IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 8, 2018, accepted September 10, 2018, date of publication September 26, 2018, date of current version October 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2871825

Nonlinear Dimensionality Reduction Based on HSIC Maximization

ZHENGMING MA[1,2], **ZENGRONG ZHAN**[2,3], **XIAOYUAN OUYANG**[2], **AND XUE SU**[2]
[1]Nanfang College, Sun Yat-sen University, Guangzhou 510275, China
[2]School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510275,China
[3]School of Information Engineering, Guangzhou Panyu Polytechnic, Guangzhou 510483, China

Corresponding author: Zhengming Ma (issmzm@mail.sysu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China through the Project ''Research on Nonlinear Alignment Algorithm of Local Coordinates in Manifold Learning'' under Grant 61773022 and in part by the Scientific Research Projects of Municipal Universities in Guangzhou, China, through the Project ''Research on Cooperative Innovation Operation Mechanism of Sports Events Management Based on Internet Plus'' under Grant 1201620532.

ABSTRACT Hilbert–Schmidt independence criterion (HSIC) is typically used to measure the statistical dependence between two sets of data. HSIC first transforms these two sets of data into two reproducing Kernel Hilbert spaces (RKHS), respectively, and then measures the statistical dependence between them using the Hilbert–Schmidt (HS) operator. This paper proposes a dimension reduction method that is based on HSIC maximization between the high dimensional data and dimension-reduced data, and it is denoted as HSIC-NDR. In the proposed method, the linear kernel is chosen as the kernel function of the RKHS of the low dimensional data after reduction, due to the reason that it can express dimensionality reduction data explicitly from the kernel matrix, thus facilitating the construction of the objective function of the data dimension reduction algorithm. And the kernel function of the RKHS of the original data set can be appropriately chosen according to the specific application. Therefore, the dimension reduction algorithm proposed in this paper can be widely applicable. The experiments are conducted in ten commonly used synthetic and real data sets in the machine learning area. And five representative data dimension reduction algorithms with different properties (linear, nonlinear global, nonlinear local, and nonlinear global + local) are used in the experiment for comparison. The experimental results show that the HSIC-NDR algorithm outperforms those representative algorithms without increasing computational complexity. The proposed HSIC-NDR algorithm and those representative algorithms are all attributed to Rayleigh's calculations.

INDEX TERMS Hilbert-Schmidt independence criterion, nonlinear dimensionality reduction; reproducing Kernel Hilbert spaces.

I. INTRODUCTION

Dimension reduction of data is an important part of machine learning. With the advent of the era of big data, the problem of dimensionality disaster is becoming more and more serious. Therefore, the algorithm of dimension reduction has also been paid more and more attention. In general, the data reduction algorithm is divided into two categories of linear and nonlinear. Some of famous linear data reduction algorithms include PCA [1], MDS [2], LDA [3], MAF [4], SFA [5], SDR [6], ICA [7], DML [8], etc. For nonlinear algorithms, Kernel PCA [9], Kernel LDA [10], ISOMAP [11], LTSA [12], LPP [13], LE [14], LLE [15], HLLE [16], Diffusion MAP [17], Sammon Mapping [18], SNE [19] are prominent. There are many thorough and comparative reviews on dimensionality reduction such as [20]–[22] that are all long articles. The first two are from machine learning theory magazine named ''Journal of Machine Learning Research'', while the last one is from statistics and probability mathematics magazine named ''Statistical Science''.

Dimensionality reduction can also be regarded as a way to extract features from data. For example, in [23]–[26], dimensionality reduction is applied to extract features from Hyper Spectral Imagery (HSI) data. In [23] and [24], Sep-NMF(Separate Nonnegative Matrix Fraction) and sparse matrix fraction are used respectively to extract the most representative hyperspectral bands of HSI. In [25], the subspace methods are exploited to reduce the dimension of HSI. The original subspace method without any constraint is exactly the same as the PCA method. In practice, subspace methods are used with various constraints. In [25], the subspace matrix is optimized under near-isometric, low-rank and sparse constraints. In dimensionality reduction, the high dimensional data are often assumed to lie in a low dimensional subspace or submanifold of a high dimensional Euclidean space. Although these high dimensional data are represented with high dimensional vectors, they are essentially low dimensional and can be dimensionally reduced. However, in practice, these high dimensional data are often polluted by noise and located outside their subspaces or submanifolds. In [26], the matrix of high dimensional data is first decomposed as a sum of a low-rank matrix and a sparse matrix. The data represented by the low-rank matrix are to be dimensionally reduced. The subspace method and manifold regularization are then exploited for the dimensionality reduction.

According to the classification of [20], the algorithms of nonlinear dimension reduction can be divided into three categories: global property preserving, local property preserving, and global and local properties preserving simultaneously. The HSIC-NDR algorithm proposed in this paper is a nonlinear data reduction algorithm with global property preserving. In particular, because HSIC involves kernel functions, the HSIC-NDR belongs to the nonlinear data dimension reduction algorithm based on kernel according to [20], such as Kernel PCA, Kernel LDA and so on. However, the data reduction algorithm based on HSIC maximization between the high dimensional original data and low dimensional data after reduction proposed in this paper has not been reported in any similar way, and there is no literature review of it on any dimensionality reduction. The experimental results provided in this paper shows that HSIC-NDR algorithm on the ten commonly used synthetic and real datasets in machine learning research outperform other data reduction algorithms include PCA [1](linear), ISOMAP [11](global nonlinear), LTSA [12] (locally nonlinear), and LPP [13] (globally and locally nonlinear). In particular, the proposed HSIC-NDR algorithm does not increase computational complexity. Like most data reduction algorithms, the objective function of HSIC-NDR proposed in this paper is also reduced to the form of Rayleigh quotient which can be calculated by the decomposition of eigenvalues and eigenvectors of a symmetric positive definite matrix.

Hilbert-Schmidt Independence Criterion (HSIC) is used to measure the statistical dependence between two random vectors. However, instead of directly measuring the statistical dependence, HSIC first transforms the two random vectors into two reproducing kernel Hilbert spaces (RKHS), and then uses the Hilbert-Schmidt (HS) operator of these two RKHS to measure the statistical dependence of them [27]. The theory of HSIC may seem a bit complicated and may affect the widely apply of HSIC to a certain extent. However, the calculation formula of HSIC (empiric HSIC) is relatively simple and sometimes triggers many generalizations. This paper indicates the meaning and formulas of HSIC through making the definition and derivation of HSIC. Further, this paper applies HSIC to data dimension reduction and proposes a data dimension reduction algorithm based on global HSIC maximization. The theoretical proofs and experimental results provided in this paper show the effectiveness of the proposed algorithm.

To sum up, the proposed HSIC-NDR algorithm has three contributions to dimensionality reduction. First, the proposed HSIC-NDR algorithm is a new algorithm and enriches the library of dimensionality reduction algorithms. Second, there are two kernel functions involved in the proposed HSIC-NDR algorithm. The kernel functions are open and can be chosen according to the specific applications. The existence of kernel functions increases the flexibility and applicability of the proposed HSIC-NDR algorithm. Third, the proposed HSIC-NDR algorithm introduces HSIC into dimensionality reduction for the first time and achieves better performance. This may inspire more attempts in this respect.

The rest of the paper is organized as follows: In the second section, the related works on HSIC are reviewed. In the third section, relevant knowledge is given such as the concept of RKHS. And in particular, the relationship between RKHS and the kernel function are detailed. In the fourth section, the theoretical origins of HSIC are described, and the calculation formula of HSIC in data analysis are derived. A global HSIC-based nonlinear data dimensionality reduction algorithm is proposed in the fifth section. The experimental results are shown in the sixth section to prove the effectiveness of the proposed algorithm. And finally, simple conclusions are made in the last section.

II. LITERATURE REVIEW ON HSIC

The HSIC mathematical theory belongs to functional analysis and it has been studied for a long time [28]. However, from a data analysis point of view, the HSIC received its attention after a series of papers [27], [29], [30] published around 2005. As methodological research, although the history is not long, there are many achievements. In this section, some research advances of HSIC related to the work of this paper in recent years will be elaborated.

From a data analysis perspective, HSIC calculates the statistical dependence of the two sets of data. In general, HSIC requires that the two sets of data contain the same size of data. For example, let $X = [x_1, ..., x_N] \in R^{D\times N}$ and $Z = [z_1, ..., z_N] \in R^{C\times N}$ be the two datasets, and the definition of HSIC between these two datasets is

$$HSIC(X, Z) = tr(K_X C_N K_Z C_N) \tag{1}$$

where

$$K_X = \begin{bmatrix} k_X(x_1, x_1) & \dots & k_X(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k_X(x_N, x_1) & \dots & k_X(x_N, x_N) \end{bmatrix} \in R^{N\times N}$$

$$K_Z = \begin{bmatrix} k_Z(z_1, z_1) & \dots & k_Z(z_1, z_N) \\ \vdots & \ddots & \vdots \\ k_Z(z_N, z_1) & \dots & k_Z(z_N, z_N) \end{bmatrix} \in R^{N\times N}$$

$$C_N = I_N - \frac{1}{N}\Gamma_N \Gamma_N^T \in R^{N\times N}, \Gamma_N = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \in R^N$$

where k_X and k_Z are two kernel functions, and they can be different. And C_N is the centralization matrix.

If X and Z contain different numbers of data (e.g. let $X = [x_1, ..., x_N] \in R^{D\times N}$ and $Z = [z_1, ..., z_M] \in R^{C\times M}, N \neq M$), their HSIC cannot be calculated directly. To solve this problem, [31], [32] proposed the surrogate kernel which is defined as follows.

$$K_{XZ} = \begin{bmatrix} k_X(x_1, z_1) & \dots & k_X(x_1, z_M) \\ \vdots & \ddots & \vdots \\ k_X(x_N, z_1) & \dots & k_X(x_N, z_M) \end{bmatrix} \in R^{N\times M}$$

$$K_{ZX} = \begin{bmatrix} k_Z(z_1, x_1) & \dots & k_Z(z_1, x_N) \\ \vdots & \ddots & \vdots \\ k_Z(z_M, x_1) & \dots & k_Z(z_M, x_N) \end{bmatrix} \in R^{M\times N}$$

$$K_{X\leftarrow Z} = K_{XZ}K_Z^{-1}K_{ZX} \in R^{N\times N},$$

$$K_{Z\leftarrow X} = K_{ZX}K_X^{-1}K_{XZ} \in R^{M\times M}$$

Therefore, two HSIC results are generated:

$$HSIC(X, Y) = tr(K_X C_N K_{X\leftarrow Y} C_N),$$
$$HSIC(Y, X) = tr(K_Y C_M K_{Y\leftarrow X} C_M)$$

In supervised learning, since each category may contain a different number of samples, [31], [32] uses the surrogate kernel to calculate the HSIC between each category sample:

$$H = \begin{bmatrix} HSIC(X^1, X^1) & \dots & HSIC(X^1, X^C) \\ \vdots & \ddots & \vdots \\ HSIC(X^C, X^1) & \dots & HSIC(X^C, X^C) \end{bmatrix} \in R^{C\times C}$$

where X^c $(c = 1, \dots, C)$ represents the samples contained in the c-th category, and C is the number of categories. The objective function of the algorithm in [31] is constructed by using the diagonally dominant matrix as the learning criterion.

In recent years, HSIC has often been applied to supervised feature selection. Let $X = [x_1, \dots, x_N] \in R^{D\times N}$ be the dataset and let $Z = [z_1, \dots, z_N] \in R^{C\times N}$ be the label of X. The label of x_n is represented by z_n in which the c-th $(1 \leq c \leq C)$ element is 1 and the other elements are 0, if x_n (n = 1, …, N) belongs to the c-th category. The purpose of supervised feature selection is to select features in x_n that are the most statistically depended on its label z_n.

