A Least-squares Approach to Mutual Information Estimation with Application in Variable Selection

Taiji Suzuki[1], Masashi Sugiyama[2], Jun Sese[3], and Takafumi Kanamori[4]

[1] Department of Mathematical Informatics, The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
s-taiji@stat.t.u-tokyo.ac.jp

[2] Department of Computer Science, Tokyo Institute of Technology,
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan
sugi@cs.titech.ac.jp

[3] Department of Information Science, Ochanomizu University,
2-1-1 Ohtsuka, Bunkyo-ku, Tokyo 112-8610, Japan
sesejun@is.ocha.ac.jp

[4] Department of Computer Science and Mathematical Informatics,
Nagoya University, Furocho, Chikusaku, Nagoya 464-8603, Japan
kanamori@is.nagoya-u.ac.jp

Abstract. We propose a new method of estimating mutual information from samples. Our method, called Least-Squares Mutual Information (LSMI), has several attractive properties, e.g., density estimation is not involved, an analytic-form solution is available, a variant of cross-validation can be used for model selection, and an approximate leave-one-out error can be computed very efficiently. Numerical experiments show that LSMI compares favorably with existing methods in mutual information estimation and variable selection. The practical usefulness of LSMI is demonstrated also in protein subcellular localization prediction.

Key words: Mutual information, Feature selection, Importance estimation, Least squares cross validation.

1 Introduction

Detecting underlying dependencies between random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ is highly useful in various machine learning problems such as feature selection [1, 2], independent component analysis [3], and RNA structure prediction [4]. Although classical correlation analysis would be still useful in these problems, it cannot detect non-linear dependencies with no correlation. On the other hand, *mutual information* (MI), which plays an important role in information theory [5], allows us to detect general nonlinear dependencies. MI is defined by

$$I(X,Y) := \iint p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y}) \log\left(\frac{p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y})}{p_{\mathrm{x}}(\boldsymbol{x})p_{\mathrm{y}}(\boldsymbol{y})}\right) d\boldsymbol{x}d\boldsymbol{y}, \tag{1}$$

and it vanishes if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent. For this reason, estimating MI from samples has gathered a lot of attention for many years.

A naive approach to estimating MI is to use a *kernel density estimator* (KDE) [6, 7], i.e., the densities $p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y})$, $p_{\mathrm{x}}(\boldsymbol{x})$, and $p_{\mathrm{y}}(\boldsymbol{y})$ are separately estimated from samples and the estimated densities are used for computing MI. The band-width of the kernel functions could be optimized based on likelihood cross-validation (LCV) [8], so there remains no open tuning parameter in this approach. However, density estimation is known to be a hard problem and therefore the KDE-based method may not be so effective in practice.

An alternative method involves estimation of the *entropies* using k-nearest neighbor (KNN) samples [9]. The KNN-based approach was shown to perform better than KDE [10], given that the number k is chosen appropriately—a small (large) k yields an estimator with small (large) bias and large (small) variance. However, appropriately determining the value of k is not straightforward in the context of MI estimation.

In this paper, we propose a new MI estimator that can overcome the limitations of the existing approaches. Our method, which we call Least-Squares Mutual Information (LSMI), does not involve density estimation and directly models the *density ratio*:

$$w(\boldsymbol{x},\boldsymbol{y}) := \frac{p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y})}{p_{\mathrm{x}}(\boldsymbol{x})p_{\mathrm{y}}(\boldsymbol{y})}. \tag{2}$$

The solution of LSMI can be computed by simply solving a system of linear equations. Therefore, LSMI is computationally very efficient. Furthermore, a variant of cross-validation (CV) is available for model selection, so the values of tuning parameters such as the regularization parameter and the kernel width can be adaptively determined in an objective manner. We also show that an approximated leave-one-out CV (LOOCV) score can be computed very efficiently without going through the hold-out loop. Numerical experiments show that LSMI compares favorably with existing methods in MI estimation and variable selection.

2 A New MI Estimator

In this section, we formulate the MI inference problem as density ratio estimation and propose a new method of estimating the density ratio.

2.1 MI Inference 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 the data domains and suppose we are given n independent and identically distributed (i.i.d.) paired samples $\{(\boldsymbol{x}_i,\boldsymbol{y}_i) \mid \boldsymbol{x}_i \in \mathcal{D}_{\mathrm{X}},\ \boldsymbol{y}_i \in \mathcal{D}_{\mathrm{Y}}\}_{i=1}^{n}$ drawn from a joint distribution with density $p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y})$. Let us denote the marginal densities of $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ by $p_{\mathrm{x}}(\boldsymbol{x})$ and $p_{\mathrm{y}}(\boldsymbol{y})$, respectively. The goal is to estimate MI defined by Eq.(1).

