*IEICE Transactions on Information and Systems,*
*vol.E97-D, no.4, pp.968–971, 2014.*

# Computationally Efficient Estimation of Squared-loss Mutual Information with Multiplicative Kernel Models

Tomoya Sakai
Tokyo Institute of Technology, Japan.
sakai@sg.cs.titech.ac.jp

Masashi Sugiyama
Tokyo Institute of Technology, Japan.
sugi@cs.titech.ac.jp
http://sugiyama-www.cs.titech.ac.jp/˜sugi

## Abstract

*Squared-loss mutual information* (SMI) is a robust measure of the statistical dependence between random variables. The sample-based SMI approximator called *least-squares mutual information* (LSMI) was demonstrated to be useful in performing various machine learning tasks such as dimension reduction, clustering, and causal inference. The original LSMI approximates the pointwise mutual information by using the kernel model, which is a linear combination of kernel basis functions located on *paired* data samples. Although LSMI was proved to achieve the optimal approximation accuracy asymptotically, its approximation capability is limited when the sample size is small due to an insufficient number of kernel basis functions. Increasing the number of kernel basis functions can mitigate this weakness, but a naive implementation of this idea significantly increases the computation costs. In this article, we show that the computational complexity of LSMI with the *multiplicative* kernel model, which locates kernel basis functions on *unpaired* data samples and thus the number of kernel basis functions is the sample size squared, is the same as that for the plain kernel model. We experimentally demonstrate that LSMI with the multiplicative kernel model is more accurate than that with plain kernel models in small sample cases, with only mild increase in computation time.

## Keywords

squared-loss mutual information, least-squares mutual information, density ratio estimation, multiplicative kernel models, independence test.

# 1 Introduction

*Squared-loss mutual information* (SMI) [1] between random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ is defined as the *Pearson divergence* from the joint density $p(\boldsymbol{x}, \boldsymbol{y})$ to the product of marginals $p(\boldsymbol{x})p(\boldsymbol{y})$:

$$\mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) := \frac{1}{2} \iint p(\boldsymbol{x})p(\boldsymbol{y}) \left( \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} - 1 \right)^2 \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}.$$

SMI is always non-negative and takes zero if and only if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are statistically independent. Thus, SMI can be used as a measure of the statistical dependence between $\boldsymbol{X}$ and $\boldsymbol{Y}$.

When SMI is used in practice, the densities $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$ are often unknown, and SMI is approximately computed using paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ drawn independently from density $p(\boldsymbol{x}, \boldsymbol{y})$. A naive way to approximate SMI is to estimate the densities $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$ from the samples and plug the estimated densities into the definition of SMI.

However, this density estimation approach tends to perform poorly due to the division by estimated densities which considerably magnifies the estimation error. To overcome this problem, the SMI approximator called *least-squares mutual information* (LSMI) [1] directly estimates the density ratio $\frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}$ without separately estimating each density. LSMI was shown to possess excellent properties, e.g., it achieves the optimal non-parametric convergence rate, it is numerically stable, its solution can be obtained analytically, and it works well in practice [2]. So far, LSMI has been successfully applied to performing various machine learning tasks such as dimension reduction, clustering, object matching, and causal inference [3].

The original LSMI approximates the density ratio $\frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}$ using the kernel model, which is a linear combination of kernel basis functions located on *paired* data samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$. Although LSMI with the kernel model was proved to achieve the optimal approximation accuracy asymptotically, its approximation capability is limited when the sample size is small because of too few kernel basis functions. A naive way to cope with this problem is to increase the number of basis functions, but this significantly increases the computation time.

In this paper, we propose to use the *multiplicative* kernel model in LSMI, which locates kernel basis functions on *unpaired* data samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_j)\}_{i,j=1}^n$ (see Fig.1). Note that the number of kernel basis functions in the multiplicative kernel model is $n^2$. Our critical theoretical contribution in this paper is that the computational complexity of LSMI with the multiplicative kernel model is proved to be the same order as that with the plain kernel model. Through experiments, we demonstrate that LSMI with the multiplicative kernel model is more accurate than that with the plain kernel model in small sample cases, with only mild increase in computation time.

