Direct Density-Derivative Estimation and Its Application in KL-Divergence Approximation

Hiroaki Sasaki
Grad. School of Frontier Sciences
The University of Tokyo
Tokyo, Japan
sasaki@ms.k.u-tokyo.ac.jp

Yung-Kyun Noh
Dept. of Mech. & Aeros. Eng.
Seoul National University
Seoul, Rep. of Korea
nohyung@snu.ac.kr

Masashi Sugiyama
Grad. School of Frontier Sciences
The University of Tokyo
Tokyo, Japan
sugi@k.u-tokyo.ac.jp

Abstract

Estimation of density derivatives is a versatile tool in statistical data analysis. A naive approach is to first estimate the density and then compute its derivative. However, such a two-step approach does not work well because a good density estimator does not necessarily mean a good density-derivative estimator. In this paper, we give a direct method to approximate the density derivative without estimating the density itself. Our proposed estimator allows analytic and computationally efficient approximation of multi-dimensional high-order density derivatives, with the ability that all hyper-parameters can be chosen objectively by cross-validation. We further show that the proposed density-derivative estimator is useful in improving the accuracy of non-parametric KL-divergence estimation via metric learning. The practical superiority of the proposed method is experimentally demonstrated in change detection and feature selection.

1 Introduction

Derivatives of probability density functions play key roles in various statistical data analysis. For example:

- *Mean shift* clustering seeks modes of the data density [1, 2, 3, 4], where the first-order density derivative is the key ingredient.
- A statistical test for modes of the data density, which is based on the second order density derivative [5].
- The optimal bandwidth of kernel density estimation (KDE) depends on the second-order density derivative [6].
- The bias of nearest-neighbor Kullback-Leibler (KL) divergence estimation is governed by the second-order density derivative [7].
- More applications in fundamental statistical problems such as regression, Fisher information estimation, parameter estimation, and hypothesis testing are discussed in [8].

Due to such a wide range of applications, accurately estimating the density derivatives from data is an important research topic in statistics and machine learning.

Given samples $\{x_i\}_{i=1}^n$ drawn from probability density $p(x)$ on $\mathbb{R}$, a naive approach to density-derivative estimation is to first perform density estimation and then compute its derivatives. For example, suppose that KDE is used for density estimation:

$$\widehat{p}(x) \propto \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right),$$

where K is a kernel function (such as the Gaussian kernel) and $h > 0$ is the bandwidth. Then the first-order density derivative is estimated as follows [9, 10]:

$$\widehat{p}'(x) \propto \sum_{i=1}^{n} K'\left(\frac{x - x_i}{h}\right).$$

A cross-validation method for selecting the bandwidth h was proposed in [11]. However, since a good density estimator is not always a good density-derivative estimator, this approach is not necessarily reliable; this

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.

problem becomes more critical if higher-order density derivatives are estimated:

$$\widehat{p}^{(j)}(x) \propto \sum_{i=1}^{n} K^{(j)}\left(\frac{x-x_i}{h}\right).$$

A more direct approach of performing kernel density estimation for density derivatives was proposed [12]:

$$\widehat{p^{(j)}}(x) \propto \sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right).$$

However, this method suffers the bandwidth selection problem because the optimal bandwidth depends on higher-order derivatives than the estimated one [13].

In this paper, we propose a novel density-derivative estimator which finds the minimizer of the mean integrated square error (MISE) to the true density-derivative. The proposed method, which we call *MISE for derivatives* (MISED), possesses various useful properties:

- Density derivatives are directly estimated without going through density estimation.
- The solution can be computed analytically and efficiently.
- All tuning parameters can be objectively optimized by cross-validation.
- Multi-dimensional density derivatives can be directly estimated.
- Higher-order density derivatives can be directly estimated.

MISED is applied to metric learning to improve the accuracy of nearest-neighbor KL-divergence approximation. Through experiments on change detection and feature selection, we demonstrate the usefulness of the proposed MISED-based metric learning method.

2 Direct Density-Derivative Estimation

In this section, we describe our proposed MISED method.

2.1 Problem Formulation

Suppose that independent and identically distributed samples $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ from unknown density $p(\boldsymbol{x})$ on $\mathbb{R}^d$ are available. Our goal is to estimate the k-th order (partial) derivative of $p(\boldsymbol{x})$,

$$p_{k,\boldsymbol{j}}(\boldsymbol{x}) = \frac{\partial^k}{\partial x_1^{j_1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} p(\boldsymbol{x}), \tag{1}$$

where $j_1 + j_2 + \cdots + j_d = k$ for $j_i \in \{0, 1, \dots, k\}$ and $\boldsymbol{j} = (j_1, j_2, \dots, j_d)$. When $k = 1$ (or $k = 2$), $p_{k,\boldsymbol{j}}(\boldsymbol{x})$ corresponds to a single element in the gradient vector (or the Hessian matrix) of $p(\boldsymbol{x})$.

2.2 MISE for Density Derivatives

Let $g_{k,\boldsymbol{j}}(\boldsymbol{x})$ be a model of $p_{k,\boldsymbol{j}}(\boldsymbol{x})$ (its specific form will be introduced later). We learn $g_{k,\boldsymbol{j}}(\boldsymbol{x})$ to minimize the MISE to $p_{k,\boldsymbol{j}}(\boldsymbol{x})$:

$$\begin{aligned} J_{\boldsymbol{j}}(g_{k,\boldsymbol{j}}) &= \int \left\{g_{k,\boldsymbol{j}}(\boldsymbol{x}) - p_{k,\boldsymbol{j}}(\boldsymbol{x})\right\}^2 \mathrm{d}\boldsymbol{x} - C \\ &= \int \left\{g_{k,\boldsymbol{j}}(\boldsymbol{x})\right\}^2 \mathrm{d}\boldsymbol{x} - 2\int g_{k,\boldsymbol{j}}(\boldsymbol{x}) p_{k,\boldsymbol{j}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \end{aligned} \tag{2}$$

where $C = \int \left\{p_{k,\boldsymbol{j}}(\boldsymbol{x})\right\}^2 \mathrm{d}\boldsymbol{x}$.