Our key constraint is that we want to avoid density estimation when estimating MI. To this end, we estimate the *density ratio* $w(\boldsymbol{x},\boldsymbol{y})$ defined by Eq.(2) (see also [11–13]). Given a density ratio estimator $\widehat{w}(\boldsymbol{x},\boldsymbol{y})$, MI can be simply estimated by

$$\widehat{I}(X,Y) = \frac{1}{n}\sum_{i=1}^{n}\log \widehat{w}(\boldsymbol{x}_i,\boldsymbol{y}_i).$$

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

$$\widehat{w}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y}) := \boldsymbol{\alpha}^\top \boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y}), \tag{3}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_b)^\top$ are parameters to be learned from samples, $^\top$ denotes the transpose of a matrix or a vector, and

$$\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y}) = (\varphi_1(\boldsymbol{x},\boldsymbol{y}), \varphi_2(\boldsymbol{x},\boldsymbol{y}), \ldots, \varphi_b(\boldsymbol{x},\boldsymbol{y}))^\top$$

are basis functions such that $\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y}) \geq \mathbf{0}_b$ for all $(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{D}_{\mathrm{X}} \times \mathcal{D}_{\mathrm{Y}}$. $\mathbf{0}_b$ denotes the b-dimensional vector with all zeros. Note that $\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y})$ could be dependent on the samples $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^n$, i.e., *kernel* models are also allowed. We explain how the basis functions $\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y})$ are chosen in Section 2.4.

2.2 A Least-squares Approach to Direct Density Ratio Estimation

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

$$\begin{aligned} J_0(\boldsymbol{\alpha}) :=& \frac{1}{2}\iint \left(\widehat{w}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y}) - w(\boldsymbol{x},\boldsymbol{y})\right)^2 p_{\mathrm{x}}(\boldsymbol{x}) p_{\mathrm{y}}(\boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y} \\ =& \frac{1}{2}\iint \widehat{w}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y})^2 p_{\mathrm{x}}(\boldsymbol{x}) p_{\mathrm{y}}(\boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y} - \iint \widehat{w}_{\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{y}) p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y} + C, \end{aligned}$$

where $C = \frac{1}{2}\iint w(\boldsymbol{x},\boldsymbol{y}) p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y}$ is a constant and therefore can be safely ignored. Let us denote the first two terms 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 $\boldsymbol{H} := \iint \boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y}) \boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y})^\top p_{\mathrm{x}}(\boldsymbol{x}) p_{\mathrm{y}}(\boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y}$, $\boldsymbol{h} := \iint \boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{y}) p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y}$.

Approximating the expectations in $\boldsymbol{H}$ and $\boldsymbol{h}$ by empirical averages, we obtain the following optimization problem:

$$\widetilde{\boldsymbol{\alpha}} := \underset{\boldsymbol{\alpha} \in \mathbb{R}^b}{\operatorname{argmin}} \left[\frac{1}{2}\boldsymbol{\alpha}^\top \widehat{\boldsymbol{H}} \boldsymbol{\alpha} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\alpha} + \lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha} \right], \tag{4}$$

where we included a regularization term $\lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha}$ and

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

Differentiating the objective function (4) with respect to $\boldsymbol{\alpha}$ and equating it to zero, we can obtain an analytic-form solution:

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

where $\boldsymbol{I}_b$ is the b-dimensional identity matrix.

Since the importance function $w(\boldsymbol{x}, \boldsymbol{y})$ is non-negative by definition, we modify the solution as

$$\widehat{\boldsymbol{\alpha}} := \max(\boldsymbol{0}_b, \widetilde{\boldsymbol{\alpha}}). \tag{5}$$

We call the above method *Least-Squares Mutual Information (LSMI)* .

Thanks to the analytic-form solution of $\widetilde{\boldsymbol{\alpha}}$, the LSMI solution $\widehat{\boldsymbol{\alpha}}$ can be computed very efficiently. Furthermore, the solution $\widehat{\boldsymbol{\alpha}}$ tends to be sparse since we rounded up negative elements to zero. This contributes to reducing the computation time in the test phase.

2.3 Convergence Bound

Here, we show a non-parametric convergence rate of the solution of the optimization problem (4). Let $\mathcal{G}$ be a general set of functions on $\mathcal{D}_{\mathrm{X}} \times \mathcal{D}_{\mathrm{Y}}$. For a function g $(\in \mathcal{G})$, let us consider a non-negative function $I(g)$ such that

$$\sup_{\boldsymbol{x},\boldsymbol{y}}[g(\boldsymbol{x}, \boldsymbol{y})] \leq I(g). \tag{6}$$

Then the problem (4) can be generalized as

$$\widehat{w} := \mathop{\mathrm{argmin}}_{g \in \mathcal{G}} \left[\frac{1}{2n^2} \sum_{i,j=1}^{n} g_{i,j}^2 - \frac{1}{n} \sum_{i=1}^{n} g_{i,i} + \lambda_n I(g)^2 \right],$$

where $g_{i,j} := g(\boldsymbol{x}_i, \boldsymbol{y}_j)$. We assume that the true density ratio function $w(\boldsymbol{x}, \boldsymbol{y})$ is contained in model $\mathcal{G}$ and satisfies

$$w(\boldsymbol{x}, \boldsymbol{y}) < M_0 \quad \text{for all} \quad (\boldsymbol{x}, \boldsymbol{y}) \in D_X \times D_Y.$$

We also assume that there exists γ $(0 < \gamma < 2)$ such that $\mathcal{H}_{[]}(\mathcal{G}_M, \epsilon, L_2(p_X p_Y)) = O((M/\epsilon)^{\gamma})$, where $\mathcal{G}_M := \{g \in \mathcal{G} \mid I(g) \leq M\}$ and $\mathcal{H}_{[]}$ is the *bracketing entropy* of $\mathcal{G}_M$ with respect to the $L_2(p_{\mathrm{x}} p_{\mathrm{y}})$-norm [14, 15]. This means the function class $\mathcal{G}$ is not too much complex. Then we have the following theorem.

