International Journal of Innovative
Computing, Information and Control
Volume **14**, Number **6**, December **2018**

 ISSN 1349-4198
pp. **2343–2351**

DISTANCE METRIC LEARNING FOR THE SELF-ORGANIZING MAP USING A CO-TRAINING APPROACH

Keisuke Yoneda and Tetsuo Furukawa

Department of Human Intelligence Systems
Kyushu Institute of Technology
2-4 Hibikino, Wakamatsu-ku, Kitakyushu 808-0196, Japan
p899037k@mail.kyutech.jp; furukawa@brain.kyutech.ac.jp

Received April 2018; revised August 2018

Abstract. *The aim of this work is to develop a method of distance metric learning for self-organizing maps. We first conducted an investigation in a multi-view learning setting, in which Mahalanobis metrics were determined so that two (or more) views reached a consensus in latent variable estimation. We examined two approaches of multi-view learning: co-training and ensemble. Although both approaches worked as expected, our results suggested that the co-training approach performed better. We further extended the method to a single-view learning setting by introducing the concept of pseudo multi-view learning.*
Keywords: Metric learning, Multi-view learning, Self-organizing map, Co-training

1. **Introduction.** The distance (or similarity) metric is crucially important in most machine learning methods, and measures the differences (or similarities) between data points, or between data and a model. In the case of the self-organizing map (SOM), the distance metric is involved in determining the winning nodes, and the Euclidean distance is typically used, sometimes without considering the validity. If the metric is changed, then the winning nodes are determined differently, and as a result, the SOM generates a different map. Therefore, using appropriate distance metrics is important for obtaining a reliable result.

Metric learning is an area of machine learning studies that aims to estimate an appropriate metric for learning tasks [17, 25, 28]. The aim of this work is to develop a method of metric learning in manifold modeling, particularly in SOMs. Thus, the task of the proposed method is to estimate an appropriate metric as the Mahalanobis distance, in parallel with modeling the dataset using a nonlinear manifold. Our main issue is how to estimate the appropriate metric in an unsupervised manner. In unsupervised learning, not only is the desired output not given but also the criteria for determining the desired metrics. Therefore, metric learning in unsupervised learning, such as the SOM, is a challenging issue. Most past works on metric learning in the SOM required auxiliary information, such as labels [12, 13, 14, 20] or episodes [16]. In this work, we propose a method of unsupervised metric learning without using auxiliary information.

To manage this problem, we adopt a method of multi-view learning. When data vectors can be naturally divided into several perspectives, which are called *views*, the metrics of the views can be determined so that they produce as similar results as possible; that is, by regarding the learning result of a view as the desired output for other views, it is possible to change the unsupervised task into a supervised task. As a result, we can obtain a compromise of multiple views, that is, a *consensus*. Such an approach is referred to as

DOI: 10.24507/ijicic.14.06.2343

co-training in the area of multi-view learning [23, 27]. Thus, the first aim of this study is to develop a method for distance metric learning of the SOM under multi-view settings using the co-training approach. Because the method is a nonlinear extension of canonical correlation analysis (CCA), it is referred to as *CCA-SOM* in this paper. Then, we extend CCA-SOM for single-view cases in which views are not defined, which is referred to as *metric learning SOM (ML-SOM)*. This is the second aim of this study. For this purpose, we introduce the concept of a *pseudo multi-view*, in which data components are divided into views randomly. Interestingly, the approach of ML-SOM has many similarities to the dropout method in deep learning [10, 22].

This paper is organized as follows. In Section 2, related works are introduced. In Section 3, the theory and algorithms of the proposed methods are described, and the simulation results are shown in Section 4. In Sections 5 and 6, the discussion and conclusion are presented.

2. Related Works.

2.1. Distance metric learning for SOM.

Because the SOM is an unsupervised learning algorithm, metric learning for the SOM is not an easy task. Most existence methods have been developed so that some auxiliary data are provided [12, 13, 14, 20]; that is, under the scenarios in which labels are given to all or part of the data, these methods estimate the metrics so that data with the same/different labels are mapped closely/apart. Therefore, these are classified as semi-supervised learning with respect to metric learning.

Another type of metric learning for the SOM is the adaptive subspace SOM (ASSOM) [16]. In the ASSOM, multiple data are assumed to be observed from each target object. Such a subset of data obtained from the same target is called an *episode* in the ASSOM. The idea of the ASSOM has been further extended to nonlinear metrics [7]. In these cases, it is possible to estimate the appropriate metric by regarding the episodes as auxiliary information.

