Estimation of Non-Normalized Mixture Models

Takeru Matsuda
The University of Tokyo
RIKEN Center for Brain Science

Aapo Hyvärinen
University College London
University of Helsinki

Abstract

We develop a general method for estimating a finite mixture of non-normalized models. A non-normalized model is defined to be a parametric distribution with an intractable normalization constant. Existing methods for estimating non-normalized models without computing the normalization constant are not applicable to mixture models because they contain more than one intractable normalization constant. The proposed method is derived by extending noise contrastive estimation (NCE), which estimates non-normalized models by discriminating between the observed data and some artificially generated noise. In particular, the proposed method provides a probabilistically principled clustering method that is able to utilize a deep representation. Applications to clustering of natural images and neuroimaging data give promising results.

1 INTRODUCTION

Many statistical models are given in the form of non-normalized densities with an intractable normalization constant; they are also called energy-based. Since maximum likelihood estimation is computationally very intensive for these models, several estimation methods have been developed which do not require the normalization constant (i.e. the partition function), or somehow estimate it as part of the estimation process. These include pseudo-likelihood (Besag, 1974), contrastive divergence (Hinton, 2002), score matching (Hyvärinen, 2005), and noise contrastive estimation (Gutmann and Hyvärinen, 2010).

On the other hand, mixture models are a well-known general-purpose approach to unsupervised modelling of complex distributions, especially in the form of the Gaussian Mixture Model. In particular, estimation of a finite mixture model leads to a probabilistically principled clustering method. Compared to other clustering methods such as hierarchical clustering and k-means clustering, such model-based methods naturally quantify the uncertainty of the cluster memberships.

It would be very interesting to be able to model data with a mixture of non-normalized densities. However, such applications are scarce, presumably because it is not known how to estimate such model. In particular, it is not known if any of the aforementioned methods is applicable in such a setting, since we have several normalization constants instead of a single one.

One motivating application where non-normalized models and mixture models naturally meet is learning a clustering based on features learned by a neural network. Deep neural networks have been shown to learn useful representations from labeled data such as ImageNet, and such representations seem to be useful for analyzing other datasets, or for performing other tasks; this is a fundamental case of transfer learning.

In this study, we develop a general method for estimating a finite mixture of non-normalized models. The proposed method is expected to significantly increase the practicality of non-normalized mixture models. As an application of great practical interest, we apply the framework for transferring a deep representation to clustering of unlabeled data. Our approach provides a probabilistically principled solution for the clustering problem, building a probabilistic model that propagates back to the original data space.

To accomplish our goal, first, we extend noise contrastive estimation (NCE) to estimate a finite mixture of non-normalized models and show its consistency. Then, based on the observation that classification learning with neural networks is implicitly assuming an exponential family as a generative model, we propose a method for clustering unlabeled data by using a deep representation. The proposed method

Proceedings of the 22$^{\text{nd}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89. Copyright 2019 by the author(s).

estimates a finite mixture of distributions in an exponential family that is derived from the deep representation, by using the proposed extension of NCE with a particular choice of noise. Finally, we apply the proposed method to clustering of natural images and neuroimaging data, with promising results.

2 BACKGROUND: NOISE CONTRASTIVE ESTIMATION

In this section, we briefly review the problem of non-normalized models, and its solution by noise contrastive estimation (Gutmann and Hyvärinen, 2010).

Suppose we have N samples $x_1, \cdots, x_N$ from a parametric distribution

$$p(x \mid \theta) = \frac{1}{Z(\theta)} \widetilde{p}(x \mid \theta), \tag{1}$$

where θ is an unknown parameter and $Z(\theta)$ is the normalization constant. For several statistical models such as Markov random fields (Li, 2001) and energy-based overcomplete ICA models (Teh et al., 2004), only the non-normalized density $\widetilde{p}(x \mid \theta)$ is given and the calculation of $Z(\theta)$ is intractable. Thus, several methods have been developed to estimate θ without explicitly computing $Z(\theta)$ (Besag, 1974; Hinton, 2002; Hyvärinen, 2005; Gutmann and Hyvärinen, 2010).

In noise contrastive estimation (NCE), the non-normalized model is rewritten as

$$\log p(x \mid \theta, c) = \log \widetilde{p}(x \mid \theta) + c, \tag{2}$$

where the scalar $c = -\log Z(\theta)$ is also viewed as an unknown parameter and estimated from data. In addition to data $x_1, \cdots, x_N$, we generate M noise samples $y_1, \cdots, y_M$ from a noise distribution $n(y)$. The noise distribution should be reasonably difficult to discriminate from the real data, while having a tractable probability density function. For example, $n(y)$ can be set to the Gaussian distribution with the same mean and covariance with data. Then, the estimate of (θ, c) is defined by learning to discriminate between the data and the noise as accurately as possible:

$$(\hat{\theta}_{\mathrm{NCE}}, \hat{c}_{\mathrm{NCE}}) = \arg\max_{\theta, c} \hat{J}_{\mathrm{NCE}}(\theta, c), \tag{3}$$

where

$$\begin{aligned} \hat{J}_{\mathrm{NCE}}(\theta, c) &= \frac{1}{N} \sum_{t=1}^{N} \log \frac{N p(x_t \mid \theta, c)}{N p(x_t \mid \theta, c) + M n(x_t)} \\ &\quad + \frac{1}{N} \sum_{t=1}^{M} \log \frac{M n(y_t)}{N p(y_t \mid \theta, c) + M n(y_t)}. \end{aligned} \tag{4}$$

The objective function $\hat{J}_{\mathrm{NCE}}$ is the log-likelihood of the logistic regression classifier. NCE has consistency and asymptotic normality under mild regularity conditions (Gutmann and Hyvärinen, 2012). Note that NCE is somewhat similar in spirit to Generative Adversarial Networks (Goodfellow et al., 2014), which aim to generate realistic data by training a generative network and a discriminative network simultaneously.