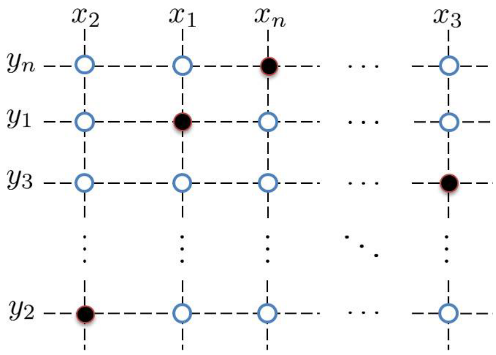


Figure 1: Kernel centers in the plain kernel model and the multiplicative kernel model. The plain kernel model locates $n$ kernels at *paired* samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{n}$ (filled circles), while the multiplicative kernel model locates $n^2$ kernels at *unpaired* samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_j)\}_{i,j=1}^{n}$ (filled and unfilled circles).

## 2 Least-Squares Mutual Information

In this section, we review the sample-based SMI approximator called *least-squares mutual information* (LSMI) [1].

**Basic Idea:** Suppose that we are given a set of paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{n}$ drawn independently from the joint distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$. The key idea of LSMI is to directly estimate the *density ratio* $r(\boldsymbol{x}, \boldsymbol{y}) := \frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}$ without going through density estimation of $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$.

Let $g(\boldsymbol{x}, \boldsymbol{y})$ be a model of the density ratio. We learn the model $g$ so that the following squared-error $J$ is minimized:

$$\begin{aligned} J(g) &:= \frac{1}{2}\iint \Big(g(\boldsymbol{x}, \boldsymbol{y}) - r(\boldsymbol{x}, \boldsymbol{y})\Big)^2 p(\boldsymbol{x})p(\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} \\ &= \frac{1}{2}\iint g(\boldsymbol{x}, \boldsymbol{y})^2 p(\boldsymbol{x})p(\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} \\ &\quad - \iint g(\boldsymbol{x}, \boldsymbol{y})p(\boldsymbol{x}, \boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} + C, \end{aligned}$$

where $C$ is a constant that does not depend on $g$. By approximating the expectations contained in $J$ by the empirical averages, including a regularization functional $R(g)$, and ignoring the irrelevant constant, the LSMI optimization problem is formulated as follows:

$$\widehat{g} := \mathop{\mathrm{argmin}}_{g} \left[\frac{1}{2n^2}\sum_{i,j=1}^{n} g(\boldsymbol{x}_i, \boldsymbol{y}_j)^2 - \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i, \boldsymbol{y}_i) + \lambda R(g)\right],$$

where $\lambda \geq 0$ is the regularization parameter.

Based on another expression of SMI,

$$\begin{aligned}\mathrm{SMI}(\boldsymbol{X},\boldsymbol{Y}) = &-\frac{1}{2}\iint r(\boldsymbol{x},\boldsymbol{y})^2 p(\boldsymbol{x})p(\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\\ &+\iint r(\boldsymbol{x},\boldsymbol{y})p(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}-\frac{1}{2},\end{aligned}$$

the SMI approximator called LSMI is given as follows:

$$\mathrm{LSMI} := -\frac{1}{2n^2}\sum_{i,j=1}^{n}\widehat{g}(\boldsymbol{x}_i,\boldsymbol{y}_j)^2+\frac{1}{n}\sum_{i=1}^{n}\widehat{g}(\boldsymbol{x}_i,\boldsymbol{y}_i)-\frac{1}{2}.$$

**LSMI with Linear Model:** As a density ratio model, let us use the linear-in-parameter model:

$$g(\boldsymbol{x},\boldsymbol{y}) = \sum_{\ell=1}^{b}\theta_\ell\phi_\ell(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{\theta}^\top\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}),$$

where $b$ denotes the number of parameters, $\boldsymbol{\theta} = (\theta_1,\ldots,\theta_b)^\top$ is the parameter vector, and $\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) = (\phi_1(\boldsymbol{x},\boldsymbol{y}),\ldots,\phi_b(\boldsymbol{x},\boldsymbol{y}))^\top$ are the basis function vector.

