Uurtio, Viivi; Bhadra, Sahely; Rousu, Juho

Sparse Non-linear CCA through Hilbert-Schmidt Independence Criterion

Published in:
2018 IEEE International Conference on Data Mining, ICDM 2018

DOI:
10.1109/ICDM.2018.00172

Published: 01/01/2018

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Uurtio, V., Bhadra, S., & Rousu, J. (2018). Sparse Non-linear CCA through Hilbert-Schmidt Independence Criterion. In *2018 IEEE International Conference on Data Mining, ICDM 2018* (pp. 1278-1283). Article 8594981 IEEE. https://doi.org/10.1109/ICDM.2018.00172

Sparse Non-Linear CCA through Hilbert-Schmidt Independence Criterion

Viivi Uurtio*, Sahely Bhadra†, Juho Rousu*
*Helsinki Institute for Information Technology HIIT
Department of Computer Science, Aalto University, Espoo, Finland
firstname.lastname@aalto.fi
†Computer Science and Engineering
Indian Institute of Technology (IIT)
Palakkad, India
sahely@iitpkd.ac.in

***Abstract*—We present SCCA-HSIC, a method for finding sparse non-linear multivariate relations in high-dimensional settings by maximizing the Hilbert-Schmidt Independence Criterion (HSIC). We propose efficient optimization algorithms using a projected stochastic gradient and Nyström approximation of HSIC. We demonstrate the favourable performance of SCCA-HSIC over competing methods in detecting multivariate non-linear relations both in simulation studies, with varying numbers of related variables, noise variables, and samples, as well as in real datasets.**

***Keywords*-dimensionality reduction, canonical correlation, sparsity, kernel methods, Hilbert-Schmidt Independence Criterion**

I. Introduction

Canonical correlation methods [1] are applied to uncover multivariate relations between variables in the columns of data matrices, or views, $\mathbf{X} \in \mathbb{R}^{n \times p}$ and $\mathbf{Y} \in \mathbb{R}^{n \times q}$ where n denotes the sample size and p and q the numbers of variables respectively. In general, these methods identify the related variables and the form of the multivariate relation [2].

The classical CCA methods find canonical coefficient vectors, $\mathbf{u} \in \mathbb{R}^p$ and $\mathbf{v} \in \mathbb{R}^q$, that maximize the linear correlation, $\rho = \langle \mathbf{Xu}, \mathbf{Yv} \rangle / ||\mathbf{Xu}||_2 ||\mathbf{Yv}||_2$, between the score vectors, $\mathbf{Xu}$ and $\mathbf{Yv}$. These methods are best applicable to relatively low-dimensional data that contains linear relations. In high-dimensional settings, different regularized variants of CCA are used, based on shrinking the associated covariance matrices [3], [4] or using sparsity inducing norms [5]–[7].

Non-linear multivariate relations can be analyzed by kernel CCA (KCCA) [8] and deep CCA [9]. In KCCA, the observations are implicitly mapped through kernel functions to potentially high-dimensional Hilbert spaces where the canonical correlation is then evaluated. Non-linear relations can be analyzed using non-linear kernels. However, the underlying projections are not explicit which complicates the interpretation of the related variables. In deep CCA (DCCA), every observation is non-linearly transformed multiple times through a layered neural network in both views, which are trained to maximize the canonical correlation of the network outputs. Although the method finds non-linear relations the related variables are difficult to identify due to the neural network.

In [10], Hilbert-Schmidt Independence Criterion (HSIC) was introduced as an alternative non-linear measure of (in)dependence, which has since been used for different machine learning tasks such as independent component analysis [11], feature selection [12] and, related to this paper, non-linear CCA [13]. The CCA-HSIC method [13] uses HSIC as a non-linear correlation measure, in place of the usual linear correlation. CCA-HSIC was shown to extract linear and non-linear multivariate relations with interpretable related variables. However, it relies on ℓ_2 norm regularization, and hence lacks sparsity which makes interpretation difficult in high-dimensional settings. Recently in [14], a sparse non-linear CCA approach relying on sparse feature-wise multiple kernel learning (MKL) framework, named two-stage kernel CCA (TSKCCA) was introduced. The approach finds non-linear relations through the kernels, however, the feature-wise MKL approach limits the capacity of the method in finding non-linear multivariate relations. Additionally, the computation of the feature-wise kernels makes TSKCCA intractable for large-scale studies.

This paper presents SCCA-HSIC that finds sparse non-linear multivariate relations from two-view data, and identifies the underlying related variables. Additionally, we propose a Nyström approximated version of SCCA-HSIC for large-scale studies. SCCA-HSIC is evaluated and compared in simulation studies together with TSKCCA [14], CCA-HSIC [13], KCCA [8], SCCA [5], and DCCA [9], as well as in real datasets.

II. Sparse CCA through HSIC Optimization

In this section, we put forward SCCA-HSIC, a novel method to find sparse non-linear relations between two sets of variables. We denote by $\mathbf{I}$ the identity matrix, by $\mathbf{1}$ the

vector of all ones, by $\|.\|_p$ the l_p norm, and by $\langle , \rangle$ the inner product.

