LETTER Communicated by Qin Wang

Conditional Density Estimation with Dimensionality Reduction via Squared-Loss Conditional Entropy Minimization

Voot Tangkaratt
voot@sg.cs.titech.ac.jp
Ning Xie
xie@sg.cs.titech.ac.jp
Masashi Sugiyama
sugi@cs.titech.ac.jp
Department of Computer Science, Tokyo Institute of Technology, Meguro-ku, Tokyo 152-8552, Japan

Regression aims at estimating the conditional mean of output given input. However, regression is not informative enough if the conditional density is multimodal, heteroskedastic, and asymmetric. In such a case, estimating the conditional density itself is preferable, but conditional density estimation (CDE) is challenging in high-dimensional space. A naive approach to coping with high dimensionality is to first perform dimensionality reduction (DR) and then execute CDE. However, a two-step process does not perform well in practice because the error incurred in the first DR step can be magnified in the second CDE step. In this letter, we propose a novel single-shot procedure that performs CDE and DR simultaneously in an integrated way. Our key idea is to formulate DR as the problem of minimizing a squared-loss variant of conditional entropy, and this is solved using CDE. Thus, an additional CDE step is not needed after DR. We demonstrate the usefulness of the proposed method through extensive experiments on various data sets, including humanoid robot transition and computer art.

1 Introduction

Analyzing an input-output relationship from samples is one of the central challenges in machine learning. The most common approach is regression, which estimates the conditional mean of output $\boldsymbol{y}$ given input $\boldsymbol{x}$. However, just analyzing the conditional mean is not informative enough, when the conditional density $p(\boldsymbol{y}|\boldsymbol{x})$ possesses multimodality, asymmetry, and heteroskedasticity (i.e., input-dependent variance) as a function of output $\boldsymbol{y}$. In such cases, it would be more appropriate to estimate the conditional density itself (see Figure 2).

Neural Computation **27, 228–254** (2015)
doi:10.1162/NECO_a_00683

The most naive approach to conditional density estimation (CDE) would be ϵ-neighbor kernel density estimation (ϵ-KDE), which performs standard KDE along $\boldsymbol{y}$ only with nearby samples in the input domain. However, ϵ-KDE does not work well in high-dimensional problems because the number of nearby samples is too few. To avoid the small sample problem, KDE may be applied twice to estimate $p(\boldsymbol{x}, \boldsymbol{y})$ and $p(\boldsymbol{x})$ separately and the estimated densities may be plugged into the decomposed form $p(\boldsymbol{y}|\boldsymbol{x}) = p(\boldsymbol{x}, \boldsymbol{y})/p(\boldsymbol{x})$ to estimate the conditional density. However, taking the ratio of two estimated densities significantly magnifies the estimation error and thus is not reliable. To overcome this problem, an approach to directly estimating the density ratio $p(\boldsymbol{x}, \boldsymbol{y})/p(\boldsymbol{x})$ without separate estimation of densities $p(\boldsymbol{x}, \boldsymbol{y})$ and $p(\boldsymbol{x})$ has been explored (Sugiyama et al., 2010). This method, called least-squares CDE (LSCDE), was proved to possess the optimal nonparametric learning rate in the mini-max sense, and its solution can be efficiently and analytically computed. Nevertheless, estimating conditional densities in high-dimensional problems is still challenging.

A natural idea to cope with the high dimensionality is to perform dimensionality reduction (DR) before CDE. Sufficient DR (Li, 1991; Cook & Ni, 2005) is a framework of supervised DR aimed at finding the subspace of input $\boldsymbol{x}$ that contains all information on output $\boldsymbol{y}$, and a method based on conditional-covariance operators in reproducing kernel Hilbert spaces has been proposed (Fukumizu, Bach, & Jordan, 2009). Although this method possesses superior theoretical properties, it is not easy to use in practice because no systematic model selection method is available for kernel parameters. To overcome this problem, an alternative sufficient DR method based on squared-loss mutual information (SMI) has been proposed recently (Suzuki & Sugiyama, 2013). This method involves nonparametric estimation of SMI that is theoretically guaranteed to achieve the optimal estimation rate, and all tuning parameters can be systematically chosen in practice by cross-validation with respect to the SMI approximation error.

Given such state-of-the-art DR methods, performing DR before LSCDE would be a promising approach to improving the accuracy of CDE in high-dimensional problems. However, such a two-step approach is not preferable because DR in the first step is performed without regard to CDE in the second step, and thus small errors incurred in the DR step can be significantly magnified in the CDE step.

In this letter, we propose a single-shot method that integrates DR and CDE. Our key idea is to formulate the sufficient DR problem in terms of the squared-loss conditional entropy (SCE), which includes the conditional density in its definition, and LSCDE is executed when DR is performed. Therefore, when DR is completed, the final conditional density estimator has already been obtained without an additional CDE step (see Figure 1). We demonstrate the usefulness of the proposed method, named least-squares conditional entropy (LSCE), through experiments on benchmark data sets, humanoid robot control simulations, and computer art.

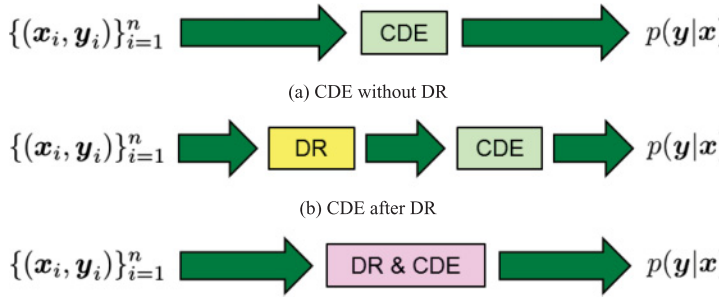

Figure 1: Conditional density estimation (CDE) and dimensionality reduction (DR). (a) CDE without DR performs poorly in high-dimensional problems. (b) CDE after DR can magnify the small DR error in the CDE step. (c) CDE with DR (proposed) performs CDE in the DR process in an integrated manner.

2 Conditional Density Estimation with Dimensionality Reduction

In this section, we describe our proposed method for conditional density estimation with dimensionality reduction.

2.1 Problem Formulation. Let $\mathcal{D}_{\mathbf{x}}(\subset \mathbb{R}^{d_{\mathbf{x}}})$ and $\mathcal{D}_{\mathbf{y}}(\subset \mathbb{R}^{d_{\mathbf{y}}})$ be the input and output domains with dimensionality $d_{\mathbf{x}}$ and $d_{\mathbf{y}}$, respectively, and let $p(\boldsymbol{x}, \boldsymbol{y})$ be a joint probability density on $\mathcal{D}_{\mathbf{x}} \times \mathcal{D}_{\mathbf{y}}$. Assume that we are given n independent and identically distributed (i.i.d.) training samples from the joint density:

$$\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, \boldsymbol{y}).$$

The goal is to estimate the conditional density $p(\boldsymbol{y}|\boldsymbol{x})$ from the samples.

Our implicit assumption is that the input dimensionality $d_{\mathbf{x}}$ is large, but its intrinsic dimensionality, denoted by $d_{\mathbf{z}}$, is rather small. More specifically, let $\boldsymbol{W}$ and $\boldsymbol{W}_\perp$ be $d_{\mathbf{z}} \times d_{\mathbf{x}}$ and $(d_{\mathbf{x}} - d_{\mathbf{z}}) \times d_{\mathbf{x}}$ matrices such that $[\boldsymbol{W}^\top, \boldsymbol{W}_\perp^\top]$ is an orthogonal matrix. Then we assume that $\boldsymbol{x}$ can be decomposed into the component $\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}$ and its perpendicular component $\boldsymbol{z}_\perp = \boldsymbol{W}_\perp \boldsymbol{x}$ so that $\boldsymbol{y}$ and $\boldsymbol{x}$ are conditionally independent given $\boldsymbol{z}$:

$$\boldsymbol{y} \perp \boldsymbol{x} | \boldsymbol{z}. \tag{2.1}$$

This means that $\boldsymbol{z}$ is the relevant part of $\boldsymbol{x}$, and the rest $\boldsymbol{z}_\perp$ does not contain any information on $\boldsymbol{y}$. The problem of finding $\boldsymbol{W}$ is called sufficient dimensionality reduction (Li, 1991; Cook & Ni, 2005).