3 NON-NORMALIZED MIXTURE MODELS AND EXTENDING NCE

In this section, we first introduce the problem of non-normalized mixture models. Then, we develop a general method for estimating non-normalized mixture models by extending NCE and discuss its application to clustering. We also prove the consistency of the extended NCE.

3.1 Non-normalized mixture models

We begin by defining the statistical model whose estimation is the central problem of this study. Suppose we have N samples $x_1, \cdots, x_N$ from a finite mixture distribution

$$p(x \mid \theta, \pi) = \sum_{k=1}^{K} \pi_k \cdot p_k(x \mid \theta_k), \tag{5}$$

where $\pi_k > 0$, $\sum_{k=1}^{K} \pi_k = 1$, and

$$p_k(x \mid \theta_k) = \frac{1}{Z(\theta_k)} \widetilde{p}_k(x \mid \theta_k). \tag{6}$$

Here, $\theta = (\theta_1, \cdots, \theta_K)$ and $\pi = (\pi_1, \cdots, \pi_K)$ are unknown parameters and the normalization constant $Z(\theta_k)$ of each component $p_k(x \mid \theta_k)$ is intractable.

Existing methods for estimating non-normalized models are not applicable to (5) since it includes more than one intractable normalization constant. Although Nair and Hinton (2008) extended the contrastive divergence method to estimate a finite mixture of restricted Boltzmann machines, that is only a special case.

For language modeling, Wang et al. (2017) proposed a statistical model called the trans-dimensional random field model, which has a similar form to (5) but they assumed the class label k is also observed. Wang et al. (2017) estimated this model based on the stochastic approximation technique (Younes, 1989) and later Wang and Ou (2018) improved its efficiency by using NCE.

3.2 Extending noise contrastive estimation

Here, we propose an extension of NCE to estimate the non-normalized mixture model (5) in a general setting.

First, we reparametrize (5) as

$$p(x \mid \theta, c) = \sum_{k=1}^{K} p_k(x \mid \theta_k, c_k), \tag{7}$$

where $c = (c_1, \cdots, c_K)$ with $c_k = \log \pi_k - \log Z(\theta_k)$ and each $p_k(x \mid \theta_k, c_k)$ is defined as

$$\log p_k(x \mid \theta_k, c_k) = \log \widetilde{p}_k(x \mid \theta_k) + c_k. \tag{8}$$

When $K = 1$, this reparametrization reduces to that used in the original NCE in (2). Similarly to the original NCE, we consider c as an additional unknown parameter. Then, we generate M noise samples $y_1, \cdots, y_M$ from a noise distribution $n(y)$ and estimate (θ, c) in the same way as the original NCE in (3) and (4), that is, we use the definition (7) in the original NCE objective function (4). This estimator has consistency under mild regularity conditions similar to the original NCE, as will be shown in the next subsection. Note that the additional parameter c_k incorporates both the mixture weight π_k and the normalization constant $Z(\theta_k)$ and so we cannot obtain an estimate of π_k from the estimate of c_k, although it is not a problem for clustering application as shown next.

The estimation result can be used for clustering of $x_1, \cdots, x_N$. Specifically, by introducing a hidden variable z taking values in $\{1, \cdots, K\}$, the non-normalized mixture model (5) is rewritten in a hierarchical form:

$$p(z = k \mid \pi) = \pi_k \quad (k = 1, \cdots, K), \tag{9}$$

$$p(x \mid z = k; \theta) = p_k(x \mid \theta_k). \tag{10}$$

Then, the posterior of z given x is

$$p(z = k \mid x; \theta, \pi) = \frac{\pi_k p_k(x \mid \theta_k)}{\sum_{j=1}^{K} \pi_j p_j(x \mid \theta_j)}, \tag{11}$$

for $k = 1, \cdots, K$. By reparametrization, (11) is rewritten as

$$p(z = k \mid x; \theta, c) = \frac{\exp\left(\log \widetilde{p}_k(x \mid \theta_k) + c_k\right)}{\sum_{j=1}^{K} \exp\left(\log \widetilde{p}_j(x \mid \theta_j) + c_j\right)}. \tag{12}$$

Thus, based on the posterior $p(z_t = k \mid x_t; \hat{\theta}, \hat{c})$ for each x_t, clustering of $x_1, \cdots, x_N$ is obtained.

3.3 Consistency of extended NCE

Here, we show the consistency of the extended NCE introduced in the previous subsection for estimating non-normalized mixture models.