A. Hilbert-Schmidt Independence Criterion

Given two separable reproducing kernel Hilbert spaces $\mathcal{H}_{\mathbf{x}}$ and $\mathcal{H}_{\mathbf{y}}$, the Hilbert-Schmidt Independence Criterion (HSIC) [15] between two sets of variables $\phi(\mathbf{x}) \in \mathcal{H}_{\mathbf{x}}$ and $\phi(\mathbf{y}) \in \mathcal{H}_{\mathbf{y}}$ is defined as the squared Hilbert-Schmidt norm of the associated cross-covariance operator $\text{cov}(\phi(\mathbf{x}), \phi(\mathbf{y}))$. When the kernel matrices $\mathbf{K}^x \in \mathbb{R}^{n \times n}$ and $\mathbf{K}^y \in \mathbb{R}^{n \times n}$ are defined in Hilbert spaces $\mathcal{H}_{\mathbf{x}}$ and $\mathcal{H}_{\mathbf{y}}$, the empirical HSIC measure, in terms of centered kernel matrices, $\hat{\mathbf{K}}^{\mathbf{x}}$ and $\hat{\mathbf{K}}^{\mathbf{y}}$, is expressed as follows:

$$\rho(\mathbf{x}, \mathbf{y}) \quad = \quad \text{HSIC}(\mathbf{K}^{\mathbf{x}}, \mathbf{K}^{\mathbf{y}}) = \frac{\text{trace}(\hat{\mathbf{K}}^{\mathbf{x}} \hat{\mathbf{K}}^{\mathbf{y}})}{(n-1)^2}. \quad (1)$$

The centered kernels are given by $\hat{\mathbf{K}}^{\mathbf{x}} = \mathbf{H}\mathbf{K}^{\mathbf{x}}\mathbf{H}$ and $\hat{\mathbf{K}}^{\mathbf{y}} = \mathbf{H}\mathbf{K}^{\mathbf{y}}\mathbf{H}$, where $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the centering operator. Intuitively, HSIC is maximized when the pairwise similarities encoded by the entries of the two kernel matrices $\mathbf{K}^{\mathbf{x}}$ and $\mathbf{K}^{\mathbf{y}}$ match.

B. SCCA-HSIC Model

We assume that sparse subsets of variables in one view, captured by $\mathbf{u} \in \mathbb{R}^p$, are potentially non-linearly related with sparse sets of variables in the other view, captured by $\mathbf{v} \in \mathbb{R}^q$. Hence we find sparse $\mathbf{u}$ and $\mathbf{v}$ by maximizing the non-linear relation in terms of HSIC of two projected variables $\mathbf{X}\mathbf{u}$ and $\mathbf{Y}\mathbf{v}$ while considering ℓ_1 norm regularization on $\mathbf{u}$ and $\mathbf{v}$. Constraining the variables by their ℓ_1 norm induces sparsity in the estimate. Hence to find a sparse set of variables from $\mathbb{R}^p$ and $\mathbb{R}^q$ we solve following optimization problem

$$\begin{aligned} \max_{\mathbf{u},\mathbf{v}} \rho(\mathbf{u}, \mathbf{v}) \quad &= \quad \frac{\text{trace}(\hat{\mathbf{K}}^{\mathbf{u}} \hat{\mathbf{K}}^{\mathbf{v}})}{(n-1)^2} \\ s.t. \qquad & \|\mathbf{u}\|_1 \leq s_x \text{ and } \|\mathbf{v}\|_1 \leq s_y \end{aligned} \quad (2)$$

where $\mathbf{K}^{\mathbf{u}}$ and $\mathbf{K}^{\mathbf{v}}$ are the Gram matrices for projected data $\mathbf{u}^T\mathbf{x}_i$ and $\mathbf{v}^T\mathbf{y}_i$, i.e., $\mathbf{K}^{\mathbf{u}}_{ij} = \langle \mathbf{u}^T\mathbf{x}_i, \mathbf{u}^T\mathbf{x}_i \rangle$ and $\mathbf{K}^{\mathbf{v}}_{ij} = \langle \mathbf{v}^T\mathbf{y}_i, \mathbf{v}^T\mathbf{y}_i \rangle$. The degree of sparsity in $\mathbf{u}$ and $\mathbf{v}$ is controlled by user-defined constants s_x and s_y. A smaller value of s_x and s_y implies greater sparsity. Henceforth, we denote this proposed model as SCCA-HSIC.

In SCCA-HSIC, ℓ_1 regularization can be applied either on both or on only one of the views. Using ℓ_2 norm regularization constraint on both $\mathbf{u}$ and $\mathbf{v}$ in (2) results in CCA-HSIC [13].

Deflation [16] is applied to obtain multiple multivariate relations. As an example for $\mathbf{u}$, upon finding the m^{th} estimate for $\mathbf{u}^{(m)}$ from the current data matrix $\mathbf{X}^{(m)}$, the following orthogonal projection vector $\mathbf{u}^{(m+1)}$ can be obtained by solving $\mathbf{X}^{(m+1)} = \mathbf{X}^{(m)} - \frac{\mathbf{u}^{(m)}\mathbf{u}^{(m)^T}\mathbf{X}^{(m)^T}}{\mathbf{u}^{(m)^T}\mathbf{u}^{(m)}}$. The same procedure is applied on $\mathbf{v}$. In this way, the resulting sets of vectors $\{\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, \ldots\}$ and $\{\mathbf{v}^{(1)}, \mathbf{v}^{(2)}, \ldots\}$ are mutually orthogonal.

