Neurocomputing 241 (2017) 90–96

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Asymmetric deep generative models

Harris Partaourides, Sotirios P. Chatzis*

Department of Electrical Engineering, Computer Engineering, and Informatics, Cyprus University of Technology, 33 Saripolou Str., Limassol 3036, Cyprus

ARTICLE INFO

Article history:
Received 8 September 2016
Revised 30 November 2016
Accepted 4 February 2017
Available online 10 February 2017

Communicated by XIANG Xiang Bai

Keywords:
Deep generative model
Variational inference
Restricted multivariate skew-Normal distribution
Semi-supervised learning

ABSTRACT

Amortized variational inference, whereby the inferred latent variable posterior distributions are parameterized by means of neural network functions, has invigorated a new wave of innovation in the field of generative latent variable modeling, giving rise to the family of deep generative models (DGMs). Existing DGM formulations are based on the assumption of a symmetric Gaussian posterior over the model latent variables. This assumption, although mathematically convenient, can be well-expected to undermine the eventually obtained representation power, as it imposes apparent expressiveness limitations. Indeed, it has been recently shown that even some moderate increase in the latent variable posterior expressiveness, obtained by introducing an additional level of dependencies upon auxiliary (Gaussian) latent variables, can result in significant performance improvements in the context of semi-supervised learning tasks. Inspired from these advances, in this paper we examine whether a more potent increase in the expressiveness and representation power of modern DGMs can be achieved by completely relaxing their typical symmetric (Gaussian) latent variable posterior assumptions: Specifically, we consider DGMs with asymmetric posteriors, formulated as restricted multivariate skew-Normal (rMSN) distributions. We derive an efficient amortized variational inference algorithm for the proposed model, and exhibit its superiority over the current state-of-the-art in several semi-supervised learning benchmarks.

1. Introduction

Amortized variational inference [1–4], whereby the inferred latent variable posteriors are parameterized via deep neural networks, is currently at the epicenter of the research on generative latent variable modeling. The class of deep generative models (DGMs) has arisen as the outcome of this research line. Existing DGM formulations postulate symmetric (Gaussian) posteriors over the model latent variables. This assumption, although computationally efficient, may undermine the representation power of DGMs, as it imposes apparent expressiveness limitations [5]. To address these issues, in one of the most recent developments in the field, [6] proposed the skip DGM (SDGM); this model introduces an extra layer of auxiliary latent variables, also imposed symmetric Gaussian posteriors, with the original latent variable posteriors assumed to be conditioned upon the auxiliary latent variables. Apparently, such a hierarchical latent variable construction gives rise to obtained variational posteriors with more expressiveness and representation power. Indeed, [6] have provided broad empirical evidence corroborating these claims, by showing that SDGM yields the state-of-the-art performance in several *semi-supervised learning* benchmarks.

Inspired from these advances, in this paper we examine whether we can achieve a higher level of expressiveness and representation power for the latent variable posteriors of modern DGMs by completely relaxing their typical symmetric (Gaussian) latent variable posterior assumptions. Indeed, in many applied problems, the data to be analyzed may contain a group or groups of observations whose distributions are moderately or severely skewed. Unfortunately, typical DGM formulations based on Gaussian posterior assumptions cannot effectively model data of such nature: A slight deviation from normality may seriously affect the obtained estimates, subsequently misleading inference from the data. Therefore, accounting for asymmetric effects and skewness in the modeled data may allow for significant improvements in the potency of DGM models. On this basis, in this work we introduce the novel class of asymmetric DGMs (AsyDGMs), characterized by asymmetric latent variable posteriors, that are formulated as *restricted multivariate skew-Normal (rMSN) distributions* [7,8].

In recent years, there has been growing interest in studying generative models based on latent variables with skew-elliptical distributions [9,10], both in the univariate and multivariate cases. Their popularity with the statistics community mainly stems from them being regarded as a more general tool for handling heterogeneous data that involve asymmetric behavior across

* Corresponding author.
E-mail addresses: c.partaourides@cut.ac.cy (H. Partaourides), sotirios.chatzis@eecei.cut.ac.cy, soteri0s@me.com (S.P. Chatzis).

http://dx.doi.org/10.1016/j.neucom.2017.02.028
0925-2312/

sub-populations. For instance, [11] and [12] proposed mixtures of multivariate skew-normal and t-distributions based on a restricted variant of the skew-elliptical family of distributions of [8]; [13] gave a systematic overview of various existing multivariate skew distributions and clarified their conditioning-type and convolution-type representations. There also is a small corpus of works proposing generative models the latent variables of which are imposed skewed priors. These are shallow, factor analysis-type models, which provide strong motivation for the work presented in this paper. For instance, [14] proposed mixtures of shifted asymmetric Laplace factor analyzers; [15] proposed mixtures of generalized hyperbolic factor analyzers; [16] proposed mixtures of skew-t factor analyzers. Finally, very recently, a finite mixture model of rMSN-distributed factor analyzers was proposed in [17], and an efficient EM algorithm comprising closed-form updates was derived for model training.

