Deep Dimension Reduction for Supervised Representation Learning

Jian Huang*, Yuling Jiao, Xu Liao, Jin Liu*, and Zhou Yu

Abstract—The goal of supervised representation learning is to construct effective data representations for prediction. Among all the characteristics of an ideal nonparametric representation of high-dimensional complex data, sufficiency, low dimensionality and disentanglement are some of the most essential ones. We propose a deep dimension reduction approach to learning representations with these characteristics. The proposed approach is a nonparametric generalization of the sufficient dimension reduction method. We formulate the ideal representation learning task as that of finding a nonparametric representation that minimizes an objective function characterizing conditional independence and promoting disentanglement at the population level. We then estimate the target representation at the sample level nonparametrically using deep neural networks. We show that the estimated deep nonparametric representation is consistent in the sense that its excess risk converges to zero. Our extensive numerical experiments using simulated and real benchmark data demonstrate that the proposed methods have better performance than several existing dimension reduction methods and the standard deep learning models in the context of classification and regression.

Index Terms—Conditional independence, distance covariance, $f$-divergence, nonparametric estimation, neural networks.

Manuscript received 2 March 2023; revised 14 September 2023; accepted 28 November 2023. Date of publication 29 February 2024; date of current version 23 April 2024. The work of Jian Huang was supported in part by the National Natural Science Foundation of China under Grant 72331005 and in part by The Hong Kong Polytechnic University. The work of Yuling Jiao was supported in part by the National Natural Science Foundation of China under Grant 12371441, in part by the Fundamental Research Funds for the Central Universities, and in part by the Research Fund of Key Laboratory of Advanced Theory and Application in Statistics and Data Science, Ministry of Education (KLATASDS-MOE) of China. The work of Jin Liu was supported in part by the University Development Fund from The Chinese University of Hong Kong, Shenzhen, under Grant UDF01003033 and in part by the National Natural Science Foundation of China under Grant 72371283. The work of Zhou Yu was supported in part by the Shanghai Pilot Program for Basic Research under Grant TQ20220105 and in part by the National Natural Science Foundation of China under Grant 12371289. (Corresponding author: Jin Liu.)

Jian Huang is with the Department of Applied Mathematics, The Hong Kong Polytechnic University, Hong Kong, SAR, China (e-mail: j.huang@polyu.edu.hk).

Yuling Jiao is with the School of Mathematics and Statistics and the Hubei Key Laboratory of Computational Science, Wuhan University, Wuhan 430072, China (e-mail: yulingjiaomath@whu.edu.cn).

Xu Liao is with the Center of Quantitative Medicine, Duke-NUS Medical School, Singapore 169857 (e-mail: liuxu@duke.nus.edu).

Jin Liu is with the School of Data Science, The Chinese University of Hong Kong (Shenzhen), Shenzhen 518172, China (e-mail: liujinlab@cuhk.edu.cn).

Zhou Yu is with the School of Statistics, Key Laboratory of Advanced Theory and Application in Statistics and Data Science, Ministry of Education (KLATASDS-MOE), East China Normal University, Shanghai 200041, China (e-mail: zyu@stat.ecnu.edu.cn).

Communicated by F. Orabona, Associate Editor for Machine Learning and Statistics.

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TIT.2023.3340658.

Digital Object Identifier 10.1109/TIT.2023.3340658

I. INTRODUCTION

OVER the past decade, deep learning has achieved impressive successes in modeling high-dimensional complex data arising from many scientific fields. A key factor for these successes is the ability of neural network models to learn nonlinear representations from complex high-dimensional data [1], [2]. For example, convolutional neural networks are able to learn effective representations of image data. However, in general, optimizing the standard cross-entropy loss for classification and the least squares loss for regression do not guarantee that the learned representations enjoy any desired properties [3]. Therefore, it is imperative to develop principled approaches for constructing effective data representations.

Representation learning has emerged as an important framework for modeling complex data [1], with wide applications in classification, regression, imaging analysis, domain adaptation and transfer learning, among others. The goal of supervised representation learning is to construct effective representations of high-dimensional input data for various supervised learning tasks. In this paper, we propose a deep dimension reduction (DDR) method for sufficient representation learning. DDR aims at estimating a sufficient representation nonparametrically using deep neural networks based on the conditional independence principle.

There is a large body of literature on dimension reduction in statistics and machine learning. A prominent approach for supervised dimension reduction and representation learning is the sufficient dimension reduction (SDR) introduced in the seminal paper by [4]. A key aspect that distinguishes SDR from many other dimension reduction methods is that it does not make any model assumptions on the conditional distribution of the response given the predictors. In the framework of SDR, a semiparametric method, called sliced inverse regression (SIR), was first proposed for estimating the linear dimension reduction direction, or linear sufficient representation [4]. SIR requires the linearity and constant covariance conditions on the distribution of the predictors. Several approaches have been developed without assuming these conditions, including methods based on conditional covariance operators [5], mutual information [6], distance correlation [7], and semiparametric modeling [8]. These SDR methods focus on linear dimension reduction, that is, the features learned are linear functions of the original input variables. However, linear functions may not be adequate for representing high-dimensional complex data such as images and natural languages, due to the highly nonlinear nature of such data. Lee et al. [9] formulated a general sufficient dimension reduction framework in the nonlinear setting and proposed a generalized inverse regression approach using...
conditional covariance operators, but this method is computationally prohibitive with high-dimensional data such as the image datasets considered in Section VI. We refer to the review papers [10], [11], [12] and the monograph [13] for thorough reviews of SDR methods.

Among all the characteristics of an ideal representation for supervised learning, sufficiency, low dimensionality and disentanglement are some of the most essential ones [14]. Sufficiency is a basic property a representation should have. It is closely related to the concept of sufficient statistics in a parametric model [10]. In supervised representation learning, sufficiency is characterized by the conditional independence principle, which states that the original input data is conditionally independent of the response given the representation. In other words, a sufficient representation contains all the relevant information in the input data about the response.

Low dimensionality means that the representation should have as few components as possible to represent the underlying structure of the data. In the context of nonparametric representation learning, disentanglement refers to the requirement that the components of the representation should be statistically independent. This is an extension of and stronger than the orthogonal constraint in the linear representation setting, where the components of the linear representation are constrained to have orthonormal directions. The components in the learned representation can often be interpreted as corresponding to the latent structure of the observed data. A representation with these characteristics can make the model more interpretable and facilitates the downstream supervised learning tasks.

Inspired by the basic idea of SDR, we propose a deep dimension reduction (DDR) approach for supervised representation learning with the properties of sufficiency, low dimensionality and disentanglement. By taking the advantage of the strong capacities of deep neural networks in approximating high-dimensional functions for nonparametric estimation, we model the DDR representations, which we refer to as DDR map (DDRM) for convenience, using deep neural networks to capture the nonlinearity in the representation space. It would be difficult to use the traditional techniques for nonparametric estimation such as kernel smoothing and splines for multi- or high-dimensional function estimation in the context of representation learning. To characterize the conditional independence of the representation, we use the distance covariance [15] as the conditional independence measure that can be computed efficiently. We also promote the disentanglement for DDRM by regularizing its distribution to have independent components based on a divergence measure.

Our main contributions are as follows:

- We formulate a new nonparametric approach to dimension reduction by characterizing the sufficient dimension reduction map as a minimizer of a loss function measuring conditional independence and disentanglement.
- We estimate the sufficient dimension reduction map at the sample level nonparametrically using deep neural networks based on distance covariance for characterizing sufficiency and use $f$-divergence to promote disentanglement of the learned representation.
- We show that the estimated deep dimension reduction map is consistent in the sense that it achieves asymptotic sufficiency under mild conditions.
- We validate DDR via comprehensive numerical experiments and real data analysis in the context of regression and classification. We use the learned features based on DDR as inputs for linear regression and nearest neighbor classification. The resulting prediction accuracies are better than those based on linear dimension reduction methods for regression and deep learning models for classification. The PyTorch code for DDR is available at https://github.com/Liao-Xu/DDR.

The rest of the paper is organized as follows. In Section II we discuss the theoretical framework for learning a DDRM. This framework leads to the formulation of an objective function using distance correlation for characterizing conditional independence in Section III. We estimate the target DDRM based on the sample version of the objective function using deep neural networks and develop an efficient algorithm for training the DDRM. In Section IV we provide sufficient conditions under which estimated nonparametric representations achieve asymptotic sufficiency. This result provides strong theoretical support for the proposed method. The algorithm for implementing DDR is described in Section V. In Section VI we validate the proposed DDR via extensive numerical experiments and real data examples.

II. SUFFICIENT REPRESENTATION AND DISTANCE CORRELATION

Consider a pair of random vectors $(X, Y) \in \mathbb{R}^p \times \mathbb{R}^q$, where $X$ is a vector of predictors and $Y$ is a vector of response variables or labels. Our goal is to construct a representation of $X$ that possesses the three characteristics: sufficiency, low dimensionality and disentanglement.

A. Sufficiency

A measurable function $s : \mathbb{R}^p \to \mathbb{R}^d$ with $d \leq p$ is said to be a sufficient representation of $X$ if

$$Y \perp X | s(X), \tag{1}$$

that is, $Y$ and $X$ are conditionally independent given $s(X)$. This condition holds if and only if the conditional distribution of $Y$ given $X$ and that of $Y$ given $s(X)$ are equal. Therefore, the information in $X$ about $Y$ is completely encoded by $s(X)$. Such a function $s$ always exists, since if we simply take $s(x) = x$, then (1) holds trivially. This formulation is a nonparametric generalization of the basic condition in sufficient dimension reduction [4], [16], where it is assumed $s(x) = B^T x$ with $B \in \mathbb{R}^{p \times d}$ belonging to the Stiefel manifold, i.e., $B^T B = I_d$. Denote the class of sufficient representations satisfying (1) by

$$\mathcal{F} = \{ s : \mathbb{R}^p \to \mathbb{R}^d, s \text{ satisfies } Y \perp X \text{ given } s(X) \}.$$ 

For an injective measurable transformation $T : \mathbb{R}^d \to \mathbb{R}^d$ and $s \in \mathcal{F}$, $T \circ s(X)$ is also sufficient by the basic property.
of conditional probability. Therefore, the class \( \mathcal{F} \) is invariant in the sense that \( T \circ \mathcal{F} \subseteq \mathcal{F} \), provided \( T \) is injective, where \( T \circ \mathcal{F} = \{ T \circ s : s \in \mathcal{F} \} \). An important class of transformations is the class of affine transformations, \( T \circ s = As + b \), where \( A \) is a \( d \times d \) nonsingular matrix and \( b \in \mathbb{R}^d \).

