International Journal of Innovative Computing, Information and Control
Volume **19**, Number **3**, June **2023**

ICIC International ©2023 ISSN 1349-4198
pp. **927–939**

UNSUPERVISED FEATURE SELECTION WITH HILBERT-SCHMIDT INDEPENDENCE CRITERION LASSO

Tinghua Wang*, Zhenwei Hu and Huiying Zhou

School of Mathematics and Computer Science
Gannan Normal University
Shida South Road, Rongjiang New District, Ganzhou 341000, P. R. China
{ 1200714004; 1200780005 }@gnnu.edu.cn
*Corresponding author: wangtinghua@gnnu.edu.cn

Received September 2022; revised December 2022

Abstract. *In recent years, it has been witnessed that feature selection can be tackled with the Hilbert-Schmidt independence criterion (HSIC) due to its effectiveness and low computational complexity. However, most of the HSIC-based feature selection methods suffer from the following limitations. First, these methods are usually just applicable to labeled data, which is not desirable since most of data in real-world applications is unlabeled. Second, existing HSIC-based unsupervised feature selection methods only addressed the general correlation between the selected features and output values that express the underlying cluster structure, while the redundancy between different features was neglected. To address these problems, we present an unsupervised feature selection method based on the HSIC least absolute shrinkage and selection operator (HSIC Lasso), which not only has a clear statistical interpretation that minimum redundant features with maximum dependence on output values can be found in terms of the HSIC, but also enables the global optimal solution to be computed efficiently by solving a Lasso optimization problem. Based on the proposed method, a unified view of feature selection based on the HSIC Lasso is also discussed. The proposed method was demonstrated with several benchmark examples.*

Keywords: Hilbert-Schmidt independence criterion (HSIC), Least absolute shrinkage and selection operator (Lasso), Feature selection, Unsupervised learning, Kernel method

1. **Introduction.** Feature selection aims to identify a subset of features from the original features for model building or data understanding [1,2]. In the big data era, feature selection is becoming increasingly important, since high-dimensional data is ubiquitous in various real-world applications. The high dimensionality usually not only incurs high computational cost and more difficulty in model interpretation, but also deteriorates the generalization ability of prediction models [3-5]. Therefore, it is necessary to perform feature selection before actual model learning. Traditionally, there are two classification methods for feature selection techniques from different perspectives [1,2]. According to the availability of supervision information such as class labels in classification problems, feature selection can be broadly categorized as supervised, unsupervised and semi-supervised methods. Supervised feature selection works when sufficient label information is available while unsupervised feature selection algorithms do not require any class labels. Semi-supervised feature selection is a trade-off between supervised and unsupervised methods which can exploit both labeled and unlabeled data when a limited number of labeled data are available. Concerning different selection strategies, feature selection algorithms

DOI: 10.24507/ijicic.19.03.927

are generally divided into filter, wrapper and embedded methods. Filter methods select subsets of features as a pre-processing step, independently of the chosen learning machine. Wrapper methods utilize the learning performance of the chosen predictor to assess the quality of selected features. Embedded methods take advantage of the intrinsic structure of a learning algorithm to embed feature selection into model learning. Among these methods, filter feature selection is attracting more attention than ever, owing to its fast processing speed, independence of learning models, and relative robustness against overfitting, although it may fail to select the best feature subset for the learning models [5].

A general filter feature selection method consists of two components: an evaluation criterion for measuring the importance of features and a search strategy for generating feature subset. This evaluation criterion is to estimate how important or useful a feature or feature subset may be when used in a learning machine. Since the filter feature selection is carried out independently of the chosen learning machine, an effective and efficient evaluation criterion plays a critical role in filter methods. In the past decades, different evaluation criteria have been developed, such as those based on distance [6], consistency [7], and dependency [8]. Among them, the dependency-based criteria are most widely used and investigated in practice. The simplest and most common dependence criterion is the Pearson correlation coefficient. However, it can only capture linear relationships between paired data. Another popular criterion is the mutual information [9-12], which can capture nonlinear relationships and has good theoretical justification. However, it is difficult to estimate since it requires distribution estimation, which is problematic for high dimensional data and continuous variables. To address this limitation, the Hilbert-Schmidt independence criterion (HSIC) [13], which is defined as the Hilbert-Schmidt norm of the cross covariance operator between reproducing kernel Hilbert spaces (RKHSs), has been introduced as an alternative to mutual information. Unlike mutual information, the HSIC does not require explicitly estimating distributions although it can be interpreted as the distance between embeddings of the joint distribution and product of the marginal distributions. With several key advantages over other classical metrics on distributions, namely easy computability, fast convergence and low bias of finite sample estimates, HSIC has been widely used for a variety of learning problems [14], such as feature selection, dimensionality reduction, clustering, and kernel learning and optimization. In this paper, we focus on the feature selection problem.

In addition to the evaluation criteria, the performance of filter methods depends heavily on the search strategies for generating feature subset. In essence, feature selection is an NP-hard combinatorial optimization problem over a discrete space. Let d be the number of original features. An exhaustive search over 2^d possible feature subsets is computationally impractical. In the HSIC-based feature selection community, there are generally three methodologies to perform effective and efficient filter feature selection. *The first methodology* employs heuristic search strategies, such as greedy approaches. Song et al. [3,15,16] proposed a supervised feature selection algorithm based on the HSIC and backward elimination, called BAHSIC. Specifically, given a set of features, BAHSIC uses the HSIC to evaluate the dependence between the data obtained from these features and the response variables (labels). Having obtained this measure of dependence, it utilizes backward elimination to select a subset of the most relevant features. In contrast to the backward elimination technique that tries to increase the HSIC as much as possible for each deletion of features, forward selection using HSIC (FOHSIC) tries to achieve this for each inclusion of features. However, these strategies tend to produce local optima. *The second methodology* applies stochastic search techniques, such as genetic algorithm and bat algorithm, to searching for a globally optimal feature subset [17,18]. The stochastic search

strategies can generally help escape from local optima by adding some randomness in the search procedure. Instead of solving feature selection as a discrete optimization, *the third methodology* formulates the problem as a continuous optimization problem [3,15,19-21]. Given a weight vector $\boldsymbol{w} = (w_1, \ldots, w_d)^{\mathrm{T}} \in \mathrm{R}^d$ (R denotes the set of real numbers) on the dimensions of the data, feature selection using HSIC can be formulated as maximizing the HSIC value subject to certain constraints on $\boldsymbol{w}$. These constraints are generally used to induce sparsity.

