GRADIENT PROJECTION NEWTON ALGORITHM FOR SPARSE COLLABORATIVE LEARNING[*]

JUN SUN[†], LINGCHEN KONG[‡], AND SHENGLONG ZHOU[§]

Abstract. Exploring the relationship among multiple sets of data from one same group enables practitioners to make better decisions in medical science and engineering. In this paper, we propose a sparse collaborative learning (SCL) model, an optimization with double-sparsity constraints, to process the problem with two sets of data and a shared response variable. It is capable of dealing with the classification problems or the regression problems dependent on the discreteness of the response variable as well as exploring the relationship between two datasets simultaneously. To solve SCL, we first present some necessary and sufficient optimality conditions and then design a gradient projection Newton algorithm which has proven to converge to a unique locally optimal solution globally with at least a quadratic convergence rate. Finally, the reported numerical experiments illustrate the efficiency of the proposed method.

Key words. sparse collaborative learning, double-sparsity, stationary point, gradient projection Newton, convergence analysis, numerical experiment

AMS subject classifications. 49M05, 90C26, 90C30, 65K05

1. Introduction.

There are many scenarios where datasets from the same group can be collected from diverse sources and, because of this, they are different but interrelated [10, 23, 21, 24]. For example, a researcher studying cancer outcomes may collect gene expression data and copy number data from a group of patients. The traditional approaches to do predictions are either merging two datasets or using two datasets separately. Both ways ignore the fact that they are from different sources with different meanings (e.g., gene expression and copy number). As stated in [20], exploring the relationship between sources allows for extracting informative biomarkers and improving clinical outcome predictions. Motivated by such practical applications, in this paper, we study the following sparse collaborative learning (SCL) problem:

$$\begin{aligned}\min_{\boldsymbol{\beta}_1,\boldsymbol{\beta}_2}\ & \frac{1}{n}\left[a\cdot\ell(\boldsymbol{\beta}_1;X,\mathbf{y})+b\cdot\ell(\boldsymbol{\beta}_2;Z,\mathbf{y})+\frac{c}{2}\|X\boldsymbol{\beta}_1-Z\boldsymbol{\beta}_2\|^2\right]=:f(\boldsymbol{\beta}_1,\boldsymbol{\beta}_2)\\ \text{s.t.}\ & \|\boldsymbol{\beta}_1\|_0\le s_1,\ \|\boldsymbol{\beta}_2\|_0\le s_2,\end{aligned}\tag{1.1}$$

where $X\in\mathbb{R}^{n\times p_1}, Z\in\mathbb{R}^{n\times p_2}$ are two datasets from two different sources and $\mathbf{y}\in\mathbb{R}^n$ is the shared response, n is the sample/subject size, and p_1,p_2 represent the feature/variable sizes of two datasets. Here, $\|\boldsymbol{\beta}\|_0$ is the zero norm of $\boldsymbol{\beta}$, counting the number of its nonzero elements, $s_1\ll p_1,\ s_2\ll p_2$ are two integers representing the prior information on the upper bounds of the signal sparsity, a,b and c are positive parameters, and $\|\cdot\|$ represents the Euclidean norm. Two typical examples of ℓ will be investigated in this paper. When ℓ is the linear regression loss,

$$\ell_{lin}(\boldsymbol{\beta};X,\mathbf{y}):=\frac{1}{2}\sum_{i=1}^{n}(y_i-\langle\mathbf{x}_i,\boldsymbol{\beta}\rangle)^2,$$

SCL is called sparse collaborative regression (SCoRe [7]) usually working for the continuous response $\mathbf{y}$. Here, $\langle\mathbf{x},\mathbf{z}\rangle$ is the inner product of two vectors $\mathbf{x}$ and $\mathbf{z}$ and $\mathbf{x}_i$ is a column vector corresponding the i-th row of X. SCoRe is a combination of linear regression and canonical correlation analysis (CCA). The former makes predictions via employing two different types of datasets and the latter explores the relationship between them. Examples of employing ℓ_{lin} include CoRe [4], multi-task CoRe [27] and the models studied in [6, 8].

We note that the aforementioned models based on ℓ_{lin} aimed to process the continuous response $\mathbf{y}$. However, various real-world applications involve discrete responses, in particular for those in classification problems including the severity of the disease, whether or not to die and to name a few. Under such

[*] Submitted to the editors DATE.

Funding: This work was funded by the National Natural Science Foundation of China (12071022) and Beijing Natural Science Foundation (Z190002).

[†] Department of Applied Mathematics, Beijing Jiaotong University, Beijing 100044, China. (junsun2017@bjtu.edu.cn).

[‡] Department of Applied Mathematics, Beijing Jiaotong University, Beijing 100044, China. (lchkong@bjtu.edu.cn).

[§] Department of EEE, Imperial College London, London SW7 2AZ, UK. (slzhou2021@163.com).

circumstances, linear regression-based models are unlikely to provide accurate predictions and hence it is necessary to consider the logistic regression loss defined by,

$$\ell_{log}(\boldsymbol{\beta}; X, \mathbf{y}) := \sum_{i=1}^{n} \Big(\log\left(1 + \exp\langle \mathbf{x}_i, \boldsymbol{\beta}\rangle\right) - y_i\langle \mathbf{x}_i, \boldsymbol{\beta}\rangle \Big).$$

SCL with such a loss is called the sparse logistic collaborative regression (SLCoRe), which can be used to deal with datasets with discrete response $\mathbf{y}$. SLCoRe is a combination of logistic regression and CCA, aiming at classifying the samples in each of the two datasets while exploring the relationship between them. It is well-known that discrete responses are frequently involved in classification problems, while most of the existing classification methods including support vector machines [12, 25] and logistic regression [11, 17, 22] only target one dataset. Very little work makes predictions for multiple sets of data and explores the relationship among them at the same time.

However, to accurately characterize the sparsity, it is suggested to impose the sparsity constraints directly instead of using the approximations/regularizations. For example, Beck and Eldar [1] thoroughly studied a general sparsity-constrained optimization model and developed the famous iterative hard thresholding algorithm, in the meanwhile, Bahmani et al. [3] and Plan et al. [17] investigated the logistic regression model with sparsity constraints. After which there is a vast body of work on developing optimization algorithms and understanding the properties of various sparse estimators for the sparsity constrained optimization [16, 15, 22, 26]. We emphasize that all those work aimed at addressing applications with single datasets rather than multiple datasets.

It this paper, we study two typical examples of SCL: SLCoRe with $\ell = \ell_{log}$ and SCoRe with $\ell = \ell_{lin}$. All results to be established are based on these two models. The main contributions of the paper are summarized as follows:

I) We propose a unified framework, SCL, for the problems with discrete or continuous response variables and two datasets. It can classify or predict the data in each dataset, and explore the relationship between the two datasets. The new model (1.1) exploits the sparsity constraints directly, which enables to select a sufficiently small portion of informative features in each dataset provided that s_1 and s_2 are small enough.

II) We investigate the first-order necessary and sufficient optimality conditions (see Theorem 3.1 and Theorem 3.4) for SCL as well as the existence and the uniqueness of its solution (see Theorem 3.5). One of the optimality conditions is associated with the α-stationary point seen Definition 3.2 that allows for algorithmic design conveniently.

III) We develop a gradient projection Newton algorithm (GPNA) that combines the gradient projection motivated by the α-stationary point and the Newton step to accelerate the convergence. We prove that GPNA not only converges to a unique local minimizer of the problem (1.1) globally (see Theorem 4.3) but also has a quadratic convergence rate for SLCoRe and termination within finite steps for SCoRe (see Theorem 4.5) under a mild assumption. These nice convergence properties indicate that our proposed algorithm should behave excellently in terms of high accuracy and speed, which is testified by its outstanding numerical performance.

To end this section, we present the organization of this paper. The next section describes the notation that will be employed through this paper and displays some properties of the objective function of (1.1). In Section 3, we establish the first-order necessary and sufficient optimality conditions as well as the existence and the uniqueness of the solutions to the problem (1.1). The algorithm GPNA and its convergence properties are provided in Section 4. Numerical experiments on synthetic and real data are reported in Section 5, and some concluding remarks are given in the Section 6.

2. Preliminaries. Before the main results ahead of us, we define some notation that will be employed throughout the paper. Let $[p] := \{1, 2, \cdots, p\}$. We denote the sparse set Σ_s^p in $\mathbb{R}^p$ by

$$\Sigma_s^p := \{\boldsymbol{\beta} \in \mathbb{R}^p : \|\boldsymbol{\beta}\|_0 \le s\},$$

where $s \ll p$ is an integer. For a vector $\boldsymbol{\beta}$, denote its neighborhood with a radius δ by $N(\boldsymbol{\beta}, \delta) := \{\mathbf{u} \in \mathbb{R}^p : \|\boldsymbol{\beta} - \mathbf{u}\| < \delta\}$, and its support set by $\Gamma(\boldsymbol{\beta}) := \{i \in [p] : \beta_i \neq 0\}$. The complement set of Γ is written as $\overline{\Gamma}$. For a given set T, its spanned subspace of $\mathbb{R}^p$ is denoted by $\mathbb{R}^p_T := \{\boldsymbol{\beta} \in \mathbb{R}^p : \Gamma(\boldsymbol{\beta}) \subseteq T\}$. Let $\boldsymbol{\beta}_\Gamma$ be the subvector of $\boldsymbol{\beta}$ indexed on Γ. We merge two vectors $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ as a single column vector via $(\boldsymbol{\beta}_1; \boldsymbol{\beta}_2) := (\boldsymbol{\beta}_1^\top \boldsymbol{\beta}_2^\top)^\top$. Finally, for a matrix $A \in \mathbb{R}^{n\times p}$, let $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ present its largest and smallest eigenvalue, respectively, and A_{TJ} denotes the sub-matrix containing rows indexed by T and columns indexed by J. In particular, $A_{T:} := A_{T[p]}$ and $A_{:J} := A_{[n]J}$.

To characterize the projection of $\boldsymbol{\beta}$ onto Σ_s^p, we denote $\boldsymbol{\beta}_i^{\downarrow}$ the ith largest element in magnitude of $\boldsymbol{\beta}$. Based on this, the projection $P_{\Sigma_s^p}(\boldsymbol{\beta})$ that is given by

$$\Pi_{\Sigma_s^p}(\boldsymbol{\beta}) := \operatorname*{argmin}_{\mathbf{u}\in\Sigma_s^p} \|\boldsymbol{\beta} - \mathbf{u}\| \tag{2.1}$$

can be derived as follows: If $\boldsymbol{\beta}_s^{\downarrow} = 0$ or $\boldsymbol{\beta}_s^{\downarrow} > \boldsymbol{\beta}_{s+1}^{\downarrow}$, then it is unique, i.e.,

$$(\Pi_{\Sigma_s^p}(\boldsymbol{\beta}))_i = \begin{cases} \beta_i, & |\beta_i| \geq \boldsymbol{\beta}_s^{\downarrow}, \\ 0, & |\beta_i| < \boldsymbol{\beta}_s^{\downarrow}. \end{cases}$$

Otherwise,

$$(\Pi_{\Sigma_s^p}(\boldsymbol{\beta}))_i = \begin{cases} \beta_i, & |\beta_i| > \boldsymbol{\beta}_s^{\downarrow}, \\ \beta_i \ \text{ or } \ 0, & |\beta_i| = \boldsymbol{\beta}_s^{\downarrow}, \\ 0, & |\beta_i| < \boldsymbol{\beta}_s^{\downarrow}. \end{cases}$$

Below are some concepts that will be used in this paper.

DEFINITION 2.1 (s-regularity [1]). *A Matrix $A \in \mathbb{R}^{n\times p}$ is called s-regular if its any s columns are linearly independent.*

DEFINITION 2.2 (Strong smoothness [9]). *If the function f is continuously differentiable, then for any $\boldsymbol{\beta}, \mathbf{d} \in \mathbb{R}^p$, we say the function f is strongly smooth on $\mathbb{R}^p$ with a parameter $L_f > 0$ if it holds that*

$$f(\boldsymbol{\beta} + \mathbf{d}) \leq f(\boldsymbol{\beta}) + \langle \nabla f(\boldsymbol{\beta}), \mathbf{d}\rangle + (L_f/2)\|\mathbf{d}\|^2.$$

DEFINITION 2.3 (Restricted strong convexity [2, 3, 19, 26]). *If the function f is twice continuously differentiable, then for any $\boldsymbol{\beta}, \mathbf{d} \in \Sigma_r^p$ satisfying $\boldsymbol{\beta} + \mathbf{d} \in \Sigma_r^p$, we say that the function f is r-restricted strongly convex on Σ_r^p with a parameter $l_f > 0$ if it holds that*

$$f(\boldsymbol{\beta} + \mathbf{d}) \geq f(\boldsymbol{\beta}) + \langle \nabla f(\boldsymbol{\beta}), \mathbf{d}\rangle + (l_f/2)\|\mathbf{d}\|^2 \quad \text{or} \quad \langle \mathbf{d}, \nabla^2 f(\boldsymbol{\beta})\mathbf{d}\rangle \geq (l_f/2)\|\mathbf{d}\|^2.$$

If the above conditions hold for $l_f = 0$, then f is called r-restricted convex on Σ_r^p.