For the convenience of description, it is assumed that each component of the data is one of its features. The problem of the supervised feature selection is to select some components that are the most statistically depended on the label from the data. Reference [33] proposed a supervised sparse learning feature selection algorithm. Let $s \in R^D$, the objective function of the HSIC-based sparse-learning feature selection algorithm is:

$$HSIC(X^T s, Z) + \lambda\|s\|_1 \underset{choose\ s}{\longrightarrow} \min \tag{2}$$

where $\|\circ\|_1$ represents the 1-norm. If we denotes

$$u = X^T s = \begin{bmatrix} x_1^T s \\ \vdots \\ x_N^T s \end{bmatrix} = \begin{bmatrix} \Sigma_{j=1}^D x_{1j}s_j \\ \vdots \\ \Sigma_{j=1}^D x_{Nj}s_j \end{bmatrix} = \begin{bmatrix} u^l \\ \vdots \\ u^N \end{bmatrix} \in R^N,$$

then

$$HSIC(X^T s, Z) = HSIC(u, Z) = HSIC(K_u C_N K_Z C_N) \tag{3}$$

where

$$K_u = \begin{bmatrix} k_u(u^1, u^1) & \dots & k_u(u^1, u^N) \\ \vdots & \ddots & \vdots \\ k_u(u^N, u^1) & \dots & k_u(u^N, u^N) \end{bmatrix}$$

and $\|s\|_1$ is called a sparse regularization term. The addition of the sparse regularization term means finding the solution with the least number of s nonzero components [34]. The position of a non-zero component of s above a certain threshold is the position of the selected data feature.

Reference [35] proposed two supervised data feature selection methods using forward and backward HSIC, denoted as FOHSIC and BAHSIC respectively. FOHSIC sorts the data's features in ascending order according to their statistical dependence to the label using HSIC, while BAHSIC sorts the data's features in descending order according to their statistical dependence to the label. FOHSIC and BAHSIC have many developments and varieties in recent years, such as [36].

References [37] and [38] apply HSIC to supervised dictionary learning. The problem of dictionary learning is expressed as follows:

$$\|X - WY\|^2 \underset{choose\ W,Y}{\longrightarrow} \min \tag{4}$$

Here $W \in R^{D\times d}$ is called a dictionary, $Y \in R^{d\times N}$ is called dictionary coefficients of X. The essence of dictionary learning is the subspace approach in machine learning [39], where WY is the projection of X on subspace $spanW$ which represents the subspace spanned from the column vector of W. If the column vectors of W are orthonormal, then according to the projection theorem of function analysis [28], the dictionary coefficient Y of X is the Fourier coefficient of X on W, which is $Y = W^T X$. Therefore the problem of dictionary learning becomes:

$$\|X - WY\|^2 = \left\|X - WW^T X\right\|^2 \underset{choose\ W}{\longrightarrow} \min \tag{5}$$

which is equal to :

$$tr\left(W^T XX^T W\right) \underset{choose\ W}{\longrightarrow} \max \tag{6}$$

This is actually the same as PCA [1]. Further, let $k_Y\left(y, y'\right) = y^T y'$, it has

$$K_Y = \begin{bmatrix} k_Y(y_1, y_1) & \dots & k_Y(y_1, y_N) \\ \vdots & \ddots & \vdots \\ k_Y(y_N, y_1) & \dots & k_Y(y_N, y_N) \end{bmatrix}$$

$$= \begin{bmatrix} y_1^T y_1 & \cdots & y_1^T y_N \\ \vdots & \ddots & \vdots \\ y_N^T y_1 & \cdots & y_N^T y_N^T \end{bmatrix}$$

$$= \begin{bmatrix} x_1^T W W^T x_1 & \cdots & x_1^T W W^T x_N \\ \vdots & \ddots & \vdots \\ x_1^T W W^T x_N & \cdots & x_N^T W W^T x_N \end{bmatrix} = X^T W W^T X$$

Thus, the problem of supervised dictionary learning based on HSIC is expressed as:

$$\begin{aligned} HSIC\left(Y,Z\right) &= tr\left(K_Y C_N K_Z C_N\right) \\ &= tr\left(X^T W W^T X C_N K_Z C_N\right) \\ &= tr\left(W^T X C_N K_Z C_N X^T W\right) \xrightarrow[chosse\ W]{} \max \end{aligned} \tag{7}$$

In [40], HSIC is applied to supervised subspace learning. As mentioned earlier, dictionary learning is subspace learning. Therefore, the method in [40] is the same as the method in [37] and [38], and the objective function is the same.

Although HSIC has been found in many applications in machine learning since it was proposed around 2005, it seems not to have been directly applied to dimensionality reduction. The HSIC-NDR proposed in this paper may be the first try in this respect.

III. RELEVANT KNOWLEDGE

HSIC involves two kernel functions and therefore the applications of HSIC belongs to the category of kernel methods in machine learning. The theory of Reproducing Kernel Hilbert Spaces (RKHS) provides a common mathematical platform for all kernel methods in machine learning.

A. RKHS DEFINITION

Let $S\left(\Omega\right) = \left\{f \left| f : \Omega \to R, \int_\Omega |f\left(x\right)|^2 < +\infty \right.\right\}$ be a square integrable function space, the inner product can be defined on $S\left(\Omega\right)$, making $H = \left(S\left(\Omega\right), \langle\bullet, \bullet\rangle\right)$ a complete inner product space, which is Hilbert space [28]. For example, the inner product can be defined as follows (but not limited to this definition):

$$\langle f, g\rangle = \int_\Omega f\left(x\right) g\left(x\right) dx \tag{8}$$

Definition: Let $H = \left(S\left(\Omega\right), \langle\bullet, \bullet\rangle\right)$ be a Hilbert space, if there is a function $k : \Omega \times \Omega \to R$ meets:

- For any $x \in \Omega$, $k_x = k\left(\bullet, x\right) \in H$;
- For any $f \in H$, $f\left(x\right) = \langle f, k\left(\bullet, x\right)\rangle$;

then H is a reproducing kernel Hilbert space (RKHS), and k is called the reproducing kernel of H.

If H is a RKHS and k is the reproducing kernel of H, then a mapping $\varphi : \Omega \to H$ can be defined for any $x \in \Omega$,

$$\varphi\left(x\right) = k\left(\bullet, x\right) = k_x \in H \tag{9}$$

Thus, by using the property of the reproducing kernel, the inner product of two elements in H can be expressed as follows.

$$\langle\varphi\left(x\right), \varphi\left(y\right)\rangle = \langle k_x, k\left(\bullet, y\right)\rangle = k_x\left(y\right) = k\left(y, x\right) = k\left(x, y\right) \tag{10}$$

equation (10) is a common formula in machine learning kernel methods such as kPCA [9], kLDA [10], kSVM [41] and so on. However, many articles seem to reverse the order. They define φ first, and then use φ to define k. The correct order should define RKHS and a reproducing kernel k first, and then it will get φ. And eventually the formula equation (10) can be derived from the property of the reproducing kernel.

B. GENERATING RKHS BY KERNEL FUNCTION

RKHS can be generated by a kernel function. The definition of kernel function is as follows:

Definition [42]: Let $k : \Omega \times \Omega \to R$, if k meets:

- Symmetry: For any $x, y \in \Omega$, $k\left(x, y\right) = k\left(y, x\right)$
- Square Integrable: For any $x \in \Omega$, $k_x = k\left(\bullet, x\right)$ is square integrable.
- Positive Definite: For any finite number of data, $x_1, \cdots, x_N \in \Omega$, the matrix

$$\begin{bmatrix} k\left(x_1, x_1\right) & \cdots & k\left(x_1, x_N\right) \\ \vdots & \ddots & \vdots \\ k\left(x_N, x_1\right) & \cdots & k\left(x_N, x_N\right) \end{bmatrix}$$

  is a positive definite matrix.

Then, k is a kernel function.

Note: Kernel functions and reproducing kernel are not the same concepts. The kernel function is a separately defined function, while the reproducing kernel is a function that depends on the definition of Hilbert Space.

Theorem: A kernel function may generate a unique RKHS, such that the kernel function is a reproducing kernel of this RKHS.

According to this theorem, as long as a kernel function is defined, the RKHS and the reproducing kernel of the RKHS are defined. Therefore, the kernel function is used to represent RKHS and its reproducing kernel in this paper.

IV. HILBERT-SCHMIDT INDEPENDENCE CRITERION (HSIC)

A. HS OPERATOR

Definition: Let H_X and H_Y be two separable Hilbert spaces, and $\left\{e_i^X \left| i \in I\right.\right\}$ is the standard orthonormal basis of H_X. Let $T : H_X \to H_Y$ is a compact operator and if $\sum_{i\in I} \left\|Te_i^X\right\|_Y^2 < +\infty$, T is a Hilbert-Schmidt (HS) operator [43].

Note 1: In this paper, $\langle\bullet, \bullet\rangle_X$ denotes the inner product of H_X, and $\|\bullet\|_X = \sqrt{\langle\bullet, \bullet\rangle_X}$ denotes the norm of H_X. Similarly, $\langle\bullet, \bullet\rangle_Y$ denotes the inner product of H_Y, and $\|\bullet\|_Y = \sqrt{\langle\bullet, \bullet\rangle_Y}$ denotes the norm of H_Y

Note 2: Compact operators are the operators that map bounded set into the compact set and are one form of bounded operators. The separable Hilbert spaces guarantee the existence of standard orthonormal basis [44].

Theorem: Let $HS\left(H_X \to H_Y\right)$ denote the linear space consist of all HS operators from H_X to H_Y. If for any $T, S \in HS\left(H_X \to H_Y\right)$, $\sum_{i\in I} \left|\left\langle Te_i^X, Se_i^X\right\rangle_Y\right| < +\infty$, then $\left(HS\left(H_X \to H_Y\right), \langle\bullet, \bullet\rangle_{HS}\right)$ is a Hilbert space in which the inner product $\langle \bullet, \bullet \rangle_{HS}$ is defined as follows:

$$\langle T, S \rangle_{HS} = \sum_{i \in I} \left\langle Te_i^X, Se_i^X \right\rangle_Y \tag{11}$$

Theorem [27]: Let H_X and H_Y be two separable Hilbert spaces, and $f_0 \in H_X$,$g_0 \in H_X$, define $f_0 \otimes g_0 : H_X \to H_Y$ as follows: If for any $f \in H_X$,$f_0 \otimes g_0 (f) = \langle f_0, f \rangle_X \, g_0 \in H_Y$, then $f_0 \otimes g_0 \in HS\,(H_X \to H_Y)$.

Note: $f_0 \otimes g_0$ is called the tensor product of f_0 and g_0, and the theorem shows that $f_0 \otimes g_0$ is a type of compact operator.

B. CROSS-COVARIANCE OPERATOR AND MEAN FUNCTION

Let $H_X = (S\,(\Omega_X), \langle \bullet, \bullet \rangle_X)$ be an RKHS and $k_X : \Omega_X \times \Omega_X \to R$ be the reproducing kernel of H_X. $\varphi : \Omega_X \to H_X$ is defined as follows: For any $x \in \Omega_X$, $\varphi\,(x) = k_X\,(\bullet, x) \in H_X$. As mentioned earlier, for any $x', x'' \in \Omega_X$, $\left\langle \varphi\left(x'\right), \varphi\left(x''\right) \right\rangle_X = k_X\left(x', x''\right)$.

Similarly, let $H_Y = (S\,(\Omega_Y), \langle \bullet, \bullet \rangle_Y)$ be an RKHS and $k_Y : \Omega_Y \times \Omega_Y \to R$ be the reproducing kernel of H_Y. $\xi : \Omega_Y \to H_Y$ is defined as follows: For any $y \in \Omega_Y$, $\xi\,(y) = k_Y\,(\bullet, y) \in H_Y$. As mentioned earlier, for any $y', y'' \in \Omega_Y$,$\left\langle \xi\left(y'\right), \xi\left(y''\right) \right\rangle_Y = k_Y\left(y', y''\right)$.

Let X be a random vector valued at Ω_X and Y be a random vector valued at Ω_Y, the Cross-covariance operator between X and Y is defined as follows.

Theorem [27]: Let $\Phi : HS\,(H_X \to H_Y) \to R$, for any $T \in HS\,(H_X \to H_Y)$

$$\Phi\,(T) = E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}\right] \tag{12}$$

If $E_{XY}\left[\|\varphi\,(X) \otimes \xi\,(Y)\|_{HS}\right] < +\infty$, then Φ is a continuous linear function on $HS\,(H_X \to H_Y)$

Proof: The rationale for the definition of Φ is illustrated first here. For any $x \in \Omega_X$ and $y \in \Omega_Y$, it has $\varphi\,(x) \in H_X$ and $\xi\,(y) \in H_Y$. Hence, the tensor product of $\varphi\,(x)$ and $\xi\,(y)$ is expressed as $\varphi\,(x) \otimes \xi\,(y) \in HS\,(H_X \to H_Y)$, and $\langle \varphi\,(x) \otimes \xi\,(y), T \rangle_{HS}$ is a numerical value. When X and Y are random vectors, $\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}$ becomes a function of random vectors X and Y. Therefore, $\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}$ becomes a randome vector, and $E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}\right]$ is a numerical value and is used to express the mathematical expectation (statistical mean) of this random variable.