2.2 Sufficient Dimensionality Reduction with SCE. Let us consider a squared-loss variant of conditional entropy, squared-loss CE (SCE):

$$\mathrm{SCE}(\boldsymbol{Y}|\boldsymbol{Z}) = -\frac{1}{2}\int\int (p(\boldsymbol{y}|\boldsymbol{z}) - 1)^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}. \tag{2.2}$$

By expanding the squared term in equation 2.2, we obtain

$$\begin{aligned}\mathrm{SCE}(\boldsymbol{Y}|\boldsymbol{Z}) &= -\frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} + \int\int p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} \\ &\quad -\frac{1}{2}\int\int p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} \\ &= -\frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} + 1 - \frac{1}{2}\int \mathrm{d}\boldsymbol{y} \\ &= \widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) + 1 - \frac{1}{2}\int \mathrm{d}\boldsymbol{y},\end{aligned} \tag{2.3}$$

where $\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z})$ is defined as

$$\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) = -\frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}. \tag{2.4}$$

Then we have the following theorem (its proof is given in appendix A), which forms the basis of our proposed method:

Theorem 1.

$$\begin{aligned}\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) - \widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{X}) &= \frac{1}{2}\int\int\left(\frac{p(\boldsymbol{z}_\perp, \boldsymbol{y}|\boldsymbol{z})}{p(\boldsymbol{z}_\perp|\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z})} - 1\right)^2 p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} \\ &\geq 0.\end{aligned}$$

This theorem shows $\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) \geq \widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{X})$, and the equality holds if and only if

$$p(\boldsymbol{z}_\perp, \boldsymbol{y}|\boldsymbol{z}) = p(\boldsymbol{z}_\perp|\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z}).$$

This is equivalent to the conditional independence, equation 2.1, and therefore sufficient dimensionality reduction can be performed by minimizing $\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z})$ with respect to $\boldsymbol{W}$:

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}\in\mathbb{G}_{d_z}^{d_x}(\mathbb{R})}{\operatorname{argmin}}\ \widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z} = \boldsymbol{W}\boldsymbol{X}). \tag{2.5}$$

Here, $\mathbb{G}_{d_z}^{d_x}(\mathbb{R})$ denotes the Grassmann manifold, which is a set of orthogonal matrices without overlaps,

$$\mathbb{G}_{d_z}^{d_x}(\mathbb{R}) = \{\boldsymbol{W} \in \mathbb{R}^{d_z \times d_x} \mid \boldsymbol{W}\boldsymbol{W}^\top = \boldsymbol{I}_{d_z}\}/\sim,$$

where $\boldsymbol{I}$ denotes the identity matrix and $\sim$ represents the equivalence relation: $\boldsymbol{W}$ and $\boldsymbol{W}'$ are written as $\boldsymbol{W} \sim \boldsymbol{W}'$ if their rows span the same subspace.

Since $p(\boldsymbol{y}|\boldsymbol{z}) = p(\boldsymbol{z}, \boldsymbol{y})/p(\boldsymbol{z})$, $\mathrm{SCE}(\boldsymbol{Y}|\boldsymbol{Z})$ is equivalent to the negative Pearson divergence (Pearson, 1900) from $p(\boldsymbol{z}, \boldsymbol{y})$ to $p(\boldsymbol{z})$, which is a member of the *f*-divergence class (Ali & Silvey, 1966; Csiszár, 1967) with the squared-loss function. Ordinary conditional entropy (CE), defined by

$$\mathrm{CE}(\boldsymbol{Y}|\boldsymbol{Z}) = -\int\!\!\int p(\boldsymbol{z}, \boldsymbol{y}) \log p(\boldsymbol{y}|\boldsymbol{z}) \mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y},$$

is the negative Kullback-Leibler divergence (Kullback & Leibler, 1951) from $p(\boldsymbol{z}, \boldsymbol{y})$ to $p(\boldsymbol{z})$. Since the Kullback-Leibler divergence is also a member of the *f*-divergence class (with the log-loss function), CE and SCE have similar properties. Indeed, theorem 1 also holds for ordinary CE. However, the Pearson divergence is shown to be more robust against outliers (Basu, Harris, Hjort, & Jones, 1998; Sugiyama, Suzuki, & Kanamori, 2012), since the log function, is very sharp near zero, is not included. Furthermore, as we show, $\widetilde{\mathrm{SCE}}$ can be approximated analytically, and thus its derivative can also be easily computed. This is a critical property for developing a dimensionality-reduction method because we want to minimize $\widetilde{\mathrm{SCE}}$ with respect to $\boldsymbol{W}$, where the gradient is highly useful in devising an optimization algorithm. For this reason, we adopt SCE instead of CE below.

2.3 SCE Approximation. Since $\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z})$ in equation 2.5 is unknown in practice, we approximate it using samples $\{(\boldsymbol{z}_i, \boldsymbol{y}_i)|\boldsymbol{z}_i = \boldsymbol{W}\boldsymbol{x}_i\}_{i=1}^n$.

The trivial inequality $(a-b)^2/2 \geq 0$ yields $a^2/2 \geq ab - b^2/2$, and thus we have

$$\frac{a^2}{2} = \max_b \left[ab - \frac{b^2}{2}\right]. \tag{2.6}$$

If we set $a = p(\boldsymbol{y}|\boldsymbol{z})$, we have

$$\frac{p(\boldsymbol{y}|\boldsymbol{z})^2}{2} \geq \max_b \left[p(\boldsymbol{y}|\boldsymbol{z})b(\boldsymbol{z}, \boldsymbol{y}) - \frac{b(\boldsymbol{z}, \boldsymbol{y})^2}{2}\right].$$

If we multiply both sides of the above inequality with $-p(\boldsymbol{z})$ and integrate over $\boldsymbol{z}$ and $\boldsymbol{y}$, we have

$$\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) \leq \min_{b} \int\int \left[\frac{b(\boldsymbol{z},\boldsymbol{y})^2 p(\boldsymbol{z})}{2} - b(\boldsymbol{z},\boldsymbol{y}) p(\boldsymbol{z},\boldsymbol{y}) \right] \mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}, \tag{2.7}$$

where minimization with respect to b is now performed as a function of $\boldsymbol{z}$ and $\boldsymbol{y}$. (For more general discussions on divergence bounding, see Keziou, 2003, and Nguyen, Wainwright, & Jordan, 2010).

Let us consider a linear-in-parameter model for b:

$$b(\boldsymbol{z},\boldsymbol{y}) = \boldsymbol{\alpha}^\top \boldsymbol{\varphi}(\boldsymbol{z},\boldsymbol{y}),$$

where $\boldsymbol{\alpha}$ is a parameter vector and $\boldsymbol{\varphi}(\boldsymbol{z},\boldsymbol{y})$ is a vector of basis functions. If the expectations over densities $p(\boldsymbol{z})$ and $p(\boldsymbol{z},\boldsymbol{y})$ are approximated by sample averages and the ℓ_2-regularizer $\lambda\boldsymbol{\alpha}^\top\boldsymbol{\alpha}/2$ ($\lambda \geq 0$) is included, the above minimization problem yields

$$\widehat{\boldsymbol{\alpha}} = \operatorname*{argmin}_{\boldsymbol{\alpha}} \left[\frac{1}{2}\boldsymbol{\alpha}^\top \widehat{\boldsymbol{G}}\boldsymbol{\alpha} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\alpha} + \frac{\lambda}{2}\boldsymbol{\alpha}^\top\boldsymbol{\alpha} \right],$$

where

$$\begin{aligned} \widehat{\boldsymbol{G}} &= \frac{1}{n}\sum_{i=1}^{n} \bar{\boldsymbol{\Phi}}(\boldsymbol{z}_i), \\ \widehat{\boldsymbol{h}} &= \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\varphi}(\boldsymbol{z}_i,\boldsymbol{y}_i), \\ \bar{\boldsymbol{\Phi}}(\boldsymbol{z}) &= \int \boldsymbol{\varphi}(\boldsymbol{z},\boldsymbol{y})\boldsymbol{\varphi}(\boldsymbol{z},\boldsymbol{y})^\top \mathrm{d}\boldsymbol{y}. \end{aligned} \tag{2.8}$$

The solution $\widehat{\boldsymbol{\alpha}}$ is analytically given by

$$\widehat{\boldsymbol{\alpha}} = \left(\widehat{\boldsymbol{G}} + \lambda \boldsymbol{I}\right)^{-1} \widehat{\boldsymbol{h}},$$

which yields $\widehat{b}(\boldsymbol{z},\boldsymbol{y}) = \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{\varphi}(\boldsymbol{z},\boldsymbol{y})$. Then, from equation 2.7, we obtain an approximator of $\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z})$ analytically as

$$\widehat{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) = \frac{1}{2}\widehat{\boldsymbol{\alpha}}^\top \widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}} - \widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\alpha}}.$$

We call this method least-squares conditional entropy (LSCE).

2.4 Model Selection by Cross-Validation. The $\widetilde{\mathrm{SCE}}$ approximator depends on the choice of models—i.e., the basis function $\boldsymbol{\varphi}(\boldsymbol{z}, \boldsymbol{y})$ and the regularization parameter λ. Such a model can be objectively selected by cross-validation as follows:

1. The training data set $\mathcal{S} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ is divided into K disjoint subsets $\{\mathcal{S}_j\}_{j=1}^K$ with (approximately) the same size.
2. For each model M in the candidate set,
   a. **For** $j = 1, \ldots, K$,
      i. For model M, the LSCE solution $\widehat{b}^{(M,j)}$ is computed from $\mathcal{S}\backslash\mathcal{S}_j$ (i.e., all samples except $\mathcal{S}_j$).
      ii. Evaluate the upper bound of $\widetilde{\mathrm{SCE}}$ obtained by $\widehat{b}^{(M,j)}$ using the hold-out data $\mathcal{S}_j$:
      $$\mathrm{CV}_j(M) = \frac{1}{2|\mathcal{S}_j|} \sum_{\boldsymbol{z} \in \mathcal{S}_j} \int \widehat{b}^{(M,j)}(\boldsymbol{z}, \boldsymbol{y})^2 \mathrm{d}\boldsymbol{y} - \frac{1}{|\mathcal{S}_j|} \sum_{(\boldsymbol{z},\boldsymbol{y}) \in \mathcal{S}_j} \widehat{b}^{(M,j)}(\boldsymbol{z}, \boldsymbol{y}),$$
      where $|\mathcal{S}_j|$ denotes the cardinality of $\mathcal{S}_j$.
   b. The average score is computed as
   $$\mathrm{CV}(M) = \frac{1}{K} \sum_{j=1}^K \mathrm{CV}_j(M).$$
3. The model that minimizes the average score is chosen:
   $$\widehat{M} = \operatorname*{argmin}_M \mathrm{CV}(M).$$
4. For the chosen model $\widehat{M}$, the LSCE solution $\widehat{b}$ is computed from all samples $\mathcal{S}$, and the approximator $\widehat{SCE}(\boldsymbol{Y}|\boldsymbol{Z})$ is computed.