When all components in the non-normalized mixture model (7) belong to the same parametric model, the parameters of (7) have indeterminacy with respect to the ordering of K components. To remove this indeterminacy, we put order constraints on $\theta_1, \cdots, \theta_K$. For example, when each θ_k is scalar, we assume $\theta_1 < \cdots < \theta_K$.

Let $\xi = (\theta, c)$. Suppose we have N samples $x_1, \cdots, x_N$ from $p(x \mid \xi^*)$ in (7). Here, from the discussion above, the true parameter value $\xi^* = (\theta^*, c^*)$ is defined uniquely. We consider the asymptotics where $N \to \infty$, $M \to \infty$ and $M/N \to \nu$, which is the same setting with Gutmann and Hyvärinen (2012). Let

$$J_{\mathrm{NCE}}(\xi) = \int p(x \mid \xi^*) \log \frac{N p(x \mid \xi)}{N p(x \mid \xi) + M n(x)} \mathrm{d}x + \nu \int n(y) \log \frac{M n(y)}{N p(y \mid \xi) + M n(y)} \mathrm{d}y. \tag{13}$$

Then, the consistency of extended NCE is stated as follows. Here, $\xrightarrow{p}$ means the convergence in probability.

Theorem 1. *Assume the following.*

- *$n(x)$ is nonzero whenever $p(x \mid \xi^*)$ is nonzero.*
- $\sup_\xi |\hat{J}_{\mathrm{NCE}}(\xi) - J_{\mathrm{NCE}}(\xi)| \xrightarrow{p} 0.$
- *The matrix $I = \int g(u) g(u)^\top P_\nu(u) p(u \mid \xi^*) \mathrm{d}u$ has full rank, where*

$$g(u) = \nabla \log_\xi p(u \mid \xi)\big|_{\xi = \xi^*}, \tag{14}$$

$$P_\nu(u) = \frac{\nu n(u)}{p(u \mid \xi^*) + \nu n(u)}. \tag{15}$$

Then, $\hat{\xi}_{\mathrm{NCE}}$ in Section 3.2 converges in probability to ξ^*: $\hat{\xi}_{\mathrm{NCE}} \xrightarrow{p} \xi^*$.

The proof is obtained by extending the proof for the original NCE (Theorem 2 of Gutmann and Hyvärinen, 2012), which is based on the nonparametric characterization of the log-pdf of data distribution (Theorem 1 of Gutmann and Hyvärinen, 2012). This characterization becomes non-trivial when the K components belong to the same parametric model. Specifically, we need the following Lemma.

Lemma 1. *Suppose that the components $p_k(x \mid \theta_k)$ in (5) belong to the same parametric model $p(x \mid \theta)$. Assume the following.*

(a) The set $\{p(x \mid \theta) \mid \theta \in \Theta\}$ is linearly independent, where Θ is the parameter space of the parametric model $p(x \mid \theta)$.

(b) The parameters $\theta_1^*, \cdots, \theta_K^*$ are all different.

(c) The parameters $\pi_1^*, \cdots, \pi_K^*$ are all nonzero.

(d) $n(x)$ is nonzero whenever $p(x \mid \xi^*)$ is nonzero.

Then,

$$\begin{aligned} &\arg\max_{\xi} J_{\mathrm{NCE}}(\xi) \\ =&\{(\xi^*_{\sigma(1)}, \cdots, \xi^*_{\sigma(K)}) \mid \sigma \in S_n\}, \end{aligned} \tag{16}$$

where S_n is the set of all permutations of $\{1, \cdots, K\}$.

The proof is given in Supplementary Material. Assumption (a) holds for general exponential families including the Gaussian distribution. Under this assumption, assumptions (b) and (c) mean that the true data-generating distribution has exactly K components. Assumption (d) is standard in noise contrastive estimation (Gutmann and Hyvärinen, 2012) and easily fulfilled by taking, for example, a Gaussian as the noise distribution.

4 CLUSTERING WITH DEEP REPRESENTATION

In this section, based on the extended NCE in the previous section, we propose a clustering method using deep representation. First, we review an interpretation where an exponential family is implicitly assumed as a generative model (probability distribution of data in each category) in classification learning with neural networks. Next, we investigate an extension of NCE with multiple noise distributions. Finally, we present an ensuing method for clustering deep representations.

4.1 Exponential family and neural networks

Here, we review the relationship between an exponential family and classification with neural networks. See Dai (2015) and Xie et al. (2016) for details.

We consider image classification for convenience. Let x be image data and z be its category. We assume that z takes values in $\{1, \cdots, L\}$. In classification with neural networks, the softmax function is commonly used in the output layer. Namely, the probability output is computed by

$$p(z = l \mid x) = \frac{\exp(\sum_{i=1}^d w_{li} f_i(x))}{\sum_{j=1}^L \exp(\sum_{i=1}^d w_{ji} f_i(x))}, \tag{17}$$

where $l = 1, \cdots, L$, d is the number of units in the last hidden layer, $f_i(x)$ is the activation of the i-th unit in the last hidden layer when x is input to this network, and w_{ji} is the connection weight between the i-th unit in the last hidden layer and the j-th output unit. Thus, neural networks learn to extract nonlinear features $f_1, \cdots, f_d$ that are useful for image classification.