We derive an efficient inference algorithm for the proposed AsyDGM approach by resorting to an elegant amortized variational inference algorithm, similar to existing DGMs. To exhibit the efficacy of our approach, and its superiority over existing symmetrically-distributed DGMs, we perform a series of experimental evaluations. Specifically, we focus on challenging semi-supervised learning tasks, where DGM-type classifier training is performed with a very limited number of labeled examples. We show that our approach yields the state-of-the-art performance in these benchmarks, with a significant improvement over the second-best method.

The remainder of this paper is organized as follows: In the following Section, we provide a brief overview of the theoretical foundation of our work: We first introduce the rMSN distribution; further, we briefly present the SDGM model, which constitutes the latest development in the field of DGMs, when it comes to addressing challenging semi-supervised learning tasks. Next, we introduce our approach, and derive its inference and prediction generation algorithms. In the subsequent experimental Section, we perform an exhaustive empirical evaluation of our approach, using well-known semi-supervised learning benchmarks. Finally, in the concluding Section of this paper, we summarize our contribution and discuss our results.

2. Theoretical foundation

2.1. The rMSN distribution

We begin with a brief review of the rMSN distribution. To establish notation, let $\mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ be the probability density function (pdf) of multivariate Gaussian with mean vector $\boldsymbol{\mu}$ and variance–covariance matrix $\boldsymbol{\Sigma}$, and $\Phi(\cdot)$ be the cumulative distribution function (cdf) of the standard normal distribution. Further, let $TN(\cdot|\mu, \sigma^2; (a, b))$ denote the truncated normal distribution for $\mathcal{N}(\cdot|\mu, \sigma^2)$ lying within a truncated interval (a, b).

Following [12], a random vector $\boldsymbol{x} \in \mathbb{R}^d$ is said to follow an rMSN distribution with location vector $\boldsymbol{\mu}$, dispersion matrix $\boldsymbol{\Sigma}$, and skewness vector $\boldsymbol{\lambda}$, denoted by $\boldsymbol{x} \sim \text{rSN}(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\lambda})$, if it can be represented as

$$\begin{aligned} \boldsymbol{x}|u &\sim \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{\lambda}u, \boldsymbol{\Sigma}) \\ u &\sim TN(0, 1; (0, \infty)) \end{aligned} \tag{1}$$

Here, the truncated Normal distribution $TN(u|\mu_u, \sigma_u^2; (0, \infty))$ with mean μ_u, variance σ_u^2, and bounds in $(0, \infty)$, is defined as

$$TN(u|\mu_u, \sigma_u^2; (0, \infty)) = \frac{\mathcal{N}(u|\mu_u, \sigma_u^2)}{\Phi(\mu_u/\sigma_u)} I(u > 0) \tag{2}$$

where $I(\cdot)$ is an indicator function. Hence, we observe that an rMSN-distributed variable can be equivalently expressed under a Gaussian conditional distribution, where the introduced conditioning latent variable follows a standard truncated normal density.

On this basis, [17] have recently proposed a generalization of the traditional factor analysis (FA) model, namely the SNFA model, where the latent variables (factors) are assumed to follow an rMSN distribution within the family defined by (1). Let us denote as $\boldsymbol{x} \in \mathbb{R}^p$ the p-dimensional observations we wish to model via an SNFA model. Denoting as $\boldsymbol{z} \in \mathbb{R}^q$ the inferred latent factors vectors ($q < p$), we have [17]

$$p(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{z}, \boldsymbol{D}) \tag{3}$$

and

$$p(\boldsymbol{z}) = \text{rSN}(\boldsymbol{z}| - c\boldsymbol{\Delta}^{-1/2}\boldsymbol{\lambda}, \boldsymbol{\Delta}^{-1}, \boldsymbol{\Delta}^{-1/2}\boldsymbol{\lambda}) \tag{4}$$

where $\boldsymbol{\mu}$ is a p-dimensional location vector, $\boldsymbol{B}$ is a parameter matrix of the SNFA model (factor loadings), $\boldsymbol{D}$ is a *diagonal* covariance matrix, $c \triangleq \sqrt{2/\pi}$, $\boldsymbol{\lambda}$ is the skewness vector of the model, and

$$\boldsymbol{\Delta} \triangleq \boldsymbol{I} + (1 - c^2)\boldsymbol{\lambda}\boldsymbol{\lambda}^T \tag{5}$$

