Semi-Supervised Information-Maximization Clustering

Daniele Calandriello
Politecnico di Milano, Milano, Italy
daniele.calandriello@mail.polimi.it

Gang Niu
Baidu Inc., Beijing, China
niugang@baidu.com

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

Abstract

Semi-supervised clustering aims to introduce prior knowledge in the decision process of a clustering algorithm. In this paper, we propose a novel semi-supervised clustering algorithm based on the information-maximization principle. The proposed method is an extension of a previous unsupervised information-maximization clustering algorithm based on squared-loss mutual information to effectively incorporate must-links and cannot-links. The proposed method is computationally efficient because the clustering solution can be obtained analytically via eigendecomposition. Furthermore, the proposed method allows systematic optimization of tuning parameters such as the kernel width, given the degree of belief in the must-links and cannot-links. The usefulness of the proposed method is demonstrated through experiments.

Keywords

Clustering, Information Maximization, Squared-Loss Mutual Information, Semi-supervised.

1 Introduction

The objective of clustering is to classify unlabeled data into disjoint groups based on their similarity, and clustering has been extensively studied in statistics and machine learning. *K-means* (MacQueen, 1967) is a classic algorithm that clusters data so that the sum of within-cluster scatters is minimized. However, its usefulness is rather limited in practice because k-means only produces linearly separated clusters. *Kernel k-means* (Girolami, 2002) overcomes this limitation by performing k-means in a feature space induced by a

reproducing kernel function (Schölkopf and Smola, 2002). *Spectral clustering* (Shi and Malik, 2000; Ng et al., 2002) first unfolds non-linear data manifolds based on sample-sample similarity by a spectral embedding method, and then performs k-means in the embedded space.

These non-linear clustering techniques are capable of handling highly complex real-world data. However, they lack objective model selection strategies, i.e., tuning parameters included in kernel functions or similarity measures need to be manually determined in an unsupervised manner. *Information-maximization clustering* can address the issue of model selection (Agakov and Barber, 2006; Gomes et al., 2010; Sugiyama et al., 2014), which learns a probabilistic classifier so that some information measure between feature vectors and cluster assignments is maximized in an unsupervised manner. In the information-maximization approach, tuning parameters included in kernel functions or similarity measures can be systematically determined based on the information-maximization principle. Among the information-maximization clustering methods, the algorithm based on *squared-loss mutual information* (SMI) was demonstrated to be promising (Sugiyama et al., 2014; Sugiyama, 2013), because it gives the clustering solution analytically via eigendecomposition.

In practical situations, additional side information regarding clustering solutions is often provided, typically in the form of *must-links* and *cannot-links*: A set of sample pairs which should belong to the same cluster and a set of sample pairs which should belong to different clusters, respectively. Such semi-supervised clustering (which is also known as clustering with side information) has been shown to be useful in practice (Wagstaff and Cardie, 2000; Goldberg, 2007; Wagstaff et al., 2001). *Spectral learning* (Kamvar et al., 2003) is a semi-supervised extension of spectral clustering that enhances the similarity with side information so that sample pairs tied with must-links have higher similarity and sample pairs tied with cannot-links have lower similarity. On the other hand, *constrained spectral clustering* (Wang and Davidson, 2010) incorporates the must-links and cannot-links as constraints in the optimization problem.

However, in the same way as unsupervised clustering, the above semi-supervised clustering methods suffer from lack of objective model selection strategies and thus tuning parameters included in similarity measures need to be determined manually. In this paper, we extend the unsupervised SMI-based clustering method to the semi-supervised clustering scenario. The proposed method, called *semi-supervised SMI-based clustering* (3SMIC), gives the clustering solution analytically via eigendecomposition with a systematic model selection strategy. Through experiments on real-world datasets, we demonstrate the usefulness of the proposed 3SMIC algorithm.

2 Information-Maximization Clustering with Squared-Loss Mutual Information

In this section, we formulate the problem of information-maximization clustering and review an existing unsupervised clustering method based on squared-loss mutual infor-

mation.

2.1 Information-Maximization Clustering

The goal of unsupervised clustering is to assign class labels to data instances so that similar instances share the same label and dissimilar instances have different labels. Let $\{\boldsymbol{x}_i | \boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^n$ be feature vectors of data instances, which are drawn independently from a probability distribution with density $p^*(\boldsymbol{x})$. Let $\{y_i | y_i \in \{1, \ldots, c\}\}_{i=1}^n$ be class labels that we want to obtain, where c denotes the number of classes and we assume c to be known through the paper.

The information-maximization approach tries to learn the class-posterior probability $p^*(y|\boldsymbol{x})$ in an unsupervised manner so that some "information" measure between feature $\boldsymbol{x}$ and label y is maximized. *Mutual information* (MI) (Shannon, 1948) is a typical information measure for this purpose (Agakov and Barber, 2006; Gomes et al., 2010):

$$\mathrm{MI} := \int \sum_{y=1}^{c} p^*(\boldsymbol{x}, y) \log \frac{p^*(\boldsymbol{x}, y)}{p^*(\boldsymbol{x}) p^*(y)} \mathrm{d}\boldsymbol{x}. \tag{1}$$

An advantage of the information-maximization formulation is that tuning parameters included in clustering algorithms such as the Gaussian width and the regularization parameter can be objectively optimized based on the same information-maximization principle. However, MI is known to be sensitive to outliers (Basu et al., 1998), due to the log function that is strongly non-linear. Furthermore, unsupervised learning of class-posterior probability $p^*(y|\boldsymbol{x})$ under MI is highly non-convex and finding a good local optimum is not straightforward in practice (Gomes et al., 2010).

To cope with this problem, an alternative information measure called *squared-loss MI* (SMI) has been introduced (Suzuki et al., 2009; Sugiyama, 2013):

$$\mathrm{SMI} := \frac{1}{2} \int \sum_{y=1}^{c} p^*(\boldsymbol{x}) p^*(y) \left(\frac{p^*(\boldsymbol{x}, y)}{p^*(\boldsymbol{x}) p^*(y)} - 1 \right)^2 \mathrm{d}\boldsymbol{x}. \tag{2}$$

Ordinary MI is the *Kullback-Leibler (KL) divergence* (Kullback and Leibler, 1951) from $p^*(\boldsymbol{x}, y)$ to $p^*(\boldsymbol{x})p^*(y)$, while SMI is the *Pearson (PE) divergence* (Pearson, 1900). Both KL and PE divergences belong to the class of the *Ali-Silvey-Csiszár divergences* (Ali and Silvey, 1966; Csiszár, 1967), which is also known as the *f-divergences*. Thus, MI and SMI share many common properties, for example, they are non-negative and equal to zero if and only if feature vector $\boldsymbol{x}$ and label y are statistically independent. Information-maximization clustering based on SMI was shown to be computationally advantageous (Sugiyama et al., 2014). Below, we review the SMI-based clustering (SMIC) algorithm.