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

$$\|\widehat{w} - w\|_2 = \mathcal{O}_p(\lambda_n^{1/2}), \tag{7}$$

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

Thus by choosing λ_n appropriately, the estimator $\widehat{w}$ converges to w with rate a little bit slower than $\mathcal{O}_p(n^{-1/(2+\gamma)})$ (for example $\mathcal{O}_p((n/\log n)^{-1/(2+\gamma)})$).

2.4 CV for Model Selection and Basis Function Design

The performance of LSMI depends on the choice of the model, i.e., the basis functions $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{y})$ and the regularization parameter λ. Here we show that model selection can be carried out based on a variant of CV.

First, the samples $\{\boldsymbol{z}_i \mid \boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ are divided into R disjoint subsets $\{\mathcal{Z}_r\}_{r=1}^R$. Then a density ratio estimator $\widehat{w}_r(\boldsymbol{x}, \boldsymbol{y})$ is obtained using $\{\mathcal{Z}_j\}_{j\neq r}$ and the cost J is approximated using the held-out samples $\mathcal{Z}_r$ as

$$\widehat{J}_r^{(R-\mathrm{CV})} = \sum_{\boldsymbol{x}',\boldsymbol{y}'\in\mathcal{Z}_r} \frac{\widehat{w}_r(\boldsymbol{x}', \boldsymbol{y}')^2}{2n_r^2} - \sum_{(\boldsymbol{x}',\boldsymbol{y}')\in\mathcal{Z}_r} \frac{\widehat{w}_r(\boldsymbol{x}', \boldsymbol{y}')}{n_r},$$

where n_r is the number of pairs in the set $\mathcal{Z}_r$. $\sum_{\boldsymbol{x}',\boldsymbol{y}'\in\mathcal{Z}_r}$ is the summation over all combinations of $\boldsymbol{x}'$ and $\boldsymbol{y}'$ (i.e., n_r^2 terms), while $\sum_{(\boldsymbol{x}',\boldsymbol{y}')\in\mathcal{Z}_r}$ is the summation over all pairs $(\boldsymbol{x}', \boldsymbol{y}')$ (i.e., n_r terms). This procedure is repeated for $r = 1, 2, \ldots, R$ and its average $\widehat{J}^{(R-\mathrm{CV})}$ is used as an estimate of J:

$$\widehat{J}^{(R-\mathrm{CV})} = \frac{1}{R}\sum_{r=1}^{R} \widehat{J}_r^{(R-\mathrm{CV})}.$$

We can show that $\widehat{J}^{(R-\mathrm{CV})}$ is an almost unbiased estimate of the true cost J, where the 'almost'-ness comes from the fact that the number of samples is reduced in the CV procedure due to data splitting [16].

A good model may be chosen by CV, given that a family of promising model candidates is prepared. As model candidates, we propose using a Gaussian kernel model: for $\boldsymbol{z} = (\boldsymbol{x}^\top, \boldsymbol{y}^\top)^\top$,

$$\varphi_\ell(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(-\frac{\|\boldsymbol{z} - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{u}_\ell\|^2}{2\sigma^2}\right)\exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{v}_\ell\|^2}{2\sigma^2}\right),$$

where $\{\boldsymbol{c}_\ell \mid \boldsymbol{c}_\ell = (\boldsymbol{u}_\ell^\top, \boldsymbol{v}_\ell^\top)^\top\}_{\ell=1}^b$ are center points randomly chosen from $\{\boldsymbol{z}_i \mid \boldsymbol{z}_i = (\boldsymbol{x}_i^\top, \boldsymbol{y}_i^\top)^\top\}_{i=1}^n$.

In the experiments, we fix the number of basis functions at $b = \min(200, n)$, and choose the Gaussian width σ and the regularization parameter λ by CV with grid search.

2.5 Efficient Approximation of LOOCV

When $R = n$, the above CV is called *leave-one-out CV* (LOOCV). Thus in the i-th each iteration, samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_j)\}_{j=1}^n$ and $\{(\boldsymbol{x}_j, \boldsymbol{y}_i)\}_{j=1}^n$ are held out from the product of empirical marginal distributions and $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ is removed from the empirical joint distribution.