As shown in [17], by using the definition (1) of the rMSN distribution, this asymmetric factor analysis model can be equivalently expressed under the following three-level hierarchical representation:

$$p(\boldsymbol{x}|\tilde{\boldsymbol{z}}) = \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{B}g(\tilde{\boldsymbol{z}}), \boldsymbol{D}) \tag{6}$$

$$p(\tilde{\boldsymbol{z}}|u) = \mathcal{N}(\tilde{\boldsymbol{z}}|(u - c)\boldsymbol{\lambda}, \boldsymbol{I}) \tag{7}$$

and

$$u \sim TN(0, 1; (0, \infty)) \tag{8}$$

where

$$g(\tilde{\boldsymbol{z}}) = \boldsymbol{\Delta}^{-1/2}\tilde{\boldsymbol{z}} \tag{9}$$

Under this equivalent representation, the SNFA model yields a simple prior formulation that is amenable to a computationally efficient EM training algorithm with closed-form expressions [17].

2.2. SDGM

As discussed in the Introduction, modern developments in the field of variational inference focus on using deep learning techniques to parameterize the variational posteriors of latent variable models. This gives rise to powerful probabilistic models, usually referred to as DGMs, constructed by an inference neural network that parameterizes the posterior $q(\boldsymbol{z}|\boldsymbol{x})$, and a generative neural network that parameterizes the conditional likelihood $p(\boldsymbol{x}|\boldsymbol{z})$.

To allow for keeping the computational requirements low, the variational distribution $q(\boldsymbol{z}|\boldsymbol{x})$ is usually chosen to be a diagonal Gaussian. Despite the computational attractiveness of this approximation, it is quite apparent though that such an assumption may not allow for capturing intricate latent dynamics in the modeled data, as well as modeling data of asymmetric nature. Hence, one could expect that by relaxing these diagonal Gaussian posterior assumptions, one may yield DGMs with increased expressive power.

Recently, [6] proposed an way of ameliorating these issues of DGMs by drawing inspiration from the variational auxiliary variable approach of [18]. The so-obtained *Skip-DGM* (SDGM) extends the variational distribution with some auxiliary variables $\boldsymbol{a}$, such that

$$q(\boldsymbol{a}, \boldsymbol{z}|\boldsymbol{x}) = q(\boldsymbol{z}|\boldsymbol{a}; \boldsymbol{x})q(\boldsymbol{a}|\boldsymbol{x}) \tag{10}$$

and

$$q(\boldsymbol{z}|\boldsymbol{x}) = \int q(\boldsymbol{a}, \boldsymbol{z}|\boldsymbol{x}) \mathrm{d}\boldsymbol{a} \tag{11}$$

where both the postulated variational posteriors $q(\boldsymbol{z}|\boldsymbol{a};\ \boldsymbol{x})$ and $q(\boldsymbol{a}|\boldsymbol{x})$ are typical inference networks with diagonal Gaussian form. Under such a two-level hierarchical formulation, it becomes well-expected that the marginal distribution $q(\boldsymbol{z}|\boldsymbol{x})$ will be able to fit more complicated posteriors compared to conventional (one-level) diagonal Gaussian distribution-based DGM formulations. Indeed, [6] have shown that the SDGM approach can be utilized in the context of semi-supervised learning tasks, with the goal of building a potent classifier, capable of obtaining state-of-the-art performance in challenging datasets by being trained with limited labeled examples combined with large unlabeled datasets.

More specifically, denoting as $\boldsymbol{x}$ the observed vectors presented as input to the postulated classifier, and as y the corresponding label variables, SDGM consecutively postulates the following generative (i.e., conditional likelihood and prior) assumptions [6]:

$$p_{\theta}(\boldsymbol{x}|\boldsymbol{a},\boldsymbol{z},y) = f(\boldsymbol{x},\boldsymbol{a},\boldsymbol{z},y;\boldsymbol{\theta}) \tag{12}$$

$$p_{\theta}(\boldsymbol{a}|\boldsymbol{z},y) = f(\boldsymbol{a},\boldsymbol{z},y;\boldsymbol{\theta}) \tag{13}$$

$$p(y) = \mathrm{Cat}(y|\boldsymbol{\pi}) \tag{14}$$

$$p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{0},\boldsymbol{I}) \tag{15}$$

