Conditional Density Estimation via Least-Squares Density Ratio Estimation

Masashi Sugiyama
Tokyo Institute of Technology & JST

Ichiro Takeuchi
Nagoya Institute of Technology

Taiji Suzuki
The University of Tokyo

Takafumi Kanamori
Nagoya University

Hirotaka Hachiya
Tokyo Institute of Technology

Daisuke Okanohara
The University of Tokyo

Abstract

Estimating the conditional mean of an input-output relation is the goal of regression. However, regression analysis is not sufficiently informative if the conditional distribution has multi-modality, is highly asymmetric, or contains heteroscedastic noise. In such scenarios, estimating the conditional distribution itself would be more useful. In this paper, we propose a novel method of conditional density estimation. Our basic idea is to express the conditional density in terms of the ratio of unconditional densities, and the ratio is directly estimated without going through density estimation. Experiments using benchmark and robot transition datasets illustrate the usefulness of the proposed approach.

1 Introduction

Regression is aimed at estimating the conditional *mean* of output $\boldsymbol{y}$ given input $\boldsymbol{x}$. When the conditional density $p(\boldsymbol{y}|\boldsymbol{x})$ is unimodal and symmetric, regression would be sufficient for analyzing the input-output dependency. However, estimating the conditional mean may not be sufficiently informative, when the conditional distribution possesses multi-modality (e.g., inverse kinematics learning of a robot, see Bishop, 2006) or a highly skewed profile with heteroscedastic noise (e.g., biomedical data analysis, see Hastie et al., 2001). In such cases, it would be more informative to estimate the conditional distribution itself. In this paper, we address the problem of estimating conditional densities when $\boldsymbol{x}$ and $\boldsymbol{y}$ are continuous and multi-dimensional.

The mixture density network (MDN) (Bishop, 2006) models the conditional density by a mixture of parametric densities, where the parameters are estimated by a neural network. MDN was shown to work well, although its training is time-consuming and only a local optimal solution may be obtained due to the non-convexity of neural network learning. Similarly, a mixture of Gaussian processes was explored for estimating the conditional density (Tresp, 2001). The mixture model is trained in a computationally efficient manner by an expectation-maximization algorithm (Dempster et al., 1977). However, since the optimization problem is non-convex, one may only access to a local optimal solution in practice.

The kernel quantile regression (KQR) method (Takeuchi et al., 2006; Li et al., 2007) allows one to predict percentiles of the conditional distribution. This implies that solving KQR for all percentiles gives an estimate of the entire conditional cumulative distribution. KQR is formulated as a convex optimization problem, and therefore a unique global solution can be obtained. Furthermore, the entire solution path with respect to the percentile parameter, which was shown to be piece-wise linear, can be computed efficiently (Takeuchi et al., 2009). However, the range of applications of KQR is limited to one-dimensional output and solution path tracking tends to be numerically rather unstable in practice.

In this paper, we propose a new method of conditional density estimation named *least-squares conditional density estimation* (LS-CDE), which can be applied to multi-dimensional inputs and outputs. The proposed method is based on the fact that the conditional density can be expressed in terms of unconditional densities as $p(\boldsymbol{y}|\boldsymbol{x}) = p(\boldsymbol{x}, \boldsymbol{y})/p(\boldsymbol{x})$. Our key

Appearing in Proceedings of the 13^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 9 of JMLR: W&CP 9. Copyright 2010 by the authors.

idea is that we do not estimate the two densities $p(\boldsymbol{x},\boldsymbol{y})$ and $p(\boldsymbol{x})$ separately, but we *directly* estimate the density ratio $p(\boldsymbol{x},\boldsymbol{y})/p(\boldsymbol{x})$ without going through density estimation. Experiments using benchmark and robot transition datasets show that our method compares favorably with existing methods in terms of the accuracy and computational efficiency.

2 A New Method of Conditional Density Estimation

In this section, we formulate the problem of conditional density estimation and give a new method.

2.1 Conditional Density Estimation via Density Ratio Estimation

Let $\mathcal{D}_{\mathrm{X}}$ $(\subset \mathbb{R}^{d_{\mathrm{X}}})$ and $\mathcal{D}_{\mathrm{Y}}$ $(\subset \mathbb{R}^{d_{\mathrm{Y}}})$ be input and output data domains, where d_{X} and d_{Y} are the dimensionality of the data domains, respectively. Let us consider a joint probability distribution on $\mathcal{D}_{\mathrm{X}} \times \mathcal{D}_{\mathrm{Y}}$ with probability density function $p(\boldsymbol{x},\boldsymbol{y})$, and suppose that we are given n independent and identically distributed (i.i.d.) paired samples of input $\boldsymbol{x}$ and output $\boldsymbol{y}$:

$$\{\boldsymbol{z}_i \mid \boldsymbol{z}_i = (\boldsymbol{x}_i,\boldsymbol{y}_i) \in \mathcal{D}_{\mathrm{X}} \times \mathcal{D}_{\mathrm{Y}}\}_{i=1}^n.$$

The goal is to estimate the conditional density $p(\boldsymbol{y}|\boldsymbol{x})$ from the samples $\{\boldsymbol{z}_i\}_{i=1}^n$. Our primal interest is the case where both variables $\boldsymbol{x}$ and $\boldsymbol{y}$ are multi-dimensional and continuous.

A key idea of our proposed approach is to consider the ratio of two unconditional densities:

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})} := r(\boldsymbol{x},\boldsymbol{y}),$$

where we assume $p(\boldsymbol{x}) > 0$ for all $\boldsymbol{x} \in \mathcal{D}_{\mathrm{X}}$. However, naively estimating the two unconditional densities and taking their ratio can result in large estimation error. In order to avoid this, we propose to estimate the *density ratio function* $r(\boldsymbol{x},\boldsymbol{y})$ directly without going through density estimation of $p(\boldsymbol{x},\boldsymbol{y})$ and $p(\boldsymbol{x})$.