More recently, Arnonkijpanich et al. proposed a method of metric learning for the SOM [2], which does not require auxiliary information. They generalized the objective function of the SOM and derived the algorithm theoretically. We refer to this type of metric learning as an *ensemble* approach. In this study, we use a different type of approach to metric learning, that is, *co-training*. In contrast to the ensemble approach, the co-training approach intentionally collapses the objective function, thereby avoiding undesirable local optima caused by a single objective function.

2.2. Multi-view learning of subspace methods.

Multi-view learning is another paradigm of machine learning [23, 27]. It aims to improve learning performance by integrating datasets observed from different views. Such integration is often led by *consensus*, which represents common factors that are consistent among the views. To obtain a consensus, one of the representative approaches is *co-training* [3]. For the V-view dataset, the co-training method uses V-learners, each of which learns the data of the corresponding view. These learners are trained so that discrepancies between V outputs are minimized. By contrast, a conventional approach that maximizes the joint likelihood of views is also used in multi-view learning, particularly in Bayesian methods. In this paper, we refer to this as an *ensemble* approach. Both co-training and ensemble approaches are examined in this work.

Among unsupervised multi-view learning approaches, multi-view subspace methods aim to estimate the subspace shared by the views as the consensus. CCA is the most basic and representative method, and estimates the projections that maximize the correlation between two views [11]. In CCA, the projections can be regarded as the metrics of the

views. Extensions of CCA have been proposed, such as Bayesian CCA [15] and kernel CCA [1]. For nonlinear subspace methods, the Gaussian process latent variable model (GPLVM) [18] has been extended for multi-view learning, such as shared GPLVM (sGPLVM) [6], manifold relevance determination [5], and factorized orthogonal latent spaces [21]. However, unlike CCA, they are not designed to estimate the metrics. Additionally, it would not be easy to extend these methods to metric learning because they assume the identical and independent distribution (i.i.d.) of data components. Therefore, it would be difficult to use these methods for our case. It should be also noted that GPLVM-based methods are classified as the ensemble approach because they are formulated under Bayesian settings.

In this work, we use the generalized SOM for manifold modeling. Unlike GPLVM, SOM allows the use of both co-training and ensemble approaches. Additionally, the SOM does not assume the i.i.d. of data components. Therefore, the SOM seems to be a good platform for studies on both multi-view learning and metric learning. It should be emphasized that our purpose is not only to develop metric learning for SOMs but also a general principle of metric learning in manifold modeling.

3. Theory and Algorithms.

In this section, we first describe metric learning methods of the SOM for a multi-view dataset: CCA-SOM. Our algorithm is based on the theoretical generalization of the SOM [4, 8, 9, 19, 24]. Thus, it is a continuous manifold modeling method based on the kernel smoothing approach rather than the conventional topology-preserving clustering method. We compare two types of CCA-SOM algorithms based on ensemble and co-training approaches. Then, we propose metric learning methods of the SOM for a single-view dataset: ML-SOM. We also compare two types of ML-SOM based on ensemble and co-training approaches.

3.1. Problem formulation.

Suppose that we have N target objects. In the case of a multi-view setting, they are observed from V views. Let $\mathbf{x}_n^{(v)} \in \mathcal{X}^{(v)} \equiv \mathbb{R}^{D^{(v)}}$ be the data of the n-th object measured from the v-th view. All the data of the n-th object is denoted by $\mathbf{x}_n := \bigoplus_{v=1}^{V} \mathbf{x}_n^{(v)}$. Let $\mathcal{Z}$ be the low-dimensional latent space and let $\mathbf{z}_n$ be the latent variable of the n-th object. In this paper, $\mathcal{Z}$ is a unit square space, that is, $\mathcal{Z} = [0,1]^L$. The aim of CCA-SOM is to model the data distribution as $\mathbf{x}_n^{(v)} \Big| \mathbf{z}_n \sim \mathcal{N}\left(f^{(v)}(\mathbf{z}_n), \mathbf{M}^{(v)^{-1}}\right)$, where $f^{(v)} : \mathcal{Z} \to \mathcal{X}^{(v)}$ is a smooth embedding, and $\mathbf{M}^{(v)}$ is the precision matrix, which represents the metric of the v-th view. The prior distribution of $\mathbf{z}$ is assumed to be uniform over $\mathcal{Z}$; thus, $p(\mathbf{z}) = 1$. Thus, the task of CCA-SOM is to estimate $\left\{\mathbf{z}_n\right\}_{n=1}^{N}$ and $\left\{f^{(v)}\right\}_{v=1}^{V}$ from $\left\{\mathbf{x}_n\right\}_{n=1}^{N}$, in addition to estimating the metrics $\left\{\mathbf{M}^{(v)}\right\}_{v=1}^{V}$.