The first term in (2) is accessible since $g_{k,\boldsymbol{j}}(\boldsymbol{x})$ is a model specified by the user. The second term in (2) seems inaccessible at a glance, but *integration by parts* allows us to transform it as

$$\begin{aligned} &\int g_{k,\boldsymbol{j}}(\boldsymbol{x}) p_{k,\boldsymbol{j}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &= \int g_{k,\boldsymbol{j}}(\boldsymbol{x}) \frac{\partial^k}{\partial x_1^{j_1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \\ &= \int \left[g_{k,\boldsymbol{j}}(\boldsymbol{x}) \frac{\partial^{k-1}}{\partial x_1^{j_1-1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} p(\boldsymbol{x}) \right]_{x_1=-\infty}^{x_1=\infty} \mathrm{d}\boldsymbol{x}_{\backslash x_1} \\ &\quad - \int \frac{\partial}{\partial x_1} g_{k,\boldsymbol{j}}(\boldsymbol{x}) \frac{\partial^{k-1}}{\partial x_1^{j_1-1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \end{aligned}$$

where $\mathrm{d}\boldsymbol{x}_{\backslash x_1}$ denotes the integration except for x_1. The first term in the last equation vanishes under a mild assumption on the tails of $g_{k,\boldsymbol{j}}(\boldsymbol{x})$ and $\frac{\partial^{k-1}}{\partial x_1^{j_1-1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} p(\boldsymbol{x})$. By repeatedly applying integration by parts k times, we arrive at

$$\begin{aligned} J_{\boldsymbol{j}}(g_{k,\boldsymbol{j}}) = &\int \left\{g_{k,\boldsymbol{j}}(\boldsymbol{x})\right\}^2 \mathrm{d}\boldsymbol{x} - 2(-1)^k \\ &\times \int \left\{ \frac{\partial^k}{\partial x_1^{j_1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} g_{k,\boldsymbol{j}}(\boldsymbol{x}) \right\} p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \end{aligned}$$

Approximating the expectation by the sample average gives

$$\begin{aligned} \widetilde{J}_{\boldsymbol{j}}(g_{k,\boldsymbol{j}}) = &\int \left\{g_{k,\boldsymbol{j}}(\boldsymbol{x})\right\}^2 \mathrm{d}\boldsymbol{x} \\ &- \frac{2(-1)^k}{n} \sum_{i=1}^{n} \frac{\partial^k}{\partial x_1^{j_1} \partial x_2^{j_2} \dots \partial x_d^{j_d}} g_{k,\boldsymbol{j}}(\boldsymbol{x}_i). \end{aligned} \tag{3}$$

2.3 Analytic Solution for Gaussian Kernels

As a density-derivative model $g_{k,\boldsymbol{j}}$, we use the Gaussian kernel model[1]:

$$g_{k,\boldsymbol{j}}(\boldsymbol{x}) = \sum_{i=1}^{n} \theta_{\boldsymbol{j},i} \underbrace{\exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}\right)}_{\psi_i(\boldsymbol{x})} = \boldsymbol{\theta}_{\boldsymbol{j}}^{\top}\boldsymbol{\psi}(\boldsymbol{x}), \tag{4}$$

for which the k-th derivative is given by

$$\begin{aligned}&\frac{\partial^k}{\partial x_1^{j_1}\partial x_2^{j_2}\ldots\partial x_d^{j_d}} g_{k,\boldsymbol{j}}(\boldsymbol{x})\\ &= \sum_{i=1}^{n}\theta_{\boldsymbol{j},i} \underbrace{\frac{\partial^k}{\partial x_1^{j_1}\partial x_2^{j_2}\ldots\partial x_d^{j_d}} \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}\right)}_{\varphi_{\boldsymbol{j},i}(\boldsymbol{x})}\\ &= \boldsymbol{\theta}_{\boldsymbol{j}}^{\top}\boldsymbol{\varphi}_{\boldsymbol{j}}(\boldsymbol{x}).\end{aligned}$$

Substituting these formulas into the objective function (3) and adding the ℓ_2-regularizer, we obtain a practical objective function:

$$\widetilde{J}_{\boldsymbol{j}}(\boldsymbol{\theta}_{\boldsymbol{j}}) = \boldsymbol{\theta}_{\boldsymbol{j}}^{\top}\mathbf{G}\boldsymbol{\theta}_{\boldsymbol{j}} - 2(-1)^k \boldsymbol{\theta}_{\boldsymbol{j}}{}^{\top}\boldsymbol{h}_{\boldsymbol{j}} + \lambda\boldsymbol{\theta}_{\boldsymbol{j}}{}^{\top}\boldsymbol{\theta}_{\boldsymbol{j}}, \tag{5}$$

where

$$\begin{aligned}[\mathbf{G}]_{ij} &= \int \psi_i(\boldsymbol{x})\psi_j(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\ &= (\pi\sigma^2)^{d/2}\exp\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2}{4\sigma^2}\right),\\ \boldsymbol{h}_{\boldsymbol{j}} &= \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{\varphi}_{\boldsymbol{j}}(\boldsymbol{x}_i).\end{aligned}$$

The minimizer of (5) is given analytically as

$$\widehat{\boldsymbol{\theta}}_{\boldsymbol{j}} = \arg\min_{\boldsymbol{\theta}_{\boldsymbol{j}}} \widetilde{J}_{\boldsymbol{j}}(\boldsymbol{\theta}_{\boldsymbol{j}}) = (-1)^k\left(\mathbf{G}+\lambda\mathbf{I}\right)^{-1}\boldsymbol{h}_{\boldsymbol{j}}, \tag{6}$$

where $\mathbf{I}$ denotes the identity matrix. Finally, a density-derivative estimator is obtained as

$$\widehat{g}_{k,\boldsymbol{j}}(\boldsymbol{x}) = \widehat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{\top}\boldsymbol{\psi}(\boldsymbol{x}).$$

We call this method the *mean integrated square error for derivatives* (MISED) estimator, which can be regarded as an extension of *score matching* for density estimation [14], *least-squares density-difference* for density-difference estimation [15, 16], and direct estimation for log-density gradients [4, 17] to higher-order derivatives.