where $f(\boldsymbol{x},\ \boldsymbol{a},\ \boldsymbol{z},\ y;\ \boldsymbol{\theta})$ is a categorical or diagonal Gaussian for discrete and continuous observations $\boldsymbol{x}$, respectively, and $f(\boldsymbol{a},\ \boldsymbol{z},\ y;\ \boldsymbol{\theta})$ is a diagonal Gaussian. Both $p_{\theta}(\cdot)$ distributions are parameterized by deep neural networks with parameters $\boldsymbol{\theta}$. On the other hand, the derived variational posteriors (inference model) are assumed to take on the following form:

$$q_{\phi}(\boldsymbol{a}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{a}|\boldsymbol{\mu}(\boldsymbol{x};\boldsymbol{\phi}), \mathrm{diag}\,\boldsymbol{\sigma}^2(\boldsymbol{x};\boldsymbol{\phi})) \tag{16}$$

$$q_{\phi}(\boldsymbol{z}|\boldsymbol{a},\boldsymbol{x},y) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}(\boldsymbol{a},\boldsymbol{x},y;\boldsymbol{\phi}), \mathrm{diag}\,\boldsymbol{\sigma}^2(\boldsymbol{a},\boldsymbol{x},y;\boldsymbol{\phi})) \tag{17}$$

$$q_{\phi}(y|\boldsymbol{a},\boldsymbol{x}) = \mathrm{Cat}(y|\boldsymbol{\pi}(\boldsymbol{a},\boldsymbol{x};\boldsymbol{\phi})) \tag{18}$$

[the computed (output) probabilities of the network $\boldsymbol{\pi}(\boldsymbol{a},\ \boldsymbol{x};\ \boldsymbol{\phi})$ are the ones used to perform the classification task]. Note that, in order to parameterize the diagonal Gaussians $p_{\theta}(\cdot)$ and $q_{\phi}(\cdot)$ in Eqs. (12)–(13) and (16)–(17), respectively, [6] define two separate outputs from the top deterministic layer in the corresponding deep neural networks, one for the distribution mean and one for the distribution (log-)variance.

An issue variational inference for DGM-type models, including SDGM, is confronted with is the analytical intractability of the expressions of the entailed expectations of the model latent variables w.r.t. the sought approximate (variational) posteriors. This is due to their nonconjugate formulation, as a consequence of their nonlinear parameterization via deep neural networks. Specifically, considering a training set $\mathcal{D} = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N}$ comprising N samples, the expression of the evidence lower bound (ELBO) of SDGM yields:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi}|\mathcal{D}) = \sum_{n=1}^{N}\Big\{ &- \mathrm{KL}\big[q_{\phi}(\boldsymbol{z}_n|\boldsymbol{a}_n,\boldsymbol{x}_n,y_n)||p(\boldsymbol{z}_n)\big] \\ &- \mathrm{KL}\big[q_{\phi}(\boldsymbol{a}_n|\boldsymbol{x}_n)||p_{\theta}(\boldsymbol{a}_n|\boldsymbol{z}_n,y_n)\big] \\ &- \mathrm{KL}\big[q_{\phi}(y_n|\boldsymbol{a}_n,\boldsymbol{x}_n)||p(y_n)\big] \\ &+ \mathbb{E}_{q_{\phi}(\boldsymbol{z},\boldsymbol{a}|\boldsymbol{x},y)}[\log p_{\theta}(\boldsymbol{x}_n|\boldsymbol{a}_n,\boldsymbol{z}_n,y_n)]\Big\}\end{aligned} \tag{19}$$

where $\mathrm{KL}[q||p]$ is the KL divergence between the distribution $q(\cdot)$ and the distribution $p(\cdot)$. Under the assumed nonlinear (hence, nonconjugate) model construction, it is easy to observe that neither the ELBO $\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi}|\mathcal{D})$ nor its derivatives w.r.t. the parameter sets $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ can be computed analytically. In addition, opting for a naive Monte Carlo gradient estimator is not an option in our context, due to its entailed prohibitively high variance that renders it completely impractical for our purposes [19].