2.2 Linear Density-ratio Model

We model the density ratio function $r(\boldsymbol{x},\boldsymbol{y})$ by the following linear model:

$$\widehat{r}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y}) := \boldsymbol{\alpha}^\top \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}), \tag{1}$$

where $^\top$ denotes the transpose of a matrix or a vector,

$$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_b)^\top$$

are parameters to be learned from samples, and

$$\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) = (\phi_1(\boldsymbol{x},\boldsymbol{y}), \phi_2(\boldsymbol{x},\boldsymbol{y}), \ldots, \phi_b(\boldsymbol{x},\boldsymbol{y}))^\top$$

are basis functions such that $\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) \geq \boldsymbol{0}_b$ for all $(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{D}_{\mathrm{X}} \times \mathcal{D}_{\mathrm{Y}}$. $\boldsymbol{0}_b$ denotes the b-dimensional vector with all zeros. The inequality for vectors is applied in an element-wise manner.

Note that the number b of basis functions is not necessarily a constant; it can depend on the number n of samples. Similarly, the basis functions $\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})$ could be dependent on the samples $\{\boldsymbol{x}_i,\boldsymbol{y}_i\}_{i=1}^n$. This means that *kernel* models (i.e., $b = n$ and $\phi_i(\boldsymbol{x},\boldsymbol{y})$ is a kernel function 'centered' at $(\boldsymbol{x}_i,\boldsymbol{y}_i)$) are also included in the above formulation. We explain how the basis functions $\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})$ are practically chosen in Section 2.5.

2.3 A Least-squares Approach to Conditional Density Estimation

We determine the parameter $\boldsymbol{\alpha}$ in the model $\widehat{r}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y})$ so that the following squared error J_0 is minimized:

$$J_0(\boldsymbol{\alpha}) := \frac{1}{2}\iint \left(\widehat{r}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y}) - r(\boldsymbol{x},\boldsymbol{y})\right)^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}.$$

This can be expressed as

$$\begin{aligned} J_0(\boldsymbol{\alpha}) &= \frac{1}{2}\iint \widehat{r}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y})^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} \\ &\quad - \iint \widehat{r}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y}) r(\boldsymbol{x},\boldsymbol{y}) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} + C \\ &= \frac{1}{2}\iint \left(\boldsymbol{\alpha}^\top \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})\right)^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} \\ &\quad - \iint \boldsymbol{\alpha}^\top \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) p(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} + C, \end{aligned} \tag{2}$$

where $C = \frac{1}{2}\iint r(\boldsymbol{x},\boldsymbol{y})p(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$ is a constant and therefore can be safely ignored. Let us denote the first two terms of Eq.(2) by J:

$$J(\boldsymbol{\alpha}) := J_0(\boldsymbol{\alpha}) - C = \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{H}\boldsymbol{\alpha} - \boldsymbol{h}^\top \boldsymbol{\alpha},$$

where

$$\begin{aligned} &\boldsymbol{H} := \int \overline{\boldsymbol{\Phi}}(\boldsymbol{x}) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \quad \boldsymbol{h} := \iint \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) p(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}, \\ &\overline{\boldsymbol{\Phi}}(\boldsymbol{x}) := \int \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})^\top \mathrm{d}\boldsymbol{y}. \end{aligned} \tag{3}$$

The matrix $\boldsymbol{H}$ and the vector $\boldsymbol{h}$ included in $J(\boldsymbol{\alpha})$ contain the expectations over unknown densities $p(\boldsymbol{x})$ and $p(\boldsymbol{x},\boldsymbol{y})$, so we approximate the expectations by sample averages. Then we have

$$\widehat{J}(\boldsymbol{\alpha}) := \frac{1}{2}\boldsymbol{\alpha}^\top \widehat{\boldsymbol{H}}\boldsymbol{\alpha} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\alpha},$$

where

$$\widehat{\boldsymbol{H}} := \frac{1}{n}\sum_{i=1}^n \overline{\boldsymbol{\Phi}}(\boldsymbol{x}_i), \quad \widehat{\boldsymbol{h}} := \frac{1}{n}\sum_{i=1}^n \boldsymbol{\phi}(\boldsymbol{x}_i,\boldsymbol{y}_i). \tag{4}$$

Note that the integral over $\boldsymbol{y}$ included in $\overline{\boldsymbol{\Phi}}(\boldsymbol{x})$ (see Eq.(3)) can be computed in principle since it does not contain any unknown quantity. As shown in Section 2.5, this integration can be computed analytically in our basis function choice.

Now our optimization criterion is summarized as

$$\widetilde{\boldsymbol{\alpha}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\alpha}\in\mathbb{R}^b} \left[\widehat{J}(\boldsymbol{\alpha}) + \lambda\boldsymbol{\alpha}^\top\boldsymbol{\alpha}\right], \tag{5}$$

where a regularizer $\lambda\boldsymbol{\alpha}^\top\boldsymbol{\alpha}$ ($\lambda > 0$) is included for stabilization purposes. Taking the derivative of the above objective function and equating it to zero, we can see that the solution $\widetilde{\boldsymbol{\alpha}}$ can be obtained just by solving the following system of linear equations. $(\widehat{\boldsymbol{H}}+\lambda\boldsymbol{I}_b)\boldsymbol{\alpha} = \widehat{\boldsymbol{h}}$, where $\boldsymbol{I}_b$ denotes the b-dimensional identity matrix. Thus, the solution $\widetilde{\boldsymbol{\alpha}}$ is given analytically as

$$\widetilde{\boldsymbol{\alpha}} = (\widehat{\boldsymbol{H}} + \lambda\boldsymbol{I}_b)^{-1}\widehat{\boldsymbol{h}}. \tag{6}$$