**B. Space of Nonparametric Sufficient Representations**

The nonparametric sufficient representations are nonuniquie and the space of such representations is large, since if (1) holds for \( s \), it also holds for any one-to-one transformation of \( s \). We propose to narrow the space of such representations by constraining the distributional properties of \( s(x) \).

Among the sufficient representations, it is preferable to have those with a simple statistical distribution and whose components are independent, that is, the components are disentangled. For a sufficient representation \( s(X) \), let \( \Sigma_s = \text{Cov}(s(X)) \). Suppose \( \Sigma_s \) is positive definite, then \( \Sigma_s^{-1/2} s(X) \) is also a sufficient representation. Therefore, we can always rescale \( s(X) \) such that it has identity covariance matrix. To further simplify the statistical structure of a representation \( s \), we also impose the constraint that it is rotation invariant in distribution, that is, \( Qs(X) = s(X) \) in distribution for any orthogonal matrix \( Q \in \mathbb{R}^{d \times d} \). By the Maxwell characterization of the Gaussian distributions \([17]\) a random vector of dimension two or more with independent components is rotation invariant in distribution if and only if it is Gaussian with zero mean and a spherical covariance matrix. Therefore, after absorbing the scaling factor, for a sufficient representation map to be have independent components and be rotation invariant, it is necessarily distributed as \( N_d(0, I_d) \). Denote

\[
\mathcal{M} = \{ R : \mathbb{R}^p \rightarrow \mathbb{R}^d, R(X) \sim N(0, I_d) \}. \tag{2}
\]

Now our problem becomes that of finding a representation in \( \mathcal{F} \cap \mathcal{M} \), the intersection of the Fisher class and the Maxwell class.

Does such a sufficient representation exist? The following result from the optimal transport theory gives an affirmative answer and guarantees the existence of such a representation under mild conditions \([18]\).

**Lemma 1:** Let \( \mu \) be a probability measure on \( \mathbb{R}^d \). Suppose it has finite second moment and is absolutely continuous with respect to the standard Gaussian measure, denoted by \( \gamma_d \). Then it admits a unique optimal transportation map \( T : \mathbb{R}^d \rightarrow \mathbb{R}^d \) such that \( T_{\#} \mu = \gamma_d \equiv N(0, I_d) \), where \( T_{\#} \mu \) denotes the pushforward distribution of \( \mu \) under \( T \). Moreover, \( T \) is injective \( \mu \)-almost everywhere.

Denote the law of a random vector \( X \) by \( \mu_X \). Lemma 1 implies that, for any \( s \in \mathcal{F} \) with \( \mathbb{E}[|s(X)|^2] < \infty \) and \( \mu_{s(X)} \) absolutely continuous with respect to \( \gamma_d \), there exists a map \( T^* \) transforming the distribution of \( s(X) \) to \( N(0, I_d) \). Therefore, \( R^* := T^* \circ s \in \mathcal{F} \cap \mathcal{M} \), that is,

\[
X \perp Y \mid R^*(X) \quad \text{and} \quad R^*(X) \sim N(0, I_d). \tag{3}
\]

The requirement that \( R^*(X) \sim N(0, I_d) \) can be considered a regularization on the distribution of \( R^*(X) \). This is similar to the ridge regression where the ridge penalty can be derived from a spherical normal prior on the regression coefficient.

We note that it suffices to estimate the function \( R^* \), not \( s \) and \( T^* \) separately, since \( R^* \) satisfies the conditional independence requirement.

The independence requirement for the components of \( R^* \) is reminiscent of the same requirement in the independent component analysis (ICA) \([19, 20]\). ICA is a method for estimating hidden factors that underlie a random vector \( X \). It posits that \( X \) is a linear transformation of an unknown random vector with independent components, but the transformation is unknown. The goal of ICA is to estimate this linear transformation. DDR differs from ICA in three crucial aspects. First, DDR is a supervised method that seeks to find a data representation such that the response is conditionally independent given this representation, while ICA is an unsupervised method that attempts to identify independent latent factors underlying the original data vector. Second, DDR seeks a nonparametric function \( R^* \) such that \( R^*(X) \) has independent components, while ICA attempts to find a matrix \( W \in \mathbb{R}^{p \times p} \) such that the components of \( WX \) are independent. Third, the distribution of \( R^*(X) \) can be Gaussian; in contrast, a basic restriction in ICA is that the independent components must be non-Gaussian. There is a large body of literature on ICA, see the Nordhausen and Oja \([21]\) and the references therein.

**III. NONPARAMETRIC ESTIMATION OF REPRESENTATION MAP**

The discussions in Section II lay the ground for formulating an objective function that can be used for constructing a DDRM \( R^* \) satisfying (3), that is, \( R^* \) is sufficient and disentangled.

**A. Population Objective Function**

Let \( \mathcal{V} \) be a measure of dependence between random variables \( X \) and \( Y \) with the following properties: (a) \( \mathcal{V}[X, Y] \geq 0 \) with \( \mathcal{V}[X, Y] = 0 \) if and only if \( X \perp Y \); (b) \( \mathcal{V}[X, Y] \geq \mathcal{V}[R(X), Y] \) for all measurable function \( R \); and (c) \( \mathcal{V}[X, Y] = \mathcal{V}[R^*(X), Y] \) if and only if \( R^* \in \mathcal{F} \). These properties imply that \( R^* \in \mathcal{F} \) if and only of \( R^* \in \text{argmin}_R \{ -\mathcal{V}[R(X), Y] \} \).

For the normality regularization in (3), we use a divergence measure \( D \) to quantify the difference between \( \mu_{R(X)} \) and the standard normal distribution \( \gamma_d \). This measure should satisfy the condition \( D(\mu_{R(X)} || \gamma_d) \geq 0 \) for every measurable function \( R \) and \( D(\mu_{R(X)} || \gamma_d) = 0 \) if and only if \( R \in \mathcal{M} \). The \( f \)-divergences, including the KL-divergence, satisfy this condition. It follows that \( R^* \in \mathcal{M} \) if and only if \( R^* \in \text{argmin}_R \{ D(\mu_{R(X)} || \gamma_d) \} \).

Then the problem of finding a sufficient and disentangle map \( R^* \) becomes a constrained minimization problem:

\[
\text{argmin}_R - \mathcal{V}[R(X), Y] \text{ subject to } D(\mu_{R(X)} || \gamma_d) = 0.
\]

The Lagrangian form of this minimization problem is

\[
\mathcal{L}(R) = -\mathcal{V}[R(X), Y] + \lambda D(\mu_{R(X)} || \gamma_d),
\]

where \( \lambda \geq 0 \) is a tuning parameter. This parameter provides a balance between the sufficiency property and the disentanglement constraint. A small \( \lambda \) leads to a representation with more emphasis on sufficiency, while a large \( \lambda \) yields...
a representation with more emphasis on disentanglement. We show in Theorem 3 below that any $R^*$ satisfying (3) is a minimizer of $\mathcal{L}(R)$. Therefore, we can train a DDRM by minimizing an empirical version of $\mathcal{L}(R)$.

There are several options for $\mathcal{V}$ with the properties (a)-(c) described above. For example, we can take $\mathcal{V}$ to be the mutual information. However, in addition to estimating the DDRM $R$, this choice requires nonparametric estimation of the ratio of the joint density and the marginal densities of $Y$ and $R(X)$, which is not an easy task. To be specific, in this work we use the distance covariance [15] between $Y$ and $R(X)$, which has an elegant $U$-statistic expression. It does not involve additional unknown quantities and is easy to compute. For the divergence measure of two distributions, we use the $f$-divergence [22], which includes the KL-divergence as a special case.

### B. Empirical Objective Function

In this subsection, we formulate the objective function for the proposed deep dimension reduction method. We first describe some essentials about distance covariance and $f$-divergence.

1) Distance Covariance: We recall the concept of distance covariance [15], which characterizes the dependence of two random variables. Let $t \in \mathbb{R}^d$, $s \in \mathbb{R}^m$, let $\psi_Z(t) = \mathbb{E}[\exp(t^T Z)]$, $\psi_Y(s) = \mathbb{E}[\exp(s^T Y)]$, and $\psi_{Z,Y}(t,s) = \mathbb{E}[\exp((t^T Z + s^T Y))]$ be the characteristic functions of random vectors $Z \in \mathbb{R}^d$, $Y \in \mathbb{R}^m$, and the pair $(Z, Y)$, respectively. The squared distance covariance $\mathcal{V}[Z, Y]$ is defined as

$$\mathcal{V}[Z, Y] = \int_{\mathbb{R}^{d+m}} \frac{|\psi_{Z,Y}(t,s) - \psi_Z(t)\psi_Y(s)|^2}{c_d c_m \|t\|^{d+1}\|s\|^{m+1}} dt ds,$$

where $c_d = \frac{\pi^{d/2}}{\Gamma(\frac{d+1}{2})}$. Given $n$ i.i.d copies $(Z_i, Y_i)_{i=1}^n$ of $(Z, Y)$, an unbiased estimator of $\mathcal{V}$ is the empirical distance covariance $\hat{\mathcal{V}}_n[Z, Y] = \frac{1}{C_n} \sum_{1 \leq i_1 < i_2 < \cdots < i_n \leq n} h((Z_{i_1}, Y_{i_1}), \ldots, (Z_{i_n}, Y_{i_n}))$,

where $h$ is the kernel defined by

$$h((z_1, y_1), \ldots, (z_i, y_i)) = \frac{1}{4} \sum_{1 \leq i_1 < i_2 < \cdots < i_n \leq n} \|z_i - z_j\| \|y_i - y_j\| + \frac{1}{24} \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq n} \|z_i - z_j\| \|y_i - y_j\| + \frac{1}{24} \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq n} \|z_i - z_j\| \|y_i - y_j\| - \frac{1}{24} \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq n} \|z_i - z_j\| \|y_i - y_j\|.$$

For a categorical response $Y$ in multi-class classification problems, we can use one-hot vectors to code the classes, i.e., for the $k$th class, $Y$ is a unit vector with $k$th element equaling 1 and the remaining elements being 0. The $L_2$ distance between two observed responses $y_i$ and $y_j$ is

$$\|y_i - y_j\|_2 = \begin{cases} 0, & \text{if } y_i = y_j, \\ \sqrt{2}, & \text{if } y_i \neq y_j. \end{cases}$$

Note that the number $\sqrt{2}$ simply scales the whole objective function and does not affect the solution.

2) $f$-Divergence: Let $\mu$ and $\gamma$ be two probability measures on $\mathbb{R}^d$. The $f$-divergence [22] between $\mu$ and $\gamma$ with $\mu \ll \gamma$ is defined as

$$\mathbb{D}_f(\mu||\gamma) = \int_{\mathbb{R}^d} f\left(\frac{d\mu}{d\gamma}\right) d\gamma,$$

where $f : \mathbb{R}^+ \rightarrow \mathbb{R}$ is a differentiable convex function satisfying $f(1) = 0$. Let $f^*$ be the Fenchel conjugate of $f$ [24], defined by

$$f^*(t) = \sup_{x \in \mathbb{R}} \{tx - f(x)\}, t \in \mathbb{R}. \quad (6)$$

The $f$-divergence (5) admits the following variational formulation [25].