We now give some properties of f in (1.1), including the strong smoothness and restricted strong convexity as well as the Lipschitz continuity of its gradient and Hessian matrix.

PROPOSITION 2.4. *Let $\boldsymbol{\beta} := (\boldsymbol{\beta}_1; \boldsymbol{\beta}_2)$ and $\ell = \ell_{log}$. The objective function f in (1.1) has the following properties.*

1) *It is convex, twice continuously differentiable and strongly smooth with a parameter L_f given by*

$$L_f := \lambda_{\max}\left(\frac{1}{n}\begin{bmatrix} (a/4+c)X^\top X & -cX^\top Z \\ -cZ^\top X & (b/4+c)Z^\top Z \end{bmatrix}\right),$$

which indicates that ∇f is Lipschitz continuous with the parameter L_f for any $\boldsymbol{\beta}$ and $\boldsymbol{\beta}'$,

$$\|\nabla f(\boldsymbol{\beta}) - \nabla f(\boldsymbol{\beta}')\| \leq L_f\|\boldsymbol{\beta} - \boldsymbol{\beta}'\|. \tag{2.2}$$

2) *Its Hessian matrix $\nabla^2 f(\boldsymbol{\beta})$ takes the form of*

$$\nabla^2 f(\boldsymbol{\beta}) = \frac{1}{n}\begin{bmatrix} X^\top(aD_1 + cI)X & -cX^\top Z \\ -cZ^\top X & Z^\top(bD_2 + cI)Z \end{bmatrix},$$

where I is the identity matrix, D_1 and D_2 are two diagonal matrices with

$$(D_1)_{ii} = \frac{\exp\langle \mathbf{x}_i, \boldsymbol{\beta}_1\rangle}{(1+\exp\langle \mathbf{x}_i, \boldsymbol{\beta}_1\rangle)^2}, \ i \in [p_1], \quad (D_2)_{ii} = \frac{\exp\langle \mathbf{z}_i, \boldsymbol{\beta}_2\rangle}{(1+\exp\langle \mathbf{z}_i, \boldsymbol{\beta}_2\rangle)^2}, \ \ i \in [p_2].$$

Moreover, $\nabla^2 f(\cdot)$ is Lipschitz continuous with the constant C_f, namely,

$$\|\nabla^2 f(\boldsymbol{\beta}) - \nabla^2 f(\boldsymbol{\beta}')\| \leq C_f\|\boldsymbol{\beta} - \boldsymbol{\beta}'\|, \tag{2.3}$$

for any $\boldsymbol{\beta}$ and $\boldsymbol{\beta}'$, where

$$C_f := \frac{3\sqrt{2}}{n} \max\left\{ a \max_{i\in[n]} \|\mathbf{x}_i\|_1 \lambda_{\max}(X^\top X), b \max_{i\in[n]} \|\mathbf{z}_i\|_1 \lambda_{\max}(Z^\top Z) \right\}.$$

3) If the matrix $[X\ Z]$ is (s_1+s_2)-regular, then it is (s_1+s_2)-restricted strongly convex with a positive parameter l_f given by

$$l_f := \min_{|T|\le s_1+s_2} \lambda_{\min}\left(\frac{c}{n} \begin{bmatrix} X^\top X & -X^\top Z \\ -Z^\top X & Z^\top Z \end{bmatrix}_{TT} \right). \tag{2.4}$$

Proof. 1) It is easy to see that f is convex and twice continuously differentiable. Since $t/(1+t)^2 \le 1/4$ for any $t \ge 0$, it follows $\lambda_{\max}(\nabla^2 f(\boldsymbol{\beta})) \le L_f$ for any $\boldsymbol{\beta} \in \mathbb{R}^{p_1+p_2}$. This can show that the gradient of f is Lipschitz continuous with the parameter L_f immediately.

2) It follows from [22, Lemma A.3] that $\nabla^2 \ell(\boldsymbol{\beta}_1; X)$ and both $\nabla^2 \ell(\boldsymbol{\beta}_2; Z)$ are Lipschitz continuous with the constants

$$C_1 := (3/n) \max_{i\in[n]} \|\mathbf{x}_i\|_1 \lambda_{\max}(X^\top X), \qquad C_2 := (3/n) \max_{i\in[n]} \|\mathbf{z}_i\|_1 \lambda_{\max}(Z^\top Z).$$

Then we have

$$\begin{aligned} \|\nabla^2 f(\boldsymbol{\beta}) - \nabla^2 f(\boldsymbol{\beta}')\| &= \|a\nabla^2\ell(\boldsymbol{\beta}_1; X) + b\nabla^2\ell(\boldsymbol{\beta}_2; Z) - a\nabla^2\ell(\boldsymbol{\beta}_1'; X) - b\nabla^2\ell(\boldsymbol{\beta}_2'; Z)\| \\ &\le aC_1\|\boldsymbol{\beta}_1 - \boldsymbol{\beta}_1'\| + bC_2\|\boldsymbol{\beta}_2 - \boldsymbol{\beta}_2'\| \\ &\le \max\{aC_1, bC_2\}(\|\boldsymbol{\beta}_1 - \boldsymbol{\beta}_1'\| + \|\boldsymbol{\beta}_2 - \boldsymbol{\beta}_2'\|) \\ &\le \sqrt{2}\max\{aC_1, bC_2\}\|\boldsymbol{\beta} - \boldsymbol{\beta}'\|. \end{aligned}$$

3) If the matrix $[X\ Z]$ is s_1+s_2-regular, then so is the matrix $[X\ -Z]$. Note that

$$\nabla^2 f(\boldsymbol{\beta}) = \frac{1}{n}\begin{bmatrix} aX^\top D_1 X & 0 \\ 0 & bZ^\top D_2 Z \end{bmatrix} + \frac{c}{n}\begin{bmatrix} X^\top X & -X^\top Z \\ -Z^\top X & Z^\top Z \end{bmatrix} =: A + B.$$

Clearly, both A and B are positive semi-definite. Moreover, $B = (c/n)[X\ -Z]^\top[X\ -Z]$. Therefore, for any $\mathbf{d} := (\mathbf{d}_1; \mathbf{d}_2)$ with $\|\mathbf{d}_1\|_0 \le s_1$ and $\|\mathbf{d}_2\|_0 \le s_2$, we have

$$\langle \mathbf{d}, \nabla^2 f(\boldsymbol{\beta})\mathbf{d}\rangle = \langle \mathbf{d}, (A+B)\mathbf{d}\rangle \ge \langle \mathbf{d}, B\mathbf{d}\rangle \ge l_f\|\mathbf{d}\|^2.$$

This displays that the (s_1+s_2)-restricted strong convexity of $f(\boldsymbol{\beta})$. □

We note that the classical logistic regression which has been shown to be only strictly convex instead of being restricted strongly convex even though the assumption of the regularity of the sample matrix is imposed. However, the objective function of SLCoRe can be restricted strongly convex if the sample matrix is regular. In addition, if we only have one dataset, SLCoRe will degenerate into the classical sparse logistic regression. At this point, see the example in [22], similar results can be obtained. Similarly, for the objective function of SCoRe, we easily obtain the following results.

Proposition 2.5. *Let $\boldsymbol{\beta} := (\boldsymbol{\beta}_1; \boldsymbol{\beta}_2)$ and $\ell = \ell_{lin}$. The objective function f in (1.1) is convex, twice continuously differentiable and has Hessian matrix $\nabla^2 f(\boldsymbol{\beta})$ in the form of*

$$\nabla^2 f(\boldsymbol{\beta}) = \frac{1}{n}\begin{bmatrix} (a+c)X^\top X & -cX^\top Z \\ -cZ^\top X & (b+c)Z^\top Z \end{bmatrix} =: Q.$$

Moreover, it is strongly smooth with the parameter $L_f := \lambda_{\max}(Q)$ and thus ∇f is Lipschitz continuous with the parameter L_f. If the matrix $[X\ Z]$ is (s_1+s_2)-regular, then it is (s_1+s_2)-restricted strongly convex with a positive parameter l_f given by

$$l_f := \min_{|T|\le s_1+s_2} \lambda_{\min}(Q_{TT}). \tag{2.5}$$

3. Optimality Conditions. This section establishes the optimality conditions of SCL, which will be useful for algorithmic design. Before the main results, for notational convenience, we define

$$\begin{aligned} \boldsymbol{\beta} &:= (\boldsymbol{\beta}_1; \boldsymbol{\beta}_2), \\ \nabla_i f(\boldsymbol{\beta}) &:= \nabla_{\boldsymbol{\beta}_i} f(\boldsymbol{\beta}), \quad i = 1, 2, \\ \Sigma_i &:= \Sigma_{s_i}^{p_i}, \quad i = 1, 2, \\ \Sigma &:= \{\boldsymbol{\beta} \in \mathbb{R}^{p_1+p_2} : \boldsymbol{\beta}_1 \in \Sigma_1, \boldsymbol{\beta}_2 \in \Sigma_2\}, \\ s &:= s_1 + s_2. \end{aligned} \tag{3.1}$$

Similar rules are also applied for $\boldsymbol{\beta}_1^*$ and $\boldsymbol{\beta}_2^*$. Based on these notation, we now establish the first-order necessary and sufficient optimality conditions for the problem (1.1).

THEOREM 3.1. *Let $\boldsymbol{\beta}^*$ be a point that satisfies*

$$\begin{aligned} (\nabla_j f(\boldsymbol{\beta}^*))_i = 0, \ \ i \in \Gamma(\boldsymbol{\beta}_j^*), \quad &\text{if} \quad \|\boldsymbol{\beta}_j^*\|_0 = s_j, \\ \nabla_j f(\boldsymbol{\beta}^*) = 0, \quad &\text{if} \quad \|\boldsymbol{\beta}_j^*\|_0 < s_j, \end{aligned} \tag{3.2}$$

for $j = 1, 2$. Then $\boldsymbol{\beta}^*$ is a local minimizer of (1.1) if and only if it satisfies (3.2).

Proof. Necessity. Based on [18, Theorem 6.12], a local minimizer $\boldsymbol{\beta}^*$ of the problem (1.1) must satisfy that $-\nabla f(\boldsymbol{\beta}^*) \in \mathcal{N}_\Sigma(\boldsymbol{\beta}^*) = \mathcal{N}_{\Sigma_1}(\boldsymbol{\beta}_1^*) \times \mathcal{N}_{\Sigma_2}(\boldsymbol{\beta}_2^*)$, where $\mathcal{N}_\Sigma(\boldsymbol{\beta}^*)$ is the normal cone of Σ at $\boldsymbol{\beta}^*$ and the equality is by [18, Theorem 6.41]. Then the explicit expression (see [15, Table 1]) of the normal cone $\mathcal{N}_{\Sigma_j}(\boldsymbol{\beta}_j^*)$ enables to derive (3.2) immediately.

Sufficiency. Let $\boldsymbol{\beta}^*$ satisfy (3.2). The convexity of f leads to

$$f(\boldsymbol{\beta}) \geq f(\boldsymbol{\beta}^*) + \langle \nabla_1 f(\boldsymbol{\beta}^*), \boldsymbol{\beta}_1 - \boldsymbol{\beta}_1^* \rangle + \langle \nabla_2 f(\boldsymbol{\beta}^*), \boldsymbol{\beta}_2 - \boldsymbol{\beta}_2^* \rangle. \tag{3.3}$$

If there is a $\delta > 0$ such that, for any $\boldsymbol{\beta} \in \Sigma \cap N(\boldsymbol{\beta}^*, \delta)$,

$$\langle \nabla_1 f(\boldsymbol{\beta}^*), \boldsymbol{\beta}_1 - \boldsymbol{\beta}_1^* \rangle = \langle \nabla_2 f(\boldsymbol{\beta}^*), \boldsymbol{\beta}_2 - \boldsymbol{\beta}_2^* \rangle = 0, \tag{3.4}$$

then the conclusion can be made immediately. Therefore, we next to show (3.4). In fact, by (3.2), we note that $\nabla_j f(\boldsymbol{\beta}^*) = 0$ if $\|\boldsymbol{\beta}_j^*\|_0 < s_j$, which indicates it suffices to consider the worst case of $\|\boldsymbol{\beta}_j^*\|_0 = s_j, j = 1, 2$. Under such a case, we define

$$\delta := \min_{j=1,2} \min_{i \in \Gamma(\boldsymbol{\beta}_j^*)} |(\boldsymbol{\beta}_j^*)_i|.$$

Then for any $\boldsymbol{\beta} \in N(\boldsymbol{\beta}^*, \delta) \cap \Sigma$, we have

$$\begin{aligned} |(\boldsymbol{\beta}_j)_i| &= |(\boldsymbol{\beta}_j)_i^* - (\boldsymbol{\beta}_j)_i^* + (\boldsymbol{\beta}_j)_i| \\ &\geq |(\boldsymbol{\beta}_j)_i^*| - |(\boldsymbol{\beta}_j)_i^* - (\boldsymbol{\beta}_j)_i| \\ &\geq |(\boldsymbol{\beta}_j)_i^*| - \|\boldsymbol{\beta}_j^* - \boldsymbol{\beta}_j\| \\ &> |(\boldsymbol{\beta}_j)_i^*| - \delta \\ &\geq 0. \end{aligned}$$

This indicates that $\Gamma(\boldsymbol{\beta}_j^*) \subseteq \Gamma(\boldsymbol{\beta}_j)$, which by $\|\boldsymbol{\beta}_j\|_0 \leq s_j = \|\boldsymbol{\beta}_j^*\|_0 = |\Gamma(\boldsymbol{\beta}_j^*)|$ yields

$$\Gamma(\boldsymbol{\beta}_j^*) = \Gamma(\boldsymbol{\beta}_j), j = 1, 2, \ \forall \ \boldsymbol{\beta} \in N(\boldsymbol{\beta}^*, \delta) \cap \Sigma.$$

Using the above fact and (3.2) derive that

$$\langle \nabla_j f(\boldsymbol{\beta}^*), \boldsymbol{\beta}_j - \boldsymbol{\beta}_j^* \rangle = \langle (\nabla_j f(\boldsymbol{\beta}^*))_{\Gamma(\boldsymbol{\beta}_j^*)}, (\boldsymbol{\beta}_j - \boldsymbol{\beta}_j^*)_{\Gamma(\boldsymbol{\beta}_j^*)} \rangle = 0.$$

The prove is completed. ◻

Based on Theorem 3.1, however, the necessary and sufficient optimality conditions (3.2) mean that there is no useful information for the case $i \notin \Gamma(\boldsymbol{\beta}_j^*)$ when $\|\boldsymbol{\beta}_j^*\|_0 = s_j$. Therefore, we introduce the concept of the α-stationary point of the problem (1.1).