From (17) and Bayes' formula, we obtain

$$\begin{aligned} &p(x \mid z = l) \\ =&p(x \mid z = 1) \frac{p(z=1)}{p(z=l)} \exp\left(\sum_{i=1}^d (w_{li} - w_{1i}) f_i(x)\right), \end{aligned} \tag{18}$$

for $l = 1, \cdots, L$. Here, the prior probability $p(z)$ is defined from the proportion of each category in the training data. Therefore, the distribution of images in the l-th category $p(x \mid z = l)$ belongs to the exponential family

$$p(x \mid \theta) = h(x) \exp\left(\sum_{i=1}^d \theta_i f_i(x) - A(\theta)\right) \tag{19}$$

with $\theta_i = w_{li}$ and $A(\theta) = \log p(z = l) - \log p(z = 1)$, where

$$h(x) = p(x \mid z = 1) \exp\left(-\sum_{i=1}^d w_{1i} f_i(x)\right). \tag{20}$$

Thus, classification with neural networks (17) implicitly assumes the exponential family (19) as a generative model.

For image data, many pretrained networks are publicly available such as AlexNet (Krizhevsky et al., 2012) and inception-v3 (Szegedy et al., 2015). Although these networks were trained for ImageNet competition, they have learned a useful representation of general natural images. Indeed, they work well empirically as feature extractors for other image data. Therefore, the distributional assumption (19) seems to be reasonable even for image categories outside of the ImageNet competition.

4.2 NCE with multiple noise distributions

In the original NCE, we generate noise samples from one noise distribution and discriminate between data and noise. In order that such discrimination learns deep structure in the data, it would intuitively seem important that the noise distribution is as close as possible to the real data distribution. Thus, it would be more efficient to use several noise distributions, since different noise distributions would enable to learn different kinds of data structure. Here, we introduce NCE with multiple noise distributions and discuss its equivalence to the original NCE with a mixture noise distribution.

Suppose we have N samples $x_1, \cdots, x_N$ from a non-normalized model (1) or a non-normalized mixture model (5). We consider L noise distributions $n_1(y), \cdots, n_L(y)$ and generate M_l noise samples

$y_1^{(l)}, \cdots, y_{M_l}^{(l)}$ from each $n_l(y)$. Then, similarly to the original NCE and its extension in Section 3.2, an estimate of (θ, c) can be defined by discriminating between $L+1$ classes (data, noise 1, $\cdots$, noise L) as correctly as possible:

$$(\hat{\theta}_{\rm MNCE}, \hat{c}_{\rm MNCE}) = \arg\max_{\theta, c} \hat{J}_{\rm MNCE}(\theta, c), \qquad (21)$$

where

$$\begin{aligned} &\hat{J}_{\rm MNCE}(\theta, c) \\ =&\frac{1}{N} \sum_{t=1}^{N} \log \frac{N p(x_t \mid \theta, c)}{N p(x_t \mid \theta, c) + \sum_{l=1}^{L} M_l n_l(x_t)} \\ &+ \frac{1}{N} \sum_{l=1}^{L} \sum_{t=1}^{M_l} \log \frac{M_l n_l(y_t^{(l)})}{N p(y_t^{(l)} \mid \theta, c) + \sum_{l=1}^{L} M_l n_l(x_t)}. \end{aligned} \qquad (22)$$

On the other hand, we can regard $y_1^{(1)}, \cdots, y_{M_1}^{(1)}, \cdots, y_1^{(L)}, \cdots, y_{M_L}^{(L)}$ as samples from the mixture distribution

$$n(y) = \sum_{l=1}^{L} \frac{M_l}{M_1 + \cdots + M_L} n_l(y), \qquad (23)$$

and use the original NCE $(\hat{\theta}_{\rm NCE}, \hat{c}_{\rm NCE})$ as (3).

In fact, these two estimators coincide as follows:

Theorem 2.

$$(\hat{\theta}_{\rm MNCE}, \hat{c}_{\rm MNCE}) = (\hat{\theta}_{\rm NCE}, \hat{c}_{\rm NCE}). \qquad (24)$$

The proof is given in Supplementary Material. From Theorem 2, NCE with multiple noise distributions has the same statistical properties with the original NCE. We will present simulation results for a typical situation where using multiple noise distributions is beneficial in Section 5.

4.3 Deep clustering method

Now, we combine the developments above to finally provide a method for transferring the representation of a deep neural network to clustering of unlabeled data, using the exponential family introduced in Section 4.1 and the extensions of NCE proposed in Section 3.2 and Section 4.2. While in the current state of research, it seems that the only way to employ the deep representation for clustering is to heuristically apply conventional clustering algorithms to the feature vectors, here we provide a probabilistically principled clustering method that leverages the deep representation. Again, for concreteness of exposition, we consider image clustering, although the method is quite general.