Rooted in the advantages of HSIC, an important property of feature selection using the HSIC is its generality. Taking the BAHSIC as an example, it has been shown that, by choosing appropriate preprocessing of the data and specific kernels on the data, BAHSIC is not only directly applicable to binary, multiclass, and regression problems, but also encapsulates many well-known feature selection criteria [3]. A variant of BAHSIC can be also applied to performing unsupervised feature selection [22,23]. In this case, the goal is to select a subset of available features such that it is strongly correlated with the full data. BAHSIC readily accommodates this by simply using the full data as the labels (pseudo lables). In other words, we want to maximize the dependence between the selected subset of features and the full set of features. However, this unsupervised method only addressed the general correlation between the selected features and output values, while the redundancy between different features was neglected. In this paper, inspired by the unsupervised BAHSIC and supervised high-dimensional feature selection by feature-wise kernelized least absolute shrinkage and selection operator (Lasso) [20], we present an HSIC-Lasso-based unsupervised feature selection method which is referred to as UHSIC-Lasso. The main contributions of the paper are outlined as follows.

- An unsupervised feature selection method based on the HSIC Lasso was proposed, which not only has a clear statistical interpretation that minimum redundant features with maximum dependence on output values can be found in terms of the HSIC, but also enables the global optimal solution to be computed efficiently by solving a Lasso optimization problem.
- A unified view of feature selection based on the HSIC Lasso is discussed. With this unified view, we are able to not only bridge the gap in current understanding of relationships among the supervised, unsupervised and semi-supervised feature selection methods, but also guide the design of new methodologies promising for feature selection.
- Experiments on real-world data sets from the UCI benchmark repository verify the effectiveness of the proposed UHSIC-Lasso method.

The rest of this paper is organized as follows. Brief introductions to kernel methods and HSIC are given in Section 2. The formulation and algorithm of the proposed HSIC-Lasso-based unsupervised feature selection are detailed in Section 3. Section 4 reports the experimental results, followed by the conclusion and further study in Section 5.

2. Kernel Methods and Hilbert-Schmidt Independence Criterion.

Kernel methods [24-26] represent a well-established learning paradigm that is able to capture the nonlinear complex patterns underlying data. These methods map data points from the input space to the feature space, i.e., higher dimensional RKHS, where even relatively simple algorithms, such as linear methods, can deliver very impressive performance. The mapping is determined implicitly by a kernel function (or simply a kernel), which computes the inner product of data points in the feature space. In other words, kernel methods generalize linear classification and regression tasks by effectively transforming the optimization over a set of linear functions into an optimization over an RKHS, which is entirely defined by the kernel. This leads to support vector machines (SVM), kernel

Fisher discriminant analysis (KFDA), kernel ridge regression (KRR), and many other now standard algorithms. The HSIC is also a kernel method which will be introduced briefly as follows.

2.1. **Measuring dependence using kernel methods.** Let $(\boldsymbol{x}, \boldsymbol{y})$ on $\mathrm{X} \times \mathrm{Y}$ (X and Y are separable metric spaces) be random variables jointly drawn from probability distribution P_{xy}, and the covariance matrix is given by

$$C_{xy} = \mathrm{E}_{xy}\left(\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}\right) - \mathrm{E}_x\left(\boldsymbol{x}\right)\mathrm{E}_y\left(\boldsymbol{y}^{\mathrm{T}}\right) \tag{1}$$

where E_{xy}, E_x and E_y are the expectations with respect to P_{xy} and the marginal probability distributions P_x and P_y, respectively, and $\boldsymbol{y}^{\mathrm{T}}$ is the transpose of $\boldsymbol{y}$. The covariance matrix encodes all second-order dependences between the random variables. A statistic that efficiently summarizes the degree of linear correlation between x and y is the Frobenius norm (also called Hilbert-Schmidt norm) of C_{xy}:

$$||C_{xy}||_{\mathrm{Fro}} = ||C_{xy}||_{\mathrm{HS}} = \sqrt{\mathrm{tr}\left(C_{xy}C_{xy}^{\mathrm{T}}\right)} \tag{2}$$

where $\mathrm{tr}(\cdot)$ is the trace operator. This quantity is zero if and only if there exists no linear dependence between x and y, and therefore can be applied to detecting linear dependence between them. However, this statistic is rather limited, especially when we are uncertain about the actual type of data we are dealing with, and the dependence relationship is nonlinear which might not be captured by the covariance at all [3].