In the experiments, we use $K = 5$.

2.5 Dimensionality Reduction with SCE. Now we solve the following optimization problem by gradient descent:

$$\operatorname*{argmin}_{\boldsymbol{W} \in \mathbb{G}_{d_{\boldsymbol{z}}}^{d_{\boldsymbol{x}}}(\mathbb{R})} \widehat{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z} = \boldsymbol{W}\boldsymbol{X}). \tag{2.9}$$

As shown in appendix B, the gradient of $\widehat{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z} = \boldsymbol{W}\boldsymbol{X})$ is given by

$$\frac{\partial \widehat{\mathrm{SCE}}}{\partial W_{l,l'}} = \widehat{\boldsymbol{\alpha}}^\top \frac{\partial \widehat{\boldsymbol{G}}}{\partial W_{l,l'}} \left(\frac{3}{2}\widehat{\boldsymbol{\alpha}} - \widehat{\boldsymbol{\beta}} \right) + \frac{\partial \widehat{\boldsymbol{h}}^\top}{\partial W_{l,l'}} (\widehat{\boldsymbol{\beta}} - 2\widehat{\boldsymbol{\alpha}}),$$

where $\widehat{\boldsymbol{\beta}} = \left(\widehat{\boldsymbol{G}} + \lambda \boldsymbol{I}\right)^{-1} \widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}}$.

In the Euclidean space, the above gradient gives the steepest direction. However, on a manifold, the natural gradient (Amari, 1998) gives the steepest direction.

The natural gradient $\nabla\widehat{\mathrm{SCE}}(\boldsymbol{W})$ at $\boldsymbol{W}$ is the projection of the ordinary gradient $\frac{\partial\widehat{\mathrm{SCE}}}{\partial W_{l,l'}}$ to the tangent space of $\mathbb{G}_{d_{\mathbf{z}}}^{d_{\mathbf{x}}}(\mathbb{R})$ at $\boldsymbol{W}$. If the tangent space is equipped with the canonical metric $\langle \boldsymbol{W}, \boldsymbol{W}' \rangle = \frac{1}{2}\mathrm{tr}(\boldsymbol{W}^\top \boldsymbol{W}')$, the natural gradient is given as follows (Edelman, Arias, & Smith, 1998):

$$\nabla\widehat{\mathrm{SCE}} = \frac{\partial\widehat{\mathrm{SCE}}}{\partial \boldsymbol{W}} - \frac{\partial\widehat{\mathrm{SCE}}}{\partial \boldsymbol{W}}\boldsymbol{W}^\top\boldsymbol{W} = \frac{\partial\widehat{\mathrm{SCE}}}{\partial \boldsymbol{W}}\boldsymbol{W}_\perp^\top\boldsymbol{W}_\perp,$$

where $\boldsymbol{W}_\perp$ is a $(d_{\mathbf{x}} - d_{\mathbf{z}}) \times d_{\mathbf{x}}$ matrix such that $[\boldsymbol{W}^\top, \boldsymbol{W}_\perp^\top]$ is an orthogonal matrix.

Then the geodesic from $\boldsymbol{W}$ to the direction of the natural gradient $\nabla\widehat{\mathrm{SCE}}$ over $\mathbb{G}_{d_{\mathbf{z}}}^{d_{\mathbf{x}}}(\mathbb{R})$ can be expressed using $t \in \mathbb{R}$ as

$$\boldsymbol{W}_t = \left[\boldsymbol{I}_{d_{\mathbf{z}}} \;\; \boldsymbol{O}_{d_{\mathbf{z}},(d_{\mathbf{x}}-d_{\mathbf{z}})}\right] \times \exp\left(-t\begin{bmatrix} \boldsymbol{O}_{d_{\mathbf{z}},d_{\mathbf{z}}} & \frac{\partial\widehat{\mathrm{SCE}}}{\partial \boldsymbol{W}}\boldsymbol{W}_\perp^\top \\ -\boldsymbol{W}_\perp\frac{\partial\widehat{\mathrm{SCE}}}{\partial \boldsymbol{W}}^\top & \boldsymbol{O}_{d_{\mathbf{x}}-d_{\mathbf{z}},d_{\mathbf{x}}-d_{\mathbf{z}}} \end{bmatrix}\right)\begin{bmatrix}\boldsymbol{W} \\ \boldsymbol{W}_\perp\end{bmatrix},$$

where "exp" for a matrix denotes the matrix exponential and $\boldsymbol{O}_{d,d'}$ denotes the $d \times d'$ zero matrix. Note that the derivative $\partial_t \boldsymbol{W}_t$ at $t = 0$ coincides with the natural gradient $\nabla\widehat{\mathrm{SCE}}$ (see Edelman et al., 1998, for details). Thus, line search along the geodesic in the natural gradient direction is equivalent to finding the minimizer from $\{\boldsymbol{W}_t | t \geq 0\}$.

Once $\boldsymbol{W}$ is updated, SCE is reestimated with the new $\boldsymbol{W}$, and gradient descent is performed again. This entire procedure is repeated until $\boldsymbol{W}$ converges. When SCE is reestimated, performing cross-validation in every step is computationally expensive. In our implementation, we perform cross-validation only once every five gradient updates. Furthermore, to find a better local optimal solution, this gradient descent procedure is executed 20 times with randomly chosen initial solutions; the one achieving the smallest value of $\widehat{\mathrm{SCE}}$ is chosen.

2.6 Conditional Density Estimation with SCE. Since the maximum of equation 2.6 is attained at $b = a$ and $a = p(\boldsymbol{y}|\boldsymbol{z})$ in the current derivation, the optimal $b(\boldsymbol{z}, \boldsymbol{y})$ is actually the conditional density $p(\boldsymbol{y}|\boldsymbol{z})$ itself. Therefore, $\widehat{\boldsymbol{\alpha}}^\top\boldsymbol{\varphi}(\boldsymbol{z}, \boldsymbol{y})$ obtained by LSCE is a conditional density estimator. This implies that the upper-bound minimization procedure described

in section 2.3 is equivalent to least-squares conditional density estimation (LSCDE) (Sugiyama et al., 2010), which minimizes the squared error:

$$\frac{1}{2}\int\int (b(\boldsymbol{z},\boldsymbol{y}) - p(\boldsymbol{y}|\boldsymbol{z}))^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}.$$

Then, in the same way as the original LSCDE, we may postprocess the solution $\widehat{\boldsymbol{\alpha}}$ to make the conditional density estimator nonnegative and normalized as

$$\widehat{p}(\boldsymbol{y}|\boldsymbol{z}=\widetilde{\boldsymbol{z}}) = \frac{\widetilde{\boldsymbol{\alpha}}^\top \boldsymbol{\varphi}(\widetilde{\boldsymbol{z}},\boldsymbol{y})}{\int \widetilde{\boldsymbol{\alpha}}^\top \boldsymbol{\varphi}(\widetilde{\boldsymbol{z}},\boldsymbol{y}')\mathrm{d}\boldsymbol{y}'}, \tag{2.10}$$

where $\widetilde{\alpha}_l = \max\left(\widehat{\alpha}_l, 0\right)$. Note that even if the solution is postprocessed as equation 2.10, the optimal estimation rate of the LSCDE solution is still maintained (Sugiyama et al., 2010).

2.7 Basis Function Design. In practice, we use the following gaussian function as the kth basis:

$$\varphi_k(\boldsymbol{z},\boldsymbol{y}) = \exp\left(-\frac{\|\boldsymbol{z}-\boldsymbol{u}_k\|^2 + \|\boldsymbol{y}-\boldsymbol{v}_k\|^2}{2\sigma^2}\right), \tag{2.11}$$

where $(\boldsymbol{u}_k, \boldsymbol{v}_k)$ denotes the kth gaussian center located at $(\boldsymbol{z}_k, \boldsymbol{y}_k)$. When the sample size n is too large, we may use only a subset of samples as gaussian centers. σ denotes the gaussian bandwidth, which is chosen by cross-validation, as explained in section 2.4. We may use different bandwidths for $\boldsymbol{z}$ and $\boldsymbol{y}$, but this will increase the computation time for model selection. In our implementation, we normalize each element of $\boldsymbol{z}$ and $\boldsymbol{y}$ to have the unit variance in advance and then use the common bandwidth for $\boldsymbol{z}$ and $\boldsymbol{y}$.

