The New England Journal of Statistics in Data Science Volume 3, 106–118 (2025)
DOI: https://doi.org/10.51387/24-NEJSDS72

# Contrastive Inverse Regression for Dimension Reduction

Sam HAWKE, Yueen MA, Hengrui LUO, and Didong LI*

**Abstract**

Supervised dimension reduction (SDR) has been a topic of growing interest in data science, as it enables the reduction of high-dimensional covariates while preserving the functional relation with certain response variables of interest. However, existing SDR methods are not suitable for analyzing datasets collected from case-control studies. In this setting, the goal is to learn and exploit the low-dimensional structure unique to or enriched by the case group, also known as the foreground group. While some unsupervised techniques such as the contrastive latent variable model and its variants have been developed for this purpose, they fail to preserve the functional relationship between the dimension-reduced covariates and the response variable. In this paper, we propose a supervised dimension reduction method called contrastive inverse regression (CIR) specifically designed for the contrastive setting. CIR introduces an optimization problem defined on the Stiefel manifold with a non-standard loss function. We prove the convergence of CIR to a local optimum using a gradient descent-based algorithm, and our numerical study empirically demonstrates the improved performance over competing methods for high-dimensional data.

KEYWORDS AND PHRASES: Case-control studies, Supervised dimension reduction, Optimization on Stiefel manifold.

## 1. INTRODUCTION

The field of data science has seen the growing importance of dimension reduction (DR) techniques as a preliminary step in processing large-scale biological datasets, such as single-cell RNA sequencing data. These techniques aid in tasks like data visualization, structural pattern discovery, and subsequent biological analyses. Within this broader context, Supervised Dimension Reduction (SDR, also known as sufficient dimension reduction) methodologies have gained significant attention [16, 37]. In the study by [61], a comparison is drawn between SDR techniques and unsupervised counterparts like Principal Component Analysis (PCA), Spherelets [40], and Spherical Rotation Component Analysis [46], emphasizing the increasing prominence and utility of SDR in contemporary data science.

Given paired observations $(x, y) \in \mathbb{R}^p \times \mathbb{R}$, where $x$ consists of $p$ covariates, and $y$ is the corresponding response or output variable, the common assumption in SDR is that

$$y = \varphi\big(V^\top x, \epsilon\big), \text{ for some function } \varphi, \tag{1.1}$$

where $V \in \mathbb{R}^{p\times d}$ with $d \ll p$ is the projection matrix from a high-dimensional to a low-dimensional space, $\epsilon$ is the measurement error independent of $x$, and $\varphi$ is an arbitrary unknown function. For example, in a single-cell RNA sequencing dataset, $x$ could be the expression of genes for a cell and $y$ could be the cell type.

*Corresponding author. Data and code available in https://github.com/myueen/contrastive-inverse-regression.

Under assumption (1.1), although the low-dimensional representation $V^\top x$ is determined by a linear transformation, the function $\varphi$ is an arbitrary unknown function. In this paper, we stick to the assumption in (1.1) to focus on linear DR methods for two reasons. First, linear methods are computationally more efficient, particularly for large $p$ and large $n$. Second, linear methods are more interpretable, which is an essential characteristic in scientific applications. For instance, in the example above, each column of $V^\top x$ is often considered as a genetic pathway [4]. Although our proposed method can be extended to nonlinear cases by the kernel method, we will leave this for future work.

Sliced Inverse Regression (SIR, [41]) is a well-established technique for supervised dimension reduction that is widely applicable in multiple scenarios due to its roots in regression analysis. It has been shown to have strong consistency results in both fixed dimensional [32] and high-dimensional [45] settings. The goal of SIR is to capture the most relevant low-dimensional linear subspace without any parametric or nonparametric model-fitting process for $\varphi$.

Moreover, SIR offers a geometric interpretation by conditioning on the sufficient statistics of the input distribution [41, 18]. SIR incorporates the idea of linear dimension reduction with statistical sufficiency. In SIR, given a pair of features $x \in \mathbb{R}^p$ and univariate response $y \in \mathbb{R}$, the goal is to find a matrix $V \in \mathbb{R}^{p\times d}$, $d < p$ such that $y$ is conditionally independent of $x$ given $V^\top x$. Although the matrix $V$ is not identifiable, the column space of $V$, denoted $\mathcal{C}(V)$, is identifiable.

Motivated by emerging modern high-dimensional [25, 44, 64] and biological datasets [28, 42], SIR evolved and ad-

mitted several generalizations, including localized SIR [68], kernel SIR [67], SIR with regularization [42], SIR for longitudinal data [34, 43], metric response values [60], and online SIR [10].

In this article, we focus on a specific type of high-dimensional biological data, where the dataset consists of two groups — a foreground group, also known as treatment group or case group, and a background group, also known as control group. The goal is to identify the low-dimensional structure, variation, and information unique to the foreground group for downstream analysis. This situation arises naturally in many scientific experiments with two subpopulations. For example, in Electronic Health Record (EHR) data, the foreground data could be health-related covariates from patients who received certain medical treatment, while the background data could be measurements from healthy patients who did not receive any treatment. In this case, the goal is to identify a unique structure in patients who received the treatment that can predict future outcomes. In a genomics context, the foreground data could be gene expression measurements from patients with a disease, and the background data could be measurements from healthy people. In this case, the goal is to predict a certain phenotype for the diseased patient for disease analysis and future therapy.

Previous contrastive models, such as the contrastive latent variable model (CLVM, [73]), contrastive principal component analysis (CPCA, [1]), probabilistic contrastive principal component analysis (PCPCA, [38]), and the contrastive Poisson latent variable model (CPLVM, [35]), have shown that using the case-control structure between foreground and background groups can greatly improve the effectiveness of dimension reduction over standard DR methods such as PCA and its variants. However, to the best of our knowledge, none of these unsupervised contrastive dimension reduction methods is directly applicable to the SDR setting.

In this work, we move from these unsupervised contrastive dimension reduction methods to a supervised contrastive dimension reduction setting. By combining the idea of contrastive loss function and the sufficient dimension reduction considered in the SIR model, we propose the Contrastive Inverse Regression (CIR) model, which exactly recovers SIR when a certain parameter is zero. The CIR model sheds light on how to explore and exploit the contrastive structures in supervised dimension reduction.

*Table 1. Categorization of DR methods by whether they are supervised or contrastive.*

| Contrastive \ Supervised | No | Yes |
|---|---|---|
| No | PCA, CCA | SIR, LDA, LASSO |
| Yes | CPCA, PCPCA | **CIR** |

Table 1 lists several popular DR methods and their properties. The table categorizes these methods as "contrastive" and "supervised", based on whether they are designed for case-control data and able to identify low-dimensional structure unique to the case group, and if they take the response variable $y$ into consideration and use $V^\top x$ to predict $y$. For example, PCA, the most well known DR method, is neither contrastive nor supervised. Similarly, canonical correlation analysis (CCA, [31]) does not utilize $y$ or the unique information of one group, which makes it neither contrastive nor supervised. Methods such as CLVM, CPCA, PCPCA, and CPLVM are contrastive but not supervised. On the other hand, classical supervised DR methods including SIR [41], linear discriminant analysis (LDA, [26]), and the least absolute shrinkage and selection operator (LASSO, [58]) are supervised but not contrastive. Our proposed method, CIR, combines both contrastive and supervised features by utilizing both the response $y$ and the case-control structure.

It is important to note that the assumption (1.1) does not limit the response variable $y$ to be continuous or categorical, and thus we do not distinguish between regression and classification. However, some methods listed in Table 1 are specifically designed for either continuous $y$ (regression, LASSO) or categorical $y$ (classification, LDA). CIR handles both scenarios with the only difference being in the choice of slices, as explained in Section 2. Furthermore, not all existing DR methods are included in this table. For example, the recently proposed linear optimal low-rank projection (LOL, [61]) is designed for the classification setting and requires the number of classes to be smaller than the reduced dimension $d$. This can be restrictive, for example, when applied to a single-cell RNA sequencing dataset, where $d$ is required to be greater than the number of cell types. In contrast, CIR does not require such data-dependent constraints on the reduced dimension $d$. Similarly, data visualization algorithms that require $d = 2$ such as the t-distributed stochastic neighbor embedding (tSNE, [59]) and uniform manifold approximation and projection (UMAP, [5]) are not listed in the table.

We now present our proposed methodology, including an algorithm for solving the associated nonconvex optimization problem on the Stiefel manifold. We also provide analysis of the convergence of the algorithm, and conduct extensive experiments to demonstrate its superior performance on high-dimensional biomedical datasets when compared to existing DR methods. All proofs are provided in the appendix, and additional experimental details are in the supplement material.

## 2. METHOD

To maintain consistency, we will use the terms "foreground group" and "background group" instead of "case-control" or "treatment-control" in the remaining sections. First, we briefly review SIR as our motivation.

**Definition 1** (Stiefel manifold)**.** $\mathrm{St}(p,d) \coloneqq \{V \in \mathbb{R}^{p\times d} : V^\top V = \mathrm{I}_d\}$ admits a smooth manifold structure equipped with a Riemannian metric, called the Stiefel manifold.

Recall that under the assumption in Equation (1.1), the centered inverse regression curve, $\mathbb{E}[x|y] - \mathbb{E}[x]$, lies exactly in the linear space spanned by columns of $\Sigma_{xx}V$, denoted by $\mathcal{C}(\Sigma_{xx}V)$, where $\Sigma_{xx}$ is the covariance matrix of $x$. This linear subspace is called the *effective dimension reduced* (e.d.r.) space [41]. As a result, the objective of SIR is to minimize the expected squared distance between $\mathbb{E}[x|y]$ and $\mathcal{C}(\Sigma_{xx}V)$:

$$\min_{V\in\mathrm{St}(p,d)} \mathbb{E}_y\big[d^2(\mathbb{E}[x|y] - \mathbb{E}[x], \mathcal{C}(\Sigma_{xx}V)\big] \tag{2.1}$$

where $d$ is the Euclidean distance.

In the contrastive setting, denote foreground data by $(x,y) \in \mathbb{R}^p \times \mathbb{R}$ and background data by $(\widetilde{x},\widetilde{y}) \in \mathbb{R}^p \times \mathbb{R}$. For convenience, we assume that $x$ and $\widetilde{x}$ are centered at the origin so that $\mathbb{E}[x] = \mathbb{E}[\widetilde{x}] = 0$.