To approximate the LOOCV score, let us consider an estimator $\widehat{\boldsymbol{\alpha}}^{(i,j)}$ obtained by removing the sample $(\boldsymbol{x}_i, \boldsymbol{y}_j)$ from the product of empirical marginal distributions and removing the sample $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ from the empirical joint distribution: $\widehat{\boldsymbol{\alpha}}^{(i,j)} := \max(\boldsymbol{0}_b, \widetilde{\boldsymbol{\alpha}}^{(i,j)})$, where $\widetilde{\boldsymbol{\alpha}}^{(i,j)} := \left(\frac{n^2\widehat{\boldsymbol{H}} - \boldsymbol{\varphi}_{i,j}\boldsymbol{\varphi}_{i,j}^\top}{n^2-1} + \lambda\boldsymbol{I}_b\right)^{-1}\frac{n\widehat{\boldsymbol{h}} - \boldsymbol{\varphi}_{i,i}}{n-1}$. Then the LOOCV score for this setting can be expressed as

$$\widehat{J}^{(\mathrm{LOOCV})} = \frac{1}{n^2}\sum_{i,j=1}^{n}\left[\frac{1}{2}\left(\boldsymbol{\varphi}_{i,j}^\top\widehat{\boldsymbol{\alpha}}^{(i,j)}\right)^2 - \boldsymbol{\varphi}_{i,i}^\top\widehat{\boldsymbol{\alpha}}^{(i,j)}\right].$$

Apply the well-known Woodbury formula to $\widetilde{\boldsymbol{\alpha}}^{(i,j)}$, we have another expression:

$$\widetilde{\boldsymbol{\alpha}}^{(i,j)} = \frac{n+1}{n}\left(\boldsymbol{a} + \frac{\boldsymbol{\varphi}_{i,j}^{\top}\boldsymbol{a}}{n^2 - \boldsymbol{a}_{i,j}^{\top}\boldsymbol{\varphi}_{i,j}}\boldsymbol{a}_{i,j}\right) - \frac{n+1}{n^2}\left(\boldsymbol{a}_{i,i} + \frac{\boldsymbol{\varphi}_{i,j}^{\top}\boldsymbol{a}_{i,i}}{n^2 - \boldsymbol{a}_{i,j}^{\top}\boldsymbol{\varphi}_{i,j}}\boldsymbol{a}_{i,j}\right),$$

where $\boldsymbol{a} := \boldsymbol{A}^{-1}\widehat{\boldsymbol{h}},\ \boldsymbol{a}_{i,j} := \boldsymbol{A}^{-1}\boldsymbol{\varphi}_{i,j},\ \boldsymbol{A} := \widehat{\boldsymbol{H}} + \frac{\lambda(n^2-1)}{n^2}\boldsymbol{I}_b$. This implies that an inverse operation of a matrix appears only at $\boldsymbol{A}^{-1}$, which is common to all i and j. Thus, when computing the approximate LOOCV score of LSMI, we do not have to go through the hold-out loop, but we only need to invert the matrix $\boldsymbol{A}$ once.

If we want to save the computation time, the n^2 iterations in $\widehat{J}^{(\mathrm{LOOCV})}$ may be reduced. Note that this does not affect the almost unbiasedness, though the variance would be slightly increased. In our experiments, we only compute $5n$ iterations when computing $\widehat{J}^{(\mathrm{LOOCV})}$.

3 Relation to Existing Methods

In this section, we discuss the characteristics of existing and proposed approaches.

3.1 Kernel Density Estimator (KDE)

KDE is a non-parametric technique to estimate a probability density function $p(\boldsymbol{x})$ from its i.i.d. samples $\{\boldsymbol{x}_i\}_{i=1}^n$. For the Gaussian kernel, KDE is expressed as $\widehat{p}(\boldsymbol{x}) = \frac{1}{n(2\pi\sigma^2)^{d/2}}\sum_{i=1}^n \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}\right)$. The performance of KDE depends on the choice of the kernel width σ and it can be optimized by *likelihood CV* as follows [8]: First, divide the samples $\{\boldsymbol{x}_i\}_{i=1}^n$ into R disjoint subsets $\{\mathcal{X}_r\}_{r=1}^R$. Then obtain a density estimate $\widehat{p}_{\mathcal{X}_k}(\boldsymbol{x})$ from $\{\mathcal{X}_r\}_{r\neq k}$ and compute its hold-out log-likelihood for $\mathcal{X}_k$: $\frac{1}{|\mathcal{X}_k|}\sum_{\boldsymbol{x}\in\mathcal{X}_k}\log\widehat{p}_{\mathcal{X}_k}(\boldsymbol{x})$. This procedure is repeated for $r = 1, 2, \ldots, R$ and choose the value of σ such that the average of the hold-out log-likelihood over all r is maximized. Note that the average hold-out log-likelihood is an almost unbiased estimate of the Kullback-Leibler divergence from $p(\boldsymbol{x})$ to $\widehat{p}(\boldsymbol{x})$, up to an irrelevant constant.