A notable advantage of using the gaussian function is that the integral over $\boldsymbol{y}$ appeared in $\bar{\boldsymbol{\Phi}}(\boldsymbol{z})$ (see equation 2.8) can be computed analytically as

$$\bar{\Phi}_{k,k'}(\boldsymbol{z}) = (\sqrt{\pi}\sigma)^{d_{\mathrm{y}}} \exp\left(-\frac{2\|\boldsymbol{z}-\boldsymbol{u}_k\|^2 + 2\|\boldsymbol{z}-\boldsymbol{u}_{k'}\|^2 + \|\boldsymbol{v}_k-\boldsymbol{v}_{k'}\|^2}{4\sigma^2}\right).$$

Similarly, the normalization term in equation 2.10 can also be computed analytically as

$$\int \widetilde{\boldsymbol{\alpha}}^\top \boldsymbol{\varphi}(\boldsymbol{z},\boldsymbol{y})\mathrm{d}\boldsymbol{y} = (\sqrt{2\pi}\sigma)^{d_{\mathrm{y}}} \sum_k \widetilde{\alpha}_k \exp\left(-\frac{\|\boldsymbol{z}-\boldsymbol{u}_k\|^2}{2\sigma^2}\right).$$

2.8 Discussion. We have proposed minimizing SCE for dimensionality reduction:

$$\mathrm{SCE}(\boldsymbol{Y}|\boldsymbol{Z}) = -\frac{1}{2}\int\int\left(\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})} - 1\right)^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}.$$

In previous work Suzuki and Sugiyama (2013), squared-loss mutual information (SMI) was maximized for dimensionality reduction:

$$\mathrm{SMI}(\boldsymbol{Y},\boldsymbol{Z}) = \frac{1}{2}\int\int\left(\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})p(\boldsymbol{y})} - 1\right)^2 p(\boldsymbol{z})p(\boldsymbol{y})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}.$$

This shows that the essential difference is whether $p(\boldsymbol{y})$ is included in the denominator of the density ratio. Thus, if $p(\boldsymbol{y})$ is uniform, the proposed dimensionality-reduction method using SCE is reduced to the existing method using SMI. However, if $p(\boldsymbol{y})$ is not uniform, the density ratio function $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})p(\boldsymbol{y})}$ included in SMI may be more fluctuated than $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}$ included in SCE. Since a smoother function can be more accurately estimated from a small number of samples in general, the proposed method using SCE is expected to work better than the existing method using SMI. We will experimentally demonstrate this effect in section 3.

Sufficient dimension reduction based on the conditional density $p(\boldsymbol{y}|\boldsymbol{z})$ has also been studied in the statistics literature. The density-minimum average variance estimation (dMAVE) method (Xia, 2007) finds a dimension-reduction subspace using local linear regression for the conditional density in a semi-parametric manner. A similar approach has also been taken in the sliced regression for dimension reduction method (Wang & Xia, 2008), where the cumulative conditional density is used instead of the conditional density. A Bayesian approach to sufficient dimension reduction called the Bayesian dimension reduction (BDR) method (Reich, Bondell, & Li, 2011) has been proposed recently. This method models the conditional density as a gaussian mixture model and obtains a dimension-reduction subspace through sampling from the learned prior distribution of low-dimensional input. These methods have been shown to work well for dimension reduction in real-world data sets, although they are applicable only to univariate output data where $d_{\mathrm{y}} = 1$.

In regression, learning with the squared loss is not robust against outliers (Huber, 1981). However, density estimation (Basu et al., 1998) and density ratio estimation (Sugiyama et al., 2012) under the Pearson divergence are known to be robust against outliers. Thus, in the same sense, the proposed LSCE estimator would also be robust against outliers. We experimentally investigate the robustness in section 3.

3 Experiments

In this section, we experimentally investigate the practical usefulness of the proposed method. We consider the following dimensionality-reduction schemes:

None: No dimensionality reduction is performed.
dMAVE: The density-minimum average variance estimation method where dimension reduction is performed through local linear regression for the conditional density (Xia, 2007).[1]
BDR: The Bayesian dimension-reduction method where the conditional density is modeled by a gaussian mixture model and dimension reduction is performed by sampling from the prior distribution of low-dimensional input (Reich et al., 2011).[2]
LSMI: Dimension reduction is performed by maximizing an SMI approximator called least-squares MI (LSMI) using natural gradients over the Grassmann manifold (Suzuki & Sugiyama, 2013).
LSCE (proposed): Dimension reduction is performed by minimizing the proposed LSCE using natural gradients over the Grassmann manifold.
True (reference): The "true" subspace is used (only for artificial data).

After dimension reduction, we execute the following conditional density estimators:

ϵ-KDE: ϵ-neighbor kernel density estimation, where ϵ is chosen by least-squares cross-validation.
LSCDE: Least-squares conditional density estimation (Sugiyama et al., 2010).

Note that the proposed method, which is the combination of LSCE and LSCDE, does not explicitly require the post-LSCDE step because LSCDE is executed inside LSCE. Since the dMAVE and BDR methods are applicable only to univariate output, they are not included in experiments with multivariate output data.

3.1 Illustration. First, we illustrate the behavior of the plain LSCDE (None/LSCDE) and the proposed method (LSCE/LSCDE). The data sets illustrated in Figure 2 have $d_{\mathbf{x}} = 5$, $d_{\mathbf{y}} = 1$, and $d_{\mathbf{z}} = 1$. The first dimension of input $\boldsymbol{x}$ and output y of the samples is plotted in the graphs, and the

[1] We use the program code provided by the author.
[2] We use the program code available at http://www4.stat.ncsu.edu/~reich/code/BayesSDR.R.

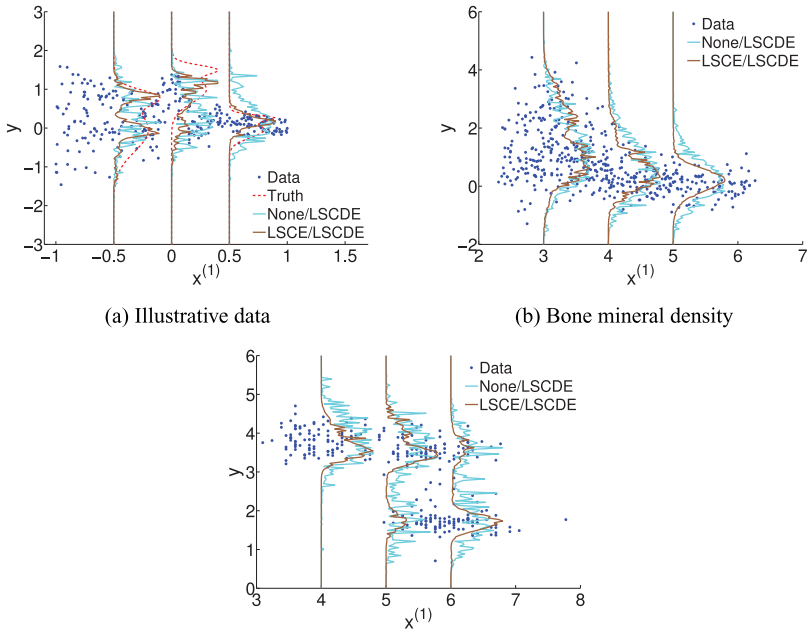

Figure 2: Examples of conditional density estimation by plain LSCDE (None/LSCDE) and the proposed method (LSCE/LSCDE).

other four dimensions of $\boldsymbol{x}$ are just standard normal noise. The results show that the plain LSCDE does not perform well due to the irrelevant noise dimensions of $\boldsymbol{x}$, while the proposed method gives much better estimates.

3.2 Artificial Data Sets. Next, we compare the proposed method with the existing dimensionality-reduction methods on conditional density estimation by LSCDE in artificial data sets.

For $d_{\mathrm{x}} = 5, d_{\mathrm{y}} = 1, \boldsymbol{x} \sim \mathcal{N}(\boldsymbol{x}|0, \boldsymbol{I}_5)$, and $\epsilon \sim \mathcal{N}(\epsilon|0, 0.25^2)$, where $\mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, we consider the following artificial data sets:

a. $d_{\mathrm{z}} = 2$ and $y = (x^{(1)})^2 + (x^{(2)})^2 + \epsilon$.
b. $d_{\mathrm{z}} = 1$ and $y = x^{(2)} + (x^{(2)})^2 + (x^{(2)})^3 + \epsilon$.
c. $d_{\mathrm{z}} = 1$ and $y = \begin{cases} (x^{(1)})^2 + \epsilon & \text{with 0.85 probability,} \\ 2\epsilon - 4 & \text{with 0.15 probability.} \end{cases}$

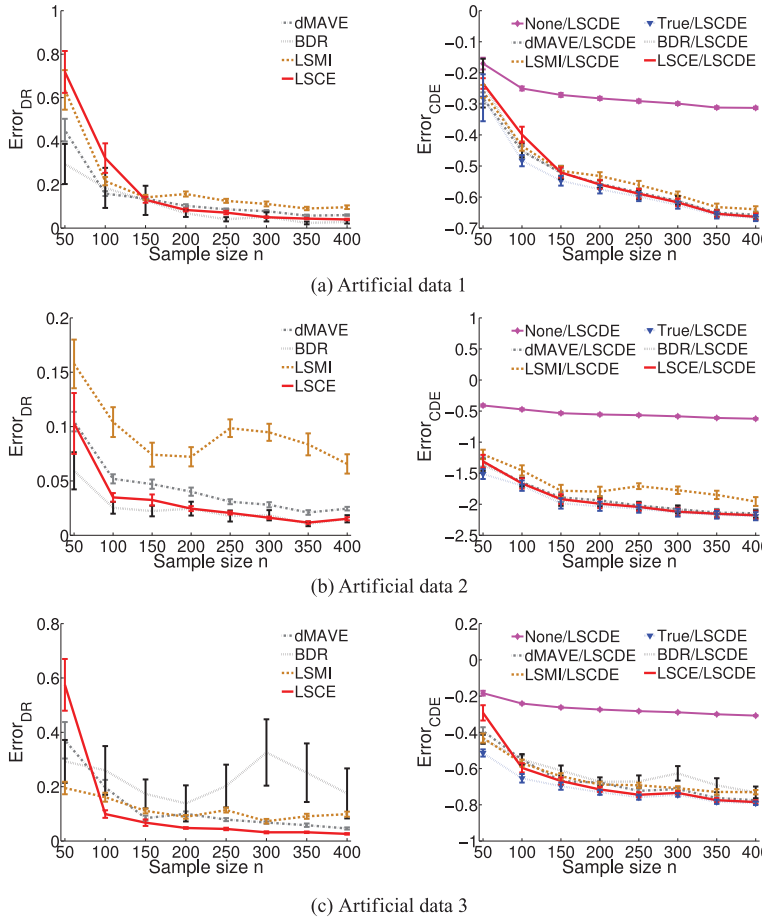

Figure 3: (Left) The mean and standard error of the dimensionality-reduction error over 20 runs. (Right) The mean and standard error of the conditional-density estimation error over 20 runs.