To address these limitations, the notion of Frobenius norm of the covariance matrix was extended to the HSIC [13]. Given two feature maps $\phi : \mathrm{X} \to \mathrm{F}$ and $\varphi : \mathrm{Y} \to \mathrm{G}$ (F and G are RKHSs), it is possible to define a cross-covariance operator between ϕ and φ as linear operator $C_{xy} : \mathrm{G} \to \mathrm{F}$, such that

$$C_{xy} = \mathrm{E}_{xy}[[\phi(\boldsymbol{x}) - \mathrm{E}_x[\phi(\boldsymbol{x})]] \otimes [\varphi(\boldsymbol{y}) - \mathrm{E}_y[\varphi(\boldsymbol{y})]]] \tag{3}$$

where $\otimes$ is the tensor product. The HSIC is then defined as the square of the Hilbert-Schmidt norm of this cross-covariance operator:

$$\begin{aligned} \mathit{HSIC}(\mathrm{F}, \mathrm{G}, \mathrm{P}_{xy}) &= ||C_{xy}||_{\mathrm{HS}}^2 \\ &= \mathrm{E}_{xx'yy'}[k(\boldsymbol{x}, \boldsymbol{x}')l(\boldsymbol{y}, \boldsymbol{y}')] - 2\mathrm{E}_{xy}[\mathrm{E}_{x'}[k(\boldsymbol{x}, \boldsymbol{x}')]\mathrm{E}_{y'}[l(\boldsymbol{y}, \boldsymbol{y}')]] \\ &\quad + \mathrm{E}_{xx'}[k(\boldsymbol{x}, \boldsymbol{x}')]\mathrm{E}_{yy'}[l(\boldsymbol{y}, \boldsymbol{y}')] \end{aligned} \tag{4}$$

where $\mathrm{E}_{xx'yy'}$ is the expectation over both $(x, y) \sim \mathrm{P}_{xy}$ and $(x', y') \sim \mathrm{P}_{xy}$ drawn independently according to the same law, and $k(\boldsymbol{x}, \boldsymbol{x}') = \langle\phi(\boldsymbol{x}), \phi(\boldsymbol{x}')\rangle$ and $l(\boldsymbol{y}, \boldsymbol{y}') = \langle\varphi(\boldsymbol{y}), \varphi(\boldsymbol{y}')\rangle$ are kernel functions. This expression follows that when kernels k and l are bounded, the Hilbert-Schmidt norm of C_{xy} exists. It is easy to find that if both feature maps are linear (i.e., $\phi(\boldsymbol{x}) = \boldsymbol{x}$ and $\varphi(\boldsymbol{y}) = \boldsymbol{y}$), HSIC is the same as the square of the Frobenius norm of the cross-covariance matrix.

2.2. **Empirical HSIC.** Given a set of observations $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ that are drawn from the joint probability distribution P_{xy} and the chosen kernels k and l, we can construct two kernel matrices $\mathbf{K}, \mathbf{L} \in \mathrm{R}^{n \times n}$, where $\mathbf{K}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $\mathbf{L}_{ij} = l(\boldsymbol{y}_i, \boldsymbol{y}_j)$. The commonly used empirical HSIC [13] is given by

$$\mathit{HSIC}_0(\mathrm{F}, \mathrm{G}, D) = \frac{1}{(n-1)^2}\mathrm{tr}(\mathbf{KHLH}) \tag{5}$$

where $\mathbf{H} = \mathbf{I}_n - \boldsymbol{e}_n\boldsymbol{e}_n^{\mathrm{T}}/n \in \mathrm{R}^{n \times n}$ is a centering matrix, where $\mathbf{I}_n \in \mathrm{R}^{n \times n}$ and $\boldsymbol{e}_n \in \mathrm{R}^n$ are the identity matrix and vector of ones, respectively. However, as shown in [3], this estimator has bias $O(n^{-1})$, i.e.,

$$HSIC(\mathrm{F},\mathrm{G},\mathrm{P}_{xy}) - \mathrm{E}_D[HSIC_0(\mathrm{F},\mathrm{G},D)] = O\big(n^{-1}\big) \tag{6}$$

To address this limitation, an unbiased estimator was proposed in [3], which has the form:

$$HSIC_1(\mathrm{F},\mathrm{G},D) = \frac{1}{n(n-3)}\left[\mathrm{tr}\left(\tilde{\mathbf{K}}\tilde{\mathbf{L}}\right) + \frac{\boldsymbol{e}_n^{\mathrm{T}}\tilde{\mathbf{K}}\boldsymbol{e}_n\boldsymbol{e}_n^{\mathrm{T}}\tilde{\mathbf{L}}\boldsymbol{e}_n}{(n-1)(n-2)} - \frac{2}{n-2}\boldsymbol{e}_n^{\mathrm{T}}\tilde{\mathbf{K}}\tilde{\mathbf{L}}\boldsymbol{e}_n\right] \tag{7}$$

where $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{L}}$ are the matrices obtained by setting the diagonal entries of $\mathbf{K}$ and $\mathbf{L}$ to zero, i.e., $\tilde{\mathbf{K}}_{i,j} = (1-\delta_{i,j})\mathbf{K}_{i,j}$ and $\tilde{\mathbf{L}}_{i,j} = (1-\delta_{i,j})\mathbf{L}_{i,j}$, where $\delta_{i,j}$ is the Kronecker delta.

2.3. **Characteristics of HSIC.** There are several advantages to use HSIC as a dependence criterion. *Firstly*, it has been shown that whenever F and G are RKHSs with universal or characteristic kernels [27] such as Gaussian kernel and Laplace kernel, then $HSIC(\mathrm{F},\mathrm{G},\mathrm{P}_{xy}) = 0$ if and only if x and y are independent [13]. This property allows us to use the HSIC as a dependence criterion to detect any nonlinear dependence. Note that non-universal or non-characteristic kernels can be also employed in the HSIC, although they may not guarantee that all dependence can be detected [3]. *Secondly*, the empirical HSIC is stable with respect to different splits of the data since it is sharply concentrated around its expected value (the empirical HSIC asymptotically converges to the true HSIC with rate $O\left(1/\sqrt{n}\right)$ [13]). That is, for random draws of observation from P_{xy}, HSIC provides values which are very similar. This is desirable since we want our learning algorithms to be robust to small changes. *Thirdly*, though the unbiased estimator of HSIC has relatively more complex form, both the biased and unbiased estimators are easy to compute, with an overall $O\big(n^2\big)$ time complexity [3,13], since only the kernel matrices $\mathbf{K}$ and $\mathbf{L}$ are needed and no density estimation is involved. *Finally*, rich choices of kernels can be directly applied to the input and output. This freedom of choosing kernels allows us to incorporate prior knowledge of the learning tasks at hand into the dependence estimation process.