**Lemma 2:** Suppose that $f$ is differentiable, proper, convex and lower-semicontinuous on its domain. Then,

$$\mathbb{D}_f(\mu||\gamma) = \max_{D: \mathbb{R}^d \rightarrow \text{dom}(f^*)} \mathbb{E}_{Z \sim \mu} D(Z) - \mathbb{E}_{W \sim \gamma} f^*(D(W)),$$

where $f^*$ is defined in (6). In addition, the maximum is attained at $D(x) = f^*(\frac{d\mu}{d\gamma}(x))$.

Commonly used divergence measures include the Kullback-Leibler (KL) divergence, the Jensen-Shanon (JS) divergence and the $\chi^2$-divergence. We summarize the details in Table I.

| $f$ | $f^*(t)$ | $\mathbb{D}_f(\mu||\gamma)$ |
|-----|----------|------------------------------|
| KL  | $-\log \frac{d\mu}{d\gamma}$ | $\int_{\mathbb{R}^d} (-\log \frac{d\mu}{d\gamma}) d\mu$ |
| JS  | $\frac{1}{2}\left(\log \frac{d\mu}{d\gamma} + \log \frac{d\gamma}{d\mu} - 1\right)$ | $\frac{1}{2}\int_{\mathbb{R}^d} \left(\log \frac{d\mu}{d\gamma} + \log \frac{d\gamma}{d\mu} - 1\right) d\mu$ |
| $\chi^2$ | $\left(\frac{d\mu}{d\gamma} - 1\right)^2$ | $\int_{\mathbb{R}^d} \left(\frac{d\mu}{d\gamma} - 1\right)^2 d\mu$ |

3) Empirical Objective Function for DDRM: We are now ready to formulate an empirical objective function for learning DDRM. Let $R \in \mathcal{M}$, where $\mathcal{M}$ is the Maxwell class defined in (2). By the variational formulation (7), we can write the population version of the objective function (4) as

$$\mathcal{L}(R) = -\mathcal{V}[R(X), Y] + \lambda \max_D \{\mathbb{E}_{X \sim \mu_X} D(R(X)) - \mathbb{E}_{W \sim \gamma_W} f^*(D(W))\}. \quad (8)$$
This expression is convenient since we can simply replace the expectations by the corresponding empirical averages.

**Theorem 3:** We have $R^* \in \arg \min_{R \in \mathcal{M}} \mathcal{L}(R)$ provided (3) holds.

According to Theorem 3, it is natural to estimate $R^*$ based on the empirical version of the objective function (8) when a random sample $\{(X_i, Y_i)\}_{i=1}^n$ is available.

We estimate $R^*$ nonparametrically using feedforward neural networks (FNN). Two networks are employed: the representor network $R_\theta$ with parameter $\theta$ for estimating $R^*$ and a second network $D_\phi$ with parameter $\phi$ for estimating the discriminator $D$. For any function $f(x) : \mathcal{X} \rightarrow \mathbb{R}^d$, denote $\|f\|_{\infty} = \sup_{x \in \mathcal{X}} |f(x)|$, where $\| \cdot \|$ is the Euclidean norm.

- **Representor network** $R_\theta$: This network is used for training $R^*$. Let $\mathbf{R} \equiv \mathbf{R}_{\mathcal{H}, \mathcal{W}, \mathcal{S}}$ be the set of such ReLU neural networks $R_\theta : \mathbb{R}^p \rightarrow \mathbb{R}^d$, with parameter $\theta$, depth $\mathcal{H}$, width $\mathcal{W}$, size $\mathcal{S}$.

  Here the depth $\mathcal{H}$ refers to the number of hidden layers, so the network has $\mathcal{H} + 1$ layers in total. A $(\mathcal{H} + 1)$-vector $(w_0, w_1, \ldots, w_{\mathcal{H}})$ specifies the width of each layer, where $w_0 = p$ is the dimension of the input data and $w_\mathcal{H} = d$ is the dimension of the output. The width $\mathcal{W} = \max\{w_1, \ldots, w_\mathcal{H}\}$ is the maximum width of the hidden layers. The size $\mathcal{S} = \sum_{i=0}^{\mathcal{H}} w_i (w_i + 1)$ is the total number of parameters in the network.

- **Discriminator network** $D_\phi$: This network is used as the witness function for checking whether the distribution of the estimator of $R^*$ is approximately the same as $\mathcal{N}(0, I)$.

  Similarly, denote $\mathbf{D} \equiv \mathbf{D}_{\mathcal{H}, \mathcal{W}, \mathcal{S}}$ as the set of ReLU neural networks $D_\phi : \mathbb{R}^d \rightarrow \mathbb{R}$ with parameter $\phi$, depth $\mathcal{H}$, width $\mathcal{W}$, size $\mathcal{S}$.

Let $\{W_i\}_{i=1}^n$ be $n$ i.i.d random vectors drawn from $\gamma_d$. The estimated DDRM is defined by

$$\hat{R}_\theta \in \arg \min_{R \in \mathbf{R}} \hat{\mathcal{L}}(R_\theta),$$

where $\hat{\mathcal{L}}(R_\theta) = -\hat{V}_n[R_\theta(X), Y] + \lambda \hat{\mathbb{D}}_f(\mu_{R_\theta(X)} \| \gamma_d)$. Here $\hat{V}_n[R_\theta(X), Y]$ is an unbiased and consistent estimator of $\mathbb{V}[R_\theta(X), Y]$ as defined in (5) based on $\{(R_\theta(X_i), Y_i), i = 1, \ldots, n\}$ and

$$\hat{\mathbb{D}}_f(\mu_{R_\theta(X)} \| \gamma_d) = \max_{D_\phi \in \mathbf{D}} \frac{1}{n} \sum_{i=1}^n |D_\phi(R_\theta(X_i)) - f^*(D_\phi(W_i))|.$$  \hfill (10)

This objective function consists of two terms: (a) the term $\lambda \hat{V}_n[R_\theta(X), Y]$ is an unbiased and consistent estimator of $\lambda \mathbb{V}[R_\theta(X), Y]$, which is a measure that quantifies the conditional independence $X \perp Y | R_\theta(X)$; (b) the term $\hat{\mathbb{D}}_f(\mu_{R_\theta(X)} \| \gamma_d)$ promotes disentanglement among the components of $R_\theta(X)$ by encouraging $R_\theta(X)$ to be distributed as $N(0, I_d)$. This is the dual form of the $f$-GAN loss [26], [28]. We note that GANs seek to find a map from a reference distribution such as Gaussian to the data space, here we do the reverse and try to find a representation of the data to be distributed like a reference distribution.

**IV. CONSISTENCY**

We establish the consistency of the estimated DDRM in the sense that the excess risk $\mathcal{L}(R_\hat{\theta}) - \mathcal{L}(R^*)$ converges to zero, where $R_\hat{\theta}$ is the deep nonparametric estimator in (9). It is clear that to achieve consistency, it is necessary to require the network parameters to increase as the sample size increases. This is similar to requiring the bandwidth of a nonparametric kernel density estimator to depend on the sample size. There is an extensive literature on how to select the bandwidth parameter and the model.

**Theorem 4:** Set $\lambda = O(1)$. Suppose conditions (A1)-(A3) hold and set the network parameters according to (N1)-(N2). Then $\mathbb{E}_{\{X_i, Y_i\}_{i=1}^n} [\mathcal{L}(R_{\hat{\theta}}) - \mathcal{L}(R^*)] \rightarrow 0$.

The proof of this theorem is given in the appendix. Conditions (A1) and (A2) are regularity conditions that are often assumed in nonparametric estimation problems. The result established in Theorem 4 shows that the learned DDRM achieves asymptotic sufficiency under the conditions (A1) and (A2) and with the specifications (N1) and (N2) for the network parameters.
There have been intensive efforts devoted to understanding the theoretical properties of deep neural network models in recent years. Several stimulating papers have studied the statistical convergence properties of nonparametric regression using neural networks [31], [32], [33]. There have also been some recent works on the non-asymptotic error bounds of GANs. For example, Liang [34] studied the rates of convergence for learning distributions implicitly with GAN under several forms of the integral probability metrics. Huang et al. [35] established nonasymptotic error bounds for GANs under a collection of integral probability metrics. Chen et al. [36] studied the convergence rates of GAN distribution estimators when both the evaluation class and the target density class are Hölder classes.

In the present problem, the objective function (9) is the combination of a loss that is a U-process indexed by a class of neural networks and a GAN-type loss indexed by two classes of neural networks. This objective function is more complicated than the least squares loss or the GAN loss analyzed in the aforementioned works. Therefore, the problem here is more difficult. To the best of our knowledge, the consistency property of the excess risk of the minimizer of estimating the discriminator is essentially that of estimating the density ratio. Therefore, in our implementation, we utilize the density-ratio estimator that most decreases the $\gamma$ of this particle method is to seek a sequence of nonlinear but simple classification or regression model using the learned representations performs better than or comparably with the optimal discriminator $\gamma^\ast$ at the current value of $\theta$ over the density of the reference distribution. The estimator $\hat{\gamma}(\mathbf{z})$ is constructed as follows. Let $Z_i = R_\theta(X_i)$ and generate $W_i \sim \gamma_{d_i}$, $i = 1, 2, \ldots, n$. We solve

$$\hat{D}_\phi \in \arg\min_{D_\phi} \frac{1}{n} \sum_{i=1}^n \left\{ \log[1 + \exp(D_\phi(Z_i))] + \log[1 + \exp(-D_\phi(W_i))] \right\},$$

(11)

with stochastic gradient descent (SGD). Then the estimated density ratio $\hat{\gamma}(\mathbf{z}) = \exp(-\hat{D}_\phi(\mathbf{z}))$. Here we note that the population version of the loss function in (11) is minimized at $-\log(r(\mathbf{z}))$. Therefore, $\hat{D}_\phi(\mathbf{z})$ in (11) provides a good estimator of $-\log(r(\mathbf{z}))$. See [38] for a detailed description of the particle approach. Here, we use this approach to transform $Z_i = R_\theta(X_i)$, $i = 1, \ldots, n$ into Gaussian samples (we still denote them as $Z_i$) directly. Once this is done, we update $\theta$ via minimizing the loss

$$\frac{1}{n} \sum_{i=1}^n \| R_\theta(X_i) - Z_i \|^2 - \lambda \hat{\gamma}_n[R_\theta(X), Y].$$

A flow chart describing the algorithm is given in the supplementary materials.