2.4 Model Selection by Cross-Validation

The performance of the MISED method depends on the choice of model parameters (the Gaussian width σ and the regularization λ in the current setup). Below, we describe a method to optimize the model by cross-validation, which essentially follows the same line as [11] for kernel density estimation:

1. Divide the sample $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^n$ into T disjoint subsets $\{\mathcal{X}_t\}_{t=1}^T$.
2. Obtain the estimator $\widehat{g}_{k,\boldsymbol{j}}^{(t)}(\boldsymbol{x})$ using $\mathcal{X}\setminus\mathcal{X}_t$, and then compute the hold-out MISE to $\mathcal{X}_t$ as

$$\begin{aligned}\mathrm{CV}(t) = \int\left\{\widehat{g}_{k,\boldsymbol{j}}^{(t)}(\boldsymbol{x})\right\}^2\mathrm{d}\boldsymbol{x} - \frac{2(-1)^k}{|\mathcal{X}_t|}\\ \times\sum_{\boldsymbol{x}\in\mathcal{X}_t}\frac{\partial^k}{\partial x_1^{j_1}\partial x_2^{j_2}\cdots\partial x_d^{j_d}}\widehat{g}_{k,\boldsymbol{j}}^{(t)}(\boldsymbol{x}),\end{aligned} \tag{7}$$

   where $|\mathcal{X}_t|$ denotes the number of elements in $\mathcal{X}_t$.
3. Choose the model that minimizes $\mathrm{CV} = \frac{1}{T}\sum_{t=1}^{T}\mathrm{CV}(t)$.

2.5 Numerical Examples

Let us illustrate the behavior of MISED using $n = 500$ samples drawn from the standard normal distribution. The Gaussian bandwidth σ and the regularization parameter λ included in MISED are chosen by 5-fold cross-validation from $\sigma \in \left\{10^{-0.3}, 10^{-0.1375}, 10^{0.025}, \ldots, 10^{1}\right\}$ and $\lambda \in \left\{10^{-1}, 10^{-0.75}, 10^{-0.5}, \ldots, 10^{1}\right\}$. For comparison, we also test two types of Gaussian KDE: the Gaussian bandwidth h is chosen by 5-fold cross-validation with respect to (a) the hold-out MISED criterion (7) [11] from $\sigma \in \left\{10^{-0.3}, 10^{-0.1375}, 10^{0.025}, \ldots, 10^{1}\right\}$ (denoted by $\mathrm{KDE_M}$) and (b) the hold-out log-likelihood [6] from $\sigma \in \left\{10^{-1}, 10^{-0.75}, 10^{-0.5}, \ldots, 10^{1}\right\}$ (denoted by $\mathrm{KDE_L}$).

Figures 1 (a) and (b) depict the estimation results of the first-order and second-order density-derivatives, showing that MISED works well in density-derivative estimation. Figure 1 (c) depicts the estimation results of the density, where the curve labeled as MISED is the integral of the first-order MISED solution. This shows that MISED also approximates the density function well, up to an unspecified constant.

On the other hand, while $\mathrm{KDE_L}$ works well as a density estimator, it performs poorly as density-derivative estimators in particular for the second-order derivative. This result clearly substantiates that a good density estimator is not necessarily a good density-derivative estimator.

[1] If n is too large, we may only use a subset of data samples as kernel centers.