These issues can be addressed by resorting to the popular reparameterization trick [1], commonly employed in the context of amortized variational inference. This consists in approximating the posterior expectations in (19) as averages over a set of L samples from the corresponding Gaussian posteriors, $\{\boldsymbol{a}_n^{(l)}, \boldsymbol{z}_n^{(l)}\}_{l=1}^{L}$; the latter samples are expressed as differentiable transformations of the form $\boldsymbol{\xi}_{\phi}(\boldsymbol{\epsilon})$ of the posterior parameters $\boldsymbol{\phi}$ given some random noise input $\boldsymbol{\epsilon}$. Specifically, we have [6]:

$$\boldsymbol{a}^{(l)} = \boldsymbol{\xi}_{\phi}(\boldsymbol{\epsilon}^{(l)};\boldsymbol{x}) = \boldsymbol{\mu}(\boldsymbol{x};\boldsymbol{\phi}) + \boldsymbol{\sigma}(\boldsymbol{x};\boldsymbol{\phi}) \circ \boldsymbol{\epsilon}^{(l)} \tag{20}$$

and

$$\begin{aligned}\boldsymbol{z}^{(l)} &= \boldsymbol{\xi}_{\phi}(\boldsymbol{\epsilon}^{(l)};\boldsymbol{a},\boldsymbol{x},y) \\ &= \boldsymbol{\mu}(\boldsymbol{a}^{(l)},\boldsymbol{x},y;\boldsymbol{\phi}) + \boldsymbol{\sigma}(\boldsymbol{a}^{(l)},\boldsymbol{x},y;\boldsymbol{\phi}) \circ \boldsymbol{\epsilon}^{(l)}\end{aligned} \tag{21}$$

where $\circ$ is the elementwise product, and the $\boldsymbol{\epsilon}^{(l)}$ are white random noise samples with unitary variance, i.e. $\boldsymbol{\epsilon}^{(l)} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{I})$.

3. Proposed approach

In the following, we introduce a DGM the latent variables of which are assumed to follow rMSN distributions. Since in this work we are interested in semi-supervised learning tasks, our exposition and derivations will be performed in the context of the graphical model of SDGM. However, a similar asymmetric modeling scheme can be employed in the context of any desired graphical formulation for a postulated DGM.

To define our model, we elect to express the rMSN-distributed latent variables under the equivalent three-level hierarchical representation scheme adopted by Lin et al. [17], described by Eqs. (6)–(9). On this basis, our proposed AsyDGM consecutively postulates the following generative (i.e., conditional likelihood and prior) assumptions:

$$p_{\theta}(\boldsymbol{x}|\boldsymbol{a},\boldsymbol{z},y) = f(\boldsymbol{x},g(\boldsymbol{a}),g(\boldsymbol{z}),y;\boldsymbol{\theta}) \tag{22}$$

$$p_{\theta}(\boldsymbol{a}|\boldsymbol{z},u,y) = f(\boldsymbol{a},g(\boldsymbol{z}),u,y;\boldsymbol{\theta}) \tag{23}$$

$$p(y) = \mathrm{Cat}(y|\boldsymbol{\pi}) \tag{24}$$

$$p(\boldsymbol{z}|u) = \mathcal{N}(\boldsymbol{z}|(u-c)\boldsymbol{\lambda},\boldsymbol{I}) \tag{25}$$

$$p(u) = TN(u|0,1;(0,\infty)) \tag{26}$$

where [denoting $\boldsymbol{\xi} \in \{\boldsymbol{z},\ \boldsymbol{a}\}$]:

$$g(\boldsymbol{\xi}) \triangleq \boldsymbol{\Delta}^{-1/2}\boldsymbol{\xi} \tag{27}$$

$$\boldsymbol{\Delta} \triangleq \boldsymbol{I} + (1-c^2)\boldsymbol{\lambda}\boldsymbol{\lambda}^T \tag{28}$$

$c \triangleq \sqrt{2/\pi}$, $\boldsymbol{\lambda}$ is the skewness vector of the model, and the pdf's $f(\cdot)$ in (22) and (23) are defined similar to SDGM.

On this basis, the derived variational posteriors (inference model) of AsyDGM are assumed to take on the following form:

$$q_{\phi}(y|\boldsymbol{a},\boldsymbol{x}) = \mathrm{Cat}(y|\boldsymbol{\pi}(\boldsymbol{a},\boldsymbol{x};\boldsymbol{\phi})) \tag{29}$$

$$q_{\phi}(\boldsymbol{a}|\boldsymbol{x},u) = \mathcal{N}(\boldsymbol{a}|\boldsymbol{\mu}(\boldsymbol{x},u;\boldsymbol{\phi}), \mathrm{diag}\,\boldsymbol{\sigma}^2(\boldsymbol{x},u;\boldsymbol{\phi})) \tag{30}$$