Our goal is to find a low-dimensional representation of $x$, denoted by $V^\top x$, such that $y$ is determined by $V^\top x$ while $\widetilde{y}$ is not determined by $V^\top\widetilde{x}$. The goal of CIR is to find a low-dimensional subspace represented by $V$ such that

$$\begin{cases} y = \varphi(V^\top x, \epsilon), & \text{for some function } \varphi, \\ \widetilde{y} \neq \psi(V^\top\widetilde{x}, \widetilde{\epsilon}), & \text{for any function } \psi. \end{cases} \tag{2.2}$$

That is, the column space of $V$ captures the low-dimensional information unique to the foreground group so that we can use $V^\top x$ to predict $y$ through $\varphi$, but cannot use $V^\top\widetilde{x}$ to predict $\widetilde{y}$ through any function $\psi$. Instead of optimizing a single loss similar to SIR, CIR aims at optimizing the subspace $\mathcal{C}(\Sigma_{xx}V)$ to minimize the "contrastive loss function":

$$\begin{aligned} f(V) \coloneqq{}& \mathbb{E}_y\big[d^2(\mathbb{E}[x \mid y], \mathcal{C}(\Sigma_{xx}V)\big] \\ &- \alpha\mathbb{E}_{\widetilde{y}}\big[d^2(\mathbb{E}[\widetilde{x} \mid \widetilde{y}], \mathcal{C}(\Sigma_{\widetilde{x}\widetilde{x}}V)\big], \end{aligned} \tag{2.3}$$

where $\alpha \geq 0$, $\Sigma_{xx} = \mathrm{Cov}(X)$ and $\Sigma_{\widetilde{x}\widetilde{x}} = \mathrm{Cov}(\widetilde{X}) \in \mathbb{R}^{p\times p}$, and $d$ is the Euclidean distance. Define the following notation:

$$\begin{gathered} v_y = \mathbb{E}[x \mid y],\ v_{\widetilde{y}} = \mathbb{E}[\widetilde{x} \mid \widetilde{y}] \in \mathbb{R}^p \\ \Sigma_x = \mathrm{Cov}(v_y),\ \Sigma_{\widetilde{x}} = \mathrm{Cov}(v_{\widetilde{y}}) \in \mathbb{R}^{p\times p} \end{gathered}$$

$v_y$ (and $v_{\widetilde{y}}$) are called the centered inverse regression curves [41, 60]. The resulting loss function $f$ balances the effectiveness of dimension reduction between the foreground and background groups. We can adjust the hyperparameter $\alpha$ to express our belief in the importance of the background group. Note that the parameter $\alpha$ appears naturally in other contrastive DR methods, including CPCA and PCPCA.

*Proposition* 1. The objective function $f$ given by Equation (2.3) is simplified as

$$f(V) = -\operatorname{tr}\big(V^\top AV\big(V^\top BV\big)^{-1}\big) + \alpha\operatorname{tr}\big(V^\top\widetilde{A}V\big(V^\top\widetilde{B}V\big)^{-1}\big)$$

where $A = \Sigma_{xx}\Sigma_x\Sigma_{xx}$, $B = \Sigma_{xx}^2$, $\widetilde{A} = \Sigma_{\widetilde{x}\widetilde{x}}\Sigma_{\widetilde{x}}\Sigma_{\widetilde{x}\widetilde{x}}$, and $\widetilde{B} = \Sigma_{\widetilde{x}\widetilde{x}}^2$.

Note that $V$ is not identifiable, and is identifiable only up to a $d$-dimensional rotation. However, the contrastive loss $f$, determined by $VV^\top$, the projection matrix to the column space of $V$, is invariant under such rotations. This nonidentifiability issue is common in other DR methods, including PCA, CPCA, SIR, etc, where the convention is to refer to the column space of $V$. Therefore, this nonidentifiability is consistent with standard practices and does not impact the validity of our results.

Observe that in the case where $\alpha = 0$, CIR reduces to SIR. In this case, the problem can be reparameterized by $V^* = B^{1/2}V$ so that the columns are orthogonal, which reduces the loss function to a quadratic form, yielding a closed-form solution (as a generalized eigenproblem). In the case where $\alpha > 0$, however, we cannot perform the same trick for both $B$ and $\widetilde{B}$, so we must resort to numerical approximations. We adopt gradient-based optimization algorithms on $\mathrm{St}(p,d)$, which are based on the gradient of $f$ given by the following lemma.

*Lemma* 1. The gradient of $f$ is given by

$$\begin{aligned} &\operatorname{grad} f(V) \\ &= -2\big(AV\big(V^\top BV\big)^{-1} - BV\big(V^\top BV\big)^{-1}V^\top AV\big(V^\top BV\big)^{-1}\big) \\ &\quad + 2\alpha\big(\widetilde{A}V\big(V^\top\widetilde{B}V\big)^{-1} \\ &\quad - \widetilde{B}V\big(V^\top\widetilde{B}V\big)^{-1}V^\top\widetilde{A}V\big(V^\top\widetilde{B}V\big)^{-1}\big). \end{aligned}$$

Note that the gradient $\operatorname{grad} f$ is different from the standard gradient in Euclidean space, denoted by $Df = \frac{\partial f}{\partial V}$. The difference is that $\operatorname{grad} f$ lies in the tangent space of $\mathrm{St}(p,d)$ at $V$, while the Euclidean version may escape from the tangent space.

*Theorem* 1. If $V$ is a local minimizer of the optimization problem (2.3), then

$$AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) = BVF(V) - \alpha\widetilde{B}V\widetilde{F}(V),$$

where $E(V) = (V^\top BV)^{-1}$, $\widetilde{E}(V) = (V^\top\widetilde{B}V)^{-1}$, $F(V) = (V^\top BV)^{-1}V^\top AV(V^\top BV)^{-1}$, and $\widetilde{F}(V) = (V^\top\widetilde{B}V)^{-1}V^\top\widetilde{A}V(V^\top\widetilde{B}V)^{-1}$.

Let $G(V) = V^\top AV$ and $\widetilde{G}(V) = V^\top\widetilde{A}V$. Note, then, that the local optimality condition is equivalent to

$$\begin{aligned} &AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) \\ &\quad = BVE(V)G(V)E(V) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V). \end{aligned} \tag{2.4}$$

In Appendix G, we discuss how Equation (2.4) may lead to a gradient-free algorithm that involves solving an asymmetric algebraic Ricatti equation.

So far, we have discussed the population version, which relies on the distributions of $x$, $\widetilde{x}$, $y$, and $\widetilde{y}$ that are unknown

**Algorithm 1:** CIR.

**Input:** Foreground data $(X, Y) \in \mathbb{R}^{n\times p} \times \mathbb{R}^n$, Background data $(\widetilde{X}, \widetilde{Y}) \in \mathbb{R}^{m\times p} \times \mathbb{R}^m$, $\alpha > 0$, $d \in \mathbb{Z}_+$.
$x_i = x_i - \frac{1}{n}\sum_{i=1}^n x_i$; $\widetilde{x}_j = \widetilde{x}_j - \frac{1}{m}\sum_{j=1}^m \widetilde{x}_j$.
$\Sigma_{xx} = \frac{1}{n}\sum_{i=1}^n x_i x_i^\top$; $\Sigma_{\widetilde{x}\widetilde{x}} = \frac{1}{m}\sum_{j=1}^m \widetilde{x}_j \widetilde{x}_j^\top$.
**for** $h = 1, \ldots, H$ **do**
  Calculate slice proportions $p_h = \frac{1}{n}\sum_{i=1}^n I(y_i \in I_h)$.
  Calculate slice mean $m_h = \frac{1}{np_h}\sum_{y_i \in I_h} x_i$.
**end for**
$\Sigma_x = \sum_{h=1}^H m_h m_h^\top$.
**for** $\widetilde{h} = 1, \ldots, \widetilde{H}$ **do**
  Calculate slice proportions $p_{\widetilde{h}} = \frac{1}{m}\sum_{j=1}^m I(\widetilde{y}_j \in I_{\widetilde{h}})$.
  Calculate slice mean $m_{\widetilde{h}} = \frac{1}{mp_{\widetilde{h}}}\sum_{\widetilde{y}_j \in I_{\widetilde{h}}} \widetilde{x}_j$.
**end for**
$\Sigma_{\widetilde{x}} = \sum_{\widetilde{h}=1}^{\widetilde{H}} m_{\widetilde{h}} m_{\widetilde{h}}^\top$.
Compute $A = \Sigma_{xx}\Sigma_x\Sigma_{xx}$, $B = \Sigma_{xx}^2$,
$\widetilde{A} = \Sigma_{\widetilde{x}\widetilde{x}}\Sigma_{\widetilde{x}}\Sigma_{\widetilde{x}\widetilde{x}}$, $\widetilde{B} = \Sigma_{\widetilde{x}\widetilde{x}}^2$.
Find $V^* = \arg\min_{V\in \mathrm{St}(p,d)} f(V;\ A,\ B,\ \widetilde{A},\ \widetilde{B},\ \alpha)$ for $f$ defined in (2.3).
**Return** $V^*$.

in practice. In real applications, we observe finite samples $(x_i, y_i)_{i=1}^n$ as foreground data and $(\widetilde{x}_j, \widetilde{y}_j)_{j=1}^m$ as background data. We denote $X \in \mathbb{R}^{n\times p}$, $\widetilde{X} \in \mathbb{R}^{m\times p}$ where each row represents a sample; similarly, each entry of $Y \in \mathbb{R}^n$ and $\widetilde{Y} \in \mathbb{R}^m$ represents a response value. In this case, we can replace the expectation by the sample mean to get estimates of $\Sigma_x$, $\Sigma_{\widetilde{x}}$, $\Sigma_{xx}$, and $\Sigma_{\widetilde{x}\widetilde{x}}$ and have the corresponding plug-in estimates for $A$, $B$, $\widetilde{A}$, and $\widetilde{B}$. Throughout this paper, we assume $p < n$ so that all covariance matrices are nonsingular. The extension to $p > n$ is discussed in Section 5. After computing these matrices, the problem is reduced to a manifold optimization problem [2]. The estimates of $\Sigma_x$ and $\Sigma_{\widetilde{x}}$ deserve further discussion. As shown by [41] and [10] among others, for continuous response $y$, the observed support of response $y$ can be discretized into *slices* $I_h = (q_{h-1}, q_h]$, for $h = 1, \ldots, H$. An estimate of $\Sigma_x$ is given by $\sum_{h=1}^H m_h m_h^\top$ where $m_h = \frac{1}{np_h}\sum_{y_i\in I_h} x_i$ with $p_h = \frac{1}{n}\sum_{i=1}^n I(y_i \in I_h)$. On the other hand, if $y$ and $\widetilde{y}$ are categorical, the slices are naturally chosen as all possible values of $y$ and $\widetilde{y}$. Combining these pieces, we present our empirical version of the CIR algorithm in Algorithm 1.