**Pseudo-Code for the DDR Algorithm:**

- **Input** \{ $X_i$, $Y_i$ \}$_{i=1}^n$.
- Tuning parameters: $s$, $\lambda$, $d$. Sample \{ $W_i$ \}$_{i=1}^n \sim \gamma_d$.
- **Outer loop for $\theta$**
  - **Inner loop (particle method)**
    * Let $Z_i = R_\theta(X_i)$, $i = 1, \ldots, n$.
    * Solve
      $$\hat{D}_\phi \in \arg\min_{D_\phi} \frac{1}{n} \sum_{i=1}^n \left\{ \log[1 + \exp(D_\phi(Z_i))] + \log[1 + \exp(-D_\phi(W_i))] \right\}.$$  
      * Define the residual map $\hat{T}(\mathbf{z}) = \mathbf{z} - s\nabla f'(\hat{r}(\mathbf{z}))$ with $\hat{r}(\mathbf{z}) = \exp(-\hat{D}_\phi(\mathbf{z}))$.
    * Update the particles $Z_i = \hat{T}(Z_i)$, $i = 1, 2, \ldots, n$.
  - **End inner loop**
    - Update $\theta$ via minimizing $-\hat{\gamma}_n[R_\theta(X), Y] + \lambda \sum_{i=1}^n \| R_\theta(X_i) - Z_i \|^2 / n$ using SGD.
  - **End outer loop**

**VI. NUMERICAL EXPERIMENTS**

We evaluate the performance of DDR using simulated and benchmark real data. Since DDR is not trying to estimate a classifier or a regression function directly, but rather to learn a representation with the desired properties of sufficiency, low-dimensionality and disentanglement, we design the experiments to evaluate the performance of the learned representations based on DDR in terms of prediction when using these representations. The results demonstrate that a simple classification or regression model using the learned representations performs better than or comparably with the best classification or regression methods using deep neural networks. Details on the network structures and hyperparameters are included in the Supplementary Material. Summary information for DDR and the compared methods, including the names of methods, their input, learning types, and models used in the methods, is given in Table II. Our experiments were conducted on a Nvidia DGX Station workstation using a single Tesla V100 GPU unit.

**A. Simulated Data**

In this subsection, we evaluate DDR on simulated regression and classification problems.

**Regression I:** We generate 10,000 data points from two models: Model (a): $Y = x_1(0.5 + (x_2 + 1.5)^2)^{-1} + (1 + x_2)^2 + \varepsilon$, where $X \sim N(0, I_2)$; Model (b): $Y = \sin^2(\pi X_1 + 1) + \varepsilon$, where $X \sim \text{Uniform}[0, 1]^2$. In both
models, \( \epsilon \sim N(0, 1) \). We use a 3-layer network with ReLU activation for \( R_\theta \) and a single hidden layer ReLU network for \( D_\phi \). We compare DDR with four prominent sufficient dimension reduction methods: sliced inverse regression (SIR) [4], sliced average variance estimation (SAVE) [39], generalized sliced inverse regression (GSIR) and generalized sliced average variance estimation (GSAVE) [9], [13]. GSIR and GSAVE are generalized versions of SIR and SAVE, respectively. Both of them estimate a central subspace in the reproducing kernel Hilbert space (RKHS) instead of using the covariance matrix in both SIR and SAVE. Also, we compare DDR with two deep learning based methods: neural networks (NN) with least square (LS) loss as the last layer, denoted as DDR with two deep learning-based methods: neural networks (NN) with least square (LS) loss as the last layer, denoted as DDR. We compare DDR with generalized sliced inverse regression (GSIR) and generalized sliced average variance estimation (GSAVE) [9], [13]. In DDR, a 4-layer network is adopted for \( R_\theta \) and a 3-layer network for \( D_\phi \) with Leaky ReLU activation.

In the given simulation, since the underlying features \( Z \) are \( X_1 \) and \( X_2 \) across all models and scenarios, we set the number of learned representation as 2. For all approaches, a linear model is fit using the learned features and the response variable, and both the prediction error and the conditional Hilbert-Schmidt independence criterion (HSIC) [41] are reported. The distance correlations between the learned representation and the response, denoted by \( dCor(\hat{Z}, Y) \), and between the learned representation and the underlying features, denoted by \( dCor(Z, Z) \), are also provided with the results shown in Table IV. Although the distance correlations between the learned representation and the underlying features are relatively lower for DDR due to the nonlinear complex transformation of the representer, we emphasize the ability to facilitate the downstream task over the reconstruction of underlying features. The representations learned with DDR exhibit higher distance correlations with the response and lower conditional HSICs, indicating that DDR can capture data information and ensure the conditional independence property more effectively than other methods. Moreover, DDR significantly surpasses GSIR and GSAVE in terms of prediction errors across all scenarios. The conditional distance correlation is also reported to assess conditional independence [42], [43], as depicted in Fig. 2. DDR demonstrates a lower or comparable conditional distance correlation compared to other methods, revealing that the representations of DDR tend to maintain the conditional independence property.

### Table II

| Method               | Name                                   | Input | Supervision | Model space       |
|----------------------|----------------------------------------|-------|-------------|-------------------|
| DDR                  | Deep Dimension Reduction               | \( X^1 \), \( Y^1 \) | Supervised | Neural networks   |
| CNN                  | Cross entropy loss with Neural Networks | \( X^1 \), \( Y^1 \) | Supervised | Neural networks   |
| dCorAE               | Distance Correlation Autoencoder       | \( X^1 \), \( Y^1 \) | Supervised | Neural networks   |
| Deep VIB             | Deep Variational Information Bottleneck| \( X^1 \), \( Y^1 \) | Supervised | Neural networks   |
| OLS                  | Ordinary Least Squares                 | \( X^1 \), \( Y^1 \) | Supervised | Linear            |
| SIR                  | Sliced Inverse Regression              | \( X^1 \), \( Y^1 \) | Supervised | Linear            |
| SAVE                 | Sliced Average Variance Estimation     | \( X^1 \), \( Y^1 \) | Supervised | Linear            |
| GSIR                 | Generalized Sliced Inverse Regression  | \( X^1 \), \( Y^1 \) | Supervised | Kernel            |
| GSAVE                | Generalized Sliced Average Variance Estimation | \( X^1 \), \( Y^1 \) | Supervised | Kernel            |
| Semi-VAB             | Semi-supervised Variational Autoencoders| \( X^1 \), \( Y^1 \) and \( X^u \) | Semi-supervised | Neural networks   |
| InfoVAB              | Information Maximizing Variational Autoencoders | \( X^1 \), \( Y^1 \) and \( X^u \) | Semi-supervised | Neural networks   |
| PCA                  | Principal Component Analysis          | \( X^1 \) and \( X^u \) | Unsupervised | Linear            |
| SPCA                 | Spare Principal Component Analysis     | \( X^1 \) and \( X^u \) | Unsupervised | Linear            |

### Table III

| Method | \( \sigma = 0.1 \) | \( \sigma = 0.4 \) | \( \sigma = 0.8 \) | \( \sigma = 0.05 \) | \( \sigma = 0.1 \) | \( \sigma = 0.2 \) |
|--------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| DDR    | 0.127 ± 0.005     | 0.555 ± 0.010     | 1.088 ± 0.009     | 0.052 ± 0.011     | 0.105 ± 0.003     | 0.241 ± 0.010     |
| NN+LS  | 0.147 ± 0.03      | 0.575 ± 0.008     | 1.150 ± 0.013     | 0.053 ± 0.001     | 0.107 ± 0.002     | 0.242 ± 0.010     |
| dCorAE | 0.125 ± 0.015     | 0.549 ± 0.012     | 1.101 ± 0.015     | 0.065 ± 0.001     | 0.135 ± 0.001     | 0.275 ± 0.004     |
| SIR    | 1.484 ± 0.047     | 1.599 ± 0.050     | 1.712 ± 0.037     | 0.252 ± 0.002     | 0.268 ± 0.002     | 0.323 ± 0.005     |
| SAVE   | 1.482 ± 0.048     | 1.588 ± 0.049     | 1.715 ± 0.038     | 0.252 ± 0.002     | 0.268 ± 0.003     | 0.323 ± 0.005     |
| GSIR   | 1.477 ± 0.047     | 1.598 ± 0.050     | 1.707 ± 0.039     | 0.267 ± 0.004     | 0.269 ± 0.004     | 0.322 ± 0.006     |
| GSAVE  | 1.478 ± 0.048     | 2.602 ± 0.079     | 2.654 ± 0.041     | 0.265 ± 0.003     | 0.267 ± 0.004     | 0.339 ± 0.006     |
### TABLE IV

DISTANCE CORRELATION BETWEEN THE LEARNED REPRESENTATION AND THE RESPONSE (dCor(\(\hat{Z}, Y\))), DISTANCE CORRELATION BETWEEN THE LEARNED REPRESENTATION AND THE UNDERLYING FEATURES (dCor(\(\hat{Z}, Z\))), CONDITIONAL HILBERT-SCHMIDT INDEPENDENCE CRITERION (HSIC), AVERAGE PREDICTION ERRORS (APE) AND THEIR STANDARD ERRORS (BASED ON 5-FOLD VALIDATION)

| Model (a) | Model (b) | Model (c) |
|-----------|-----------|-----------|
| Method    | dCor(\(\hat{Z}, Y\)) | dCor(\(\hat{Z}, Z\)) | dCor(\(\hat{Z}, Z\)) |
| **Scenario (i)** |             |             |             |
| DDR       | 0.36 ± 0.05 | 0.19 ± 0.02 | 38.49 ± 1.82 |
| GSRR      | 0.14 ± 0.04 | 0.87 ± 0.01  | 46.66 ± 1.99  |
| GSAVE     | 0.14 ± 0.03 | 0.41 ± 0.02  | 73.95 ± 2.28  |
| **Scenario (ii)** |             |             |             |
| DDR       | 0.90 ± 0.08 | 0.12 ± 0.04  | 44.63 ± 3.49  |
| GSRR      | 0.06 ± 0.02 | 0.76 ± 0.02  | 46.37 ± 1.89  |
| GSAVE     | 0.05 ± 0.03 | 0.69 ± 0.02  | 49.01 ± 1.91  |
| **Scenario (iii)** |             |             |             |
| DDR       | 0.80 ± 0.30 | 0.20 ± 0.20  | 35.39 ± 0.70  |
| GSRR      | 0.08 ± 0.01 | 0.72 ± 0.01  | 53.39 ± 3.31  |
| GSAVE     | 0.14 ± 0.02 | 0.63 ± 0.03  | 58.36 ± 3.57  |

| Model (a) | Model (b) | Model (c) |
|-----------|-----------|-----------|
| Method    | dCor(\(\hat{Z}, Y\)) | dCor(\(\hat{Z}, Z\)) | dCor(\(\hat{Z}, Z\)) |
| **Scenario (i)** |             |             |             |
| DDR       | 0.97 ± 0.01 | 0.09 ± 0.03 | 57.64 ± 3.12 |
| GSRR      | 0.09 ± 0.02 | 0.93 ± 0.00 | 18.79 ± 0.20 |
| GSAVE     | 0.10 ± 0.02 | 0.47 ± 0.03 | 17.13 ± 1.74 |
| **Scenario (ii)** |             |             |             |
| DDR       | 0.98 ± 0.03 | 0.05 ± 0.03 | 15.64 ± 0.27 |
| GSRR      | 0.12 ± 0.03 | 0.97 ± 0.01 | 28.65 ± 1.64 |
| GSAVE     | 0.26 ± 0.01 | 0.63 ± 0.02 | 32.41 ± 1.58 |
| **Scenario (iii)** |             |             |             |
| DDR       | 0.97 ± 0.00 | 0.14 ± 0.05 | 50.04 ± 6.12 |
| GSRR      | 0.19 ± 0.03 | 0.72 ± 0.01 | 111.05 ± 6.16 |
| GSAVE     | 0.35 ± 0.04 | 0.62 ± 0.02 | 128.07 ± 7.33 |

Fig. 2. The violin plot of conditional distance correlations across three scenarios.