Since the density ratio function is non-negative by definition, we modify the solution $\widetilde{\boldsymbol{\alpha}}$ as[1]

$$\widehat{\boldsymbol{\alpha}} := \max(\boldsymbol{0}_b, \widetilde{\boldsymbol{\alpha}}), \tag{7}$$

where the 'max' operation for vectors is applied in an element-wise manner. Thanks to this rounding-up processing, the solution $\widehat{\boldsymbol{\alpha}}$ tends to be sparse, which contributes to reducing the computation time in the test phase.

In order to assure that the obtained density-ratio function is a conditional density, we renormalize the solution in the test phase—given a test input point $\widetilde{\boldsymbol{x}}$, our final solution is given as

$$\widehat{p}(\boldsymbol{y}|\boldsymbol{x} = \widetilde{\boldsymbol{x}}) = \frac{\widehat{\boldsymbol{\alpha}}^\top\boldsymbol{\phi}(\widetilde{\boldsymbol{x}}, \boldsymbol{y})}{\int \widehat{\boldsymbol{\alpha}}^\top\boldsymbol{\phi}(\widetilde{\boldsymbol{x}}, \boldsymbol{y})\mathrm{d}\boldsymbol{y}}. \tag{8}$$

We call the above method *Least-Squares Conditional Density Estimation (LS-CDE)*. LS-CDE can be regarded as an application of the direct density-ratio estimation method called the *unconstrained Least-Squares Importance Fitting (uLSIF)* (Kanamori et al., 2009) to the problem of density ratio estimation.

2.4 Convergence Analysis

Here, we briefly show a non-parametric convergence rate of the LS-CDE solution.

Let $\mathcal{G}$ be a general set of functions on $\mathcal{D}_\mathrm{X} \times \mathcal{D}_\mathrm{Y}$. Note that $\mathcal{G}$ corresponds to the span of our model, which could be non-parametric (i.e., an infinite dimensional linear space). For a function g ($\in \mathcal{G}$), let us consider a non-negative function $R(g)$ such that

$$\max\left\{\sup_{\boldsymbol{x}}\left[\int g(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{y}\right],\ \sup_{\boldsymbol{x},\boldsymbol{y}}[g(\boldsymbol{x},\boldsymbol{y})]\right\} \le R(g).$$

Then the problem (5) can be generalized as

$$\widehat{r} := \mathop{\mathrm{argmin}}_{g\in\mathcal{G}}\left[\frac{1}{2n}\sum_{i=1}^{n}\int g(\boldsymbol{x}_i,\boldsymbol{y})^2\mathrm{d}\boldsymbol{y} - \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i,\boldsymbol{y}_i) + \lambda_n R(g)^2\right],$$

where λ_n is the regularization parameter depending on n. We assume that the true density ratio function $r(\boldsymbol{x},\boldsymbol{y})$ is contained in $\mathcal{G}$ and there exists M (> 0) such that $R(r) < M$. We also assume that there exists γ ($0 < \gamma < 2$) such that

$$\mathcal{H}_{[]}(\mathcal{G}_M, \epsilon, L_2(p_\mathrm{x}\times\mu_\mathrm{Y})) = \mathcal{O}\left((M/\epsilon)^\gamma\right),$$

where $\mathcal{G}_M := \{g \in \mathcal{G} \mid R(g) \le M\}$. μ_Y is the Lebesgue measure on $\mathcal{D}_\mathrm{Y}$, $p_\mathrm{x}\times\mu_\mathrm{Y}$ is a product measure of p_x and μ_Y, and $\mathcal{H}_{[]}$ is the *bracketing entropy* of $\mathcal{G}_M$ with respect to the $L_2(p_\mathrm{x}\times\mu_\mathrm{Y})$-norm (van der Vaart & Wellner, 1996).

Under the above assumptions, we have the following theorem (its proof is omitted since it follows essentially the same line as Sugiyama et al., 2008).

Theorem 1 *Under the above setting, if* $\lambda_n \to 0$ *and* $\lambda_n^{-1} = o(n^{2/(2+\gamma)})$*, then*

$$\|\widehat{r} - r\|_2 = \mathcal{O}_p(\lambda_n^{1/2}),$$

where $\|\cdot\|_2$ *denotes the* $L_2(p_\mathrm{x}\times\mu_\mathrm{Y})$*-norm and* $\mathcal{O}_p$ *denotes the asymptotic order in probability.*

Note that the conditions $\lambda_n \to 0$ and $\lambda_n^{-1} = o(n^{2/(2+\gamma)})$ intuitively means that λ_n should converge to zero as n tends to infinity, but the speed of convergence should not be too fast.