Definition 3.2. *We say that $\boldsymbol{\beta}^*$ is an α-stationary point of the problem (1.1) if there exists $\alpha > 0$ such that*

$$\boldsymbol{\beta}_1^* \in \Pi_{\Sigma_1}(\boldsymbol{\beta}_1^* - \alpha\nabla_1 f(\boldsymbol{\beta}^*)), \quad \boldsymbol{\beta}_2^* \in \Pi_{\Sigma_2}\left(\boldsymbol{\beta}_2^* - \alpha\nabla_2 f\left(\boldsymbol{\beta}^*\right)\right).$$

If there is only one variable, the definition of the α-stationary points is the same as that in [1, 15] which allows us to derive its explicit expression as follows.

Lemma 3.3. *For a given $\alpha > 0$, the point $\boldsymbol{\beta}^*$ is an α-stationary point of the problem (1.1) if and only if for $j = 1, 2$, it satisfies*

$$\begin{array}{ll} \alpha(\nabla_j f(\boldsymbol{\beta}^*))_i \begin{cases} = 0, & i \in \Gamma(\boldsymbol{\beta}_j^*), \\ \leq (\boldsymbol{\beta}^*)_s^{\downarrow}, & i \in \overline{\Gamma}(\boldsymbol{\beta}_j^*), \end{cases} & \text{if} \quad \|\boldsymbol{\beta}_j^*\|_0 = s_j, \\ \nabla_j f(\boldsymbol{\beta}^*) = 0, & \text{if} \quad \|\boldsymbol{\beta}_j^*\|_0 < s_j. \end{array} \tag{3.5}$$

Comparing the conditions (3.2) and (3.5), the latter provides more information for the case $i \in \overline{\Gamma}(\boldsymbol{\beta}_j^*)$. It can be clearly seen that the latter is a stronger condition and suffices to the former.

The following result reveals the relationships among the α-stationary point and the global/local minimizers of (1.1).

Theorem 3.4. *Let $\boldsymbol{\beta}^*$ be an α-stationary point of (1.1), then it is a local minimizer. Furthermore, if $\|\boldsymbol{\beta}_1^*\|_0 < s_1, \|\boldsymbol{\beta}_2^*\|_0 < s_2$, then it is also a global minimizer. Conversely, if $\boldsymbol{\beta}^*$ is a global minimizer of (1.1), then it is an α-stationary point with $0 < \alpha < 1/L_f$.*

Proof. Since the conditions (3.5) imply (3.2) and a point satisfying (3.2) is a local minimizer by Theorem 3.1, an α-stationary point of (1.1) is a local minimizer.

Conversely, suppose that a global minimizer $\boldsymbol{\beta}^*$ of (1.1) is not an α-stationary point with $0 < \alpha < 1/L_f$, that is, there exists $\boldsymbol{\eta}_1^* \neq \boldsymbol{\beta}_1^*$ or $\boldsymbol{\eta}_2^* \neq \boldsymbol{\beta}_2^*$ such that

$$\boldsymbol{\eta}_1^* \in \Pi_{\Sigma_1}\left(\boldsymbol{\beta}_1^* - \alpha\nabla_1 f\left(\boldsymbol{\beta}^*\right)\right) \quad \text{or} \quad \boldsymbol{\eta}_2^* \in \Pi_{\Sigma_2}\left(\boldsymbol{\beta}_2^* - \alpha\nabla_2 f\left(\boldsymbol{\beta}^*\right)\right).$$

Without loss of any generality, we assume both of the above conditions are true. Then

$$\|\boldsymbol{\eta}_j^* - \boldsymbol{\beta}_j^* + \alpha\nabla_j f(\boldsymbol{\beta}^*)\|^2 \leq \|\boldsymbol{\beta}_j^* - \boldsymbol{\beta}_j^* + \alpha\nabla_j f(\boldsymbol{\beta}^*)\|^2, \quad j = 1, 2,$$

by the definition of the projection $\Pi(\cdot)$, which implies

$$\langle \boldsymbol{\eta}_j^* - \boldsymbol{\beta}_j^*, \nabla_j f(\boldsymbol{\beta}^*)\rangle \leq -(1/2\alpha)\|\boldsymbol{\eta}_j^* - \boldsymbol{\beta}_j^*\|^2, \quad i = 1, 2.$$

Using the above conditions and the strong smoothness of f with the parameter L_f results in

$$\begin{aligned} f(\boldsymbol{\eta}^*) \leq\ & f(\boldsymbol{\beta}^*) + \langle \nabla f(\boldsymbol{\beta}^*), \boldsymbol{\eta}^* - \boldsymbol{\beta}^*\rangle + (L_f/2)\|\boldsymbol{\eta}^* - \boldsymbol{\beta}^*\|^2 \\ \leq\ & f(\boldsymbol{\beta}^*) + (L_f/2 - 1/(2\alpha))\,\|\boldsymbol{\eta}^* - \boldsymbol{\beta}^*\|^2 < f(\boldsymbol{\beta}^*), \end{aligned}$$

where the last inequality is from $0 < \alpha < 1/L_f$. The above condition contradicts with the optimality of $\boldsymbol{\beta}^*$. Therefore, $\boldsymbol{\beta}^*$ is an α-stationary point with $0 < \alpha < 1/L_f$. □

To end this section, we would like to see the existence and uniqueness of solutions to (1.1), which is revealed by the following theorem.

Theorem 3.5. *If the matrix $[X\ Z]$ is s-regular, then the the global minimizer of (1.1) exists, and the local minimizers are finitely many and each of them is unique.*

Proof. The original problem (1.1) can be written as

$$\min_{|T_1|=s_1, |T_2|=s_2} \left\{ \min_{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2} \ f(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2), \quad \text{s.t. } \boldsymbol{\beta}_1 \in \mathbb{R}^{p_1}_{T_1}, \ \boldsymbol{\beta}_2 \in \mathbb{R}^{p_2}_{T_2} \right\}. \tag{3.6}$$

If the matrix $[X\ \ Z]$ is s-regular, then f is s-restricted strongly convex on $\mathbb{R}^{p_1}_{T_1} \times \mathbb{R}^{p_2}_{T_2}$. Moreover, for any given $|T_1| = s_1$ and $|T_2| = s_2$, the inner problem of (3.6) is a convex program and has a strongly convex objective function. Therefore the inner program admits a unique global minimizer denoted by $(\boldsymbol{\beta}_1^*(T_1), \boldsymbol{\beta}_2^*(T_2))$. Note that $T_1 \subseteq [p_1]$ and $T_2 \subseteq [p_2]$. Thus there are finitely many T_1 and T_2 such that $|T_1| = s_1$ and $|T_2| = s_2$, and so are the inner programs. This indicates that $(\boldsymbol{\beta}_1^*(T_1), \boldsymbol{\beta}_2^*(T_2))$ is finitely many. To derive the global minimizer of (3.6), we only pick one $(\boldsymbol{\beta}_1^*(T_1), \boldsymbol{\beta}_2^*(T_2))$ that makes the objective function value of (3.6) minimal. Global minimizers exist.

We next show that any local minimizer $\boldsymbol{\beta}^*$ is unique. To proceed with that, denote $\delta := \min\{\delta_1, \delta_2\}$ where

$$\delta_j := \begin{cases} +\infty, & \boldsymbol{\beta}_j^* = 0, \\ \min_{i \in \Gamma(\boldsymbol{\beta}_j^*)} |(\boldsymbol{\beta}_j^*)_i|, & \boldsymbol{\beta}_j^* \neq 0, \end{cases} \qquad j = 1, 2.$$

Clearly, $\delta_1, \delta_2 > 0$ and hence $\delta > 0$. Then, similar reasoning allows us to derive (3.4) for any $\boldsymbol{\beta} \in \Sigma \cap N(\boldsymbol{\beta}^*, \delta)$. This and f being s-restricted strongly convex lead to

$$f(\boldsymbol{\beta}) \geq f(\boldsymbol{\beta}^*) + (l_f/2)\|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|^2. \tag{3.7}$$

The above condition indicates $\boldsymbol{\beta}^*$ is the unique global minimizer of the problem $\min\{f(\boldsymbol{\beta}) : \boldsymbol{\beta} \in \Sigma \cap N(\boldsymbol{\beta}^*, \delta)\}$, namely, $\boldsymbol{\beta}^*$ is the unique local minimizer of (1.1). □

4. Gradient Projection Newton Algorithm.

In this section, we propose the gradient projection Newton algorithm (GPNA) for the problem (1.1). Again, for notational simplicity, we define some notation

$$\begin{array}{llll} \mathbf{u}^k & := (\mathbf{u}_1^k; \mathbf{u}_2^k), & \boldsymbol{\beta}^k & := (\boldsymbol{\beta}_1^k; \boldsymbol{\beta}_2^k), \\ \Gamma_k & := \Gamma(\mathbf{u}^k), & H^k & := \nabla^2 f(\mathbf{u}^k), \\ \boldsymbol{\beta}^k(\alpha) & := (\boldsymbol{\beta}_1^k(\alpha); \boldsymbol{\beta}_2^k(\alpha)), & \boldsymbol{\beta}_j^k(\alpha) & \in \ \Pi_{\Sigma_j}(\boldsymbol{\beta}_j^k - \alpha \nabla_j f(\boldsymbol{\beta}^k)),\ j = 1, 2. \end{array} \tag{4.1}$$

Based on the notation in (3.1), we actually have

$$\boldsymbol{\beta}^k(\alpha) \in \Pi_\Sigma(\boldsymbol{\beta}^k - \alpha \nabla f(\boldsymbol{\beta}^k)). \tag{4.2}$$

The algorithmic framework of GPNA summarized in Algorithm 4.1 consists of two major components. The first one is based on the two projected gradient steps, which enforces two variables to satisfy the sparsity constraints. The second part adopts a Newton step to speed up the convergence. However, the Newton step is only performed when one of the following conditions is satisfied,

$$\begin{array}{ll} \text{Cond 1)} & \Gamma(\boldsymbol{\beta}_1^k) = \Gamma(\mathbf{u}_1^k), \ \ \Gamma(\boldsymbol{\beta}_2^k) = \Gamma(\mathbf{u}_2^k), \\ \text{Cond 2)} & \|\nabla_1 f(\mathbf{u}^k)\| < \epsilon, \ \ \Gamma(\boldsymbol{\beta}_2^k) = \Gamma(\mathbf{u}_2^k), \\ \text{Cond 3)} & \|\nabla_2 f(\mathbf{u}^k)\| < \epsilon, \ \ \Gamma(\boldsymbol{\beta}_1^k) = \Gamma(\mathbf{u}_1^k), \\ \text{Cond 4)} & \|\nabla_1 f(\mathbf{u}^k)\| < \epsilon, \ \ \|\nabla_2 f(\mathbf{u}^k)\| < \epsilon, \end{array} \tag{4.3}$$

where $\epsilon > 0$ is a given tolerance.

Remark 4.1. We have some comments on the halting condition and computational complexity for GPNA in Algorithm 4.1.