Classification: We visualize the learned features of DDR on three simulated datasets. We first generate (1) 2-dimensional concentric circles from two classes as in Fig. 1 (a); (2) 2-dimensional moons data from two classes as in Fig. 1 (e); (3) 3-dimensional Gaussian data from six classes as in Fig. 1 (i). In each dataset, we generate 5,000 data points for each class. We next map the data into 100-dimensional space using matrices with entries i.i.d Uniform([0,1]). Finally, we apply DDR to these 100-dimensional datasets to learn 2-dimensional features. We use a 10-layer dense convolutional network (DenseNet) [44] as \(R_\theta\) and a 4-layer network as \(D_\phi\). We display the evolutions of the learned 2-dimensional features by DDR in Fig. 1. For ease of visualization, we push all the distributions onto the uniform distribution on the unit circle, which is done by normalizing the standard Gaussian random vectors to length one. Clearly, the learned features for different classes in the examples are well disentangled.

B. Real Datasets

We benchmark DDR on a variety of real datasets from both regression and classification problems. Summary information of those datasets used in the analysis is given in Table V.

Regression: We benchmark the prediction performance of DDR using the representations learned based on DDR. Here, we use the YearPredictionMSD dataset\(^1\) and the Pole-Telecommunication dataset.\(^2\) The YearPredictionMSD dataset contains 515,345 observations with 90 predictors. The problem is to predict the year of song release. The Pole-Telecommunication dataset consists of 15,000 observations with 48 predictors for determining the placement of antennas. We randomly split the data into five folds to evaluate the prediction performance using 5-fold cross validation. We employ a 3-layer network for both \(D_\phi\) and \(R_\theta\) on the YearPredictionMSD dataset; a 2-layer network for \(D_\phi\) and a 4-layer network \(R_\theta\) are adopted on the Pole-Telecommunication dataset. In comparison, we conduct a nonlinear regression using neural networks (NN) with a least squares (LS) loss in the last layer, denoted as NN+LS. That is, we do not impose any desired characteristics for the learned representations in the pultimate layer. Note that, for both DDR and NN+LS, we use the same networks to learn representative features. We also consider the popular dimension reduction methods, including principal component analysis (PCA) and sparse principal component analysis (SPCA), to obtain data representation. For the comparison with supervised dimension reduction methods, we consider SIR and SAVE and the deep learning based sufficient dimension reduction method dCorAE. For those methods, we first obtain the estimated representative features and fit a linear regression model of the response on the learned representations. The average prediction errors and their standard errors based on DDR, NN+LS, dCorAE, PCA, SPCA, SIR, SAVE and the ordinary least squares (OLS) regression with the original data are reported in Tables VI and VII. DDR outperforms other methods in terms of prediction accuracy.

---

\(^1\)The YearPredictionMSD dataset is available at https://archive.ics.uci.edu/ml/datasets/YearPredictionMSD.

\(^2\)The Pole-Telecommunication dataset is available at https://www.dcc.fc.up.pt/ltorgo/Regression/DataSets.html.
loss, denoted as DDR+CN, by applying the transfer learning technique [50], [51], [52] on CIFAR-10 and CIFAR-100. The pretrained WideResnet-101 model [53] on the ImageNet dataset with Spinal FC [54] is chosen for \( R_\theta \). In our experiments for transfer learning, we first train the WideResnet model on ImageNet. We then use the parameters of the pretrained neural network as the initialization parameters to train CIFAR-10 and CIFAR-100. In contrast to transfer learning, the initialization parameters of learning from scratch are random. The discriminator network \( D_\phi \) is a 4-layer network. The architecture of \( R_\theta \) and most hyperparameters are shared across all four methods - DDR, CNN, DDR+CN and dCorAE [40]. Finally, we use the \( k \)-nearest neighbor (\( k = 5 \)) classifier on the learned features for all methods.

As shown in Table VIII, the classification accuracies of DDR for MNIST and FashionMNIST are better than or comparable with those of CNN, dCorAE and Deep VIB. As shown in Table IX, the classification accuracy of DDR using the CN loss outperforms that of CNN on CIFAR-10 and CIFAR-100. We also calculate the estimated distance correlation (DC) between the learned features and their labels. Fig. 3 shows the values of DC for MNIST, FashionMNIST and CIFAR-10 data. Higher DC values mean that the learned features are of higher quality. DDR and DDR+CN achieves higher DC values.

Because both GSIR and GSAVE require computation of \( n \times n \) kernel matrices, which is computationally prohibitive when \( n = 10,000 \) to 60,000 and \( p \approx 1,000 \), it does not allow us to apply both methods to analyze datasets from MNIST, FashionMNIST, CIFAR-10 and CIFAR-100.

**Classification II:** To compare the performance of DDR with semi-supervised methods, we benchmark DDR on MNIST dataset with varying amounts of labeled data for training. In detail, we consider some widely used semi-supervised learning methods, including semi-supervised variational autoencoders (Semi-VAE) [55] and information maximizing variational autoencoders (InfoVAE) with the semi-supervised setting [55], [56], and the supervised method, CNN. InfoVAE utilizes all training images to learn representations, and then trains the \( k \)-nearest neighbor (\( k = 5 \)) classifier with the learned representations and partially known labels. All four methods share the same network architecture for 50-dimensional learning representation. We adopt the double-hidden-layer MLP networks, with 600 neurons for each layer, the softplus activation function, and the Adam optimizer. For both Semi-VAE and InfoVAE, we apply a semi-supervised setting to analyze all 60\( k \) images with the varying number of labeled images as the training data and validate the

**Classification I:** We conduct a comprehensive benchmarking analysis to evaluate the classification performance of DDR using various datasets such as MNIST [45], FashionMNIST [46], CIFAR-10, and CIFAR-100 [47]. The comparison is performed against several existing methodologies, including Convolutional Neural Networks (CNN) utilizing cross entropy (CN) loss as the terminal layer, denoted as CNN, Distance Correlation Autoencoder (dCorAE) [40], and Deep Variational Information Bottleneck (Deep VIB) [48], [49]. Specifically, for the CNN approach, we employ the feature extractor by omitting the final layer responsible for CN loss within the neural network trained for classification. It is important to emphasize that identical network architectures are implemented for both DDR and other methods in the process of learning representations.

The MNIST and FashionMNIST datasets consist of 60\( k \) and 10\( k \) grayscale images with \( 28 \times 28 \) pixels for training and testing, respectively, while the CIFAR-10 and CIFAR-100 datasets contain 50\( k \) and 10\( k \) colored images with \( 32 \times 32 \) pixels for training and testing, respectively. The representer network \( R_\theta \) contains 20 layers for MNIST data and 100 layers for FashionMNIST and CIFAR-10 data.

To fully utilize computational resources and improve classification accuracy, we further combine DDR with the CN
TABLE VIII
CLASSIFICATION ACCURACY FOR MNIST AND FASHIONMNIST DATASETS

|      | MNIST | FashionMNIST |
|------|-------|--------------|
|      | DDR   | dCorAE | CNN | Deep VB | DDR   | dCorAE | CNN | Deep VB |
| \(d = 16\) | 99.41 | 99.58 | 99.39 | 98.97 | \textbf{94.44} | 94.18 | 94.21 | 91.27 |
| \(d = 32\) | 99.61 | 99.54 | 99.45 | 98.79 | 94.18 | 93.89 | 94.41 | 91.13 |
| \(d = 64\) | 99.56 | 99.53 | 99.49 | 98.65 | 94.13 | 94.24 | 94.38 | 91.01 |

TABLE IX
CLASSIFICATION ACCURACY FOR CIFAR-10 AND CIFAR-100 DATASETS

|      | CIFAR-10 | CIFAR 100 |
|------|----------|-----------|
|      | Learning from scratch | Transfer learning | d | Transfer learning |
| \(d = 16\) | dCorAE | CNN | DDR | CNN | DDR | DDR+CN | d | dCorAE | CNN | DDR | DDR+CN |
| \(d = 16\) | 94.15 | 94.21 | \textbf{94.29} | 97.44 | 97.52 | \textbf{97.68} | \(d = 200\) | 85.39 | 86.29 | \textbf{86.36} |
| \(d = 32\) | 94.18 | 94.92 | 94.58 | 97.79 | 97.53 | \textbf{97.96} | \(d = 400\) | 85.57 | 85.95 | \textbf{86.04} |
| \(d = 64\) | 94.66 | 95.09 | 94.46 | 97.90 | 97.49 | \textbf{97.91} | \(d = 400\) | 85.55 | 86.21 | \textbf{86.30} |

Fig. 3. The distance correlations of labels with learned features based on DDR, CNN and dCorAE with \(d = 16, 32 \text{ and } 64\) for MNIST, FashionMNIST (FMNIST) and CIFAR-10 (CIFAR) data (FS: from scratch; TL: transfer learning).

Fig. 4. Comparison of classification errors on varying amounts of labeled MNIST data.

Guidance of Choosing \(d\): Although our experimental results demonstrate that DDR exhibits robustness to varying choices of dimensionality \(d\), we acknowledge the significance of providing guidance on this crucial aspect. First, recognizing that the dimensionality \(d\) serves as a hyperparameter, we advocate employing Cross-Validation to determine the optimal value, in alignment with specific target metrics, such as predictive errors. Second, for classification tasks, we recommend selecting a dimensionality multiple times greater than the number of categories, ensuring that sufficient information is captured for precise classification. For regression problems, we advise choosing \(d\) within a range from 2 to 100, tailored to the particular nature and requirements of the task at hand. These guidelines, derived from our in-depth understanding and experience, are intended to support practitioners in effectively utilizing DDR across diverse applications and scenarios.