It is worth noting that our optimization of $f$ as a function of $V \in \mathrm{St}(p, d)$ cannot be considered as an optimization problem in $\mathbb{R}^{p\times d}$ with orthogonality constraints $V^\top V = \mathrm{I}_d$ [23, 7]. Because the term $(V^\top BV)^{-1}$ in $f$ is not well defined unless $V$ is full rank, our loss function $f$ cannot be extended to the full Euclidean space $\mathbb{R}^{p\times d}$. We consider it as an optimization problem *intrinsically defined* on $\mathrm{St}(p, d)$ as laid out by [2]. This key property excludes some commonly used optimizers on manifolds, and we will discuss more details in the next section.

**Algorithm 2:** SGPM [50].

**Input:** $V_0 \in \mathrm{St}(p, d)$, $\eta \in [0, 1]$, $\mu, \rho_1, \epsilon, \delta \in (0, 1)$.
Set $Q_0 = 1$, $C_0 = f(V_0)$, and $k = 0$
**while** $\|\nabla_V \mathcal{L}(V_k)\| > \epsilon$ **do**
  Set $A = \nabla f(V_k)V_k^\top - V_k \nabla f(V_k)^\top$
  Set $D_{\mu,\tau_k} = (I_p + \mu\tau_k A)^{-1}$
  Let $Y(\tau) := V_k - \tau(I_p - \mu\tau A)^{-1}\nabla f(V_k)$
  Pick $\tau_k > 0$ so that $f(Y(\tau_k)) > C_k + \rho_1\tau_k Df(X_k)[\dot{Y}(0)]$
  Update $V_{k+1} = \pi(V_k - \tau D_{\mu,\tau}\nabla f(V_k))$
  Update $\mu$ and $C_{k+1}$
  Set $k = k + 1$
**end while**
$V^* = V_k$

## 3. THEORY

In this section, we discuss two concrete optimization algorithms for the last step in Algorithm 1 to find $V^*$ and show their convergence. The optimization problem outlined in Equation (2.3) does not follow the classic Li-Duan theorem for regression-based dimension reduction due to its nonconvex nature (see, e.g., [17, Prop. 8.1]). The convergence of the algorithm is discussed in detail below.

The first algorithm we consider is the scaled gradient projection method (SGPM) specifically designed for optimization on the Stiefel manifold [50]. We first define an analog to Lagrangian multiplier $\mathcal{L}(V, \Lambda) := f(V) - \frac{1}{2}\langle\Lambda, V^\top V - \mathrm{I}_d\rangle$, then the SGPM algorithm is summarized in Algorithm 2, where $\pi(X) = \arg\min_{Q\in\mathrm{St}(p,d)} ||X - Q||_F$ is the orthogonal projection to the Stiefel manifold. Note for Algorithm 2 that the update for $\mu$ and $C_{k+1}$ is intricate; see [50] for more details.

To study the convergence of Algorithm 2, we need to study the Karush-Kuhn-Tucker (KKT) conditions for CIR:

*Lemma* 2 ([50]). The KKT conditions are given by

$$D_V\mathcal{L}(V, \Lambda) = \nabla f - V\Lambda = 0$$
$$D_\Lambda\mathcal{L}(V, \Lambda) = V^\top V - \mathrm{I}_d\,.$$

Now we can state the convergence theorem of Algorithm 2:

*Theorem* 2. Let $\{V_k\}_{k=1}^\infty$ be an infinite sequence generated by Algorithm 2, then any accumulation point $V_*$ of $\{V_k\}_{k=1}^\infty$ satisfies the KKT conditions in Lemma 2, and $\lim_{k\to\infty}\|D_V\mathcal{L}(V_k)\| = 0$.

Although Algorithm 2 is guaranteed to converge, there are two drawbacks. First, the accumulation point $V_*$ is only guaranteed to satisfy the KKT conditions, but not necessarily a critical point. Second, we do not know how fast $V_k$ will converge to $V_*$. Next, we introduce an accelerated line search (ALS) algorithm as an alternative to SGPM that converges to a critical point with a known convergence rate. ALS is summarized by Algorithm 3, where $t_k^A$ is the step

**Algorithm 3:** ALS [2].

**Input:** $V_0 \in \mathrm{St}(p,d)$, retraction $R$ from $T\,\mathrm{St}(p,d)$ to $\mathrm{St}(p,d)$; scalars $\overline{\alpha} > 0$, $c, \beta, \sigma \in (0,1)$.
**for** $k = 0, 1, \ldots$ **do**
  $\eta_k = -\operatorname{grad} f(V_k)$
  Select $V_{k+1}$ so that
  $f(V_k) - f(V_{k+1}) \geq c(f(V_k) - f(R_{V_k}(t_k^A \eta_k)))$
**end for**

size, called the Armijo step size for given $\overline{\alpha}$, $\beta$, $\sigma$, $\eta_k$ and $R$ is a retraction to $\mathrm{St}(p,d)$, see [2] for more details.

Algorithm 3 can be shown to have linear convergence to critical points if the hyperparameters are chosen properly. For other choices of $\eta_k$, see Appendix E for more details. The following adaptation of Theorem 4.5.6 in [2] indicates linear convergence to stationary points.

*Theorem* 3. Let $\{V_k\}_{k=1}^{\infty}$ be an infinite sequence generated by Algorithm 3 with $\eta_k = -\operatorname{grad} f(V_k)$ converging to an accumulation point $V_*$ of $\{V_k\}_{k=1}^{\infty}$, then $V_*$ is a critical point of $f$, and $\lim_{k\to\infty} \|\operatorname{grad} f(V_k)\| = 0$.

Furthermore, assuming $V_*$ is a local minimizer of $f$ with $0 < \lambda_l := \min \mathrm{eig}(\mathrm{Hess}(f)(V_*))$ and $\lambda_u := \max \mathrm{eig}(\mathrm{Hess}(f)(V_*))$, then, for any $r \in (r_*, 1)$ where

$$r_* := 1 - \min\left(2\sigma\bar{\alpha}\lambda_l, 4\sigma(1-\sigma)\beta\frac{\lambda_l}{\lambda_u}\right),$$

there exists an integer $K \neq 0$ such that

$$f(V_{k+1}) - f(V_*) \leq \big(r + (1-r)(1-c)\big)\big(f(V_k) - f(V_*)\big),$$

for all $k \geq K$, where $\bar{\alpha}$, $\beta$, $\sigma$, $c$ are the hyperparameters in Algorithm 3.

The difference between Algorithm 2 and 3 deserves further comment. While we empirically observe that Algorithm 2 often converges faster, Algorithm 3 has theoretical properties which allow for a proof of linear convergence in terms of an upper bound on the number of iterations. In practice, for smaller datasets we suggest running Algorithm 3, while for larger datasets we recommend using Algorithm 2 for efficiency.

The computational complexity of CIR for both the SGPM-based optimization and the ALS-based optimization is compared to various competitors in the table below. Here, we assume that $1 \leq d < p < m, n$, where the background and foreground data have $m$ and $n$ samples, respectively. Specifically, we assume that $\epsilon$ denotes the stopping error such that $f(V_k) - f(V_*) \leq \epsilon$. It is noteworthy that a $p$-dimensional singular value decomposition can be achieved within $\mathcal{O}(p^3)$. We present the comparison in the table below.

*Table 2. Computational time-complexity of CIR and competitors.*

| Algorithm | Theoretical Algorithmic Complexity |
|---|---|
| CIR, SGPM-based | $\mathcal{O}((m+n)p^2)$ |
| CIR, ALS-based | $\mathcal{O}(-\log(\epsilon)p^3 + (m+n)p^2)$ |
| LDA | $\mathcal{O}(np^2)$ |
| PCA | $\mathcal{O}(np^2)$ |
| CPCA | $\mathcal{O}((m+n)p^2)$ |
| SIR | $\mathcal{O}(np^2)$ |

## 4. APPLICATION

When applying CIR, several hyperparameters must be tuned, such as the weight $\alpha$, the reduced dimension $d$, and the slices $I_h$ and $I_{\widetilde{h}}$ for estimation of $\Sigma_x$ and $\Sigma_{\widetilde{x}}$. In some cases, it may also be necessary to determine the definition of the foreground and background groups and to assign background labels $\widetilde{Y}$.

The value of $\alpha \geq 0$ can be determined by cross-validation. In particular, we suggest the following 2-step method. First, identify the rough range of $\alpha$ at the logarithmic scale. Because CIR coincides with SIR when $\alpha = 0$, running SIR initially can provide insights: better performance of SIR suggests a smaller $\alpha$, and vice versa. Once a rough range is identified, standard cross-validation can be used within this range. Our numerical experiments show that CIR is robust to the choice of $\alpha$; that is, the performance of the method changes continuously with $\alpha$. Tables supporting this observation are provided in the supplement material.

Additionally, the choice of reduced dimension $d$ may depend on the goal of the investigator. If visualization is considered important, $d = 2$ is appropriate. If the goal is prediction, the elbow point of the $d$ versus prediction error plot may suggest an optimal $d$. However, as with other DR methods, determining the optimal value of $d$ is still a topic of ongoing research [12, 13].

The definition of foreground data $X$ and $Y$ should be the data and the target variable of interest, while the choices of background data $\widetilde{X}$ and $\widetilde{Y}$ may not be as straightforward. These data are intended to represent "noise" that is "subtracted" from the foreground data. For example, in the biomedical context, if the population of interest is a group of sick patients, the background dataset may include observations of healthy individuals. In other contexts, however, it may be appropriate to use $\widetilde{X} = X$. In this case, the choice of background label $\widetilde{Y}$ may be unclear. If another outcome variable was collected, it could be used as $\widetilde{Y}$; otherwise, randomly selected values in the support of $Y$ could be used to represent "noise".