2.5 Basis Function Design

It is straightforward to show that cross-validation is available for model selection. A good model may be chosen by cross-validation, given that a family of promising model candidates is prepared. As model candidates, we propose to use a Gaussian kernel model:

[1] A variant of the proposed method would be to include the positivity constraint $\boldsymbol{\alpha} \ge \boldsymbol{0}_n$ directly in Eq.(6). Our preliminary experiments showed that the estimation accuracy of this modified algorithm turned out to be comparable to Eq.(7), while the constrained version was computationally less efficient than Eq.(7) since we need to use a numerical quadratic program solver for computing the solution. For this reason, we only consider Eq.(7) in the rest of this paper.

for $\boldsymbol{z} = (\boldsymbol{x}^\top, \boldsymbol{y}^\top)^\top$,

$$\begin{aligned}\phi_\ell(\boldsymbol{x}, \boldsymbol{y}) &= \exp\left(-\frac{\|\boldsymbol{z} - \boldsymbol{w}_\ell\|^2}{2\sigma^2}\right)\\ &= \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{u}_\ell\|^2}{2\sigma^2}\right)\exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{v}_\ell\|^2}{2\sigma^2}\right),\end{aligned} \tag{9}$$

where $\{\boldsymbol{w}_\ell \mid \boldsymbol{w}_\ell = (\boldsymbol{u}_\ell^\top, \boldsymbol{v}_\ell^\top)^\top\}_{\ell=1}^b$ are center points randomly chosen from $\{\boldsymbol{z}_i \mid \boldsymbol{z}_i = (\boldsymbol{x}_i^\top, \boldsymbol{y}_i^\top)^\top\}_{i=1}^n$. We may use different Gaussian widths for $\boldsymbol{x}$ and $\boldsymbol{y}$. However, for simplicity, we decided to use the common Gaussian width σ for both $\boldsymbol{x}$ and $\boldsymbol{y}$ under the setting where the variance of each element of $\boldsymbol{x}$ and $\boldsymbol{y}$ is normalized to one.

An advantage of the above Gaussian kernel model is that the integrals over $\boldsymbol{y}$ in matrix $\overline{\boldsymbol{\Phi}}$ (see Eq.(3)) and in the normalization factor (see Eq.(8)) can be computed analytically; indeed, a simple calculation yields

$$\begin{aligned}\overline{\Phi}_{\ell,\ell'}(\boldsymbol{x}) &= \int \phi_\ell(\boldsymbol{x}, \boldsymbol{y})\phi_{\ell'}(\boldsymbol{x}, \boldsymbol{y})\mathrm{d}\boldsymbol{y}\\ &= (\sqrt{\pi}\sigma)^{d_{\mathrm{Y}}}\exp\left(-\frac{\xi_{\ell,\ell'}(\boldsymbol{x})}{4\sigma^2}\right),\end{aligned}$$

$$\int \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{\phi}(\widetilde{\boldsymbol{x}}, \boldsymbol{y})\mathrm{d}\boldsymbol{y} = (\sqrt{2\pi}\sigma)^{d_{\mathrm{Y}}}\sum_{\ell=1}^b \widehat{\alpha}_\ell \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{u}_\ell\|^2}{2\sigma^2}\right),$$

where $\xi_{\ell,\ell'}(\boldsymbol{x}) := 2\|\boldsymbol{x} - \boldsymbol{u}_\ell\|^2 + 2\|\boldsymbol{x} - \boldsymbol{u}_{\ell'}\|^2 + \|\boldsymbol{v}_\ell - \boldsymbol{v}_{\ell'}\|^2$.

In the experiments, we fix the number of basis functions to $b = \min(100, n)$, and choose the Gaussian width σ and the regularization parameter λ by CV from $\sigma, \lambda \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10\}$.

3 Discussion

In this section, we discuss the characteristics of existing and proposed methods of conditional density estimation.

3.1 ϵ-neighbor Kernel Density Estimation (ϵ-KDE)

For estimating the conditional density $p(\boldsymbol{y}|\boldsymbol{x})$, ϵ-neighbor kernel density estimation (ϵ-KDE) employs the standard kernel density estimator using a subset of samples, $\{\boldsymbol{y}_i\}_{i \in \mathcal{I}_{\boldsymbol{x},\epsilon}}$ for some threshold ϵ (≥ 0), where $\mathcal{I}_{\boldsymbol{x},\epsilon}$ is the set of sample indices such that $\|\boldsymbol{x}_i - \boldsymbol{x}\| \leq \epsilon$.

In the case of Gaussian kernels, ϵ-KDE is expressed as

$$\widehat{p}(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{|\mathcal{I}_{\boldsymbol{x},\epsilon}|}\sum_{i \in \mathcal{I}_{\boldsymbol{x},\epsilon}} N(\boldsymbol{y}; \boldsymbol{y}_i, \sigma^2 \boldsymbol{I}_{d_{\mathrm{Y}}}),$$

where $N(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the Gaussian density with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. The threshold ϵ and the bandwidth σ may be chosen based on CV. ϵ-KDE is simple and easy to use, but it may not be reliable in high-dimensional problems. Slightly more sophisticated variants have been proposed based on weighted kernel density estimation (Fan et al., 1996; Wolff et al., 1999), but they may still share the same weakness.

3.2 Mixture Density Network (MDN)

The mixture density network (MDN) models the conditional density by a mixture of parametric densities (Bishop, 2006). In the case of Gaussian densities, MDN is expressed as

$$\widehat{p}(\boldsymbol{y}|\boldsymbol{x}) = \sum_{\ell=1}^t \pi_\ell(\boldsymbol{x}) N(\boldsymbol{y}; \boldsymbol{\mu}_\ell(\boldsymbol{x}), \sigma_\ell^2(\boldsymbol{x})\boldsymbol{I}_{d_{\mathrm{Y}}}),$$

where $\pi_\ell(\boldsymbol{x})$ denotes the mixing coefficient such that $\sum_{\ell=1}^t \pi_\ell(\boldsymbol{x}) = 1$ and $0 \leq \pi_\ell(\boldsymbol{x}) \leq 1$ for all $\boldsymbol{x} \in \mathcal{D}_{\mathrm{X}}$. All the parameters $\{\pi_\ell(\boldsymbol{x}), \boldsymbol{\mu}_\ell(\boldsymbol{x}), \sigma_\ell^2(\boldsymbol{x})\}_{\ell=1}^t$ are learned as a function of $\boldsymbol{x}$ by a neural network with regularized maximum likelihood estimation. The number t of Gaussian components, the number of hidden units in the neural network, and the regularization parameter may be chosen based on CV. MDN has been shown to work well, although its training is time-consuming and only a local solution may be obtained due to the non-convexity of neural network learning.