Suppose we have N images $x_1, \cdots, x_N$ and a neural network previously trained ("pretrained") on some other image dataset (e.g., AlexNet, inception-v3). We assume that $x_1, \cdots, x_N$ belongs to the same exponential family (19) with the image data on which the network was pretrained, in other words, the difference is only in the last layer weights. Then, the generative model of $x_1, \cdots, x_N$ is a finite mixture of distributions in the same exponential family:

$$p(x \mid \theta, \pi) = \sum_{k=1}^{K} \pi_k \cdot h(x) \exp\left(\sum_{i=1}^{d} \theta_{ki} f_i(x) - A(\theta_k) \right), \qquad (25)$$

where K is the number of image categories in $x_1, \cdots, x_N$. Note that $A(\theta_k)$ here are not known and intractable, although they were known for the categories used in training. Like (7), we reparametrize (25) as

$$p(x \mid \theta, c) = h(x) \sum_{k=1}^{K} \exp\left(\sum_{i=1}^{d} \theta_{ki} f_i(x) + c_k \right), \qquad (26)$$

where $c = (c_1, \cdots, c_K)$. From (20), the function h is a function of the distribution of one image category $p(x \mid z=1)$ and so it is totally unknown. Yet, clustering of $x_1, \cdots, x_N$ is possible if we can estimate θ and c, since the function h cancels out in the posterior:

$$p(z = k \mid x; \theta, c) = \frac{\exp\left(\sum_{i=1}^{d} \theta_{ki} f_i(x) + c_k \right)}{\sum_{j=1}^{K} \exp\left(\sum_{i=1}^{d} \theta_{ji} f_i(x) + c_j \right)}, \qquad (27)$$

where $k = 1, \cdots, K$.

We use the NCE extensions in Section 3.2 and Section 4.2 to estimate θ and c in (26). Here, we have to be careful in the choice of the noise distribution because of the unknown function h in (26). If we generate noise samples artificially, h remains in the objective function of NCE (4) and so the optimization is impossible. To get rid of h, we propose here to use the original training data of the pretrained network as noise samples. Specifically, let $\widetilde{x}_1^{(1)}, \cdots, \widetilde{x}_{M_1}^{(1)}, \cdots, \widetilde{x}_1^{(L)}, \cdots, \widetilde{x}_{M_L}^{(L)}$ be the training data of the pretrained network, where L is the number of categories and M_l is the number of samples in the l-th category. Then, the prior probability is $p(z = l) = M_l / M$ where $M = M_1 + \cdots + M_L$. Therefore, from (18) and (20), the distribution of images in the l-th pre-training category (here used as noise) is

$$q_l(\widetilde{x}) = h(\widetilde{x}) \frac{M_1}{M_l} \exp\left(\sum_{i=1}^{d} w_{li} f_i(\widetilde{x}) \right) \qquad (28)$$

for $l = 1, \cdots, L$. Thus, we regard $q_1, \cdots, q_L$ as noise distributions and the training data $\widetilde{x}_1^{(l)}, \cdots, \widetilde{x}_{M_l}^{(l)}$ as samples from q_l for $l = 1, \cdots, L$, respectively[1].

In summary, the estimate of (θ, c) is given by

$$(\hat{\theta}_{\mathrm{MNCE}}, \hat{c}_{\mathrm{MNCE}}) = \arg \max_{\theta, c} \hat{J}_{\mathrm{MNCE}}(\theta, c), \tag{29}$$

where

$$\begin{aligned} \hat{J}_{\mathrm{MNCE}}(\theta, c) &= \frac{1}{N} \sum_{t=1}^{N} \log \frac{N \bar{p}(x_t \mid \theta, c)}{N \bar{p}(x_t \mid \theta, c) + M_1 \bar{n}(x_t)} \\ &+ \frac{1}{N} \sum_{l=1}^{L} \sum_{t=1}^{M_l} \log \frac{M_1 \exp\left(\sum_{i=1}^{d} w_{li} f_i(\widetilde{x}_t^{(l)})\right)}{N \bar{p}(\widetilde{x}_t^{(l)} \mid \theta, c) + M_1 \bar{n}(\widetilde{x}_t^{(l)})}, \end{aligned} \tag{30}$$

$$\bar{p}(x \mid \theta, c) = \sum_{k=1}^{K} \exp\left(\sum_{i=1}^{d} \theta_{ki} f_i(x) + c_k \right), \tag{31}$$

$$\bar{n}(x) = \sum_{l=1}^{L} \exp\left(\sum_{i=1}^{d} w_{li} f_i(x) \right). \tag{32}$$

Note that h cancels out in $\hat{J}_{\mathrm{MNCE}}$, and so the objective function only depends on quantities we can readily compute. Using the estimate (29), clustering of $x_1, \cdots, x_N$ is obtained by the posterior (27).