For the squared regularization functional $R(g) = \boldsymbol{\theta}^\top\boldsymbol{\theta}/2$, the LSMI optimization criterion is expressed as

$$\widehat{\boldsymbol{\theta}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}\in\mathbb{R}^b}\left[\frac{1}{2}\boldsymbol{\theta}^\top\widehat{\boldsymbol{G}}\boldsymbol{\theta}-\boldsymbol{\theta}^\top\widehat{\boldsymbol{h}}+\frac{\lambda}{2}\boldsymbol{\theta}^\top\boldsymbol{\theta}\right],$$

where $\widehat{\boldsymbol{G}}$ and $\widehat{\boldsymbol{h}}$ are defined by

$$\widehat{\boldsymbol{G}} := \frac{1}{n^2}\sum_{i,j=1}^{n}\boldsymbol{\phi}(\boldsymbol{x}_i,\boldsymbol{y}_j)\boldsymbol{\phi}(\boldsymbol{x}_i,\boldsymbol{y}_j)^\top, \quad \widehat{\boldsymbol{h}} := \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{\phi}(\boldsymbol{x}_i,\boldsymbol{y}_i).$$

By taking the derivative of the above objective function with respect to the parameter vector $\boldsymbol{\theta}$, the following system of linear equations is obtained:

$$\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\theta}}+\lambda\widehat{\boldsymbol{\theta}} = \widehat{\boldsymbol{h}}. \tag{1}$$

This linear system can be solved analytically as $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{G}}+\lambda\boldsymbol{I}_b)^{-1}\widehat{\boldsymbol{h}}$, where $\boldsymbol{I}_b$ is the $b$-dimensional identity matrix. Finally, the density ratio estimator $\widehat{g}(\boldsymbol{x},\boldsymbol{y})$ is given by $\widehat{g}(\boldsymbol{x},\boldsymbol{y}) = \widehat{\boldsymbol{\theta}}^\top\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})$, and thus LSMI is expressed as

$$\mathrm{LSMI} = -\frac{1}{2}\widehat{\boldsymbol{\theta}}^\top\widehat{\boldsymbol{G}}\widehat{\boldsymbol{\theta}}+\widehat{\boldsymbol{\theta}}^\top\widehat{\boldsymbol{h}}-\frac{1}{2}.$$

**LSMI with Kernel Models:** As an example of basis functions $\boldsymbol{\phi}$, let us use the *kernel model*:

$$g(\boldsymbol{x},\boldsymbol{y}) := \sum_{i=1}^{n}\theta_i K(\boldsymbol{x},\boldsymbol{x}_i)L(\boldsymbol{y},\boldsymbol{y}_i) = \boldsymbol{\theta}^\top\left[\boldsymbol{k}(\boldsymbol{x})\circ\boldsymbol{l}(\boldsymbol{y})\right],$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ and $L(\boldsymbol{y}, \boldsymbol{y}')$ are kernel functions for $\boldsymbol{x}$ and $\boldsymbol{y}$, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)^\top$ is a parameter vector, $\boldsymbol{k}(\boldsymbol{x}) = (K(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, K(\boldsymbol{x}, \boldsymbol{x}_n))^\top$ and $\boldsymbol{l}(\boldsymbol{y}) = (L(\boldsymbol{y}, \boldsymbol{y}_1), \ldots, L(\boldsymbol{y}, \boldsymbol{y}_n))^\top$ are empirical kernel vectors, and $\circ$ denotes the *Hadamard product*.

For the kernel model, $\widehat{\boldsymbol{G}}$ and $\widehat{\boldsymbol{h}}$ are expressed as

$$\widehat{\boldsymbol{G}} = \frac{1}{n^2}(\boldsymbol{K}^\top \boldsymbol{K}) \circ (\boldsymbol{L}^\top \boldsymbol{L}), \quad \widehat{\boldsymbol{h}} = \frac{1}{n}(\boldsymbol{K} \circ \boldsymbol{L})^\top \boldsymbol{1}_n,$$

where $K_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $L_{i,j} = L(\boldsymbol{y}_i, \boldsymbol{y}_j)$, and $\boldsymbol{1}_n$ is the $n$-dimensional vector with all ones. Thus, the computational complexity for computing LSMI for the kernel model is $\mathcal{O}(n^3)$.