3.3 Kernel Quantile Regression (KQR)

Kernel quantile regression (KQR) allows one to predict the 100τ-percentile of conditional distributions for a given τ $(\in (0, 1))$ when y is one-dimensional (Takeuchi et al., 2006; Li et al., 2007). For the Gaussian kernel model

$$\widehat{f}_\tau(\boldsymbol{x}) = \sum_{i=1}^n \alpha_{i,\tau}\phi_i(\boldsymbol{x}) + b_\tau,$$

where $\phi_i(\boldsymbol{x}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}_i\|^2}{2\sigma^2}\right)$, the parameters $\{\alpha_{i,\tau}\}_{i=1}^n$ and b_τ are learned by

$$\min_{\{\alpha_{i,\tau}\}_{i=1}^n, b_\tau}\left[\sum_{i=1}^n \psi_\tau(y_i - \widehat{f}_\tau(\boldsymbol{x}_i)) + \lambda \sum_{i,j=1}^n \phi_i(\boldsymbol{x}_j)\alpha_{i,\tau}\alpha_{j,\tau}\right],$$

where $\psi_\tau(r)$ denotes the pin-ball loss function defined by

$$\psi_\tau(r) = \begin{cases}(1 - \tau)|r| & (r \leq 0),\\ \tau|r| & (r > 0).\end{cases}$$

Thus, solving KQR for all $\tau \in (0, 1)$ gives an estimate of the entire conditional distribution. The bandwidth σ and the regularization parameter λ may be chosen based on CV.

A notable advantage of KQR is that the solution of KQR is piece-wise linear with respect to τ, so the entire solution path can be computed efficiently (Takeuchi et al., 2009). This implies that the conditional cumulative distribution can be computed efficiently. However, solution path tracking tends to be numerically rather unstable and the range of applications of KQR is limited to one-dimensional output y. Furthermore, some heuristic procedure is needed to convert conditional cumulative distributions into conditional densities, which can cause additional estimation errors.

3.4 Other Methods of Density Ratio Estimation

A naive method for estimating the density ratio $p(\boldsymbol{x}, \boldsymbol{y})/p(\boldsymbol{x})$ is to first approximate the two densities $p(\boldsymbol{x}, \boldsymbol{y})$ and $p(\boldsymbol{x})$ by standard kernel density estimation and then taking the ratio of the estimated densities. We refer to this method as the ratio of kernel density estimators (RKDE). As we will show through experiments in the next section, RKDE does not work well since taking the ratio of estimated quantities significantly magnifies the estimation error.

To overcome the above weakness, we decided to directly estimate the density ratio without going through density estimation under the squared-loss. The *kernel mean matching* method (Huang et al., 2007) and the *logistic regression* based method (Qin, 1998; Cheng & Chu, 2004; Bickel et al., 2007) also allow one to directly estimate a density ratio $q(\boldsymbol{x})/q'(\boldsymbol{x})$. However, the derivation of these methods heavily relies on the fact that the two density functions $q(\boldsymbol{x})$ and $q'(\boldsymbol{x})$ share the same domain, which is not fulfilled in the current setting. For this reason, these methods may not be employed for conditional density estimation.

Other methods of direct density ratio estimation (Sugiyama et al., 2008; Nguyen et al., 2008) employs the *Kullback-Leibler (KL) divergence* as the loss function, instead of the squared-loss. It is possible to use these methods for conditional density estimation in the same way as the proposed method. Indeed, our preliminary experiments showed that a KL-based method was comparable to the squared-loss method in terms of accuracy. However, the KL-based method was computationally less efficient. For this reason, we decided to focus on the squared-loss method.

4 Numerical Experiments

In this section, we investigate the experimental performance of the proposed and existing methods.

4.1 Illustrative Examples

Here we illustrate how the proposed LS-CDE method behaves using toy datasets.

Let $d_{\mathrm{X}} = d_{\mathrm{Y}} = 1$. Inputs $\{x_i\}_{i=1}^n$ were independently drawn from $U(-1, 1)$, where $U(a, b)$ denotes the uniform distribution on (a, b). Outputs $\{y_i\}_{i=1}^n$ were generated by the following heteroscedastic noise model:

$$y_i = \mathrm{sinc}(2\pi x_i) + \frac{1}{8}\exp(1 - x_i) \cdot \varepsilon_i.$$

We tested the following three different distributions for $\{\varepsilon_i\}_{i=1}^n$:

(a) Gaussian: $\varepsilon_i \overset{\text{i.i.d.}}{\sim} N(0, 1)$.

(b) Bimodal: $\varepsilon_i \overset{\text{i.i.d.}}{\sim} \frac{1}{2}N(-1, \frac{4}{9}) + \frac{1}{2}N(1, \frac{4}{9})$.

(c) Skewed: $\varepsilon_i \overset{\text{i.i.d.}}{\sim} \frac{3}{4}N(0, 1) + \frac{1}{4}N(\frac{3}{2}, \frac{1}{9})$.

'$\overset{\text{i.i.d.}}{\sim}$' denotes 'independent and identically distributed' and $N(\mu, \sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2. The number of training samples was set to $n = 200$. The numerical results were depicted in Figure 1, illustrating that LS-CDE well captures heteroscedasticity, bimodality, and asymmetricity.