Input: $\mathbf{X}$, $\mathbf{Y}$, M (components), R (repetitions),
δ (convergence limit), p_x and p_y (norms of $\mathbf{u}$ and $\mathbf{v}$)
s_x and s_y (l_1 or l_2 norm constraints for $\mathbf{u}$ and $\mathbf{v}$),
σ_u and σ_v (standard deviations of the Gaussian kernels)
Nyström approximation: *π (proportion of columns)*
Output: $\mathbf{U}$, $\mathbf{V}$
for all $m = \{1, 2, \ldots, M\}$ **do**
  for all $r = \{1, 2, \ldots, R\}$ **do**
    Initialize $\mathbf{u}_{mr}$ and $\mathbf{v}_{mr}$
    (Sample π columns uniformly without replacement)
    *Compute $\mathbf{K}^{\mathbf{u}}$, $\mathbf{K}^{\mathbf{v}}$, $\hat{\mathbf{K}}^{\mathbf{u}}$, and $\hat{\mathbf{K}}^{\mathbf{v}}$
    repeat
      *Compute $f_{\text{old}} = \rho(\mathbf{u}, \mathbf{v})$
      Compute $\nabla_{\mathbf{u}} = \frac{\partial \rho(\mathbf{u},\mathbf{v})}{\partial \mathbf{u}}$
      Update $\mathbf{u}_{mr} = \prod_{\|.\|_{p_x} \leq s_x}(\mathbf{u}_{mr} + \gamma \nabla_{\mathbf{u}})$
      (The step size γ determined by line search)
      *Compute $\mathbf{K}^{\mathbf{u}}$ and $\hat{\mathbf{K}}^{\mathbf{u}}$
      Compute $\nabla_{\mathbf{v}} = \frac{\partial \rho(\mathbf{u},\mathbf{v})}{\partial \mathbf{v}}$
      Update $\mathbf{v}_{mr} = \prod_{\|.\|_{p_y} \leq s_y}(\mathbf{v}_{mr} + \gamma \nabla_{\mathbf{v}})$
      (The step size γ determined by line search)
      *Compute $\mathbf{K}^{\mathbf{v}}$ and $\hat{\mathbf{K}}^{\mathbf{v}}$
      *Compute $f_{\text{current}} = \rho(\mathbf{u}, \mathbf{v})$
    until $|f_{\text{old}} - f_{\text{current}}|/|f_{\text{old}} + f_{\text{current}}| < \delta$
    $f_r = f_{\text{current}}$, $\mathbf{u}_r = \mathbf{u}_{mr}$, $\mathbf{v}_r = \mathbf{v}_{mr}$
  end for
  Select $r^* = \arg\max_r f_r$
  Store $\mathbf{U}(:, m) = \mathbf{u}_{r^*}$, $\mathbf{V}(:, m) = \mathbf{v}_{r^*}$
  Deflate $\mathbf{X}^{(m)}$, $\mathbf{Y}^{(m)}$ by $\mathbf{U}(:, m)$ and $\mathbf{V}(:, m)$
end for
Return: $\mathbf{U}$, $\mathbf{V}$

Figure 1. SCCA-HSIC. The Nyström approximated version is given in italics. The approximations, shown by asterisks, of $\mathbf{K}^{\mathbf{u}}$, $\hat{\mathbf{K}}^{\mathbf{u}}$,$\mathbf{K}^{\mathbf{v}}$, and $\hat{\mathbf{K}}^{\mathbf{v}}$ are $\tilde{\Phi}_X$, $\tilde{\Phi}_X$, $\tilde{\Phi}_Y$ and $\tilde{\Phi}_Y$ respectively. The approximated objective is (3).

C. Algorithms

We propose a projected stochastic gradient ascent algorithm (Figure 1) for solving (2), as well as a large-scale variant relying on Nyström approximation of the kernel matrices. As the kernels, any universal kernel can be used, including the Gaussian kernel [15]. Different kernels can be used for the two views.

To maximize the objective function in (2) along the gradient, we apply a stochastic mini-batch gradient. Let $\nabla_{\mathbf{u}}$ and $\nabla_{\mathbf{v}}$ denote the gradients of the kernel functions with respect to $\mathbf{u}$ and $\mathbf{v}$. In every step, we select a mini-batch of random combinations of (i, j) and calculate the gradient with respect to only the selected set of data points. Let $\{\mathcal{I}, \mathcal{J}\}$ denote the random index pair for an iteration. The gradient with respect to $\mathbf{u}$ using the Gaussian kernel function, $K^{\mathbf{u}}_{ij} = \exp^{-\sigma_{\mathbf{x}}(\mathbf{u}^T\mathbf{x}_i - \mathbf{u}^T\mathbf{x}_j)^2}$, is calculated as: $\nabla_{\mathbf{u}} = -2\sigma_x \sum_{\{i,j\} \in \{\mathcal{I},\mathcal{J}\}} \mathbf{K}^{\mathbf{u}}_{ij} \hat{\mathbf{K}}^{\mathbf{v}}_{ij} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{u}$. The gradient can be computed similarly for $\mathbf{v}$. Regularization is achieved by a projection $\prod_{\|.\|_p \leq s}(\mathbf{u})$ onto l_p norm ball. For $p = 2$, $\prod_{\|.\|_p \leq s}(\mathbf{u}) = s\mathbf{u}/\|\mathbf{u}\|_2$. For $p = 1$ we follow the method of ℓ_1 ball projection described in [17].