$$q_{\phi}(\boldsymbol{z}|u,\boldsymbol{a},\boldsymbol{x},y) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}(u,\boldsymbol{a},\boldsymbol{x},y;\boldsymbol{\phi}), \mathrm{diag}\,\boldsymbol{\sigma}^2(u,\boldsymbol{a},\boldsymbol{x},y;\boldsymbol{\phi})) \tag{31}$$

and

$$q_{\phi}(u|\boldsymbol{x}) = TN(u|m(\boldsymbol{x};\boldsymbol{\phi}), s^2(\boldsymbol{x};\boldsymbol{\phi}), (0,\infty)) \tag{32}$$

where

$$\boldsymbol{\mu}(u,\boldsymbol{a},\boldsymbol{x},y;\boldsymbol{\phi}) = \boldsymbol{\mu}(u,\boldsymbol{h}(\boldsymbol{a},\boldsymbol{x},y);\boldsymbol{\phi}) \tag{33}$$

and

$$\boldsymbol{\sigma}^2(u,\boldsymbol{a},\boldsymbol{x},y;\boldsymbol{\phi}) = \boldsymbol{\sigma}^2(u,\boldsymbol{h}(\boldsymbol{a},\boldsymbol{x},y);\boldsymbol{\phi}) \tag{34}$$

All the postulated generative and inference networks, which parameterize the generative components $p_{\boldsymbol{\theta}}(\cdot)$ and the variational posteriors $q_{\boldsymbol{\phi}}(\cdot)$ of AsyDGM, constitute deep neural networks. In cases of Gaussian or truncated Gaussian densities, these networks define two separate outputs from their top deterministic layer, one for the distribution mean and one for the distribution (log-)variance. Specifically, we have

$$\boldsymbol{\mu}(\boldsymbol{x}, u; \boldsymbol{\phi}) = \text{Linear}(u, \boldsymbol{h}_1(\boldsymbol{x})) \tag{35}$$

$$\boldsymbol{\sigma}^2(\boldsymbol{x}, u; \boldsymbol{\phi}) = \exp(\text{Linear}(u, \boldsymbol{h}_1(\boldsymbol{x}))) \tag{36}$$

$$\boldsymbol{\pi}(\boldsymbol{a}, \boldsymbol{x}; \boldsymbol{\phi}) = \text{Softmax}(\boldsymbol{h}_2(\boldsymbol{a}, \boldsymbol{x})) \tag{37}$$

$$\boldsymbol{m}(\boldsymbol{x}; \boldsymbol{\phi}) = \text{Linear}(\boldsymbol{h}_3(\boldsymbol{x})) \tag{38}$$

$$\boldsymbol{s}^2(\boldsymbol{x}; \boldsymbol{\phi}) = \exp(\text{Linear}(\boldsymbol{h}_3(\boldsymbol{x}))) \tag{39}$$

$$\boldsymbol{\mu}(u, \boldsymbol{a}, \boldsymbol{x}, y; \boldsymbol{\phi}) = \text{Linear}(u, \boldsymbol{h}(\boldsymbol{a}, \boldsymbol{x}, y)) \tag{40}$$

$$\boldsymbol{\sigma}^2(u, \boldsymbol{a}, \boldsymbol{x}, y; \boldsymbol{\phi}) = \exp(\text{Linear}(u, \boldsymbol{h}(\boldsymbol{a}, \boldsymbol{x}, y))) \tag{41}$$

where Linear$(\cdot)$ is a linear layer, Softmax$(\cdot)$ is a softmax layer, and the $\boldsymbol{h}_1(\cdot)$, $\boldsymbol{h}_2(\cdot)$, $\boldsymbol{h}_3(\cdot)$, and $\boldsymbol{h}(\cdot)$ are deep neural networks.

Note also our introduced linear dependence assumptions for the mean and variance of $q_{\boldsymbol{\phi}}(\boldsymbol{z}|u, \boldsymbol{a}, \boldsymbol{x}, y)$ and $q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x}, u)$ upon the latent variable u [Eqs. (40)–(41) and (35)–(36), respectively]. This selection is motivated by the related derivations that apply to the case of simple factor analysis-type models postulating rMSN-distributed latent factors, e.g. [17]. In addition, it facilitates the derivation of a computationally efficient inference algorithm for the proposed model. Specifically, the ELBO expression of AsyDGM can be shown to yield