The first row of Figure 3 shows the dimensionality-reduction error between true $\boldsymbol{W}^*$ and its estimate $\widehat{\boldsymbol{W}}$ for different sample size n, measured by

$$\mathrm{Error}_{\mathrm{DR}} = \|\widehat{\boldsymbol{W}}^\top \widehat{\boldsymbol{W}} - \boldsymbol{W}^{*\top} \boldsymbol{W}^*\|_{\mathrm{Frobenius}},$$

where $\|\cdot\|_{\mathrm{Frobenius}}$ denotes the Frobenius norm. All methods perform similarly for data set a, and the dMAVE and BDR methods outperform LSCE and LSMI when $n = 50$.

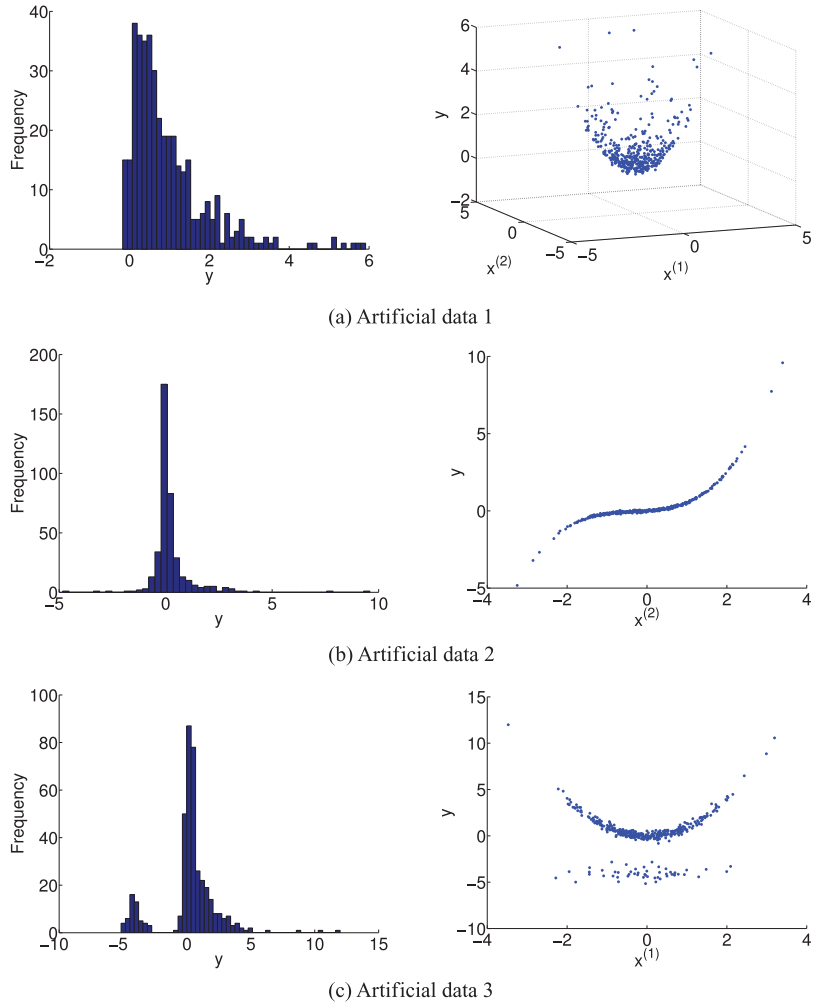

Figure 4: (Left) Example histograms of $\{y_i\}_{i=1}^{400}$ on the artificial data sets. (Right) Example data plot of relevant features of $\boldsymbol{x}$ against y when $n = 400$ on the artificial data sets. The left distribution in the histogram of data set c is regarded as outliers.

In data set b, LSMI does not work well compared to other methods especially when $n \geq 250$. To explain this behavior, we plot the histograms of $\{y\}_{i=1}^{400}$ in the left column of Figure 4. They show that the profile of the histogram (a sample approximation of $p(y)$) in data set b is much sharper than that in data set a. As discussed in section 2.8, the density ratio $\frac{p(\boldsymbol{z},y)}{p(\boldsymbol{z})p(y)}$ used in

LSMI contains $p(\boldsymbol{y})$. Thus, for data set b, the density ratio $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})p(\boldsymbol{y})}$ would be highly nonsmooth and thus is hard to approximate. The conditional density used in other methods is $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}$, where $p(\boldsymbol{y})$ is not included. Therefore, $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}$ would be smoother than $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})p(\boldsymbol{y})}$ and $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}$ is easier to estimate than $\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})p(\boldsymbol{y})}$.

For data set c we consider the situation where $\{y_i\}_{i=1}^n$ contain outliers that are not related to $\boldsymbol{x}$. The data profile of data set c in the right column of Figure 4 illustrates such a situation. The result on data set c shows that the proposed LSCE method is robust against outliers and gives the best subspace estimation accuracy, while the BDR method performs unreliably with large standard errors.

The right column of Figure 3 plots the conditional density estimation error between true $p(\boldsymbol{y}|\boldsymbol{x})$ and its estimate $\widehat{p}(\boldsymbol{y}|\boldsymbol{x})$, evaluated by the squared loss:

$$\mathrm{Error}_{\mathrm{CDE}} = \frac{1}{2n'}\sum_{i=1}^{n'}\int \widehat{p}(\boldsymbol{y}|\widetilde{\boldsymbol{x}}_i)^2 \mathrm{d}\boldsymbol{y} - \frac{1}{n'}\sum_{i=1}^{n'}\widehat{p}(\widetilde{\boldsymbol{y}}_i|\widetilde{\boldsymbol{x}}_i),$$

where $\{(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{y}}_i)\}_{i=1}^{n'}$ is a set of test samples that have not been used for training. We set $n' = 1000$. For data sets a and c, all methods with dimension reduction perform equally well, which is much better than no dimension reduction (None/LSCDE) and is comparable to the method with the true subspace (True/LSCDE). For data set b, all methods except LSMI/LSCDE perform well overall and are comparable to the methods with the true subspace.

3.3 Benchmark Data Sets. Next, we use the UCI benchmark data sets (Bache & Lichman, 2013). We randomly select n samples from each data set for training, and the rest are used to measure the conditional density estimation error in the test phase. Since the dimensionality of the subspace $d_{\boldsymbol{z}}$ is unknown, we chose it by cross-validation. More specifically, five-fold cross-validation is performed for each combination of the dimensionality-reduction and conditional-density estimation methods to choose subspace dimensionalities $d_{\boldsymbol{z}}$ such that the conditional-density estimation error is minimized. Note that tuning parameters λ and σ are also chosen based on cross-validation for each method. Since the conditional-density estimation error is equivalent to SCE, choosing the subspace dimensionalities by the conditional-density estimation error in LSCE is equivalent to choosing subspace dimensionalities that give the minimum SCE value.

The results of univariate output benchmark data sets averaged over 10 runs are summarized in Table 1, showing that LSCDE tends to outperform ϵ-KDE and the proposed LSCE/LSCDE method works well overall. Both LSMI/LSCDE and dMAVE/LSCDE methods also perform well in all data sets, while BDR/LSCDE does not work well in the data sets containing outliers such as Red Wine, White Wine, and Forest Fires. Table 2 describes

Table 1: Mean and Standard Error of the Conditional Density Estimation Error over 10 Runs for Univariate Output Data Sets.

	LSCE		LSMI		dMAVE		BDR		No reduction	
Data Set	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE
Servo	**−2.95(.17)**	**−3.03(.14)**	−2.69(.18)	**−2.95(.11)**	**−3.13(.13)**	**−3.17(.10)**	**−2.96(.10)**	**−2.95(.12)**	−2.62(.09)	−2.72(.06)
Yacht	**−6.46(.02)**	**−6.30(.14)**	−5.63(.26)	−5.47(.29)	−6.25(.06)	−5.97(.12)	**−6.45(.04)**	−6.05(.18)	−1.72(.04)	−2.95(.02)
Auto MPG	−1.80(.04)	−1.75(.05)	−1.85(.04)	−1.77(.05)	**−1.98(.04)**	**−1.97(.04)**	**−1.91(.04)**	−1.84(.05)	−1.75(.04)	−1.46(.04)
Concrete	**−1.37(.03)**	−1.18(.06)	**−1.30(.03)**	−1.18(.04)	**−1.42(.06)**	−1.15(.05)	**−1.37(.04)**	−1.10(.04)	−1.11(.02)	−0.80(.03)
Physicochem	**−1.19(.01)**	−0.99(.02)	**−1.20(.01)**	−0.97(.02)	−1.17(.01)	−0.93(.02)	−1.13(.02)	−0.96(.02)	**−1.19(.01)**	−0.91(.01)
Red Wine	**−2.85(.02)**	−1.95(.17)	**−2.82(.03)**	−1.93(.17)	**−2.82(.02)**	−1.93(.20)	−2.66(.03)	−2.18(.14)	−2.03(.02)	−1.13(.04)
White Wine	**−2.31(.01)**	**−2.47(.15)**	**−2.35(.02)**	**−2.60(.12)**	−2.17(.01)	**−2.65(.20)**	−1.97(.02)	−1.91(.02)	−2.06(.01)	−1.89(.01)
Forest Fires	**−7.18(.02)**	−6.91(.03)	−6.93(.04)	−6.96(.02)	−7.10(.03)	−6.93(.04)	−7.08(.03)	−6.97(.01)	−3.40(.07)	−6.96(.02)
Housing	**−1.72(.09)**	−1.58(.08)	**−1.91(.05)**	−1.62(.08)	**−1.76(.11)**	−1.50(.13)	**−1.86(.09)**	−1.74(.03)	−1.41(.05)	−1.13(.01)

Note: The best methods in terms of the mean error and comparable methods according to the two-sample paired t-test at the significance level 5% are specified in bold.