The calculation of all the entries of the kernel matrices $\mathbf{K}^{\mathbf{v}}$ and $\mathbf{K}^{\mathbf{u}}$ is $\mathcal{O}(n^2 d), d \in \{p, q\}$ which makes the method in-

tractable for large datasets. Very recently, an approximation of the HSIC criterion [18] relying on the Nyström method [19] $\hat{\mathbf{K}}_{nys} = \mathbf{C}\mathbf{W}^{-\frac{1}{2}}(\mathbf{C}\mathbf{W}^{-\frac{1}{2}})^T = \tilde{\Phi}\tilde{\Phi}^T$ was presented, where the matrix $\mathbf{C} \in \mathbb{R}^{n \times n_{nys}}$ consists of n_{nys} randomly chosen columns of kernel $\mathbf{K}$ and $\mathbf{W} \in \mathbb{R}^{n_{nys} \times n_{nys}}$ is the kernel submatrix induced by the chosen columns and rows. Fast computation is achieved by approximating $\hat{\mathbf{K}}_{nys}$ with the uncentered covariance matrix $\tilde{\mathbf{\Sigma}}$ of size $n_{nys} \times n_{nys}$: $\tilde{\mathbf{\Sigma}} = (\mathbf{C}\mathbf{W}^{-\frac{1}{2}})^T\mathbf{C}\mathbf{W}^{-\frac{1}{2}} = \tilde{\Phi}^T\tilde{\Phi}$. Let $\tilde{\mathbf{\Sigma}}_X = \tilde{\Phi}_X^T\tilde{\Phi}_X$ and $\tilde{\mathbf{\Sigma}}_Y = \tilde{\Phi}_Y^T\tilde{\Phi}_Y$ denote the uncentered covariance matrices of view $\mathbf{X}$ and $\mathbf{Y}$ respectively. To obtain an approximate estimator of the HSIC, given in (1), the $\tilde{\Phi}_X$ and $\tilde{\Phi}_Y$ are first centered through $\hat{\Phi} = (\mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\tilde{\Phi}$, where $\hat{\Phi} \in \mathbb{R}^{n \times n_{nys}}$. Using the approximated kernel functions $\hat{\Phi}_X$ and $\hat{\Phi}_Y$, the biased Nyström estimator of HSIC, $\tilde{\rho}$, is given by

$$\tilde{\rho} = ||\frac{1}{n}\hat{\Phi}_X^T\hat{\Phi}_Y||_F^2. \quad (3)$$

The Nyström approximations reduce the computational cost to $\mathcal{O}(n_{nys}(n + n_{nys}^2)d)$, which is less than $\mathcal{O}(n^2)$ when $n_{nys}^3 < n^2$.

We sample the columns uniformly without replacement, which is cheap and efficient, and also the most popular method in practice [20].

III. Experiments

The performance of SCCA-HSIC is evaluated on both simulated and the publicly available Boston housing [21] and body fat [22] datasets [1]. We use the Gaussian kernel as the non-linear kernel due to its universality property [15]. The standard deviation of every kernel is set by the median heuristic. All variables are standardized to have a zero mean and unit variance. In all experiments, to obtain an optimal result for each method, we tune the hyperparameters by repeated 3-fold cross validation. For SCCA-HSIC, TSKCCA, and SCCA, we tune the ℓ_1 norm constraints. For CCA-HSIC, the ℓ_2 norm constraints are tuned. For KCCA, the regularization constants are optimized. For DCCA, which is applied only in simulation studies, we apply three hidden layers with 18 units in each layer. When extracting a linear relation, we apply a linear activation function and for all the other relations a sigmoid activation function. We tune the learning rate and the momentum hyperparameters.

A. Evaluation Metrics

As in [13], to have a consistent metric for assessing the predictiveness of a relation of any form, we employ the test HSIC value which is obtained by computing the Gram matrices $\mathbf{K}^{\mathbf{u}}$ and $\mathbf{K}^{\mathbf{v}}$ of (2) on the test data $\mathbf{X}_{test}\mathbf{u}$ and $\mathbf{Y}_{test}\mathbf{v}$ using the projections $\mathbf{u}$ and $\mathbf{v}$ obtained form the training data.

[1]The MATLAB codes are available on https://github.com/aalto-ics-kepaco/scca-hsic.

In addition to the test HSIC, when the ground truth is known, we apply the F1 score $F1 = \frac{2\text{TP}}{2\text{TP}+\text{FP}+\text{FN}}$ where TP, FP, and FN denote the true positives, false positives, and false negatives respectively. In our case, the labels refer to the zero/non-zero coefficient values in the estimated vector $\mathbf{u}$ (resp. $\mathbf{v}$) and the ground truth. For KCCA, the explicit projections are not available, so to compare KCCA with the other methods, we compute approximations of the projection directions by $\tilde{\mathbf{u}} = \mathbf{X}_{train}^T\boldsymbol{\alpha}$ and $\tilde{\mathbf{v}} = \mathbf{Y}_{train}^T\boldsymbol{\beta}$. For KCCA and CCA-HSIC methods that produce non-sparse projections, the most highly activated entries are picked out by first scaling the vectors so that the absolute values are between zero and one, and zeroing out all entries less than 0.05 setting the rest of the entries equal to 1.

B. Finding Non-Linear Relations

In this experiment, we analyze how SCCA-HSIC, TSKCCA, CCA-HSIC, KCCA, DCCA and SCCA extract linear, sinusoidal, and hyperbolic multivariate relations. Every relation is simulated in a single dataset. For every relation, we generate two data matrices $\mathbf{X}$ and $\mathbf{Y}$ of sizes $n \times p$ and $n \times q$, where $n = 300$, $p = 20$ and $q = 20$ respectively. The variables in the columns, denoted by superscripts, of $\mathbf{X}$ and $\mathbf{Y}$ are generated from a random univariate normal distribution, $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{20} \sim N(0,1)$ and $\mathbf{y}^3, \mathbf{y}^4, \ldots, \mathbf{y}^{20} \sim N(0,1)$. The three relations are simulated in three datasets by $\mathbf{y}^1 + \mathbf{y}^2 = f(\mathbf{x}^1 + \mathbf{x}^2) + \boldsymbol{\xi}$ where $\boldsymbol{\xi} \sim N(0, 0.05)$ denotes a vector of normal noise and the relations are the linear $f(\mathbf{x}) = \mathbf{x}$, sinusoidal $f(\mathbf{x}) = \sin \mathbf{x}$, and hyperbolic $f(\mathbf{x}) = 1/\mathbf{x}$.