The proof of the linearity and continuity of Φ is as follows:

$$\begin{aligned} \Phi\,(\alpha T + \beta S) &= E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), \alpha T + \beta S \rangle_{HS}\right] \\ &= \alpha E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}\right] \\ &\quad + \beta E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), S \rangle_{HS}\right] \qquad (13) \\ &= \alpha \Phi\,(T) + \beta \Phi\,(S) \\ |\Phi\,(T)| &= \left|E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}\right]\right| \\ &\leq E_{XY}\left[\left|\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}\right|\right] \\ &\leq E_{XY}\left[\|\varphi\,(X) \otimes \xi\,(Y)\|_{HS} \, \|T\|_{HS}\right] \\ &= \|T\|_{HS} \, E_{XY}\left[\|\varphi\,(X) \otimes \xi\,(Y)\|_{HS}\right] \qquad (14) \end{aligned}$$

The above equation shows that Φ is a bounded operator, which is also a continuous operator if it is a linear operator.

According to the representation theorem of continuous linear function (Riesz theorem), there exists a unique HS operator $T_\Phi \in HS\,(H_X \to H_Y)$ such that for any HS operator $T \in HS\,(H_X \to H_Y)$, there is

$$\Phi\,(T) = E_{XY}\left[\langle \varphi\,(X) \otimes \xi\,(Y), T \rangle_{HS}\right] = \langle T, T_\Phi \rangle_{HS} \tag{15}$$

T_Φ is called cross-covariance operator, which is often denoted as C_{XY}.

The definition of the mean function of X in H_X is detailed in the rest of this section.

Theorem: Let $H_X = (S\,(\Omega_X), \langle \bullet, \bullet \rangle_X)$ be an RKHS, and $k_X : \Omega_X \times \Omega_X \to R$ be a reproducing kernel function of H_X. And let $\varphi : \Omega_X \to H_X$, for any $x \in \Omega_X$, $\varphi\,(x) = k_X\,(\bullet, x)$. And let $\Phi : H_X \to R$, for any $f \in H_X$,

$$\begin{aligned} \Phi\,(f) &= E_X\left[\langle \varphi\,(X), f \rangle_X\right] \\ &= E_X\left[\langle k\,(\bullet, X), f \rangle_X\right] = E_X\,[f\,(X)] \end{aligned} \tag{16}$$

then Φ is a continuous linear function on H_X.

Proof: The rationale for the definition of Φ is illustrated first here. For any $x \in \Omega_X$ and $\varphi\,(x) \in H_X$, $\langle \varphi\,(x), f \rangle_X$ is a numeric value. When X is a random vector, $\langle \varphi\,(x), f \rangle_X$ becomes the function of the random vector X. Hence, $\langle \varphi\,(x), f \rangle_X$ becomes a random vector and $E_X\left[\langle \varphi\,(X), f \rangle_X\right]$ is used to denote the mathematical expectation (statistical mean) of this random vector. The proof of the linearity and continuity of Φ is as follows:

$$\begin{aligned} \Phi\,(\alpha f + \beta g) &= E_X\left[\langle \varphi\,(X), \alpha f + \beta g \rangle_X\right] \\ &= \alpha E_X\left[\langle \varphi\,(X), f \rangle_X\right] + \beta E_X\left[\langle \varphi\,(X), g \rangle_X\right] \\ &= \alpha \Phi\,(f) + \beta \Phi\,(g) \qquad (17) \\ |\Phi\,(f)| &= \left|E_X\left[\langle f, \varphi\,(X) \rangle_X\right]\right| \leq E_X\left[\left|\langle f, \varphi\,(X) \rangle_X\right|\right] \\ &\leq E_X\left[\|f\|_X \, \|\varphi\,(X)\|_X\right] = \|f\|_X \, E_X\left[\|\varphi\,(X)\|_X\right] \\ & \qquad (18) \end{aligned}$$

The above equation shows that Φ is a bounded operator, which is also a continuous operator if it is a linear operator.

Similarly, according to the representation theorem of continuous linear functional (Riesz theorem), there exists a unique function $f_\Phi \in H_X$ such that for any function HS operator $f \in H_X$, there is

$$\Phi\,(f) = E_X\left[\langle \varphi\,(X), f \rangle_X\right] = \langle f, f_\Phi \rangle_X \tag{19}$$

where f_Φ is called the mean function of X in H_X, and it is denoted as μ_X. Similarly, the mean function of Y in H_Y is denoted as μ_Y.

The relationship between the cross-covariance operator C_{XY} and the mean functions μ_X and μ_Y can be represented by Fig.1.

C. HSIC

1) HSIC DEFINITION AND SIGNIFICANCE

Definition: The HSIC definition of two random vectors X and Y is

$$HSIC\,(X, Y) = E_{XY}\left[\|(\varphi\,(X) - \mu_X) \otimes (\xi\,(Y) - \mu_Y)\|_{HS}^2\right] \tag{20}$$

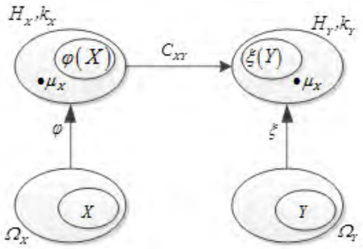


FIGURE 1. Schematic diagram of the cross-covariance operator C_{XY} and the averaging functions μ_X and μ_Y.

$HSIC\,(X,Y)$ does not directly measure the covariance $E_{XY}\left[(X-E_X[X])(Y-E_Y[Y])\right]$ of two random vector X and Y. Instead, by using mappings φ and ξ,X and Y are mapped to two RKHS spaces H_X and H_Y respectively. And the covariance between $\varphi(X)$ and $\xi(Y)$ is then measured. Proper selection of mappings φ and ξ show some of the intrinsic features of X and Y. HSIC measures the covariance of these intrinsic features.

Since mapping φ and mapping ξ are defined by the reproducing kernels of H_X and H_Y in RKHS space, and the RKHS and their reproducing kernels are uniquely defined by the kernel function, choosing mappings φ and ξ is to choose two kernel functions k_X and k_Y. It can be adapted to different applications by selecting different kernel functions.

In probability theory [45], if the covariance is normalized, it is the correlation coefficient. The correlation coefficient measures the degree of linear dependence between two random vectors. Therefore, HSIC is essentially a measure of the degree of linear dependence between $\varphi(X)$ and $\xi(Y)$.

D. HSIC CALCULATION

1) STATISTICAL MEAN CALCULATION FORMULA

The calculation steps of $HSIC\,(X,Y)$ is shown in this section. If the joint probability distribution of random vectors X and Y is known, $HSIC\,(X,Y)$ can be calculated according to the following formula:

$$\begin{aligned}HSIC\,(X,Y) &= E_{XY}\left[\|(\varphi(X)-\mu_X)\otimes(\xi(Y)-\mu_Y)\|_{HS}^2\right]\\ &= \|C_{XY}-\mu_X\otimes\mu_Y\|_{HS}^2\\ &= \langle C_{XY},C_{XY}\rangle_{HS}-2\langle C_{XY},\mu_X\otimes\mu_Y\rangle_{HS}\\ &\quad+\langle\mu_X\otimes\mu_Y,\mu_X\otimes\mu_Y\rangle_{HS}\end{aligned}\tag{21}$$

where

$$\begin{aligned}&\langle C_{XY},C_{XY}\rangle_{HS}\\ &\quad= E_{XY}E_{X'Y'}\left[k_X\left(X,X'\right)k_Y\left(Y,Y'\right)\right]\end{aligned}\tag{22}$$

$$\begin{aligned}&\langle C_{XY},\mu_X\otimes\mu_Y\rangle_{HS}\\ &\quad= E_{XY}\left[E_{X'}\left[k_X\left(X,X'\right)\right]E_{Y'}\left[k_Y\left(Y,Y'\right)\right]\right]\end{aligned}\tag{23}$$

$$\begin{aligned}&\langle\mu_X\otimes\mu_Y,\mu_X\otimes\mu_Y\rangle_{HS}\\ &\quad= \langle\mu_X,\mu_X\rangle_X\langle\mu_Y,\mu_Y\rangle_Y\end{aligned}\tag{24}$$

2) THE COMPUTATION FORMULA FOR THE MEAN OF SAMPLES

In general, the joint probability distribution between random vectors X and Y is unknown, and only some sample values of the random vectors X and Y are given. In this case, the mean of samples is used to represent the statistical mean to calculate $HSIC\,(X,Y)$.

Given two sets of data $\{x_1,\cdots,x_N\}\subseteq\Omega_X$ and $\{y_1,\cdots,y_N\}\subseteq\Omega_Y$, which are treated as samples of the random vectors X and Y, and assuming that the probability of the random event $\left\{X=x_i;\,Y=y_j\right\}$ is zero, ie. $P\left\{X=x_i;\,Y=y_j\right\}=0$ when $i\neq j$, the following equation holds.

$$\begin{aligned}C_{XY}&\approx\frac{1}{N}\sum_{n=1}^{N}\varphi(x_n)\otimes\xi(y_n),\\ \mu_X&\approx\frac{1}{N}\sum_{n=1}^{N}\varphi(x_n),\quad\mu_Y\approx\frac{1}{N}\sum_{n=1}^{N}\xi(y_n)\end{aligned}\tag{25}$$

Substituting the above results into equations (22)-(24), there are

$$\langle C_{XY},C_{XY}\rangle_{HS}\approx\frac{1}{N^2}tr\,(K_XK_Y)\tag{26}$$

$$\langle C_{XY},\mu_X\otimes\mu_Y\rangle_{HS}\approx\frac{1}{N^3}\Gamma_N^TK_XK_Y\Gamma_N\tag{27}$$

$$\langle\mu_X\otimes\mu_Y,\mu_X\otimes\mu_Y\rangle_{HS}\approx\frac{1}{N^4}\Gamma_N^TK_X\Gamma_N\Gamma_N^TK_Y\Gamma_N\tag{28}$$

where $\Gamma_N=\begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix}\in R^N$ is an all-1 vector of N dimensions, and

$$K_X=\begin{bmatrix}k_X(x_1,x_1)&\cdots&k_X(x_1,x_N)\\ \vdots&\ddots&\vdots\\ k_X(x_N,x_1)&\cdots&k_X(x_N,x_N)\end{bmatrix},$$

$$K_Y=\begin{bmatrix}k_Y(y_1,y_1)&\cdots&k_Y(y_1,y_N)\\ \vdots&\ddots&\vdots\\ k_Y(y_N,y_1)&\cdots&k_Y(y_N,y_N)\end{bmatrix}$$

Then, substituting equations (26)-(28) into (21), we get:

$$\begin{aligned}HSIC\,(X,Y)&=\|C_{XY}-\mu_X\otimes\mu_Y\|^2\\ &=\frac{1}{N^2}tr\,(K_YC_NK_XC_N)=\frac{1}{N^2}tr\left(\hat{K}_Y\hat{K}_X\right)\end{aligned}\tag{29}$$

where $C_N=I_N-\frac{1}{N}\Gamma_N\Gamma_N^T$ is the centralization matrix, and $\hat{K}_X=K_XC_N$, $\hat{K}_Y=K_YC_N$. $\hat{K}_X$ and $\hat{K}_Y$ are called the centralization matrix of K_X and K_Y respectively.

E. SUMMARY

1) HSIC essentially calculates the covariance of two random vectors. If covariance is normalized, is the correlation coefficient. The correlation coefficient measures the degree of linear dependence between two random vectors. Only when the random vectors obey the Gaussian distribution, the linear independence is equal to the statistical independence. Therefore, instead of measuring the degree of statistical independence of two random vectors as the name implies, HSIC measures

the degree of linear dependence between two random vectors.

2) Instead of directly measuring the degree of linear dependence between two random vectors X and Y, HSIC measures the degree of linear dependence between their transformations $\varphi(X)$ and $\xi(Y)$, where φ and ξ are defined by the kernel functions k_X and k_Y, respectively. φ and ξ denote a certain degree of preprocessing of the data, revealing some properties and features in RKHS (H_X and H_Y) that does not exhibit in the original data space (Ω_X and Ω_Y).
3) The calculation of HSIC is simple and clear. If the data is regarded as a specific implementation of a random vector (sample), then HSIC is the trace of the product of the two (centralization) kernel matrix. The kernel matrix is composed of the values of the kernel function on the data sample. The calculation formula of HSIC also shows that HSIC is not only related to the data but also related to the kernel function.