- One can discern that if $\boldsymbol{\beta}^{k+1} = \mathbf{u}^k$, then $\boldsymbol{\beta}^{k+1}_{\overline{\Gamma}_k} = 0$. If $\boldsymbol{\beta}^{k+1} = \mathbf{v}^k$, then the updating rule (4.5) for $\mathbf{v}^k$ indicates that

  $$\Gamma(\mathbf{v}^k) \subseteq \Gamma_k = \Gamma(\mathbf{u}^k), \tag{4.7}$$

  which also implies $\boldsymbol{\beta}^{k+1}_{\overline{\Gamma}_k} = 0$. Now suppose $\texttt{tol}_k = 0$, i.e., $(\nabla f(\boldsymbol{\beta}^{k+1}))_{\Gamma_k} = 0$. Then $\boldsymbol{\beta}^{k+1}$ satisfies (3.2) and thus is a local minimizer of (1.1). Therefore, it makes sense to terminate the algorithm when $\texttt{tol}_k < \varepsilon$.

Algorithm 4.1 GPNA: Gradient projection Newton algorithm

```
Initialize $\boldsymbol{\beta}^0$. Let $0<\sigma, 0<\epsilon, 0<\alpha_0\leq 1, 0<\gamma<1, 0<\varepsilon<\mathtt{tol}_0$ and set $k \Leftarrow 0$.
while $\mathtt{tol}_k > \varepsilon$ do
    Gradient projection: Find the smallest integer $q_k = 0, 1, \cdots$ such that
    (4.4)    $f(\boldsymbol{\beta}^k(\alpha_0\gamma^{q_k})) \leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\boldsymbol{\beta}^k(\alpha_0\gamma^{q_k}) - \boldsymbol{\beta}^k\|^2.$
    Set $\alpha_k = \alpha_0\gamma^{q_k}$, $\mathbf{u}^k = \boldsymbol{\beta}^k(\alpha_k)$ and $\boldsymbol{\beta}^{k+1} = \mathbf{u}^k$.
    Newton step: if one of the conditions in (4.3) is satisfied then
        If the following equations are solvable
        (4.5)    $H^k(\mathbf{v}^k_{\Gamma_k} - \mathbf{u}^k_{\Gamma_k}) = -(\nabla f(\mathbf{u}^k))_{\Gamma_k}, \quad \mathbf{v}^k_{\overline{\Gamma}_k} = 0,$
        and the solution $\mathbf{v}^k$ satisfies
        (4.6)    $f(\mathbf{v}^k) \leq f(\mathbf{u}^k) - (\sigma/2)\|\mathbf{v}^k - \mathbf{u}^k\|^2,$
        then set $\boldsymbol{\beta}^{k+1} = \mathbf{v}^k$.
    end
    Compute $\mathtt{tol}_k := \|(\nabla f(\boldsymbol{\beta}^{k+1}))_{\Gamma_k}\|$ and set $k := k+1$.
end
Output the solution $\boldsymbol{\beta}^k$.
```

- We note that the calculations of $\Pi_{\Sigma_1}, \Pi_{\Sigma_2}$ and the gradient ∇f dominate the computation for the gradient projection step. And these three terms are easy to calculate and their total computational complexity is about $O(n(p_1+p_2))$. For the Newton step, if the matrix $[X \;\; Z]$ is s-regular, then the inverse of $H^k_{\Gamma_k\Gamma_k}$ exists due to $|\Gamma_k| \leq s$, which means that every Newton step is well defined. Moreover, the worst-case computational complexity of deriving $\mathbf{v}^k$ is about $O(s^3+ns^2)$. Overall, the entire computational complexity of the kth iteration of Algorithm 4.1 is $O(s^3+ns^2+q_kn(p_1+p_2))$. We prove that α_k is bounded by upper and lower bounds. If we know the strong smooth parameter L_f of the objective function f, then q_k may be taken as 1 or a small positive integer.

4.1. Global convergence.

Before establishing the main convergence results, we define

$$\underline{\alpha} := \min\left\{1,\ \gamma(\sigma+L_f)^{-1}\right\}, \tag{4.8}$$

which is a positive scalar. We first need the following lemma.

Lemma 4.2. *Let $\{\boldsymbol{\beta}^k\}$ be the sequence generated by GPNA. The following statements are true.*

1) For any $0<\alpha\leq 1/(\sigma+L_f)$, it holds that

$$f(\boldsymbol{\beta}^k(\alpha)) \leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k\|^2, \tag{4.9}$$

and thus $\inf_{k\geq 0}\{\alpha_k\} \geq \underline{\alpha} > 0$.

2) $\{f(\boldsymbol{\beta}^k)\}$ is a non-increasing sequence and

$$\lim_{k\to\infty}\|\mathbf{u}^k - \boldsymbol{\beta}^k\| = \lim_{k\to\infty}\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\| = 0.$$

3) Any accumulating point of the sequence $\{\boldsymbol{\beta}^k\}$ is an α-stationary point with $0<\alpha\leq\underline{\alpha}$ of (1.1).

Proof. 1) It follows from (4.2) that $\boldsymbol{\beta}^k(\alpha) \in \Pi_\Sigma(\boldsymbol{\beta}^k - \alpha\nabla f(\boldsymbol{\beta}^k))$ and thus

$$\|\boldsymbol{\beta}^k(\alpha) - (\boldsymbol{\beta}^k - \alpha\nabla f(\boldsymbol{\beta}^k))\|^2 \leq \|\boldsymbol{\beta}^k - (\boldsymbol{\beta}^k - \alpha\nabla f(\boldsymbol{\beta}^k))\|^2,$$

which results in

$$2\alpha\langle\nabla f(\boldsymbol{\beta}^k), \boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k\rangle \leq -\|\boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k\|^2. \tag{4.10}$$

This and the strong smoothness of f with the constant L_f derive that

$$\begin{aligned} f(\boldsymbol{\beta}^k(\alpha)) &\leq f(\boldsymbol{\beta}^k) + \langle \nabla f(\boldsymbol{\beta}^k), \boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k \rangle + (L_f/2)\|\boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k\|^2 \\ &\leq f(\boldsymbol{\beta}^k) - (1/(2\alpha) - (L_f/2))\|\boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k\|^2 \\ &\leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\boldsymbol{\beta}^k(\alpha) - \boldsymbol{\beta}^k\|^2, \end{aligned}$$

where the last inequality is from $0 < \alpha \leq 1/(\sigma + L_f)$. Invoking the Armijo-type step size rule, one has $\alpha_k \geq \gamma/(\sigma + L_f)$, which by $\alpha_k \leq 1$ proves the desired assertion.

2) By (4.9) and $\mathbf{u}^k = \boldsymbol{\beta}^k(\alpha_k)$, we have

$$f(\mathbf{u}^k) \leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\mathbf{u}^k - \boldsymbol{\beta}^k\|^2. \tag{4.11}$$

By the framework of Algorithm 4.1, if $\boldsymbol{\beta}^{k+1} = \mathbf{u}^k$, then the above condition implies,

$$f(\boldsymbol{\beta}^{k+1}) \leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\|^2. \tag{4.12}$$

If $\boldsymbol{\beta}^{k+1} = \mathbf{v}^k$, then we obtain

$$\begin{aligned} f(\boldsymbol{\beta}^{k+1}) = f(\mathbf{v}^k) &\leq f(\mathbf{u}^k) - (\sigma/2)\|\boldsymbol{\beta}^{k+1} - \mathbf{u}^k\|^2 \\ &\leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\mathbf{u}^k - \boldsymbol{\beta}^k\|^2 - (\sigma/2)\|\boldsymbol{\beta}^{k+1} - \mathbf{u}^k\|^2 \\ &\leq f(\boldsymbol{\beta}^k) - (\sigma/4)\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\|^2, \end{aligned} \tag{4.13}$$

where the second and last inequalities used (4.11) and a fact $\|\mathbf{a}+\mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$ for all vectors $\mathbf{a}$ and $\mathbf{b}$. Both cases lead to

$$\begin{aligned} f(\boldsymbol{\beta}^{k+1}) &\leq f(\boldsymbol{\beta}^k) - (\sigma/4)\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\|^2, \\ f(\boldsymbol{\beta}^{k+1}) &\leq f(\boldsymbol{\beta}^k) - (\sigma/2)\|\mathbf{u}^k - \boldsymbol{\beta}^k\|^2. \end{aligned} \tag{4.14}$$

Therefore, $\{f(\boldsymbol{\beta}^k)\}$ is a non-increasing sequence, which with (4.14) and $f \geq 0$ yields

$$\begin{aligned} &\sum_{k\geq 0} \max\{(\sigma/4)\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\|^2, (\sigma/2)\|\mathbf{u}^k - \boldsymbol{\beta}^k\|^2\} \\ \leq &\sum_{k\geq 0} \left[f(\boldsymbol{\beta}^k) - f(\boldsymbol{\beta}^{k+1})\right] = f(\boldsymbol{\beta}^0) - \lim_{k\to\infty} f(\boldsymbol{\beta}^{k+1}) \leq f(\boldsymbol{\beta}^0). \end{aligned}$$

The above condition suffices to $\lim_{k\to\infty}\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\| = \lim_{k\to\infty}\|\mathbf{u}^k - \boldsymbol{\beta}^k\| = 0$.

3) Let $\boldsymbol{\beta}^*$ be any accumulating point of $\{\boldsymbol{\beta}^k\}$. Then there exists a subset M of $\{0, 1, 2, \cdots\}$ such that $\lim_{k(\in M)\to\infty} \boldsymbol{\beta}^k = \boldsymbol{\beta}^*$. This further implies $\lim_{k(\in M)\to\infty} \mathbf{u}^k = \boldsymbol{\beta}^*$ by applying 2). In addition, as stated in 1), we have $\{\alpha_k\} \subseteq [\underline{\alpha}, 1]$, which indicates that one can find a subsequence K of M and a scalar $\alpha_* \in [\underline{\alpha}, 1]$ such that $\{\alpha_k : k \in K\} \to \alpha_*$. Overall, we have

$$\lim_{k(\in K)\to\infty} \boldsymbol{\beta}^k = \lim_{k(\in K)\to\infty} \mathbf{u}^k = \boldsymbol{\beta}^*, \qquad \lim_{k(\in K)\to\infty} \alpha_k = \alpha_* \in [\underline{\alpha}, 1]. \tag{4.15}$$

Let $\boldsymbol{\eta}^k := \boldsymbol{\beta}^k - \alpha_k \nabla f(\boldsymbol{\beta}^k)$. The framework of Algorithm 4.1 implies

$$\mathbf{u}^k \in \Pi_\Sigma(\boldsymbol{\eta}^k), \qquad \lim_{k(\in K)\to\infty} \boldsymbol{\eta}^k = \boldsymbol{\beta}^* - \alpha_* \nabla f(\boldsymbol{\beta}^*) =: \boldsymbol{\eta}^*. \tag{4.16}$$

The first condition means $\mathbf{u}^k \in \Sigma$ for any $k \geq 1$. Note that Σ is closed and $\boldsymbol{\beta}^*$ is the accumulating point of $\{\mathbf{u}^k\}$ by (4.15). Therefore, $\boldsymbol{\beta}^* \in \Sigma$, which results in

$$\min_{\boldsymbol{\beta}\in\Sigma} \|\boldsymbol{\beta} - \boldsymbol{\eta}^*\| \leq \|\boldsymbol{\beta}^* - \boldsymbol{\eta}^*\|. \tag{4.17}$$

If the strict inequality holds in the above condition, then there is an $\varepsilon_0 > 0$ such that

$$\begin{aligned} \|\boldsymbol{\beta}^* - \boldsymbol{\eta}^*\| - \varepsilon_0 &= \min_{\boldsymbol{\beta}\in\Sigma} \|\boldsymbol{\beta} - \boldsymbol{\eta}^*\| \\ &\geq \min_{\boldsymbol{\beta}\in\Sigma} (\|\boldsymbol{\beta} - \boldsymbol{\eta}^k\| - \|\boldsymbol{\eta}^k - \boldsymbol{\eta}^*\|) \\ &= \|\mathbf{u}^k - \boldsymbol{\eta}^k\| - \|\boldsymbol{\eta}^k - \boldsymbol{\eta}^*\|, \end{aligned}$$

where the last equality is from (4.16). Taking the limit of both sides of the above condition along $k(\in K) \to \infty$ yields $\|\boldsymbol{\beta}^* - \boldsymbol{\eta}^*\| - \varepsilon_0 \geq \|\boldsymbol{\beta}^* - \boldsymbol{\eta}^*\|$ by (4.15) and (4.16), a contradiction with $\varepsilon_0 > 0$. Therefore, we must have the equality holds in (4.17), showing that

$$\boldsymbol{\beta}^* \in \Pi_\Sigma(\boldsymbol{\eta}^*) = \Pi_\Sigma\left(\boldsymbol{\beta}^* - \alpha_*\nabla f(\boldsymbol{\beta}^*)\right).$$

The above relation means the conditions in (3.5) hold for $\alpha = \alpha_*$, then these conditions must hold for any $0 < \alpha \leq \underline{\alpha}$ due to $\underline{\alpha} \leq \alpha_*$ from (4.15), namely,

$$\boldsymbol{\beta}^* \in \Pi_\Sigma\left(\boldsymbol{\beta}^* - \alpha\nabla f(\boldsymbol{\beta}^*)\right),$$

displaying that $\boldsymbol{\beta}^*$ is an α-stationary point of (1.1), as desired. □

The above lemma allows us to conclude that the whole sequence converges.