VII. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a nonparametric DDR approach to achieving a good data representation for supervised learning with certain desired characteristics including sufficiency, low-dimensionality and disentanglement. We estimate the representation map nonparametrically by taking the best by fully utilizing all 60k training images. However, the accuracy of Semi-VAE does not improve over the increasing proportion of images with labels. This is because the objective function of Semi-VAE is primary to maximize the lower bound of the joint likelihood rather than the classification loss. In all, compared with semi-supervised learning, DDR uses a small amount of data with labels in training, but achieves better classification accuracy when the number of labeled images is greater than 1,000.

For supervised methods, we observe that DDR outperforms CNN for the varying number of images with labels. When the proportion of images with labels is low, Semi-VAE performs performance using 10k test data. But for both DDR and CNN, we apply a supervised setting to analyze only data with labels and discard the rest of the images in the training set.

The classification errors using images with a varying sizes of labeled data from 600 to 10,000 are shown in Fig. 4. For supervised methods, we observe that DDR outperforms CNN for the varying number of images with labels. When the proportion of images with labels is low, Semi-VAE performs
advantage of the powerful capabilities of deep neural networks in approximating multi-dimensional functions.

Several questions deserve further study. First, it would be interesting to consider other measures of conditional independence such as conditional covariance operators on reproducing kernel Hilbert spaces [5] and heteroscedastic conditional variance operator on Hilbert spaces [9]. It is also possible to use mutual information for measuring conditional independence [6], although with this measure the loss function itself needs to be estimated.

We used the standard Gaussian as the reference distribution for DDRM to promote disentanglement of the representation. Another convenient choice is the uniform distribution on the unit cube. It is worth examining whether there is any difference in the performance of DDR with different reference distributions. Another question is how to determine the dimension of the learned representation. This is also an important problem in linear SDR. Since the purpose of dimension reduction is often to build prediction models in high-dimensional settings, the problem of determining the dimension of the representation can be best addressed based on cross validation or related data-driven methods in the model building phase.

Finally, we note that even in linear SDR [4], [16], there does not exist a unique linear sufficient dimension-reduction map. In this appendix, we prove Lemmas 1 and 2, and Theorems 3 and 4.

A. Proof of Lemma 1

Proof: By assumption \( \mu \) and \( \gamma_d \) are both absolutely continuous with respect to the Lebesgue measure. The desired result holds since it is a special case of the well-known results on the existence of optimal transport [57], [58], see also Theorem 1.28 on page 24 of [59] for details.

B. Proof of Lemma 2

Proof: Our proof follows [25]. Since \( f(t) \) is convex, then for all \( t \in \mathbb{R} \), we have \( f(t) = f^*(t) \), where

\[
  f^*(t) = \sup_{s \in \mathbb{R}} \{ st - f^*(s) \}
\]

is the Fenchel conjugate of \( f^* \). By Fermat’s rule, the maximizer \( s^* \) satisfies

\[
  t \in \partial f^*(s^*),
\]

i.e.,

\[
  s^* \in \partial f(t)
\]

Plugging the above display with \( t = \frac{d\mu}{dx}(x) \) into the definition of \( f \)-divergence, we derive (6).

C. Proof of Theorem 3

Proof: Without loss of generality, we assume \( d = 1 \). For \( R^* \) satisfying (3) and any \( R \in \mathcal{R} \), we have \( R = \rho(R,R^*)R^* + \varepsilon_R \), where \( \rho(R,R^*) \) is the correlation coefficient between \( R \) and \( R^* \), \( \varepsilon_R = R - \rho(R,R^*)R^* \). It is easy to see that \( \varepsilon_R \perp R^* \) and thus \( Y \perp \varepsilon_R \). As \( \langle \rho(R,R^*)R^*, Y \rangle \) is independent of \( (\varepsilon_R, 0) \), then by Theorem 3 of [60]

\[
  \mathbb{V}[R, y] = \mathbb{V}[\rho(R,R^*)R^* + \varepsilon_R, y] = \mathbb{V}[\rho(R,R^*)R^*, y] = \mathbb{V}[\varepsilon_R, y] \leq \mathbb{V}[\varepsilon_R, y].
\]

As \( R(x) \sim \mathcal{N}(0, 1) \) and \( R^*(x) \sim \mathcal{N}(0, 1) \), then \( \mathbb{D}_f(\mu_{R(x)} || \gamma_d) = \mathbb{D}_f(\mu_{R^*(x)} || \gamma_d) = 0 \), and

\[
  \mathcal{L}(R) - \mathcal{L}(R^*) = \mathbb{V}[R*, y] - \mathbb{V}[R, y] \geq 0.
\]

The proof is completed.

D. Proof of Theorem 4

Recall that \( B_2 = \max\{|f'(c_1)|, |f'(c_2)|\} \), \( B_3 = \max_{|s| \leq 2B_2} |f^*(s)| \). We set the network parameters of the representer \( \tilde{R}_0 \) and the discriminator \( \tilde{d}_R \) as follows.

(N1) Representer network \( R \equiv R\mathcal{H}_{\mathcal{W},S} \) parameters: depth \( \mathcal{H} = o(\log n) \) with \( \mathcal{W} = O(n^{-\frac{1}{m-1}}/\log n) \), size \( \mathcal{S} = O((d+n^{-\frac{1}{m-1}})/\log^4(npd)) \), and \( \|R\|_{L^\infty} \leq B = 2\|\mathcal{R}^*\|_{L^\infty}, \forall R \in \mathcal{R} \).

(N2) Discriminator network \( D \equiv D\mathcal{H}_{\mathcal{W},\tilde{S}} \) parameters: depth \( \tilde{\mathcal{H}} = O(\log n) \), width \( \tilde{\mathcal{W}} = O(n^{-\frac{1}{m-1}}/\log n) \), size \( \tilde{\mathcal{S}} = O((d+n^{-\frac{1}{m-1}})/\log^4(npd)) \), and \( \|D\|_{L^\infty} \leq 2B_2, \forall D \in \mathcal{D} \).

Before getting into the details of the proof of Theorem 4, we first give an outline of the basic structure of the proof.

Without loss of generality, we assume that \( \lambda = 1 \) and \( m = 1 \), i.e., \( y \in \mathbb{R} \). For any \( R \in \mathcal{R}_{\mathcal{H},\mathcal{W},S} \), we have:

\[
  \mathcal{L}(\tilde{R}_0) - \mathcal{L}(R^*) = \mathcal{L}(\tilde{R}_0) - \tilde{\mathcal{L}}(\tilde{R}_0) + \tilde{\mathcal{L}}(\tilde{R}_0) - \mathcal{L}(\tilde{R})
  \leq \tilde{\mathcal{L}}(\tilde{R}) - \mathcal{L}(\tilde{R}) + \mathcal{L}(\tilde{R}) - \mathcal{L}(R^*)
  \leq 2 \sup_{R \in \mathcal{R}_{\mathcal{H},\mathcal{W},S}} |\mathcal{L}(R) - \tilde{\mathcal{L}}(R)|
  + \inf_{R \in \mathcal{R}_{\mathcal{H},\mathcal{W},S}} |\mathcal{L}(R) - \mathcal{L}(R^*)|,
\]

where we use the definition of \( \tilde{R}_0 \) in (9) and the feasibility of \( \tilde{R} \). Next we bound the two error terms.

- the approximation error: \( \inf_{R \in \mathcal{R}_{\mathcal{H},\mathcal{W},S}} |\mathcal{L}(R) - \mathcal{L}(R^*)| \);
- the statistical error: \( \sup_{R \in \mathcal{R}_{\mathcal{H},\mathcal{W},S}} |\mathcal{L}(R) - \tilde{\mathcal{L}}(R)| \).

Then Theorem 4 follows after bounding these two error terms.

1) The Approximation Error:

Lemma 5: Suppose that (A1)-(A3) hold and the network parameters satisfy (N1) and (N2). Then,

\[
  \inf_{\tilde{R} \in \mathcal{R}_{\mathcal{D},\mathcal{W},S}} |\mathcal{L}(\tilde{R}) - \mathcal{L}(R^*)| \leq 320C_1 L_1 B_1 \sqrt{p(d+1)^{-1/2}} + o(1).
\]

as \( n \to \infty \).
Proof: By (3) and (6) and the definition of $\mathcal{L}$, we have
\[
\inf_{\bar{R} \in \mathcal{R}_{D,W,S}} |\mathcal{L}(\bar{R}) - \mathcal{L}(R^*)| \leq |D_f(\mu_{\bar{R}_0}(x))| \gamma_d | + \frac{1}{2} |\mathbb{E}[\|R^*(x) - \bar{R}_0(x)\| - \|\bar{R}_0(x) - \bar{R}_0(x)\|]|y_1 - y_2|\ |
\]
where $\bar{R}_0 \in \mathcal{R}_{D,W,S}$ is specified in Lemma 7 below. We finish the proof by (16) in Lemma 8 and (17) in Lemma 9, which will be proved below.

Lemma 6: For any function $f : [-B, B]^p \rightarrow \mathbb{R}$ with Lipschitz constant $L$ there exist a ReLU network $\bar{f}$ with depth $O(12L + C_1p)$ and width $O(C_2pW)$ such that
\[
\|f - \bar{f}\|_{L\infty} \leq 19L\sqrt{\beta}B(\mathcal{H}W)^{-2/p},
\]
where $C_1p = 14 + 2p$, $C_2p = 3p^3$.
Proof: This Lemma follows directly from Theorem 1.1 of [61].

Lemma 7: Suppose that (A1) and (A3) hold and the network parameters satisfy (N1). Then, there exists a $\bar{R}_0 \in \mathcal{R}_{H,W,S}$ with the network parameters satisfying (N1) such that
\[
\|\bar{R}_0 - R^*\|_{L^2(p)} \leq 19L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-2/p}.
\]
Proof: Let $R^*_i(x)$ be the $i$-th entry of $R^*(x) : \mathbb{R}^d \rightarrow \mathbb{R}^d$. By the assumption on $R^*$, it is easy to check that $R^*_i(x)$ is Lipschitz continuous on $[-B_1, B_1]^d$ with the Lipschitz constant $L_1$. By Lemma 6, there exists a ReLU network $\bar{R}_0$, with depth $O(\mathcal{H})$ and width $O(\mathcal{W})$ such that
\[
\|\bar{R}_0 - R^*_i\|_{L^2(p)} \leq 19L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-2/p}.
\]
Then
\[
\|\bar{R}_0 - R^*\|_{L^2(p)} = \left(\sum_{i=1}^n (R^*_i(x) - \bar{R}_0(x))^2 dx\right)^{1/2} \leq \left(\int (R^*_i(x) - \bar{R}_0(x))^2 dx\right)^{1/2} \leq 19L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-2/p}.
\]
Define $\bar{R}_0 = [\bar{R}_0, \ldots, \bar{R}_0] \in \mathcal{R}_{H,W,S}$. The above three display implies
\[
\|\bar{R}_0 - R^*\|_{L^2(p)} \leq 19L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-2/p} \leq 19L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-2(p+2)/p}.
\]
where in the last inequality we use the choice of $\mathcal{H}$ and $\mathcal{W}$ in (N1).