Under some technical conditions, LSMI with the kernel model was proved to achieve the optimal approximation accuracy asymptotically [2]. However, its approximation capability is limited when the sample size is small, partially because the number of kernel basis functions is too small. This drawback may be overcome by increasing the number of basis functions, but this in turn significantly increases the computation time.

## 3 LSMI with Multiplicative Kernel Models

In this section, we propose to use the multiplicative kernel model in LSMI, which locates kernel basis functions at *unpaired* data samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_j)\}_{i,j=1}^n$. As illustrated in Fig.1, the multiplicative kernel model contains $n^2$ kernel basis functions. This allows us to utilize the Kronecker structure to significantly reduce the computational cost.

The multiplicative kernel model is expressed as

$$\begin{aligned} g(\boldsymbol{x}, \boldsymbol{y}) &:= \sum_{i,j=1}^{n} \theta_{i,j} K(\boldsymbol{x}, \boldsymbol{x}_i) L(\boldsymbol{y}, \boldsymbol{y}_j) \\ &= \mathrm{vec}\,(\boldsymbol{\Theta})^\top \left[(\boldsymbol{1}_n \otimes \boldsymbol{k}(\boldsymbol{x})) \circ (\boldsymbol{l}(\boldsymbol{y}) \otimes \boldsymbol{1}_n)\right], \end{aligned}$$

where $\boldsymbol{\Theta}$ is the $n \times n$ parameter matrix with $\Theta_{i,j} = \theta_{i,j}$, $\mathrm{vec}\,(\cdot)$ denotes the vectorization of a matrix, and $\otimes$ denotes the *Kronecker product*.

For the above multiplicative kernel model, $\widehat{\boldsymbol{G}}$ and $\widehat{\boldsymbol{h}}$ are expressed as

$$\widehat{\boldsymbol{G}} = \widetilde{\boldsymbol{L}} \otimes \widetilde{\boldsymbol{K}}, \quad \widehat{\boldsymbol{h}} = \mathrm{vec}(\widetilde{\boldsymbol{H}}),$$

where $\widetilde{\boldsymbol{L}} = \frac{1}{n}\boldsymbol{L}^\top \boldsymbol{L}$, $\widetilde{\boldsymbol{K}} = \frac{1}{n}\boldsymbol{K}^\top \boldsymbol{K}$, and $\widetilde{\boldsymbol{H}} = \frac{1}{n}\boldsymbol{K}^\top \boldsymbol{L}$. The Kronecker structure of $\widehat{\boldsymbol{G}}$ is brought by the fact that kernel basis functions share the same centers in the multiplicative kernel model. Then, Eq.(1) yields that the solution $\widehat{\boldsymbol{\Theta}}$ satisfies

$$\widetilde{\boldsymbol{K}} \widehat{\boldsymbol{\Theta}} \widetilde{\boldsymbol{L}} + \lambda \widehat{\boldsymbol{\Theta}} = \widetilde{\boldsymbol{H}}.$$

This is called the *discrete Sylvester equation*, and can be solved with computational complexity $\mathcal{O}(n^3)$ [4]. Finally, LSMI with the multiplicative kernel model is expressed as

$$\mathrm{LSMI} = -\frac{1}{2}\mathrm{tr}(\widehat{\boldsymbol{\Theta}}^\top \widetilde{\boldsymbol{K}} \widehat{\boldsymbol{\Theta}} \widetilde{\boldsymbol{L}}) + \mathrm{tr}(\widehat{\boldsymbol{\Theta}}^\top \widetilde{\boldsymbol{H}}) - \frac{1}{2}. \tag{2}$$

The computational complexity for calculating Eq.(2) is $\mathcal{O}(n^3)$, and therefore the overall computational complexity of LSMI with the multiplicative kernel model is the same as that with the plain kernel model, even though the number of kernel basis functions is increased from $n$ to $n^2$.

# 4 Experiments

In this section, we experimentally evaluate the performance of LSMI with the plain kernel model and the multiplicative kernel model.

