilinear logistic regression from the binary case to multi-class case in several experiments.

## 8.1 Brain Computer Interface

### 8.1.1 Binary Case

We tested the classification performance of sparse bilinear logistic regression (4) on some EEG dataset with binary labels. We used the EEG dataset IVb from from BCI competition III [2] . Dataset IVb concerns motor imagery with uncued classification task. The 118 channel EEG was recorded from a healthy subject sitting in a comfortable chair with arms resting on armrests. Visual cues (letter presentation) were showed for 3.5 seconds, during which the subject performed: left hand, right foot, or tongue. The data was sampled at 100 Hz, and the cues of "left hand" and "right foot" were marked in the training data. We chose all the 210 marked data points for test and downsampled each point to have 100 temporal slices, namely, $s = 118, t = 100$ in this test.

In (4), there are five parameters $\mu_1, \mu_2, \nu_1, \nu_2$ and $r$ to be tuned. Leave-one-out cross validation was performed on the training dataset to tune these data. First, we fixed $\mu_1 = \mu_2 = \nu_1 = \nu_2 = 0$ (i.e., unregularized) and tuned $r$. Then, we fixed $r$ to the previously tuned one ($r = 1$ in this test) and selected the best $(\mu_1, \mu_2, \nu_1, \nu_2)$ from a $6 \times 5 \times 6 \times 5$ grid.

Table 2: Classification performance for BCI EEG dataset.

| Models | Prediction Accuracy |
|---|---|
| Logistic Regression | 0.75 |
| Sparse Logistic Regression | 0.76 |
| Bilinear Logistic Regression | 0.84 |
| Sparse Bilinear Logistic Regression | **0.89** |

Table 2 shows the prediction accuracy on the testing dataset [3]. We use the ROC analysis to compute the Az value (area under ROC curve) for both the unregularized model and the regularized model, where the best hyperparameters for the regularized model are tuned on the validation dataset using cross validation. We compare (sparse) logistic regression with

[2] `http://www.bbci.de/competition/iii/`

[3] In Table 2 - Table 5, higher prediction accuracy indicates better generalization performance.

(sparse) bilinear logistic regression. We solve the $\ell_1$-regularized logistic regression using FISTA [16]. We observe that bilinear logistic regression gives much better predictions than logistic regression. In addition, sparse bilinear logistic regression performs better than the unregularized bilinear logistic regression.

### 8.1.2 Multi-class Case

We further extended our sparse bilinear logistic regression to the multi-class case using one-vs-all method. The EEG dataset in this experiment was based on a cognitive experiment where the subject view images of three categories and tried to make a decision about the category [24]. The data was recorded at 2048 Hz using a 64-channel EEG cap. We downsampled this data to 100 Hz.

Table 3 shows classification performance for the multi-class classification. Consistently for all the three stimuli, bilinear logistic regression outperforms logistic regression, and sparse bilinear logistic regression further improves the generalization performance by introducing sparsity.

Table 3: Classification performance for multi-class EEG dataset.

| Models | Prediction Accuracy |
|---|---|
| Logistic Regression | 0.54 |
| Sparse Logistic Regression | 0.54 |
| Bilinear Logistic Regression | 0.55 |
| Sparse Bilinear Logistic Regression | **0.65** |

## 8.2 Separating Style and Content

As mentioned earlier, one benefit of the bilinear model is to separate style and content. In order to exploit this property, we classified images with various camera viewpoints. We used the Amsterdam Library of Object Images [4], where the frontal camera was used to record 72 viewpoints of the objects by rotating the object in the plane at 5° resolution from 0° to 355°. Figure 4 shows some sample images with various camera viewpoints.

Table 4 shows the comparison between (sparse) logistic regression and (sparse) bilinear logistic regression. We observe a significant improvement using the bilinear model, and sparse bilinear logistic regression achieves the best generalization performance.