It is because of these advantages that wide applications of the HSIC have been presented [14]. In the following, we will discuss a new application of the HSIC, i.e., HSIC Lasso with application to unsupervised feature selection.

3. The Proposed HSIC-Lasso-Based Unsupervised Feature Selection Method.

This section contains our unsupervised feature selection method based on the HSIC Lasso (UHSIC-Lasso) and addresses the extensions arising in practical applications.

3.1. **Unsupervised feature selection based on HSIC Lasso.** HSIC Lasso was first introduced by Yamada et al. [20] for feature selection. Suppose $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]^{\mathrm{T}} \in \mathrm{R}^{n\times d}$ represents the input data which has n samples with d features and $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^{\mathrm{T}} \in \mathrm{R}^n$ is the output label vector, and $\mathbf{K}$ and $\mathbf{L}$ are corresponding kernel matrices respectively defined on the input data $\mathbf{X}$ and output label vector $\boldsymbol{y}$, HSIC Lasso is given by

$$\begin{aligned} &\boldsymbol{w} = \underset{\boldsymbol{w}}{\mathrm{argmin}}\, \frac{1}{2}\left\|\bar{\mathbf{L}} - \sum_{i=1}^{d} w_i\bar{\mathbf{K}}_i\right\|_{\mathrm{Fro}}^2 + \lambda\|\boldsymbol{w}\|_1 \\ &\text{s.t. } w_i \geq 0,\ i = 1,\ldots,d \end{aligned} \tag{8}$$

where $\bar{\mathbf{K}} = \mathbf{HKH}$, $\bar{\mathbf{L}} = \mathbf{HLH}$, and $\mathbf{K}_i$ is the kernel matrix on the ith feature for all samples. In (8), the first term indicates that the linear combination of the centered kernel matrices $\left\{\bar{\mathbf{K}}_i\right\}_{i=1}^{d}$ (one feature associates with a kernel) for the input data is aligned with the centered kernel matrix $\bar{\mathbf{L}}$ for the output label, and the second term indicates that the

weights (combination coefficients) for the irrelevant features (kernels) tend to be zero since the l_1 norm inclines to generate a sparse solution. HSIC Lasso has been demonstrated with many benchmark examples and real-word applications [20,28-33].

However, the original HSIC Lasso was presented for supervised feature selection. For the task of unsupervised feature selection, the dependence between the subsets of features and the full input data that expresses the underlying cluster structure rather than that between the subsets of features and class labels in supervised settings should be considered, which was well demonstrated by the unsupervised BAHSIC [22,23]. Inspired by the ideas of HSIC Lasso and unsupervised BAHSIC, we present the following alternative representation of $\boldsymbol{w}$:

$$\begin{aligned} \boldsymbol{w} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \frac{1}{2} \left\| \bar{\mathbf{K}} - \sum_{i=1}^{d} w_i \bar{\mathbf{K}}_i \right\|_{\mathrm{Fro}}^2 + \lambda \|\boldsymbol{w}\|_1 \\ \text{s.t. } w_i \geq 0,\ i = 1, \ldots, d \end{aligned} \tag{9}$$

Compared with (8), the only difference is that the centered kernel matrix $\bar{\mathbf{L}}$ defined on the class labels is replaced by the centered kernel matrix $\bar{\mathbf{K}}$ defined on the full input data. After estimating $\boldsymbol{w}$, we normalize each element of $\boldsymbol{w}$ as $w_i \leftarrow w_i \Big/ \sum_{i=1}^{d} w_i$.

Note that $\left\langle \bar{\mathbf{K}}, \bar{\mathbf{K}}_i \right\rangle_{\mathrm{Fro}} = \left\langle \mathbf{K}, \bar{\mathbf{K}}_i \right\rangle_{\mathrm{Fro}} = \left\langle \bar{\mathbf{K}}, \mathbf{K}_i \right\rangle_{\mathrm{Fro}} = \operatorname{tr}(\mathbf{K}_i \mathbf{H} \mathbf{K} \mathbf{H}) = (n-1)^2 HSIC(\mathbf{K}, \mathbf{K}_i)$, where $HSIC(\mathbf{K}, \mathbf{K}_i)$ is an empirical HSIC shown as (5), the first term of (9) can be written as

$$\begin{aligned} & \frac{1}{2} \left\| \bar{\mathbf{K}} - \sum_{i=1}^{d} w_i \bar{\mathbf{K}}_i \right\|_{\mathrm{Fro}}^2 \\ = & \frac{1}{2} \left\langle \bar{\mathbf{K}} - \sum_{i=1}^{d} w_i \bar{\mathbf{K}}_i, \bar{\mathbf{K}} - \sum_{i=1}^{d} w_i \bar{\mathbf{K}}_i \right\rangle_{\mathrm{Fro}} \\ = & \frac{1}{2} \left\langle \bar{\mathbf{K}}, \bar{\mathbf{K}} \right\rangle_{\mathrm{Fro}} - \sum_{i=1}^{d} w_i \left\langle \bar{\mathbf{K}}, \bar{\mathbf{K}}_i \right\rangle_{\mathrm{Fro}} + \frac{1}{2} \sum_{i=1}^{d} \sum_{j=1}^{d} w_i w_j \left\langle \bar{\mathbf{K}}_i, \bar{\mathbf{K}}_j \right\rangle_{\mathrm{Fro}} \\ = & \frac{(n-1)^2}{2} HSIC(\mathbf{K}, \mathbf{K}) - (n-1)^2 \sum_{i=1}^{d} w_i HSIC(\mathbf{K}, \mathbf{K}_i) \\ & + \frac{(n-1)^2}{2} \sum_{i=1}^{d} \sum_{j=1}^{d} w_i w_j HSIC(\mathbf{K}_i, \mathbf{K}_j) \end{aligned} \tag{10}$$