Based on KDE, MI can be approximated by separately estimating the densities $p_{\mathrm{xy}}(\boldsymbol{x},\boldsymbol{y})$, $p_{\mathrm{x}}(\boldsymbol{x})$ and $p_{\mathrm{y}}(\boldsymbol{y})$ using $\{\boldsymbol{x}_i,\boldsymbol{y}_i\}_{i=1}^n$. However, density estimation is known to be a hard problem and therefore the KDE-based approach may not be so effective in practice.

3.2 *K*-nearest Neighbor Method (KNN)

Let $\mathcal{N}_k(i)$ be the set of k-nearest neighbor samples of $(\boldsymbol{x}_i,\boldsymbol{y}_i)$, and let $\epsilon_{\mathrm{x}}(i) := \max\{\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\| \mid (\boldsymbol{x}_{i'},\boldsymbol{y}_{i'}) \in \mathcal{N}_k(i)\}$, $\epsilon_{\mathrm{y}}(i) := \max\{\|\boldsymbol{y}_i - \boldsymbol{y}_{i'}\| \mid (\boldsymbol{x}_{i'},\boldsymbol{y}_{i'}) \in \mathcal{N}_k(i)\}$, $n_{\mathrm{x}}(i) := \#\{\boldsymbol{z}_{i'} \mid \|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\| \leq \epsilon_{\mathrm{x}}(i)\}$, $n_{\mathrm{y}}(i) := \#\{\boldsymbol{z}_{i'} \mid \|\boldsymbol{y}_i - \boldsymbol{y}_{i'}\| \leq \epsilon_{\mathrm{y}}(i)\}$. Then the KNN-based MI estimator is given as follows [9]: $\widehat{I}(X,Y) = \psi(k) + \psi(n) - \frac{1}{k} - \frac{1}{n}\sum_{i=1}^n\left[\psi(n_x(i)) + \psi(n_y(i))\right]$, where ψ is the *digamma* function.

Table 1. Relation between existing and proposed MI estimators. If the order of the Edgeworth expansion is regarded a tuning parameter, model selection of EDGE should be 'Not available'.

	Density estimation	Model selection	Distribution
KDE	Involved	**Available**	**Free**
KNN	**Not involved**	Not available	**Free**
EDGE	**Not involved**	**Not necessary**	Nearly normal
LSMI	**Not involved**	**Available**	**Free**

A practical drawback of the KNN-based approach is that the estimation accuracy depends on the value of k and there seems no systematic strategy to choose the value of k appropriately.

3.3 Edgeworth Expansion (EDGE)

MI can be expressed in terms of the entropies as $I(X,Y) = H(X) + H(Y) - H(X,Y)$, where $H(X)$ denotes the entropy of X. Thus MI can be approximated if the entropies are estimated.

[17] proposed an entropy approximation method based on the *Edgeworth expansion*, where the entropy of a distribution is approximated by that of the normal distribution and some additional higher-order correction terms. More specifically, for a d-dimensional distribution, the entropy is approximated by $H \approx H_{\mathrm{normal}} - \frac{1}{12}\sum_{i=1}^{d} \kappa_{i,i,i}^2 - \frac{1}{4}\sum_{i,j=1,i\neq j}^{d} \kappa_{i,i,j}^2 - \frac{1}{72}\sum_{i,j,k=1,i<j<k}^{d} \kappa_{i,j,k}^2$, where H_{normal} is the entropy of the normal distribution with covariance matrix equal to the target distribution and $\kappa_{i,j,k}$ $(1 \leq i,j,k \leq d)$ is the standardized third cumulant of the target distribution. In practice, all the cumulants are estimated from samples.

If the underlying distribution is close to the normal distribution, the above approximation is quite accurate and the EDGE method works very well. However, if the distribution is far from the normal distribution, the approximation error gets large and therefore the EDGE method may be unreliable.

In principle, it is possible to include the fourth and even higher cumulants for further reducing the estimation bias. However, this in turn increases the estimation variance; the expansion up to the third cumulants would be reasonable.

3.4 Discussions

The characteristics of the proposed and existing MI estimators are summarized in Tab.1. KDE is distribution-free and model selection is possible by LCV. However, a hard task of density estimation is involved. KNN is distribution-free and does not involve density estimation directly. However, there is no model selection method for determining the number of nearest neighbors. EDGE does not involve density estimation and any tuning parameters. However, it is based on the assumption that the target distribution is close to normal. LSMI is distribution-free, it does not involve density estimation, and model selection is possible by (LOO)CV. Thus LSMI overcomes the limitations of the existing approaches.

4 Numerical Experiments

In this section, we experimentally investigate the performance of the proposed and existing MI estimators using artificial datasets.

Let us employ the following four datasets (see Fig.1).

(a) Linear dependence: y has a linear dependence on x as

$$x \sim \mathcal{N}(x; 0, 0.5) \quad \text{and} \quad y|x \sim \mathcal{N}(y; 3Mx, 1),$$

where $\mathcal{N}(x; \mu, \sigma^2)$ denotes the normal density with mean μ and variance σ^2 and M $(\in \mathbb{R})$ controls strength of the dependence between x and y.