The estimates for $\Sigma_x$ and $\Sigma_{\widetilde{x}}$ are partly determined by whether $y$ and $\widetilde{y}$ are categorical or continuous. If these variables are categorical, then each value of $y$ (or $\widetilde{y}$) can be considered as a separate slice, resulting in $|\mathrm{supp}(Y)|$ (or $|\mathrm{supp}(\widetilde{Y})|$) total slices. On the other hand, if these variables are continuous, slices can be taken to represent an equally spaced partition of the range of $Y$ (or $\widetilde{Y}$), with the number of slices being tunable hyperparameters.

## 4.1 Mouse Protein Expression

The first dataset we consider was collected for the purpose of identifying proteins critical to learning in a mouse model of Down syndrome [27]. The data contain 1095 observations of expression levels of 77 different proteins, along with genotype (t=Ts65Dn, c=control), behavior (CS=context-shock, SC=shock-context), and treatment (m=memantine, s=saline). The behavior of CS corresponds to the scenario in which the mouse was first placed in a new cage and permitted to explore for a few minutes before being exposed to a brief electric shock; conversely, SC corresponds to mice immediately given an electric shock upon being placed in a new cage, and then being permitted to explore. Of the data, 543 samples contain at least one missing value. Taking into account the relatively large sample size, we consider only the 552 observations with complete data. We do not perform any normalization or any other type of preprocessing to the raw data prior to analysis.

In this example, $X \in \mathbb{R}^{552\times 77}$ represents the expression of 77 proteins of all mice without a missing value, while $y_i \in \{0, 1, \ldots, 7\}$ represents the combination of 3 binary variables: genotype, treatment, and behavior. For example, $y_i = 1$ means that the $i$-th mouse received memantine, was exposed to context-shock, and has genotype Ts65Dn. To visualize the data, we apply unsupervised DR algorithms PCA, tSNE and UMAP to $X$ and supervised DR methods LDA, LASSO and SIR to $(X, Y)$, with $d = 2$ for all algorithms. The 2-dimensional representation is given in Figure 1, where each color represents a class of mice among 8 total classes.

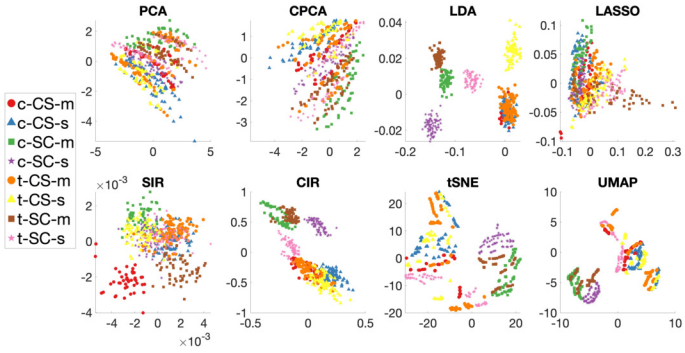


Figure 1: 2-d representation of mouse protein data. Silhouette scores: (PCA, −.20), (CPCA, −.13), (LDA, .42), (LASSO, −.17), (SIR, .03), (CIR, .29), (tSNE, −.14), (UMAP, −.00).

PCA, LASSO, SIR, tSNE, and UMAP fail to distinguish between classes, whereas LDA successfully separates 5 classes but with 3 classes (c-CS-m, c-CS-s, t-CS-m) mixed together. Now we take advantage of the background data. We let $\widetilde{X}$ be the protein expression from mice with genotype = control, which coincides with the background group used in previous studies of this application [1, 38]. We set $\widetilde{Y}$ as the binary variable representing behavior and apply CPCA to $(X, \widetilde{X})$ and CIR to $(X, Y, \widetilde{X}, \widetilde{Y})$ with $d = 2$ as well. The 2-dimensional representations with their Silhouette scores [53] are shown in Figure 1, which indicates that CIR outperforms all other competitors except LDA in terms of the Silhouette score. In particular, the three classes that were not separated in LDA are less mixed in CIR, supported by the Silhouette scores for these three classes: −0.10 for LDA and 0.23 for CIR. We provide other objective scores [11, 22] in the supplementary material.

Next, we show the classification accuracy based on $XV$, the dimension-reduced data. Here, we vary $d$ from 2 to 7 because for higher $d$, the accuracy is close to 1. The mean prediction accuracy of KNN, the best classifier for this example, over 10 replicates versus the reduced dimension $d$ is shown in Figure 2, clearly indicating that CIR outperforms all competitors especially when $d$ is small. We present the accuracy of other classifiers and their standard deviations in the supplement material.

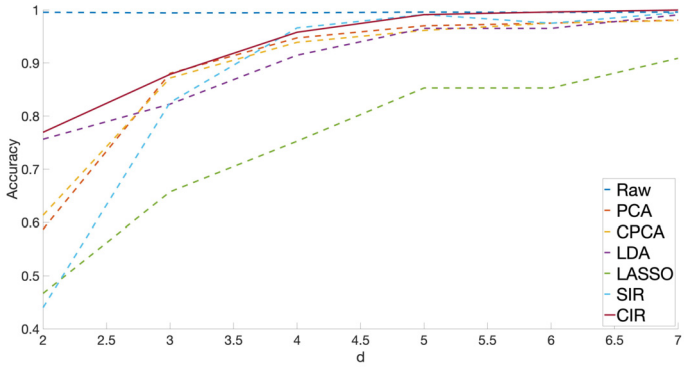


Figure 2: Classification accuracy by KNN for mouse protein data.

## 4.2 Single Cell RNA Sequencing

The second dataset we consider is from a study of single-cell RNA sequencing used to classify cells into cell types based on their transcriptional profile [3]. The data include 3500 observations of expression levels of 32838 different genes, along with cell labels as one of the 9 different cell types, namely CD8 T cell, CD4 T cell, classical monocyte, B cell, NK cell, plasmacytoid dendritic cell, non-classical monocyte, classic dendritic cell, and plasma cell. We select the top 100 most variable genes for our analysis to be consistent with previous analyses of these data [71, 1]. In this example, $X \in \mathbb{R}^{3500\times 100}$ represents the expression of 100 genes, while $y_i \in \{0, 1, \ldots, 8\}$ represents the cell type. For example, $y_i = 1$ means that the $i$-th cell is a CD4 T cell.

To visualize the data, we apply unsupervised DR algorithms PCA, tSNE, and UMAP to $X$ and supervised DR methods LDA, LASSO, and SIR to $(X, Y)$, for $d = 2$ for all algorithms. In this case, there is no obvious choice

of background data. So, we use $\widetilde{X} = X$ and randomly draw independent and identically distributed samples $\widetilde{Y} \sim$ uniform$\{0, \ldots, 8\}$ in order to apply CPCA and CIR. The 2-dimensional representations with their Silhouette scores are given in Figure 3, which indicates that CIR has the best performance.

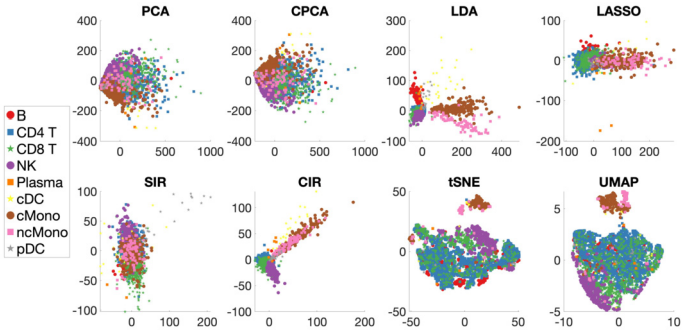


Figure 3: 2-d representation of single-cell RNA sequencing data. Silhouette scores: (PCA, −.25), (CPCA, −.25), (LDA, .11), (LASSO, −.30), (SIR, −.10), (CIR, .15), (tSNE, −.17), (UMAP, −.17).

For each $d = 2, \ldots, 10$, we compare the accuracy of a KNN classifier based on dimension-reduced data among various methods, with the raw data as the baseline. We repeat this process 10 times to reduce the impact of random split in cross-validation. The prediction accuracy versus reduced dimension $d$ is shown in Figure 4, where CIR has the best overall performance especially when $d = 2, 3$. We show the accuracy of other classifiers and their standard deviations in the supplement material.

The improved performance of CIR over SIR deserves further comment. While the background data and labels $(\widetilde{X}, \widetilde{Y})$ used in CIR do not add new information beyond what SIR used, because the background label is chosen randomly, we attribute the improved performance to CIR "denoising" the foreground data.

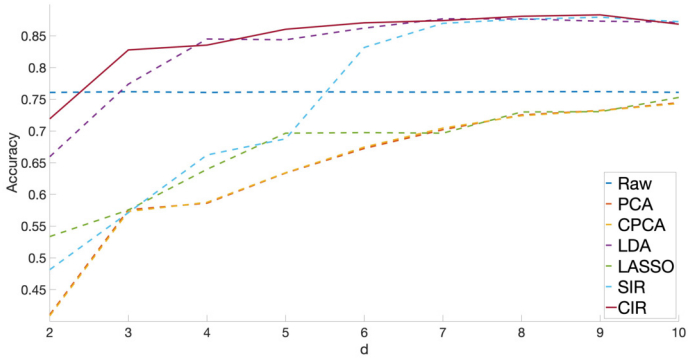


Figure 4: Classification accuracy by KNN for single-cell RNA sequencing data.

## 4.3 COVID-19 Cell States

The third dataset we consider is also a single-cell RNA sequencing dataset, with samples from 90 patients with COVID-19 and 23 healthy volunteers [56]. In total, the dataset contains 48083 cells from diseased patients and 14426 cells from healthy volunteers. We treat the cells from the patients with disease as foreground and the cells from the healthy volunteers as background. On each cell, RNA expression levels on 24727 different genes were measured. For the features, we selected the 500 genes with the largest variances in RNA expression, in accordance with prior analysis with this dataset [21].