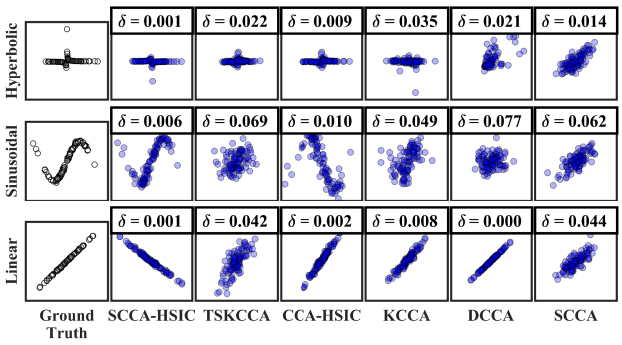

Figure 2. The transformations on test data are shown. The horizontal and vertical axes of every plot correspond to $\mathbf{X}_{\text{test}}\mathbf{u}$ and $\mathbf{Y}_{\text{test}}\mathbf{v}$ respectively. The columns correspond to the extracted relation and rows to the method. The δ corresponds to the difference between the predicted HSIC and the ground truth HSIC.

The transformations on test data are shown in Figure 2. The linear relations, visualized in the first column, are best extracted by DCCA, SCCA-HSIC, and CCA-HSIC. KCCA also performs reasonably well. SCCA and TSKCCA perform similarly in extracting linear relations which may result from the penalized matrix decomposition procedure used in both. As for the the non-linear relations, SCCA-HSIC finds both

the sinusoidal, in the second column, and the hyperbolic, in the third column, relations most clearly and accurately. CCA-HSIC is also able to find the sinusoidal relation, but the hyperbolic relation is not accurately extracted. The other methods do not extract the sinusoidal and hyperbolic patterns accurately. As in the case of the linear relation, although TSKCCA maximizes HSIC between the feature-wise kernels the penalized matrix decomposition procedure may make the extraction of multivariate relations difficult.

C. Predictive Performance

We compare SCCA-HSIC, TSKCCA, CCA-HSIC, KCCA, and DCCA when the number of related and noise variables increases. We analyze the performance using three polynomial relations of the form $f(\mathbf{x}) = \mathbf{x}^d$, where $d = 1, 2, 3$ that is linear, quadratic, and cubic relations respectively. We also analyze two transcendental relations, the exponential and the logarithmic relations, of the form $f(\mathbf{x}) = \exp(\mathbf{x})$ and $f(\mathbf{x}) = \log(\mathbf{x})$ respectively. We resample every simulated relation 10 times to report average performance.

As performance metrics, we apply the test HSIC and the F1 score on $\mathbf{u}$ and $\mathbf{v}$. The reported test HSIC values and F1 scores are averages over the 10 repetitions and the five simulated relations. The final reported F1 score is the average of the F1 scores computed for $\mathbf{u}$ and $\mathbf{v}$.

1) Increasing the Number of Related Variables: This experiment shows how SCCA-HSIC, TSKCCA, CCA-HSIC, KCCA, and DCCA perform when the number of related variables in the multivariate functions increases. We generate two data matrices $\mathbf{X}$ and $\mathbf{Y}$ of sizes $n \times p$ and $n \times q$, where $n = 300$, $p = 20$ and $q = 20$ respectively. The variables in the columns of $\mathbf{X}$ and $\mathbf{Y}$ are generated from a random univariate uniform distribution, $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{20} \sim U[0, 1]$ and $\mathbf{y}^3, \mathbf{y}^4, \ldots, \mathbf{y}^{20} \sim U[0, 1]$. The simulated functions, given in the first paragraph of the Section III-C, are of the form $\mathbf{y}^1 + \mathbf{y}^2 = f(\sum_{i=1}^{l} \mathbf{x}^i) + \boldsymbol{\xi}$ where $l = 1, 2, 3, 4$ and $\boldsymbol{\xi} \sim N(0, 0.05)$ denotes a vector of normal noise.

The results are shown in Figure 3 (A). SCCA-HSIC is marginally better than CCA-HSIC and KCCA in terms of test HSIC, independently from the number of related variables, with TSKCCA and DCCA significantly worse. In terms of F1 score, SCCA-HSIC is clearly more accurate than the competing methods, independently of the number of related variables.

2) Increasing the Number of Noise Variables: This experiment demonstrates how SCCA-HSIC, TSKCCA, CCA-HSIC, KCCA, and DCCA perform when the number of noise variables increases. The functions, given in the first paragraph of the Section III-C, are of the form $\mathbf{y}^1 + \mathbf{y}^2 = f(\mathbf{x}^1 + \mathbf{x}^2 + \mathbf{x}^3) + \boldsymbol{\xi}$ where $\boldsymbol{\xi} \sim N(0, 0.05)$ denotes a vector of normal noise. We generate two data matrices $\mathbf{X}$ and $\mathbf{Y}$ of sizes $n \times p$ and $n \times q$, where $n = 300$ respectively. The dimensions tested are $p = q = 10, 20, 30, 40$. The variables in the columns of $\mathbf{X}$ and $\mathbf{Y}$ are generated from a random univariate uniform distribution, $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^p \sim U[0, 1]$ and $\mathbf{y}^3, \mathbf{y}^4, \ldots, \mathbf{y}^q \sim U[0, 1]$.