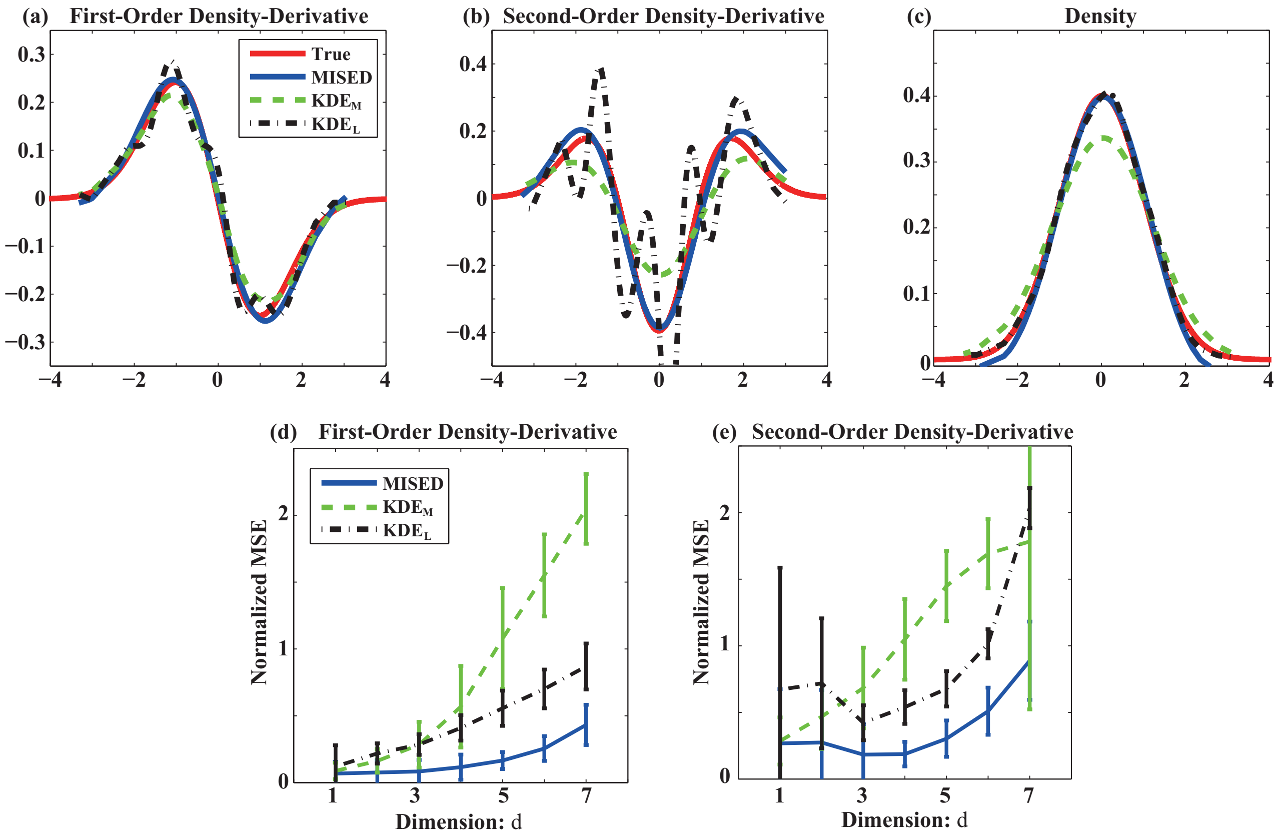


Figure 1: Estimation of the density and density-derivatives of the standard normal distribution. (a), (b), and (c): estimated densities and density-derivatives. (d) and (e): normalized mean squared errors (MSE) of density-derivative estimation as functions of the data dimensionality.

KDE$_\text{M}$ performs better than KDE$_\text{L}$ as density-derivative estimators, but it is not as good as MISED. Also, Figure 1 (c) shows that KDE$_\text{M}$ (using the first-order MISED criterion for cross-validation) provides an overly smoothed density estimate, which is poorer than the integral of MISED. We conjecture that such poor performance of KDE$_\text{M}$ is caused by the limited adaptivity of KDE: the coefficients of Gaussian kernels are fixed to $1/n$ in KDE, while they are learned adaptively in MISED.

Next, we evaluate how the performance of density-derivative estimation is affected when the dimensionality of the standard normal distribution is increased. The performance is evaluated by the normalized mean squared error (MSE):

$$\frac{\frac{1}{n}\sum_{i=1}^{n}\sum_{\boldsymbol{j}}(\widehat{g}_{k,\boldsymbol{j}}(\boldsymbol{x}_i)-p_{k,\boldsymbol{j}}(\boldsymbol{x}_i))^2}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\sum_{\boldsymbol{j}}\widehat{g}_{k,\boldsymbol{j}}(\boldsymbol{x}_i)^2}\sqrt{\frac{1}{n}\sum_{i=1}^{n}\sum_{\boldsymbol{j}}p_{k,\boldsymbol{j}}(\boldsymbol{x}_i)^2}},$$

where $\sum_{\boldsymbol{j}}$ denotes the summation of all elements in the gradient vector (or the Hessian matrix) when $k=1$ (or $k=2$). We use the common σ (and λ for MISED) for all elements in the gradient vector or Hessian matrix, which is selected by cross-validation with respect to the hold-out MISED criterion (7) summed over all elements. Figures 1 (d) and (e) show that the normalized MSE for MISED increases much more mildly than those for KDE$_\text{M}$ and KDE$_\text{L}$, illustrating high reliability of MISED in high-dimensional problems.

3 Application to KL-Divergence Approximation

In this section, we apply density-derivative estimation to KL-divergence approximation.

3.1 Nearest-Neighbor KL-Divergence Approximation

The KL-divergence from one density $p_1(\boldsymbol{x})$ to another density $p_2(\boldsymbol{x})$, defined as

$$\mathrm{KL}(p_1\|p_2)=\int p_1(\boldsymbol{x})\log\frac{p_1(\boldsymbol{x})}{p_2(\boldsymbol{x})}\mathrm{d}\boldsymbol{x},$$

is useful for various purposes such as two-sample homogeneity testing [18], feature selection [19], and change detection [20]. Here, we consider the KL-divergence approximator based on *nearest-neighbor density estimation* (NNDE) [21] from two sets of independent samples $\mathcal{X}_1 = \{\boldsymbol{x}_i\}_{i=1}^{n_1}$ and $\mathcal{X}_2 = \{\boldsymbol{x}_i\}_{i=n_1+1}^{n_1+n_2}$ following $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ on $\mathbb{R}^d$:

$$\widehat{\mathrm{KL}}(p_1 \| p_2) = \frac{1}{n_1}\sum_{i=1}^{n_1} \log \frac{(n_1-1)\mathrm{dist}_1(\boldsymbol{x}_i)^d}{n_2 \mathrm{dist}_2(\boldsymbol{x}_i)^d},$$

where $\mathrm{dist}_1(\boldsymbol{x})$ and $\mathrm{dist}_2(\boldsymbol{x})$ denote the distance from $\boldsymbol{x}$ to the nearest samples except for $\boldsymbol{x}$ itself in $\mathcal{X}_1$ and $\mathcal{X}_2$, respectively.

3.2 Metric Learning for NNDE-Based KL-Divergence Approximation

Although the KL-divergence itself is metric-invariant, the NNDE-based KL-divergence approximator is metric-dependent. Indeed, it was shown in [7] that the bias of the NNDE-based KL-divergence approximator at $\boldsymbol{x}$ is approximately proportional to

$$\frac{\mathrm{tr}(\nabla\nabla p_1)}{((n_1-1)p_1)^{2/d}p_1} - \frac{\mathrm{tr}(\nabla\nabla p_2)}{(n_2 p_2)^{2/d}p_2},$$

where $\nabla\nabla p_1$ and $\nabla\nabla p_2$ are the Hessian matrices which are metric-dependent. Therefore, changing the metric in the input space is expected to reduce the bias.