Considering that $HSIC(\mathbf{K}, \mathbf{K})$ is a constant and can be ignored, (9) can be given by

$$\begin{aligned} \boldsymbol{w} = \underset{\boldsymbol{w}}{\operatorname{argmin}} - \sum_{i=1}^{d} w_i HSIC(\mathbf{K}, \mathbf{K}_i) + \sum_{i=1}^{d} \sum_{j=1}^{d} w_i w_j HSIC(\mathbf{K}_i, \mathbf{K}_j) + \lambda \|\boldsymbol{w}\|_1 \\ \text{s.t. } w_i \geq 0,\ i = 1, \ldots, d \end{aligned} \tag{11}$$

It becomes apparent that this unsupervised feature selection is driven by three components. The first term measures the dependence between the considered features and the full set of features. On the one hand, if the ith kernel matrix $\mathbf{K}_i$ has high dependence on the full data matrix $\mathbf{K}$, $HSIC(\mathbf{K}, \mathbf{K}_i)$ takes a large value and thus w_i should also be large. On the other hand, if $\mathbf{K}_i$ and $\mathbf{K}$ are independent, $HSIC(\mathbf{K}, \mathbf{K}_i)$ is close to zero and thus w_i is forced to zero by the l_1-regularizer (the third term). This means that features with high dependence on the full data tend to be selected. The second term measures the

dependence between different features. If $\mathbf{K}_i$ and $\mathbf{K}_j$ are highly dependent, which means one of them is related to a redundant feature, $HSIC(\mathbf{K}_i, \mathbf{K}_j)$ takes a large value and thus either w_i or w_j tends to be zero. This means that redundant features tend to be removed. To summarize, it tends to find non-redundant features with high dependence on the full input data.

The first term in (9) can be rewritten as the usual Lasso loss, i.e.,

$$\frac{1}{2}\left\|\bar{\mathbf{K}} - \sum_{i=1}^{d} w_i \bar{\mathbf{K}}_i\right\|_{\mathrm{Fro}}^2 = \frac{1}{2}\left\|\mathrm{vec}\left(\bar{\mathbf{K}}\right) - \left[\mathrm{vec}\left(\bar{\mathbf{K}}_1\right), \mathrm{vec}\left(\bar{\mathbf{K}}_2\right), \ldots, \mathrm{vec}\left(\bar{\mathbf{K}}_d\right)\right] \boldsymbol{w}\right\|_2^2 \quad (12)$$

where $\mathrm{vec}(\cdot)$ is the vectorization operator. Since both $\bar{\mathbf{K}}$ and $\bar{\mathbf{K}}_i$ are $n \times n$ matrices, $\mathrm{vec}\left(\bar{\mathbf{K}}\right)$ and $\mathrm{vec}\left(\bar{\mathbf{K}}_i\right)$ are $n^2 \times 1$ vectors. This is the same form as regular Lasso with n^2 samples and d features. After the vectorization, the optimization problem (9) can be solved by a regular Lasso solver.

3.2. **Discussion and extension.** HSIC Lasso can be regarded as a convex variant of the widely used minimum redundancy maximum relevance (mRMR) feature selection algorithm [10]. In recent years, relevance redundancy trade-off criteria such as mRMR have become very promising and popular for feature selection [10,34,35]. However, traditional algorithmic frameworks which are based on the mutual information criterion have certain limitations. For example, the optimization problem of mutual information-based mRMR approaches is discrete and estimating mutual information from finite samples is not easy. Furthermore, it is unclear whether the mutual information-based mRMR approaches have neat theoretical properties such as guaranteeing the optimal solutions. Compared with the mutual information-based mRMR algorithms, HSIC Lasso employs the HSIC instead of mutual information as the dependence criterion, in which estimating HSIC is much easier. Moreover, HSIC Lasso is a convex optimization problem, for which the globally optimal solution can therefore be found.

More interestingly, the framework (8) is very general: by designing appropriate kernels (kernel matrices) for the output data, various feature selection models can be achieved. In theory, any universal kernel such as Gaussian kernel and Laplace kernel can be employed in HSIC to detect any nonlinear dependence between two random variables [13]. Non-universal kernels can be also employed in the HSIC, although they may not guarantee that all dependence can be detected [3]. For the input features, kernels can be either the universal or non-universal kernels. Generally speaking, kernels on the output data can be as general as those defined on the input data. However, prior knowledge of the learning tasks should be more considered to define such kernels. For example, for the classification problems ($y_i \in \{+1, -1\}$, $i = 1, 2, \ldots, n$), the kernel matrix can be defined as

$$\mathbf{L}_{ij} = \begin{cases} +1 & y_i = y_j \\ -1 & y_i \neq y_j \end{cases} \quad (13)$$

This definition reveals the ideal pairwise similarities between samples, i.e., the similarities from the same class are set to $+1$ while those from different classes are -1. For the regression problems ($y_i \in \mathrm{R}$, $i = 1, 2, \ldots, n$), the kernel matrix can be given by

$$\mathbf{L}_{ij} = \exp\left(-\frac{(y_i - y_j)^2}{2\sigma^2}\right) \quad (14)$$

where σ is the Gaussian kernel width. Both classification and regression are supervised learning problems. For the unsupervised learning problems, we can define the kernel matrix as

$$\mathbf{L}_{ij} = \mathbf{K}_{ij} = \exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)^2}{2\sigma^2}\right) \tag{15}$$

To summarize, we can obtain a family of feature selection methods by defining appropriate kernels for the output data. This unified view of feature selection is a useful abstract tool since it not only bridges different kinds of feature selection methods under a common umberlla, but also guides the design of new feature selection methods. For example, for the semisupervised feature selection, the only difficulty is how to define an appropriate kernel that can fully exploit the label information of labeled data and data distribution or local structure of both labeled and unlabeled data [36].