Theorem 4.3. *Let $\{\boldsymbol{\beta}^k\}$ be the sequence generated by GPNA. Then the whole sequence converges to a unique local minimizer of (1.1) if $[X\ Z]$ is s-regular.*

Proof. As shown in Lemma 4.2, $\{\boldsymbol{\beta}^k\} \subseteq \{\boldsymbol{\beta} : f(\boldsymbol{\beta}) \leq f(\boldsymbol{\beta}^0), \boldsymbol{\beta} \in \Sigma\}$ and Ω is a bounded set due to s-restricted strong convexity of f from the s-regularity of $[X\ Z]$. Therefore, one can find a subsequence of $\{\boldsymbol{\beta}^k\}$ that converges to the α-stationary point $\boldsymbol{\beta}^*$ with $0 < \alpha \leq \underline{\alpha}$ of (1.1). Recall that an α-stationary point $\boldsymbol{\beta}^*$ is also a local minimizer by Theorem 3.4, which by Theorem 3.5 indicates that $\boldsymbol{\beta}^*$ is unique if $[X\ Z]$ is s-regular. In other words, $\boldsymbol{\beta}^*$ is an isolated local minimizer of (1.1). Finally, it follows from $\boldsymbol{\beta}^*$ being isolated, [14, Lemma 4.10] and $\lim_{k\to\infty} \|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k\| = 0$ by Lemma 4.2 that the whole sequence converges to the unique local minimizer $\boldsymbol{\beta}^*$. □

4.2. Convergence rate. This part aims to establish the convergence rate of GPNA when the sequence falls into a local area of its limiting point. Before the main result, we claim the following facts.

Lemma 4.4. *Suppose $[X\ Z]$ is s-regular. Let $\{\boldsymbol{\beta}^k\}$ be the sequence generated by GPNA and $\boldsymbol{\beta}^*$ be its limiting point. The following results hold for sufficiently large k.*

1) The support set of $\boldsymbol{\beta}^*$ can be identified by

$$\Gamma(\boldsymbol{\beta}_j^*) \begin{cases} \subseteq (\Gamma(\boldsymbol{\beta}_j^k) \cap \Gamma(\mathbf{u}_j^k)), & \textit{if} \quad \|\boldsymbol{\beta}_j^*\|_0 < s_j, \\ \equiv \Gamma(\boldsymbol{\beta}_j^k) \equiv \Gamma(\mathbf{u}_j^k), & \textit{if} \quad \|\boldsymbol{\beta}_j^*\|_0 = s_j, \end{cases} \qquad j = 1, 2. \tag{4.18}$$

2) The Newton step is always admitted if we set $\sigma \in (0, l_f/2)$.

Proof. 1) If $\|\boldsymbol{\beta}_j^*\|_0 = s_j$, then by $\boldsymbol{\beta}_j^k \to \boldsymbol{\beta}_j^*, \mathbf{u}_j^k \to \boldsymbol{\beta}_j^*$ and $\|\boldsymbol{\beta}_j^k\|_0 \leq s_j, \|\mathbf{u}_j^k\|_0 \leq s_j$, we must have $\Gamma(\boldsymbol{\beta}_j^*) \equiv \Gamma(\boldsymbol{\beta}_j^k) \equiv \Gamma(\mathbf{u}_j^k)$ for sufficiently large k. If $\|\boldsymbol{\beta}_j^*\|_0 < s_j$, similar reasoning allows for deriving $\Gamma(\boldsymbol{\beta}_j^*) \subseteq \Gamma(\boldsymbol{\beta}_j^k)$ and $\Gamma(\boldsymbol{\beta}_j^*) \subseteq \Gamma(\mathbf{u}_j^k)$.

2) By Theorem 4.3, the limit $\boldsymbol{\beta}^*$ is a local minimizer of the problem (1.1). Therefore, it satisfies (3.2) from Theorem 3.1. We first conclude that for sufficiently large k, one of the four conditions in (4.3) must be satisfied. In fact, there are four cases for $\boldsymbol{\beta}^*$ and each case can imply one condition in (4.3) as follows.

$$\begin{array}{llcl} \text{Case 1)} & \|\boldsymbol{\beta}_1^*\|_0 = s_1,\ \|\boldsymbol{\beta}_2^*\|_0 = s_2 & \Longrightarrow & \text{Cond 1)}, \\ \text{Case 2)} & \|\boldsymbol{\beta}_1^*\|_0 < s_1,\ \|\boldsymbol{\beta}_2^*\|_0 = s_2 & \Longrightarrow & \text{Cond 2)}, \\ \text{Case 3)} & \|\boldsymbol{\beta}_1^*\|_0 = s_1,\ \|\boldsymbol{\beta}_2^*\|_0 < s_2 & \Longrightarrow & \text{Cond 3)}, \\ \text{Case 4)} & \|\boldsymbol{\beta}_1^*\|_0 < s_1,\ \|\boldsymbol{\beta}_2^*\|_0 < s_2 & \Longrightarrow & \text{Cond 4)}. \end{array} \tag{4.19}$$

We now show them one by one. The Lipschitz continuity of ∇f indicates that

$$\begin{aligned}&\max\{\|\nabla_1 f(\mathbf{u}^k)-\nabla_1 f(\boldsymbol{\beta}^*)\|, \|(\nabla f(\mathbf{u}^k))_{\Gamma_k}-(\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k}\}\\ \leq\ &\|\nabla f(\mathbf{u}^k)-\nabla f(\boldsymbol{\beta}^*)\|\leq L_f\|\mathbf{u}^k-\boldsymbol{\beta}^*\|.\end{aligned}\tag{4.20}$$

The relation of Case 1) ⇒ Cond 1) can be derived by (4.18) immediately. For Case 2), we have $\Gamma(\boldsymbol{\beta}_2^k)\equiv\Gamma(\mathbf{u}_2^k)$ by (4.18) and

$$\begin{aligned}\|\nabla_1 f(\mathbf{u}^k)\| &= \|\nabla_1 f(\mathbf{u}^k)-\nabla_1 f(\boldsymbol{\beta}^*)\| && \text{(by (3.2))}\\ &\leq L_f\|\mathbf{u}^k-\boldsymbol{\beta}^*\| && \text{(by (4.20))}\\ &\leq \epsilon. && \text{(by } \mathbf{u}^k\to\boldsymbol{\beta}^*)\end{aligned}\tag{4.21}$$

Therefore, Case 2) ⇒ Cond 2). Similarly, we can show the last two relations.

Next, since $[X\ Z]$ is s-regular, H^k is non-singular, which means that the equations (4.5) are solvable. Finally, we show the inequality (4.6) is true when $\sigma\in(0,l_f/2)$. In fact, the conditions (4.18) and (3.2) enable to derive

$$(\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k}=0,\tag{4.22}$$

for sufficiently large k. Then it follows from (4.5) that

$$\begin{aligned}\|\mathbf{v}^k-\mathbf{u}^k\| &= \|\mathbf{v}^k_{\Gamma_k}-\mathbf{u}^k_{\Gamma_k}\| && \text{(by (4.7))}\\ &= \|(H^k)^{-1}(\nabla f(\mathbf{u}^k))_{\Gamma_k}\| && \text{(by (4.5))}\\ &\leq (1/l_f)\|(\nabla f(\mathbf{u}^k))_{\Gamma_k}\| && \text{(by (2.4) or (2.5))}\\ &= (1/l_f)\|(\nabla f(\mathbf{u}^k))_{\Gamma_k}-(\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k}\| && \text{(by (4.22))}\\ &\leq (L_f/l_f)\|\mathbf{u}^k-\boldsymbol{\beta}^*\|\to 0. && \text{(by (4.20))}\end{aligned}$$

The above condition indicates that $\|\mathbf{v}^k-\mathbf{u}^k\|\to 0$, resulting in

$$o(\|\mathbf{v}^k-\mathbf{u}^k\|^2)\leq(l_f/4)\|\mathbf{v}^k-\mathbf{u}^k\|^2,\tag{4.23}$$

for sufficiently large k. Now, we have the following chain of inequalities,

$$\begin{aligned}2f(\mathbf{v}^k)-2f(\mathbf{u}^k) &= 2\langle\nabla f(\mathbf{u}^k),\mathbf{v}^k-\mathbf{u}^k\rangle+2o(\|\mathbf{v}^k-\mathbf{u}^k\|^2)\\ &+\langle\nabla^2 f(\mathbf{u}^k)(\mathbf{v}^k-\mathbf{u}^k),\mathbf{v}^k-\mathbf{u}^k\rangle && \text{(by Taylor expansion)}\\ &= 2\langle(\nabla f(\mathbf{u}^k))_{\Gamma_k},(\mathbf{v}^k-\mathbf{u}^k)_{\Gamma_k}\rangle+2o(\|\mathbf{v}^k-\mathbf{u}^k\|^2)\\ &+\langle H^k(\mathbf{v}^k-\mathbf{u}^k)_{\Gamma_k},(\mathbf{v}^k-\mathbf{u}^k)_{\Gamma_k}\rangle && \text{(by (4.7))}\\ &= -\langle H^k(\mathbf{v}^k-\mathbf{u}^k)_{\Gamma_k},(\mathbf{v}^k-\mathbf{u}^k)_{\Gamma_k}\rangle+2o(\|\mathbf{v}^k-\mathbf{u}^k\|^2) && \text{(by (4.5))}\\ &\leq -l_f\|(\mathbf{v}^k-\mathbf{u}^k)_{\Gamma_k}\|^2+o(\|\mathbf{v}^k-\mathbf{u}^k\|^2) && \text{(by (2.4) or (2.5))}\\ &= -l_f\|\mathbf{v}^k-\mathbf{u}^k\|^2+2o(\|\mathbf{v}^k-\mathbf{u}^k\|^2) && \text{(by (4.7))}\\ &\leq -(l_f/2)\|\mathbf{v}^k-\mathbf{u}^k\|^2 && \text{(by (4.23))}\\ &\leq -\sigma\|\mathbf{v}^k-\mathbf{u}^k\|^2. && \text{(by } \sigma\in(0,l_f/2))\end{aligned}$$

Overall, the Newton step is always admitted for sufficiently large k. □

Finally, we conclude that GPNA can converge quadratically for SLCoRe and terminate within finite steps for SCoRe by the following theorem.

THEOREM 4.5. *Suppose $[X\ \ Z]$ is s-regular and set $\sigma \in (0, l_f/2)$. Then the sequence generated by GPNA eventually converges to its limit quadratically for SLCoRe or within finitely many steps for SCoRe, namely, for sufficiently large k,*

$$\begin{aligned} \|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^*\| &\le \tfrac{(1+L_f)^2 C_f}{2l_f}\|\boldsymbol{\beta}^k - \boldsymbol{\beta}^*\|^2, && \text{if} \quad \ell = \ell_{log},\\ \boldsymbol{\beta}^{k+1} &= \boldsymbol{\beta}^*, && \text{if} \quad \ell = \ell_{lin}. \end{aligned}$$

Proof. We first estimate $\|\mathbf{u}^k - \boldsymbol{\beta}^*\|$. Recalling (4.2) that

$$\mathbf{u}^k = \boldsymbol{\beta}^k(\alpha_k) \in \Pi_\Sigma(\boldsymbol{\beta}^k - \alpha_k \nabla f(\boldsymbol{\beta}^k))$$

and $\Gamma_k = \Gamma(\mathbf{u}^k)$, we have

$$\mathbf{u}^k_{\Gamma_k} = \boldsymbol{\beta}^k_{\Gamma_k} - \alpha_k(\nabla f(\boldsymbol{\beta}^k))_{\Gamma_k}, \qquad \mathbf{u}^k_{\overline{\Gamma}_k} = 0.$$

This enables us to deliver that

$$\begin{aligned} \|\mathbf{u}^k - \boldsymbol{\beta}^*\| &= \|\boldsymbol{\beta}^k_{\Gamma_k} - \alpha_k(\nabla f(\boldsymbol{\beta}^k))_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k}\| && \text{(by (4.18))}\\ &= \|\boldsymbol{\beta}^k_{\Gamma_k} - \alpha_k(\nabla f(\boldsymbol{\beta}^k))_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k} - \alpha_k(\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k})\| && \text{(by (4.22))}\\ &\le \|\boldsymbol{\beta}^k_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k}\| + \alpha_k\|(\nabla f(\boldsymbol{\beta}^k))_{\Gamma_k} - (\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k})\| \\ &\le (1 + L_f)\|\boldsymbol{\beta}^k - \boldsymbol{\beta}^*\|. && \text{(by } 0 < \alpha_k \le 1 \text{ and (4.20))} \end{aligned} \tag{4.24}$$

