Abstract

In this supplementary materials, we provide more details about the concepts of conditional independence and cross-covariance operator. Furthermore, we report more experiments on tuning model parameter and discovering meaningful intrinsic dimension for helping readers better understand our proposed algorithms.

1 Conditional independence and Cross-covariance operator

In this section, we will introduce two important concepts, conditional independence and cross-covariance operators, which have close relationship to our proposed KDR-tree algorithms.

1.1 Conditional independence

To achieve conditional independence under a low-dimensional subspace, we can decompose the original m-dimensional explanatory variable \( X \) into two parts, \( U = B^T X \) and \( V = C^T X \). When \((B, C)\) is an orthogonal matrix, we have the following relationship of probability density functions between the original variable and its decomposition [4]:

\[
p_X(x) = p_{U,V}(u, v), \quad p_{X,Y}(x, y) = p_{U,V,Y}(u, v, y).
\]

Considering the conditional independence, \( Y \perp\perp X | B^T X \), we have the following equation [4]:

\[
p_{Y|U,V}(y|u, v) = p_{Y|U}(y|u).
\]

It means that \( Y \) and \( V \) are conditional independence given \( U \).

1.2 Cross-covariance operator

Similar to the definition of covariance, [1] proposed the concept of cross-covariance operator under a real and separable space. Formally, let \( H_x(H_y) \) be a real and separable Hilbert space with inner product \( \langle \cdot, \cdot \rangle_{H_x} \) \( (\langle \cdot, \cdot \rangle_{H_y}) \) and Borel \( \sigma \)-field \( \Gamma_x(\Gamma_y) \). Define \( H_x \times H_y = \{ (u, z) : u \in H_x, z \in H_y \} \).

Then the cross-covariance operator is defined as [1]:

\[
\langle \Sigma_{XY} z, u \rangle_{H_x} = \int_{H_x \times H_y} \langle x - m_x, u \rangle_{H_x} \langle y - m_y, z \rangle_{H_y} d\mu_{XY}(x, y).
\]

Here \( m_x \) and \( m_y \) are the mean of \( X \in H_x \) and \( Y \in H_y \), respectively. And \( \mu_{XY} \) denotes a joint measure on \((H_x \times H_y, \Gamma_x \times \Gamma_y)\).

To generalize the cross-covariance operator to the reproducing kernel Hilbert space, we here describe the following reproducing property of RKHS at first:

\[
\langle f, k(\cdot, x) \rangle_{H} = f(x) \quad \text{for all} \quad x \in \Omega \text{ and } f \in H.
\]
where \((\mathcal{H}, k)\) is a reproducing kernel Hilbert space of functions on a set \(\Omega\) with a positive definite kernel \(k : \Omega \times \Omega \to \mathbb{R}\). A commonly used kernel is the Gaussian radial basis function kernel as follows:
\[
k(x_1, x_2) = \exp(-\|x_1 - x_2\|^2 / 2\sigma^2).
\] (5)

Then the cross-covariance operator from RKHS \(\mathcal{H}_x\) to another RKHS \(\mathcal{H}_y\) is defined as follows [4]:
\[
\langle g, \Sigma_{Y,X} f \rangle_{\mathcal{H}_y} = \mathbb{E}_{XY}[f(X)g(Y)] - \mathbb{E}_X[f(X)]\mathbb{E}_Y[g(Y)],
\] (6)
for all \(f \in \mathcal{H}_x\) and \(g \in \mathcal{H}_y\) [4].

Note that \(\Sigma_{Y,X}\) has the following representation [11]:
\[
\Sigma_{Y,X} = \Sigma_{Y,Y}^{1/2} V_{Y,X} \Sigma_{X,X}^{1/2},
\] (7)
where \(V_{Y,X} : \mathcal{H}_x \to \mathcal{H}_y\) is a unique bounded operator such that \(\|V_{Y,X}\| \leq 1\).

At last, the conditional covariance operator \(\Sigma_{Y|X}\) used in our paper is defined based on cross-covariance operator as follows:
\[
\Sigma_{Y|X} = \Sigma_{Y,Y} - \Sigma_{Y,X} \Sigma_{X,X}^{-1} \Sigma_{X,Y}.
\] (8)

2 Analysis of the model parameters

Here we would like to introduce more experimental results on parameter influence, which were not reported in the paper for saving space.

- **Regularization parameter \(\epsilon\)**: The aim of the regularization parameter is to enable the inversion of the kernel matrix \(G\). Generally speaking, this regularization should be set a small number. Conversely, a large regularization parameter distorts the information contained in the kernel matrix. Therefore, we fix the bandwidth \(\sigma^2\) of kernel matrix be the median of pair squared distances and vary the regularization parameter \(\epsilon\) from \(10^{-6}\) to \(0.5\). The associated results are reported in Fig. [1]. Note that the red points shown in Fig. [1] are obtained when we empirically set the regularization be \(10^{-4}\) in our paper. Furthermore, it can be seen from the figure that selecting the parameter to be \(10^{-4}\) produces a good performance in all applications. In addition, we observed the performance does not strongly depends on the selection of the regularization parameter.