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}|\mathcal{D}) = \sum_{n=1}^{N} \Big\{ & - \text{KL}\big[q_{\boldsymbol{\phi}}(u_n|\boldsymbol{x}_n)||p(u_n)\big] \\ & - \text{KL}\big[q_{\boldsymbol{\phi}}(\boldsymbol{z}_n|u_n, \boldsymbol{a}_n, \boldsymbol{x}_n, y_n)||p(\boldsymbol{z}_n|u_n)\big] \\ & - \text{KL}\big[q_{\boldsymbol{\phi}}(\boldsymbol{a}_n|\boldsymbol{x}_n, u_n)||p_{\boldsymbol{\theta}}(\boldsymbol{a}_n|\boldsymbol{z}_n, u_n, y_n)\big] \\ & - \text{KL}\big[q_{\boldsymbol{\phi}}(y_n|\boldsymbol{a}_n, \boldsymbol{x}_n)||p(y_n)\big] \\ & + \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}), q_{\boldsymbol{\phi}}(\boldsymbol{a})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_n|\boldsymbol{a}_n, \boldsymbol{z}_n, y_n)] \Big\}\end{aligned} \tag{42}$$

Computation of the term KL$[q_{\boldsymbol{\phi}}(u_n|\boldsymbol{x}_n)||p(u_n)]$ in (42) can be tractably performed in an analytical fashion. In addition, due to the aforementioned linear dependence scheme, the same holds for the posterior expectations w.r.t. $q(u)$ which are entailed in the computation of KL$[q_{\boldsymbol{\phi}}(\boldsymbol{z}_n|u_n, \boldsymbol{a}_n, \boldsymbol{x}_n, y_n)||p(\boldsymbol{z}_n|u_n)]$ and KL$[q_{\boldsymbol{\phi}}(\boldsymbol{a}_n|\boldsymbol{x}_n, u_n)||p_{\boldsymbol{\theta}}(\boldsymbol{a}_n|\boldsymbol{z}_n, u_n, y_n)]$. This way, the need of applying the reparameterization trick in the context of the inference algorithm of AsyDGM is limited to the latent vectors $\boldsymbol{z}$ and $\boldsymbol{a}$ (similar to SDGM). This clearly facilitates computational efficiency for our method, since application of the reparameterization trick in the case of truncated normal distributions would require computation of Gaussian quantile functions, which is quite complex.

4. Experimental evaluation

To exhibit the efficacy of our approach, we perform evaluation in a series of challenging semi-supervised learning tasks. We especially focus on tasks that entail high-dimensional observations with several artifacts that render the Gaussian assumption too simplistic; these include both skewness and outliers. In the experimental evaluations of Sections 4.1 and 4.2, we randomly split the available datasets into a training set and a test set that contain half of the available video frames in each case. In the experimental evaluations of Sections 4.1–4.3, we retain a randomly selected 10% of the available training data labels, and we discard the rest; the used deep neural networks [denoted as $\boldsymbol{h}_k(\cdot)$ in Eqs. (35)–(41)] comprise two fully connected hidden layers, with 50 ReLU [20] units each, while the size of the latent vectors $\boldsymbol{z}$, as well as the auxiliary latent vectors, $\boldsymbol{a}$, is set to 50. In the experimental evaluations of Section 4.4, we use the available splits of the considered datasets into a training set and a test set; network configuration is adopted from [6], while the number of retained training data labels is provided in Table 3.

In all cases, to alleviate the effect of this random dataset selection, we repeat our experiments 50 times, with different splits of the data each time. To provide some comparative results, apart from our method we also evaluate in the same experiments some alternative DGM-type models, recently proposed for addressing the problem of semi-supervised learning. Specifically, we compare to the closely-related SGDM method [6], the M1+M2 and M1+TSVM approaches proposed in [2], and the VAT approach recently presented in [21].

In all our experiments, the matrix power $\boldsymbol{\Delta}^{-1/2}$ entailed in (27) is approximated by means of a first-order Taylor expansion; this facilitates computational efficiency. Specifically, we have

$$\begin{aligned}\boldsymbol{\Delta}^{-1/2} &= (\boldsymbol{I} + (1 - c^2)\boldsymbol{\lambda}\boldsymbol{\lambda}^T)^{-1/2} \\ &\approx \boldsymbol{I} - \frac{1}{2}(1 - c^2)\boldsymbol{\lambda}\boldsymbol{\lambda}^T\end{aligned} \tag{43}$$

To optimize the ELBO $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}|\mathcal{D})$ of our model w.r.t. its trainable parameters, we resort to the Adam optimization algorithm [22]; we use a learning rate of 3×10^{-4}, and an exponential decay rate for the first and second moment at 0.9 and 0.999, respectively. Initialization of the network parameters is performed by adopting a Glorot-style uniform initialization scheme [23]. Model training is performed using only one sample from $q_{\boldsymbol{\phi}}(\boldsymbol{z})$ and $q_{\boldsymbol{\phi}}(\boldsymbol{a})$, i.e. $L = 1$; it is currently well-known that using $L > 1$ samples in the context of amortized variational inference does not yield any noticeable improvement over $L = 1$, as long as the used batch-size is quite large (effectively, at least 100 samples) [1,2,4].