2.2 SMI-Based Clustering

In unsupervised clustering, it is not straightforward to approximate SMI (2) because labeled samples are not available. To cope with this problem, let us expand the squared

term in Eq.(2). Then SMI can be expressed as

$$\begin{aligned}\mathrm{SMI} &= \frac{1}{2}\int \sum_{y=1}^{c} p^*(\boldsymbol{x})p^*(y)\left(\frac{p^*(\boldsymbol{x},y)}{p^*(\boldsymbol{x})p^*(y)}\right)^2 \mathrm{d}\boldsymbol{x} \\ &\quad - \int \sum_{y=1}^{c} p^*(\boldsymbol{x})p^*(y)\frac{p^*(\boldsymbol{x},y)}{p^*(\boldsymbol{x})p^*(y)}\mathrm{d}\boldsymbol{x} + \frac{1}{2} \\ &= \frac{1}{2}\int \sum_{y=1}^{c} p^*(y|\boldsymbol{x})p^*(\boldsymbol{x})\frac{p^*(y|\boldsymbol{x})}{p^*(y)}\mathrm{d}\boldsymbol{x} - \frac{1}{2}. \end{aligned} \tag{3}$$

Suppose that the class-prior probability $p^*(y)$ is uniform, i.e.,

$$p(y) = \frac{1}{c} \text{ for } y = 1, \ldots, c.$$

Then we can express Eq.(3) as

$$\frac{c}{2}\int \sum_{y=1}^{c} p^*(y|\boldsymbol{x})p^*(\boldsymbol{x})p^*(y|\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \frac{1}{2}. \tag{4}$$

Let us approximate the class-posterior probability $p^*(y|\boldsymbol{x})$ by the following kernel model:

$$p(y|\boldsymbol{x};\boldsymbol{\alpha}) := \sum_{i=1}^{n} \alpha_{y,i} K(\boldsymbol{x},\boldsymbol{x}_i), \tag{5}$$

where $\boldsymbol{\alpha} = (\alpha_{1,1}, \ldots, \alpha_{c,n})^\top \in \mathbb{R}^{cn}$ is the parameter vector, $^\top$ denotes the transpose, and $K(\boldsymbol{x},\boldsymbol{x}')$ denotes a kernel function. Let $\boldsymbol{K}$ be the kernel matrix whose (i,j) element is given by $K(\boldsymbol{x}_i,\boldsymbol{x}_j)$ and let $\boldsymbol{\alpha}_y = (\alpha_{y,1}, \ldots, \alpha_{y,n})^\top \in \mathbb{R}^n$. Approximating the expectation over $p^*(\boldsymbol{x})$ in Eq.(4) with the empirical average of samples $\{\boldsymbol{x}_i\}_{i=1}^n$ and replacing the class-posterior probability $p^*(y|\boldsymbol{x})$ with the kernel model $p(y|\boldsymbol{x};\boldsymbol{\alpha})$, we have the following SMI approximator:

$$\widehat{\mathrm{SMI}} := \frac{c}{2n}\sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K}^2 \boldsymbol{\alpha}_y - \frac{1}{2}. \tag{6}$$

Under orthonormality of $\{\boldsymbol{\alpha}_y\}_{y=1}^c$, a global maximizer is given by the normalized eigenvectors $\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_c$ associated with the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$ of $\boldsymbol{K}$. Because the sign of eigenvector $\boldsymbol{\phi}_y$ is arbitrary, we set the sign as

$$\widetilde{\boldsymbol{\phi}}_y = \boldsymbol{\phi}_y \times \mathrm{sign}(\boldsymbol{\phi}_y^\top \mathbf{1}_n),$$

where $\mathrm{sign}(\cdot)$ denotes the sign of a scalar and $\mathbf{1}_n$ denotes the n-dimensional vector with all ones. On the other hand, since

$$p^*(y) = \int p^*(y|\boldsymbol{x})p^*(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \approx \frac{1}{n}\sum_{i=1}^{n} p(y|\boldsymbol{x}_i;\boldsymbol{\alpha}) = \boldsymbol{\alpha}_y^\top \boldsymbol{K}\mathbf{1}_n,$$

and the class-prior probability was set to be uniform, we have the following normalization condition:

$$\boldsymbol{\alpha}_y^\top \boldsymbol{K} \mathbf{1}_n = \frac{1}{c}.$$

Furthermore, negative outputs are rounded up to zero to ensure that outputs are non-negative.

Taking these post-processing issues into account, cluster assignment y_i for $\boldsymbol{x}_i$ is determined as the maximizer of the approximation of $p(y|\boldsymbol{x}_i)$:

$$y_i = \operatorname*{argmax}_y \frac{[\max(\mathbf{0}_n, \boldsymbol{K}\widetilde{\boldsymbol{\phi}}_y)]_i}{c\ \max(\mathbf{0}_n, \boldsymbol{K}\widetilde{\boldsymbol{\phi}}_y)^\top \mathbf{1}_n} = \operatorname*{argmax}_y \frac{[\max(\mathbf{0}_n, \widetilde{\boldsymbol{\phi}}_y)]_i}{\max(\mathbf{0}_n, \widetilde{\boldsymbol{\phi}}_y)^\top \mathbf{1}_n},$$

where $\mathbf{0}_n$ denotes the n-dimensional vector with all zeros, the max operation for vectors is applied in the element-wise manner, and $[\cdot]_i$ denotes the i-th element of a vector. Note that $\boldsymbol{K}\widetilde{\boldsymbol{\phi}}_y = \lambda_y \widetilde{\boldsymbol{\phi}}_y$ is used in the above derivation.

For out-of-sample prediction, cluster assignment y' for new sample $\boldsymbol{x}'$ may be obtained as

$$y' := \operatorname*{argmax}_y \frac{\max\left(0, \sum_{i=1}^n K(\boldsymbol{x}', \boldsymbol{x}_i)[\widetilde{\boldsymbol{\phi}}_y]_i\right)}{\lambda_y \max(\mathbf{0}_n, \widetilde{\boldsymbol{\phi}}_y)^\top \mathbf{1}_n}. \tag{7}$$

This clustering algorithm is called the *SMI-based clustering* (SMIC).