Table 2: Mean and Standard Error of the Chosen Subspace Dimensionality over 10 Runs for Univariate Output Data Sets.

			LSCE		LSMI		dMAVE		BDR	
Data Set	(d_x, d_y)	n	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE
Servo	(4, 1)	50	1.6(0.27)	2.4(0.40)	2.2(0.33)	1.6(0.31)	1.5(0.22)	1.5(0.31)	1.2(0.13)	2.0(0.37)
Yacht	(6, 1)	80	1.0(0)	1.0(0)	1.0(0)	1.0(0)	1.2(0.13)	1.0(0)	1.0(0)	1.0(0)
Auto MPG	(7, 1)	100	3.2(0.66)	1.3(0.15)	2.1(0.67)	1.1(0.10)	1.5(0.22)	1.0(0)	1.4(0.16)	1.2(0.13)
Concrete	(8, 1)	300	1.0(0)	1.0(0)	1.2(0.13)	1.0(0)	1.7(0.15)	1.0(0)	2.3(0.21)	1.0(0)
Physicochem	(9, 1)	500	6.5(0.58)	1.9(0.28)	6.6(0.58)	2.6(0.86)	7.5(0.48)	5.0(1.33)	2.6(0.16)	1.7(0.26)
Red Wine	(11, 1)	300	1.0(0)	1.3(0.15)	1.2(0.20)	1.0(0)	1.0(0)	1.1(0.10)	1.5(0.22)	1.0(0)
White Wine	(11, 1)	400	1.2(0.13)	1.0(0)	1.4(0.31)	1.0(0)	1.8(0.70)	1.0(0)	3.1(0.23)	2.7(0.30)
Forest Fires	(12, 1)	100	1.2(0.20)	4.4(0.87)	1.4(0.22)	5.6(1.25)	1.5(0.27)	5.2(1.31)	1.2(0.20)	2.8(0.33)
Housing	(13, 1)	100	3.9(0.74)	1.9(0.80)	2.0(0.39)	1.3(0.15)	3.0(0.77)	1.2(0.13)	1.6(0.22)	1.0(0)

the subspace dimensionalities chosen by cross-validation averaged over 10 runs. It shows that all dimensionality-reduction methods reduce the input dimension significantly, especially for Yacht, Red Wine, and White Wine, where the best method always chooses $d_{\mathbf{z}} = 1$ in all runs.

The results of multivariate output Stock and Energy benchmark data sets are summarized in Table 3, showing that the proposed LSCE/LSCDE method also works well for multivariate output data sets and significantly outperforms methods without dimensionality reduction. Table 4 describes the subspace dimensionalities selected by cross-validation, showing that LSMI/LSCDE tends to more aggressively reduce the dimensionality than LSCE/LSCDE.

3.4 Humanoid Robot. We evaluate the performance of the proposed method on humanoid robot transition estimation. We use a simulator of the upper-body part of the humanoid robot *CB-i* (Cheng et al., 2007; see Figure 5). The robot has nine controllable joints: shoulder pitch, shoulder roll, elbow pitch of the right arm, shoulder pitch, shoulder roll, elbow pitch of the left arm, waist yaw, torso roll, and torso pitch joints.

The posture of the robot is described by 18-dimensional real-valued state vector $\boldsymbol{s}$, which corresponds to the angle and angular velocity of each joint in radians and radians per seconds, respectively. We can control the robot by sending the action command $\boldsymbol{a}$ to the system. The action command $\boldsymbol{a}$ is a nine-dimensional real-valued vector that corresponds to the target angle of each joint. When the robot is at state $\boldsymbol{s}$ and receives action $\boldsymbol{a}$, the physical control system of the simulator calculates the amount of torque to be applied to each joint. These torques are calculated by the proportional-derivative (PD) controller as

$$\tau_i = K_{p_i}(a_i - s_i) - K_{d_i}\dot{s}_i,$$

where s_i, $\dot{s}_i$, and a_i denote the current angle, the current angular velocity, and the received target angle of the ith joint, respectively. K_{p_i} and K_{d_i} denote the position and velocity gains for the ith joint, respectively. We set $K_{p_i} = 2000$ and $K_{d_i} = 100$ for all joints except $K_{p_i} = 200$ and $K_{d_i} = 10$ for the elbow pitch joints. After the torques are applied to the joints, the physical control system updates the state of the robot to $\boldsymbol{s}'$.

In the experiment, we randomly choose the action vector $\boldsymbol{a}$ and simulate a noisy control system by adding a bimodal gaussian noise vector. More specifically, the action a_i of the ith joint is first drawn from uniform distribution on $[s_i - 0.087, s_i + 0.087]$. The drawn action is then contaminated by gaussian noise with mean 0 and standard deviation 0.034 with probability 0.6 and gaussian noise with mean -0.087 and standard deviation 0.034 with probability 0.4. By repeatedly controlling the robot n times, we obtain the transition samples $\{(\boldsymbol{s}_j, \boldsymbol{a}_j, \boldsymbol{s}'_j)\}_{j=1}^{n}$. Our goal is to learn the

Table 3: Mean and Standard Error of the Conditional Density Estimation Error over 10 Runs for Multivariate Output Data Sets.

	LSCE		LSMI		No Reduction		
Data Set	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE	Scale
Stock	−8.37(0.53)	**−9.75(0.37)**	**−9.42(0.50)**	**−10.27(0.33)**	−7.35(0.13)	−9.25(0.14)	×1
Energy	**−7.13(0.04)**	−4.18(0.22)	−6.04(0.47)	−3.41(0.49)	−2.12(0.06)	−1.95(0.14)	×10
2 joints	**−10.49(0.86)**	−7.50(0.54)	**−8.00(0.84)**	−7.44(0.60)	−3.95(0.13)	−3.65(0.14)	×1
4 joints	**−2.81(0.21)**	−1.73(0.14)	−2.06(0.25)	−1.38(0.16)	−0.83(0.03)	−0.75(0.01)	×10
9 joints	**−8.37(0.83)**	−2.44(0.17)	**−9.74(0.63)**	−2.37(0.51)	−1.60(0.36)	−0.89(0.02)	×100
Sumi-e 1	**−9.96(1.60)**	−1.49(0.78)	**−6.00(1.28)**	1.24(1.99)	−5.98(0.80)	−0.17(0.44)	×10
Sumi-e 2	**−16.83(1.70)**	−2.22(0.97)	−9.54(1.31)	−3.12(0.75)	−7.69(0.62)	−0.66(0.13)	×10
Sumi-e 3	**−24.92(1.92)**	−6.61(1.25)	−18.0(2.61)	−4.47(0.68)	−8.98(0.66)	−1.45(0.43)	×10

Note: The best methods in terms of the mean error and comparable methods according to the two-sample paired t-test at the significance level 5% are specified in bold.

Table 4: Mean and Standard Error of the Chosen Subspace Dimensionality over 10 Runs for Multivariate Output Data Sets.

			LSCE		LSMI	
Data Set	(d_x, d_y)	n	LSCDE	ϵ-KDE	LSCDE	ϵ-KDE
Stock	(7, 2)	100	3.2(0.83)	2.1(0.59)	2.1(0.60)	2.7(0.67)
Energy	(8, 2)	200	5.9(0.10)	3.9(0.80)	2.1(0.10)	2.0(0.30)
2 joints	(6, 4)	100	2.9(0.31)	2.7(0.21)	2.5(0.31)	2.0(0)
4 joints	(12, 8)	200	5.2(0.68)	6.2(0.63)	5.4(0.67)	4.6(0.43)
9 joints	(27, 18)	500	13.8(1.28)	15.3(0.94)	11.4(0.75)	13.2(1.02)
Sumi-e 1	(9, 6)	200	5.3(0.72)	2.9(0.85)	4.5(0.45)	3.2(0.76)
Sumi-e 2	(9, 6)	250	4.2(0.55)	4.4(0.85)	4.6(0.87)	2.5(0.78)
Sumi-e 3	(9, 6)	300	3.6(0.50)	2.7(0.76)	2.6(0.40)	1.6(0.27)

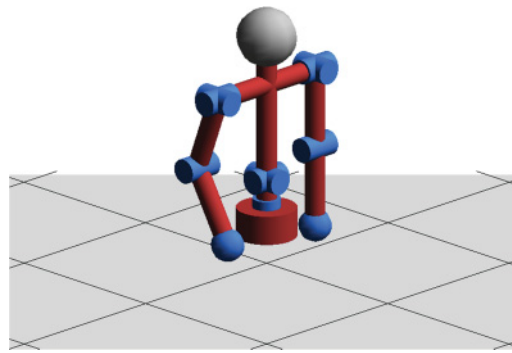

Figure 5: Simulator of the upper-body part of the humanoid robot *CB-i*.

system dynamic as a state transition probability $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$ from these samples. Thus, as the conditional density estimation problem, the state-action pair $(\boldsymbol{s}^\top, \boldsymbol{a}^\top)^\top$ is regarded as input variable $\boldsymbol{x}$, while the next state $\boldsymbol{s}'$ is regarded as output variable $\boldsymbol{y}$. Such state-transition probabilities are highly useful in model-based reinforcement learning (Sutton & Barto, 1998).