4. **Experiments.** In this section, we investigate experimentally the performance of the proposed unsupervised feature selection method on several real-world data sets.

4.1. **Experimental setup.** We selected six popular data sets, i.e., *Glass Identification*, *Statlog (Vehicle Silhouettes)*, *Wine*, *Trains*, *DBWorld e-mails*, and *MicroMass* from the UCI machine learning repository [37]. Table 1 provides the statistics of these data sets. The name, number of samples, number of features, and number of classes of each data set are presented. For each data set, since the HSIC changes with respect to kernel types or kernel parameters, i.e., the kernel type and kernel parameters should be fixed for all features, we first normalized the features to have unit standard deviation and then used the Gaussian kernel with the kernel width $\sigma = 1$[1].

Table 1. Statistics of the selected six data sets

ID	Data sets	Number of samples	Number of features	Number of classes
1	Glass Identification	214	9	6
2	Wine	178	13	3
3	Statlog (Vehicle Silhouettes)	946	18	4
4	Trains	10	32	2
5	MicroMass	931	1300	10
6	DBWorld e-mails	64	4702	2

We compared our proposed UHSIC-Lasso with the following state-of-the-art HSIC-based unsupervised feature selection algorithms, i.e., UBHSIC, SP-BAHSIC, SPC-BAHSIC, SPC-FOHSIC, SP-LRHSIC, and SPC-LRHSIC. All these methods employ heuristic search strategies with backward elimination or forward selection. For more detailed information, please refer to [22,23]. To solve the Lasso problem (9), a technique called dual augmented Lagrangian (DAL) was shown to be computationally highly efficient [38,39]. Because DAL can incorporate the non-negativity constraint without losing its computational advantages, we directly used the DAL to solve our HSIC Lasso problem[2]. Furthermore, we used SVM with the Gaussian kernel for evaluating the classification accuracy when the top five features selected by each method were used. The SVM classifier was implemented by the software LIBSVM [40], and the regularization parameter C and kernel parameter σ were determined by the 5-fold cross-validation with a grid search over two dimensions, i.e., $C = \{2^{-5}, 2^{-3}, \ldots, 2^{15}\}$ and $\gamma = \{2^{-15}, 2^{-14}, \ldots, 2^{3}\}$, where $\gamma = 1/\sigma^2$.

[1] In [20], the authors have demonstrated that the HSIC Lasso is not so sensitive to the width parameter of the Gaussian kernel.

[2] A MATLAB implementation of HSIC Lasso is available at http://www.kecl.ntt.co.jp/icl/ls/members/myamada/hsiclasso.html.

4.2. **Results and discussion.** To obtain stable results, we used the 5-fold cross-validation technique to assess the classification accuracy. The average classification accuracy and standard deviation of each method are reported in Table 2. The bold numbers denote the best performance of these methods on each data set. To conduct a rigorous comparison, the paired t-test [41] is performed. The paired t-test is used to analyze whether the difference between two compared methods on one data set is significant. The p-value of the paired t-test represents the probability that two sets of compared results come from distributions with an equal mean. A p-value of 0.05 is considered to be statistically significant. The win-tie-loss (W-T-L) summarizations based on the paired t-test are listed in Table 3, where the proposed UHSIC-Lasso and other methods are respectively compared. In comparing two methods such as method 1 versus method 2, a win or loss means that method 1 is better or worse than method 2 on a data set. A tie means that both methods achieve the same performance.

Table 2. Classification accuracy (%) comparison of different feature selection methods

ID	UBHSIC	SP-BAHSIC	SPC-BAHSIC	SPC-FOHSIC	SP-LRHSIC	SPC-LRHSIC	UHSIC-Lasso
1	91.3±1.8	89.2±4.1	86.2±6.2	85.2±6.2	89.8±5.4	84.8±5.5	**91.4±2.3**
2	89.1±4.2	91.8±4.1	90.0±3.4	91.1±3.4	**93.1±3.7**	92.3±2.6	93.0±4.0
3	63.7±5.2	63.7±3.5	70.2±3.1	57.3±2.5	67.2±5.8	61.3±4.7	**71.6±3.8**
4	64.0±2.9	63.0±4.8	75.0±5.5	54.2±6.1	61.4±2.4	62.5±2.1	**77.2±4.4**
5	30.7±5.7	53.6±4.1	43.1±4.7	36.8±5.7	58.6±4.6	56.3±4.2	**62.7±5.1**
6	45.0±11.3	52.1±7.1	67.2±3.5	40.3±5.8	73.7±3.1	79.1±4.8	**83.1±3.2**

Table 3. Significance test of classification results

ID	UHSIC-Lasso vs. UBHSIC	UHSIC-Lasso vs. SP-BAHSIC	UHSIC-Lasso vs. SPC-BAHSIC	UHSIC-Lasso vs. SPC-FOHSIC	UHSIC-Lasso vs. SP-LRHSIC	UHSIC-Lasso vs. SPC-LRHSIC
1	T	W	W	W	W	W
2	W	W	W	W	T	W
3	W	W	W	W	W	W
4	W	W	W	W	W	W
5	W	W	W	W	W	W
6	W	W	W	W	W	W

From Table 2, we find that, the proposed UHSIC-Lasso consistently achieves the overall best classification performance to other baseline approaches. Of the six data sets evaluated, SP-LRHSIC reports one best result, while our UHSIC-Lasso reports five best results. Furthermore, from Table 3, although SP-LRHSIC reports the best result for the *Wine* data set, the performance difference between SP-LRHSIC and our UHSIC-Lasso is not statistically significant. We attribute this superior performance to the fact that the proposed UHSIC-Lasso has the advantage of selecting features relevant to the full input data, and in the meanwhile independent of each other. Different from our method, the baseline approaches only address the dependence between the selected features and the full input data, while the redundancy between different features is neglected. Besides, all the baseline methods employ heuristic search strategies which tend to produce local optima, while our method can find out the globally optimal solution.