By Lemma 4.4 2), the Newton step is always admitted for sufficiently large k. Then direct calculations lead the following chain of inequalities,

$$\begin{aligned} \|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^*\| &= \|\mathbf{v}^k - \boldsymbol{\beta}^*\| = \|\mathbf{v}^k_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k}\| && \text{(by (4.18))}\\ &= \|\mathbf{u}^k_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k} - (H^k)^{-1}(\nabla f(\mathbf{u}^k))_{\Gamma_k}\| && \text{(by (4.5))}\\ &= \|\mathbf{u}^k_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k} - (H^k)^{-1}((\nabla f(\mathbf{u}^k))_{\Gamma_k} - (\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k})\| && \text{(by (4.22))}\\ &\le (1/l_f)\|H^k(\mathbf{u}^k_{\Gamma_k} - \boldsymbol{\beta}^*_{\Gamma_k}) - ((\nabla f(\mathbf{u}^k))_{\Gamma_k} - (\nabla f(\boldsymbol{\beta}^*))_{\Gamma_k})\| && \text{(by (2.4) or (2.5))}\\ &\le (1/l_f)\|\nabla^2 f(\mathbf{u}^k)(\mathbf{u}^k - \boldsymbol{\beta}^*) - (\nabla f(\mathbf{u}^k) - \nabla f(\boldsymbol{\beta}^*))\| \\ &= (1/l_f)\|\textstyle\int_0^1 (\nabla^2 f(\mathbf{u}^* + t(\mathbf{u}^k - \boldsymbol{\beta}^*)) - \nabla^2 f(\mathbf{u}^k))(\mathbf{u}^k - \boldsymbol{\beta}^*)dt\| \\ &\le (1/l_f)\textstyle\int_0^1 \|\nabla^2 f(\mathbf{u}^* + t(\mathbf{u}^k - \boldsymbol{\beta}^*)) - \nabla^2 f(\mathbf{u}^k)\|\|\mathbf{u}^k - \boldsymbol{\beta}^*\|dt. \end{aligned}$$

Note that if $\ell = \ell_{lin}$, then $\nabla^2 f(\mathbf{u}^* + t(\mathbf{u}^k - \boldsymbol{\beta}^*)) = \nabla^2 f(\mathbf{u}^k) = Q$, then the above condition implies $\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^*\| \le 0$, namely, $\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^*$. If $\ell = \ell_{log}$, then above condition implies

$$\begin{aligned} \|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^*\| &\le (1/l_f)\textstyle\int_0^1 C_f\|\mathbf{u}^* + t(\mathbf{u}^k - \boldsymbol{\beta}^*) - \mathbf{u}^k\|\|\mathbf{u}^k - \boldsymbol{\beta}^*\|dt && \text{(by (2.3))}\\ &\le (C_f/l_f)\|\mathbf{u}^k - \boldsymbol{\beta}^*\|^2 \textstyle\int_0^1 (1 - t)dt \\ &= (C_f/(2l_f))\|\mathbf{u}^k - \boldsymbol{\beta}^*\|^2. \end{aligned}$$

which combining (4.24) can make the conclusion immediately. □

5. Numerical experiments.

This section implements GPNA to solve SCL with synthetic datasets and real datasets. All numerical experiments are conducted by running MATLAB (R2018b) on an ideapad with CPU @2.30GHz 2.40GHz and 4GB memory. Apart from the stopping criterion outlined in the algorithm, we also set the maximum number of iterations to 1000. We set $\sigma = \varepsilon = 0.0001, \epsilon = 0.001, \alpha_0 = 1$ and $\gamma = 0.5$. The initial point is chosen as $\boldsymbol{\beta}^0 = 0$.

5.1. Solving SLCoRe. In this subsection, we solve SCL with $\ell = \ell_{log}$, namely, SLCoRe. This model usually works well for the data with discrete response variables. In the sequel, we first present two testing examples, followed by the parameters' tuning for GPNA and its numerical comparisons with some benchmark methods on synthetic and real datasets.

5.1.1. Testing examples. Synthetic and real data are tested for SLCoRe.

Example 5.1 (Synthetic data). *Similar to [3], each sample $\mathbf{x}_i, i \in [n]$ in $X \in \mathbb{R}^{n\times p_1}$ is independently generated by an autoregressive process*

$$x_{i(j+1)} = \theta x_{ij} + \sqrt{1-\theta^2} c_j \quad \forall\ j \in [p_1 - 1],$$

with $x_{i1} \in \mathcal{N}(0,1), c_j \in \mathcal{N}(0,1)$ and $\theta \in [0,1)$ being the correlation parameter. Note that the larger θ is, the more correlated two columns are. Let $Z = X + 0.01 \cdot \Lambda$ with $\Lambda_{ij} \in \mathcal{N}(0,1)$. Therefore, for such an example, $p_1 = p_2 =: p$. The sparse parameters $\boldsymbol{\beta}_1 \in \mathbb{R}^p$ and $\boldsymbol{\beta}_2 \in \mathbb{R}^p$ have s_1 and s_2 nonzero entries that are drawn independently from the standard Gaussian distribution, respectively. Finally, the response $\mathbf{y} \in \{0,1\}^n$ is randomly generated from the Bernoulli distribution with

$$\mathrm{Prob}\{y_i = 0 \mid \mathbf{x}_i, \mathbf{z}_i\} = \frac{1}{2}\left[\frac{1}{1+\exp\left(-\langle \mathbf{x}_i, \boldsymbol{\beta}_1\rangle\right)} + \frac{1}{1+\exp\left(-\langle \mathbf{z}_i, \boldsymbol{\beta}_2\rangle\right)}\right].$$

Example 5.2 (Real data). *Two real datasets are taken into account. They are the alcohol dependence data with $n = 46$, $p_1 = 500$ and $p_2 = 300$ [24][1] and Diffuse large B-cell lymphoma (DLBCL) data with $n = 203$, $p_1 = 17350$ and $p_2 = 386165$ [10] [2]. All datasets are feature-wisely scaled to $[-1,1]$.*

To evaluate the performance of one method, we report the CPU time (in seconds), the classification error rate (CER) [22] and the canonical correlation value (CCV) defined by

$$\mathrm{CER} := \frac{\|\operatorname{sign}(X\boldsymbol{\beta}_1) - \mathbf{y}\|_0 + \|\operatorname{sign}(Z\boldsymbol{\beta}_2) - \mathbf{y}\|_0}{n}, \qquad \mathrm{CCV} := \frac{\|X\boldsymbol{\beta}_1 - Z\boldsymbol{\beta}_2\|}{n},$$

where $\boldsymbol{\beta} = (\boldsymbol{\beta}_1; \boldsymbol{\beta}_2)$ is the solution obtained by one method and $(\operatorname{sign}(\mathbf{x}))_i = 1$ if $x_i > 0$ and $(\operatorname{sign}(\mathbf{x}))_i = 0$ otherwise for $i \in [n]$. Note that the large CER (or the smaller CCV or the shorter CPU time) the better performance.

5.1.2. Effect of parameters. We now implement GPNA to see its performance under different choices of (a,b,c,s_1,s_2) .

(a) Effect of (a,b,c)**.** Recall that there are three parameters (a,b,c) involved in (1.1). We fix $a = 1, c = 0.01$ but vary $b \in [0.01, 10]$ to see the effect of b and fix $a = 1, b = 1$ but change $c \in [0.0001, 1]$ to see the effect of c. The average results over 100 instances for Example 5.1 are presented in Figure 1, where $n = 200, p = 2000$ and $s_1 = s_2 = 20$.

When a and c are fixed, from the three above sub-figures in Figure 1, one can observe that CER is declining steadily when $b \in [0.01, 1)$ but dramatically when $b \in [1, 10]$. However, the best choice of b for CCV and CUP time is $b = 1 = a$. Therefore, for Example 5.1 with $p_1 = p_2$ and $s_1 = s_2$, the best option to set a and b should be $a = b$.

When a and b are fixed, from the three bottom sub-figures in Figure 1, it can be clearly seen that the larger values of c, the smaller CCV and longer CPU time. One can observe that the variance of $c \in [0.0001, 0.01]$ do not influence CER significantly.

We test some other choices and find the following options for (a,b,c) that allows GPNA to render desirable overall performance:

$$a = \frac{s_1}{s_1+s_2}, \quad b = \frac{s_2}{s_1+s_2}, \quad c = \frac{1}{s_1+s_2}.$$

Therefore, in the following numerical experiments, we fix a,b,c as above choices if no additional information is provided.

[1] Available at https://github.com/cran/CVR/blob/master/data/alcohol.rda

[2] Available at http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE11318

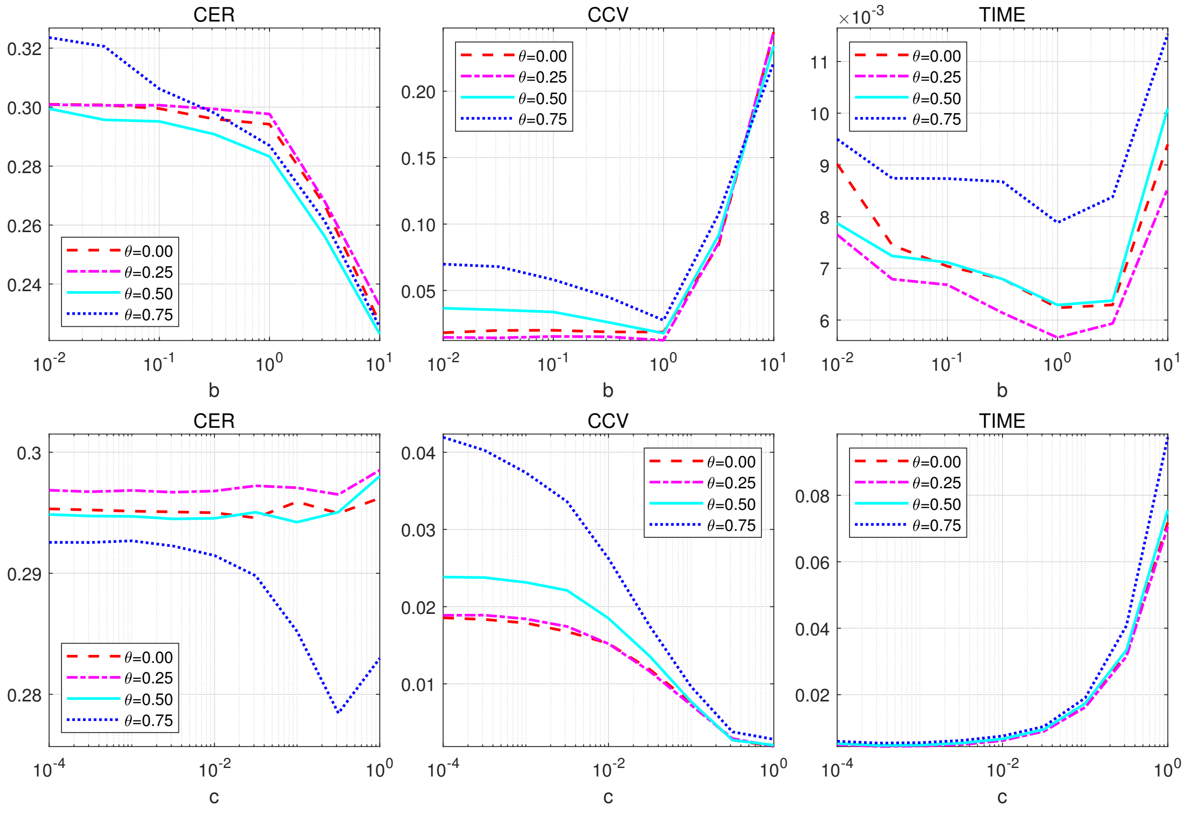

Fig. 1. *Effect of b and c for Example 5.1.*

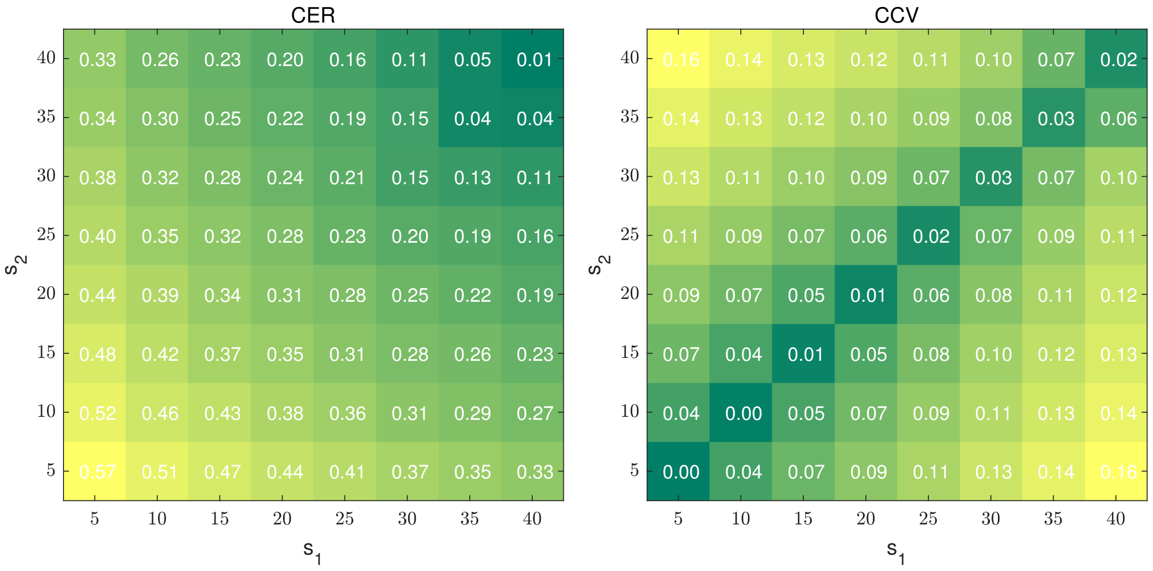

Fig. 2. *Effect of s_1 and s_2 for Example 5.1.*

(b) Effect of (s_1, s_2)**.** To see the effect of s_1 and s_2, we choose both s_1 and s_2 from $\{5, 10, \cdots, 40\}$. The average results of GPNA for Example 5.1 are shown in Figure 2 where $n = 200, p = 2000, \theta = 0.5$. The figure demonstrats that the larger s_1 or s_2 the higher values of CER, leading to better performance. Moreover, the closer between s_1 and s_2 is, the smaller CCV is.