For regression, we use the Gaussian kernel with the common bandwidth for $K(\boldsymbol{x}, \boldsymbol{x}')$ and $L(\boldsymbol{y}, \boldsymbol{y}')$ after element-wise standardization of $\boldsymbol{x}$ and $\boldsymbol{y}$. For classification, we use the delta kernel for $L(\boldsymbol{y}, \boldsymbol{y}')$. The Gaussian width and regularization parameter are determined by 5-fold cross-validation.

**Numerical Illustration:** First, we use the following toy datasets with one-dimensional $x$ and $y$:

**(A) Dependent:** $x$ and $y$ are dependent as

$$p(x, y) = \tfrac{1}{2} N\left(\boldsymbol{z}; \mathbf{1}_2, \boldsymbol{I}_2\right) + \tfrac{1}{2} N\left(\boldsymbol{z}; -\mathbf{1}_2, \boldsymbol{I}_2\right),$$

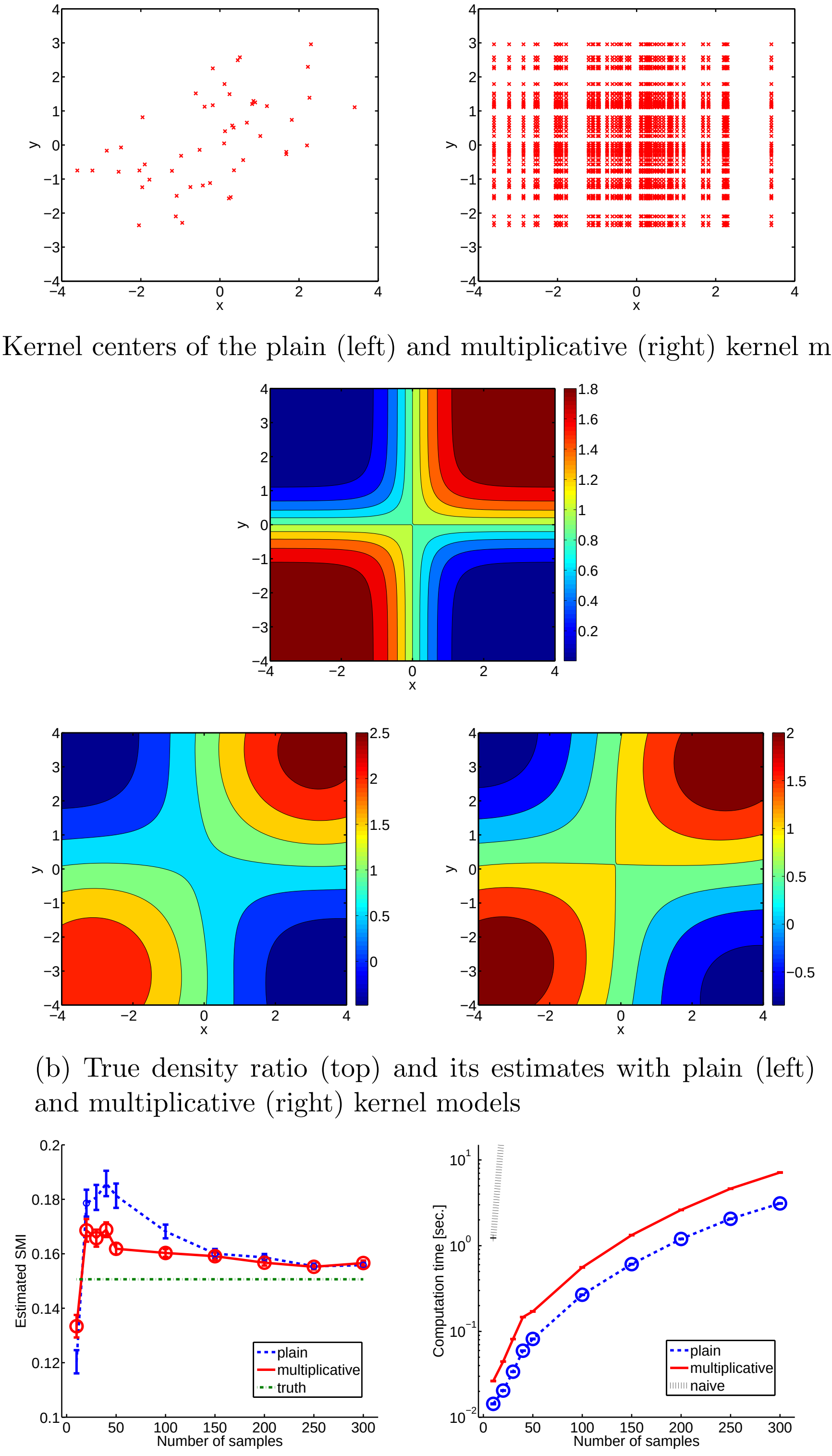
where $\boldsymbol{z} = (x, y)^\top$ and $\mathbf{1}_2 = (1, 1)^\top$. $N(\boldsymbol{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the multi-dimensional normal density with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