We have also investigated the experimental performance of LS-CDE using the following real datasets:

(d) Bone Mineral Density dataset: Relative spinal bone mineral density measurements on 485 North American adolescents (Hastie et al., 2001), having a heteroscedastic asymmetric conditional distribution.

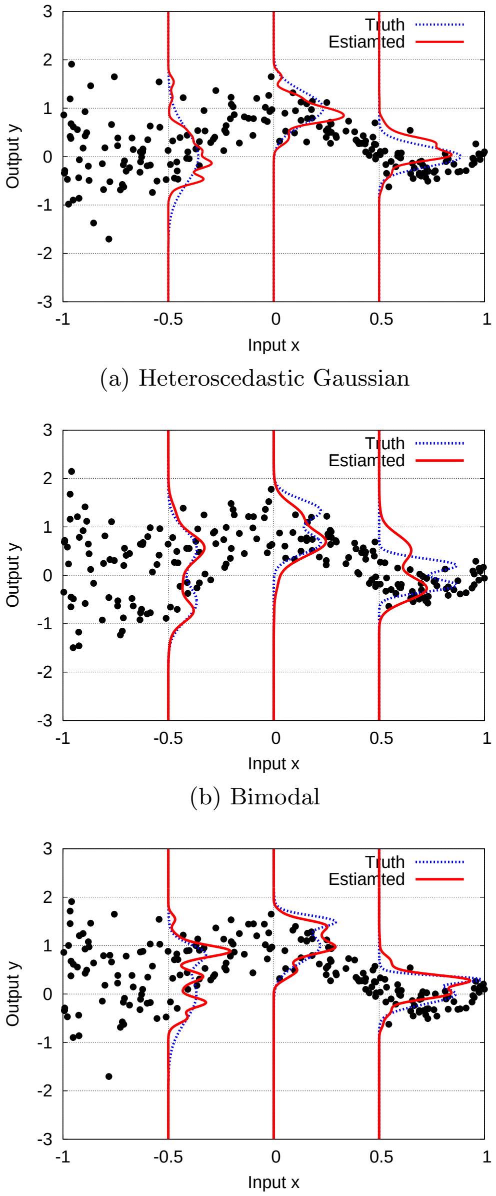
(e) Old Faithful Geyser dataset: The durations of 299 eruptions of the Old Faithful Geyser (Weisberg, 1985), having a bimodal conditional distribution.

Figure 2 depicts the experimental results, showing that heteroscedastic and multi-modal structures were nicely revealed by LS-CDE.

4.2 Benchmark Datasets

We applied the proposed and existing methods to the benchmark datasets accompanied with the *R* package (see Table 1) and evaluate their experimental performance.

In each dataset, 50% of samples were randomly chosen for conditional density estimation and the rest was



(a) Heteroscedastic Gaussian

(b) Bimodal

(c) Skewed

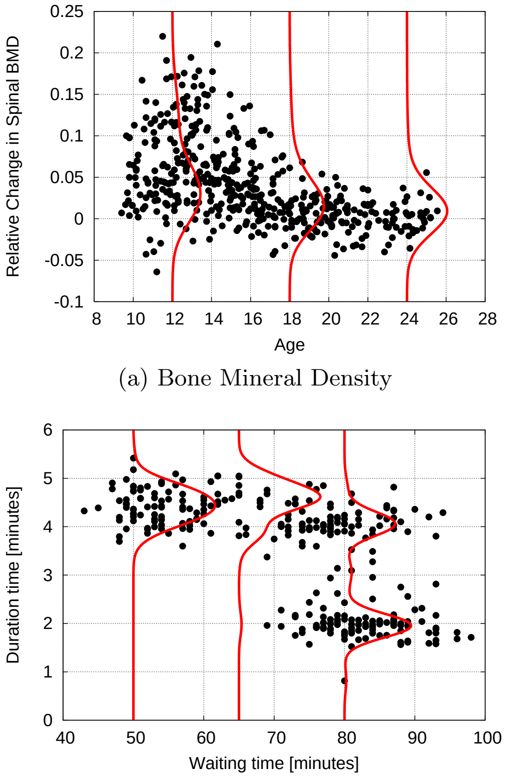
Figure 1: Illustrative examples of LS-CDE for artificial datasets.

used for computing the estimation accuracy. The accuracy of a conditional density estimator $\widehat{p}(\boldsymbol{y}|\boldsymbol{x})$ was measured by the negative log-likelihood for test samples $\{\widetilde{\boldsymbol{z}}_i \mid \widetilde{\boldsymbol{z}}_i = (\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{y}}_i)\}_{i=1}^{\widetilde{n}}$:

$$\mathrm{NLL} := -\frac{1}{\widetilde{n}} \sum_{i=1}^{\widetilde{n}} \log \widehat{p}(\widetilde{\boldsymbol{y}}_i|\widetilde{\boldsymbol{x}}_i). \tag{10}$$

Thus, the smaller the value of NLL is, the better the performance of the conditional density estimator $\widehat{p}(\boldsymbol{y}|\boldsymbol{x})$ is.

We compared LS-CDE, ϵ-KDE, MDN, KQR, and RKDE. For model selection, we used CV based on the log-likelihood. In MDN, CV over three tuning parameters (the number of Gaussian components, the number of hidden units in the neural network, and the regularization parameter) was unbearably slow, so the number of Gaussian components was fixed to $t = 3$ and the other two tuning parameters were chosen by CV.