5.1.3. Numerical comparison. To illustrate the effectiveness of our proposed model SLCoRe as well as the method GPNA, several alternative approaches are selected. They are SCoRe [7], GraSP [3] and IIHT [16]. The first one is used to solve the SCoRe, which can be used to illustrate that SLCoRe is a better model than SCoRe for the discrete response variables. GraSP and IIHT solve the sparse logistic regression that merges two datasets into a single one, which can be used to highlight the advantage of the model SLCoRe for two interrelated datasets.

(c) Comparison for Example 5.1. For simplicity, we fix $n = 1000, p = 10000$ while choose $\theta \in \{0, 0.5, 0.8\}$ and $s_1, s_2 \in \{200.300, 500\}$. For each case of (θ, s_1, s_2), we test 100 instances and report the average results of PGNA, SCoRe, IIHT and GraSP. Some comments on the reported data in Table 1 can be made.

TABLE 1
Comparison of the results for Example 5.1.

		CER			CCV			TIME		
			s_2			s_2			s_2	
s_1	Algs.	200	300	500	200	300	500	200	300	500
$\theta = 0$										
200	GPNA	0.013	0.016	0.02	0.040	0.074	0.086	00.5	00.6	00.5
	SCoRe	0.410	0.452	0.398	0.231	0.332	0.354	15.5	17.1	18.1
	IIHT	0.396	0.259	0.263	0.286	0.512	0.519	01.5	01.7	01.5
	GraSP	0.224	0.363	0.245	0.384	0.467	0.596	02.9	02.2	01.6
300	GPNA	0.012	0.000	0.000	0.084	0.032	0.062	00.4	00.5	00.5
	SCoRe	0.391	0.384	0.423	0.382	0.518	0.521	20.3	24.2	23.6
	IIHT	0.241	0.256	0.407	0.476	0.561	0.869	01.5	01.7	01.5
	GraSP	0.237	0.266	0.304	0.561	0.627	0.922	03.8	02.1	01.6
500	GPNA	0.024	0.000	0.000	0.087	0.061	0.014	00.5	00.7	00.6
	SCoRe	0.425	0.459	0.480	0.231	0.242	0.318	20.3	24.6	26.4
	IIHT	0.323	0.413	0.328	0.396	0.461	0.469	01.5	01.3	01.5
	GraSP	0.243	0.261	0.252	0.853	0.886	0.877	01.6	01.3	01.1
$\theta = 0.5$										
200	GPNA	0.014	0.021	0.023	0.050	0.086	0.087	00.4	00.4	00.5
	SCoRe	0.423	0.398	0.451	0.213	0.252	0.385	16.7	18.9	21.1
	IIHT	0.264	0.253	0.246	0.319	0.478	0.491	02.1	01.5	02.0
	GraSP	0.243	0.258	0.251	0.343	0.437	0.553	04.7	02.6	01.5
300	GPNA	0.017	0.000	0.000	0.086	0.032	0.043	00.4	00.6	00.7
	SCoRe	0.423	0.384	0.366	0.247	0.342	0.425	21.5	23.7	24.8
	IIHT	0.246	0.273	0.282	0.324	0.363	0.513	01.7	01.9	01.5
	GraSP	0.257	0.253	0.271	0.337	0.472	0.438	03.2	01.9	01.4
500	GPNA	0.018	0.000	0.000	0.089	0.071	0.020	00.5	00.7	00.7
	SCoRe	0.443	0.483	0.456	0.343	0.462	0.437	18.4	22.9	28.4
	IIHT	0.239	0.252	0.399	0.478	0.526	0.854	01.8	01.6	01.7
	GraSP	0.258	0.264	0.285	0.523	0.528	0.694	01.5	01.3	01.1
$\theta = 0.8$										
200	GPNA	0.055	0.051	0.060	0.085	0.104	0.105	00.4	00.5	00.5
	SCoRe	0.491	0.473	0.432	0.252	0.344	0.335	14.7	19.4	21.6
	IIHT	0.282	0.260	0.251	0.274	0.417	0.423	02.0	02.1	01.9
	GraSP	0.301	0.253	0.268	0.284	0.349	0.464	06.4	04.1	03.5
300	GPNA	0.052	0.000	0.000	0.105	0.059	0.093	00.4	00.5	00.5
	SCoRe	0.435	0.412	0.397	0.334	0.338	0.396	16.4	20.3	23.7
	IIHT	0.268	0.271	0.257	0.434	0.475	0.587	01.6	01.7	01.4
	GraSP	0.276	0.284	0.245	0.376	0.433	0.639	03.9	03.2	01.8
500	GPNA	0.061	0.000	0.000	0.108	0.085	0.031	00.5	00.5	00.6
	SCoRe	0.457	0.423	0.382	0.324	0.356	0.431	18.4	22.8	28.7
	IIHT	0.255	0.258	0.266	0.527	0.529	0.912	01.6	01.4	02.0
	GraSP	0.239	0.254	0.263	0.518	0.538	0.883	02.4	01.8	01.4

Regarding CER, GPNA achieves the minimum values compared with other methods regardless of the sparsity and correlation how to change. The error rate of the other three methods is more than 20% for the case of two data sets. Moreover, CERs obtained by GPNA, IIHT and GraSP are smaller than

SCoRe, which indicates that ℓ_{log} is more advantageous than ℓ_{lin} for the discrete responses.

Regarding CCV, GPNA delivers tiny values, which shows that there is a high correlation between the two datasets. Although SCoRe can also reveal the relationship between two datasets, the result is not as good as GPNA. Nevertheless, they both perform smaller CCVs than IIHT and GraSP since the latter two methods solve the model that ignores the relationship between two datasets.

Regarding TIME, it is obvious that GPGN is the fastest and the calculations take less than a second for all scenarios. By contrast, the other methods need much longer time, especially for the higher dimensional data DLBCL.

TABLE 2
Comparison of the results for Example 5.2.

			Training			Testing	
	s_1	s_2	CER	CCV	TIME(s)	CER	CCV
			AUD				
SCoRe			0.617	0.200	001.6	0.582	0.278
GPNA	20	10	0.025	0.004	000.2	0.004	0.005
	20	20	0.020	0.009	000.2	0.002	0.007
	40	20	0.018	0.008	000.2	0.002	0.014
	40	40	0.016	0.006	000.3	0.000	0.004
IIHT	20	10	0.525	0.248	001.6	0.480	0.890
	20	20	0.472	0.251	001.6	0.530	0.893
	40	20	0.453	0.249	001.5	0.460	0.885
	40	40	0.468	0.250	001.7	0.430	0.867
GraSP	20	10	0.528	0.338	001.5	0.410	0.932
	20	20	0.443	0.336	001.3	0.500	0.919
	40	20	0.493	0.327	001.6	0.420	0.924
	40	40	0.479	0.331	001.7	0.340	0.913
			DLBCL				
SCoRe			0.753	0.592	070.4	0.682	0.634
GPNA	50	50	0.054	0.036	000.3	0.024	0.029
	50	100	0.034	0.067	000.3	0.039	0.085
	100	100	0.000	0.024	000.3	0.001	0.022
	100	200	0.000	0.015	000.3	0.002	0.008
IIHT	50	50	0.471	0.796	042.5	0.464	0.732
	50	100	0.458	0.763	043.6	0.483	0.746
	100	100	0.488	0.743	046.3	0.462	0.737
	100	200	0.480	0.738	048.7	0.457	0.743
GraSP	50	50	0.456	0.854	235.4	0.472	0.861
	50	100	0.458	0.846	254.7	0.483	0.867
	100	100	0.432	0.852	228.6	0.457	0.854
	100	200	0.423	0.843	231.5	0.462	0.848

(d) Comparison for Example 5.2. This part reports the numerical comparisons of PGNA, SCoRe, IIHT and GraSP for analysing two real datasets.

We first apply our method to jointly analyze methylation and gene expression data in an alcohol dependence study [24]. SLCoRe can be used to identify the canonical variates from DNA methylation (corresponding to X) and gene expression (corresponding to Z) supervised by the phenotypical information, e.g., alcohol use disorder (AUD), which is observed as a binary indicator variable $\mathbf{y}$. In this study, genome-wide DNA methylation levels and genome-wide expression levels of genes are quantified for $n = 46$ European Australians. Similar to [13], we choose top $p_1 = 500$ CpG sites and $p_2 = 300$ genes associated with AUD.

We use a random splitting procedure to compare the four methods. At each split, 10 observations are randomly chosen as the testing data and the remaining 34 observations are the training data. The random splitting is repeated 100 times. We choose different sparsity and the average results are reported in Table 2 and show the better behaviour of GPNA since it obtains lower CER (meaning better predictions), smaller CCV and runs much faster.

We next deal with a higher dimensional real dataset DLBCL [10]. It comprises of $n = 203$ patients, each of which has $p_1 = 17350$ gene expression and $p_2 = 386165$ copy numbers. We fixate on the case where $\mathbf{y}$ is a binary variable indicating the survival or death or the cancer subtype. Again, the 203 samples are split into 153 ones as the training set and 50 ones as the testing set. The random splitting is repeated 100 times. Similar phenomenon to AUD data can be be observed for DLBCL in Table 2, showing the better performance of GPNA.

5.2. Solving SCoRe. In the subsequent numerical experiments, we focus on SCL with $\ell = \ell_{lin}$, namely, SCoRe. This model is proper for the data with continuous response variables. For such a model, we also do parameters' tuning for GPNA and get similar observations to that for SLCoRe. Therefore, we keep the same setting of parameters as previous examples for GPNA.

5.2.1. Testing examples. Again, synthetic and real data are tested for SCoRe.

Example 5.3 (Synthetic data). *The sample data X and Z as well as the sparse parameters $\boldsymbol{\beta}_1 \in \mathbb{R}^p$ and $\boldsymbol{\beta}_2 \in \mathbb{R}^p$ are generated the same as Example 5.1, while the response $\mathbf{y}$ is generated by $\mathbf{y} = (X\boldsymbol{\beta}_1 + Z\boldsymbol{\beta}_2)/2$.*

Example 5.4 (Real data). *Two real datasets are taken into consideration. They are the body mass index (BMI) of mouse data with $n = 294$, $p_1 = 163$ and $p_2 = 215$ [23][3] and DLBCL data. All datasets are feature-wisely scaled to $[-1, 1]$.*

To evaluate the performance of one method, we report the CPU time (in seconds), the mean square error (MSE) and CCV defined by

$$\text{MSE} := \frac{\|\mathbf{y} - X\boldsymbol{\beta}_1\| + \|\mathbf{y} - Z\boldsymbol{\beta}_2\|}{n}, \qquad \text{CCV} := \frac{\|X\boldsymbol{\beta}_1 - Z\boldsymbol{\beta}_2\|}{n},$$

where $\boldsymbol{\beta} = (\boldsymbol{\beta}_1; \boldsymbol{\beta}_2)$ is the solution obtained by one method.

5.2.2. Numerical comparison. Besides three aforementioned methods SCoRe, GraSP, IIHT, we also select an additional one SP [5] for comparisons. Again, GraSP, IIHT, and SP are solving the problem without consider the interrelationship between two datasets.

(e) Comparison for Example 5.3. We first compare GPNA with the other four methods for Example 5.3. For simplicity, we fix $n = 2000, p = 6000$ while choose $\theta \in \{0, 0.5\}$ and $s_1, s_2 \in \{100, 200, 500\}$. For each case of (θ, s_1, s_2), we test 100 instances and report the average results of PGNA, SCoRe, IIHT, GraSP and SP. Some comments on the reported data in Table 3 can be made.

Regarding MSE, GPNA achieves the smallest values in comparison with the other methods regardless of how the sparsity levels s_1, s_2 and correlation parameter θ change. Once again, GPNA produces relatively small CCVs, which indicates that there is a high correlation between the two datasets. By contrast, since IIHT, GraSP and SP do not take the correlation into account, their generated CCVs are higher than these by GPNA and SCoRe. It can be clearly seen that GPGN runs the fastest, such as 0.56 seconds consumed for DLBCL when $s_1 = s_2 = 50$ v.s. 83.36, 39.23, 235.5 and 13.51 seconds by the other four methods.