We consider three scenarios: using only two joints (right shoulder pitch and right elbow pitch), only four joints (in addition, right shoulder roll and waist yaw), and all nine joints. Thus, $d_x = 6$ and $d_y = 4$ for the two-joint case, $d_x = 12$ and $d_y = 8$ for the four-joint case, and $d_x = 27$ and $d_y = 18$ for the nine-joint case. We generate 500, 1000, and 1500 transition samples for the two-joint, four-joint, and nine-joint cases. We then randomly choose $n = 100$, 200, and 500 samples for training, and use the rest for evaluating the test error. The results are summarized also in Table 3, showing that the proposed method performs well for all three cases. Table 4 describes the

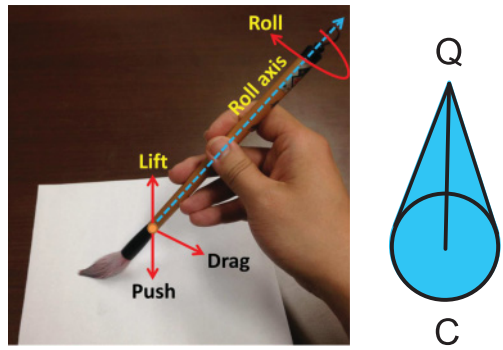

Figure 6: Three actions of the brush, which is modeled as the footprint on a paper canvas.

dimensionalities selected by cross-validation, showing that the humanoid robot's transition is highly redundant.

3.5 Computer Art. Finally, we consider the transition estimation problem in *sumi-e* style brush drawings for nonphotorealistic rendering (Xie, Hachiya, & Sugiyama, 2012). Our aim is to learn the brush dynamics as state transition probability $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$ from the real artists' stroke-drawing samples.

From a video of real brushstrokes, we extract footprints and identify corresponding three-dimensional actions (see Figure 6). The state vector consists of six measurements: the angle of the velocity vector and the heading direction of the footprint relative to the medial axis of the drawing shape, the ratio of the offset distance from the center of the footprint to the nearest point on the medial axis over the radius of the footprint, the relative curvatures of the nearest current point and the next point on the medial axis, and the binary signal of the reverse driving or not. Thus, the state transition probability $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$ has nine-dimensional input and six-dimensional output. We collect 722 transition samples. We randomly choose $n = 200, 250,$ and 300 for training and use the rest for testing.

The estimation results are summarized at the bottom of Tables 3 and 4. These tables show that there exists a low-dimensional sufficient subspace and the proposed method can find it.

4 Conclusion

We proposed a new method for conditional-density estimation in high-dimension problems. The key idea of the proposed method is to perform sufficient dimensionality reduction by minimizing the square-loss conditional entropy (SCE), which can be estimated by least-squares conditional-density

estimation. Thus, dimensionality-reduction and conditional-density estimation are carried out simultaneously in an integrated manner.

We have shown that SCE and the squared-loss mutual information (SMI) are similar but different in that the output density is included in the denominator of the density ratio in SMI. This means that estimation of SMI is hard when the output density is fluctuated, while the proposed method using SCE does not suffer from this problem. The proposed method is also robust against outliers since minimization of the Pearson divergence automatically weighs down the effects of outlier points. Moreover, the proposed method is applicable to multivariate output data, which is not straightforward to handle in other dimensionality-reduction methods based on conditional probability density. The effectiveness of the proposed method was demonstrated through extensive experiments, including humanoid robot transition and computer art.

Appendix A: Proof of Theorem 1

The $\widetilde{\mathrm{SCE}}$ is defined as

$$\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) = -\frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}.$$

Then we have

$$\begin{aligned}\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) - \widetilde{\mathrm{SCE}}(\mathbf{Y}|\mathbf{X}) &= \frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{x})^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{y}\mathrm{d}\boldsymbol{x} \\ &\quad - \frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} \\ &= \frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{x})^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} \\ &\quad + \frac{1}{2}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} \\ &\quad - \int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}.\end{aligned}$$

Let $p(\boldsymbol{x}) = p(\boldsymbol{z}, \boldsymbol{z}_\perp)$, and $\mathrm{d}\boldsymbol{x} = \mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{z}_\perp$. Then the final term can be expressed as

$$\begin{aligned}\int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} &= \int\int \frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}\frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y} \\ &= \int\int \frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})}p(\boldsymbol{z},\boldsymbol{y})\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{y}\end{aligned}$$

$$
\begin{aligned}
&= \int\int \frac{p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z})} p(\boldsymbol{z}_\perp | \boldsymbol{z}, \boldsymbol{y}) p(\boldsymbol{z}, \boldsymbol{y}) \mathrm{d}\boldsymbol{z} \mathrm{d}\boldsymbol{z}_\perp \mathrm{d}\boldsymbol{y} \\
&= \int\int \frac{p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z})} p(\boldsymbol{z}, \boldsymbol{z}_\perp, \boldsymbol{y}) \mathrm{d}\boldsymbol{z} \mathrm{d}\boldsymbol{z}_\perp \mathrm{d}\boldsymbol{y} \\
&= \int\int \frac{p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z})} p(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \\
&= \int\int \frac{p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z})} \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})} p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \\
&= \int\int p(\boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y},
\end{aligned}
$$

where $p(\boldsymbol{z}, \boldsymbol{z}_\perp, \boldsymbol{y}) = p(\boldsymbol{x}, \boldsymbol{y})$, and $\mathrm{d}\boldsymbol{z}\mathrm{d}\boldsymbol{z}_\perp = \mathrm{d}\boldsymbol{x}$ are used. Therefore,

$$
\begin{aligned}
\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) - \widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{X}) &= \frac{1}{2} \int\int p(\boldsymbol{y}|\boldsymbol{x})^2 p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \\
&\quad + \frac{1}{2} \int\int p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{z}) \mathrm{d}\boldsymbol{z} \mathrm{d}\boldsymbol{y} \\
&\quad - \int\int p(\boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \\
&= \frac{1}{2} \int\int (p(\boldsymbol{y}|\boldsymbol{x}) - p(\boldsymbol{y}|\boldsymbol{z}))^2 p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y}.
\end{aligned}
$$

We can also express $p(\boldsymbol{y}|\boldsymbol{x})$ in terms of $p(\boldsymbol{y}|\boldsymbol{z})$ as

$$
\begin{aligned}
p(\boldsymbol{y}|\boldsymbol{x}) &= \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})} \\
&= \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})} \frac{p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z}, \boldsymbol{y})} \\
&= \frac{p(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z}_\perp|\boldsymbol{z}) p(\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{z})} \\
&= \frac{p(\boldsymbol{z}, \boldsymbol{z}_\perp, \boldsymbol{y}) p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z}_\perp|\boldsymbol{z}) p(\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{z})} \\
&= \frac{p(\boldsymbol{z}_\perp, \boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{z}, \boldsymbol{y})}{p(\boldsymbol{z}_\perp|\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{z})} \\
&= \frac{p(\boldsymbol{z}_\perp, \boldsymbol{y}|\boldsymbol{z})}{p(\boldsymbol{z}_\perp|\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{z})} p(\boldsymbol{y}|\boldsymbol{z}).
\end{aligned}
$$

Finally, we obtain

$$\begin{aligned}\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z})-\widetilde{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{X}) &= \frac{1}{2}\int\int \big(p(\boldsymbol{y}|\boldsymbol{x})-p(\boldsymbol{y}|\boldsymbol{z})\big)^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\\ &= \frac{1}{2}\int\int\left(\frac{p(\boldsymbol{z}_\perp,\boldsymbol{y}|\boldsymbol{z})}{p(\boldsymbol{z}_\perp|\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z})}p(\boldsymbol{y}|\boldsymbol{z})-p(\boldsymbol{y}|\boldsymbol{z})\right)^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\\ &= \frac{1}{2}\int\int\left(\frac{p(\boldsymbol{z}_\perp,\boldsymbol{y}|\boldsymbol{z})}{p(\boldsymbol{z}_\perp|\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z})}-1\right)^2 p(\boldsymbol{y}|\boldsymbol{z})^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\\ &\geq 0,\end{aligned}$$

which concludes the proof.