SMIC may include a tuning parameter, say θ, in the kernel function, and the clustering results of SMIC depend on the choice of θ. A notable advantage of information-maximization clustering is that such a tuning parameter can be systematically optimized by the same information-maximization principle. More specifically, cluster assignments $\{y_i^\theta\}_{i=1}^n$ are first obtained for each possible θ. Then the quality of clustering is measured by the SMI value estimated from paired samples $\{(\boldsymbol{x}_i, y_i^\theta)\}_{i=1}^n$. For this purpose, the method of *least-squares mutual information* (LSMI) (Suzuki et al., 2009) is useful because LSMI was theoretically proved to be the optimal non-parametric SMI approximator (Suzuki and Sugiyama, 2013); see A for the details of LSMI. Thus, we compute LSMI as a function of θ and the tuning parameter value that maximizes LSMI is selected as the most suitable one:

$$\max_\theta \mathrm{LSMI}(\theta).$$

3 Semi-Supervised SMIC

In this section, we extend SMIC to a semi-supervised clustering scenario where a set of *must-links* and a set of *cannot-links* are provided. A must-link (i, j) means that $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are encouraged to belong to the same cluster, while a cannot-link (i, j) means that $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are encouraged to belong to different clusters. Let $\boldsymbol{M}$ be the must-link matrix with

$M_{i,j} = 1$ if a must-link between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is given and $M_{i,j} = 0$ otherwise. In the same way, we define the cannot-link matrix $\boldsymbol{C}$. We assume that $M_{i,i} = 1$ for all $i = 1, \ldots, n$, and $C_{i,i} = 0$ for all $i = 1, \ldots, n$. Below, we explain how must-link constraints and cannot-link constraints are incorporated into the SMIC formulation.

3.1 Incorporating Must-Links in SMIC

When there exists a must-link between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, we want them to share the same class label. Let

$$\boldsymbol{p}_i^* = (p^*(y = 1|\boldsymbol{x}_i), \ldots, p^*(y = c|\boldsymbol{x}_i))^\top$$

be the soft-response vector for $\boldsymbol{x}_i$. Then the inner product $\langle \boldsymbol{p}_i^*, \boldsymbol{p}_j^* \rangle$ is maximized if and only if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ belong to the same cluster with perfect confidence, i.e., $\boldsymbol{p}_i^*$ and $\boldsymbol{p}_j^*$ are the same vector that commonly has 1 in one element and 0 otherwise. Thus, the must-link information may be utilized by increasing $\langle \boldsymbol{p}_i^*, \boldsymbol{p}_j^* \rangle$ if $M_{i,j} = 1$. We implement this idea as

$$\widehat{\mathrm{SMI}} + \gamma \frac{c}{n} \sum_{i,j=1}^{n} M_{i,j} \sum_{y=1}^{c} p(y|\boldsymbol{x}_i; \boldsymbol{\alpha}) p(y|\boldsymbol{x}_j; \boldsymbol{\alpha}),$$

where $\gamma \geq 0$ determines how strongly we encourage the must-links to be satisfied.

Let us further utilize the following fact: If $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ belong to the same class and $\boldsymbol{x}_j$ and $\boldsymbol{x}_k$ belong to the same class, $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$ also belong to the same class (i.e., a friend's friend is a friend). Letting $M'_{i,j} = \sum_{k=1}^{n} M_{i,k} M_{k,j}$, we can incorporate this in SMIC as

$$\begin{aligned}
&\widehat{\mathrm{SMI}} + \gamma \frac{c}{n} \sum_{i,j=1}^{n} M_{i,j} \sum_{y=1}^{c} p(y|\boldsymbol{x}_i; \boldsymbol{\alpha}) p(y|\boldsymbol{x}_j; \boldsymbol{\alpha}) \\
&\quad + \gamma' \frac{c}{2n} \sum_{i,j=1}^{n} M'_{i,j} \sum_{y=1}^{c} p(y|\boldsymbol{x}_i; \boldsymbol{\alpha}) p(y|\boldsymbol{x}_j; \boldsymbol{\alpha}) \\
&= \frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K}^2 \boldsymbol{\alpha}_y - \frac{1}{2} + \gamma \frac{c}{n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} \boldsymbol{M} \boldsymbol{K} \boldsymbol{\alpha}_y + \gamma' \frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} \boldsymbol{M}^2 \boldsymbol{K} \boldsymbol{\alpha}_y \\
&= \frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} (\boldsymbol{I} + 2\gamma \boldsymbol{M} + \gamma' \boldsymbol{M}^2) \boldsymbol{K} \boldsymbol{\alpha}_y - \frac{1}{2}.
\end{aligned}$$

If we set $\gamma' = \gamma^2$, we have a simpler form:

$$\frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} (\boldsymbol{I} + \gamma \boldsymbol{M})^2 \boldsymbol{K} \boldsymbol{\alpha}_y - \frac{1}{2},$$

which will be used later.

3.2 Incorporating Cannot-Links in SMIC

We may incorporate cannot-links in SMIC in the opposite way to must-links, by decreasing the inner product $\langle \boldsymbol{p}_i^*, \boldsymbol{p}_j^* \rangle$ to zero. This may be implemented as

$$\widehat{\mathrm{SMI}} - \eta \frac{c}{n} \sum_{i,j=1}^{n} C_{i,j} \sum_{y=1}^{c} p(y|\boldsymbol{x}_i;\boldsymbol{\alpha}) p(y|\boldsymbol{x}_j;\boldsymbol{\alpha}), \tag{8}$$

where $\eta \geq 0$ determines how strongly we encourage the cannot-links to be satisfied.

In binary clustering problems where $c = 2$, if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ belong to different classes and $\boldsymbol{x}_j$ and $\boldsymbol{x}_k$ belong to different classes, $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$ actually belong to the same class (i.e., an enemy's enemy is a friend). Let $C'_{i,j} = \sum_{k=1}^{n} C_{i,k} C_{k,j}$, and we will take this also into account as must-links in the following way:

$$\begin{aligned}
&\widehat{\mathrm{SMI}} - \eta \frac{c}{n} \sum_{i,j=1}^{n} C_{i,j} \sum_{y=1}^{c} p(y|\boldsymbol{x}_i;\boldsymbol{\alpha}) p(y|\boldsymbol{x}_j;\boldsymbol{\alpha}) \\
&\quad + \eta' \frac{c}{2n} \sum_{i,j=1}^{n} C'_{i,j} \sum_{y=1}^{c} p(y|\boldsymbol{x}_i;\boldsymbol{\alpha}) p(y|\boldsymbol{x}_j;\boldsymbol{\alpha}) \\
&= \frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} (\boldsymbol{I} - 2\eta \boldsymbol{C} + \eta' \boldsymbol{C}^2) \boldsymbol{K} \boldsymbol{\alpha}_y - \frac{1}{2}.
\end{aligned}$$

If we set $\eta' = \eta^2$, we have

$$\frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} (\boldsymbol{I} - \eta \boldsymbol{C})^2 \boldsymbol{K} \boldsymbol{\alpha}_y - \frac{1}{2},$$

which will be used later.