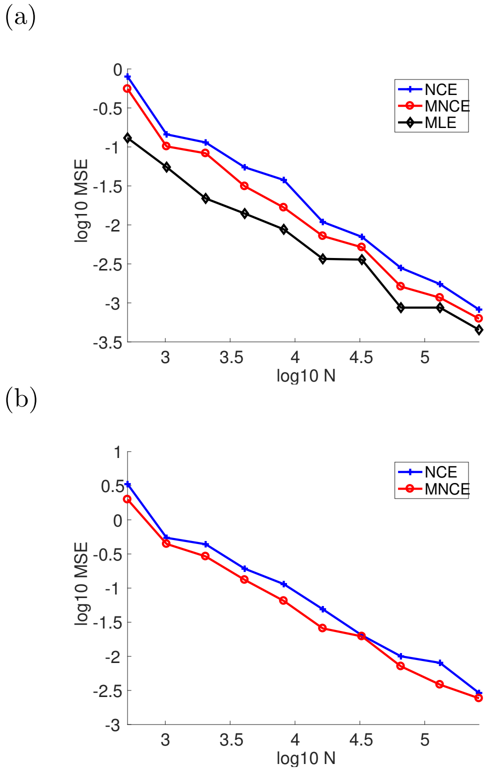

Figure 1: Estimation errors for (a) θ and (b) c in the Gaussian mixture distribution (33).

5 SIMULATION RESULTS

In this section, we use simulations to further confirm the validity of the estimation of non-normalized mixture models by extensions of NCE proposed in Section 3.2 and Section 4.2. As a special case of finite mixture models (7), we consider the one-dimensional Gaussian mixture distribution. Namely,

$$p(x \mid \theta, c) = \sum_{k=1}^{K} \exp(\theta_{k1} x^2 + \theta_{k2} x + c_k). \tag{33}$$

where we pretend not to be able to compute the normalization constants for the purpose of this simulation. We generated N samples $x_1, \cdots, x_N$ from the two-component Gaussian mixture distribution $0.5 \cdot \mathrm{N}(0,1) + 0.5 \cdot \mathrm{N}(4,1)$. The sample size N was set to $2^9, 2^{10}, \cdots, 2^{18}$ and the simulation was repeated 100 times for each sample size.

We consider two estimation methods, both of which are based on the proposed extensions of NCE. The first method is NCE with $M = N$ noise samples generated from the Gaussian distribution $\mathrm{N}(2,5)$, which has the same mean and variance with the true data-generating distribution $0.5 \cdot \mathrm{N}(0,1) + 0.5 \cdot \mathrm{N}(4,1)$. The second method is NCE with $M_1 = M_2 = N/2$ noise samples generated from two Gaussian distributions $\mathrm{N}(0,1)$ and $\mathrm{N}(4,1)$. We solved the optimization (3) in NCE by the nonlinear conjugate gradient method (Rasmussen, 2006).

Figure 1 plots the median of the squared errors for θ and c of each estimation method with respect to the sample size N. Here, among the two estimated components that are non-normalized Gaussian distributions, we regarded the one with the smaller mean as the first component $p_1(x \mid \theta_1, c_1)$. For θ, we also plot the median of the squared error of the maximum likelihood estimator computed by the MATLAB function *fitgmdist*. The estimation errors converge to zero for both θ and c, which provides evidence for the consistency of NCE extensions. Also, the estimation accuracy of the second method is slightly better than that of the first method, which is understood as follows. From Theorem 1, the second method is equivalent to NCE with the noise distribution equal to the true data-generating distribution. Therefore, noise in the second method is more difficult to discriminate from data than in the first method.

[1] In practice, using only categories relevant to the new data may suffice and it reduces computational cost.

6 APPLICATION TO IMAGE CLUSTERING

In this section, we apply the proposed method to image clustering. We use the training data of "Dogs vs. Cats" competition at kaggle[2] ($N = 25000$), which consists of 12500 dog images and 12500 cat images. As a pretrained network, we use inception-v3 (Szegedy et al., 2015), which extracts a $d = 2048$ dimensional feature vector from image data. This network was trained for ImageNet competition. For noise samples, we use canine and feline images in the training data of inception-v3[3] ($M = 186125$, $L = 149$). We set the number of clusters to $K = 2$.

We solved the optimization (3) in NCE by the nonlinear conjugate gradient method (Rasmussen, 2006) with 10 random initial values of (θ, c). Among 10 converged solutions, we picked the one with the maximum value of objective function $\hat{J}_{\mathrm{MNCE}}$.

For comparison, we fitted the two-component Gaussian mixture model (GMM) with diagonal covariance matrices to the feature vectors of N images by using the MATLAB function *fitgmdist*. We also fitted the two-component GMM with isotropic covariance matrices by EM algorithm. Although these models also provide clustering, it is heuristic and not probabilistically rigorous.

Table 1 shows the clustering results. Here, we classify an image x into cluster k if $p(z = k \mid x) > 0.5$. In all methods, the two clusters seem to separate dogs and cats well, although the training of inception-v3 was done with more detailed categories like "Scotch terrier" or "snow leopard." Furthermore, the proposed method has better classification accuracy compared to GMMs.

Table 1: Image clustering results. (a) The proposed method. (b) GMM with diagonal covariance matrices. (c) GMM with isotropic covariance matrices.

a)	dog	cat
cluster 1	12400	145
cluster 2	100	12355
b)	dog	cat
cluster 1	12490	325
cluster 2	10	12175
c)	dog	cat
cluster 1	12490	792
cluster 2	10	11708

Figure 2 shows the histogram of the posterior probability in the first cluster $p(z = 1 \mid x)$. While the GMMs cluster all images with high confidence, the proposed method provides more nuanced probabilities. The histogram of the logit score of the posterior probability is given in Supplementary Material.