By contrast, in the single-view setting, we have a single dataset only. Thus, the task of ML-SOM is to estimate $\left\{\mathbf{z}_n\right\}_{n=1}^{N}$, $f : \mathcal{Z} \to \mathcal{X}$, and $\mathbf{M}^{(v)}$ from $\left\{\mathbf{x}_n\right\}_{n=1}^{N}$.

3.2. Objective functions.

Many preceding works have shown that the objective function of the SOM is given by

$$\mathcal{L}(\mathbf{Z}, f) = -\frac{1}{2} \sum_{n=1}^{N} \int_{\mathcal{Z}} h(\boldsymbol{\zeta}, \mathbf{z}_n) \left\|\mathbf{x}_n - f(\boldsymbol{\zeta})\right\|^2 d\boldsymbol{\zeta}, \tag{1}$$

where $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)$ and $h(\boldsymbol{\zeta}, \mathbf{z})$ is the neighborhood function [4, 8, 9, 19, 24]. This objective function represents the log-likelihood when the posterior of $\mathbf{z}_n$ is given by the neighborhood function. Thus, we assume that $\int_{\mathcal{Z}} h(\boldsymbol{\zeta}, \mathbf{z})\, d\boldsymbol{\zeta} = 1$ without loss of generality.

This objective function can be easily extended for V-view data with the Mahalanobis distance as

$$\begin{aligned}&\mathcal{L}\left(\mathbf{Z}, \left\{f^{(v)}\right\}, \left\{\mathbf{M}^{(v)}\right\}\right)\\ &= \sum_{v=1}^{V}\sum_{n=1}^{N}\int_{\mathcal{Z}} h(\boldsymbol{\zeta}, \mathbf{z}_n)\left[-\frac{1}{2}\left\|\mathbf{x}_n^{(v)} - f^{(v)}(\boldsymbol{\zeta})\right\|^2_{\mathbf{M}^{(v)}} + \frac{1}{2}\ln\left|\det \mathbf{M}^{(v)}\right|\right] d\boldsymbol{\zeta}. \end{aligned} \tag{2}$$

$\|\cdot\|_{\mathbf{M}^{(v)}}$ is the norm of the v-th view under metric $\mathbf{M}^{(v)}$ defined by $\left\|\mathbf{x}^{(v)}\right\|^2_{\mathbf{M}^{(v)}} := \mathbf{x}^{(v)\mathrm{T}}\mathbf{M}^{(v)}\mathbf{x}^{(v)}$. This is the objective function of CCA-SOM with the ensemble approach. If $V = 1$, then Equation (2) becomes the objective function of ML-SOM with the ensemble approach.

For CCA-SOM with the co-training approach, we propose the following objective functions by modifying Equation (2). To estimate $\{\mathbf{z}_n\}$ and $\left\{f^{(v)}\right\}$, the objective function is given by

$$\mathcal{L}_{\mathbf{Z},f}\left(\mathbf{Z}, \left\{f^{(v)}\right\} \mid \left\{\mathbf{M}^{(v)}\right\}\right) = \sum_{v=1}^{V}\sum_{n=1}^{N}\int_{\mathcal{Z}} h(\boldsymbol{\zeta}, \mathbf{z}_n)\left[-\frac{1}{2}\left\|\mathbf{x}_n^{(v)} - f^{(v)}(\boldsymbol{\zeta})\right\|^2_{\mathbf{M}^{(v)}}\right] d\boldsymbol{\zeta} \tag{3}$$

under that the metrics $\left\{\mathbf{M}^{(v)}\right\}$ are given. Thus, there is no difference from the ensemble approach. By contrast, the objective function for metric learning is given by

$$\begin{aligned}&\mathcal{L}_{\mathbf{M}}\left(\left\{\mathbf{M}^{(v)}\right\} \mid \mathbf{Z}^{(-v)}, f^{(v)}\right)\\ &= \sum_{v=1}^{V}\sum_{n=1}^{N}\int_{\mathcal{Z}} h\left(\boldsymbol{\zeta}, \mathbf{z}_n^{(-v)}\right)\left[-\frac{1}{2}\left\|\mathbf{x}_n^{(v)} - f^{(v)}(\boldsymbol{\zeta})\right\|^2_{\mathbf{M}^{(v)}} + \frac{1}{2}\ln\left|\det \mathbf{M}^{(v)}\right|\right] d\boldsymbol{\zeta}, \end{aligned} \tag{4}$$

where $\mathbf{z}_n^{(-v)}$ is the latent variable estimated without using the v-th view. Thus, $\mathbf{z}_n^{(-v)}$ is given by minimizing Equation (3) without using the v-th view. Unlike the ensemble approach, the objective functions of the co-training approach cannot be integrated into a single function. This objective function is also applied to ML-SOM with the co-training approach by introducing the pseudo multi-view.