In our formulation, we do not explicitly impose the density $p(y|\boldsymbol{x}_i;\boldsymbol{\alpha})$ to be non-negative, for computational reasons. Therefore, simply introducing a penalty as in (8) will not work in the maximization problem. When $c = 2$, we can overcome this problem by introducing the term $\sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K} \boldsymbol{C}^2 \boldsymbol{K} \boldsymbol{\alpha}_y$, as described above. However, when $c > 2$, this addition cannot be justified and thus we decided to set η to 0.

3.3 Kernel Matrix Modification

Another approach to incorporating must-links and cannot-links is to modify the kernel matrix $\boldsymbol{K}$. More specifically, $K_{i,j}$ is increased if there exists a must-link between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, and $K_{i,j}$ is decreased if there exists a cannot-link between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. In this paper, we assume $K_{i,j} \in [0, 1]$, and set $K_{i,j} = 1$ if there exists a must-link between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ and $K_{i,j} = 0$ if there exists a cannot-link between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Let us denote the modified kernel matrix by $\boldsymbol{K}'$:

$$\boldsymbol{K}' \longleftarrow \boldsymbol{K}.$$

This modification idea has been employed in spectral clustering (Kamvar et al., 2003) and demonstrated to be promising.

3.4 Semi-Supervised SMIC

Finally, we combine the above three ideas as

$$\widetilde{\mathrm{SMI}} := \frac{c}{2n}\sum_{y=1}^{c}\boldsymbol{\alpha}_y^\top \boldsymbol{U}\boldsymbol{\alpha}_y - \frac{1}{2},$$

where

$$\boldsymbol{U} := \boldsymbol{K}'(2\boldsymbol{I} + 2\gamma\boldsymbol{M} + \gamma^2\boldsymbol{M}^2 - 2\eta\boldsymbol{C} + \eta^2\boldsymbol{C}^2)\boldsymbol{K}' \tag{9}$$

When $c > 2$, we fix η at zero.

This is the learning criterion of *semi-supervised SMIC* (3SMIC), whose global maximizer can be analytically obtained under orthonormality of $\{\boldsymbol{\alpha}_y\}_{y=1}^c$ by the leading eigenvectors of $\boldsymbol{U}$. Then the same post-processing as the original SMIC is applied and cluster assignments are obtained. Out-of-sample prediction is also possible in the same way as the original SMIC.

3.5 Tuning Parameter Optimization in 3SMIC

In the original SMIC, an SMI approximator called LSMI is used for tuning parameter optimization (see A). However, this is not suitable in semi-supervised scenarios because the 3SMIC solution is biased to satisfy must-links and cannot-links. Here, we propose using

$$\max_{\theta} \mathrm{LSMI}(\theta) + \mathrm{Penalty}(\theta),$$

where θ indicates tuning parameters in 3SMIC; in the experiments, γ, η, and the parameter t included in the kernel function $K(\boldsymbol{x}, \boldsymbol{x}')$ is optimized. "Penalty" is the penalty for violating must-links and cannot-links, which is the only tuning factor in the proposed algorithm.

4 Experiments

In this section, we experimentally evaluate the performance of the proposed 3SMIC method in comparison with popular semi-supervised clustering methods: *Spectral Learning* (SL) (Kamvar et al., 2003) and *Constrained Spectral Clustering* (CSC) (Wang and Davidson, 2010). Both methods first perform semi-supervised spectral embedding and then k-means to obtain clustering results. However, we observed that the post k-means step is often unreliable, so we decided to use simple thresholding (Shi and Malik, 2000) in the case of binary clustering for CSC.

In all experiments, we will use a sparse version of the *local-scaling kernel* (Zelnik-Manor and Perona, 2005) as the similarity measure:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \begin{cases} \exp\left(-\dfrac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma_i\sigma_j}\right) & \text{if } \boldsymbol{x}_i \in \mathcal{N}_t(\boldsymbol{x}_j) \text{ or } \boldsymbol{x}_j \in \mathcal{N}_t(\boldsymbol{x}_i), \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathcal{N}_t(\boldsymbol{x})$ denotes the set of t nearest neighbors for $\boldsymbol{x}$ (t is the kernel parameter), σ_i is a local scaling factor defined as $\sigma_i = \|\boldsymbol{x}_i - \boldsymbol{x}_i^{(t)}\|$, and $\boldsymbol{x}_i^{(t)}$ is the t-th nearest neighbor of $\boldsymbol{x}_i$. For SL and CSC, we test $t = 1, 4, 7, 10$ (note that there is no systematic way to choose the value of t), except for the **spam** dataset with $t = 1$ that caused numerical problems in the eigensolver when testing SL. On the other hand, in 3SMIC, we choose the value of t from $\{1, \ldots, 10\}$ based on the following criterion:

$$\frac{\mathrm{LSMI}(\theta)}{\max_\theta \mathrm{LSMI}(\theta)} - \frac{n_v}{\max_\theta(n_v)}, \tag{10}$$

where n_v is the number of violated links. Here, both the LSMI value and the penalty are normalized so that they fall into the range $[0, 1]$. The γ and η parameters in 3SMIC are also chosen based on Eq.(10).

We use the following real-world datasets:

parkinson ($d = 22$, $n = 195$, and $c = 2$): The UCI dataset consisting of voice registrations from patients affected by Parkinson's disease and other individuals that were not affected. From the voice, 22 features are extracted.

spam ($d = 57$, $n = 4601$, and $c = 2$): The UCI dataset consisting of e-mails, categorized in spam and non-spam. 48 word-frequency features and 9 other frequency features such as specific characters and capitalization are extracted.

sonar ($d = 60$, $n = 208$, and $c = 2$): The UCI dataset consisting of sonar responses from a metal object or a rock. The features represent energy in each frequency band.

digits500 ($d = 256$, $n = 500$, and $c = 10$): The USPS digits dataset consisting of images of written numbers from 0 to 9, 256 (16×16) pixels in gray-scale. We randomly sampled 50 images for each digit class, and normalized each pixel intensity in the image between -1 and 1.

digits5k ($d = 256$, $n = 5000$, and $c = 10$): The same USPS digits dataset but with 500 images for each class.

faces100 ($d = 4096$, $n = 100$, and $c = 10$): The Olivetti Face dataset consisting of images of human faces in gray-scale, 4096 (64×64) pixels. We randomly selected 10 persons, and used 10 images for each person.

We evaluate the clustering performance by the *Adjusted Rand Index* (ARI) (Hubert and Arabie, 1985) between learned and true labels. Larger ARI values mean better clustering performance, and the zero ARI value means that the clustering result is equivalent to random. We investigate the ARI score as functions of the number of links used. The x-axis reports the number of links provided to the various algorithms, as a percentage of the total number of possible links.