(a) Bone Mineral Density

(b) Old Faithful Geyser

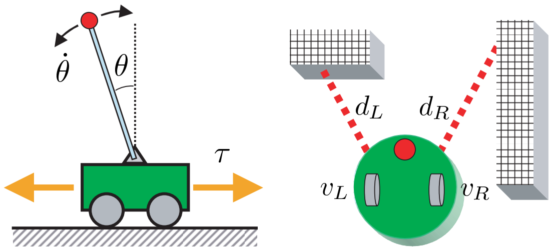
Figure 2: Illustrative examples of LS-CDE for real datasets.

The experimental results are summarized in Table 1. ϵ-KDE was computationally very efficient, but it tended to perform rather poorly. MDN worked well, but it is computationally highly demanding. KQR overall performed well and it was computationally slightly more efficient than LS-CDE. However, its solution path tracking algorithm was numerically rather unstable and we could not obtain solutions for the 'engel' and 'cpus' datasets. RKDE did not perform well for all cases, implying that density ratio estimation via density estimation is not reliable in practice. Overall, the proposed LS-CDE was shown to be a promising method for conditional density estimation in terms of the accuracy and computational efficiency.

Table 1: Experimental results on benchmark datasets ($d_{\mathrm{Y}} = 1$). The average and the standard deviation of NLL (see Eq.(10)) over 10 runs are described (smaller is better). The best method in terms of the mean error and comparable methods according to the two-sided paired *t-test* at the significance level 5% are specified by bold face. Mean computation time is normalized so that LS-CDE is one.

Dataset	(n, d_X)	LS-CDE	ϵ-KDE	MDN	KQR	RKDE
caution	(50,2)	**1.24 ± 0.29**	**1.25 ± 0.19**	**1.39 ± 0.18**	**1.73 ± 0.86**	17.11 ± 0.25
ftcollinssnow	(46,1)	**1.48 ± 0.01**	**1.53 ± 0.05**	**1.48 ± 0.03**	**2.11 ± 0.44**	46.06 ± 0.78
highway	(19,11)	**1.71 ± 0.41**	**2.24 ± 0.64**	7.41 ± 1.22	5.69 ± 1.69	15.30 ± 0.76
heights	(687,1)	**1.29 ± 0.00**	1.33 ± 0.01	**1.30 ± 0.01**	**1.29 ± 0.00**	54.79 ± 0.10
sniffer	(62,4)	**0.69 ± 0.16**	**0.96 ± 0.15**	**0.72 ± 0.09**	**0.68 ± 0.21**	26.80 ± 0.58
snowgeese	(22,2)	**0.95 ± 0.10**	1.35 ± 0.17	**2.49 ± 1.02**	**2.96 ± 1.13**	28.43 ± 1.02
ufc	(117,4)	**1.03 ± 0.01**	1.40 ± 0.02	**1.02 ± 0.06**	**1.02 ± 0.06**	11.10 ± 0.49
birthwt	(94,7)	**1.43 ± 0.01**	1.48 ± 0.01	**1.46 ± 0.01**	1.58 ± 0.05	15.95 ± 0.53
crabs	(100,6)	-0.07 ± 0.11	0.99 ± 0.09	**-0.70 ± 0.35**	**-1.03 ± 0.16**	12.60 ± 0.45
GAGurine	(157,1)	**0.45 ± 0.04**	0.92 ± 0.05	**0.57 ± 0.15**	**0.40 ± 0.08**	53.43 ± 0.27
geyser	(149,1)	**1.03 ± 0.00**	1.11 ± 0.02	1.23 ± 0.05	1.10 ± 0.02	53.49 ± 0.38
gilgais	(182,8)	0.73 ± 0.05	1.35 ± 0.03	**0.10 ± 0.04**	0.45 ± 0.15	10.44 ± 0.50
topo	(26,2)	**0.93 ± 0.02**	1.18 ± 0.09	2.11 ± 0.46	2.88 ± 0.85	10.80 ± 0.35
BostonHousing	(253,13)	0.82 ± 0.05	1.03 ± 0.05	**0.68 ± 0.06**	**0.48 ± 0.10**	17.81 ± 0.25
CobarOre	(19,2)	**1.58 ± 0.06**	**1.65 ± 0.09**	**1.63 ± 0.08**	6.33 ± 1.77	11.42 ± 0.51
engel	(117,1)	**0.69 ± 0.04**	1.27 ± 0.05	**0.71 ± 0.16**	N.A.	52.83 ± 0.16
mcycle	(66,1)	**0.83 ± 0.03**	1.25 ± 0.23	1.12 ± 0.10	**0.72 ± 0.06**	48.35 ± 0.79
BigMac2003	(34,9)	**1.32 ± 0.11**	**1.29 ± 0.14**	**2.64 ± 0.84**	**1.35 ± 0.26**	13.34 ± 0.52
UN3	(62,6)	**1.42 ± 0.12**	1.78 ± 0.14	**1.32 ± 0.08**	**1.22 ± 0.13**	11.43 ± 0.58
cpus	(104,7)	1.04 ± 0.07	1.01 ± 0.10	**-2.14 ± 0.13**	N.A.	15.16 ± 0.72
Time		1	0.004	267	0.755	0.089

4.3 Robot Transition Estimation



(a) Pendulum robot. (b) Khepera robot.

Figure 3: Illustration of robots used for experiments.

We further applied the proposed and existing methods to the problem of robot transition estimation. We used the pendulum robot and the Khepera robot simulators illustrated in Figure 3.

The pendulum robot consists of wheels and a pendulum hinged to the body. The state of the pendulum robot consists of angle θ and angular velocity $\dot{\theta}$ of the pendulum. The amount of torque τ applied to the wheels can be controlled, by which the robot can move left or right and the state of the pendulum is changed to θ' and $\dot{\theta}'$. The task is to estimate $p(\theta', \dot{\theta}'|\theta, \dot{\theta}, \tau)$, the transition probability density from state $(\theta, \dot{\theta})$ to state $(\theta', \dot{\theta}')$ by action τ.