[4] `http://staff.science.uva.nl/~aloi/`

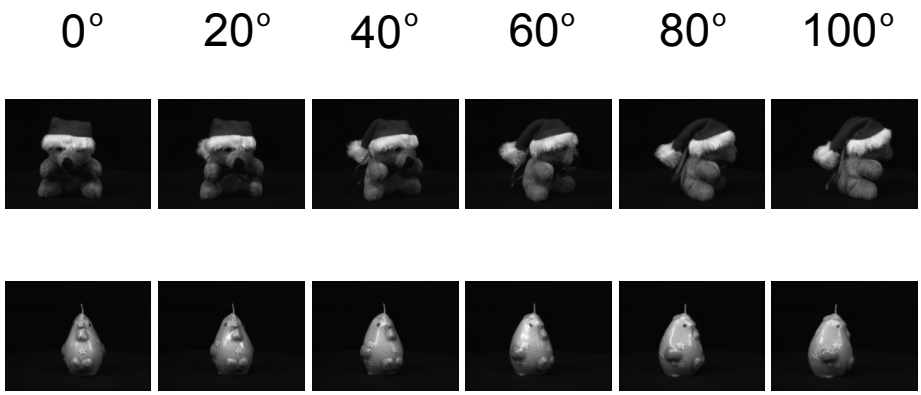


Figure 4: Some sample images with various camera viewpoints.

Table 4: Classification performance for images with various camera viewpoints.

| Models | Prediction Accuracy |
|---|---|
| Logistic Regression | 0.86 |
| Sparse Logistic Regression | 0.86 |
| Bilinear Logistic Regression | 0.94 |
| Sparse Bilinear Logistic Regression | **1.00** |

### 8.3 Visual Recognition of Videos

We used sparse bilinear logistic regression to videos [13], in the context of visual recognition for UCF sports action dataset [5]. Since the size of the original video is big, we reduced the dimensionality of feature space by extracting histograms based on SIFT descriptors for each frame.

Figure 5 illustrates such a procedure. We first built a vocabulary for the codebook assuming 100 words, using k-mean clustering based on all the SIFT descriptors across frames for all the videos. We then constructed histograms for each frame according to the codebook. Some tiling technique was used to improve the performance. This procedure reduced the feature space to $s = 400$ and $t = 55$.

We focused on five classes of sports action and we used the following abbreviations: Diving (Diving-Side), Riding (Riding-Horse), Run (Run-Side), Swing (Swing-Sideangle), Walk (Walk-Front). We picked 6 videos out of each class, and used 6-fold cross validation to test discrimination accuracy in the context of transfer learning.

Table 5 shows the classification performance for (sparse) logistic regression and (sparse)

[5] `http://crcv.ucf.edu/data/UCF\_Sports\_Action.php`

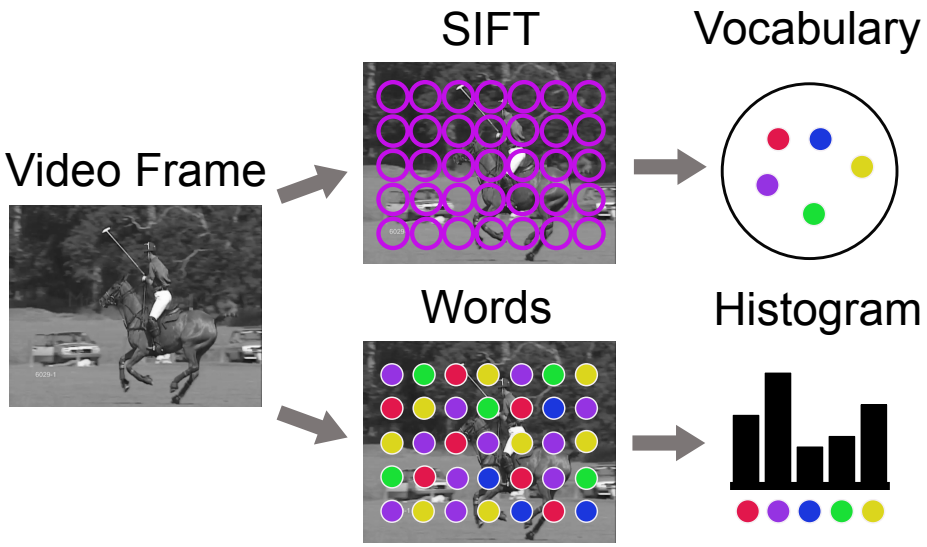


Figure 5: Illustration for building SIFT histogram features.