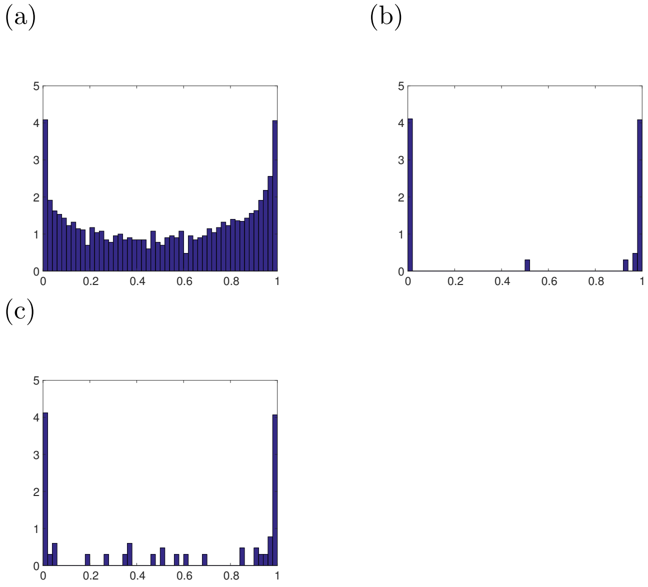

Figure 2: Histogram of the posterior probability $p(z = 1 \mid x)$. The y-axis is in scale $\log_{10}(1+y)$, where y is the frequency. (a) The proposed method. (b) GMM with diagonal covariance matrices. (c) GMM with isotropic covariance matrices.

Next, we show that the proposed method gives a good estimate of the uncertainty of clustering. Table 2 shows the estimated and actual numbers of misclassification. Here, the estimated number of misclassification is defined as the sum of the misclassification probability $\min(p(z = 1 \mid x), p(z = 2 \mid x))$ over all images x. We assume that the learned clusters really coincide with the pre-defined classes of cats and dogs, so we take the "actual" misclassification results in Table 1 as the ground truth. Compared to the proposed method, GMMs significantly underestimate the uncertainty in clustering results. Thus, the proposed method quantifies uncertainty in classification more accurately than GMMs.

7 APPLICATION TO CLUSTERING OF NEUROIMAGING DATA

In this section, we apply the proposed method to clustering of neuroimaging data by using a deep representation obtained by time contrastive learning (Hyvärinen and Morioka, 2016), which is a nonlinear ICA method that finds a representation by solving the

[2] https://www.kaggle.com/c/dogs-vs-cats

[3] From the 152-th category "Chihuahua" to the 300-th category "meerkat". We use only color images.

Table 2: Estimated and actual number of misclassification. (a) The proposed method. (b) GMM with diagonal covariance matrices. (c) GMM with isotropic covariance matrices.

	estimate	actual
(a)	169.98	245
(b)	0.66	335
(c)	5.58	802

task of discriminating time segments.

We used magnetoencephalography (MEG) data from the CamCAN repository [4] (Taylor et al., 2016; Shafto et al., 2014). The data are electromagnetic measurements of the human brain activity taken at a sampling rate of 1000 Hz with 306 channels, consisting of two categories: resting MEG and task MEG. As preprocessing, we applied Morlet filtering around the alpha frequency band, reduced the dimension from 306 to 100 by using PCA, and downsampled to 50 Hz.

First, we trained a three-layer neural network on resting MEG (400 seconds) from four subjects by TCL. Following Hyvärinen and Morioka (2016), the length of each time segment was set to 12.5 seconds (i.e., 625 data points). The number of nodes in each hidden layer was set to 40–20–10. As the activation function, we adopted the maxout units with two affine fully connected weight groups in middle layers and the absolute value units in the output layer. Thus, we obtained a feature extractor that outputs a 10-dimensional feature vector from each data point of 306-ch MEG.

Then, we applied the proposed method to clustering of resting MEG (8000 data points, separate from training data) and task MEG (7000 data points) from one subject. For noise samples in NCE, we used the training data of TCL (see Section 4.3). For comparison, we also fitted the Gaussian mixture model (GMM) with diagonal covariance matrices to the feature vectors of MEG by using the MATLAB function *fitgmdist*. We classified each data point of MEG into the cluster with the maximum posterior probability. Each cluster is considered to represent a brain state.

Figure 3 shows the scatter plots of consecutive posterior probabilities for $K = 2$. It indicates that the posterior changes almost randomly in GMM, which is unrealistic since the brain states should be relatively stable.

Figure 4 shows the histogram of brain states for $K = 10$. Table 3 shows their entropy values. They imply that resting MEG has more temporal variability of brain states than task MEG, especially with the proposed method. This result is consistent with previous findings in neuroscience (Chang and Glover, 2010).

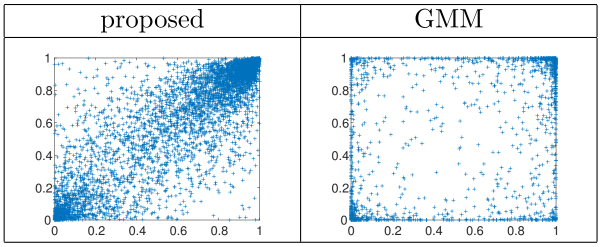

Figure 3: Scatter plots of consecutive posterior probabilities $(p(z_{t-1} = 1 \mid x_{t-1}), p(z_t = 1 \mid x_t))$ for $K = 2$.