In our first setting, must-links and cannot-links are generated from the true labels, by randomly selecting two samples (x_i, x_j) and adding the corresponding 1 to the $\boldsymbol{M}$ or $\boldsymbol{C}$ matrices depending on the labels of the chosen pair of points. CSC is excluded from **digits5k** and **spam** because it needs to solve the complete eigenvalue problem and its computational cost was too high on these large datasets. Averages and standard deviations of ARI over 20 runs with different random seeds are plotted in Figure 1.

For **digits500**, **digits5k**, and **faces100**, the baseline performances without links are reasonable, and the introduction of links significantly increases the performance, bringing it around 0.9–0.95 from 0.5–0.8.

For **parkinson**, **spam**, and **sonar** where the baseline performances without links are poor, introduction of links quickly allows the clustering algorithms to find better solutions. In particular, only 6% of links (relative to all possible pairs) was sufficient for **parkinson** to achieve reasonable performance and surprisingly only 0.2% for **spam**.

As shown in Figure 1, the performance of SL depends heavily on the choice of t, but there is no systematic way to choose t for SL. It is important to notice that 3SMIC with t chosen systematically based on Eq.(10) performs as good as SL with t tuned optimally with hindsight. On the other hand, CSC performs rather stably for different values of t, and it works particularly well for binary problems with a small number of links. However, it performs very poorly for multi-class problems; we observed that the post k-means step is highly unreliable and poor local optimal solutions are often produced. For the binary problems, simply performing thresholding (Shi and Malik, 2000) instead of using k-means was found to be useful. However, there seem no simple alternatives in multi-class cases. The performance of CSC drops in **parkinson** and **sonar** when the number of links is increased, although such phenomena were not observed in SL and 3SMIC.

The random selection of must-links and cannot-links shows significant increases in performance compared to the unsupervised problem. But, although random selection has been widely used in the previous literature, other sampling strategies could also be considered. For example from a labeled subset of samples, a fully connected network of must-links and cannot-links can be extracted. In Figure 2, we report results for this sampling strategy, where, instead of randomly selecting a pair of samples and connecting them with the appropriate link, we selected a random subset of samples, and introduced all the links among the subset members. We encountered many numerical problems for CSC with $t = 1$, so it was excluded from the comparison.

As we can see, this different link selection strategy is much less informative, and the number of links required for the information to propagate and influence the final decision is much higher. Using a much larger number of links, the drawbacks that CSC had in **parkinson** and **sonar** worsen, but the performance improves for **digits500** and **faces100**

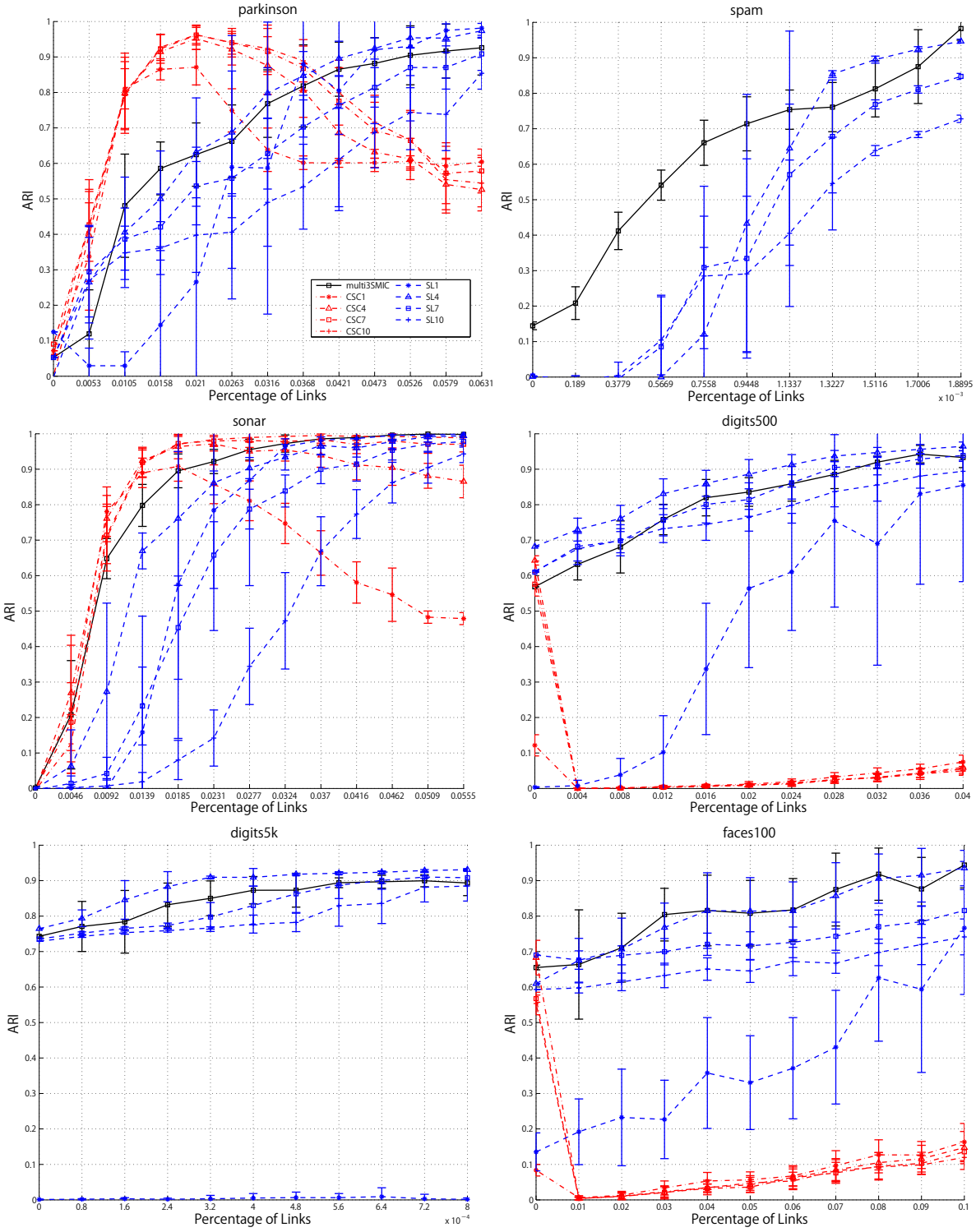

Figure 1: Experimental results for randomly selected links.