3.3. **Algorithm for CCA-SOM with the ensemble approach.** In the ensemble approach, the objective function (2) is optimized alternately for $\mathbf{Z}$, $\left\{f^{(v)}\right\}$, and $\left\{\mathbf{M}^{(v)}\right\}$. Like the ordinary SOM, $\{\mathbf{z}_n\}$ and $\left\{f^{(v)}\right\}$ are updated as follows:

$$\mathbf{z}_n = \arg\min_{\mathbf{z}} \sum_{v} \left\|\mathbf{x}_n^{(v)} - f^{(v)}(\mathbf{z})\right\|^2_{\mathbf{M}^{(v)}} \tag{5}$$

$$f^{(v)}(\mathbf{z}) = \frac{1}{H(\mathbf{z})}\sum_{n=1}^{N} h(\mathbf{z}, \mathbf{z}_n)\,\mathbf{x}_n^{(v)}, \tag{6}$$

where $H(\mathbf{z}) = \sum_n h(\mathbf{z}, \mathbf{z}_n)$. In (5), we used the following approximation:

$$\arg\min_{\mathbf{z}} \sum_{v}\int_{\mathcal{Z}} h(\boldsymbol{\zeta}, \mathbf{z})\left\|\mathbf{x}_n^{(v)} - f^{(v)}(\boldsymbol{\zeta})\right\|^2_{\mathbf{M}^{(v)}} d\boldsymbol{\zeta} \sim \arg\min_{\mathbf{z}} \sum_{v}\left\|\mathbf{x}_n^{(v)} - f^{(v)}(\mathbf{z})\right\|^2_{\mathbf{M}^{(v)}}, \tag{7}$$

which is widely used in SOM. In the same manner, precision matrix $\mathbf{M}^{(v)}$ is updated using the inverse of the covariance matrix:

$$\mathbf{S}^{(v)} = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{x}_n^{(v)} - f^{(v)}(\mathbf{z}_n)\right)\left(\mathbf{x}_n^{(v)} - f^{(v)}(\mathbf{z}_n)\right)^{\mathrm{T}} \tag{8}$$

$$\mathbf{M}^{(v)}{}_{\mathrm{new}} := (1-\varepsilon)\mathbf{M}^{(v)}{}_{\mathrm{old}} + \varepsilon\mathbf{S}^{(v)^{-1}}. \tag{9}$$

Equation (7) is applied again. These estimations are repeated, with the neighborhood size reduced gradually until the calculation converges. The algorithm is summarized in Algorithm 1.

In an actual calculation, latent space $\mathcal{Z}$ is discretized to regular grid nodes, like the ordinary SOM, and the integral is evaluated numerically for the discrete nodes.

Algorithm 1 Algorithm of CCA-SOM (ensemble approach)

For all n, initialize $\{\mathbf{z}_n\}$ randomly.
For all v, initialize $\left\{f^{(v)}\right\}$ using Equation (6), and initialize $\left\{\mathbf{M}^{(v)}\right\}$ using identical matrices $\mathbf{I}_{D^{(v)}}$.
repeat
  For all n, update $\{\mathbf{z}_n\}$ using Equation (5).
  For all v, update $\left\{f^{(v)}\right\}$ using Equation (6), then update $\left\{\mathbf{M}^{(v)}\right\}$ using Equations (8) and (9).
until the calculation converges

3.4. **Algorithm for CCA-SOM with the co-training approach.** $\{\mathbf{z}_n\}$ and $\left\{f^{(v)}\right\}$ are updated using Equations (5) and (6), respectively, as in the case of the ensemble approach. By contrast, matrices $\left\{\mathbf{M}^{(v)}\right\}$ are estimated using the co-training approach. Thus, to estimate the v-th view metric, the latent variables are estimated without using the data of the v-th view using

$$\mathbf{z}_n^{(-v)} = \arg\min_{\mathbf{z}} \sum_{v' \neq v} \left\| \mathbf{x}_n^{(v')} - f^{(v')}(\mathbf{z}) \right\|^2_{\mathbf{M}^{(v')}}. \tag{10}$$