For each cell in the dataset, its cell type is identified, which we use as the labels. As recommended in a previous analysis [21], we consider only the cells for which at least 250 observations were available. This filtering resulted in 14 distinct cell types, with 40411 observations in the foreground and 13041 in the background. As in the previous example, we have $X \in \mathbb{R}^{40411 \times 500}$ to represent the expressions of 500 genes in the cells of patients with COVID-19 and $\widetilde{X} \in \mathbb{R}^{13041 \times 500}$ to represent the gene expressions in the healthy volunteers, while $y_i, \widetilde{y}_j \in \{0, 1, \ldots, 13\}$ represents the cell type.

As with the previous two examples, we first apply DR methods to visualize the data. With $d = 2$, we apply PCA, tSNE, and UMAP to $X$, and we apply LDA, LASSO, and SIR to $(X, Y)$. We also apply CPCA to $(X, \widetilde{X})$ and CIR to $(X, Y, \widetilde{X}, \widetilde{Y})$. The 2-dimensional representations with their Silhouette scores are provided in Figure 5. Although CIR is not the best overall in terms of the Silhouette score, it outperforms its direct competitors, CPCA and SIR.

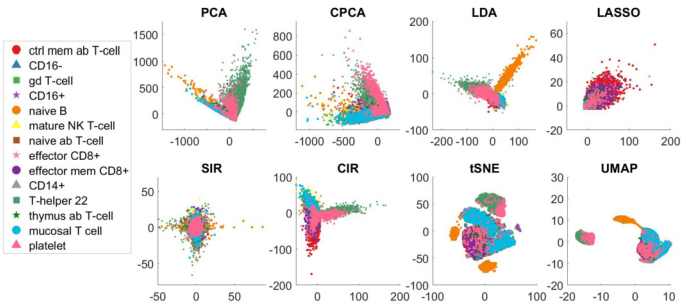


Figure 5: 2-d representation of COVID-19 data. Silhouette scores: (PCA, −.29), (CPCA, −.29), (LDA, .02), (LASSO, −.48), (SIR, −.27), (CIR, −.05), (tSNE, −.03), (UMAP, .11).

For $d = 2, \ldots, 7$, we compare the accuracy of a KNN classifier based on the dimension-reduced data among various DR methods, with the raw data as baseline. As with the previous examples, we repeat this process 10 times for each method to reduce the effect of the cross-validation random split on the results. The prediction accuracy for each reduced dimension $d$ is shown in Figure 6, where we see that

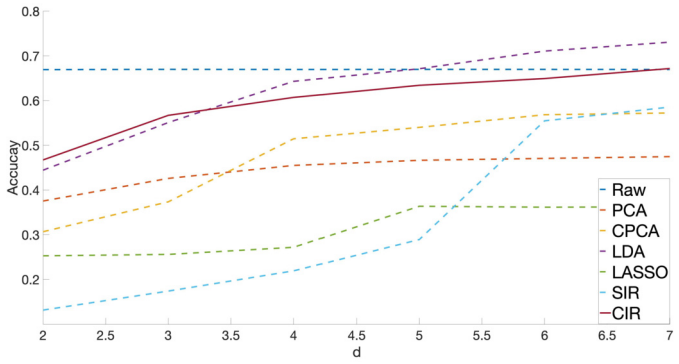


Figure 6: Classification accuracy by KNN for COVID-19 data.

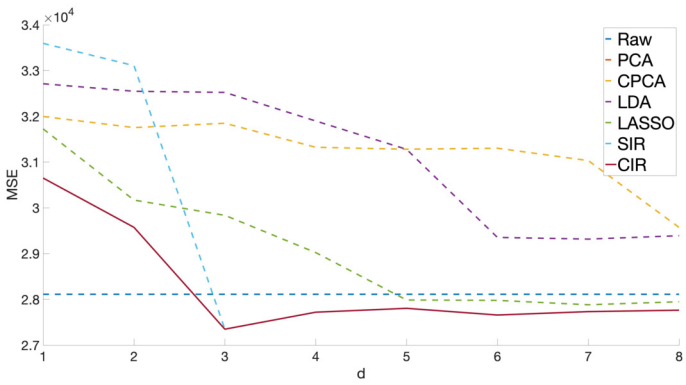


Figure 7: MSE of linear regression for plasma retinol data.

CIR is an improvement over all other methods for $d = 2, 3$. We show the accuracy of both the KNN-based classifier and the tree-based classifier in the supplementary material.

### 4.4 Plasma Retinol

The final dataset we consider is the plasma retinol dataset [49]. The dataset contains 315 observations of 14 variables, including age, sex, smoking status, BMI, vitamin use, calories, fat, fiber, cholesterol, dietary beta-carotene, dietary retinol consumed per day, number of alcoholic drinks consumed per week, and levels of plasma beta-carotene and plasma retinol.

In this example, $X \in \mathbb{R}^{315\times 12}$ represents measurements of the first 12 variables listed for all subjects, while $y_i$ represents the measurement of plasma beta-carotene, a variable of particular interest to scientists [49]. In contrast to the previous two examples, note that here $y_i$ is continuous, not categorical.

We apply unsupervised DR algorithms PCA, tSNE, and UMAP to $X$ and supervised DR algorithms LDA, LASSO, and SIR to $(X, Y)$ for $d = 1, \ldots, 8$. Similarly to the single-cell RNA sequencing application, we let $\widetilde{X} = X$ because there is no natural choice of background data. For the background label, we set $\widetilde{Y}$ as the continuous variable representing the level of plasma retinol, which shares certain information with $y_i$, and apply CPCA to $(X, \widetilde{X})$ and CIR to $(X, Y, \widetilde{X}, \widetilde{Y})$ for $d = 1, \ldots, 8$. We skip the visualization step in this case due to the poor visibility in terms of $y_i$.

After trying a few regression methods, namely linear regression [24], regression trees [9], Gaussian process regression [14], and neural networks [30], we present the prediction mean squared error (MSE) for the best method for this dataset, linear regression. That is, for each $d$ and the output $V$ from each DR algorithm, we fit a linear regression model to $(XV, Y)$. We also compare to a linear regression model based on raw data $(X, Y)$ as the baseline. Figure 7 demonstrates that CIR outperforms all other competitors, but matches SIR when $d \geq 3$.

Note that because $Y$ and $\widetilde{Y}$ are continuous, the number of slices to estimate $\Sigma_x$ and $\Sigma_{\widetilde{x}}$ needs to be carefully chosen and adjusted to ensure optimal performance. We use cross-validation to select 4 equally spaced partitions for the support of $Y$ and $\widetilde{Y}$. In the three applications presented above, CIR demonstrates superior overall performance over its supervised, unsupervised, contrastive, and non-contrastive competitors, especially in low dimension, i.e., $d = 2, 3$, which are the most crucial dimensions for visualization purposes.

In all four examples we considered, we consistently observed that CIR is the best among all methods when $d = 2, 3$. However, the gain is not obvious for higher dimensions. A possible reason for this is that when $d$ is relatively large, methods that use only the foreground data, such as SIR or LDA, capture both shared information and unique information in the foreground. Consequently, the improvement from incorporating the background group, or any contrastive model, becomes incremental. Fortunately, $d \leq 3$ are often the most important dimensions because they allow for visualization and interpretation.

## 5. DISCUSSION AND FUTURE WORK

In this work, we propose the CIR model and the associated optimization algorithm for supervised dimension reduction for datasets that are split into foreground and background groups. We provide a theoretical guarantee of the convergence of the CIR algorithm under mild conditions. We have shown that our CIR model outperforms competitors in multiple biomedical datasets, including mouse protein expression data, single-cell RNA sequencing data, and plasma retinol data. However, there are several important future directions that remain unaddressed in this paper, as outlined below.

*Multi-Treatment* It is natural to consider how our model can be extended to studies with multiple treatments. For example, in medical treatment, there might be more than one treatment for patients with certain disease. In [61], it has

been shown that the number of treatment groups puts a hard constraint on the target dimension. It is interesting to generalize from a single-treatment structure to a multi-treatment structure (e.g., [47]), where the loss function needs more sophisticated design.

Another direction in multi-group scenario is to combine multiple CIR models trained on different pairs of bi-groups. As pointed out by [65], the generalization error in the contrastive regression model stacking needs to be controlled, and one possible way is to follow divergence mixing as proposed by [29], with a careful normalization. The major difficulty in training such stacking model is how to devise a sequential optimization for model training.

*Consistency and Sufficient Dimension Reduction* The consistency of the proposed CIR model remains open. Theorems 2 and 3 ensure that the resulting solution must be a stationary point, but we did not discuss whether these stationary points are consistent estimates. The consistency of the estimates is also affected by the choice of $\alpha$, because $\alpha = 0$ will render this CIR into a classical SIR for the foreground group. This consistency problem also has practical importance, as it explicitly expresses the trade-off between the expressive contrastiveness and the emphasis on the effective lower-dimensional structures. The group information and statistical sufficiency compete against each other, as we observed in the experiments, thus a range of $\alpha$ that balances between these two factors are of interest and might be answered by the consistency result.

Furthermore, SIR has the drawback of missing the totality central subspace when the symmetry assumption in $x$ is lost [37]. [18] proposed the sliced average variance estimator (SAVE) estimator for addressing this problem, which raises the natural question of how to generalize this high-moment SDR method to the contrastive setting.

*Loss Function* Our loss function (2.3) is nonstandard, which raises many questions. For example, the relation between number of local minima and $A$, $B$, $\widetilde{A}$, $\widetilde{B}$, $\alpha$ remains open. Moreover, although $f$ cannot be continuously extended to the full Euclidean space $\mathbb{R}^{p\times d}$, if we restrict the domain to be a submanifold of $\mathrm{St}(p,d)$, $f$ might be extended to the convex hull of the submanifold. This extension will enable us to apply some other efficient optimization algorithms with strong theoretical guarantee [7]. Furthermore, Appendix G raises the question of the validity of a fixed-point algorithm based on Ricatti equations that may lead to a more efficient algorithm to minimize $f$ without involving the gradient.

*Scalability* The method we presented in this paper does not handle high-dimensional data in the sense of $p > n, m$, because matrices $B$ and $\widetilde{B}$ are singular in this situation. A possible extension of CIR to $p > n$ is to use the same technique as sparse PCA [72], which introduces a penalty term to enforce sparsity.