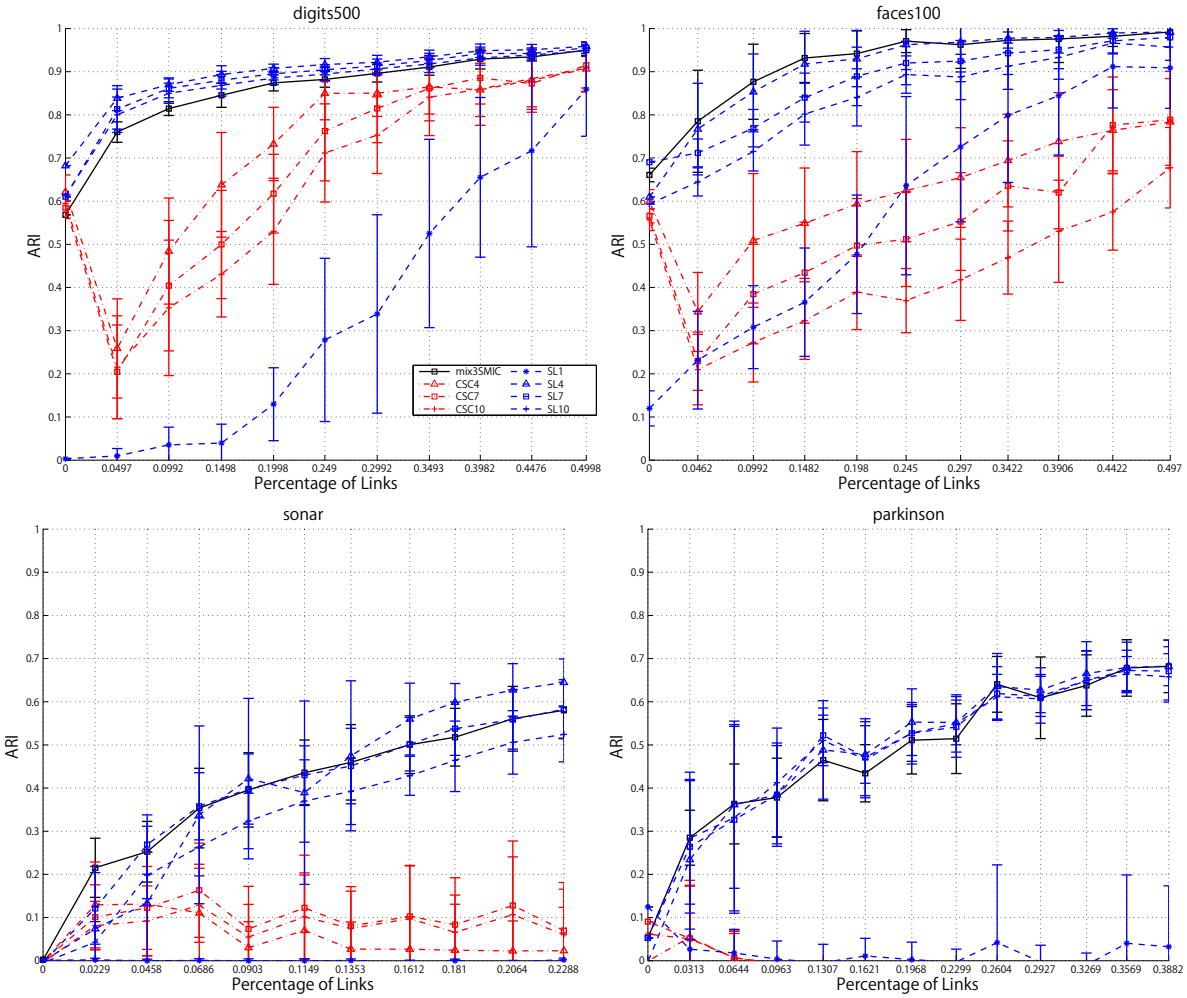

Figure 2: Experimental results for fully connected, randomly selected subset.

for the largest numbers of links, that were not observed in the previous experiments. The model selection property of 3SMIC still gives clear advantages.

In particular, to investigate the contribution of the increasing number of links on the solutions provided by 3SMIC, we introduce a final experiment reported in Figure 3. In the left column, "3SMIC*" represents a "cheating" version of 3SMIC that selects the model based on the true labels so that ARI is maximized. As we can see, as the number of samples increases, ordinary 3SMIC correctly starts to recognize the most suitable model, providing an increase in performance. At the same time, in the right columns we provide performances of 3SMIC for various candidate models, without any model selection. As we can see, the quality of each candidate improves when the number of samples is increased, and the best candidate changes depending on the number of samples. This would highlight why model selection in our methods improves the performance. Thus, we conclude that both the quality of each solution and the precision of selecting the best solution contribute to improving the performance.