Note that $\mathbf{z}_n^{(-v)}$ is regarded as the desired $\mathbf{z}_n$ for the metric learning of the v-th view. Thus, $\mathbf{M}^{(v)}$ is updated as

$$\mathbf{S}^{(v)} = \frac{1}{N} \sum_{n=1}^{N} \left(\mathbf{x}_n^{(v)} - f^{(v)}\left(\mathbf{z}_n^{(-v)}\right)\right)\left(\mathbf{x}_n^{(v)} - f^{(v)}\left(\mathbf{z}_n^{(-v)}\right)\right)^{\mathrm{T}} \tag{11}$$

$$\mathbf{M}^{(v)}{}_{\text{new}} := (1-\varepsilon)\mathbf{M}^{(v)}{}_{\text{old}} + \varepsilon {\mathbf{S}^{(v)}}^{-1}. \tag{12}$$

The algorithm for the co-training approach is shown in Algorithm 2.

Algorithm 2 Algorithm of CCA-SOM (co-training approach)

For all n, initialize $\{\mathbf{z}_n\}$ randomly.
For all v, initialize $\left\{f^{(v)}\right\}$ using Equation (6), and initialize $\left\{\mathbf{M}^{(v)}\right\}$ using identical matrices $\mathbf{I}_{D^{(v)}}$.
repeat
  For all n, update $\{\mathbf{z}_n\}$ using Equation (5).
  For all v, update $\left\{f^{(v)}\right\}$ using Equation (6).
  for all v **do**
    For all n, obtain $\left\{\mathbf{z}_n^{(-v)}\right\}$ using Equation (10).
    Calculate $\left\{\mathbf{S}^{(v)}\right\}$ using Equation (11).
  end for
  For all v, update $\left\{\mathbf{M}^{(v)}\right\}$ using Equation (12).
until the calculation converges.

3.5. **Algorithms for ML-SOM.** The objective function of ML-SOM for the single-view case is given by Equation (2), where $V = 1$. Therefore, the algorithm of ML-SOM with the ensemble approach becomes Algorithm 1, where $V = 1$.

By contrast, the pseudo multi-view is introduced to ML-SOM of the co-training approach. Suppose that $\mathcal{V}$ is the set of all components. At every iteration loop, a subset of data components $\mathcal{U} \subset \mathcal{V}$ is generated randomly as a pseudo view. Then, the latent variable determined by components $\mathcal{U}$,

$$\mathbf{z}_n^{(\mathcal{U})} = \arg\min_{\mathbf{z}} \|\mathbf{x}_n - f(\mathbf{z})\|^2_{\mathbf{M}^{(\mathcal{U})}}, \tag{13}$$

is used for metric estimation, where $\mathbf{M}^{(\mathcal{U})}$ is the submatrix of $\mathbf{M}$ with respect to $\mathcal{U}$ and is updated using

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n^{(v)} - f\left(\mathbf{z}_n^{(\mathcal{U})}\right)\right)\left(\mathbf{x}_n^{(v)} - f\left(\mathbf{z}_n^{(\mathcal{U})}\right)\right)^{\mathrm{T}} \tag{14}$$

$$\mathbf{M}_{\mathrm{new}} := (1-\varepsilon)\mathbf{M}_{\mathrm{old}} + \varepsilon\mathbf{S}^{-1}. \tag{15}$$

Note that mapping f is updated in the same manner as the ensemble approach Equations (5) and (6). The algorithm of ML-SOM of the co-training approach is summarized in Algorithm 3.

Algorithm 3 Algorithm of ML-SOM (co-training approach)

For all n, initialize $\{\mathbf{z}_n\}$ randomly.
Initialize f using Equation (6), and initialize $\mathbf{M}$ using identical matrices $\mathbf{I}_D$.
repeat
  For all n, update $\{\mathbf{z}_n\}$ using Equation (5).
  Update f using Equation (6).
  Choose m components randomly and obtain pseudo view $\mathcal{U}$.
  For all n, obtain $\left\{\mathbf{z}_n^{(\mathcal{U})}\right\}$ using Equation (13).
  Update $\mathbf{M}$ using Equations (14) and (15).
until the calculation converges.

4. Simulation Results.

4.1. **Results of CCA-SOM.** The abilities of the proposed methods were examined using an artificial dataset. Two-view data were synthesized using a uniform distribution on a pair of one-dimensional manifolds in three-dimensional space, and non-white Gaussian noise was added to each sample (Figure 1, ground truth). We examined three algorithms: CCA-SOM with the co-training approach, CCA-SOM with the ensemble approach, and the standard SOM. The precisions of latent variable estimation were assessed using the mutual information between the ground truth and estimated result.