## APPENDIX A. PROOF TO PROPOSITION 1

We need to simplify the loss function $f(V)$ for subsequent analyses. Recall that the projection to the subspace $\mathcal{C}(\Sigma_{xx}V)$ and $\mathcal{C}(\Sigma_{\widetilde{x}\widetilde{x}}V)$ is given by the following projection matrices:

$$P_{\Sigma_{xx}V} \coloneqq \Sigma_{xx}V\big[V^\top\Sigma_{xx}^2V\big]^{-1}V^\top\Sigma_{xx}$$
$$P_{\Sigma_{\widetilde{x}\widetilde{x}}V} \coloneqq \Sigma_{\widetilde{x}\widetilde{x}}V\big[V^\top\Sigma_{\widetilde{x}\widetilde{x}}^2V\big]^{-1}V^\top\Sigma_{\widetilde{x}\widetilde{x}}.$$

Because projection matrices are idempotent, that is, $P^2_{\Sigma_{xx}V} = P_{\Sigma_{xx}V}$ and $P^2_{\Sigma_{\widetilde{x}\widetilde{x}}V} = P_{\Sigma_{\widetilde{x}\widetilde{x}}V}$, we can rewrite the loss function as follows:

$$\begin{aligned}
f(V) &= \mathbb{E}_y\big[d^2\big(\mathbb{E}[x \mid y], \mathcal{C}(\Sigma_{xx}V)\big)\big] \\
&\quad - \alpha\mathbb{E}_{\widetilde{y}}\big[d^2\big(\mathbb{E}[\widetilde{x} \mid \widetilde{y}], \mathcal{C}(\Sigma_{\widetilde{x}\widetilde{x}}V)\big)\big] \\
&= \mathbb{E}_y\big[\|v_y - P_{\Sigma_{xx}V}v_y\|^2\big] - \alpha\mathbb{E}_{\widetilde{y}}\big[\|v_{\widetilde{y}} - P_{\Sigma_{\widetilde{x}\widetilde{x}}V}v_{\widetilde{y}}\|^2\big] \\
&= \mathbb{E}_y\big[v_y^\top v_y - v_y^\top P^2_{\Sigma_{xx}V}v_y\big] - \alpha\mathbb{E}_{\widetilde{y}}\big[v_{\widetilde{y}}^\top v_{\widetilde{y}} - v_{\widetilde{y}}^\top P_{\Sigma_{\widetilde{x}\widetilde{x}}V}v_{\widetilde{y}}\big] \\
&= \mathbb{E}_y\big[v_y^\top v_y - v_y^\top P_{\Sigma_{xx}V}v_y\big] - \alpha\mathbb{E}_{\widetilde{y}}\big[v_{\widetilde{y}}^\top v_{\widetilde{y}} - v_{\widetilde{y}}^\top P_{\Sigma_{\widetilde{x}\widetilde{x}}V}v_{\widetilde{y}}\big].
\end{aligned}$$

The solution to the optimization problem defined by this loss function, if it exists, leads to our CIR model.

We remove the constant terms $\mathbb{E}_y[v_y^\top v_y]$ and $\mathbb{E}_{\widetilde{y}}[v_{\widetilde{y}}^\top v_{\widetilde{y}}]$ that are independent of $V$ and continue to simplify $f(V)$:

$$\begin{aligned}
f(V) &= -\mathbb{E}_y\big[v_y^\top P_{\Sigma_{xx}V}v_y\big] + \alpha\mathbb{E}_{\widetilde{y}}\big[v_{\widetilde{y}}^\top P_{\Sigma_{\widetilde{x}\widetilde{x}}V}v_{\widetilde{y}}\big] \\
&= -\mathbb{E}_y\big[\mathrm{tr}\big(v_y^\top P_{\Sigma_{xx}V}v_y\big)\big] + \alpha\mathbb{E}_{\widetilde{y}}\big[\mathrm{tr}\big(v_{\widetilde{y}}^\top P_{\Sigma_{\widetilde{x}\widetilde{x}}V}v_{\widetilde{y}}\big)\big] \\
&= -\mathrm{tr}(\Sigma_x P_{\Sigma_{xx}V}) + \alpha\,\mathrm{tr}(\Sigma_{\widetilde{x}}P_{\Sigma_{\widetilde{x}\widetilde{x}}V}) \\
&= -\mathrm{tr}\big(V^\top\Sigma_{xx}\Sigma_x\Sigma_{xx}V\big[V^\top\Sigma_{xx}^2V\big]^{-1}\big) \\
&\quad + \alpha\,\mathrm{tr}\big(V^\top\Sigma_{\widetilde{x}\widetilde{x}}\Sigma_{\widetilde{x}}\Sigma_{\widetilde{x}\widetilde{x}}V\big[V^\top\Sigma_{\widetilde{x}\widetilde{x}}^2V\big]^{-1}\big) \\
&= -\mathrm{tr}\big(V^\top AV\big(V^\top BV\big)^{-1}\big) + \alpha\,\mathrm{tr}\big(V^\top\widetilde{A}V\big(V^\top\widetilde{B}V\big)^{-1}\big),
\end{aligned}$$

where $A = \Sigma_{xx}\Sigma_x\Sigma_{xx}$, $B = \Sigma_{xx}^2$, $\widetilde{A} = \Sigma_{\widetilde{x}\widetilde{x}}\Sigma_{\widetilde{x}}\Sigma_{\widetilde{x}\widetilde{x}}$, and $\widetilde{B} = \Sigma_{\widetilde{x}\widetilde{x}}^2$.

## APPENDIX B. PROOF TO LEMMA 1

$$\begin{aligned}
\frac{\partial f}{\partial V} &= -2AV\big(V^\top BV\big)^{-1} \\
&\quad - 2BV\big(V^\top BV\big)^{-1}V^\top AV\big(V^\top BV\big)^{-1} \\
&\quad + \alpha\big\{2\widetilde{A}V\big(V^\top\widetilde{B}V\big)^{-1} \\
&\quad - 2\widetilde{B}V\big(V^\top\widetilde{B}V\big)^{-1}V^\top\widetilde{A}V\big(V^\top\widetilde{B}V\big)^{-1}\big\}.
\end{aligned}$$

Recall that the projection to the tangent space of the Stiefel manifold $\mathrm{St}(p,d)$ at $V$ is given by

$$\mathrm{Proj}_V(Z) = Z - V\,\mathrm{Sym}\big(V^\top Z\big),\ \forall Z \in T_V\,\mathrm{St}(p,d),$$

where $\mathrm{Sym}(X) \coloneqq \frac{X+X^\top}{2}$ is the symmetrizer. Then observe that the following equations involving the pair $A$, $B$ and the

pair $\widetilde{A}$, $\widetilde{B}$ have to satisfy the following equations:

$$V^\top\big(2AV\big(V^\top BV\big)^{-1}-2BV\big(V^\top BV\big)^{-1}V^\top AV\big(V^\top BV\big)^{-1}\big) = 0$$
$$V^\top\big(2\widetilde{A}V\big(V^\top \widetilde{B}V\big)^{-1}-2\widetilde{B}V\big(V^\top \widetilde{B}V\big)^{-1}V^\top \widetilde{A}V\big(V^\top \widetilde{B}V\big)^{-1}\big) = 0.$$

That is, $V^\top \frac{\partial f}{\partial V} = 0$. As a result, the gradient of $f$ is given by

$$\operatorname{grad} f(V) = \operatorname{Proj}_V\left(\frac{\partial f}{\partial V}\right) = \frac{\partial f}{\partial V} - V\operatorname{Sym}\left(V^\top \frac{\partial f}{\partial V}\right) = \frac{\partial f}{\partial V}.$$

## APPENDIX C. PROOF OF THEOREM 1

If $V$ is a local minimizer (i.e., a stationary point for the optimization problem (2.3)), then $\operatorname{grad} f(V) = 0$, from Lemma 1 we have

$$AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) = BVF(V) - \alpha\widetilde{B}V\widetilde{F}(V),$$

where $E(V) = (V^\top BV)^{-1}$, $\widetilde{E}(V) = (V^\top \widetilde{B}V)^{-1}$, $F(V) = (V^\top BV)^{-1}V^\top AV(V^\top BV)^{-1}$ and $\widetilde{F}(V) = (V^\top \widetilde{B}V)^{-1}V^\top \widetilde{A}V(V^\top \widetilde{B}V)^{-1}$.

## APPENDIX D. PROOF OF THEOREM 2

By Theorem 1 and Corollary 1 in [50], it suffices to show $f$ is continuously differentiable, which is a direct corollary of Lemma 1.

## APPENDIX E. OPTIONS FOR $\eta_k$

To introduce other options for $\eta_k$, we need the following definition.

**Definition 2** (Gradient-related sequence, see [2, p. 62, Definition 4.2.1])**.** Given a function $f$ on a Riemannian manifold $M$, a sequence in tangent space $\{\eta_k\}, \eta_k \in T_{V_k}M$, where $V_k$ are defined through the iterative formula $V_{k+1} = R_{V_k}(t_k\eta_k)$, and $R_{x_k}$ can be any retraction (e.g., global retraction mapping $\operatorname{Retr}_V : T_VM \to M, \xi \mapsto (V+\xi)(I_d + \xi^\top\xi)^{-1/2}$ on $St(n,p)$), is called **gradient-related** if, for any subsequence of $\{V_k\}_{k\in K\subset\{1,2,\ldots,n\}}$ that converges to a non-critical point of $f$, the corresponding subsequence $\{\eta_k\}_{k\in K}$ is bounded and satisfies

$$\lim_{k\to\infty, k\in K} \sup \big\langle \operatorname{grad} f(V_k), \eta_k\big\rangle_M < 0.$$

This means that the cosine of gradient and update $\eta_k$ needs to form an acute angle for only critical points. Note that a naive Newton step is not necessarily gradient-related (see p. 122 in [2]). In particular, $\eta_k = -\operatorname{grad} f(V_k)$ results in a gradient-related sequence, and is suggested by [2] as a natural choice.

## APPENDIX F. PROOF OF THEOREM 3

The first assertion regarding consistency is from Theorem 4.3.1 in [2], which requires our loss function $f$ to be continuously differentiable, a direct corollary of Lemma 1.

By the compactness of $\operatorname{St}(p,d)$, the level set $\mathcal{L} \coloneqq \{V \in \operatorname{St}(p,d) : \ f(V) \le f(V_0)\}$ is compact for any $V_0 \in \operatorname{St}(p,d)$, the second assertion follows Corollary 4.3.2 in [2].