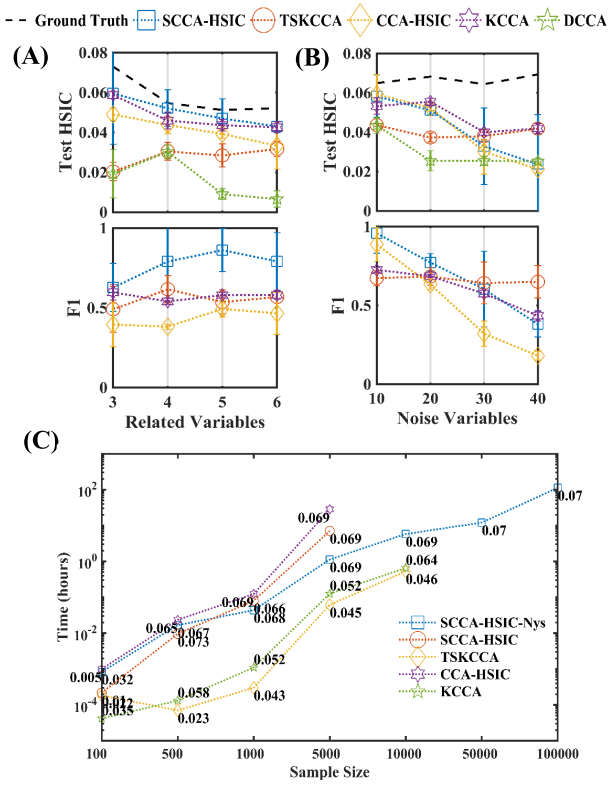

Figure 3. The results when the number of related and noise variables increases are shown in (A) and (B) respectively. The test HSIC and F1 score are shown on the first and second row respectively. (C) The performance of SCCA-HSIC-Nys, SCCA-HSIC, TSKCCA, and KCCA when the sample size increases. The points on the plot are labeled by the final test HSIC value at the convergence. The computation time is given in hours on a logarithmic scale. TSKCCA and KCCA cannot be performed at the sample size of 50000 or greater due to memory requirements of 165.6 GB and 16.6 GB respectively. CCA-HSIC and SCCA-HSIC were not performed at sample sizes greater than 5000 since the computation times at 5000 were 29 and 7 hours, respectively, in comparison to 1.1, 0.06, and 0.12 hours for SCCA-HSIC-Nys, TSKCCA, and KCCA respectively.

The results are shown in Figure 3 (B). The test HSIC plot shows that SCCA-HSIC, CCA-HSIC, and KCCA tolerate noise equally until the number of 30 noise variables. TSKCCA and KCCA perform similarly at the number of 40 noise variables. In terms of F1 score, SCCA-HSIC outperforms the other methods until the number of 30 noise variables. TSKCCA and DCCA follow a different pattern, having a low but stable test HSIC value independent of the number of noise variables. In the case of TSKCCA, where the coefficients are available and the F1 score can be computed, it can be deduced that it finds some of the related variables correctly, regardless of the number of noise variables. For DCCA, we cannot compute the F1 score since we do not have access to the coefficient vectors.

D. Scalability of Nyström Approximated SCCA-HSIC

This experiment demonstrates the scalability of Nyström approximated SCCA-HSIC, that will be referred to as SCCA-HSIC-Nys. We compare SCCA-HSIC-Nys with the kernel-based methods, that is TSKCCA, CCA-HSIC, and KCCA. Additionally, we compare SCCA-HSIC-Nys with the unapproximated SCCA-HSIC. We apply the KCCA version of [8] that employs the partial Gram-Schmidt orthogonalization for decomposing the kernel matrices. All computations are performed on a MacBook Pro with Intel Core i7 (2.2 GHz quad core processor) with 16 GB main memory.

We generate two data matrices $\mathbf{X}$ and $\mathbf{Y}$ of sizes $n \times p$ and $n \times q$, where $p = 20$ and $q = 20$ respectively. The sample sizes tested are $n = 100, 500, 1000, 5000, 10000, 50000$, and 100000. Due to memory requirements KCCA and TSKCCA are only run with maximum of 10000 examples, and due to time-consumption, CCA-HSIC and SCCA-HSIC are only run with up to 5000 examples.

The variables in the columns of $\mathbf{X}$ and $\mathbf{Y}$ are generated from a random univariate uniform distribution, $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{20} \sim U[0,1]$ and $\mathbf{y}^3, \mathbf{y}^4, \ldots, \mathbf{y}^{20} \sim U[0,1]$. As the simulated relation, we select the exponential multivariate relation of the form $\mathbf{y}^1 + \mathbf{y}^2 = \exp(\mathbf{x}^1 + \mathbf{x}^2 + \mathbf{x}^3) + \boldsymbol{\xi}$ where $\boldsymbol{\xi} \sim N(0, 0.05)$ denotes a vector of normal noise. The hyperparameter π, that is the proportion of columns for SCCA-HSIC-Nys, is set to $\pi = \{1, 1, 0.5, 0.1, 0.05, 0.01, 0.005\}$ for the various sample sizes respectively.