**(B) Independent:** $x$ and $y$ are independent as $p(x, y) = 1/4$ if $-1 < x, y < 1$ and zero otherwise.

Fig.2(a) depicts kernel centers of the plain kernel model and the multiplicative kernel model for 50 samples in the dependent case (A). Fig.2(b) depicts the true density-ratio function $\frac{p(x,y)}{p(x)p(y)}$ and its estimates with the plain kernel model and the multiplicative kernel model, respectively, for 50 samples. The graphs show that the function obtained with the multiplicative kernel model approximates the true density-ratio function better than that obtained with the plain kernel model at around the origin.

More qualitatively, Figs.2(c) and 2(d) show the mean and standard error of the LSMI values and the computation time, respectively, over 1000 runs. 'naive' denotes the naive implementation of the multiplicative kernel model (i.e., solving the system of $n^2$ linear equations). The graphs show that LSMI with the multiplicative kernel model is more accurate than that with the plain kernel model. In terms of the computation time, the efficient implementation of LSMI with the multiplicative kernel model is shown to be much faster than its naive implementation and is only slightly slower than LSMI with the plain kernel model. Therefore, given a certain approximation level, LSMI with the multiplicative kernel model is computationally more efficient than that with the plain kernel model.

The results in the independent case (B) are plotted in Fig.3, again showing that LSMI with the multiplicative kernel model is more accurate than that with the plain kernel



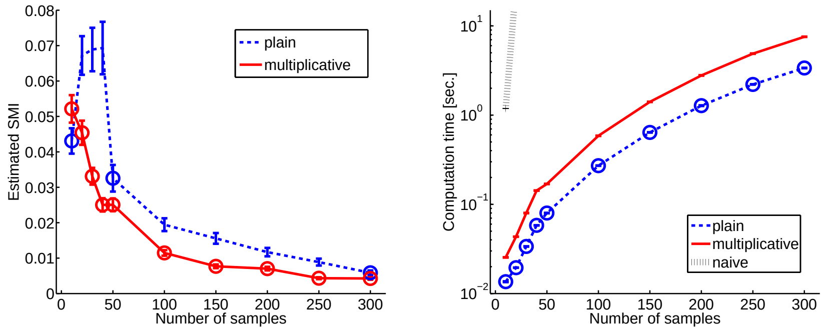
(a) Kernel centers of the plain (left) and multiplicative (right) kernel models

(b) True density ratio (top) and its estimates with plain (left) and multiplicative (right) kernel models