It was shown in [7] that the best local Mahalanobis metric $(\boldsymbol{x}-\boldsymbol{x}')^\top \widehat{\mathbf{A}}(\boldsymbol{x}-\boldsymbol{x}')$ for point $\boldsymbol{x}$ that minimizes the approximative bias is given by as the solution of the following optimization problem:

$$\widehat{\mathbf{A}} = \min_{\mathbf{A}} \mathrm{tr}\left(\mathbf{A}^{-1}\mathbf{B}\right),$$
$$\text{s.t. } \mathbf{A}^\top = \mathbf{A},\ |\mathbf{A}| = 1, \text{ and } \mathbf{A} \succ 0,$$

where

$$\mathbf{B} = \frac{1}{((n_1-1)p_1)^{2/d}}\frac{\nabla\nabla p_1}{p_1} - \frac{1}{(n_2 p_2)^{2/d}}\frac{\nabla\nabla p_2}{p_2}.$$

It was shown that the solution $\widehat{\mathbf{A}}$ is given analytically up to a scaling factor as

$$\widehat{\mathbf{A}} \propto [\mathbf{U}_+ \ \mathbf{U}_-]\begin{pmatrix} d_+\Lambda_+ & 0 \\ 0 & -d_-\Lambda_- \end{pmatrix}[\mathbf{U}_+ \ \mathbf{U}_-]^\top,$$

where $\Lambda_+ \in \mathbb{R}^{d_+ \times d_+}$ and $\Lambda_- \in \mathbb{R}^{d_- \times d_-}$ are the diagonal matrices which contain d_+ positive and d_- negative eigenvalues of $\mathbf{B}$, respectively. The matrices $\widehat{\mathbf{A}}$ and $\mathbf{B}$ share the same eigenvectors, and $\mathbf{U}_+ \in \mathbb{R}^{d \times d_+}$ and $\mathbf{U}_- \in \mathbb{R}^{d_- \times d_-}$ are collections of eigenvectors corresponding to the eigenvalues in Λ_+ and Λ_-, respectively.

In [7], the authors assumed that p_1 and p_2 are both nearly Gaussian, and estimated densities p_1 and p_2 as well as their Hessian matrices $\nabla\nabla p_1$ and $\nabla\nabla p_2$ from the Gaussian models with maximum likelihood estimation. It was demonstrated that the accuracy of NNDE-based KL-divergence approximation is significantly improved when p_1 and p_2 are nearly Gaussian.

3.3 Applying MISED to Metric Learning for NNDE-Based KL-Divergence Approximation

However, the above method does not work well if p_1 and p_2 are apart from Gaussian. To cope with this problem, a naive approach to estimating $\mathbf{B}$ is to perform density estimation for p_1 and p_2 separately, compute their Hessian matrices $\nabla\nabla p_1$ and $\nabla\nabla p_2$, and plug them in the definition of $\mathbf{B}$. However, such a plug-in approach can be unreliable because a good density estimator does not necessarily mean a good estimator of its Hessian matrix, as we have already shown in Section 2.5. In addition, division by estimated densities in $\mathbf{B}$ can significantly magnify the estimation errors of Hessian matrices. Here, we propose to use MISED to cope with this problem.

Since the scale of $\mathbf{B}$ is arbitrary, let us use the following rescaled matrix $\widetilde{\mathbf{B}}$ instead:

$$\widetilde{\mathbf{B}} = \frac{1}{(n_1-1)^{2/d}}\left\{\frac{p_2}{p_1}\right\}^{2/d+1}\nabla\nabla p_1 - \frac{1}{n_2^{2/d}}\nabla\nabla p_2. \tag{8}$$

We then estimate the Hessian matrices $\nabla\nabla p_1$ and $\nabla\nabla p_2$ by MISED and the density ratio p_2/p_1 by the unconstrained least-squares density-ratio estimator [22] that directly estimates the density ratio in a non-parametric manner without estimating each density. By this, we can perform metric learning in a non-parametric way without explicitly estimating the densities p_1 and p_2.

3.4 Numerical Examples

We experimentally compare the behavior of the NNDE-based KL-divergence approximator with MISED-based metric learning to the following methods:

- NNDE-based KL-divergence approximator without metric learning (NN) [21].
- NNDE-based KL-divergence approximator with Gaussian-based metric learning (NNG) [7].
- NNDE-based KL-divergence approximator with $\mathrm{KDE_L}$-based metric learning ($\mathrm{KDE_L}$).
- Density-ratio-based non-parametric KL-divergence estimator (Ratio) [23].

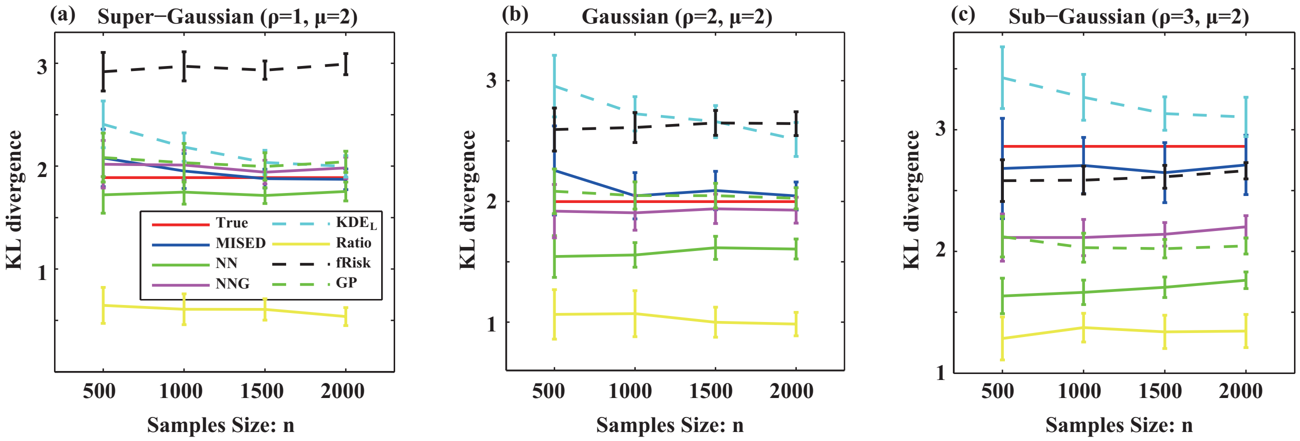


Figure 2: KL-divergence estimation for (a) super-Gaussian, (b) Gaussian and (c) sub-Gaussian data as functions of sample size n.