V. NONLINEAR DIMENSIONALITY REDUCTION BASED ON HSIC MAXIMIZATION (HSIC-NDR ALGORITHM)

The problem of data dimension reduction can be described as follows: Given a set of data $X = \left[x_1 \cdots x_N \right] \in R^{D\times N}$ in a high-dimensional Euclidean space R^D, it is required to find a set of data $Y = \left[y_1 \cdots y_N \right] \in R^{d\times N}$ in a low-dimensional Euclidean space R^d as the dimension reduction result according to certain criteria. Y is the dimension reduction result of X and $d << D$.

A. DIMENSIONALITY REDUCTION CRITERION: HSIC MAXIMIZATION

In this paper, the maximization of $HSIC\ (X, Y)$ is used as the criterion for data dimension reduction. That is

$$HSIC\ (X,Y)=\frac{1}{N^2}tr\ (K_Y C_N K_X C_N) \underset{choose\ Y\in R^{d\times N}}{\longrightarrow} \max \quad (30)$$

In other words, the goal is to find a set of data Y in a low dimensional Euclidean space R^d, which is as far as possible linearly dependent (statistical dependence) to the data X in a high dimensional Euclidean space R^D using the HSIC. And Y is referred to as the reduced dimension result of X. To facilitate the narrative, in the following part of this article, the algorithm proposed here is referred to as HSIC-NDR.

Compared with other dimensionality reduction algorithms with linearly dependent requirements (such as PCA where $Y = W^T X$ and W in the linear transformation matrix), the HSIC-NDR algorithm respects the intrinsic nature of data itself more.

B. THE OBJECTIVE FUNCTION OF HSIC-NDR ALGORITHM

In $HSIC\ (X, Y)$, the dimensionality reduction result Y is hidden in the kernel matrix K_Y, which is not conducive to the solution of the HSIC-NDR problem shown in formula (30). To explicitly represent Y, the kernel function of Y in HSIC-NDR is defined as $k_Y : R^d \times R^d \to R$. And for any $y', y'' \in R^d$,

$$k_Y\left(y', y''\right) = y'^T y'' + \kappa\delta\left(y', y''\right) \quad (31)$$

where $\kappa > 0, \delta\left(y', y''\right) = \begin{cases} 1 & y' = y'' \\ 0 & others \end{cases}$. δ is added in order to theoretically guarantee the positive definiteness of k_Y. From the following derivation, it can be seen that κ does not appear in the objective function of HSIC-NDR.

Obviously, the function k_Y shown in the formula (31) is a kernel function. According to the discussion in Section III, k_Y can uniquely produce an RKHS H_Y such that k_Y is the reproducing kernel of H_Y. Thus, K_Y can be expressed as follows.

$$\begin{aligned} K_Y &= \begin{bmatrix} k_Y\left(y_1, y_1\right) & \cdots & k_Y\left(y_1, y_N\right) \\ \vdots & \ddots & \vdots \\ k_Y\left(y_N, y_1\right) & \cdots & k_Y\left(y_N, y_N\right) \end{bmatrix} \\ &= \begin{bmatrix} y_1^T y_1 & \cdots & y_1^T y_N \\ \vdots & \ddots & \vdots \\ y_N^T y_1 & \cdots & y_N^T y_N \end{bmatrix} + \kappa I_N = Y^T Y + \kappa I_N \quad (32) \end{aligned}$$

Substituting equation (32) into equation (30), it gets

$$\begin{aligned} &HSIC\ (X, Y) \\ &\quad = \frac{1}{N^2} tr\ (K_Y C_N K_X C_N) \\ &\quad = \frac{1}{N^2} tr\left(Y^T Y C_N K_X C_N\right) + \frac{\kappa}{N^2} tr\ (C_N K_X C_N) \\ &\quad = \frac{1}{N^2} tr\left(Y C_N K_X C_N Y^T\right) + \frac{\kappa}{N^2} tr\ (C_N K_X C_N) \quad (33) \end{aligned}$$

Since $tr\ (C_N K_X C_N)$ has nothing to do with Y, and N or κ are also irrelevant to Y, the problem of equation (30) can be equivalent to the following problem:

$$tr\left(Y C_N K_X C_N Y^T\right) \underset{choose\ Y}{\longrightarrow} \max \quad (34)$$

Geometrically, YC_N implies that the center of Y is shifted from $\bar{y}$ to the origin of the low-dimensional Euclidean space R^d, where $\bar{y}=\frac{1}{N}\sum_{n=1}^{N} y_n \in R^d$. From the point of dimension reduction view, YC_N and Y is the same. Therefore, the problem shown in equation (34) can be further reduced to the following problem:

$$tr\left(Y K_X Y^T\right) \underset{choose\ Y}{\longrightarrow} \max \quad (35)$$

Equation (35) is the objective function of the HSIC-NDR algorithm. Obviously, the equation (35) is simple, easy to understand and use. Besides, the solution to the problem shown in equation (35) is also very simple. In fact, since the kernel matrix K_X is a symmetric positive definite matrix, the solution of the problem shown in equation (35) can be transformed into the problem of calculating the Rayleigh quotient maximum under the condition of $YY^T = I_d$. The Rayleigh quotient calculation problem is a common problem in matrix calculation. There are many ready-made source programs available for calling.

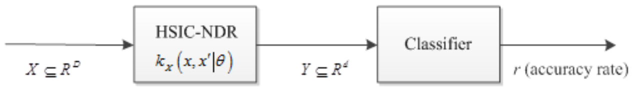


FIGURE 2. HSIC-NDR dimension reduction with classifier.

C. THE ADAPTABILITY OF HSIC-NDR ALGORITHM

In the objective function of the HSIC-NDR shown in equation (35), the kernel matrix K_X is optional and depends on the kernel function k_X and the data X to be reduced. In practice, different kernel functions can be chosen depending on the application. Therefore, HSIC-NDR is fundamentally an algorithmic framework. Only when the kernel function and the parameters contained in the kernel function are selected, HSIC-NDR becomes a specific algorithm.

For example, given a dataset X in high-dimensional Euclidean space R^D to train a classifier, the proposed HSIC-NDR algorithm is used to reduce the high-dimensional dataset X to the low-dimensional dataset in Euclidean space R^d, where $d << D$. And the classifier is trained in a low-dimensional Y Euclidean space R^d (See Fig.2).

In Fig.2, $k_X\left(x, x'\left|\theta\right.\right)$ represents the kernel function, where θ represents the parameter of the kernel function and r represents the classification accuracy. Since the HSIC-NDR algorithm depends on the kernel function $k_X\left(x, x'\left|\theta\right.\right)$, the classification accuracy r depends on $k_X\left(x, x'\left|\theta\right.\right)$. Therefore, it can be denoted as $r\left(k_X\left(x, x'\left|\theta\right.\right)\right)$.

If there are N_k kernels to choose from $k_X^i\left(x, x'\left|\theta_i\right.\right)$, where $i = 1, \cdots N_k$, then the optimal kernel selection procedure is as follows:

(1) According to the classification accuracy r, determine the optimal parameters of each kernel function:

$$\theta_i^* = \arg\max_{\theta_i} r\left(k_X^i\left(x, x'\left|\theta_i\right.\right)\right) \tag{36}$$

where $i = 1, \cdots N_k$

(2) According to the classification accuracy r, choose the best kernel function:

$$i^* = \arg\max_{1 \le i \le N_k} r\left(k_X^i\left(x, x'\left|\theta_i^*\right.\right)\right) \tag{37}$$

Thus, the kernel function used by the HSIC-NDR algorithm is $k_X^{i^*}\left(x, x'\left|\theta_i^*\right.\right)$.

VI. EXPERIMENTAL RESULTS

A. KERNEL POOL

As mentioned above, the proposed HSIC-NDR algorithm first transforms original dataset X and dimension reduction result Y into two RKHS spaces H_X and H_Y, and then it uses the HS operator between the two RKHSs to measure the linear dependence of these two datasets. In order to explicitly represent the dataset Y, the reproducing kernel k_Y of H_Y is selected as a positive definite linear kernel, while the reproducing kernel k_X of H_X is an optional kernel. The advantage of the HSIC-NDR algorithm is that one can choose the best kernel function in the kernel pool according to the needs of the practical application. Reference [35] discusses the properties of some available kernel functions. However, because of the variety of applications and different learning models, the best approach may be to determine the optimal kernel function according to the given application, and a particular learning model is testing on each of the given kernel functions by using given samples. In the experiment provided in this paper, the kernel pool contains the following eight kernel functions.

1) Polynomial Kernel(poly)

$$k\left(x, x'\right) = \left(\alpha x^T x' + \beta\right)^\gamma \tag{38}$$

When $\alpha = \gamma = 1$ and $\beta = 0$, it is a linear kernel(lin).

B-spline

2) Kernel (bspline)

$$k\left(x, x'\right) = \prod_{n=1}^{D} B_3\left(x_n - x_n'\right) \tag{39}$$

where B_3 is cubic spline whose formula is as follows.

$$B_3(\omega) = \begin{cases} \dfrac{4 - 6\left|\omega\right|^2 + 3\left|\omega\right|^3}{6} & 0 \le \left|\omega\right| < 1 \\ \dfrac{\left(2 - \left|\omega\right|\right)^3}{6} & 1 \le \left|\omega\right| < 2 \\ 0 & others, \quad \omega \in R \end{cases}$$

Chi-Square

3) Kernel (chi2)

$$k\left(x, x'\right) = \sum_{n=1}^{D} \frac{2x_n x_n'}{x_n + x_n'} \tag{40}$$

4) Generalized T-Student Kernel (tst)

$$k\left(x, x'\right) = \frac{1}{1 + \left\|x - x'\right\|^\gamma} \tag{41}$$

5) Wave Kernel (wave)

$$k\left(x, x'\right) = \frac{\theta}{\left\|x - x'\right\|} \sin\left(\frac{\left\|x - x'\right\|}{\theta}\right) \tag{42}$$

6) Wavelet Kernel (wavelet)

$$k\left(x, x'\right) = \prod_{n=1}^{D} h\left(\frac{x_n - x_n'}{a}\right) \tag{43}$$

where $h(\omega) = \cos(1.75\omega)\exp\left(-\frac{\omega^2}{2}\right)$, $\omega \in R$

7) Gaussian Kernel (rbf)

$$k\left(x, x'\right) = \exp\left(-\frac{||d(x, x')||^2}{2\sigma^2}\right) \tag{44}$$

where $d(x, x')$ is the distance of two vectors, and normally $||d(x, x')||^2 = \left\|x - x'\right\|^2$ which is Euclidean Distance. However, $d(x, x')$ can also be the geodesic distance of these two vectors, and this type of kernel is marked as "rbf-geo" in the experiments.

8) Sigmoid Kernel (sigmod)

$$k\left(x, x'\right) = \tanh\left(ax^T x' + c\right) \tag{45}$$

B. COMPARISON ALGORITHM

In the experiment, the proposed HSIC-NDR algorithm is compared with PCA, MDS, ISOMAP, LTSA and LPP algorithms. From the perspective of keeping the data unchanged in the process of dimension reduction, the data dimension reduction algorithm can be broadly divided into three categories: global preserving, local preserving, local and global preserving simultaneously. PCA, MDS, and ISOMAP belong to the algorithm of global preserving. LTSA belongs to the algorithm of local preserving. LPP belongs to the algorithm of local and global preserving simultaneously. The proposed HSIC-NDR algorithm belongs to the algorithm of global preserving.

We have conducted the experiments on synthetic and real datasets. The main algorithm is implemented in Python. The running time is measured on a 2.4GHz PC with 8G memory running on Windows 7.

C. DATA SET

The experimental datasets used in this paper are all commonly used datasets in machine learning research. Many articles compare the effects of various algorithms on these datasets.

1) SYNTHETIC BENCHMARK DATASETS

The Swiss Roll and S-Curve datasets are typically used for evaluating manifold learning algorithms. Both datasets are 1000-point uniformly sampled. We select the RBF kernel with different choices of σ for HSIC algorithm. And the 2-D visual results of Swiss Roll and S-Curve are shown in Fig.3 and Fig.5 correspondingly. Visual results compare to other classical algorithms are shown in Fig.4 and Fig.6. From the experimental results show in Fig.3, Fig.4, Fig.5 and Fig.6, HSIC shows better results than the traditional MDS, PCA, and LPP algorithms. The experimental results of the HSIC algorithm are comparable to the experimental results of the globally preserving ISOMAP algorithm. And the calculation time of these two algorithms is also close.