![Figure 1: Tuning regularization term \(\epsilon\) on 4 reference sets: Sonar, Synthetic_control, Waveform_noise and Letter.](image)

- **Bandwidth parameter \(\sigma^2\)**: One approach to select the bandwidth parameter is to utilize cross-validation for supervised learning. For unsupervised learning, an alternative and widely used way is to choose the median of the pair squared distances of data [2,6,8]. We show the effect of bandwidth to our proposed method in Fig. [2]. The red points shown in Fig. [2] represent the results of bandwidth being set as median of pair squared distances. The remaining points in this figure were selected by varying \(\sigma^2\). It can be seen that the median of pair squared distances almost achieves the good performance in all applications.

- **Size of subsets**: We analyze the performance of sKDR-tree on four benchmark datasets, and report the average of 20 random repetitions. The results of different size of subsets are shown in Fig. [3]. Here red dot in each subplot represents the result reported in the original
Figure 2: Tuning bandwidth term $\sigma^2$ on 4 reference sets: Sonar, Synthetic control, Waveform noise and Letter.

Figure 3: Varying size of subset on 4 reference sets: Sonar, Synthetic control, Waveform noise and Letter.

paper. Moreover, the result with the largest size of subsets is equal to that of KDR-tree. Fig. 3 outlines that, as expected, the larger the size of subset is, the more accuracy the projection direction estimates. For a large reference set, sKDR-tree can offer significant computation savings without significantly compromising performance.

3 Discovering intrinsic dimension on more datasets

In this section, we illustrate more visualization results that are used to discover the intrinsic dimension from high-dimensional data.

- **Hand rotation data set**: This data set contains 481 image samples, and each sample is a vector in a 3048-dimensional space. These hand images are collected under one degree of freedom, i.e., horizontally rotating a cup-in-hand. It means that the data points can be viewed as lying on a one-dimensional curve embedded in the original high-dimensional image space. Considering its intrinsic dimension, we utilize rKDR-tree to split the data points into a two-layer tree, visualizing them in a 2-dimensional space. The visualization of result is shown in right plot of Fig. 4. It is obvious that the meaningful intrinsic dimension of data set is the rotation of hand, which has been confirmed by Isomap algorithm in left plot of Fig. 4.

- **MNIST digit ‘1’**: An obviously meaningful intrinsic dimension of this dataset is the direction of digit ‘1’ from Fig. 5. As the depth increases, a significant difference between two child nodes whose parents are in depth 2 is that one is a thick ‘1’, the other is a thin ‘1’. It indicates that the second meaningful intrinsic dimension of this dataset may be the thickness of digit ‘1’. This can also be confirmed in Fig. 6. The left and right plots in Fig. 6 are the visualization results of residual samples in the left and right parts of Fig. 5 with removing the first projection information, respectively. We can see the thickness of digit ‘1’ changes from bottom to top in Fig. 6. Note that the first two projection directions corresponds to the direction of digit ‘1’ since the direction variable may dominate the data distribution and cannot be wholly removed by using only one projection.

\[\text{CMU database} \text{http://vasc.ri.cmu.edu/idb/html/motion/hand/index.html}\]
• **Frey face data set** [7]: By observing the first and second dimension or projection directions, it can be seen that our proposed rKDR-tree can discover two meaningful intrinsic dimensions, i.e., pose and expression variables, in left plot of Fig. 7. Roweis and Lawrance [7] outlined that these are the intrinsic dimensions of this data set, confirming that the application of the rKDR-tree correctly reveals an intrinsic dimension of two for this image set.

• **MNIST digit ‘2’** [6]: rKDR-tree can discover two meaningful intrinsic dimensions in the middle plot of Fig 7. From the figure it can be seen that the first and second dimension or projection directions corresponds to top arch articulation and bottom loop articulation of digit ‘2’. Both of them have been confirmed by Isomap algorithm [9].

• **MNIST digit ‘0’** [6]: From right plot of Fig. 7 a meaningful intrinsic dimension discovered by rKDR-tree is the rotation of digit ‘0’.

Figure 4: Left: The 2-dimensional embedding of hands rotation data sets by Isomap algorithm [9]. Right: Partitioning result of Hand rotation data set using rKDR-tree of depth 2.

Figure 5: Left: Partitioning result of MNIST digit ‘1’ using rKDR-tree of depth 2. Right: Partitioning result of MNIST digit ‘1’ using rKDR-tree of depth 3. Each node is represented by the mean of data points belonging to this node.
Figure 6: Left: Partitioning result of the left child of root with removing the first projection information, showing the second and third projection directions. Right: Partitioning result of the right child of root with removing the first projection information, showing the second and third projection directions. An additional meaningful intrinsic dimension is the thickness of digit ‘1’.

References


Figure 7: Partitioning result of Frey face data set, MNIST digit ‘2’ and MNIST digit ‘0’ using rKDR-tree of depth 2.