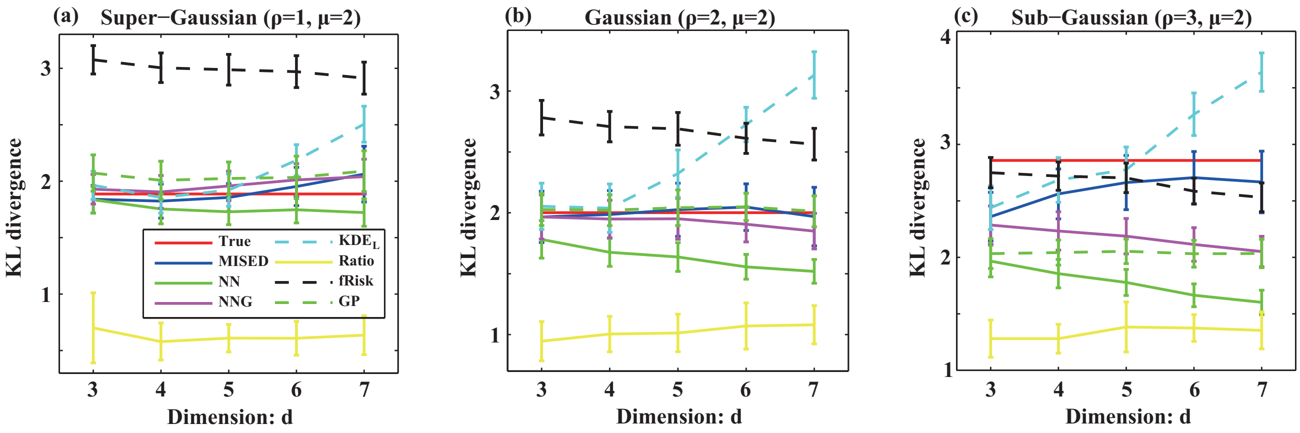


Figure 3: KL-divergence estimation for (a) super-Gaussian, (b) Gaussian, and (c) sub-Gaussian data as functions of dimension d.

- Risk-based nearest-neighbor KL-divergence estimator (fRisk) [24].

- Gaussian parametric KL-divergence estimator with maximum likelihood estimation (GP).

For model selection of MISED, we perform cross-validation as in Section 2.4 from $\sigma \in \{10^{-0.4}, 10^{-0.225}, 10^{0.125}, \ldots, 10^{1}\}$ and $\lambda \in \{10^{-1}, 10^{-0.75}, \ldots, 10^{1}\}$. For KDE$_\mathrm{L}$, the Gaussian bandwidth is selected by log-likelihood cross-validation from the same candidate values as MISED.

We generate data samples from the generalized Gaussian distribution:

$$p_{\mathrm{GG}}(x; \mu, \beta, \rho) = \frac{\beta^{1/2}}{2\Gamma(1+1/\rho)} \exp\left(-\beta^{\rho/2}|x-\mu|^{\rho}\right),$$

where $\mu \in \mathbb{R}$ denotes the mean, $\beta > 0$ controls the variance, and $\rho > 0$ controls the Gaussianity: $\rho < 2$, $\rho = 2$, and $\rho > 2$ correspond to super-Gaussian, Gaussian, and sub-Gaussian distributions, respectively. For $\boldsymbol{x} = (x^{(1)}, \ldots, x^{(d)})^\top$ with $d = 6$, we set

$$p_1(\boldsymbol{x}) = \prod_{j=1}^{d} p_{\mathrm{GG}}(x^{(j)}; 0, \beta, \rho),$$

$$p_2(\boldsymbol{x}) = p_{\mathrm{GG}}(x^{(1)}; 2, \beta, \rho) \prod_{j=2}^{d} p_{\mathrm{GG}}(x^{(j)}; 0, \beta, \rho),$$

where the value of β is selected so that the variance is one. We evaluate the performance of each method when sample size n and Gaussianity ρ are changed.

The experimental results for $\rho = 1, 2, 3$ and $n = 500, 1000, 1500, 2000$ are presented in Figure 2. The proposed MISED outperforms the plain NN (without metric learning) for all three cases, and it outperforms NNG and GP for the super-Gaussian and sub-Gaussian cases. Even for the Gaussian case, MISED is comparable with GP and NNG both of which assume the Gaussianity in KL divergence estimation, while MISED provides a completely non-parametric KL divergence approximator. KDE_L is also a non-parametric approximator, but does not work well especially when the sample size is small. fRisk is comparable to MISED for the sub-Gaussian case, but it largely overestimates for the other two cases. Ratio is a non-parametric method, but it systematically underestimates for all three cases.

Figure 3 indicates dimension scalability of each method when $n = 1,000$. For all data, MISED performs well for a wide range of data dimensionalities. On the other hand, the performance of KDE_L gets worse for all data types as the dimensionality of data increases. The performance of the other methods largely depends on data types. These results show that our approach of avoiding density estimation and division is a promising approach.