Appendix B: Derivatives of SCE

Here we show the formula of derivatives of $\widehat{\mathrm{SCE}}(Y|Z)$ using the LSCE estimator. SCE approximation by the LSCE estimator is

$$\widehat{\mathrm{SCE}}(\boldsymbol{Y}|\boldsymbol{Z}) = \frac{1}{2}\widehat{\boldsymbol{\alpha}}^\top\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}} - \widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\alpha}}.$$

Taking its partial derivatives with respect to $\boldsymbol{W}$, we obtain

$$\begin{aligned}\frac{\partial\widehat{\mathrm{SCE}}}{\partial W_{l,l'}} &= -\frac{1}{2}\frac{\partial\widehat{\boldsymbol{\alpha}}^\top\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}}}{\partial W_{l,l'}} - \frac{\partial\widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\alpha}}}{\partial W_{l,l'}}\\ &= \frac{1}{2}\left(\frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}} + \frac{(\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}})^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{\alpha}}\right) - \frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{h}} - \frac{\partial\widehat{\boldsymbol{h}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{\alpha}}\\ &= \frac{1}{2}\frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}} + \frac{1}{2}\frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}} + \frac{1}{2}\widehat{\boldsymbol{\alpha}}^\top\frac{\partial\widehat{\boldsymbol{G}}}{\partial W_{l,l'}}\widehat{\boldsymbol{\alpha}} - \frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{h}} - \frac{\partial\widehat{\boldsymbol{h}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{\alpha}}\\ &= \frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\alpha}} + \frac{1}{2}\widehat{\boldsymbol{\alpha}}^\top\frac{\partial\widehat{\boldsymbol{G}}}{\partial W_{l,l'}}\widehat{\boldsymbol{\alpha}} - \frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{h}} - \frac{\partial\widehat{\boldsymbol{h}}^\top}{\partial W_{l,l'}}\widehat{\boldsymbol{\alpha}}. \qquad \text{(A.1)}\end{aligned}$$

Next we consider the partial derivatives of $\widehat{\boldsymbol{\alpha}}$ as follows:

$$\begin{aligned}\frac{\partial\widehat{\boldsymbol{\alpha}}}{\partial W_{l,l'}} &= \frac{\partial(\widehat{\boldsymbol{G}}+\lambda\boldsymbol{I})^{-1}\widehat{\boldsymbol{h}}}{\partial W_{l,l'}}\\ &= \frac{\partial(\widehat{\boldsymbol{G}}+\lambda\boldsymbol{I})^{-1}}{\partial W_{l,l'}}\widehat{\boldsymbol{h}} + (\widehat{\boldsymbol{G}}+\lambda\boldsymbol{I})^{-1}\frac{\partial\widehat{\boldsymbol{h}}}{\partial W_{l,l'}}\\ \frac{\partial\widehat{\boldsymbol{\alpha}}^\top}{\partial W_{l,l'}} &= \left(\frac{\partial(\widehat{\boldsymbol{G}}+\lambda\boldsymbol{I})^{-1}}{\partial W_{l,l'}}\widehat{\boldsymbol{h}}\right)^\top + \frac{\partial\widehat{\boldsymbol{h}}^\top}{\partial W_{l,l'}}(\widehat{\boldsymbol{G}}+\lambda\boldsymbol{I})^{-1}. \qquad \text{(A.2)}\end{aligned}$$

Using $\frac{\partial X^{-1}}{\partial t} = -X^{-1}\frac{\partial X}{\partial t}X^{-1}$, we obtain

$$\begin{aligned}\frac{\partial(\widehat{G}+\lambda I)^{-1}}{\partial W_{l,l'}}\widehat{h} &= -(\widehat{G}+\lambda I)^{-1}\frac{\partial \widehat{G}}{\partial W_{l,l'}}(\widehat{G}+\lambda I)^{-1}\widehat{h} - (\widehat{G}+\lambda I)^{-1} \\ &\quad \times \frac{\partial \lambda I}{\partial W_{l,l'}}(\widehat{G}+\lambda I)^{-1}\widehat{h} \\ &= -(\widehat{G}+\lambda I)^{-1}\frac{\partial \widehat{G}}{\partial W_{l,l'}}\widehat{\alpha} - 0 \\ \left(\frac{\partial(\widehat{G}+\lambda I)^{-1}}{\partial W_{l,l'}}\widehat{h}\right)^{\top} &= -\widehat{\alpha}^{\top}\frac{\partial \widehat{G}}{\partial W_{l,l'}}(\widehat{G}+\lambda I)^{-1}. \end{aligned} \tag{A.3}$$

Substitute equation A.3 into equation A.2 to obtain

$$\frac{\partial \widehat{\alpha}^{\top}}{\partial W_{l,l'}} = -\widehat{\alpha}^{\top}\frac{\partial \widehat{G}}{\partial W_{l,l'}}(\widehat{G}+\lambda I)^{-1} + \frac{\partial \widehat{h}^{\top}}{\partial W_{l,l'}}(\widehat{G}+\lambda I)^{-1}. \tag{A.4}$$

Finally, substituting equation A.4 into equation A.1 and using $(\widehat{G}+\lambda I)^{-1}\widehat{G}\widehat{\alpha} = \widehat{\beta}$, we have

$$\begin{aligned}\frac{\partial \widehat{\mathrm{SCE}}}{\partial W_{l,l'}} &= -\widehat{\alpha}^{\top}\frac{\partial \widehat{G}}{\partial W_{l,l'}}\widehat{\beta} + \frac{\partial \widehat{h}^{\top}}{\partial W_{l,l'}}\widehat{\beta} + \frac{1}{2}\widehat{\alpha}^{\top}\frac{\partial \widehat{G}}{\partial W_{l,l'}}\widehat{\alpha} \\ &\quad + \widehat{\alpha}^{\top}\frac{\partial \widehat{G}}{\partial W_{l,l'}}\widehat{\alpha} - \frac{\partial \widehat{h}^{\top}}{\partial W_{l,l'}}\widehat{\alpha} - \frac{\partial \widehat{h}^{\top}}{\partial W_{l,l'}}\widehat{\alpha} \\ &= \widehat{\alpha}^{\top}\frac{\partial \widehat{G}}{\partial W_{l,l'}}\left(\frac{3}{2}\widehat{\alpha} - \widehat{\beta}\right) + \frac{\partial \widehat{h}^{\top}}{\partial W_{l,l'}}(\widehat{\beta} - 2\widehat{\alpha}), \end{aligned}$$

where the partial derivatives of $\widehat{G}$ and $\widehat{h}$ depend on the choice of basis function.

Here we consider the gaussian basis function described in section 2.4. Their partial derivatives are given by

$$\begin{aligned}\frac{\partial \widehat{G}_{k,k'}}{\partial W_{l,l'}} &= -\frac{1}{\sigma^2 n}\sum_{i=1}^{n}\bar{\Phi}_{k,k'}(z_i)((z_i^{(l)} - u_k^{(l)})(x_i^{(l')} - \tilde{u}_k^{(l')}) \\ &\quad + (z_i^{(l)} - u_{k'}^{(l)})(x_i^{(l')} - \tilde{u}_{k'}^{(l')}))\end{aligned}$$

$$\frac{\partial \widehat{h}_k}{\partial W_{l,l'}} = -\frac{1}{\sigma^2 n} \sum_{i=1}^{n} \varphi_k(\boldsymbol{z}_i, \boldsymbol{y}_i)((\boldsymbol{z}_i^{(l)} - \boldsymbol{u}_k^{(l)})(\boldsymbol{x}_i^{(l')} - \tilde{\boldsymbol{u}}_k^{(l')})).$$

References

Ali, S. M., & Silvey, S. D. (1966). A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society, Series B*, *28*(1), 131–142.

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation*, *10*(2), 251–276.

Bache, K., & Lichman, M. (2013). *UCI machine learning repository*. http://archive.ics.uci.edu/m1

Basu, A., Harris, I. R., Hjort, N. L., & Jones, M. C. (1998). Robust and efficient estimation by minimising a density power divergence. *Biometrika*, *85*(3), 549–559.

Cheng, G., Hyon, S., Morimoto, J., Ude, A., Joshua, G., Colvin, G., . . . Stephen, C. J. (2007). Cb: A humanoid research platform for exploring neuroscience. *Advanced Robotics*, *21*(10), 1097–1114.

Cook, R. D., & Ni, L. (2005). Sufficient dimension reduction via inverse regression. *Journal of the American Statistical Association*, *100*(470), 410–428.

Csiszár, I. (1967). Information-type measures of difference of probability distributions and indirect observation. *Studia Scientiarum Mathematicarum Hungarica*, *2*, 229–318.

Edelman, A., Arias, T. A., & Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, *20*(2), 303–353.

Fukumizu, K., Bach, F. R., & Jordan, M. I. (2009). Kernel dimension reduction in regression. *Annals of Statistics*, *37*(4), 1871–1905.

Huber, P. J. (1981). *Robust statistics*. New York: Wiley.

Keziou, A. (2003). Dual representation of ϕ-divergences and applications. *Comptes rendus mathématique*, *336*(10), 857–862.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, *22*, 79–86.

Li, K. (1991). Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association*, *86*(414), 316–342.

Nguyen, X., Wainwright, M. J., & Jordan, M. I. (2010). Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, *56*(11), 5847–5861.

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine Series 5*, *50*(302), 157–175.

Reich, B. J., Bondell, H. D., & Li, L. (2011). Sufficient dimension reduction via Bayesian mixture modeling. *Biometrics*, *67*(3), 886–895.

Sugiyama, M., Suzuki, T., & Kanamori, T. (2012). Density ratio matching under the Bregman divergence: A unified framework of density ratio estimation. *Annals of the Institute of Statistical Mathematics*, *64*(5), 1009–1044.

Sugiyama, M., Takeuchi, I., Kanamori, T., Suzuki, T., Hachiya, H., & Okanohara, D. (2010). Conditional density estimation via least-squares density ratio estimation. In Y. W. Teh & M. Tiggerington (Eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS2010)* (pp. 781–788). JMLR.

Sutton, R. S., & Barto, G. A. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

Suzuki, T., & Sugiyama, M. (2013). Sufficient dimension reduction via squared-loss mutual information estimation. *Neural Computation*, *3*(25), 725–758.

Wang, H., & Xia, Y. (2008). Sliced regression for dimension reduction. *Journal of the American Statistical Association*, *103*(482), 811–821.

Xia, Y. (2007). A constructive approach to the estimation of dimension reduction directions. *Annals of Statistics*, *35*(6), 2654–2690.

Xie, N., Hachiya, H., & Sugiyama, M. (2012). Artist agent: A reinforcement learning approach to automatic stroke generation in oriental ink painting. In J. Langford & J. Pineau (Eds.), *Proceedings of 29th International Conference on Machine Learning (ICML2012)* (pp. 153–160). Madison, WI: Omnipress.

Received April 28, 2014; accepted July 19, 2014.