The results are shown in Figure 3 (C). In terms of test HSIC, SCCA-HSIC-Nys outperforms on both small and large datasets. It is more accurate in identifying the simulated relation than the faster TSKCCA and KCCA. In general, CCA-HSIC and SCCA-HSIC find the simulated relation equally well but at a much longer computation time which is best seen when the sample size is 5000.

E. Real Datasets

1) Boston Housing Dataset: We analyze the Boston housing dataset [21], [23]. The dataset was analyzed for determining a household's willingness to pay for air quality improvements in the Boston metropolitan area. It contains 506 observations of 16 variables. The descriptions of the variables can be found in [23]. Here we use the standard abbreviations. A negative non-linear relation between the median value of the owner occupied homes (CMEDV) and lower socioeconomic status (LSTAT) has been previously reported in [13]. For the two-view analysis, the variables TRACT, LON, LAT, CMEDV, CRIM, ZN, INDUS, and NOX form the first view and RM, AGE, DIS, RAD, PTRATIO, B, and LSTAT form the second view.

The test results in terms of test HSIC, ρ and the related variables are shown in Figure 4. SCCA-HSIC finds the transformations of highest HSIC value for all three components, while CCA-HSIC and TSKCCA alternate for the second and third best position, KCCA being the weakest methods in this sense. In particular, the known non-linear relation between CMEDV and LSTAT is captured in the third component of SCCA-HSIC. The first components of SCCA-HSIC, TSKCCA, and CCA-HSIC are intrinsically capturing the same pattern, that is the TRACT is positively related with RAD and TAX. TSKCCA selects the same variables, RAD and TAX for the second view in all components and associates them with a single variable from the first view. The second and third components of CCA-HSIC and all components of KCCA are difficult to interpret due to the lack of sparsity.

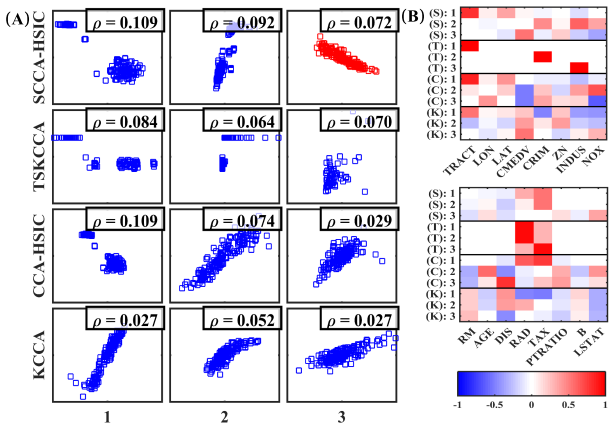

Figure 4. (A) The numbers correspond to the three leading test transformations of the Boston housing dataset. The horizontal and vertical axes of every plot correspond to $\mathbf{X}_{\text{test}}\mathbf{u}$ and $\mathbf{Y}_{\text{test}}\mathbf{v}$ respectively. The test HSIC values are indicated by ρ. (B) The heatmaps visualize the $\mathbf{u}$ and $\mathbf{v}$.

2) Body Fat Dataset: The body fat dataset [22] contains 252 observations of 15 variables. The variables describe estimates of the percentage of body fat determined by underwater weighing and various body circumference measurements for 252 men. Five of the variables, which are density determined by underwater weighing, percent body fat, age, weight, and height, form the first view. The second view describes the circumference measurements, that is neck, chest, abdomen, hip, thigh, knee, ankle, biceps, forearm, and wrist circumference measurements.

The leading three test transformations and the corresponding $\mathbf{u}$ and $\mathbf{v}$ are visualized in Figure 5. The HSIC values of the first test transformations of SCCA-HSIC, TSKCCA, and CCA-HSIC are very similar. SCCA-HSIC identifies a multivariate relation between percentage body fat, weight, height, and abdominal circumference. Height has a negative effect on the abdominal circumference while body fat and weight increase it. CCA-HSIC finds the same relation except that the circumference view is not as straightforward to interpret. TSKCCA also identifies a relation between weight and abdominal circumference but since it can only assign positive values on the variables it does not find the negative

contribution of the height.

SCCA-HSIC finds the relations of highest test HSIC in the second and third transformations. In the second transformation, SCCA-HSIC identifies that weight can also have a negative effect on the abdominal circumference. The third transformation of SCCA-HSIC shows that chest and hip circumferences increase with increasing weight. This pattern is also picked up in the first transformation of TSKCCA. As in the results of the Boston housing dataset, the relations captured by CCA-HSIC and KCCA cannot be determined since the coefficients of almost all variables are nonzero.

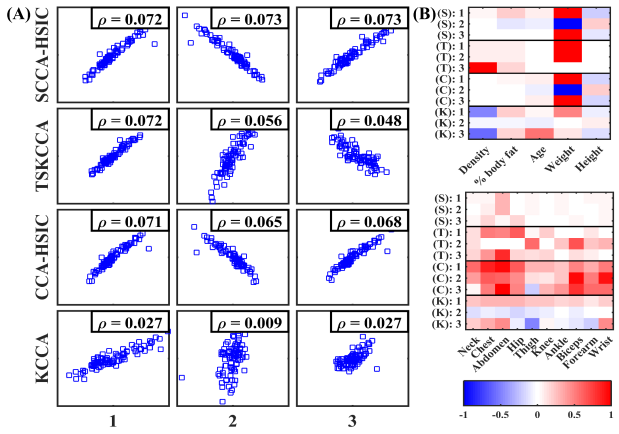

Figure 5. (A) The numbers correspond to the three leading transformations of the body fat dataset. See the caption of Figure 4. (B) The heatmaps visualize the $\mathbf{u}$ and $\mathbf{v}$.