Lemma 8: Suppose that (A1) and (A3) hold and the network parameters satisfy (N1). Then,
\[
\mathbb{V}[R^*(x), y] - \mathbb{V}[\bar{R}_0(x), y] \leq 320C_1L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-1(p+2)},
\]
Proof: Recall that [15]
\[
\mathbb{V}[z, y] = \mathbb{E}[\|z_1 - z_2\||y_1 - y_2|| + 2\mathbb{E}[\|z_1 - z_2\||y_1 - y_3|| + \mathbb{E}[\|z_1 - z_2\||y_1 - y_2]],
\]
where $(z_i, y_i), i = 1, 2, 3$ are i.i.d. copies of $(z, y)$. We have
\[
\mathbb{V}[R^*(x), y] - \mathbb{V}[\bar{R}_0(x), y] \leq 320C_1L_1B_1\sqrt{\beta}B(\mathcal{H}W)^{-1(p+2)}.
\]

2) The Statistical Error:

Lemma 9: Suppose that (A1) and (A2) hold and the network parameters satisfy (N1). Then,
\[
\mathbb{D}_f(\mu_{\bar{R}_0}(x)) |y_\gamma_d| \rightarrow 0,
\]
as $n \rightarrow \infty$.
Proof: By Lemma 7 $\bar{R}_0$ can approximate $R^*$ arbitrarily well as $n \rightarrow \infty$, the desired result follows from the fact that
\[
\mathbb{D}_f(\mu_{R^*(x)}) |y_\gamma_d| = 0
\]
and the continuity of $\mathbb{D}_f(\mu_{R^*(x)}) |y_\gamma_d|$ on $R$. We present the sketch of the proof and omit the details here. Let $r^*_n(z) = \frac{d\mu_{R^*(x)}}{dy_\gamma_d}(z)$ and $\bar{r}(z) = \frac{d\mu_{\bar{R}_0}(x)}{dy_\gamma_d}(z)$. By definition we have
\[
\mathbb{D}_f(\mu_{R^*(x)}) |y_\gamma_d| = \mathbb{E}_{W \sim \gamma_d}[f(r^*_n(W))].
\]
We can represent $\mathbb{D}_f(\mu_{\bar{R}_0}(x)) |y_\gamma_d|$ similarly. Therefore,
\[
\mathbb{D}_f(\mu_{\bar{R}_0}(x)) |y_\gamma_d| = \mathbb{D}_f(\mu_{\bar{R}_0}(x)) |y_\gamma_d| - \mathbb{D}_f(\mu_{R^*(x)})) |y_\gamma_d| < \mathbb{E}_{W \sim \gamma_d}[f(r^*_n(W)) - f(\bar{r}(W))] \leq \int |f'(\bar{r}(z))||r^*_n(z) - \bar{r}(z)||dy_\gamma_d(z) \leq B_2 \int |r^*_n(z) - \bar{r}(z)||dy_\gamma_d(z),
\]
where the second inequality we use mean value theorem and boundness assumption on $f'(\bar{r})$ in (A2). Then last inequality goes to zero due to continuity and the fact $\bar{R}_0$ converge to $R^*$ as $n \rightarrow \infty$ by Lemma 7.

Proof: By the definition and the triangle inequality we have
\[
\mathbb{D}_f(\mu_{\bar{R}_0}(x)) |y_\gamma_d| \leq C_{19}(2B_2 + B_3)n^{-\frac{1}{p+2}} + 19(1 + B_3)L_2\sqrt{\beta} \log n n^{-\frac{1}{p+2}} + 4C_6C_7C_{10}Bn^{-\frac{1}{p+2}}.
\]

We finish the proof based on (19) in Lemma 11 and (24) in Lemma 12, which will be proved below.

Lemma 10: Suppose that (A1)-(A2) hold and the network parameters satisfy (N1) and (N2). Then,
\[
\mathbb{E}_{R \sim \mathcal{R}_{H,W,S}}[|\mathcal{L}(R) - \mathcal{L}(\bar{R})|] \leq C_{13}(2B_2 + B_3)n^{-\frac{1}{p+2}} + 19(1 + B_3)L_2\sqrt{\beta} \log n n^{-\frac{1}{p+2}} + 4C_6C_7C_{10}Bn^{-\frac{1}{p+2}}.
\]

Proof: By the definition and the triangle inequality we have
\[
\mathbb{E}_{R \sim \mathcal{R}_{H,W,S}}[|\mathcal{L}(R) - \mathcal{L}(\bar{R})|] \leq \mathbb{E}_{R \sim \mathcal{R}_{H,W,S}}[\mathbb{V}[R(x), y] - \mathbb{V}[\bar{R}(x), y]] + \mathbb{E}_{R \sim \mathcal{R}_{H,W,S}}[\mathbb{D}_f(\mu_{R^*(x)}) |y_\gamma_d| - \mathbb{D}_f(\mu_{\bar{R}_0}(x)) |y_\gamma_d|].
\]
We finish the proof based on (19) in Lemma 11 and (24) in Lemma 12, which will be proved below.

Proof: By the definition and the triangle inequality we have
\[
\mathbb{E}_{R \sim \mathcal{R}_{H,W,S}}[\mathbb{V}[R(x), y] - \mathbb{V}[\bar{R}(x), y]] \leq 4C_6C_7C_{10}Bn^{-\frac{1}{p+2}}.
\]

Proof: By the definition and the triangle inequality we have
Proof: We first fix some notation for simplicity. Denote $O = (x, y) \in \mathbb{R}^p \times \mathbb{R}^1$ and $O_i = (x_i, y_i), i = 1, \ldots, n$ are i.i.d copy of $O$, and denote $\mu_{xy}$ and $\mathbb{P}^n$ as $\mathbb{P}$ and $\mathbb{P}^n$, respectively. $\forall R \in \mathbb{R}^{H_{W,S}}$, let $\hat{O} = (R(x), y)$ and $\hat{O}_i = (R(x_i), y_i), i = 1, \ldots, n$ be i.i.d copy of $\hat{O}$. Define centered kernel $\bar{h}_R: (\mathbb{R}^p \times \mathbb{R}^1)^4 \to \mathbb{R}$ as

$$
\bar{h}_R(\hat{O}_1, \hat{O}_2, \hat{O}_3, \hat{O}_4) = \frac{1}{4} \sum_{1 \leq i < j \leq 4} \|R(x_i) - R(x_j)\| y_i - y_j
$$

Then, the centered $U$-statistics $\hat{V}_n[R(x), y] - V[R(x), y]$ can be represented as

$$
\|U_n(\bar{h}_R)\| = \frac{1}{C^n_{4^n}} \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq n} \bar{h}_R(\hat{O}_{i_1}, \hat{O}_{i_2}, \hat{O}_{i_3}, \hat{O}_{i_4}).
$$

Our goal is to bound the supremum of the centered $U$-process $U_n(\bar{h}_R)$ with the nondegenerate kernel $\bar{h}_R$. By the symmetrization randomization Theorem 3.5.3 in [62], we have

$$
\mathbb{E}\left[ \sup_{R \in \mathbb{R}^{H_{W,S}}} |U_n(\bar{h}_R)| \right] \leq C_3 \mathbb{E}\left[ \sup_{R \in \mathbb{R}^{H_{W,S}}} \frac{1}{C^n_{4^n}} \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq n} \epsilon_{i_1} \epsilon_{i_2} \epsilon_{i_3} \epsilon_{i_4} \bar{h}_R(\hat{O}_{i_1}, \hat{O}_{i_2}, \hat{O}_{i_3}, \hat{O}_{i_4}) \right],
$$

where, $\epsilon_{i_1}, \epsilon_{i_2}, \epsilon_{i_3}, \epsilon_{i_4}, i = 1, \ldots, n$ are i.i.d Rademacher variables that are also independent with $O_i, i = 1, \ldots, n$. We finish the proof by upper bounding the above Rademacher process with the metric entropy of $R \in \mathbb{R}^{H_{W,S}}$. To this end we need the following lemma.

**Lemma 12:** If $\xi_{i}, i = 1, \ldots, m$ are $m$ finite linear combinations of Rademacher variables $\epsilon_{j}, j = 1, \ldots, J$. Then

$$
\mathbb{E}_{\epsilon_{j}, j = 1, \ldots, J} \max_{1 \leq i \leq m} |\xi_{i}| \leq C_6 (\log m)^{1/2} \max_{1 \leq i \leq m} (\mathbb{E}_{\epsilon_{i}}^2)^{1/2}.
$$

(22)

Proof: This result follows directly from Corollary 3.2.6 and inequality (4.3.1) in [62] with $\Phi(x) = \exp(x^2)$.

By Lemma 7, we can assume the boundness of $R \in \mathbb{R}^{H_{W,S}}$, i.e., we can assume $\|R\|_{L^\infty} \leq B = 2^r \|R\|_{L^\infty}$ as $n$ large enough. Then by the boundedness assumption on $y$, we have that the kernel $\bar{h}_R$ is also bounded, say

$$
\|\bar{h}_R\|_{L^\infty} \leq C_7 B.
$$

(23)

$\forall R \in \mathbb{R}^{H_{W,S}}$, define a random empirical measure (depends on $\hat{O}_i, i = 1, \ldots, n$)

$$
e_{n,1}(R, \hat{R}) = \frac{1}{n} \sum_{i=1}^{n} \epsilon_{i} (\bar{h}_R(\hat{O}_i, \hat{O}_i, \hat{O}_i, \hat{O}_i)).
$$

Condition on $O_i, i = 1, \ldots, n$, let $C(R, \epsilon_{n,1}, \delta)$ be the covering number of $R_{H_{W,S}}$ with respect to the empirical distance $\epsilon_{n,1}$ at scale of $\delta > 0$. Denote $R_k$ as the covering set of $R_{H_{W,S}}$ with cardinality of $C(R, \epsilon_{n,1}, \delta)$. Then,

$$
\mathbb{E}_{\epsilon_{i}, i=1,\ldots,n} \left[ \sup_{R \in R_k} \frac{1}{C^n_{4^n}} \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq n} \epsilon_{i_1} \epsilon_{i_2} \epsilon_{i_3} \epsilon_{i_4} \bar{h}_R(\hat{O}_{i_1}, \hat{O}_{i_2}, \hat{O}_{i_3}, \hat{O}_{i_4}) \right]
\leq \delta + C_6 \frac{1}{C^n_{4^n}} \left( \log C(R, \epsilon_{n,1}, \delta) \right)^{1/2}
$$

where the first inequality follows from the triangle inequality, the second inequality uses (22), the third and fourth inequalities follow after some algebra, and the fifth inequality holds since $C(R, \epsilon_{n,1}, \delta) \leq C(R, \epsilon_{n,\infty}, \delta)$ and the relationship between the metric entropy and the VC-dimension of the ReLU networks $R_{H_{W,S}}$ [63], i.e.,

$$
\log C(R, \epsilon_{n,\infty}, \delta) \leq VC_R \log \frac{2 \text{Bn}}{\delta VC_R} + \frac{2 \text{Bn}}{\delta VC_R},
$$

and the last inequality holds due to the upper bound of VC-dimension for the ReLU network $\mathbb{R}^{H_{W,S}}$ satisfying

$$
C_B \text{HS log S} \leq VC_R \leq C_B \text{HS log S},
$$

see [64]. Then (19) holds by the selection of the network parameters and set $\delta = \frac{1}{2}$ and some algebra.