(b) Non-linear dependence with correlation: y has a quadratic dependence on x as

$$x \sim \mathcal{N}(x; 0, 1) \quad \text{and} \quad y|x \sim \mathcal{N}(y; x^2, 2 - M).$$

(c) Non-linear dependence without correlation: y has a lattice-structured dependence on x as

$$x \sim \mathcal{U}(x; -0.5, 0.5) \;\; \text{and} \;\; y|x \sim \begin{cases} \mathcal{N}(x; 0, M') & \text{if } x \leq |\frac{1}{6}|, \\ \frac{1}{2}\mathcal{N}(x; 1, M') + \frac{1}{2}\mathcal{N}(x; -1, M') & \text{otherwise,} \end{cases}$$

where $\mathcal{U}(x; a, b)$ denotes the uniform density on (a, b) and $M' = (2 + M)^{-1}$.

(d) Independence: x and y are independent to each other as

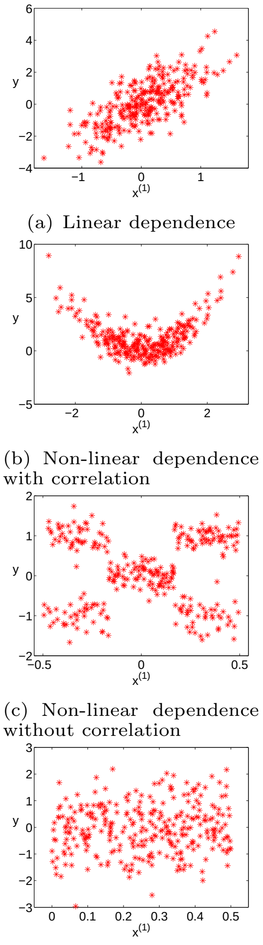
$$x \sim \mathcal{U}(x; 0, 0.5) \quad \text{and} \quad y|x \sim \mathcal{N}(y; 0, 1).$$

4.1 MI Estimation Performance

The task is to estimate MI between x and y. We compare the performance of LSMI(LOOCV), KDE(LCV), and KNN(k) for $k = 1, 5, 15$, where the approximation error of an MI estimate $\widehat{I}$ is measured by $|\widehat{I} - I|$.

Fig.2 depicts the average approximation error over 100 trials as a function of the sample size n when $M = 1$. For the linear dataset (a), the performance of EDGE is the best among all. This is intuitively understandable since all the distributions of x, y, and (x, y) are normal in this case and therefore the Edgeworth approximation is exact (though the EDGE method is not exact since the cumulants are estimated from samples). LSMI performs reasonably well. For the quadratic dataset (b), LSMI outperforms all other estimators. For the dataset (c), EDGE performs poorly since all the distributions are far from the normal distribution. LSMI performs moderately well. For the independent dataset (d), LSMI tends to be better than the others.

In the above simulation, KDE works moderately well for the dependent datasets, but it performs poorly for the independent dataset. KNN works excellently given that the value of k is chosen optimally. Since the best value of k varies depending on the datasets, it needs to be chosen adaptively using the data samples. However, there is no systematic model selection strategy for KNN and therefore KNN would be rather unreliable in practice. EDGE works well for datasets with high normality, but its performance is poor for non-normal datasets. In contrast, LSMI with LOOCV performs reasonably well for all the datasets in a stable manner. Thus LSMI would be reliable.



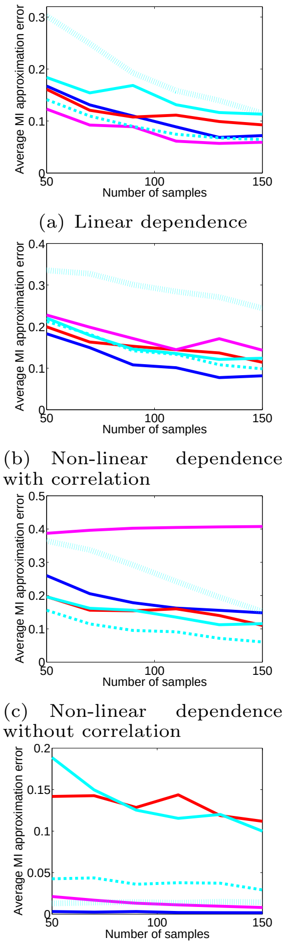
(a) Linear dependence

(b) Non-linear dependence with correlation

(c) Non-linear dependence without correlation

(d) Independence

Fig. 1. Toy datasets



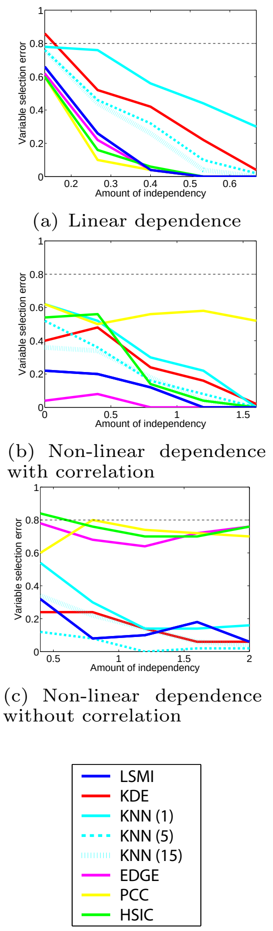
(a) Linear dependence

(b) Non-linear dependence with correlation

(c) Non-linear dependence without correlation

(d) Independence

Fig. 2. Average MI approximation error measured by $|\widehat{I} - I|$.