The Khepera robot is equipped with two infra-red sensors and two wheels. The infra-red sensors d_L and d_R measure the distance to the left-front and right-front walls. The speed of left and right wheels v_L and v_R can be controlled separately, by which the robot can move forward/backward and rotate left/right. The task is to estimate $p(d'_L, d'_R|d_L, d_R, v_L, v_R)$, where d'_L and d'_R are the next state.

The state transition of the pendulum robot is highly stochastic due to slip, friction, or measurement errors with strong heteroscedasticity. Sensory inputs of the Khepera robot suffer from occlusions and contain highly heteroscedastic noise, so the transition probability density may possess multi-modality and heteroscedasticity. Thus transition estimation of dynamic robots is a challenging task. Note that transition estimation is highly useful in model-based reinforcement learning.

For both robots, 100 samples were used for conditional density estimation and additional 900 samples were used for computing NLL (see Eq.(10)). The number of Gaussian components was fixed to $t = 3$ in MDN, and all other tuning parameters were chosen by CV based on the log-likelihood. Experimental results are summarized in Table 2, showing that LS-CDE is still useful in this challenging task of robot transition estimation.

Table 2: Experimental results on robot transition estimation. The average and the standard deviation of NLL (see Eq.(10)) over 10 runs are described (smaller is better). The best method in terms of the mean error and comparable methods according to the two-sided paired *t-test* at the significance level 5% are specified by bold face. Mean computation time is normalized so that LS-CDE is one. KQR was not included here since it is applicable only when $d_Y = 1$.

Dataset	(n, d_X, d_Y)	LS-CDE	ϵ-KDE	MDN	RKDE
Pendulum1	(100,3,2)	**1.27 ± 0.05**	2.04 ± 0.10	**1.44 ± 0.67**	11.24 ± 0.32
Pendulum2	(100,3,2)	**1.38 ± 0.05**	2.07 ± 0.10	**1.43 ± 0.58**	11.24 ± 0.32
Khepera1	(100,4,2)	**1.69 ± 0.01**	2.07 ± 0.02	**1.90 ± 0.36**	11.03 ± 0.03
Khepera2	(100,4,2)	**1.86 ± 0.01**	2.10 ± 0.01	**1.92 ± 0.26**	11.09 ± 0.02
Time		1	0.164	1134	0.431

5 Conclusions

We proposed a novel approach to conditional density estimation called LS-CDE. Our basic idea was to directly estimate the ratio of unconditional density functions without going through density estimation. LS-CDE was shown to offer a sparse solution in an analytic form and therefore is computationally efficient. A non-parametric convergence rate of the LS-CDE algorithm was also provided. Experiments on benchmark and robot-transition datasets demonstrated the usefulness of LS-CDE.

Acknowledgments

MS was supported by AOARD, SCAT, and the JST PRESTO program.

References

Bickel, S., Brückner, M., & Scheffer, T. (2007). Discriminative learning for differing training and test distributions. *Proceedings of the 24th International Conference on Machine Learning* (pp. 81–88).

Bishop, C. M. (2006). *Pattern recognition and machine learning.* New York, NY, USA: Springer.

Cheng, K. F., & Chu, C. K. (2004). Semiparametric density estimation under a two-sample density ratio model. *Bernoulli*, *10*, 583–604.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, series B*, *39*, 1–38.

Fan, J., Yao, Q., & Tong, H. (1996). Estimation of conditional densities and sensitivity measures in nonlinear dynamical systems. *Biometrika*, *83*, 189–206.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: Data mining, inference, and prediction.* New York: Springer.

Huang, J., Smola, A., Gretton, A., Borgwardt, K. M., & Schölkopf, B. (2007). Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt and T. Hoffman (Eds.), *Advances in neural information processing systems 19*, 601–608. Cambridge, MA: MIT Press.

Kanamori, T., Hido, S., & Sugiyama, M. (2009). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, *10*, 1391–1445.

Li, Y., Liu, Y., & Zhu, J. (2007). Quantile regression in reproducing kernel Hilbert spaces. *Journal of the American Statistical Association*, *102*, 255–268.

Nguyen, X., Wainwright, M., & Jordan, M. (2008). Estimating divergence functionals and the likelihood ratio by penalized convex risk minimization. In J. C. Platt, D. Koller, Y. Singer and S. Roweis (Eds.), *Advances in neural information processing systems 20*, 1089–1096. Cambridge, MA: MIT Press.

Qin, J. (1998). Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, *85*, 619–639.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., & Kawanabe, M. (2008). Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, *60*, 699–746.

Takeuchi, I., Le, Q. V., Sears, T. D., & Smola, A. J. (2006). Nonparametric quantile estimation. *Journal of Machine Learning Research*, *7*, 1231–1264.

Takeuchi, I., Nomura, K., & Kanamori, T. (2009). Nonparametric conditional density estimation using piecewise-linear solution path of kernel quantile regression. *Neural Computation*, *21*, 533–559.

Tresp, V. (2001). Mixtures of gaussian processes. *Advances in Neural Information Processing Systems 13* (pp. 654–660). MIT Press.

van der Vaart, A. W., & Wellner, J. A. (1996). *Weak convergence and empirical processes with applications to statistics.* New York, NY, USA: Springer.

Weisberg, S. (1985). *Applied linear regression.* New York, NY, USA: John Wiley.

Wolff, R. C. L., Yao, Q., & Hall, P. (1999). Methods for estimating a conditional distribution function. *Journal of the American Statistical Association*, *94*, 154–163.