The third assertion regarding the convergence rate involves second-order conditions, i.e., the Hessian of $f$. Let $D^2f$ be the Hessian computed in Euclidean coordinates, that is, $(D^2f|_V)_{ij,kl} \coloneqq \frac{\partial f}{\partial V_{ij}\partial V_{kl}}$, then for tangent vectors $\Omega_1, \Omega_2 \in T_V\operatorname{St}(p,d)$, the Hessian is given by [2]

$$\operatorname{Hess}(f)(\Omega_1,\Omega_2) = \underbrace{D^2f|_V(\Omega_1,\Omega_2)}_{①}$$
$$+\underbrace{\frac{1}{2}\operatorname{tr}\big(\big(\operatorname{grad} f(V)^\top\Omega_1V^\top + V^\top\Omega_1\operatorname{grad} f(V)^\top\big)\Omega_2\big)}_{②}$$
$$\underbrace{-\frac{1}{2}\operatorname{tr}\big(\big(V^\top\operatorname{grad} f(V) + \operatorname{grad} f(V)^\top V\big)\Omega_1^\top\big(I_p - VV^\top\big)\Omega_2\big)}_{③}.$$

By the definition of $f$, ① is $C^\infty$ in the Euclidean sense, so is continuous. By the continuity of $\operatorname{grad} f$, ② and ③ are also continuous since they are product or summation of continuous functions. Then the convergence rate follows Theorem 4.5.6 in [2].

## APPENDIX G. FIXED-POINT APPROACH TO OPTIMIZATION

Motivated by the first order optimality condition for the loss function (2.3), we seek a fixed-point method as an alternative to a gradient descent-based algorithm. Instead of solving equation (2.4) in one algebraic step, we separate the problem into the following 8 equations, which can be solved cyclically. Recall that $E(V) = (V^\top BV)^{-1}$, $\widetilde{E}(V) = (V^\top \widetilde{B}V)^{-1}$, $G(V) = V^\top AV$, and $\widetilde{G}(V) = V^\top \widetilde{A}V$ and suppress the index of $V_k$, i.e., $V = V_k$ and $V_1 = V_{k+1}$ for now for legibility:

$$AV_1E(V) - \alpha\widetilde{A}V\widetilde{E}(V) = BVE(V)G(V)E(V) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V)$$
$$AVE(V) - \alpha\widetilde{A}V_1\widetilde{E}(V) = BVE(V)G(V)E(V) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V)$$
$$AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) = BV_1E(V)G(V)E(V) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V)$$
$$AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) = BVE(V)G(V)E(V) - \alpha\widetilde{B}V_1\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V)$$
$$AVE(V_1) - \alpha\widetilde{A}V\widetilde{E}(V)$$

$$
\begin{aligned}
&= BVE(V_1)G(V)E(V_1) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V) \\
&AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V_1) \\
&= BVE(V)G(V)E(V) - \alpha\widetilde{B}V\widetilde{E}(V_1)\widetilde{G}(V)\widetilde{E}(V_1) \\
&AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) \\
&= BVE(V)G(V_1)E(V) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V) \\
&AVE(V) - \alpha\widetilde{A}V\widetilde{E}(V) \\
&= BVE(V)G(V)E(V) - \alpha\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V_1)\widetilde{E}(V)
\end{aligned}
$$

In each of the first four of these equations, $V_{k+1}$ can change independently, suggesting a convenient corresponding update rule. For the next two equations, we can premultiply by $[(BV)^\top(BV)]^{-1}(BV)^\top$ (and $[(\widetilde{B}V)^\top(\widetilde{B}V)]^{-1}(\widetilde{B}V)^\top$, respectively) to obtain the following equation:

$$
\begin{aligned}
&\big[(BV)^\top(BV)\big]^{-1}(BV)^\top AVE(V_{k+1}) - E(V_{k+1})G(V)E(V_{k+1}) \\
&\quad = \alpha\big[(BV)^\top(BV)\big]^{-1}(BV)^\top H_1, \qquad \text{(G.1)}
\end{aligned}
$$

where $H_1 = (\widetilde{A}V\widetilde{E}(V) - \widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V))$.

In practice, the cyclic update may not converge to stationary points of the optimization problem (2.3). However, when the designated cyclic update converges, it can be shown that equation (G.1) is in the form of an asymmetric algebraic Riccati equation in $E(V_{k+1})$ [6]. When we obtain a solution $E^* = E(V_{k+1})$ where $V = V_k$ is not a local optimum, the $E^*$ is not in $S^d_{++}$, which means we cannot use the Cholesky decomposition to solve for $V_{k+1}$ in the next update.

For the final two equations, we can write

$$
\begin{aligned}
&V_{k+1}^\top AV_{k+1} \\
&\quad = \big[\big(VE(V)\big)^\top\big(VE(V)\big)\big]^{-1}\big(VE(V)\big)^\top B^{-1}H_2E(V)^{-1},
\end{aligned}
$$

where

$$
H_2 = AVE(V) + \alpha\big(\widetilde{B}V\widetilde{E}(V)\widetilde{G}(V)\widetilde{E}(V) - \widetilde{A}V\widetilde{E}(V)\big).
$$

However, when $V = V_k$ is not a local optimum, again the right-hand side is not symmetric positive-definite, and so we cannot use the Cholesky decomposition to solve for $V_{k+1}$ in the next update.

Note that in order to require $V_{k+1} \in \mathrm{St}(p,d)$, the final step of each update rule should project the solution for $V_{k+1}$ onto $\mathrm{St}(p,d)$, which can be done by SVD; if $A = U\Sigma V^\top$, then $\pi(A) = UV^\top$.

Although this cyclic update regime does not immediately lead to a practical fixed-point optimization algorithm, it shows that our loss function has the classical link to a Ricatti equation (G.1), indicating that more efficient algorithms are possible.

## SUPPLEMENTARY MATERIAL

Additional experimental details are included in the supplementary material.

## FUNDING

SH was supported by NIH grants T32ES007018 and UM1 TR004406; DL was supported by NIH grants R01 AG079291, R56 LM013784, R01 HL149683, and UM1 TR004406, R01 LM014407, P30 ES010126.

*Accepted 3 October 2024*

## REFERENCES

[1] Abid, A., Zhang, M. J., Bagaria, V. K. and Zou, J. (2018). Exploring patterns enriched in a dataset with contrastive principal component analysis. *Nature Communications* **9**(1) 1–7.

[2] Absil, P.-A., Mahony, R. and Sepulchre, R. (2009). Optimization algorithms on matrix manifolds. In *Optimization Algorithms on Matrix Manifolds* Princeton University Press. https://doi.org/10.1515/9781400830244. MR2364186

[3] Alquicira-Hernandez, J., Sathe, A., Ji, H. P., Nguyen, Q. and Powell, J. E. (2019). scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. *Genome Biology* **20**(1) 1–17.

[4] Bader, G. D., Cary, M. P. and Sander, C. (2006). Pathguide: a pathway resource list. *Nucleic Acids Research* **34** 504–506.

[5] Becht, E., McInnes, L., Healy, J., Dutertre, C.-A., Kwok, I. W., Ng, L. G., Ginhoux, F. and Newell, E. W. (2019). Dimensionality reduction for visualizing single-cell data using UMAP. *Nature Biotechnology* **37**(1) 38–44.

[6] Bini, D. A., Iannazzo, B., Meini, B. and Poloni, F. (2008). Nonsymmetric algebraic Riccati equations associated with an M-matrix: recent advances and algorithms. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[7] Boumal, N., Absil, P.-A. and Cartis, C. (2019). Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis* **39**(1) 1–33. https://doi.org/10.1093/imanum/drx080. MR4023745

[8] Breiman, L. (1996). Bias, variance, and arcing classifiers. Technical Report, Tech. Rep. 460, Statistics Department, University of California, Berkeley .... https://doi.org/10.1214/aos/1024691079. MR1635406

[9] Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J. (2017) *Classification and regression trees*. Routledge. MR0726392

[10] Cai, Z., Li, R. and Zhu, L. (2020). Online sufficient dimension reduction through sliced inverse regression. *J. Mach. Learn. Res.* **21**(10) 1–25. MR4071193

[11] Caliński, T. and Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics – Theory and Methods* **3**(1) 1–27. https://doi.org/10.1080/03610927408827101. MR0375641

[12] Camastra, F. and Staiano, A. (2016). Intrinsic dimension estimation: Advances and open problems. *Information Sciences* **328** 26–41.

[13] Campadelli, P., Casiraghi, E., Ceruti, C. and Rozza, A. (2015). Intrinsic dimension estimation: Relevant techniques and a benchmark framework. *Mathematical Problems in Engineering* **2015** 759567. https://doi.org/10.1155/2015/759567. MR3417646

[14] Chen, Z., Wang, B. and Gorban, A. N. (2020). Multivariate Gaussian and Student-t process regression for multi-output prediction. *Neural Computing and Applications* **32**(8) 3005–3028.

[15] Chopra, S., Hadsell, R. and LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* **1** 539–546. IEEE.

[16] Cook, R. D. (1996). Graphics for regressions with a binary response. *Journal of the American Statistical Association* **91**(435) 983–992. https://doi.org/10.2307/2291717. MR1424601

[17] Cook, R. D. (2009) *Regression graphics: Ideas for studying regressions through graphics*. John Wiley & Sons. https://doi.org/10.1002/9780470316931. MR1645673

[18] Cook, R. D. and Weisberg, S. (1991). Sliced inverse regression for dimension reduction: Comment. *Journal of the American Statistical Association* **86**(414) 328–332. MR1137117

[19] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning* **20**(3) 273–297.

[20] Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory* **13**(1) 21–27.

[21] Dann, E., Cujba, A.-M., Oliver, A. J., Meyer, K. B., Teichmann, S. A. and Marioni, J. C. (2023). Precise identification of cell states altered in disease using healthy single-cell references. *Nature Genetics* **55**(11) 1998–2008.

[22] Davies, D. L. and Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2) 224–227.