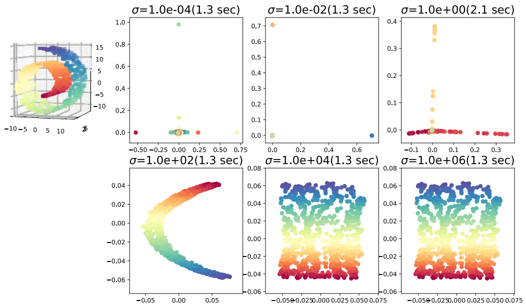


FIGURE 3. Experiment results of HSIC on swiss roll dataset using RBF kernel with different parameters.

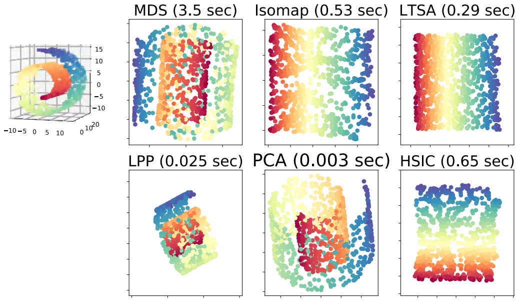


FIGURE 4. Experiment results of different algorithms on swiss roll dataset.

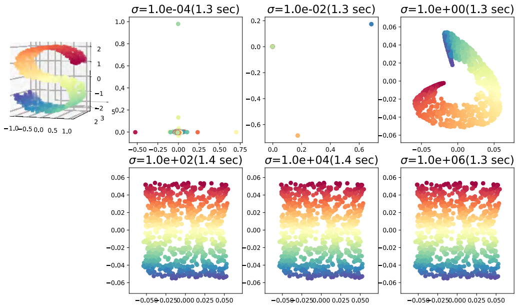


FIGURE 5. Experiment results of HSIC on S-curve dataset using RBF kernel with different parameters.

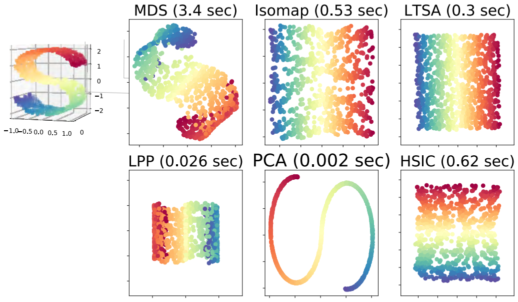


FIGURE 6. Experiment results of different algorithms on S-curve dataset.

2) IRIS

Iris dataset is collected by Edgar Anderson to quantify the morphologic variation of Iris flowers of three related species. The dataset contains 3 classes (Iris setosa, Iris virginica, and Iris versicolor) of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other two, and the latter are NOT linearly separable from each other. Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters.

During the experiment, we plot the dataset using the first three feature in 3-D mode. And we use HSIC-NDR algorithm to reduce the data to two dimensions by using Chi-Square kernel function with different parameters. And the experiment results are plotted in Fig. 7. Experiment results compare to other classical algorithms are shown in Fig.8. From the experimental results, the HSIC algorithm has shown reasonable visual results. In addition, the calculation time of the HSIC algorithm is relatively small.

3) EXTEND YALEB

This dataset has 38 individuals and around from 59 to 64 near grayscale images under different illuminations per individual. The whole image dataset contains 2414 images and

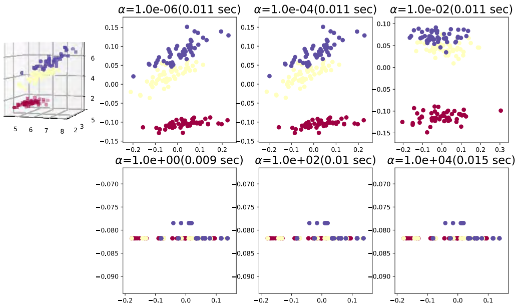


FIGURE 7. Experiment results of HSIC on iris dataset using chi-square kernel with different parameters.

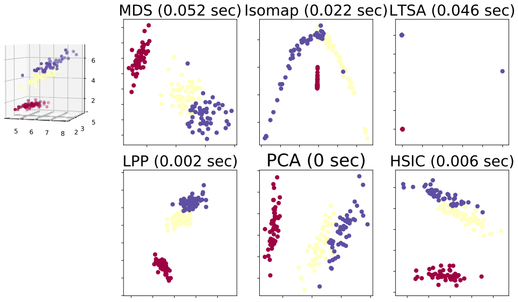


FIGURE 8. Experiment results of different algorithms on iris dataset.

is downloaded from `http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html`. Fig.9 is part of this dataset.

FIGURE 9. Part of extend YaleB image dataset.

During the experiment, each image is organized into a 32×32 pixel image. In classification, it randomly takes 20 images of each individual as the training set, and the rest is the test set. A total of 10 randomized experiments has been run and the average of 10 randomized experimental results is taken as the final experimental results. A10-NN classifier is used for classification.

Table 1 shows the experimental results of HSIC-NDR using different kernel functions. In Table 1, the first column is the number of dimensions after dimensionality reduction, and the number of dimensions of the original data is $32 \times 32 = 1024$. The highest accuracy rate for each dimension has been bolded. The results of HSIC-NDR vary widely with different kernel functions. Therefore, within a certain range, it is necessary to choose the best kernel function.

Table 2 shows the experimental comparison results of HSIC-NDR and other algorithms. In Table 2, the accuracy of the HSIC-NDR is the best accuracy picking from Table 1. As can be seen from Table 2, HSIC-NDR achieves the best results for each dimension reduction. Also, the accuracy of classification of HSIC-NDR and other algorithms are higher than that of non-dimensionality reduction data whose direct classification rate is 39.71%. In the case of HSIC-NDR, the HSIC-NDR uses a linear kernel as the kernel function for data dimension reduction, which is equivalent to the criterion that requires the covariance of data after dimension reduction to be the largest, thus improving the classification accuracy.

FIGURE 10. Part of AR images dataset.

FIGURE 11. Part of ORL images dateset.

4) AR

The AR contains over 4,000 color images corresponding to126 people's faces with 70 men and 56 women for each. The images are shot during two weeks. Each person took pictures of different expressions, illumination conditions and occlusions in each week. Fig.10 shows part of images of the AR dataset, where the first row of the figure are the pictures taken in the first week and the second row of the figure are the pictures taken in the second week. The AR dataset is downloaded from `http://www2.ece.ohio-state.edu/~aleix/ARdatabase.html`.

When conducting the experiment, each image is organized into a gray-scale image of 60×43 pixels. During the classification, the pictures taken in the first week were used as the training set, and the pictures taken in the second week were used as the test set. Since each person has a small number of pictures, a 5-NN classifier is used for classification.

Table 3 shows the experimental results of HSIC-NDR using different kernel functions. In Table 3, the first column is the number of dimension after dimension reduction, and the number of dimensions in the original data is $60 \times 43 = 2580$. The highest accuracy rate for each dimensionality has been bolded. As can be seen from Table 3, the results of HSIC-NDR vary widely with different kernel functions. Therefore, for a specific application, it is necessary to choose the proper kernel function.

Table 4 shows the experimental comparison results of HSIC-NDR and other algorithms. In Table 4, the accuracy of the HSIC-NDR is the best accuracy picking from Table 3. HSIC-NDR still achieves the best results for each dimension.

Tables 5 and 6 show the experimental results on the unobstructed (no glasses, no mouth cover) images in the AR dataset. As the image quality is better, the accuracy is improved.

5) ORL

ORL involves 40 people, each taking 10 grayscale images of different expressions, different lighting, and different shades. The entire image dataset has a total of 400 images. Fig.11 shows part of the images of ORL. The download

TABLE 1. Experimental results of HSIC-NDR on Extend YaleB dataset using different kernel functions.

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	34.98	33.14	53.75	19.7	3.72	34.49	32.96	33.36	13.83	**81.52**
20	58.33	57.56	**64.28**	49.63	19.76	33.35	57.12	58.28	40.11	47.68
30	66.77	66.14	**69.94**	59.17	30.47	38.7	66.12	64.7	51.95	32.99
40	71.23	70.64	**73.50**	66.05	35.17	43.46	70.71	70.25	59.17	24.51
50	**73.45**	72.65	73.36	70.91	41.14	44.49	72.59	71.84	62.69	19.75
60	74.75	**74.76**	73.19	73.97	44.47	46.45	74.72	72.82	66.26	16.90
70	74.76	**76.4**	72.64	75.59	48.13	46.4	74.81	74.67	66.6	15.73
80	75.87	76.23	70.44	**77.42**	50.59	48.42	75.44	76.05	67.81	14.22
90	76.67	77.68	68.94	**78.95**	52.66	49.26	77.64	76.9	70.73	14.32
100	76.33	78.2	65.94	**80.07**	53.69	49.35	76.6	77.28	71.09	12.71
Time(s)	1.6×10^1	1.7×10^1	3.1×10^1	1.1×10^2	2.5×10^1	1.6×10^1	4.8×10^2	3.7×10^1	3.0×10^1	1.6×10^1

TABLE 2. Experimental results of HSIC-NDR compare with other algorithms on Extend YaleB dataset. (Without dimensionality reduction, the accuracy of direct classification is 39.71%).

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	30.55	35.13	8.42	34.17	13.27	**81.52**
20	38.29	40.87	22.61	40.91	23.68	**64.28**
30	40.73	42.74	37.24	44.66	28.35	**69.94**
40	41.04	44.09	47.29	46.64	30.93	**73.50**
50	40.91	44.15	57.94	49.48	33.62	**73.45**
60	40.25	44.47	61.08	52.19	34.85	**74.76**
70	40.31	44.55	62.42	52.62	35.73	**76.4**
80	40.42	44.61	65.51	53.14	36.38	**77.42**
90	39.96	44.03	66.37	54.12	37.33	**78.95**
100	39.01	44.07	66.58	55.15	36.74	**80.07**
Time(s)	7.9×10^1	1.2×10^1	1.9×10^1	1.8×10^1	4.1×10^{-1}	7.3×10^1

TABLE 3. Experimental results of HSIC-NDR on AR dataset using different kernel functions.

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	19	19.31	30.77	23.77	14.62	19.77	21.62	21.62	0.38	**35.69**
20	31.46	34.15	**37.85**	31.92	18.62	29.38	32.54	32.54	0.77	27.92
30	42.62	**45.23**	44.15	40.62	22.77	34.00	42	42.08	1.54	14.00
40	**50.23**	49.15	53.54	45.00	23.85	38.92	49.92	50.08	0.85	10.31
50	52.62	51.85	**58.62**	51.92	27.77	40.92	53.08	53.31	0.77	7.54
60	52.85	54.77	**62.85**	54.69	27.69	42.54	53.77	53.69	1.62	5.77
70	54.46	56.23	**64.31**	56.85	28.46	43.92	55.15	55.15	0.92	4.85
80	55.15	57.46	**65.08**	58.00	29.77	45.23	56	56.23	0.62	4.31
90	57.38	57.23	**66.38**	59.54	30.85	44.77	57.38	57.62	0.38	3.77
100	55.23	56.15	**67.69**	59.92	31.69	44.77	56.15	56.69	0.85	4.15
Time(s)	1.9×10^1	1.9×10^1	6.1×10^1	3.9×10^2	1.4×10^2	1.9×10^1	1.3×10^3	1.5×10^2	1.4×10^2	1.8×10^1

URL of ORL dataset is `http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html`.

When running the experiment, each image is organized into an image of 32 × 32 pixels. Since each person has a small number of images, a 3-NN classifier is used for classification.

For the classification result shown in Table 7 and 8, it randomly takes 3 images of each person as the training set, the rest of the images is used as the test set. For the classification result shown in Table 9 and 10, it randomly takes 4 images of each person as the training set, the rest

TABLE 4. Experimental results of HSIC-NDR compare with other algorithms on AR dataset. (Without dimensionality reduction, the accuracy of direct classification is 25.77%).