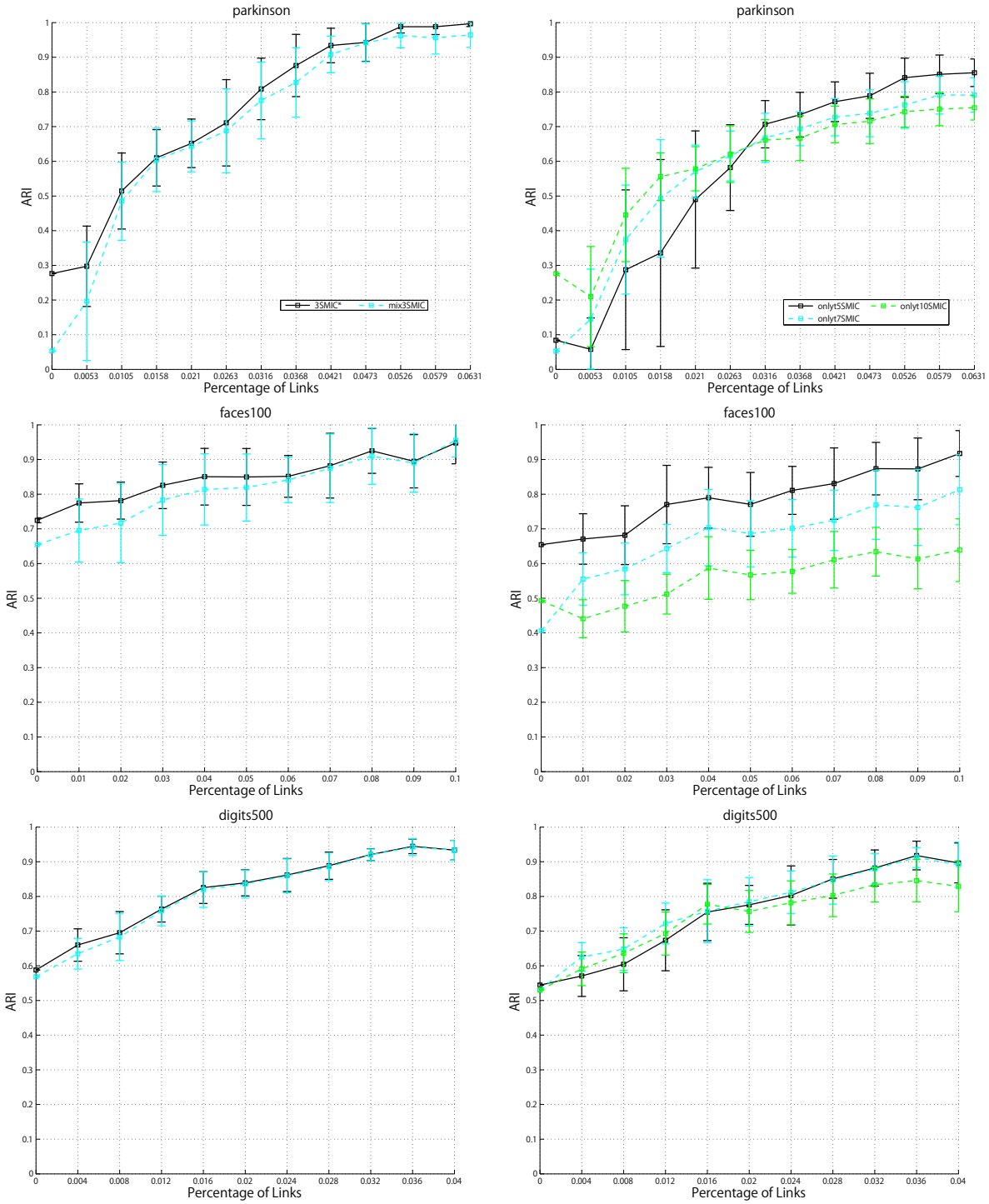

Figure 3: Contribution to performance improvement.

Overall, the proposed 3SMIC method was shown to be a promising semi-supervised clustering method.

5 Conclusions

In this paper, we proposed a novel information-maximization clustering method that can utilize side information provided as must-links and cannot-links. The proposed method, named *semi-supervised SMI-based clustering* (3SMIC), allows us to compute the clustering solution analytically. This is a strong advantage over conventional approaches such as *constrained spectral clustering* (CSC) that requires a post k-means step, because this post k-means step can be unreliable and cause significant performance degradation in practice. Furthermore, 3SMIC allows us to systematically determine tuning parameters such as the kernel width based on the information-maximization principle, given our reliance on the provided side information. Through experiments, we demonstrated that automatically-tuned 3SMIC performs as good as optimally-tuned *spectral learning* (SL) with hindsight.

The focus of our method in this paper was to inherit the analytical treatment of the original unsupervised SMIC in semi-supervised learning scenarios. Although this analytical treatment was demonstrated to be highly useful in experiments, our future work will explore more efficient use of must-links and cannot-links.

In the previous work (Laub and Müller, 2004), negative eigenvalues were found to contain useful information. Because must-link and cannot-link matrices can possess negative eigenvalues, it is interesting to investigate the role and effect of negative eigenvalues in the context of information-maximization clustering.

Another interesting line of work is extending the 3SMIC penalty to be able to perform not only model selection, but also to give indication on the selection of the number of clusters. One of the main challenges that makes this extension not trivial is that, as discussed in the original SMIC paper Sugiyama et al. (2014), SMI is monotone increasing as the number of clusters c grows. A principled method to choose the number of clusters based on SMI is therefore not clear. In the semi-supervised setting, the expert that provides the links could also have enough insight into giving a sensible choice of the number of clusters. If this was not possible, an accurate balance between the SMI score and the number of violations might be an alternative for this additional model selection task.

Acknowledgements

This work was carried out when DC was visiting at Tokyo Institute of Technology by the YSEP program. GN was supported by the MEXT scholarship and the FIRST program, and MS was supported by MEXT KAKENHI 25700022 and AOARD.

A Least-Squares Mutual Information

The solution of SMIC depends on the choice of the kernel parameter included in the kernel function $K(\boldsymbol{x}, \boldsymbol{x}')$. Since SMIC was developed in the framework of SMI maximization, it would be natural to determine the kernel parameter so as to maximize SMI. A direct approach is to use the SMI estimator $\widehat{\mathrm{SMI}}$ given by Eq.(6) also for kernel parameter choice. However, this direct approach is not favorable because $\widehat{\mathrm{SMI}}$ is an unsupervised SMI estimator (i.e., SMI is estimated only from unlabeled samples $\{\boldsymbol{x}_i\}_{i=1}^n$). On the other hand, in the model selection stage, we have already obtained labeled samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, and thus supervised estimation of SMI is possible. For supervised SMI estimation, a non-parametric SMI estimator called *least-squares mutual information* (LSMI) (Suzuki et al., 2009) was proved to achieve the optimal convergence rate to the true SMI. Here we briefly review LSMI.

The key idea of LSMI is to learn the following *density-ratio function* (Sugiyama et al., 2012),

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

without going through probability density/mass estimation of $p^*(\boldsymbol{x}, y)$, $p^*(\boldsymbol{x})$, and $p^*(y)$. More specifically, let us employ the following density-ratio model:

$$r(\boldsymbol{x}, y; \boldsymbol{\omega}) := \sum_{\ell: y_\ell = y} \omega_\ell L(\boldsymbol{x}, \boldsymbol{x}_\ell), \tag{11}$$

where $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)^\top$ and $L(\boldsymbol{x}, \boldsymbol{x}')$ is a kernel function. In practice, we use the Gaussian kernel

$$L(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\kappa^2}\right),$$

where the Gaussian width κ is the kernel parameter. To save the computation cost, we limit the number of kernel bases to 500 with randomly selected kernel centers.

The parameter $\boldsymbol{\omega}$ in the above density-ratio model is learned so that the following squared error is minimized:

$$\min_{\boldsymbol{\omega}} \frac{1}{2} \int \sum_{y=1}^{c} \Big(r(\boldsymbol{x}, y; \boldsymbol{\omega}) - r^*(\boldsymbol{x}, y)\Big)^2 p^*(\boldsymbol{x}) p^*(y) \mathrm{d}\boldsymbol{x}. \tag{12}$$