5. **Conclusion and Further Study.** This paper presents an HSIC-Lasso-based unsupervised feature selection method, called UHSIC-Lasso. This method takes account of both feature relevance to the target variable and dependencies among the features, in order to select a subset of relevant but non-redundant features. In discussing the connection between unsupervised feature selection and HSIC Lasso, we find that the proposed method not only has a clear statistical interpretation that minimum redundant features with maximum dependence on the full input data can be found in terms of the HSIC, but also enables the global optimal solution to be computed efficiently by solving a Lasso optimization problem. Furthermore, a unified feature selection famework applicable to multiple learning scenarios based on the HSIC Lasso is discussed, which is a useful abstract tool since different kinds of feature selection methods can be viewed and investigated in a unified way.

As of future work, the usefulness of the proposed method will be further investigated on more real-world applications such as computer vision, bioinformatics, and speech and signal processing. Moreover, extending the proposed method to other learning models such as multiple kernel learning, multi-task learning, and multi-label learning and investigating theoretical properties of the proposed formulation are important issues to be investigated. Last but not least, it should be noted that, in this paper, only the biased estimator of the HSIC was used and demonstrated. Using the unbiased estimator of the HSIC instead of the biased one as the objective function for unsuperised feature selection is worthy of being explored.

Acknowledgment. This work is partially supported by the National Natural Science Foundation of China (No. 61966002) and the Science and Technology Program Foundation of Jiangxi Education Committee of China (No. GJJ201407). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] I. Guyon and A. Elisseef, An introduction to variable and feature selection, *Journal of Machine Learning Research*, vol.3, pp.1157-1182, 2003.

[2] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang and H. Liu, Feature selection: A data perspective, *ACM Computing Surveys*, vol.50, no.6, Article No.94, 2018.

[3] L. Song, A. Smola, A. Gretton, J. Bedo and K. Borgwardt, Feature selection via dependence maximization, *Journal of Machine Learning Research*, vol.13, pp.1393-1434, 2012.

[4] S. Liao and Y. Lin, Variable costs-based multi-granularity feature selection with test cost constraint, *International Journal of Innovative Computing, Information and Control*, vol.16, no.6, pp.2047-2061, 2020.

[5] K. Yu, L. Liu and J. Li, A unified view of causal and non-causal feature selection, *ACM Trans. Knowledge Discovery from Data*, vol.15, no.4, Article No.63, 2021.

[6] Q. Gu, Z. Li and J. Han, Generalized Fisher score for feature selection, *Proc. of the 27th Conference on Uncertainty in Artificial Intelligence*, Barcelona, Spain, pp.266-273, 2011.

[7] M. Dash and H. Liu, Consistency-based search in feature selection, *Artificial Intelligence*, vol.15, nos.1-2, pp.155-176, 2003.

[8] A. A. Ding, J. G. Dy, Y. Li and Y. Chang, A robust-equitable measure for feature ranking and selection, *Journal of Machine Learning Research*, vol.18, pp.1-46, 2017.

[9] R. Battiti, Using mutual information for selecting features in supervised neural net learning, *IEEE Trans. Neural Networks*, vol.5, no.4, pp.537-550, 1994.

[10] H. Peng, F. Long and C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.27, no.8, pp.1226-1238, 2005.

[11] J. R. Vergara and P. A. Estévez, A review of feature selection methods based on mutual information, *Neural Computing and Applications*, vol.24, pp.175-186, 2014.