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	22.23	18.77	14.23	20.77	20.23	**35.69**
20	25.77	20.92	16.15	22.62	23.23	**37.85**
30	27.77	21.62	19.77	23.08	24.46	**45.23**
40	27.62	22.23	21.38	23.15	25.15	**50.23**
50	27.08	23.38	27.31	23.77	25	**58.62**
60	26.62	22.62	32.15	23.92	24.92	**62.85**
70	26.69	23.69	36.77	24.23	25.92	**64.31**
80	26.69	23.46	42.23	24.23	26.46	**65.08**
90	26.85	23.38	46.23	24.31	26.62	**66.38**
100	27.23	23.38	48.23	25.15	26.38	**67.69**
Time(s)	2.4×10^{1}	4.5×10^{1}	5.7×10^{1}	4.7×10^{1}	6.9×10^{-1}	1.9×10^{1}

TABLE 5. Experimental results of HSIC-NDR on the unobstructed images dataset of AR using different kernel function.

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	30	26.86	**51.71**	30.43	21.43	31.00	28.29	29.43	0.71	24.86
20	46	45.57	**58.14**	47.00	31.57	42.57	45.29	47	1.14	8.43
30	58.43	57	**61.29**	61.29	35.71	49.57	58	59.29	0.86	4.71
40	63.14	63.14	65.00	**66.29**	38.29	50.43	61.57	62.29	1	3.00
50	63.57	63	**69.29**	68.14	40.57	54.43	63.14	64	1.43	2.71
60	63.71	63.71	**73.86**	69.71	43.14	55.29	65.86	67	1	2.86
70	64.86	63.14	**73.14**	69.43	44.00	54.29	66.86	67.57	1.71	2.29
80	63.29	63.71	**74.29**	70.86	45.86	52.57	65.43	67.14	1.14	2.43
90	63.29	61.71	**73.14**	71.57	47.00	52.00	66.57	67.14	1.29	1.43
100	62	61.71	68.43	**69.86**	49.29	51.57	66.14	66.86	1.43	1.14
Time(s)	4.3×10^{0}	4.3×10^{0}	1.8×10^{1}	9.9×10^{1}	2.2×10^{1}	4.2×10^{0}	3.9×10^{2}	2.6×10^{1}	2.3×10^{1}	4.0×10^{0}

TABLE 6. Experimental results of HSIC-NDR compare with other algorithms on the unobstructed images dataset of AR. (Without dimensionality reduction, the accuracy of direct classification is 35.71%).

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	32.57	27.86	23	29.14	27	**51.71**
20	36.43	30	27	32.86	33.71	**58.14**
30	38.29	30.86	30.14	35.29	34.29	**61.29**
40	39.57	31.71	33.29	37.14	35.57	**66.29**
50	38.29	31.86	41.43	35.86	36.86	**69.29**
60	38	33.29	50.14	37.29	37	**73.86**
70	37.86	33.86	56.43	38.14	36.29	**73.14**
80	37.86	33.43	60.43	40.71	36.57	**74.29**
90	36.57	33.71	63.71	39.29	36.86	**73.14**
100	36.57	34.43	62.57	41.29	37.14	**69.86**
Time(s)	7.9×10^{1}	1.2×10^{1}	1.9×10^{1}	1.8×10^{1}	4.1×10^{-1}	1.3×10^{1}

of the images is used as the test set. For the classification result shown in Table 11 and 12, it randomly takes 5 images of each person as the training set, the rest of the images is used as the test set. A total of 10 randomized experiments has been run and the average of 10 randomized experimental results is taken as the final experimental results.

In Table 7, 8, 9, 10, 11 and 12, the first column is the number of dimension after dimension reduction, and the number of dimension of the original data is $32 \times 32 = 1024$. The highest accuracy rate for each dimensionality has been bolded in Table 7, 9 and 11. In Table 8, 10 and 12, the accuracy of the HSIC-NDR is the best accuracy picking from Table 7, 9 and 11 respectively.

6) VEHICLE

Vehicle dataset is grouped into four categories, each containing 199-218 samples, for a total of 846 samples.

TABLE 7. Experimental results of HSIC-NDR on ORL dataset using different kernel functions. (3 images per person are taken as the training samples, the other 7 images are taken as the test samples.)

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	56.32	60.57	**80.25**	60.21	54.96	7.57	62.86	0	62.46	18.14
20	68.93	73.64	**84.25**	71.11	67.29	66.96	63.21	72.71	71.46	12.14
30	66.64	72.43	**78.32**	67	70.43	63.57	73.93	72.18	71.07	14.32
40	63.86	69.68	**71.75**	61.71	69.82	60.82	70.71	68.89	68.86	13.93
50	61.86	68.4	62.14	61.11	68.07	55.89	66.43	**68.46**	67.64	12.25
60	57.18	68.04	53.61	57.36	66.04	53.96	**68.93**	66.61	64.64	11.29
70	53.79	62.71	43.64	51.64	63.21	51.79	**64.29**	62.07	61.11	9.57
80	49.71	**60.86**	36.96	45.07	59.11	49.43	60	57.54	59.25	10.25
90	44.07	**57.89**	30.64	41.61	57.86	42.71	55	56.54	52.86	8.93
100	43.18	55.07	25.46	42.07	53.54	43.21	**57.86**	51.29	54.57	8.32
Times(s)	3.2×10^{-1}	2.9×10^{-1}	6.9×10^{-1}	7.1×10^{-1}	6.5×10^{-1}	5.5×10^{-1}	1.0×10^{1}	9.8×10^{-1}	7.5×10^{-1}	5.7×10^{-1}

TABLE 8. Experimental results of HSIC-NDR compare with other algorithms on ORL dataset. (3 images per person are taken as the training samples, the other 7 images are taken as the test samples. Without dimensionality reduction, the accuracy of direct classification is 61.79%).

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	58.61	60.75	47.18	53.14	53.29	**80.25**
20	62.64	60.39	57.93	63.18	60.43	**84.25**
30	65.25	61.79	65.32	62.54	62.71	**78.32**
40	64.54	62.18	65.79	64.54	63.57	**71.75**
50	63	60.21	63.32	67.5	63.14	**68.46**
60	64.11	59.25	58.18	63.68	64.57	**68.93**
70	62.07	60.46	55.11	**65.71**	64.79	64.29
80	61.96	60.43	54.07	63	**63.86**	60.86
90	61.86	61.18	47.93	63.21	**63.5**	57.89
100	**62.68**	59.21	47.04	61.5	62.11	57.86
Time(s)	6.1×10^{-1}	5.1×10^{-1}	1.5×10^{0}	1.0×10^{0}	9.0×10^{-2}	3.6×10^{0}

TABLE 9. Experimental results of HSIC-NDR on ORL dataset using different kernel functions. (4 images per person are taken as the training samples, the other 6 images are taken as the test samples.)

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	63.12	67.5	**83.67**	66.29	62.17	64.42	63.75	7.71	69.96	9.71
20	77.29	81.21	**88.75**	77.67	74.08	73.83	71.25	81.88	77.54	13.71
30	74.83	79.42	**84.33**	75.88	77.29	72.58	70.83	79.46	79.46	18.21
40	73.08	77.92	**78.50**	71.63	76.5	67.96	73.75	76.54	77.38	17.25
50	70.67	77.04	71.79	70.79	**78.42**	66.54	76.25	76.08	75.92	12.63
60	68.42	76.67	63.33	66.33	**77.62**	63.12	71.25	73.33	73.63	10.63
70	64.25	74.04	54.79	63.58	73.13	58.29	**75**	71.92	71.92	8.92
80	56.88	71.25	46.00	57.25	**71.83**	53.5	67.92	67.04	68.71	8.63
90	54.46	69.13	37.46	51.46	**71.08**	52.5	68.75	66.46	65.63	7.92
100	50.75	65.08	30.92	50.13	67.04	47.79	57.92	65.04	**67.67**	6.67
Times(s)	3.4×10^{-1}	3.3×10^{-1}	7.0×10^{-1}	8.1×10^{-1}	5.5×10^{-1}	3.5×10^{-1}	1.2×10^{1}	8.8×10^{-1}	5.5×10^{-1}	3.7×10^{-1}

The download URL of Vehicle dataset is `http://archive.ics.uci.edu/ml/datasets/Statlog+%28Vehicle+Silhouettes%29`.

When running the experiment, it randomly takes 100 samples from each category as the training set, and the rest is used as the test set. A total of 10 randomized experiments has been run and the average of 10 randomized experimental results is taken as the final experimental results. A 3-NN classifier is used for classification.

Table 13 shows the experimental results of HSIC-NDR using different kernel functions. In Table 13 and 14, the first column is the number of dimension after dimension

TABLE 10. Experimental results of HSIC-NDR compare with other algorithms on ORL dataset. (4 images per person are taken as the training samples, the other 6 images are taken as the test samples. Without dimension reduction, the accuracy of direct classification is 72.67%.)

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	66.04	64	54.54	57.08	60.92	**83.67**
20	72.92	67.88	67.5	68.46	70.33	**88.75**
30	72.17	66.63	72.54	68.17	71.92	**84.33**
40	70.88	66.21	73.04	69.42	71.33	**78.50**
50	72.13	65.75	69.79	71.04	72.75	**78.42**
60	71.04	65.54	65.08	71.25	71.29	**77.62**
70	71.17	65.83	61.92	73.08	72.04	**75**
80	70.25	64.96	58.75	**72.21**	71.88	71.83
90	71.83	65.33	56.08	**73.13**	71.75	71.08
100	71.67	66.08	55	**69.83**	71.83	67.67
Time(s)	5.7×10^{-1}	5.5×10^{-1}	1.7×10^{1}	1.2×10^{0}	1.2×10^{-1}	1.8×10^{0}

TABLE 11. Experimental results of HSIC-NDR on ORL dataset using different kernel functions. (5 images per person are taken as the training samples, the other 5 images are taken as the test samples.)

Dimens-	Kernels								
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	71.4	73	**78.75**	73.85	67.8	70	72.5	73.5	76.15	22.00
20	81.6	84.56	79.60	82.2	80.05	80.6	81	**84.9**	81.25	15.10
30	80	83.6	79.50	81.05	83	77.15	**87.5**	82.35	85.5	21.00
40	79.6	82.75	76.75	79.35	81.75	73.3	82	83.5	**82.9**	17.15
50	77.6	83.75	77.25	76.95	**84.55**	70.4	80.5	81.55	83.05	14.55
60	75.65	**82.75**	77.40	74.6	82.3	70.3	78.5	79.6	82.65	12.50
70	71.25	80.9	74.30	71.05	81.45	67.95	**82**	79.15	80.25	9.35
80	66.4	78.45	72.15	63.4	**79.4**	63.1	79	77.15	76.2	8.95
90	63.85	76.45	67.25	62.15	**76.5**	55.8	76	74.3	74.15	7.45
100	59.55	**75.55**	62.05	55.95	75.35	53.5	75.5	72.3	75	5.55
Times(s)	3.6×10^{-1}	4.1×10^{-1}	7.8×10^{-1}	7.1×10^{-1}	6.5×10^{-1}	3.8×10^{-1}	1.1×10^{1}	9.8×10^{-1}	5.4×10^{-1}	4.2×10^{-1}

TABLE 12. Experimental results of HSIC-NDR compare with other algorithms on ORL dataset. (5 images per person are taken as the training samples, the other 5 images are taken as the test samples. Without dimension reduction, the accuracy of direct classification is 77.15%.)

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	71.8	66.75	58.45	60.05	66.65	**78.75**
20	75.35	70.5	70.2	70.95	75.15	**84.9**
30	78.45	71.9	77.45	71.75	78.5	**87.5**
40	78.6	70.35	77.35	71.8	77.3	**82.9**
50	77.7	70.1	76.7	77.45	78.4	**84.55**
60	78.55	70.1	72.3	77.5	79	**82.75**
70	77.35	69.9	68.1	74.75	78.8	**82**
80	76.85	68.5	66.15	78.2	78.9	**79.4**
90	77.5	68.15	64.2	**79.3**	77.65	76.5
100	77.5	69.75	61.4	77.35	**77.5**	75.55
Time(s)	4.7×10^{-1}	5.7×10^{-1}	1.3×10^{1}	1.5×10^{0}	1.5×10^{-1}	1.4×10^{0}

reduction, and the number of dimension of the original data is 18. The highest accuracy rate for each dimensionality has been bolded. Table 14 shows the comparison experimental results of HSIC-NDR and other algorithms. In Table 14, the accuracy of the HSIC-NDR is the best accuracy picking from Table 13. HSIC-NDR outperforms other algorithms in different dimensions in the experiments.

7) MNIST

The MNIST database of handwritten digits, available from `http://yann.lecun.com/exdb/mnist/`, has a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of a larger set available from NIST. For computational reasons, we selected the first 2,000 digits for our experiments. The digits have been

TABLE 13. Experimental results of HSIC-NDR on Vehicle dataset using different kernel functions .