The results are shown in Table 1 and Figure 1. The results show that CCA-SOM with the co-training approach demonstrated the best performance among the three methods. As shown in Figure 1, CCA-SOM with the co-training approach also estimated the manifold shape well compared with the other two methods. For the standard-SOM, it often failed to estimate the manifold shapes. It should also be noted that the metric matrix was often degenerated in CCA-SOM with the ensemble approach.

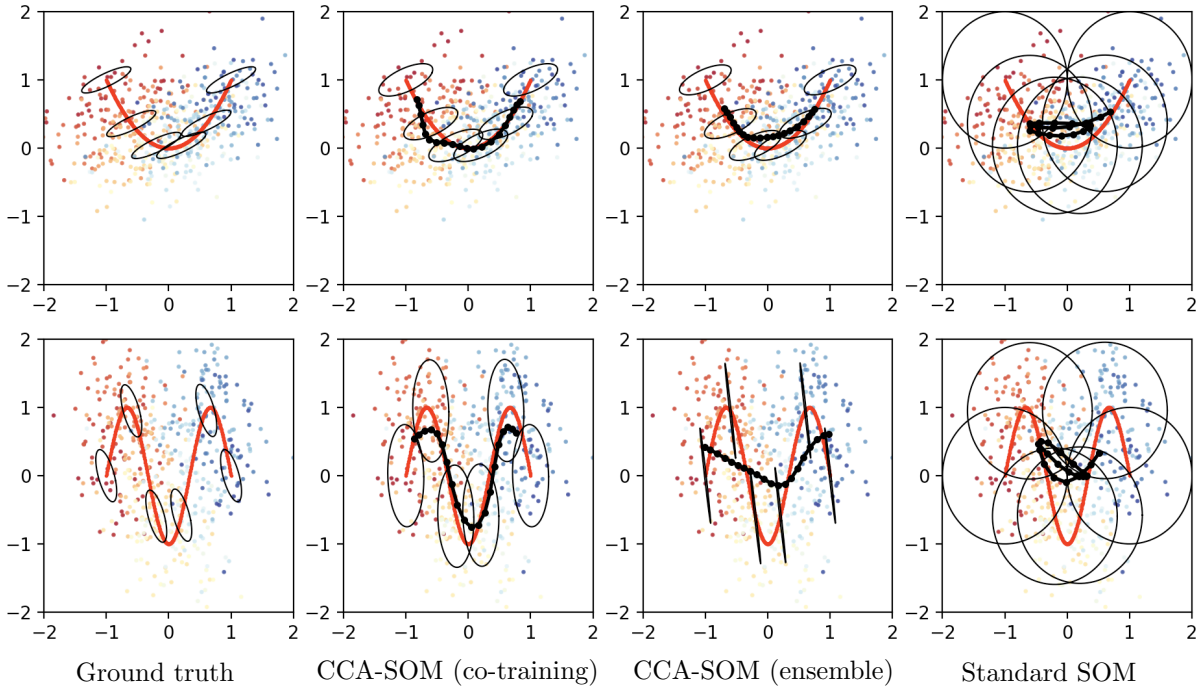

FIGURE 1. (color online) Results on the two-view dataset. Black/red curves are the estimated manifold shapes and ground truth, respectively. The colors of the sample points represent the true latent variables, and the ellipses represent the equidistance curves of Mahalanobis metrics.

TABLE 1. Precisions of latent variable estimation assessed using the mutual information

Algorithm	CCA-SOM (multi-view)	ML-SOM (single-view)
Co-training	**1.26 ± 0.02**	**1.386 ± 0.017**
Ensemble	0.98 ± 0.02	1.282 ± 0.011
Standard SOM	0.95 ± 0.02	1.292 ± 0.008