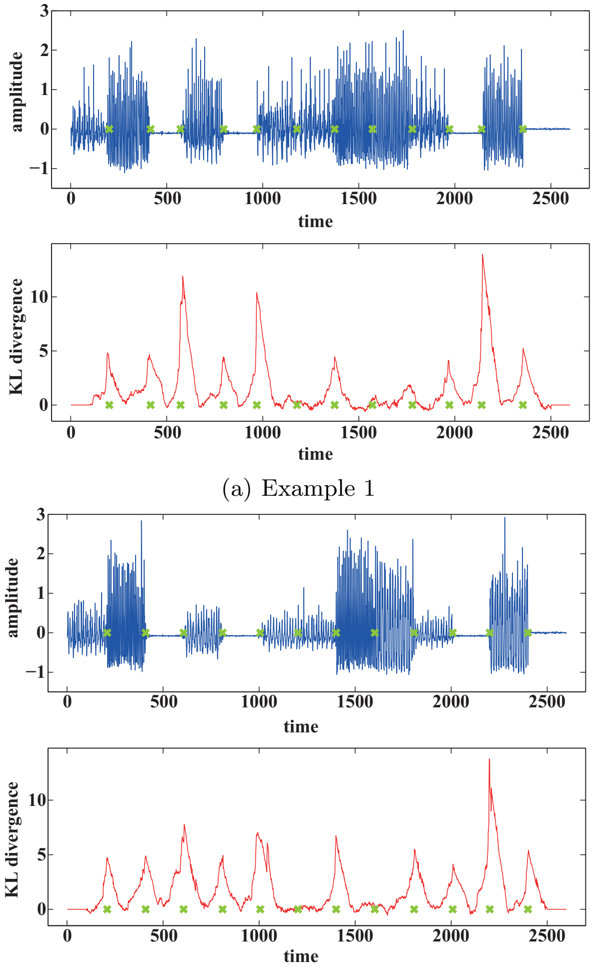
3.5 Experiments on Distributional Change Detection

The goal of change detection is to find abrupt changes in time-series data. We use an m-dimensional real vector $\boldsymbol{y}(t)$ to represent a segment of time series at time t, and a collection of r such vectors is obtained in a sliding-window manner:

$$\boldsymbol{Y}(t) := \{\boldsymbol{y}(t), \boldsymbol{y}(t+1), \ldots, \boldsymbol{y}(t+r-1)\}.$$

Following [20], we consider an underlying density function that generates r retrospective vectors in $\boldsymbol{Y}(t)$. We measure the KL-divergence between the underlying density functions of the two sets, $\boldsymbol{Y}(t)$ and $\boldsymbol{Y}(t+r+m)$ for every t, and determine a point t_0+r+m as a change point if the KL-divergence for $\boldsymbol{Y}(t_0)$ and $\boldsymbol{Y}(t_0+r+m)$ is greater than a predefined threshold. In the experiment, we set $r = 3$ and $m = 100$.

We use the *Human Activity Sensing Consortium (HASC) Challenge 2011* data collection[2], which provides human activity information collected by a portable three-axis accelerometer. Our task is to segment different activities such as “stay”, “walk”, “jog”, and “skip”. Because the orientation of the accelerometer is not necessarily fixed, we took the ℓ_2-norm of 3-dimensional accelerometer data and obtained one-dimensional data, following [20].



(a) Example 1

(b) Example 2

Figure 4: HASC time series data (top) and the KL-divergence estimated by MISED (bottom). Green symbols represent the true change points.

Figure 4 depicts examples of time-series data and their KL-divergences (which are regarded as change scores). These graphs show that the change scores tend to be large at the true change points. Next, we more systematically evaluate the performance of change detection using the AUC (area under the ROC curve) scores. The results are summarized in Table 1, where each point was classified as a change point if it is within a small tolerance region (±10 ms around an “exact” change point). The table shows that the proposed MISED outperforms GP and NNG, and is comparable to fRisk. In the experiments in Figure 2, fRisk gave similar values for different distributions even when the true KL-divergence is large. This was poor as a KL-divergence approximator, but this property seems to work as a “regularizer” to stabilize the change score to avoid incurring big errors. Similar tendencies were also reported in the previous work [7].

[2] `http://hasc.jp/hc2011/`

Table 1: Means and standard deviations of the area under the ROC curve (AUC) over 10 runs. The best method and methods comparable to the best one in terms of the mean AUC by the one-tailed Welch's t-test with significance level 5% are highlighted in boldface.

GP	NNG [7]	fRisk [24]	MISED
0.747(0.050)	0.822(0.030)	**0.858(0.022)**	**0.839(0.028)**

3.6 Experiments on Information-Theoretic Feature Selection

Finally, KL-divergence approximation is applied to selecting relevant features for classification. The *Jensen-Shannon (JS) divergence* is an information-theoretic measure between binary class labels $y \in \{1, 2\}$ and features $\boldsymbol{x} \in \mathbb{R}^d$:

$$\begin{aligned}\mathrm{JS}(\mathcal{X}; \boldsymbol{y}) &= -\sum_{y=1}^{2} \int p(\boldsymbol{x}, y) \log \frac{p(\boldsymbol{x})p(y)}{p(\boldsymbol{x}, y)} \mathrm{d}\boldsymbol{x} \\ &= p(y = 1)\mathrm{KL}(p(\boldsymbol{x}|y = 1)\|p(\boldsymbol{x})) \\ &\quad + p(y = 2)\mathrm{KL}(p(\boldsymbol{x}|y = 2)\|p(\boldsymbol{x})),\end{aligned}$$

where $p(\boldsymbol{x}) = p(y = 1)p(\boldsymbol{x}|y = 1) + p(y = 2)p(\boldsymbol{x}|y = 2)$. Here, we non-parametrically estimate the JS divergence for feature selection.

We use two gene expression datasets of breast cancer prognosis studies: "SMK-CAN-187" [25] and "VANTVEER" [26]. The SMK-CAN-187 dataset contains 90 positive (alive) and 97 negative (dead after 5 years) samples with 19993 features. We use 65 randomly selected samples per class for training and use the rest for evaluating the test classification performance. The VANTVEER dataset contains 46 positive and 51 negative samples with 24481 features. We use 35 randomly selected data per class for training and use the rest for evaluating the test classification performance.

For comparison, the JS divergence is estimated by NNG and mIMR, which provide a non-parametric and parametric approximations to the divergence [27], respectively. As another feature selection method, we employ the *t-score* defined by

$$\text{t-score} = \frac{|\hat{\mu}_1 - \hat{\mu}_2|}{\sqrt{\frac{\hat{\sigma}_1^2}{n_1} + \frac{\hat{\sigma}_2^2}{n_2}}},$$

where $\hat{\mu}_1$ (and $\hat{\mu}_2$) and $\hat{\sigma}_1$ (and $\hat{\sigma}_2$) are the mean value and standard deviation of class 1 (and class 2), respectively.