IV. Discussion

In this paper, we have introduced SCCA-HSIC, a method that extends the family of canonical correlation methods for finding sparse and non-linear multivariate relations from potentially high-dimensional and noisy data. The technical contributions of the paper include the application of the projected stochastic gradient ascent for maximizing the HSIC and the use of the Nyström approximation to improve the tractability in large-scale settings.

To conclude, we have established SCCA-HSIC as an effective method for finding sparse, potentially non-linear, multivariate relations in high-dimensional two-view data settings. The method is applicable to both small and large datasets, however, the non-convex formulation of the problem makes it more computationally intensive than previous convex models such as (K)CCA. This deficiency is in our opinion more than compensated by the interpretability (due to sparsity) and the generality of relations (due to HSIC) that can be uncovered accurately.

Acknowledgment

This work has been supported by Academy of Finland grants 295296 (D4Health) and 313268 (TensorBiomed).

References

[1] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3, pp. 321–377, 1936.

[2] V. Uurtio, J. Monteiro, J. Kandola, J. Shawe-Taylor, D. Fernandez-Reyes, and J. Rousu, "A tutorial on canonical correlation methods," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 95:1–95:33, Nov. 2017.

[3] S. Leurgans, R. Moyeed, and B. Silverman, "Canonical correlation analysis when the data are curves," *J R Stat Soc Ser A Stat Soc*, pp. 725–740, 1993.

[4] A. Cichonska, J. Rousu, P. Marttinen, A. Kangas, P. Soininen *et al.*, "metacca: summary statistics-based multivariate meta-analysis of genome-wide association studies using canonical correlation analysis," *Bioinformatics*, vol. 32, no. 13, pp. 1981–1989, 2016.

[5] D. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, p. kxp008, 2009.

[6] J. Rousu, D. Agranoff, O. Sodeinde, J. Shawe-Taylor, and D. Fernandez-Reyes, "Biomarker discovery by sparse canonical correlation analysis of complex clinical phenotypes of tuberculosis and malaria," *PLoS Comput. Biol.*, vol. 9, no. 4, p. e1003018, 2013.

[7] V. Uurtio, M. Bomberg, K. Nybo, M. Itävaara, and J. Rousu, "Canonical correlation methods for exploring microbe-environment interactions in deep subsurface," in *International Conference on Discovery Science*. Springer, 2015, pp. 299–307.

[8] D. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural computation*, vol. 16, no. 12, pp. 2639–2664, 2004.

[9] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *ICML*, 2013, pp. 1247–1255.

[10] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf, "Kernel methods for measuring independence," *JMLR*, vol. 6, no. Dec, pp. 2075–2129, 2005.

[11] H. Shen, S. Jegelka, and A. Gretton, "Fast kernel-based independent component analysis," *IEEE Trans Signal Process*, vol. 57, no. 9, pp. 3498–3511, 2009.

[12] J. Chen, M. Stern, M. J. Wainwright, and M. Jordan, "Kernel feature selection via conditional covariance minimization," in *Adv Neural Inf Process Syst*. Curran Associates, Inc., 2017, pp. 6949–6958.

[13] B. Chang, U. Kruger, R. Kustra, and J. Zhang, "Canonical correlation analysis based on hilbert-schmidt independence criterion and centered kernel target alignment," in *ICML*, 2013, pp. 316–324.

[14] K. Yoshida, J. Yoshimoto, and K. Doya, "Sparse kernel canonical correlation analysis for discovery of nonlinear interactions in high-dimensional data," *BMC bioinformatics*, vol. 18, no. 1, p. 108, 2017.

[15] A. Gretton, K. Fukumizu, C. Teo, L. Song, B. Schölkopf, and A. Smola, "A kernel statistical test of independence," in *Adv Neural Inf Process Syst*, 2008, pp. 585–592.

[16] L. Mackey, "Deflation methods for sparse pca," in *Adv Neural Inf Process Syst*, 2009, pp. 1017–1024.

[17] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the l 1-ball for learning in high dimensions," in *ICML*. ACM, 2008, pp. 272–279.

[18] Q. Zhang, S. Filippi, A. Gretton, and D. Sejdinovic, "Large-scale kernel methods for independence testing," *Stat Comput*, vol. 28, no. 1, pp. 113–130, 2018.

[19] C. Williams and M. Seeger, "Using the nyström method to speed up kernel machines," in *Adv Neural Inf Process Syst*, no. EPFL-CONF-161322, 2001, pp. 682–688.

[20] S. Kumar, M. Mohri, and A. Talwalkar, "Sampling methods for the nyström method," *JMLR*, vol. 13, no. Apr, pp. 981–1006, 2012.

[21] D. Harrison Jr and D. L. Rubinfeld, "Hedonic housing prices and the demand for clean air," *J Environ Econ Manage*, vol. 5, no. 1, pp. 81–102, 1978.

[22] K. W. Penrose, A. Nelson, and A. Fisher, "Generalized body composition prediction equation for men using simple measurement techniques," *Medicine & Science in Sports & Exercise*, vol. 17, no. 2, p. 189, 1985.

[23] R. K. Pace and O. W. Gilley, "Using the spatial configuration of the data to improve estimation," *The Journal of Real Estate Finance and Economics*, vol. 14, no. 3, pp. 333–340, 1997.