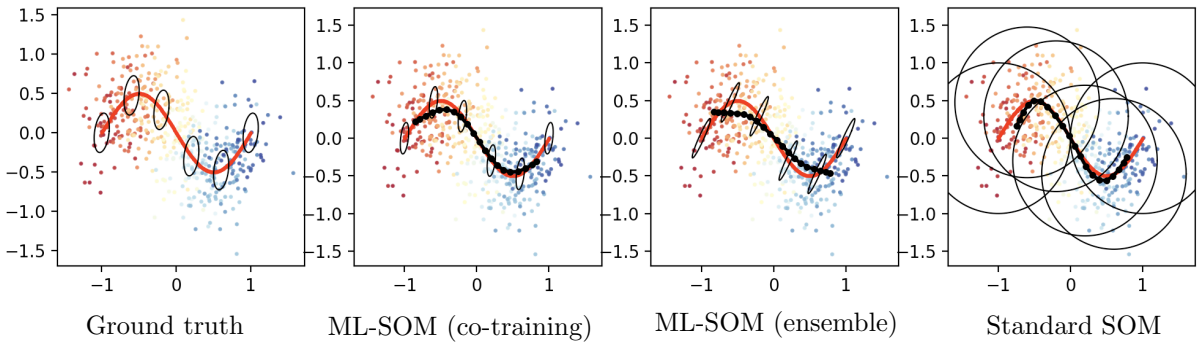

FIGURE 2. (color online) Representative results on a single-view dataset

4.2. **Results of ML-SOM.** The proposed methods were examined on an artificial dataset. The ML-SOM of the co-training approach demonstrated the best performance for latent variable estimation among the three methods (Table 1). It is also noted that the ensemble approach was worse than the standard SOM.

Figure 2 shows an example of manifold estimation. Because of non-Gaussian noise, the results of the standard SOM were biased, whereas such bias was compensated for in the ML-SOM of the co-training approach. In the ensemble approach, the metric matrix

tended to be degenerated, and as a result, the estimation accuracy of the latent variables became worse than that for the co-training approach. These results suggest that the co-training approach is effective for metric learning in both multi-view and single-view cases.

5. **Discussion.** In unsupervised learning, it is known that there is no criterion for determining which features are better than others. This problem is known as *"the ugly duckling theorem"* [26]. The theorem also asserts that we cannot determine the best metric under unsupervised settings without providing additional assumptions. In the co-training approach, ML-SOM learns the metric so that the result becomes robust for missing components. Thus, if a metric is the most robust for any missing component, then the metric is regarded as *the consensus of all components*. This is the assumption that we added into the method. Because this approach resembles the dropout method in deep learning [10, 22], we believe that further investigation will demonstrate the underlying principle.

Interestingly, although the ensemble approach appears to be theoretically plausible, its performance was worse than that of the co-training approach. In the ensemble approach, once a weight of a component became excessively large by coincidence, the component became more dominant in latent variable estimation. As the result, the weight of the component grew further and defeated other components. This is the reason why the metric matrix tended to be degenerated in the ensemble approach. Figure 3 shows a typical example of this phenomenon. In the case of the ensemble approach (dashed lines), this undesired phenomenon occurred around the 200 epoch, whereas such a phenomenon never occurred in the co-training approach (solid lines). This fact implies that the Bayesian approach may not always be optimal. It is also worth stressing that the metric tended to be non-singular in the co-training approach because it was necessary to use any components to be robust against missing components. Therefore, both approaches worked in the opposite direction in metric learning.

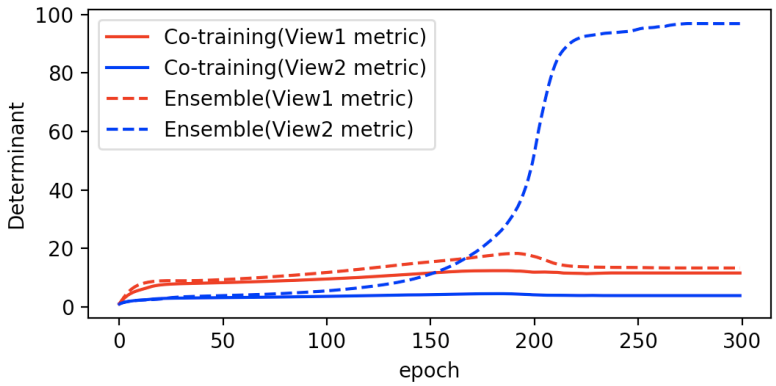

FIGURE 3. Representative results for the metric learning process

6. **Conclusion.** We demonstrated that the co-training approach provides more robust results in metric learning using a simulation. Theoretical analysis on the co-training approach, such as proof of convergence, is a future topic of research. In this study, the proposed method estimated the global metric. It should be further extended to local metric estimation, which is also a future topic of research.

Acknowledgment. This work was supported by JSPS KAKENHI Grant Number 18K11 472. We thank Maxine Garcia, PhD, from Edanz Group (www.edanzediting.com/ac) for editing a draft of this manuscript.

REFERENCES

[1] S. Akaho, A kernel method for canonical correlation analysis, *Proc. of International Meeting on Psychometric Society*, 2001.