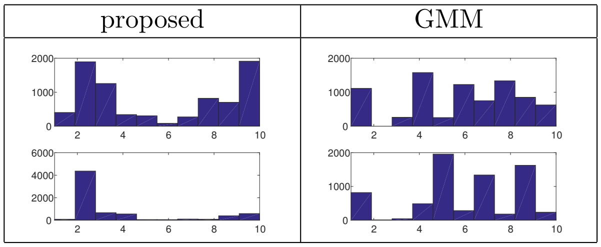

Figure 4: Histogram of the brain states (upper: resting, lower: task) for $K = 10$.

Table 3: Entropy values of the brain states for $K = 10$ (in bits).

	resting	task
proposed	1.996	1.351
GMM	2.062	1.833

8 CONCLUSION

We extended noise contrastive estimation (NCE) to estimate a finite mixture of non-normalized models. Both theory and simulation results showed the validity of this extension of NCE.

Based on the extended NCE, we proposed a method for clustering unlabeled data by using deep representation. Application to clustering of natural images and neuroimaging data gave promising results. In particular, the proposed method was shown to give good estimates of uncertainty in the clustering, in contrast to a heuristic application of Gaussian mixture models. Concurrently to our present contribution, a related method was proposed by Rhodes and Gutmann (2019)[5].

[4] http://www.mrc-cbu.cam.ac.uk/datasets/camcan

[5] We thank Hiroshi Morioka for providing the script for TCL. Also we thank Ricardo Monti for helpful comments.

References

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society B*, **36**, 192–236.

Chang, C. & Glover, G. H. (2010). Time-frequency dynamics of resting-state brain connectivity measured with fMRI. *NeuroImage*, **50**, 81–98.

Dai, J., Lu, Y. & Wu, Y. N. (2015). Generative modeling of convolutional neural networks. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, J. (2014). Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27*.

Gutmann, M. U. & Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for non-normalized statistical models. In *Proceedings of the 13th International Workshop on Artificial Intelligence and Statistics (AISTATS)*.

Gutmann, M. U. & Hyvärinen, A. (2012). Noise-contrastive estimation of non-normalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, **13**, 307–361.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, **14**, 1771–1800.

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

Hyvärinen, A. & Morioka, H. (2016). Unsupervised feature extraction by time-contrastive learning and nonlinear ICA. In *Advances in Neural Information Processing Systems 29*.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*.

Li, S. Z. (2001). *Markov Random Field Modeling in Image Analysis*. Springer.

Mardia, K. V. & Jupp, P. E. (2008). *Directional Statistics*. Wiley.

Nair, V. & Hinton, G. (2008). Implicit mixtures of restricted Boltzmann machines. In *Advances in Neural Information Processing Systems 21*.

Rasmussen, C. E. (2006). Conjugate gradient algorithm. Matlab code version 2006-09-08. `http://learning.eng.cam.ac.uk/carl/code/minimize/minimize.m`

Rhodes, B. & Gutmann, M. U. (2019). Variational noise-contrastive estimation. In *Proceedings of the 22nd International Workshop on Artificial Intelligence and Statistics (AISTATS)*.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). Rethinking the inception architecture for computer vision. arXiv:1512.00567.

Shafto, M. A., Tyler, L. K., Dixon, M., Taylor, J. R., Rowe, J. B., Cusack, R., Calder, A. J., Marslen-Wilson, W. D., Duncan, J., Dalgleish, T., Henson, R. N., Brayne, C., CamCAN & Matthews, F. E. (2014). The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) study protocol: a cross-sectional, lifespan, multidisciplinary examination of healthy cognitive ageing. *BMC Neurology*, **14**, 204.

Taylor, J. R., Williams, N., Cusack, R., Auer, T., Shafto, M. A., Dixon, M., Tyler, L. K., CamCAN & Henson, R. N. (2017). The Cambridge Centre for Ageing and Neuroscience (CamCAN) data repository: Structural and functional MRI, MEG, and cognitive data from a cross-sectional adult lifespan sample. *NeuroImage*, **114**, 262–269.

Teh, Y., Welling, M., Osindero, S. & Hinton, G. E. (2004). Energy-based models for sparse overcomplete representations. *Journal of Machine Learning Research*, **4**, 1235–1260.

Wang, B., Ou, Z. & Tan, Z. (2017). Learning trans-dimensional random fields with applications to language modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **40**, 876–890.

Wang, B. & Ou, Z. (2018). Learning neural trans-dimensional random field language models with noise-contrastive estimation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Xie, J., Lu, Y., Zhu, S. C. & Wu, Y. N. (2016). A theory of generative ConvNet. In *Proceedings of the 33th Annual International Conference on Machine Learning (ICML)*.

Younes, L. (1989). Parametric inference for imperfectly observed gibbsian fields. *Probability Theory and Related Fields*, **82**, 625–645.