We choose 20 features based on the forward selection strategy and compare the AUC of classification. The results are summarized in Figure 5, showing that the proposed method works reasonably well in this challenging feature selection scenario.

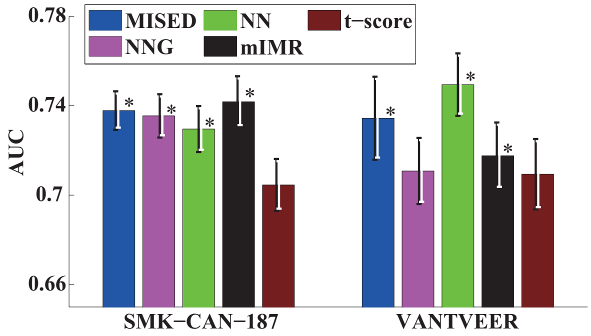


Figure 5: Gene expression classification with feature selection. The best method and methods comparable to the best one in terms of the mean AUC by the one-tailed Welch's t-test with significance level 5% are highlighted by the asterisks.

4 Conclusion

We proposed a method to directly estimate density derivatives. The proposed estimator, called MISED, was shown to possess various useful properties, e.g., analytic and computationally efficient estimation of multi-dimensional high-order density derivatives is possible and all hyper-parameters can be chosen objectively by cross-validation. We further proposed a MISED-based metric learning method to improve the accuracy of nearest-neighbor KL-divergence approximation, and its practical usefulness was experimentally demonstrated in change detection and feature selection.

Estimation of density derivatives is versatile and useful in various machine learning tasks beyond KL-divergence approximation. In our future work, we will explore more applications based on the proposed MISED method.

Acknowledgements

The authors are very grateful to Song Liu for providing us his MATLAB codes for change detection. Sasaki is supported by MEXT KAKENHI 23120004, Noh is supported by the BK21Plus 2015 project, and Sugiyama is supported by JST CREST.

References

[1] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, 1975.

[2] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799, 1995.

[3] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[4] H. Sasaki, A. Hyvärinen, and M. Sugiyama. Clustering via mode seeking by direct estimation of the gradient of a log-density. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD 2014)*, pages 19–34, Nancy, France, 2014.

[5] C. Genovese, M. Perone-Pacifico, I. Verdinelli, and L. Wasserman. Nonparametric inference for density modes. *arXiv preprint arXiv:1312.7567*, 2013.

[6] B.W. Silverman. *Density estimation for statistics and data analysis*. CRC press, 1986.

[7] Y. K. Noh, M. Sugiyama, S. Liu, M. C. du Plessis, F. C. Park, and D. D. Lee. Bias reduction and metric learning for nearest-neighbor estimation of kullback-leibler divergence. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 669–677, 2014.

[8] R.S. Singh. Applications of estimators of a density and its derivatives to certain statistical problems. *Journal of the Royal Statistical Society. Series B*, 39(3):357–363, 1977.

[9] P.K. Bhattacharya. Estimation of a probability density function and its derivatives. *Sankhyā: The Indian Journal of Statistics, Series A*, 29(4):373–382, 1967.

[10] E.F. Schuster. Estimation of a probability density function and its derivatives. *The Annals of Mathematical Statistics*, 40(4):1187–1195, 1969.

[11] W. Hardle, J.S. Marron, and M.P. Wand. Bandwidth choice for density derivatives. *Journal of the Royal Statistical Society, Series B*, 52(1):223–232, 1990.

[12] R.S. Singh. Improvement on some known nonparametric uniformly consistent estimators of derivatives of a density. *The Annals of Statistics*, 5(2):394–399, 1977.

[13] R.S. Singh. On the exact asymptotic behavior of estimators of a density and its derivatives. *The Annals of Statistics*, 9(2):453–456, 1981.

[14] A. Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

[15] J. Kim and C. Scott. L_2 kernel classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(10):1822–1831, 2010.

[16] M. Sugiyama, T. Suzuki, T. Kanamori, M. C. du Plessis, S. Liu, and I. Takeuchi. Density-difference estimation. *Neural Computation*, 25(10):2734–2775, 2013.

[17] D. D. Cox. A penalty method for nonparametric estimation of the logarithmic derivative of a density function. *Annals of the Institute of Statistical Mathematics*, 37(1):271–288, 1985.

[18] T. Kanamori, T. Suzuki, and M. Sugiyama. f-divergence estimation and two-sample homogeneity test under semiparametric density-ratio models. *IEEE Transactions on Information Theory*, 58(2):708–720, 2012.

[19] G. Brown. A new perspective for information theoretic feature selection. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS-09)*, volume 5, pages 49–56, 2009.

[20] S. Liu, M. Yamada, N. Collier, and M. Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.

[21] Q. Wang, S. R. Kulkarni, and S. Verdu. A nearest-neighbor approach to estimating divergence between continuous random vectors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 55(5):2392–2405, 2006.

[22] T. Kanamori, S. Hido, and M. Sugiyama. A least-squares approach to direct importance estimation. *The Journal of Machine Learning Research*, 10:1391–1445, 2009.

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

[24] D. Garcia-Garcia, U. von Luxburg, and R. Santos-Rodriguez. Risk-based generalizations of f-divergences. In *Proceedings of 28th International Conference on Machine Learning*, pages 417–424, 2011.

[25] W. A. Freije, et al. Gene expression profiling of gliomas strongly predicts survival. *Cancer Research*, 15;64(18):6503–6510, 2004.

[26] L. J. van't Veer, H. Dai, M. J. Van De Vijver, Y. D. He, A. A.M. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, and S. H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *nature*, 415(6871):530–536, 2002.

[27] G. Bontempi and P. E. Meyer. Causal filter selection in microarray data. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 95–102, Haifa, Israel, 2010.