(a) Linear dependence

(b) Non-linear dependence with correlation

(c) Non-linear dependence without correlation

Fig. 3. Variable selection error. The black dotted line at 0.8 is the error of the random guess.

4.2 Variable Selection Performance

We apply the MI estimators to variable selection and investigate the performance. We consider 5-dimensional input variables $\boldsymbol{x} = (x^{(1)}, x^{(2)}, \ldots, x^{(5)})^\top$, where each input variable $x^{(i)}$ $(i = 1, 2, \ldots, 5)$ independently follows the input distribution (a). The output variable y follows the same conditional distribution as (a), which is dependent only on the first input variable $x^{(1)}$. We choose the most "informative" variable by finding the dimension $x^{(i)}$ with the largest MI estimate against y. Let us measure the error of variable selection by the frequency of finding a wrong dimension (i.e., $i \neq 1$) over 50 trials. Thus, the error of the random guess is 0.8 $(= 1 - 1/5)$.

In addition to LSMI, KDE, EDGE, and KNN, we also test the *Pearson correlation coefficient (PCC)* and the *Hilbert-Schmidt independent criterion (HSIC)*

[18]. HSIC is a dependence measure based on kernelized correlation. The performance of HSIC depends on the Gaussian kernel width; we use the median distance between all pairs of samples as the kernel width, which is suggested as a useful heuristic in [18].

Fig.3 depicts the average variable selection error over 50 trials as a function of strength of the dependence M when $n = 50$; the larger the value of M is, the stronger the dependence of y on $x^{(1)}$ is. In the same figure, we also included the simulation results for the datasets (b) and (c). For the linear dataset (a), LSMI, EDGE, PCC, and HSIC tend to perform better than KDE and KNN. For the quadratic dataset (b), EDGE works the best and is followed by LSMI. For the lattice-structured dataset (c), LSMI, KDE, and KNN perform well, but EDGE, PCC, and HSIC perform poorly; their error is close to the random guess. The failure of PCC is due to uncorrelatedness of the data. On the other hand, the failure of HSIC is caused by an inappropriate choice of the Gaussian kernel width, implying that the heuristic of using the median sample distance as the kernel width is not always appropriate.

Overall, LSMI with LOOCV is shown to be a useful variable selection method that performs stably well in various situations.

5 Protein Subcellular Localization Prediction

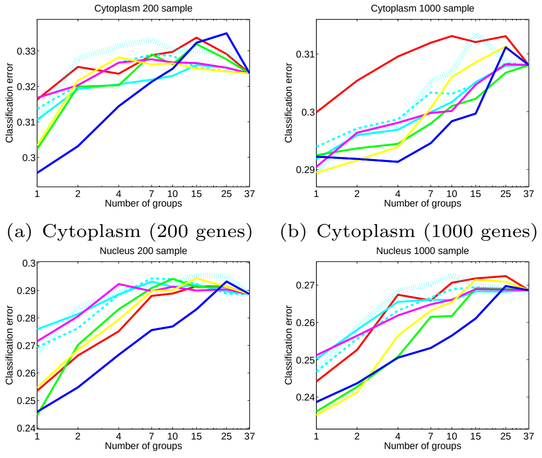
In this section, we apply the proposed method to a real-world biology problem.

The task is to predict protein subcellular localizations of yeast [19] based on 172 microarray data [20]. The 172 microarray data can be categorized into 37 groups depending on the type of stimulations. In this scenario, the use of feature selection methods allows us not only to predict the localization of proteins, but also to associate localized positions of activated proteins with stimulations.

Here we consider the *forward* feature-group addition strategy, i.e., a feature-group score (such as an MI estimate, PCC, or HSIC) between each input feature-group and output y is computed and the top m feature-groups are used for training a classifier. Protein subcellular localizations in the *cytoplasm* and the *nucleus* are predicted. We randomly choose 200 or 1000 genes (samples) from totally 5520 genes for feature-group selection and training a classifier; the rest is used for evaluating the generalization performance. We use a Gaussian kernel support vector machine (GK-SVM) [16], where the kernel width is set at the median distance among all samples and the regularization parameter is fixed at $C = 10$.

Figure 4 depicts the mean classification error over 10 trials as a function of the number of groups used for training a classifier. The results show that LSMI performs quite well for all four cases. Fig.5 depicts the ranking of feature groups obtained by LSMI when 1000 genes are used. This shows that the feature ranking does not significantly change over trials, implying that feature selection by LSMI is highly stable.

