SUPPLEMENTARY MATERIALS

A RELATIONSHIP BETWEEN THE DISCRIMINATIVE GAUSSIAN AND LOGISTIC REGRESSION

We show that a fully connected layer with the softmax function, or logistic regression, can be regarded as a discriminative model based on a Gaussian distribution by utilizing transformation of the equations. Let us consider a case in which the class-conditional probability $P(\boldsymbol{x}|c)$ is a Gaussian distribution. In this case, we can omit m from the equations (3)–(6).

If all classes share the same covariance matrix and the mixture weight π_{cm}, the terms π_{cm} in (1), $x_1^2, x_1x_2, \cdots, x_1x_D, x_2^2, x_2x_3, \cdots, x_2x_D, \cdots, x_D^2$ in (2), and $-\frac{1}{2}s_{c11}, \cdots, -\frac{1}{2}s_{cDD}$ in (6) can be canceled; hence the calculation of the posterior probability $P(c|\boldsymbol{x})$ is also simplified as

$$P(c|\boldsymbol{x}) = \frac{\exp({\boldsymbol{w}_c}^{\mathrm{T}}\boldsymbol{\phi})}{\sum_{c=1}^{C}\exp({\boldsymbol{w}_c}^{\mathrm{T}}\boldsymbol{\phi})},$$

where

$$\boldsymbol{w}_c = [\log P(c) - \frac{1}{2}\sum_{i=1}^{D}\sum_{j=1}^{D}s_{cij}\mu_{ci}\mu_{cj} + \frac{D}{2}\log 2\pi + \frac{1}{2}\log|\boldsymbol{\Sigma_c}|, \sum_{i=1}^{D}s_{ci1}\mu_{ci}, \cdots, \sum_{i=1}^{D}s_{ciD}\mu_{ci}]^{\mathrm{T}},$$

$$\boldsymbol{\phi} = \left[1, \boldsymbol{x}^{\mathrm{T}}\right]^{\mathrm{T}}.$$

This is equivalent to a fully connected layer with the softmax function, or linear logistic regression.

B EVALUATION OF CHARACTERISTICS USING SYNTHETIC DATA

To evaluate the characteristics of the SDGM, we conducted classification experiments using synthetic data. The dataset comprises two classes. The data were sampled from a Gaussian mixture model with eight components for each class. The numbers of training data and test data were 320 and 1,600, respectively. The scatter plot of this dataset is shown in Figure 6.

In the evaluation, we calculated the error rates for the training data and the test data, the number of components after training, the number of nonzero weights after training, and the weight reduction ratio (the ratio of the number of the nonzero weights to the number of initial weights), by varying the number of initial components as $2, 4, 8, \ldots, 20$. We repeated evaluation for five times while regenerating the training and test data and calculated the average value for each evaluation criterion. We used the dual form of the SDGM in this experiment.

Figure 6 displays the changes in the learned class boundaries according to the number of initial components. When the number of components is small, such as that shown in Figure 6(a), the decision boundary is simple; therefore, the classification performance is insufficient. However, according to the increase in the number of components, the decision boundary fits the actual class boundaries. It is noteworthy that the SDGM learns the GMM as a discriminative model instead of a generative model; an appropriate decision boundary was obtained even if the number of components for the model is less than the actual number (e.g., 6(c)).

Figure 7 shows the evaluation results of the characteristics. Figures 7(a), (b), (c), and (d) show the recognition error rate, number of components after training, number of nonzero weights after training, and weight reduction ratio, respectively. The horizontal axis shows the number of initial components in all the graphs.

In Figure 7(a), the recognition error rates for the training data and test data are almost the same with the few numbers of components and decrease according to the increase in the number of initial components while it is 2 to 6. This implied that the representation capability was insufficient when the number of components was small, and that the network could not accurately separate the classes. Meanwhile, changes in the training and test error rates were both flat when the number of initial components exceeded eight, even though the test error rates were slightly higher than the training error rate. In general, the training error decreases and the test error increases when the complexity of

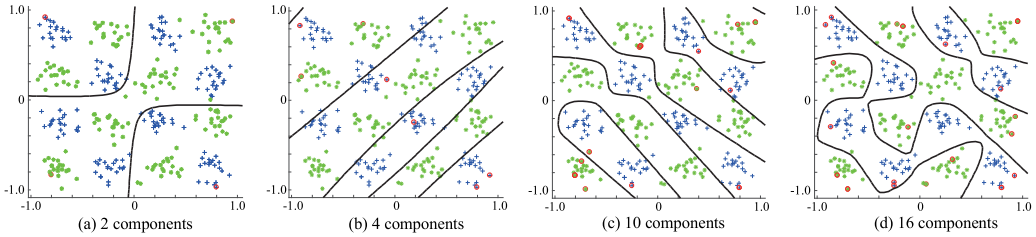

Figure 6: Changes in learned class boundaries according to number of initial components. The blue and green markers represent the samples from class 1 and class 2, respectively. Samples in red circles represent relevant vectors. The black lines are class boundaries where $P(c \mid \boldsymbol{x}) = 0.5$.

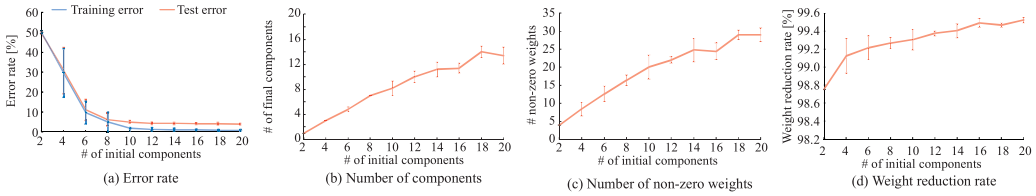

Figure 7: Evaluation results using synthetic data. (a) recognition error rate, (b) the number of components after training, (c) the number of nonzero weights after training, and (d) weight reduction ratio. The error bars indicate standard deviation for five trials.

the classifier is increased. However, the SDGM suppresses the increase in complexity using sparse Bayesian learning, thereby preventing overfitting.

In Figure 7(b), the number of components after training corresponds to the number of initial components until the number of initial components is eight. When the number of initial components exceeds ten, the number of components after training tends to be reduced. In particular, eight components are reduced when the number of initial components is 20. The results above indicate the SDGM can reduce unnecessary components.

From the results in Figure 7(c), we confirm that the number of nonzero weights after training increases according to the increase in the number of initial components. This implies that the complexity of the trained model depends on the number of initial components, and that the minimum number of components is not always obtained.

Meanwhile, in Figure 7(d), the weight reduction ratio increases according to the increase in the number of initial components. This result suggests that the larger the number of initial weights, the more weights were reduced. Moreover, the weight reduction ratio is greater than 99 % in any case. The results above indicate that the SDGM can prevent overfitting by obtaining high sparsity and can reduce unnecessary components.