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
2	52.35	52.02	**94.89**	51.28	44.55	51.46	51.79	49.69	46.73	77.40
3	49.62	53.36	**92.31**	48.9	52.53	50.83	51.12	52.11	52.33	90.85
4	49.28	51.12	**94.51**	47.09	52.15	48.61	49.1	52.67	52	87.47
5	58.5	60.43	**94.19**	47.56	51.14	58.18	61.88	59.8	52.06	89.22
6	58.45	60.65	**94.06**	48.36	51.93	58.34	62.33	60.9	57.2	91.82
7	62.91	65.27	**94.37**	56.82	51.3	57.8	63.23	63.88	59.96	85.74
8	67.06	66.32	**94.17**	62.35	53	60.85	69.73	65.81	65.54	78.14
9	66.17	67.67	**95.16**	67.06	52.96	66.39	67.04	67.42	69.08	75.96
10	66.66	68.12	**95.18**	65.25	54.24	67.47	66.37	67.78	70.2	92.78
11	68.45	69.78	**95.22**	65.43	54.98	66.26	68.83	69.44	69.06	90.78
12	70.16	70.96	**95.43**	66.26	54.04	66.88	70.63	70.85	71.12	89.55
13	70.36	71.61	**95.16**	65.29	54.69	67.96	70.4	70.9	72.2	89.78
14	73.16	73.16	**95.22**	66.59	54.35	66.39	71.75	73.43	73.92	87.78
15	74.64	74.22	**93.68**	67.35	54.33	67.47	74.66	72.62	73.57	84.46
16	75.34	76.59	**94.66**	67.31	54.33	69.64	76.01	73.65	74.51	83.09
17	75.67	76.37	**93.39**	67.78	55.9	69.8	75.78	76.59	75.07	79.98
Time(s)	1.4×10^{0}	1.5×10^{0}	1.8×10^{0}	1.5×10^{0}	1.5×10^{0}	1.5×10^{0}	2.5×10^{0}	3.1×10^{0}	1.9×10^{0}	1.4×10^{0}

TABLE 14. Experimental results of HSIC-NDR compare with other algorithms on Vehicle dataset. (Without dimensionality reduction, the accuracy of direct classification is 59.51%.)

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
2	51.12	51.91	50.04	50.61	53.23	**94.89**
3	53.74	52.89	50.49	50.11	53.36	**92.31**
4	58.77	53.63	51.32	49.24	56.28	**94.51**
5	58.34	55	50.45	54.44	57.4	**94.19**
6	59.73	56.46	50.13	55.11	58	**94.06**
7	59.19	57.13	52.15	63.9	58.97	**94.37**
8	59.26	57.29	54.13	64.44	59.28	**94.17**
9	59.39	56.5	56.57	69.82	59.37	**95.16**
10	60.31	58.12	65.83	69.93	58.81	**95.18**
11	60.11	58.09	68.88	68.59	59.78	**95.22**
12	58.77	58.88	69.22	72.87	60.4	**95.43**
13	59.24	58.59	71.39	73.5	60.36	**95.16**
14	59.33	58.16	70.47	74.73	59.91	**95.22**
15	58.83	59.15	74.01	75.54	59.19	**93.68**
16	59.64	58.92	75.07	73.12	60.18	**94.66**
17	60.18	58.05	74.62	73.83	59.1	**93.39**
Time(s)	2.2×10^{0}	4.3×10^{-1}	3.6×10^{-1}	3.0×10^{-2}	1.0×10^{-2}	1.9×10^{0}

size-normalized and centred in a fixed-size image. These images have 28×28 pixels, and can thus be considered as points in a 784-dimensional space.

When running the experiment, it randomly takes 1000 samples as the training set, and the rest is used as the test set. A total of 10 randomized experiments has been run and the average of 10 randomized experimental results is taken as the final experimental results. A 10-NN classifier is used for classification.

Table 15 shows the experimental results of HSIC-NDR using different kernel functions. In Table 15, the first column is the number of dimension after dimension reduction. The highest accuracy rate for each dimensionality has been bolded. Table 16 shows the comparison experimental results of HSIC-NDR and other algorithms. In Table 16, the accuracy of the HSIC-NDR is the best accuracy picking from Table 15.

8) BREAST CANCER

The breast cancer dataset is a classic and very easy binary classification dataset obtained from `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`. There are

TABLE 15. Experimental results of HSIC-NDR on MNIST dataset using different kernel functions.

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
10	84.75	84.75	**88.02**	85.72	82.78	80.62	84.68	85.3	11.47	9.5
20	88.09	88.33	89.2	**89.31**	88.38	83.16	88.45	88.6	9.95	9.87
30	86.54	86.72	**89.45**	88.95	88.88	82.3	86.89	88.55	9.89	9.43
40	83.38	83.69	88.88	89.15	**89.31**	80.91	83.92	87.02	10.94	9.9
50	80.13	81.87	88.4	88.93	**89.67**	79.64	81.48	86.72	10.35	9.63
60	76.79	81.66	87.63	88.82	**89.72**	77.76	78.43	86.09	9.8	9.64
70	71.24	81.45	86.82	89.32	**89.55**	76.14	74.13	84.98	10.03	9.53
80	66.11	81.69	85.98	**89.49**	89.44	73.66	69.93	84.37	9.81	9.6
90	59.37	81.06	84.88	89.51	**89.6**	71.85	66.84	82.72	9.49	9.87
100	53.38	80.37	83.48	**89.91**	89.64	68.97	67.81	79.72	9.37	10.01
Time(s)	9.8×10^{0}	1.1×10^{1}	2.0×10^{1}	1.6×10^{1}	1.1×10^{1}	9.8×10^{0}	1.6×10^{2}	1.9×10^{1}	1.5×10^{1}	9.8×10^{1}

TABLE 16. Experimental results of HSIC-NDR compare with other algorithms on MNIST dataset. (Without dimensionality reduction, the accuracy of direct classification is 85.99%.)

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
10	48.14	84.36	76.62	19.94	85.59	**88.02**
20	52.85	85.57	82.74	19.86	**89.52**	89.31
30	56.64	85.91	84.21	19.88	88.73	**89.45**
40	58.87	85.49	84.69	19.88	88.4	**89.31**
50	57.43	85.68	84.91	19.88	88.03	**89.67**
60	59.34	85.45	84.96	19.88	87.5	**89.72**
70	60.24	85.25	85.18	19.89	87.44	**89.55**
80	60.67	85.08	85.02	19.88	87.4	**89.49**
90	57.88	84.76	84.51	19.89	87.27	**89.6**
100	61.64	84.61	83.65	19.88	87.07	**89.91**
Time(s)	1.3×10^{1}	1.0×10^{1}	1.2×10^{1}	7.8×10^{0}	1.9×10^{-1}	1.4×10^{1}

TABLE 17. Experimental results of HSIC-NDR on Breast Cancer dataset dataset using different kernel functions .

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
5	93.79	57.25	64.31	93.01	91.08	**94.35**	92.38	91.34	66.77	65.84
6	92.64	57.25	64.31	93.05	91.49	**94.57**	93.27	91.41	65.43	65.84
7	91.3	61.08	66.47	93.46	91.56	**94.57**	93.31	91.71	67.96	65.84
8	89.96	86.73	86.99	92.97	91.52	**94.57**	93.42	92.57	66.77	65.84
9	90.07	93.38	93.2	93.27	91.67	**94.57**	92.68	92.57	65.43	65.84
10	88.7	92.64	91.9	94.39	91.71	**94.57**	92.6	92.6	65.24	65.84
11	90.07	93.75	91.45	93.98	91.45	**94.57**	92.64	92.6	64.13	65.84
12	88.85	94.01	91.3	94.42	91.23	**94.57**	92.27	92.27	65.35	65.84
13	87.25	93.68	91.3	94.09	91.75	**94.57**	92.49	92.27	62.68	65.84
14	86.51	93.68	91.3	93.72	92.45	**94.57**	92.3	92.12	62.45	65.84
Time(s)	4.0×10^{-1}	4.1×10^{-1}	5.2×10^{-1}	4.3×10^{-1}	4.0×10^{-1}	4.0×10^{-1}	1.0×10^{0}	1.0×10^{0}	6.9×10^{-1}	4.1×10^{-1}

569 samples in total. And 30 features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. There are 212 samples are diagnosed as Malignant, and the left 357 are diagnosed as Benign.

When running the experiment, it randomly takes 300 samples as the training set, and the rest is used as the test set. A 5-NN classifier is used for classification and a total of 10 randomized experiments have been run and the average of 10 randomized experimental results is taken as the final experimental results.

Table 17 shows the experimental results of HSIC-NDR using different kernel functions. Comparison experimental results of HSIC-NDR and other algorithms are shown in Table 18. The first column in Table 17 and Table 18 is the number of Dimensionality after dimension reduction. The highest accuracy rate for each dimensionality has been

TABLE 18. Experimental results of HSIC-NDR compare with other algorithms on Breast Cancer dataset. (Without dimension reduction, the accuracy of direct classification is 92.49%.)

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
5	48.2	84.36	76.62	19.94	85.57	**94.35**
6	54.13	85.57	82.74	19.86	89.49	**94.57**
7	55.84	85.91	84.21	19.88	88.73	**94.57**
8	57.74	85.49	84.69	19.88	88.39	**94.57**
9	57.64	85.68	84.91	19.88	88.08	**94.57**
10	57.53	85.45	84.96	19.88	87.59	**94.57**
11	60.8	85.25	85.18	19.89	87.37	**94.57**
12	59.04	85.08	85.02	19.88	87.42	**94.57**
13	59.03	84.76	84.51	19.89	87.18	**94.57**
14	60.1	84.61	83.65	19.88	87.02	**94.57**
Time(s)	1.4×10^{1}	1.0×10^{1}	1.2×10^{1}	7.9×10^{0}	1.9×10^{-1}	4.0×10^{-1}

TABLE 19. Experimental results of HSIC-NDR on Wine dataset using different kernel functions .

Dimens-	Kernels									
ionality	lin	rbf	rbf-geo	chi2	tst	poly	wavelet	wave	bspline	sigmod
2	73.72	73.97	75.26	**85.77**	75	74.36	74.36	67.56	45	68.59
3	81.03	81.15	86.79	**85**	64.74	79.74	73.72	67.31	38.21	82.18
4	91.41	92.18	88.46	**93.97**	66.92	91.79	80.26	73.33	35.13	81.28
5	93.46	93.46	88.59	**95.38**	67.31	91.54	91.79	77.31	34.36	85
6	93.46	93.46	86.54	**94.23**	68.33	93.08	92.31	81.15	40.9	83.21
7	94.62	94.62	83.08	**94.87**	68.72	93.33	91.79	91.79	38.33	81.54
8	**94.36**	94.23	81.03	94.1	67.31	93.72	91.54	91.54	36.03	78.33
9	92.18	93.59	87.31	**94.1**	66.28	93.21	93.08	91.54	36.54	77.95
10	92.18	94.36	89.1	**94.36**	66.92	92.44	94.1	93.97	37.56	74.1
11	92.05	92.44	**95.51**	93.97	67.05	90.38	92.18	93.33	39.36	74.1
12	90.51	91.79	**95.51**	93.59	66.54	90.38	92.31	93.59	38.72	73.08
13	89.74	92.69	**95.38**	93.59	66.41	89.49	92.69	94.36	36.03	72.44
Time(s)	8.0×10^{-2}	8.2×10^{-2}	1.0×10^{-1}	8.2×10^{-2}	8.5×10^{-2}	9.0×10^{-2}	1.0×10^{-1}	1.5×10^{-1}	1.1×10^{-1}	8.2×10^{-2}

TABLE 20. Experimental results of HSIC-NDR compare with other algorithms on Wine dataset. (Without dimension reduction, the accuracy of direct classification is 68.87%.)

	Methods					
Dimensionality	MDS	ISOMAP	LTSA	LPP	PCA	HSIC-NDR
2	69.49	70.51	49.49	64.49	69.23	**85.77**
3	67.95	70.64	58.46	73.21	69.49	**86.79**
4	68.72	70.77	68.46	90.26	69.62	**93.97**
5	69.87	71.15	65	88.59	69.87	**95.38**
6	70.51	71.28	64.62	90	69.87	**94.23**
7	69.23	71.28	65.77	90.13	69.87	**94.87**
8	68.72	71.28	64.62	90.13	69.87	**94.36**
9	68.72	71.54	62.82	90.9	69.87	**94.1**
10	68.85	70.77	63.97	90.13	69.87	**94.36**
11	68.97	70.64	66.03	89.62	69.87	**95.51**
12	69.36	71.03	66.54	88.21	69.87	**95.51**
13	68.21	71.15	66.92	88.46	69.87	**95.38**
Time(s)	8.5×10^{-2}	2.1×10^{-2}	5.2×10^{-2}	1.3×10^{-2}	1.0×10^{-2}	9.0×10^{-2}

bolded. In Table 18, the accuracy of the HSIC-NDR is the best accuracy picking from Table 17.