Let $\boldsymbol{\omega}^{(y)}$ be the parameter vector corresponding to the kernel bases $\{L(\boldsymbol{x}, \boldsymbol{x}_\ell)\}_{\ell: y_\ell = y}$, i.e., $\boldsymbol{\omega}^{(y)}$ is the sub-vector of $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)^\top$ consisting of indices $\{\ell \mid y_\ell = y\}$. Let n_y be the number of samples in class y, which is the same as the dimensionality of $\boldsymbol{\omega}^{(y)}$. Then an empirical and regularized version of the optimization problem (12) is given for each y as follows:

$$\min_{\boldsymbol{\omega}^{(y)}} \left[\frac{1}{2}\boldsymbol{\omega}^{(y)\top}\widehat{\boldsymbol{H}}^{(y)}\boldsymbol{\omega}^{(y)} - \boldsymbol{\omega}^{(y)\top}\widehat{\boldsymbol{h}}^{(y)} + \frac{\delta}{2}\boldsymbol{\omega}^{(y)\top}\boldsymbol{\omega}^{(y)}\right], \tag{13}$$

where δ (≥ 0) is the regularization parameter. $\widehat{\boldsymbol{H}}^{(y)}$ is the $n_y \times n_y$ matrix and $\widehat{\boldsymbol{h}}^{(y)}$ is the n_y-dimensional vector defined as

$$\widehat{H}_{\ell,\ell'}^{(y)} := \frac{n_y}{n^2} \sum_{i=1}^{n} L(\boldsymbol{x}_i, \boldsymbol{x}_\ell^{(y)}) L(\boldsymbol{x}_i, \boldsymbol{x}_{\ell'}^{(y)}), \quad \widehat{h}_\ell^{(y)} := \frac{1}{n} \sum_{i:y_i=y} L(\boldsymbol{x}_i, \boldsymbol{x}_\ell^{(y)}),$$

where $\boldsymbol{x}_\ell^{(y)}$ is the ℓ-th sample in class y (which corresponds to $\widehat{\omega}_\ell^{(y)}$).

A notable advantage of LSMI is that the solution $\widehat{\boldsymbol{\omega}}^{(y)}$ can be computed analytically as

$$\widehat{\boldsymbol{\omega}}^{(y)} = (\widehat{\boldsymbol{H}}^{(y)} + \delta \boldsymbol{I})^{-1} \widehat{\boldsymbol{h}}^{(y)}.$$

Then a density-ratio estimator is obtained analytically as follows:

$$\widehat{r}(\boldsymbol{x}, y) = \sum_{\ell=1}^{n_y} \widehat{\omega}_\ell^{(y)} L(\boldsymbol{x}, \boldsymbol{x}_\ell^{(y)}).$$

The accuracy of the above least-squares density-ratio estimator depends on the choice of the kernel parameter κ included in $L(\boldsymbol{x}, \boldsymbol{x}')$ and the regularization parameter δ in Eq.(13). These tuning parameter values can be systematically optimized based on cross-validation as follows: First, the samples $\mathcal{Z} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ are divided into M disjoint subsets $\{\mathcal{Z}_m\}_{m=1}^M$ of approximately the same size (we use $M = 5$ in the experiments). Then a density-ratio estimator $\widehat{r}_m(\boldsymbol{x}, y)$ is obtained using $\mathcal{Z} \backslash \mathcal{Z}_m$ (i.e., all samples without $\mathcal{Z}_m$), and its out-of-sample error (which corresponds to Eq.(12) without irrelevant constant) for the hold-out samples $\mathcal{Z}_m$ is computed as

$$\mathrm{CV}_m := \frac{1}{2|\mathcal{Z}_m|^2} \sum_{\boldsymbol{x},y \in \mathcal{Z}_m} \widehat{r}_m(\boldsymbol{x}, y)^2 - \frac{1}{|\mathcal{Z}_m|} \sum_{(\boldsymbol{x},y) \in \mathcal{Z}_m} \widehat{r}_m(\boldsymbol{x}, y),$$

where $\sum_{\boldsymbol{x},y \in \mathcal{Z}_m}$ denotes the summation over all combinations of $\boldsymbol{x}$ and y in $\mathcal{Z}_m$ (and thus $|\mathcal{Z}_m|^2$ terms), while $\sum_{(\boldsymbol{x},y) \in \mathcal{Z}_m}$ denotes the summation over all pairs $(\boldsymbol{x}, y)$ in $\mathcal{Z}_m$ (and thus $|\mathcal{Z}_m|$ terms). This procedure is repeated for $m = 1, \ldots, M$, and the average of the above hold-out error over all m is computed as

$$\mathrm{CV} := \frac{1}{M} \sum_{m=1}^{M} \mathrm{CV}_m.$$

Then the kernel parameter κ and the regularization parameter δ that minimize the average hold-out error CV are chosen as the most suitable ones.

Finally, given that SMI (2) can be expressed as

$$\mathrm{SMI} = -\frac{1}{2} \int \sum_{y=1}^{c} r^*(\boldsymbol{x}, y)^2 p^*(\boldsymbol{x}) p^*(y) \mathrm{d}\boldsymbol{x} + \int \sum_{y=1}^{c} r^*(\boldsymbol{x}, y) p^*(\boldsymbol{x}, y) \mathrm{d}\boldsymbol{x} - \frac{1}{2},$$

an SMI estimator based on the above density-ratio estimator, called *least-squares mutual information* (LSMI), is given as follows:

$$\mathrm{LSMI} := -\frac{1}{2n^2}\sum_{i,j=1}^{n}\widehat{r}(\boldsymbol{x}_i, y_j)^2 + \frac{1}{n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i, y_i) - \frac{1}{2},$$

where $\widehat{r}(\boldsymbol{x}, y)$ is a density-ratio estimator obtained above.

References

F. Agakov and D. Barber. Kernelized infomax clustering. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 17–24, Cambridge, MA, USA, 2006. MIT Press.

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

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

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

M. Girolami. Mercer kernel-based clustering in feature space. *IEEE Transactions on Neural Networks*, 13(3):780–784, 2002.

A. B. Goldberg. Dissimilarity in graph-based semi-supervised classification. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS2007)*, pages 155–162, 2007.

R. Gomes, A. Krause, and P. Perona. Discriminative clustering by regularized information maximization. In J. Lafferty, C. K. I. Williams, R. Zemel, J. Shawe-Taylor, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 766–774, 2010.

L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

S. D. Kamvar, D. Klein, and C. D. Manning. Spectral learning. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI2003)*, pages 561–566, 2003.

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

J. Laub and K.-R. Müller. Feature discovery in non-metric pairwise data. *Journal of Machine Learning Research*, 5:801–818, Jul. 2004.

J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, Berkeley, CA, USA, 1967. University of California Press.

A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 849–856, Cambridge, MA, USA, 2002. MIT Press.

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

B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, USA, 2002.

C. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 1948.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

M. Sugiyama. Machine learning with squared-loss mutual information. *Entropy*, 15(1): 80–112, 2013.

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

M. Sugiyama, G. Niu, M. Yamada, M. Kimura, and H. Hachiya. Information-maximization clustering based on squared-loss mutual information. *Neural Computation*, 26(1):84–131, 2014.

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

T. Suzuki, M. Sugiyama, T. Kanamori, and J. Sese. Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, 10 (1):S52 (12 pages), 2009.

K. Wagstaff and C. Cardie. Clustering with instance-level constraints. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML2000)*, pages 1103–1110, 2000.

K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl. Constrained k-means clustering with background knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML2001)*, pages 577–584, 2001.

X. Wang and I. Davidson. Flexible constrained spectral clustering. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD2010)*, pages 563–572, 2010.

L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1601–1608, Cambridge, MA, USA, 2005. MIT Press.