(c) SMI values

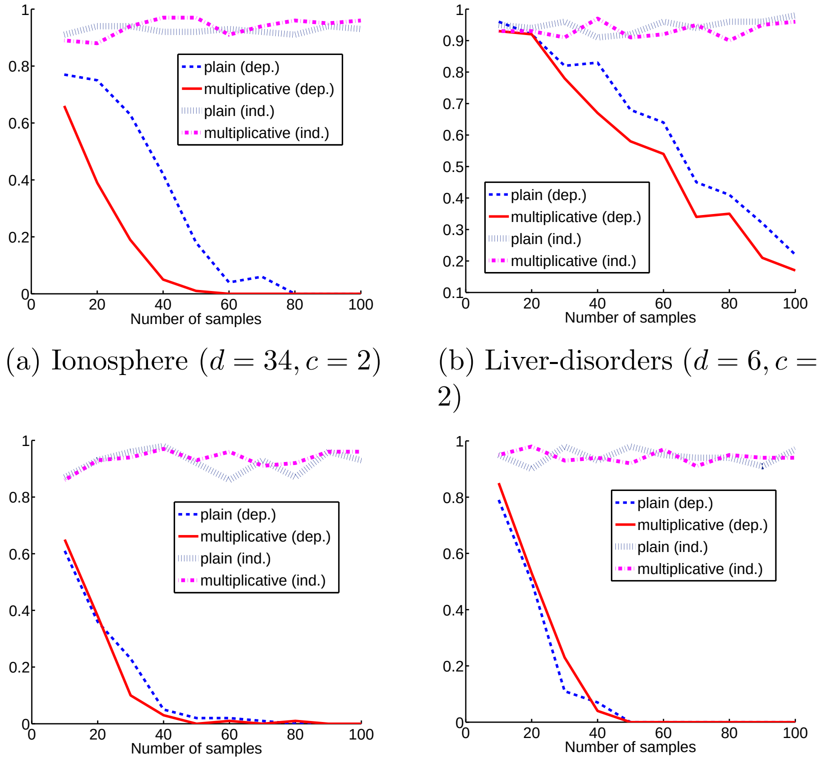
(d) Computation time

Figure 2: Experimental results for the dependent dataset. The best method and comparable ones according to the t-test at the significance level 1% are specified by '○'.



(a) SMI values (b) Computation time

Figure 3: Experimental results for the independent dataset. The best method and comparable ones according to the t-test at the significance level 1% are specified by '○'.



(a) Ionosphere ($d = 34, c = 2$) (b) Liver-disorders ($d = 6, c = 2$)

(c) Shuttle ($d = 9, c = 7$) (d) Vehicle ($d = 18, c = 4$)

Figure 4: Experimental results for the benchmark datasets. Frequency of accepting the null hypothesis over 100 runs under the significance level 0.05 is depicted. $d$ and $c$ denote the input dimensionality and the number of classes of the dataset, respectively.

model. Similarly, the efficient implementation of LSMI with the multiplicative kernel model is much faster than its naive implementation and is only slightly slower than LSMI with the plain kernel model.

**Benchmark Datasets:** Finally, we apply LSMI to independence testing in the framework of the *permutation test* [5].

We employ 4 real-world classification datasets taken from the *UCI repository* available from `http://archive.ics.uci.edu/ml/`. We use the original dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ (where $\boldsymbol{x}$ and $y$ are dependent) to evaluate the *type-II error* (i.e., whether a statistical test can reject the wrong null hypothesis that $\boldsymbol{x}$ and $y$ are independent). We also use its randomly shuffled dataset $\{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ (where $\boldsymbol{x}$ and $y$ are independent) for evaluating the *type-I error* (i.e., whether a statistical test can accept the correct null hypothesis that $\boldsymbol{x}$ and $y$ are independent).

Fig.4 shows the type-I and type-II errors for 100 runs under the significance level 0.05. The graphs show that the multiplicative kernel model tends to provide lower type-II errors than the plain kernel model, while their type-I errors are comparable.

# 5 Conclusions

In this paper, we proposed to use the multiplicative kernel model for approximating squared-loss mutual information. The key contribution of the proposal is that, even though the number of parameters is squared, its computational complexity does not exceed that of the original method with the plain kernel model. Through numerical experiments, we showed that the proposed method achieves lower type-II errors and comparable type-I errors in independence testing.

This work was supported by KAKENHI 23120004 and AOARD.

# References

[1] T. Suzuki, M. Sugiyama, T. Kanamori, and J. Sese, "Mutual information estimation reveals global associations between stimuli and biological processes," BMC Bioinformatics, vol.10, pp.S52:1–S52:12, 2009.

[2] M. Sugiyama, T. Suzuki, and T. Kanamori, Density Ratio Estimation in Machine Learning, Cambridge University Press, Cambridge, UK, 2012.

[3] M. Sugiyama, "Machine learning with squared-loss mutual information," Entropy, vol.15, pp.80–112, 2013.

[4] V. Sima, Algorithms for Linear-Quadratic Optimization, Marcel Dekker, New York, NY, USA, 1996.

[5] T. Suzuki and M. Sugiyama, "Least-squares independence test," IEICE Transactions on Information and Systems, vol.E94-D, pp.1333–1336, 2011.