**Lemma 13:** Suppose that (A1)-(A3) hold and the network parameters satisfy (N1) and (N2). Then,

$$
\mathbb{E}\left[ \sup_{R \in \mathbb{R}^{H_{W,S}}} |\hat{D}_f(\mu_{R(x)}(|\gamma_d) - D_f(\mu_{R(x)}(|\gamma_d))\right]
\leq C_{14} (L_2 \sqrt{d} + B_2 + B_3) (n^{\frac{3}{2}} n^\frac{3}{2} + \log mn^{-\frac{3}{2}}).
$$

(24)

Proof: For every $R \in \mathbb{R}^{H_{W,S}}$, let $r(z) = \frac{d \mu_{R(x)}}{dy_{R(x)}}(z)$, $g_R(z) = f'(r(z))$. By assumption $g_R(z) < \infty$ and $z \sim \mu_{R(x)}$, hence $\text{supp}(g_R) \subseteq [-\log n, \log n]^d$. Then, by Lemma 6 there exists a $D_\phi \in \mathbb{R}^{H_{W,S}}$ with the network parameters satisfying (N2) such that for $z \sim \gamma$ and $z \sim \mu_{R(x)}$,

$$
\mathbb{E}_{\mu}[|\hat{D}_\phi(z) - g_R(z)|] \leq 19L_2 \sqrt{d} \log mn^{-\frac{3}{2}}.
$$

(25)
By the above display, we can further assume that the element in \( E_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \) is bounded by \( 2B_2 \) as \( n \) large enough. For any \( g : \mathbb{R}^d \to \mathbb{R} \), define
\[
\mathcal{E}(g) = \mathbb{E}_{x \sim \nu_{\mu_0}}[g(R(x))] - \mathbb{E}_{W \sim \gamma_d} f^*(g(W)),
\]
\[
\tilde{\mathcal{E}}(g) = \tilde{\mathcal{E}}(g, R) = \frac{1}{n} \sum_{i=1}^n [g(R(x_i)) - f^*(g(W_i))].
\]
By (6) we have
\[
\mathcal{E}(g_R) = \mathcal{D}_f(\mu_R | \gamma_d) = \sup_{D, R : D \text{ measurable}} \mathcal{E}(D). \tag{26}
\]
Then,
\[
|D_f(\mu_R | \gamma_d) - \tilde{D}_f(\mu_R | \gamma_d)| \\
\leq |\mathcal{E}(g_R) - \tilde{\mathcal{E}}(g_R)| + \sup_{D, R : D \text{ measurable}} |\mathcal{E}(D) - \tilde{\mathcal{E}}(D)|.
\]
where we use the triangle inequality in the first inequality follows from the triangle inequality, the second inequality follows from \( \mathcal{E}(g_R) \geq \sup_{D, R : D \text{ measurable}} \mathcal{E}(D) \) due to (26) and the triangle inequality, the third inequality follows from the triangle inequality, and the last inequality follows from (25) and the mean value theorem.

We finish the proof by bounding the second term in probability in the last line above, i.e., \( \sup_{D, R : D \text{ measurable}} |\mathcal{E}(D) - \tilde{\mathcal{E}}(D)| \). This can be done by bounding the empirical process
\[
\mathbb{P}(D, R) = \mathbb{E}_R \sup_{D, R : D \text{ measurable}} |\mathcal{E}(D) - \tilde{\mathcal{E}}(D)|.
\]
Let \( S = (x, z) \sim \mu_x \otimes \gamma_d \) and \( S_i, i = 1, \ldots, n \) be \( n \) i.i.d. copy of \( S \). Denote
\[
b(D, R; S) = D(R(x)) - f^*(D(z)).
\]
Then
\[
\mathcal{E}(D, R) = \mathbb{E}_S[b(D, R; S)]
\]
and
\[
\tilde{\mathcal{E}}(D, R) = \frac{1}{n} \sum_{i=1}^n b(D, R; S_i).
\]
Let
\[
g(D \times R) = \frac{1}{n} \sum_{i=1}^n b(D, R; S_i)
\]
be the Rademacher complexity of \( E_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \times \mathcal{R}_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \) \([65]\).

Then \( \mathcal{E}(g_R) \) is bounded by \( R_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \) with respect to the empirical distance (depends on \( S_i \))
\[
d_{n,1}((D, R), (\tilde{D}, \tilde{R})) = \frac{1}{n} \sum_{i=1}^n |b_i(D, R; S_i) - b_i(\tilde{D}, \tilde{R}; S_i)|
\]
at scale of \( \delta > 0 \). Let \( D_\Phi \times R_\delta \) be such a converging set of \( D_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \times \mathcal{R}_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \). Then,
\[
\mathbb{P}(D, R) = 2g(D \times R)
\]
\[
= 2E_{S_1, \ldots, S_n} \left[ |\mathcal{E}(D, R) - \mathcal{E}(\tilde{D}, \tilde{R})| \right].
\]

\[
\leq 2\delta + \frac{2}{n} \sum_{i=1}^n |E_{S_i, i=1, \ldots, n} \left[ \mathcal{E}(D, R) - \mathcal{E}(\tilde{D}, \tilde{R}) \right] |
\]

where the first equality follows from the standard symmetrization technique, the first inequality holds due to the iteration law of conditional expectation, the second inequality follows from the triangle inequality, and the third inequality uses (22), the fourth inequality uses the fact that \( b(D, R; S) \) is bounded, i.e., \( |b(D, R; S)| \leq 2B_2 + B_3 \), and the fifth inequality follows from some algebra, and the sixth inequality follows from \( \mathcal{C}(R, \nu_{\mu_0}, \delta) \leq \mathcal{C}(R, \nu_{\mu_0}, \delta) \) (similar result for \( D \) and \( \mathcal{C}(R, \nu_{\mu_0}, \delta) \) \( \leq \mathcal{C}(R, \nu_{\mu_0}, \delta) \)) \([64]\). Then (24) follows from the above display with the selection of the network parameters of \( D_{\mathcal{H}, \mathcal{W}, \mathcal{S}}, R_{\mathcal{H}, \mathcal{W}, \mathcal{S}} \) and with \( \delta = \frac{1}{n} \).

Finally, Theorem 4 is a direct consequence of (12) in Lemma 5 and (18) in Lemma 10. This completes the proof of Theorem 4.

\section*{Acknowledgment}

The numerical calculations for this study were conducted at the Supercomputing Center, Wuhan University.

\section*{References}

[1] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 433–444, 2015.

[3] G. Alain and Y. Bengio, “Understanding intermediate layers using linear classifier probes,” 2016, arXiv:1610.01644.

[4] K.-C. Li, “Sliced inverse regression for dimension reduction,” J. Amer. Stat. Assoc., vol. 86, no. 414, pp. 316–327, Jun. 1991.

[5] K. Fukumizu, F. R. Bach, and M. I. Jordan, “Kernel dimension reduction in regression,” Ann. Statist., vol. 37, no. 4, pp. 1871–1905, Aug. 2009.

[6] T. Suzuki and M. Sugiyama, “Sufficient dimension reduction via squared-loss mutual information estimation,” Neural Comput., vol. 25, no. 3, pp. 725–758, Mar. 2013.
[64] P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian, “Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks,” J. Mach. Learn. Res., vol. 20, no. 63, pp. 1–17, 2019.

[65] P. L. Bartlett and S. Mendelson, “Rademacher and Gaussian complexities: Risk bounds and structural results,” J. Mach. Learn. Res., vol. 3, pp. 463–482, Mar. 2003.

Jian Huang received the B.Sc. degree in mathematics from Wuhan University, Wuhan, China, in 1985, and the Ph.D. degree in statistics from the University of Washington, Seattle, USA, in 1994.

He is currently a Chair Professor of applied statistics and financial mathematics with the Department of Applied Mathematics, The Hong Kong Polytechnic University, Hong Kong, China. He has authored or coauthored over 200 publications, including the Annals of Statistics, Journal of the American Statistical Association, American Journal of Human Genetics, Proceedings of the National Academy of Sciences of the United States of America, Econometrica, Statistical Science, and Journal of Machine Learning Research. His current research interests include semiparametric models, statistical genetics, analysis of high-dimensional data, machine learning, and statistical computing.

Dr. Huang was a fellow of the American Statistical Association in 2009 and a Clarivate Analytics Highly Cited Researcher in 2015, 2016, 2017, and 2018. He was an Associate Editor of the Annals of Statistics and Statistica Sinica.

Yuling Jiao received the B.Sc. degree in applied mathematics from Shangqiu Normal University, Shangqiu, China, in 2008, and the Ph.D. degree in applied mathematics from Wuhan University, Wuhan, China, in 2014.

He is currently an Associate Professor with the School of Statistics and Mathematics, Wuhan University. He has authored or coauthored over 50 research articles, including SIAM Journal on Numerical Analysis, SIAM Journal on Scientific Computing, Journal of Machine Learning Research, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE SIGNAL PROCESSING LETTERS. His current research interests include signal/image processing, inverse problems, statistical computing, and deep learning.

Zhou Yu received the B.Sc. degree from the Hefei University of Technology in 2004 and the Ph.D. degree from East China Normal University, China, in 2010.

He is currently a Professor and the Ph.D. Supervisor with East China Normal University, the Vice Dean of the School of Statistics, and the Deputy Director of the School of Economics and Management. He has published more than 40 articles in well-known statistics journals, such as Annals of Statistics, Biometrika, and Journal of the American Statistical Association. His current research interests include statistical analysis of high-dimensional data and statistical machine learning.