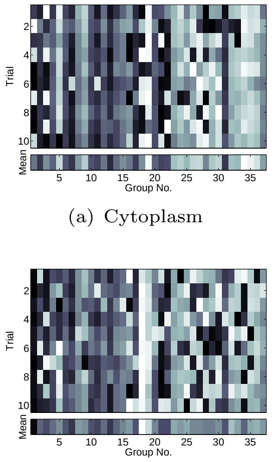
In cytoplasm location prediction, the feature groups 23 to 37 are frequently chosen, which correspond to continuous carbon sources and temperatures. The best classification accuracy is obtained when 4 feature groups are used (see Fig.4(b)), so these feature groups would have a high correlation with protein



(a) Cytoplasm (200 genes) (b) Cytoplasm (1000 genes)

(c) Nucleus (200 genes) (d) Nucleus (1000 genes)

Fig. 4. Classification error against the number of feature groups for the yeast cell datasets. The legends are the same as Fig.3



(a) Cytoplasm

(b) Nucleus

Fig. 5. The ranking of feature groups when 1000 samples are used. A brighter color means a higher feature group ranking.

localization in the cytoplasm. On the other hand, in nucleus localization prediction, the best classification accuracy is obtained with only one feature group (see Fig.4(d)). Since the group 18 (nitrogen depletion) is the most frequently chosen, this would be strongly correlated with protein localization in the nucleus.

6 Conclusions

In this paper, we proposed a new method of estimating mutual information. The proposed method LSMI has several useful properties, e.g., it does not involve density estimation, it is equipped with a cross-validation procedure for model selection, and the solution as well as an approximate leave-one-out error can be computed analytically. We showed the usefulness of LSMI through simulations and biological data analysis.

References

1. Guyon, I., Elisseeff, A.: An introduction to variable feature selection. Journal of Machine Learning Research **3** (2003) 1157–1182
2. Torkkola, K.: Feature extraction by non-parametric mutual information maximization. Journal of Machine Learning Research **3** (2003) 1415–1438
3. Comon, P.: Independent component analysis, a new concept? Signal Processing **36**(3) (1994) 287–314
4. Chiu, D.K., Kolodziejczak, T.: Inferring consensus structure from nucleic acid sequences. Computer Applications in the Biosciences **7**(3) (1991) 347–352

5. Cover, T.M., Thomas, J.A.: Elements of Information Theory. John Wiley & Sons, Inc., N. Y. (1991)
6. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Chapman & Hall/CRC (April 1986)
7. Fraser, A.M., Swinney, H.L.: Independent coordinates for strange attractors from mutual information. Physical Review A **33**(2) (1986) 1134–1140
8. Härdle, W., Müller, M., Sperlich, S., Werwatz, A.: Nonparametric and Semiparametric Models. Springer Series in Statistics. Springer, Berlin (2004)
9. Kraskov, A., Stögbauer, H., Grassberger, P.: Estimating mutual information. Physical Review E **69** (2004) 066138
10. Khan, S., Bandyopadhyay, S., Ganguly, A., Saigal, S.: Relative performance of mutual information estimation methods for quantifying the dependence among short and noisy data. Physical Review E **76** (2007) 026209
11. Sugiyama, M., Nakajima, S., Kashima, H., von Bünau, P., Kawanabe, M.: Direct importance estimation with model selection and its application to covariate shift adaptation. In: Advances in Neural Information Processing Systems 20. (2008)
12. Nguyen, X., Wainwright, M.J., Jordan, M.I.: Estimating divergence functions and the likelihood ratio by penalized convex risk minimization. In: Advances in Neural Information Processing Systems 20. (2008)
13. Tsuboi, Y., Kashima, H., Hido, S., Bickel, S., Sugiyama, M.: Direct density ratio estimation for large-scale covariate shift adaptation. In: SDM. (2008) 443–454
14. van der Vaart, A.W., Wellner, J.A.: Weak Convergence and Empirical Processes. With Applications to Statistics. Springer, New York (1996)
15. van de Geer, S.: Empirical Processes in M-Estimation. Cambridge University Press (2000)
16. Schölkopf, B., Smola, A.J.: Learning with Kernels. MIT Press, Cambridge, MA (2002)
17. Hulle, M.M.V.: Edgeworth approximation of multivariate differential entropy. Neural Computation **17**(9) (2005) 1903–1910
18. Song, L., Smola, A., Gretton, A., Borgwardt, K.M., Bedo, J.: Supervised feature selection via dependence estimation. In: Proceedings of the 24th International Conference on Machine learning, New York, NY, USA, ACM (2007) 823–830
19. Huh, W.K., Falvo, J.V., Gerke, L.C., Carroll, A.S., Howson, R.W., Weissman, J.S., O'Shea, E.K.: Global analysis of protein localization in budding yeast. Nature **425** (2003) 686–691
20. Gasch, A.P., Spellman, P.T., Kao, C.M., Carmel-Harel, O., Eisen, M.B., Storz, G., Botstein, D., Brown, P.O.: Genomic expression programs in the response of yeast cells to environmental changes. Molecular Biology of the Cell **11**(12) (2000) 4241–4257