[12] H. Peng and Y. Fan, Feature selection by optimizing a lower bound of conditional mutual information, *Information Sciences*, vols.418-419, pp.652-667, 2017.
[13] A. Gretton, O. Bousquet, A. Smola and B. Schölkopf, Measuring statistical dependence with Hilbert-Schmidt norms, *Proc. of the 16th International Conference on Algorithmic Learning Theory*, Singapore, pp.63-77, 2005.
[14] T. Wang, X. Dai and Y. Liu, Learning with Hilbert-Schmidt independence criterion: A review and new perspectives, *Knowledge-Based Systems*, vol.234, 107567, 2021.
[15] L. Song, A. Smola, A. Gretton, K. Borgwardt and J. Bedo, Supervised feature selection via dependence estimation, *Proc. of the 24th International Conference on Machine Learning*, Corvallis, USA, pp.823-830, 2007.
[16] L. Song, J. Bedo, K. Borgwardt, A. Gretton and A. Smola, Gene selection via the BAHSIC family of algorithms, *Bioinformatics*, vol.23, pp.i490-i498, 2007.
[17] C. Liu, Q. Ma and J. Xu, Multi-label feature selection method combining unbiased Hilbert-Schmidt independence criterion with controlled genetic algorithm, in *Neural Information Processing. ICONIP 2018. Lecture Notes in Computer Science*, L. Cheng, A. Leung and S. Ozawa (eds.), Cham, Springer, 2018.
[18] S. Geeitha and M. Thangamani, Incorporating EBO-HSIC with SVM for gene selection associated with cervical cancer classification, *Journal of Medical Systems*, vol.42, no.11, Article No.225, 2018.
[19] M. Masaeli, G. Fung and J. G. Dy, From transformation-based dimensionality reduction to feature selection, *Proc. of the 27th International Conference on Machine Learning*, Haifa, Israel, pp.751-758, 2010.
[20] M. Yamada, W. Jitkrittum, L. Sigal, E. P. Xing and M. Sugiyama, High-dimensional feature selection by feature-wise kernelized Lasso, *Neural Computation*, vol.26, no.1, pp.185-207, 2014.
[21] 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. Computational Biology and Bioinformatics*, vol.14, no.1, pp.167-181, 2017.
[22] J. Bedo, Microarray design using the Hilbert-Schmidt independence criterion, *Proc. of the 3rd IAPR International Conference on Pattern Recognition in Bioinformatics*, Melbourne, Australia, pp.288-298, 2008.
[23] S. Liaghat and E. G. Mansoori, Filter-based unsupervised feature selection using Hilbert-Schmidt independence criterion, *International Journal of Machine Learning and Cybernetics*, vol.10, no.9, pp.2313-2328, 2019.
[24] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda and B. Schölkopf, An introduction to kernel-based learning algorithms, *IEEE Trans. Neural Networks*, vol.38, no.2, pp.181-202, 2001.
[25] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, Cambridge, U.K., 2004.
[26] T. Wang, L. Zhang and W. Hu, Bridging deep and multiple kernel learning: A review, *Information Fusion*, vol.67, pp.3-13, 2021.
[27] I. Steinwart, On the influence of the kernels on the consistency of support vector machines, *Journal of Machine Learning Research*, vol.2, pp.67-93, 2001.
[28] M. Yamada, A. Kimura, F. Naya and H. Sawada, Change-point detection with feature selection in high-dimensional time-series data, *Proc. of the 23rd International Joint Conference on Artificial Intelligence*, Beijing, China, pp.1827-1833, 2013.
[29] M. Yamada, J. Tang, J. Lugo-Martinez, E. Hodzic, R. Shrestha, A. Saha, H. Ouyang, D. Yin, H. Mamitsuka, C. Sahinalp, P. Radivojac, F. Menczer and Y. Chang, Ultra high-dimensional nonlinear feature selection for big biological data, *IEEE Trans. Knowledge and Data Engineering*, vol.30, no.7, pp.1352-1365, 2018.
[30] A. Abugabah, A. A. AlZubi, F. Al-Obeidat, A. Alarifi and A. Alwadain, Data mining techniques for analyzing healthcare conditions of urban space-person lung using meta-heuristic optimized neural networks, *Cluster Computing*, vol.23, no.3, pp.1781-1794, 2020.
[31] W. Ren, B. Li and M. Han, A novel Granger causality method based on HSIC-Lasso for revealing nonlinear relationship between multivariate time series, *Physica A: Statistical Mechanics and Its Applications*, vol.541, 123245, 2020.
[32] T. Freidling, B. Poignard, H. Climente-González and M. Yamada, Post-selection inference with HSIC-Lasso, *Proc. of the 38th International Conference on Machine Learning*, Virtual Event, pp.3439-3448, 2021.

[33] K. Koyama, K. Kiritoshi, T. Okawachi and T. Izumitani, Effective nonlinear feature selection method based on HSIC Lasso and with variational inference, *Proc. of the 25th International Conference on Artificial Intelligence and Statistics*, Virtual Event, pp.10407-10421, 2022.
[34] A. Unler, A. Murat and R. B. Chinnam, mr^{2}PSO: A maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification, *Information Sciences*, vol.181, no.20, pp.4625-4641, 2011.
[35] J. Che, Y. Yang, L. Li, X. Bai, S. Zhang and C. Deng, Maximum relevance minimum common redundancy feature selection for nonlinear data, *Information Sciences*, vols.409-410, pp.68-86, 2017.
[36] R. Sheikhpour, M. A. Sarram, S. Gharaghani and M. A. Z. Chahooki, A survey on semi-supervised feature selection methods, *Pattern Recognition*, vol.64, pp.141-158, 2017.
[37] D. Dua and C. Graff, *UCI Machine Learning Repository*, http://archive.ics.uci.edu/ml, 2019.
[38] R. Tomioka and M. Sugiyama, Dual-augmented Lagrangian method for efficient sparse reconstruction, *IEEE Signal Processing Letters*, vol.16, no.2, pp.1067-1070, 2009.
[39] R. Tomioka and M. Sugiyama, Super-linear convergence of dual augmented Lagrangian algorithm for sparsity regularized estimation, *Journal of Machine Learning Research*, vol.12, pp.1537-1586, 2011.
[40] C.-C. Chang and C.-J. Lin, LIBSVM: A library for support vector machines, *ACM Trans. Intelligent Systems and Technology*, vol.2, no.3, Article No.27, 2011.
[41] J. Demšar, Statistical comparisons of classifiers over multiple data sets, *Journal of Machine Learning Research*, vol.7, pp.1-30, 2006.

Author Biography

Tinghua Wang received the M.Sc. degree in Computer Science from Nanchang University, Nanchang, China, 2006; the Ph.D. degree in Computer Science from Beijing Jiaotong University, Beijing, China, 2010. From October 2011 to October 2013, he was a Postdoctoral Research Fellow with the Institute of Computer Science and Technology, Peking University, Beijing, China. From March 2016 to March 2017, he was a Visting Scholar with the Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW, Australia.

Prof. Wang is currently a full-time professor with the School of Mathematics and Computer Science, Gannan Normal University, Ganzhou, China. His research interests include artificial intelligence and machine learning. He has authored and coauthored more than 50 papers in *IEEE Transactions on Fuzzy Systems*, *Information Fusion*, other refereed journals, and conference proceedings.

Zhenwei Hu received the B.Sc. degree in Computer Science and Technology from Gannan Normal University, Ganzhou, China in 2020. He is studying for the M.Sc. degree at the School of Mathematics and Computer Science, Gannan Normal University. His research interests include machine learning and data mining.

Huiying Zhou received the B.Sc. degree in Computer Science and Technology from Gannan Normal University, Ganzhou, China in 2020. She is studying for the M.Sc. degree at the School of Mathematics and Computer Science, Gannan Normal University. Her research interests include machine learning and data mining.