[23] Edelman, A., Arias, T. A. and Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications* **20**(2) 303–353. https://doi.org/10.1137/S0895479895290954. MR1646856

[24] Freedman, D. A. (2009) *Statistical models: theory and practice*. cambridge university press. https://doi.org/10.1017/CBO9780511815867. MR2489600

[25] Girard, S., Lorenzo, H. and Saracco, J. (2022). Advanced topics in sliced inverse regression. *Journal of Multivariate Analysis* **188** 104852. https://doi.org/10.1016/j.jmva.2021.104852. MR4353861

[26] Hastie, T., Tibshirani, R. and Buja, A. (1994). Flexible discriminant analysis by optimal scoring. *Journal of the American Statistical Association* **89**(428) 1255–1270. MR1310220

[27] Higuera, C., Gardiner, K. J. and Cios, K. J. (2015). Self-organizing feature maps identify proteins critical to learning in a mouse model of Down syndrome. *PloS One* **10**(6) 0129126.

[28] Hilafu, H. and Safo, S. E. (2022). Sparse sliced inverse regression for high dimensional data analysis. *BMC Bioinformatics* **23**(1) 1–19.

[29] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation* **14**(8) 1771–1800.

[30] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences* **79**(8) 2554–2558. https://doi.org/10.1073/pnas.79.8.2554. MR0652033

[31] Hotelling, H. (1992). Relations between two sets of variates. In *Breakthroughs in Statistics* 162–190 Springer.

[32] Hsing, T. and Carroll, R. J. (1992). An asymptotic theory for sliced inverse regression. *The Annals of Statistics* **20**(2) 1040–1061. https://doi.org/10.1214/aos/1176348669. MR1165605

[33] Jiang, B. and Liu, J. S. (2014). Variable selection for general index models via sliced inverse regression. *The Annals of Statistics* **42**(5) 1751–1786. https://doi.org/10.1214/14-AOS1233. MR3262467

[34] Jiang, C.-R., Yu, W. and Wang, J.-L. (2014). Inverse regression for longitudinal data. *The Annals of Statistics* **42**(2) 563–591. https://doi.org/10.1214/13-AOS1193. MR3210979

[35] Jones, A., Townes, F. W., Li, D. and Engelhardt, B. E. (2022). Contrastive latent variable modeling with application to case-control sequencing experiments. *The Annals of Applied Statistics* **16**(3) 1268–1291. https://doi.org/10.1214/21-aoas1534. MR4455880

[36] Journée, M., Nesterov, Y., Richtárik, P. and Sepulchre, R. (2010). Generalized power method for sparse principal component analysis. *Journal of Machine Learning Research* **11**(2) 517–553. MR2600619

[37] Li, B. (2018) *Sufficient dimension reduction: Methods and applications with R*. Chapman and Hall/CRC.

[38] Li, D., Jones, A. and Engelhardt, B. (2020). Probabilistic contrastive principal component analysis. *arXiv preprint arXiv:2012.07977*.

[39] Li, D., Mukhopadhyay, M. and Dunson, D. B. (2017). Efficient manifold and subspace approximations with spherelets. *arXiv preprint arXiv:1706.08263*. https://doi.org/10.1111/rssb.12508. MR4494155

[40] Li, D., Mukhopadhyay, M. and Dunson, D. B. (2022). Efficient manifold approximation with spherelets. *Journal of the Royal Statistical Society Series B* **84**(4) 1129–1149. https://doi.org/10.1111/rssb.12508. MR4494155

[41] Li, K.-C. (1991). Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association* **86**(414) 316–327. MR1137117

[42] Li, L. and Yin, X. (2008). Sliced inverse regression with regularizations. *Biometrics* **64**(1) 124–131. https://doi.org/10.1111/j.1541-0420.2007.00836.x. MR2422826

[43] Li, L., Simonoff, J. S. and Tsai, C.-L. (2007). Tobit model estimation and sliced inverse regression. *Statistical Modelling* **7**(2) 107–123. https://doi.org/10.1177/1471082X0700700201. MR2749982

[44] Liao, Y.-t., Luo, H. and Ma, A. (2023). Efficient Bayesian selection of hyper-parameters for dimension reduction: Case studies for t-SNE and UMAP. *In preparation*.

[45] Lin, Q., Zhao, Z. and Liu, J. S. (2018). On consistency and sparsity for sliced inverse regression in high dimensions. *The Annals of Statistics* **46**(2) 580–610. https://doi.org/10.1214/17-AOS1561. MR3782378

[46] Luo, H. and Li, D. (2022). Spherical rotation dimension reduction with geometric loss functions, 1–56. arXiv:2204.10975. MR4777417

[47] Luo, H. and Strait, J. D. (2022). Nonparametric multi-shape modeling with uncertainty quantification. *arXiv preprint arXiv:2206.09127*.

[48] Murray, R., Demmel, J., Mahoney, M. W., Erichson, N. B., Melnichenko, M., Malik, O. A., Grigori, L., Dereziński, M., Lopes, M. E., Liang, T. and Luo, H. (2022). *Randomized Numerical Linear Algebra: a perspective on the field with an eye to software*.

[49] Nierenberg, D. W., Stukel, T. A., Baron, J. A., Dain, B. J., Greenberg, E. R. and Group, S. C. P. S. (1989). Determinants of plasma levels of beta-carotene and retinol. *American Journal of Epidemiology* **130**(3) 511–521. MR0210418

[50] Oviedo, H. and Dalmau, O. (2019). A scaled gradient projection method for minimization over the Stiefel manifold. In *Mexican International Conference on Artificial Intelligence* 239–250. Springer. https://doi.org/10.1007/s11075-020-01001-9. MR4269662

[51] Oviedo, H., Dalmau, O. and Lara, H. (2021). Two adaptive scaled gradient projection methods for Stiefel manifold constrained optimization. *Numerical Algorithms* **87**(3) 1107–1127. https://doi.org/10.1007/s11075-020-01001-9. MR4269662

[52] Quinlan, J. R. (1987). Simplifying decision trees. *International Journal of Man-Machine Studies* **27**(3) 221–234.

[53] Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* **20** 53–65.

[54] Ruhe, A. (1970). Perturbation bounds for means of eigenvalues and invariant subspaces. *BIT Numerical Mathematics* **10**(3) 343–354. https://doi.org/10.1007/bf01934203. MR0273802

[55] Severson, K. A., Ghosh, S. and Ng, K. (2019). Unsupervised learning with contrastive latent variable models. In *Proceedings of the AAAI Conference on Artificial Intelligence* **33** 4862–4869.

[56] Stephenson, E., Reynolds, G., Botting, R., Calero-Nieto, F., Morgan, M., Tuong, Z., Bach, K., Sungnak, W., Worlock, K., Yoshida, M. et al. (2021). Cambridge Institute of therapeutic immunology and infectious disease-national Institute of health research (CITIID-NIHR) COVID-19 BioResource collaboration, single-cell multi-omics analysis of the immune response in COVID-19. *Nat. Med* **27**(5) 904–916.

[57] Tagare, H. D. (2011). Notes on optimization on Stiefel manifolds.

*Yale University, New Haven.*

[58] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* **58**(1) 267–288. MR1379242

[59] Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research* **9**(11).

[60] Virta, J., Lee, K.-Y. and Li, L. (2022). Sliced inverse regression in metric spaces. *arXiv preprint arXiv:2206.11511*. MR4485085

[61] Vogelstein, J. T., Bridgeford, E. W., Tang, M., Zheng, D., Douville, C., Burns, R. and Maggioni, M. (2021). Supervised dimensionality reduction for big data. *Nature Communications* **12**(1) 1–9.

[62] Weinberger, E., Lin, C. and Lee, S.-I. (2023). Isolating salient variations of interest in single-cell data with contrastiveVI. *Nature Methods* **20**(9) 1336–1345.

[63] Weiss, M. and Tonella, P. (2022). Simple techniques work surprisingly well for neural network test prioritization and active learning (replicability study). In *Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis* 139–150.

[64] Wilkinson, L. and Luo, H. (2022). A distance-preserving matrix sketch. *Journal of Computational and Graphical Statistics* **31** 945–959. https://doi.org/10.1080/10618600.2022.2050246. MR4513361

[65] Wolpert, D. H. (1992). Stacked generalization. *Neural Networks* **5**(2) 241–259.

[66] Wu, H.-M., Kao, C.-H. and Chen, C.-h. (2020). Dimension reduction and visualization of symbolic interval-valued data using sliced inverse regression. *Advances in Data Science: Symbolic, Complex and Network Data* **4** 49–77.

[67] Wu, Q., Liang, F. and Mukherjee, S. (2013). Kernel sliced inverse regression: Regularization and consistency. In *Abstract and Applied Analysis* **2013**. Hindawi. https://doi.org/10.1155/2013/540725. MR3081598

[68] Wu, Q., Mukherjee, S. and Liang, F. (2008). Localized sliced inverse regression. *Advances in Neural Information Processing Systems* **21**. https://doi.org/10.1198/jcgs.2010.08080. MR2791260

[69] Xiao, H., Rasul, K. and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

[70] Young, M. D., Mitchell, T. J., Vieira Braga, F. A., Tran, M. G., Stewart, B. J., Ferdinand, J. R., Collord, G., Botting, R. A., Popescu, D.-M., Loudon, K. W. et al. (2018). Single-cell transcriptomes from human kidneys reveal the cellular identity of renal tumors. *Science* **361**(6402) 594–599.

[71] Zheng, G. X., Terry, J. M., Belgrader, P., Ryvkin, P., Bent, Z. W., Wilson, R., Ziraldo, S. B., Wheeler, T. D., McDermott, G. P., Zhu, J. et al. (2017). Massively parallel digital transcriptional profiling of single cells. *Nature Communications* **8**(1) 1–12.

[72] Zou, H., Hastie, T. and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics* **15**(2) 265–286. https://doi.org/10.1198/106186006X113430. MR2252527

[73] Zou, J. Y., Hsu, D. J., Parkes, D. C. and Adams, R. P. (2013). Contrastive learning using spectral methods. *Advances in Neural Information Processing Systems* **26**.

Sam Hawke. Department of Biostatistics, University of North Carolina at Chapel Hill, USA. E-mail address: shawke@unc.edu

Yueen Ma. Department of Statistics and Operations Research, University of North Carolina at Chapel Hill, USA. E-mail address: myueen@unc.edu

Hengrui Luo. Computational Research Division, Lawrence Berkeley Laboratory, USA. Department of Statistics, Rice University, USA. E-mail address: hrluo@lbl.gov; hrluo@rice.edu

Didong Li. Department of Biostatistics, University of North Carolina at Chapel Hill, USA. E-mail address: didongli@unc.edu