[2] B. Arnonkijpanich, A. Hasenfuss and B. Hammer, Local matrix adaptation in topographic neural maps, *Neurocomputing*, vol.74, no.4, pp.522-539, 2011.

[3] A. Blum and T. Mitchell, Combining labeled and unlabeled data with co-training, *Proc. of the 11th Annual Conference on Computational Learning Theory*, pp.92-100, 1998.

[4] Y. Cheng, Convergence and ordering of Kohonen's batch map, *Neural Computation*, vol.9, no.8, pp.1667-1676, 1997.

[5] A. C. Damianou, C. H. Ek, M. K. Titsias and N. D. Lawrence, Manifold relevance determination, *Proc. of the 29th International Conference on Machine Learning*, vol.1, pp.145-152, 2012.

[6] C. H. Ek, P. Jaeckel, N. Campbell, N. D. Lawrence and C. Melhuish, Shared Gaussian process latent variable models for handling ambiguous facial expressions, *AIP Conference Proceedings*, vol.1107, pp.147-153, 2009.

[7] T. Furukawa, SOM of SOMs, *Neural Networks*, vol.22, no.4, pp.463-478, 2009.

[8] T. Graepel, M. Burger and K. Obermayer, Self-organizing maps: Generalizations and new optimization techniques, *Neurocomputing*, vol.21, pp.173-190, 1998.

[9] T. Heskes, J.-J. Spanjers and W. Wiegerinck, EM algorithms for self-organizing maps, *Proc. of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, pp.9-14, 2000.

[10] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, *Computer Science*, 2012.

[11] H. Hotelling, Relations between two sets of variates, *Biometrika*, vol.28, pp.321-377, 1936.

[12] S. Kaski, J. Sinkkonen and J. Peltonen, Bankruptcy analysis with self-organizing maps in learning metrics, *IEEE Trans. Neural Networks*, vol.12, no.4, pp.936-947, 2001.

[13] S. Kaski, J. Sinkkonen and J. Peltonen, Data visualization and analysis with self-organizing maps in learning metrics, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol.2114, pp.162-173, 2001.

[14] S. Kaski, J. Sinkkonen and J. Peltonen, Learning metrics for self-organizing maps, *Proc. of the International Joint Conference on Neural Networks*, vol.2, pp.914-919, 2001.

[15] A. Klami, A. K. Fi, S. J. V. Fi, S. Kaski and S. K. Fi, Bayesian canonical correlation analysis, *Journal of Machine Learning Research*, vol.14, pp.965-1003, 2013.

[16] T. Kohonen, S. Kaski and H. Lappalainen, Self-organized formation of various invariant-feature filters in the adaptive-subspace SOM, *Neural Computation*, vol.9, no.6, pp.1321-1344, 1997.

[17] B. Kulis, Metric learning: A survey, *Foundations and Trends in Machine Learning*, vol.5, no.4, pp.287-364, 2012.

[18] N. Lawrence, Probabilistic non-linear principal component analysis with Gaussian process latent variable models, *Journal of Machine Learning Research*, vol.6, pp.1783-1816, 2005.

[19] S. P. Luttrell, Self-organization: A derivation from first principle of a class of learning algorithms, *IEEE Conference on Neural Networks*, pp.495-498, 1989.

[20] P. Płoński and K. Zaremba, Improving performance of self-organising maps with distance metric learning method, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol.7267, pp.169-177, 2012.

[21] M. Salzmann, C. H. Ek, R. Urtasun and T. Darrell, Factorized orthogonal latent spaces, *Journal of Machine Learning Research*, vol.9, pp.701-708, 2010.

[22] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *Journal of Machine Learning Research*, vol.15, pp.1929-1958, 2014.

[23] S. Sun, A survey of multi-view machine learning, *Neural Computing and Applications*, vol.23, no.7, pp.2031-2038, 2013.

[24] J. Verbeek, N. Vlassis and B. Krose, Self-organizing mixture models, *Neurocomputing*, vol.63, pp.99-123, 2005.

[25] F. Wang and J. Sun, Survey on distance metric learning and dimensionality reduction in data mining, *Data Mining and Knowledge Discovery*, vol.29, no.2, pp.534-564, 2014.

[26] S. Watanabe, *Knowing and Guessing: Quantitative Study of Inference and Information*, 1969.

[27] C. Xu, D. Tao and C. Xu, A survey on multi-view learning, *arXiv:1304.5634*, 2013.

[28] L. Yang, Distance metric learning: A comprehensive survey, *Technical Report*, Department of Computer Science and Engineering, Michigan State University, 2006.