(f) Comparison for Example 5.4. Finally, we report results of five methods for analysing two real datasets: Mouse data and DLBCL. For mouse gene expression data, similar to [13], we choose $p_1 = 163$ single nucleotide polymorphisms (SNPs corresponding to X) and $p_2 = 215$ genes (corresponding to Z) of $n = 294$ for analysis. Again random splitting procedure is employed. At each split, 140 observations are randomly chosen as the testing data and the remaining 154 observations are the training data. The random splitting is repeated 100 times. We choose different sparsity and the average results are reported in Table 4 and display the better behaviour of GPNA since it obtains lower MSE (meaning better predictions) and smaller CCV and runs much faster. For DLBCL, results present in Table 4, where the random splitting procedure being same as Example 5.2, demonstrate better performance of GPNA.

6. Conclusions. The SCL model proposed in this paper not only fulfils the tasks of classification or regression for each dataset but also explores the relationship between two datasets. The usage of the double sparsity constraints makes it more efficient for feature selections. To solve the SCL problem, the optimality conditions have been investigated, leading to a gradient projection strategy in the algorithm. To accelerate the convergence, we employed a Newton step when the iteration met some conditions. The final developed gradient projection Newton algorithm has proven to be global and at least quadratic convergent and possessed an excellent numerical performance. We feel that the proposed method is capable of addressing some other general sparsity constrained optimization problems.

[3] Available at https://github.com/cran/CVR/blob/master/data/mouse.rda

TABLE 3
Comparison of the results for Example 5.3.

		MSE			CCV			TIME		
			s_2			s_2			s_2	
s_1	Algs.	100	200	500	100	200	500	100	200	500
					$\theta = 0$					
100	GPNA	0.083	0.097	0.171	0.015	0.081	0.169	00.3	00.4	00.4
	SCoRe	0.243	0.268	0.284	0.031	0.092	0.201	16.5	17.6	19.3
	IIHT	0.157	0.189	0.267	0.155	0.189	0.245	01.3	01.9	07.3
	GraSP	0.232	0.173	0.322	0.163	0.211	0.258	14.4	18.5	33.6
	SP	0.226	0.255	0.364	0.160	0.185	0.263	01.4	01.9	21.1
200	GPNA	0.097	0.115	0.161	0.081	0.010	0.139	00.4	00.5	00.5
	SCoRe	0.214	0.277	0.286	0.082	0.093	0.175	19.3	19.2	21.7
	IIHT	0.207	0.227	0.287	0.195	0.223	0.240	01.9	03.2	09.8
	GraSP	0.211	0.252	0.324	0.243	0.245	0.251	21.7	27.2	47.4
	SP	0.243	0.309	0.394	0.174	0.223	0.271	01.8	03.1	24.6
500	GPNA	0.165	0.148	0.182	0.163	0.142	0.063	00.4	00.4	00.6
	SCoRe	0.291	0.318	0.285	0.184	0.147	0.185	17.4	21.4	23.7
	IIHT	0.288	0.286	0.367	0.243	0.235	0.282	08.6	11.7	16.4
	GraSP	0.317	0.264	0.334	0.259	0.221	0.273	35.6	50.2	87.6
	SP	0.351	0.411	0.476	0.253	0.288	0.252	16.5	39.8	56.1
					$\theta = 0.5$					
100	GPNA	0.094	0.096	0.188	0.015	0.082	0.172	00.3	00.6	00.9
	SCoRe	0.242	0.265	0.267	0.192	0.177	0.184	16.7	19.4	22.3
	IIHT	0.172	0.196	0.285	0.163	0.182	0.241	01.8	02.5	11.5
	GraSP	0.187	0.224	0.273	0.152	0.187	0.252	18.3	24.2	32.6
	SP	0.224	0.252	0.377	0.159	0.183	0.269	01.3	01.9	27.4
200	GPNA	0.095	0.117	0.168	0.088	0.031	0.137	00.4	00.6	00.8
	SCoRe	0.212	0.224	0.281	0.179	0.145	0.174	18.8	20.5	23.1
	IIHT	0.196	0.233	0.317	0.203	0.218	0.263	02.6	03.9	11.9
	GraSP	0.248	0.236	0.339	0.179	0.253	0.256	21.7	35.3	47.8
	SP	0.289	0.322	0.368	0.214	0.217	0.282	01.8	03.5	29.7
500	GPNA	0.187	0.167	0.182	0.183	0.129	0.056	00.6	00.7	00.7
	SCoRe	0.287	0.245	0.272	0.195	0.146	0.122	20.4	21.8	23.3
	IIHT	0.275	0.315	0.346	0.242	0.264	0.144	14.4	21.7	24.3
	GraSP	0.244	0.337	0.386	0.296	0.302	0.221	42.2	51.7	70.4
	SP	0.358	0.402	0.461	0.263	0.278	0.266	21.8	32.3	57.4

Acknowledgments. The authors would like to thank the Associate Editor and anonymous referees for their helpful suggestions.

REFERENCES

[1] A.Beck and Y. Eldar, *Sparsity constrained nonlinear optimization: Optimality conditions and algorithms*, SIAM Journal on Optimization, 23 (2013), pp. 1480–1509.

[2] A. Agarwal, S. Negahban, and M. Wainwright, *Fast global convergence of gradient methods for high-dimensional statistical recovery*, The Annals of Statistics, (2012), pp. 2452–2482.

[3] S. Bahmani, B. Raj, and P. Boufounos, *Greedy sparsity-constrained optimization*, Journal of Machine Learning Research, 14 (2013), pp. 807–841.

[4] U. Brefeld, T. Gartner, T. Scheffer, and S. Wrobel, *Efficient co-regularised least squares regression*, 2006, pp. 137–144.

[5] W. Dai and O. Milenkovic, *Subspace pursuit for compressive sensing signal reconstruction*, IEEE Transactions on Information Theory, 55 (2009), pp. 2230–2249.

[6] Z. Feng, G. Hu, J. Kittler, W. Christmas, and X. Wu, *Cascaded collaborative regression for robust facial landmark detection trained using a mixture of synthetic and real images with dynamic weighting*, IEEE Transactions on Image Processing, 24 (2015), pp. 3425–3440.

[7] S. Gross and R. Tibshirani, *Collaborative regression*, Biostatistics, (2015), pp. 326–338.

[8] W. Hu, B. Cai, A. Zhang, V. Calhoun, and Y. Wang, *Deep collaborative learning with application to multimodal brain development study*, IEEE Transactions on Biomedical Engineering, 66 (2019), pp. 3346–3359.

[9] A. Jalali, C. Johnson, and P. Ravikumar, *Trading accuracy for sparsity in optimization problems with sparsity constraints*, Advances in Neural Information Processing Systems, 24 (2011), pp. 1935–1943.

Table 4
Comparison of the results for Example 5.4.

			Training			Testing	
	s_1	s_2	MSE	CCV	TIME(s)	MSE	CCV
				Mouse			
SCoRe			0.423	0.183	001.2	0.386	0.172
GPNA	20	10	0.196	0.121	000.1	0.256	0.151
	20	20	0.174	0.120	000.1	0.223	0.136
	40	20	0.141	0.103	000.1	0.184	0.126
	40	40	0.125	0.088	000.1	0.167	0.103
IIHT	20	10	0.325	0.228	000.5	0.315	0.233
	20	20	0.323	0.197	000.6	0.301	0.198
	40	20	0.319	0.159	000.6	0.305	0.227
	40	40	0.318	0.162	000.7	0.312	0.173
GraS	20	10	0.324	0.265	000.7	0.336	0.302
	20	20	0.312	0.263	000.7	0.328	0.273
	40	20	0.286	0.237	000.8	0.313	0.262
	40	40	0.294	0.258	000.9	0.327	0.235
SP	20	10	0.337	0.169	000.4	0.344	0.183
	20	20	0.335	0.158	000.4	0.342	0.129
	40	20	0.338	0.146	000.6	0.353	0.187
	40	40	0.334	0.138	000.9	0.355	0.159
				DLBCL			
SCoRe			0.533	0.315	083.4	0.546	0.307
GPNA	50	50	0.267	0.166	000.6	0.313	0.213
	50	100	0.243	0.167	000.6	0.339	0.225
	100	100	0.234	0.159	000.6	0.324	0.212
	100	200	0.232	0.156	000.7	0.306	0.219
IIHT	50	50	0.417	0.352	039.2	0.445	0.326
	50	100	0.412	0.346	042.7	0.437	0.317
	100	100	0.403	0.337	045.6	0.431	0.314
	100	200	0.413	0.347	047.4	0.434	0.321
GraSP	50	50	0.456	0.434	235.5	0.441	0.362
	50	100	0.458	0.457	254.8	0.451	0.353
	100	100	0.432	0.442	228.6	0.439	0.351
	100	200	0.423	0.443	231.5	0.442	0.362
SP	50	50	0.426	0.423	013.5	0.435	0.334
	50	100	0.428	0.437	014.8	0.443	0.341
	100	100	0.416	0.425	015.6	0.432	0.331
	100	200	0.423	0.431	018.5	0.427	0.336

[10] G. Lenz, G. Wright, N. Emre, H. Kohlhammer, S. Dave, R. Davis, S. Carty, L. Lam, A. Shaer, W. Xiao, J. Powell, A. Rosenwald, G. Ott, H. Muller, R. Gascoyne, J. Connors, E. Campo, E. Jae, J. Delabie, E. Smeland, L. Rimsza, R. Fisher, D. Weisenburger, W. Chano, and L. Staudt, *Molecular subtypes of diffuse large b-cell lymphoma arise by distinct genetic pathways*, Proceedings of the National Academy of Sciences, (2008), pp. 13520–13525.

[11] F. Liu, X. Huang, C. Gong, J. Yang, and J. Suykens, *Indefinite kernel logistic regression with concave-inexact-convex procedure*, IEEE Transactions on Neural Networks & Learning Systems, (2018), pp. 1–12.

[12] X. Liu, B. Zhao, and W. He, *Simultaneous feature selection and classification for data-adaptive kernel-penalized svm*, Mathematics, 6 (2020), p. 1846.

[13] C. Luo, J. Liu, D. K. Dey, and K. Chen, *Canonical variate regression*, Biostatistics, 17 (2017), pp. 468–483.

[14] J. Moré and D. Sorensen, *Computing a trust region step*, SIAM Journal on Scientific and Statistical Computing, 4 (1983), pp. 553–572.

[15] L. Pan, N. Xiu, and S. Zhou, *On solutions of sparsity constrained optimization*, Journal of the Operations Research Society of China, 3 (2017), pp. 421–439.

[16] L. Pan, S. Zhou, N. Xiu, and H. Qi, *A convergent iterative hard thresholding for sparsity and nonnegativity constrained optimization*, Pacific Journal of Optimization, 13 (2017), pp. 325–353.

[17] Y. Plan and R. Vershynin, *Robust 1-bit compressed sensing and sparse logistic regression: a convex programming approach*, IEEE Transactions on Information Theory, 59 (2013), pp. 482–494.

[18] R. Rockafellar and R. Wets, *Variational analysis*, Springer Science & Business Media., 2009.

[19] S. Shalev-Shwartz, N. Srebro, and T. Zhang, *Trading accuracy for sparsity in optimization problems with sparsity constraints*, SIAM Journal on Optimization, 20 (2010), pp. 2807–2832.

[20] P. Thompson, N. Martin, and M. Wright, *Imaging genomics*, Current Opinion in Neurology, 23 (2010), pp. 368–373.

[21] P. Visscher, M. Brown, M. Mccarthy, and J. Yang, *Five years of gwas discovery*, The American Journal of Human Genetics, 90 (2012), pp. 7–24.

[22] R. Wang, N. Xiu, and C. Zhang, *Greedy projected gradient-Newton method for sparse logistic regression*, IEEE Transactions on Neural Networks and Learning Systems, 31 (2020), pp. 527–538.

[23] S. Wang, N. Yehya, E. Schadt, H. Wang, T. Drake, and A. Lusis, *Genetic and genomic analysis of a fat mass trait with complex inheritance reveals marked sex specificity*, Plos Genetics, 2 (2006), pp. 148–159.

[24] H. Zhang, F. Wang, H. Xu, Y. Liu, J. Liu, H. Zhao, and J. Gelernter, *Differentially co-expressed genes in postmortem prefrontal cortex of individuals with alcohol use disorders: influence on alcohol metabolism-related pathways*, Human Genetics, 133 (2014), pp. 1383–1394.

[25] X. Zhang, Y. Wu, L. Wang, and R. Li, *Variable selection for support vector machines in moderately high dimensions*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 78 (2016), pp. 53–76.

[26] S. Zhou, N. Xiu, and H. Qi, *Global and quadratic convergence of Newton hard-thresholding pursuit*, Journal of Machine Learning Research, 22 (2021), pp. 1–45.

[27] P. Zille, V. Calhoun, and Y. Wang, *Enforcing co-expression within a brain-imaging genomics regression framework*, IEEE Transactions on Medical Imaging, 37 (2018), pp. 2561–2571.