9) WINE

The Wine dataset includes results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. The dataset is available on `https://archive.ics.uci.edu/ml/datasets/wine` and there are 178 instances.

When running the experiment, it randomly takes 100 samples as the training set, and the rest is used as the test set. A 5-NN classifier is used for classification and a total of 10 randomized experiments have been run and the average of 10 randomized experimental results is taken as the final experimental results.

Table 19 shows the experimental results of HSIC-NDR using different kernel functions. Comparison experimental results of HSIC-NDR and other algorithms are shown in Table 20. The first column in Table 19 and Table 20 are the number of Dimensionality after dimension reduction. The highest accuracy rate for each dimensionality has been bolded. In Table 20, the accuracy of the HSIC-NDR is the best accuracy picking from Table 19.

VII. CONCLUSIONS

1) The theory of HSIC may sound a little complicated, which may affect the wide application of HSIC to a certain extent. This paper is brief, but completely and accurately introduces the HSIC theory. As long as one has the basic knowledge of function analysis, through this paper, he/she should have a clear understanding of the ins and outs of HSIC theory.
2) So far, HSIC has not been directly applied to data dimensionality reduction. There are some HSIC applications similar to data dimensionality reduction, such as supervised feature selection based on HSIC [33]–[36] dictionary learning based on HSIC [37], [38], and supervised subspace learning based on HISC [40]. However, these HSIC-based methods are essentially different from the HSIC-NDR method proposed in this paper. First, these methods are all supervised machine learning methods, while the proposed HSIC-NDR algorithm is an unsupervised machine learning method. Secondly, the prerequisite for the supervised feature selection is that the feature of the data has been determined. The feature selection method is based on the existing features. However, the dimension reduction data to be determined by the HSIC-NDR algorithm is unknown. And it is sought through the optimization algorithm according to certain criteria (the data after dimension reduction and the original data maintain the maximum statistical dependence criterion). In terms of supervised dictionary learning or subspace learning, the high-dimensional data and dimensionality-reduced data are limited to a linear relationship. That means the HSICs of high-dimensional data and dimensionality-reduced data are constant and cannot be used as a basis for dictionary or subspace selection. They are based on maximizing the HSIC of dimensionality reduction data and data Labels as a dictionary or subspace selection. The HSIC-NDR algorithm proposed in this paper directly maximizes the HSIC between high dimensional data and dimensionality reduction data as the basis for data dimensionality reduction. Therefore, our dimensionality reduction algorithm is a nonlinear data dimension reduction algorithm.
3) In the framework of HSIC, there are two kernel functions, so two RKHS spaces are generated, which are the workspaces of two sets of data before and after dimension reduction respectively. In the proposed HSIC-NDR algorithm, the kernel function used in the dimension reduction result is defined as a linear kernel, so that the objective function of the algorithm can be transformed into the form of the Rayleigh quotient. Also, the linear kernel matrix of the dimension reduction data is the covariance matrix of the dimension reduction data. The maximization of the data covariance matrix is helpful to improve the accuracy of data discrimination. In the proposed HSIC-NDR algorithm, the kernel function used for the original data (before dimension reduction) is optional. Hence the most suitable kernel function can be chosen according to the specific application. From this point of view, the proposed HSIC-NDR algorithm is a framework in which the kernel functions need to be determined based on the specific application. Hence it can be widely used.

REFERENCES

[1] F. R. S. K. Pearson, “LIII. On lines and planes of closest fit to systems of points in space,” *London, Edinburgh, Dublin Philosoph. Mag. J. Sci.*, vol. 2, no. 11, pp. 559–572, 1901.

[2] W. S. Torgerson, “Multidimensional scaling: I. Theory and method,” *Psychometrika*, vol. 17, no. 4, pp. 401–419, Dec. 1952.

[3] R. A. Fisher, “The use of multiple measurements in taxonomic problems,” *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[4] P. Switzer and A. A. Green, “Min/max autocorrelation factors for multivariate spatial imagery,” *Comput. Sci. Statist.*, vol. 16, pp. 13–16, 1984.

[5] L. Wiskott and T. J. Sejnowski, “Slow feature analysis: Unsupervised learning of invariances,” *Neural Comput.*, vol. 14, no. 4, pp. 715–770, Apr. 2002.

[6] K. Fukumizu, F. R. Bach, and M. I. Jordan, “Dimensionality reduction for supervised learning with reproducing kernel Hilbert spaces,” *J. Mach. Learn. Res.*, vol. 5, pp. 73–99, Jan. 2004.

[7] A. Hyvärinen, J. Karhunen, and E. Oja, *What is Independent Component Analysis?* Hoboken, NJ, USA: Wiley, 2002, ch. 7, pp. 145–164. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/0471221317.ch7

[8] L. Yang and R. Jin, “Distance metric learning: A comprehensive survey,” Michigan State Univ., Tech. Rep., 2006. [Online]. Available: https://www.cse.msu.edu/~yangliu1/frame_survey_v2.pdf

[9] B. Schölkopf, A. Smola, and K.-R. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, 1998.

[10] S. Mika, G. Ratsch, J. Weston, B. Schölkopf, and K.-R. Mullers, “Fisher discriminant analysis with kernels,” in *Proc. IEEE Signal Process. Soc. Workshop Neural Netw. Signal Process.*, 1999, pp. 41–48.

[11] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[12] Z. Zhang and H. Zha, “Principal manifolds and nonlinear dimensionality reduction via tangent space alignment,” *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, 2005.

[13] X. He and P. Niyogi, “Locality preserving projections,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 153–160.

[14] M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 585–591.

[15] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[16] D. L. Donoho and C. Grimes, “Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data,” *Proc. Nat. Acad. Sci. USA*, vol. 100, no. 10, pp. 5591–5596, 2003.

[17] S. Lafon and A. B. Lee, "Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1393–1403, Sep. 2006.

[18] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Trans. Comput.*, vol. C-18, no. 5, pp. 401–409, May 1969.

[19] G. E. Hinton and S. T. Roweis, "Stochastic neighbor embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 857–864.

[20] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: A comparative," *J. Mach. Learn. Res.*, vol. 10, pp. 66–71, Oct. 2009.

[21] J. P. Cunningham and Z. Ghahramani, "Linear dimensionality reduction: Survey, insights, and generalizations," *J. Mach. Learn. Res.*, vol. 16, pp. 2859–2900, 2015.

[22] M. G. B. Blum, M. A. Nunes, D. Prangle, and S. A. Sisson, "A comparative review of dimension reduction methods in approximate bayesian computation," *Stat. Sci.*, vol. 28, no. 2, pp. 189–208, 2013.

[23] W. Sun, L. Tian, Y. Xu, D. Zhang, and Q. Du, "Fast and robust self-representation method for hyperspectral band selection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 11, pp. 5087–5098, Nov. 2017.

[24] W. Sun, M. Jiang, and W. Li, "Band selection using sparse self-representation for hyperspectral imagery," *Geomatics Inf. Sci. Wuhan Univ.*, vol. 42, no. 4, pp. 441–448, 2017.

[25] W. Sun, G. Yang, B. Du, L. Zhang, and L. Zhang, "A sparse and low-rank near-isometric linear embedding method for feature extraction in hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 4032–4046, Jul. 2017.

[26] W. Sun and Q. Du, "Graph-regularized fast and robust principal component analysis for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3185–3195, Jun. 2018.

[27] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf, "Measuring statistical dependence with Hilbert–Schmidt norms," in *Proc. Int. Conf. Algorithmic Learn. Theory*. Basel, Switzerland: Birkhäuser, 2005, pp. 63–77.

[28] E. Kreyszig, *Introductory Functional Analysis With Applications*, vol. 1. New York, NY, USA: Wiley, 1978.

[29] L. Song, A. Gretton, K. M. Borgwardt, and A. J. Smola, "Colored maximum variance unfolding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1385–1392.

[30] L. Song, A. Smola, A. Gretton, K. M. Borgwardt, and J. Bedo, "Supervised feature selection via dependence estimation," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 823–830.

[31] B. B. Damodaran, N. Courty, and S. Lefèvre, "Sparse Hilbert Schmidt independence criterion and surrogate-kernel-based feature selection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 4, pp. 2385–2398, Apr. 2017.

[32] K. Zhang, V. W. Zheng, Q. Wang, J. T. Kwok, Q. Yang, and I. Marsic, "Covariate shift in Hilbert space: A solution via surrogate kernels," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 388–395.

[33] M. J. Gangeh, H. Zarkoob, and A. Ghodsi, "Fast and scalable feature selection for gene expression data using hilbert-schmidt independence criterion," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 14, no. 1, pp. 167–181, Jan./Feb. 2017.

[34] Y. Dodge, Ed., *Statistical Data Analysis Based on the L_1-Norm and Related Methods*. Cambridge, MA, USA: Birkhäuser, 2012.

[35] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt, "Feature selection via dependence maximization," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 1393–1434, Jan. 2012.

[36] G. Camps-Valls, J. Mooij, and B. Schölkopf, "Remote sensing feature selection by kernel dependence measures," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 3, pp. 587–591, Jul. 2010.

[37] M. J. Gangeh, A. Ghodsi, and M. S. Kamel, "Kernelized supervised dictionary learning," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4753–4767, Oct. 2013.

[38] M. J. Gangeh, P. Fewzee, A. Ghodsi, M. S. Kamel, and F. Karray, "Multiview supervised dictionary learning in speech emotion recognition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 6, pp. 1056–1068, Jun. 2014.

[39] E. Oja, *Subspace Methods of Pattern Recognition* (Pattern Recognition & Image Processing Series), vol. 6. Baldock, U.K.: Research Studies Press, 1983.

[40] E. Barshan, A. Ghodsi, Z. Azimifar, and M. Z. Jahromi, "Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds," *Pattern Recognit.*, vol. 44, no. 7, pp. 1357–1371, 2011.

[41] M. Hu, Y. Chen, and J. T.-Y. Kwok, "Building sparse multiple-kernel SVM classifiers," *IEEE Trans. Neural Netw.*, vol. 20, no. 5, pp. 827–839, May 2009.

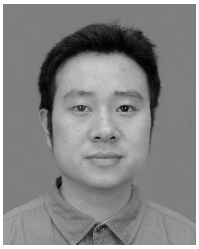
[42] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[43] I. Gohberg, S. Goldberg, and M. A. Kaashoek, "Hilbert-Schmidt operator," in *Classes of Linear Operators*, vol. 1. Berlin, Germany: Springer, 1990, pp. 138–147.

[44] M. Hein and O. Bousquet, "Kernels, associated structures and generalizations," Max-Planck-Inst. Biologische Kybernetik, Tübingen, Germany, Tech. Rep. 127, 2004.

[45] A. M. Mood, *Introduction to the Theory of Statistics*. New York, NY, USA: McGraw-Hill, 1950.

ZHENGMING MA received the B.Sc. and M.Sc. degrees from the South China University of Technology, Guangzhou, China, in 1982 and 1985, respectively, and the Ph.D. degree in pattern recognition and intelligent control from Tsinghua University, Beijing, China, in 1989. He is currently a Professor with the Nanfang College, Sun Yat-sen University, Guangzhou. His current research interests include machine learning and signal processing.



ZENGRONG ZHAN received the B.Sc. degree from Guangdong Polytechnic Normal University, Guangzhou, China, in 2005, and the M.Sc. degree in dependable computer system from the Chalmers University of Technology, Gothenburg, Sweden, in 2007. He is currently pursuing the Ph.D. degree with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou. His current research interest is machine learning.

XIAOYUAN OUYANG received the bachelor's degree in communication engineering from Wuyi University, Jiangmen, China, in 2015. He is currently pursuing the master's degree with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China. His current research interest is machine learning.

XUE SU received the B.Sc. degree in electronic and information from Yangtze University, Jingzhou, China, in 2017. She is currently pursuing the master's degree with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou. Her current research interests include machine learning and pattern recognition.

• • •