Table 5: Classification performance for UCF sports action video dataset.

| Models | Prediction Accuracy |
|---|---|
| Logistic Regression | 0.70 |
| Sparse Logistic Regression | 0.70 |
| Bilinear Logistic Regression | 0.73 |
| Sparse Bilinear Logistic Regression | **0.77** |

bilinear logistic regression. In overall, sparse bilinear logistic regression achieves the best classification performance.

## 9 Conclusions

We proposed sparse bilinear logistic regression, and developed an efficient numerical algorithm using the block coordinate proximal descent method. Theoretical analysis revealed its global convergence as well as convergence rate. We demonstrated its generalization performance on several real-world applications.

## References


[1] D. W. Hosmer and S. Lemeshow, *Applied logistic regression*, Probability and Statistics. Wiley, 2nd edition, 2000.

[2] C.M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2007.

[3] Y. Tsuruoka, J. McNaught, J. Tsujii, and S. Ananiadou, "Learning string similarity measures for gene/protein name dictionary look-up using logistic regression," *Bioinformatics*, vol. 23, no. 20, pp. 2768–74, 2007.

[4] J.G. Liao and K.V. Chin, "Logistic regression for disease classification using microarray data: model selection in a large p and small n cas," *Bioinformatics*, vol. 23, no. 15, pp. 1945–51, 2007.

[5] L.C. Parra, C.D. Spence, A.D. Gerson, and P. Sajda, "Recipes for the linear analysis of eeg," *Neuroimage*, vol. 28(2), pp. 326–341, 2005.

[6] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. B*, vol. 58(1), pp. 267–288, 1996.

[7] A. Ng, "Feature selection, l1 vs l2 regularization, and rotational invariance," in *International Conference on Machine Learning (ICML)*. 2004, pp. 78–85, ACM Press, New York.

[8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86(11), pp. 2278–2324, 1998.

[9] D. G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision*, 1999.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[11] J. Vidal, "Real-time detection of brain events in EEG," *IEEE Proceedings*, vol. 65(5), pp. 633–641, 1977.

[12] M. Dyrholm, C. Chistoforou, L. C. Parra, and P. Kaelbling, "Bilinear discriminant component analysis," *Journal of Machine Learning Research*, vol. 8, pp. 1007–1021, 2007.

[13] H. Pirsiavash, D. Ramanan, and C. Fowlkes, "Bilinear classifiers for visual recognition," in *Neural Information Processing Systems*, 2009.

[14] J. B. Tenenbaum and W. T. Freeman, "Separating style and content with bilinear models," *Neural Comput.*, vol. 12(6), pp. 1247–1283, 2000.

[15] R. A. Harshman, "Foundations of the PARAFAC procedure: models and conditions for an" explanatory" multi-modal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16(1), pp. 1–84, 1970.

[16] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaing Sciences*, vol. 2(1), pp. 183–202, 2009.

[17] S. Łojasiewicz, "Sur la géométrie semi-et sous-analytique," *Ann. Inst. Fourier (Grenoble)*, vol. 43, no. 5, pp. 1575–1595, 1993.

[18] K. Kurdyka, "On gradients of functions definable in o-minimal structures," in *Annales de l'institut Fourier*. Chartres: L'Institut, 1950-, 1998, vol. 48, pp. 769–784.

[19] J. Bolte, A. Daniilidis, and A. Lewis, "The Lojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 1205–1223, 2007.

[20] R.T. Rockafellar and R.J.B. Wets, *Variational analysis*, vol. 317, Springer Verlag, 1998.

[21] J. Bochnak, M. Coste, and M.F. Roy, *Real algebraic geometry*, vol. 36, Springer Verlag, 1998.

[22] Y. Xu and W. Yin, "A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion," *To appear in SIAM Journal on Imaging Science*, 2013.

[23] H. Attouch and J. Bolte, "On the convergence of the proximal algorithm for nonsmooth functions involving analytic features," *Math. Programming*, vol. 116, pp. 5–16, 2009.

[24] B. Lou, J. M. Walz, J. V. Shi, and P. Sajda, "Learning EEG components for discriminating multi-class perceptual decisions," in *Proc. IEEE Conference on Neural Engineering*, 2011, pp. 675–678.