Our source codes have been developed in Python, and make use of the Theano[1] [24], Lasagne[2], and Parmesan[3] libraries, as well as source code from the authors of [6][4].

4.1. Workflow recognition dataset

We first consider a public benchmark dataset involving action recognition of humans, namely the *Workflow Recognition* database [25]. Specifically, we use the first two workflows pertaining to car assembly (see [25] for more details). The frame-level *tasks to recognize* in these workflows are the following:

1. Worker 1 picks up part 1 from rack 1 (upper) and places it on the welding cell; mean duration is 8–10 s.
2. Worker 1 and worker 2 pick part 2a from rack 2 and place it on the welding cell.

[1] http://deeplearning.net/software/theano/
[2] https://github.com/Lasagne/Lasagne
[3] https://github.com/casperkaae/parmesan
[4] https://github.com/larsmaaloee/auxiliary-deep-generative-models

Table 1
Activity recognition experiments: Test error (%) of the evaluated methods (means and standard deviations over multiple repetitions).

Method	Workflow recognition	Honeybee dance
M1+TSVM	22.12 (± 0.05)	45.48 (± 0.11)
M1+M2	20.58 (± 0.05)	38.62 (± 0.10)
VAT	17.29 (± 0.05)	36.13 (± 0.14)
SDGM	13.90 (± 0.04)	30.38 (± 0.14)
AsyDGM	13.02 (± 0.03)	24.11 (± 0.12)

3. Worker 1 and worker 2 pick part 2b from rack 3 and place it on the welding cell.
4. Worker 2 picks up spare parts 3a, 3b from rack 4 and places them on the welding cell.
5. Worker 2 picks up spare part 4 from rack 1 and places it on the welding cell.
6. Worker 1 and worker 2 pick up part 5 from rack 5 and place it on the welding cell.

Feature extraction is performed as follows: To extract the spatiotemporal variations, *we use pixel change history images to capture the motion history* (see, e.g., [26]), and compute the complex Zernike moments A_{00}, A_{11}, A_{20}, A_{22}, A_{31}, A_{33}, A_{40}, A_{42}, A_{44}, A_{51}, A_{53}, A_{55}, A_{60}, A_{62}, A_{64}, A_{66}, for each of which we compute the norm and the angle. Additionally the center of gravity and the area of the found blobs are also used. In total, this feature extraction procedure results in *31-dimensional observation vectors.* Zernike moments are calculated in rectangular regions of interest of approximately 15K pixels in each image to limit the processing and allow real time feature extraction (performed at a rate of approximately 50–60 fps). In our experiments, we use a total of *40 sequences* representing full assembly cycles and containing at least one of the considered behaviors, with *each sequence being approximately 1K frames long*. Frame annotation has been performed manually. We provide the so-obtained test error rates of the evaluated methods in Table 1. As we observe, our approach yields a statistically significant improvement over the competition.

4.2. Honeybee dance dataset

Further, we evaluate our method using the *Honeybee Dance* dataset [27]; it contains video sequences of honeybees which communicate the location and distance to a food source through a dance that takes place within the hive. The dance can be decomposed into *three* different movement patterns that must be recognized by the evaluated algorithms: *waggle, right-turn,* and*left-turn*. During the waggle dance, the bee moves roughly in a straight line while rapidly shaking its body from left to right; the duration and orientation of this phase correspond to the distance and the orientation to the food source. At the endpoint of a waggle dance, the bee turns in a clockwise or counter-clockwise direction to form a turning dance. Our dataset consists of *six video sequences with lengths 1058, 1125, 1054, 757, 609, and 814 frames, respectively,* and is based on the *raw pixel change history images*, without further preprocessing, contrary to the previous experiment; this renders this experimental scenario more challenging for all the evaluated deep generative models. The obtained results are provided in Table 1. We observe that our approach yields a clear improvement over the competition, including an almost 20% improvement over the second best performing method.

4.3. Yearly song classification using audio features

In this experiment, we consider application of our method to automatic prediction of a song track's release year. This problem entails surprisingly challenging complexity issues, stemming from the great diversity of style and genre of the songs released each year. Under this motivation, we utilize a subset of the "Million song dataset" benchmark [28], which comprises 515,345 tracks with available release year information (both training and test sets). The tracks are mostly western, commercial tracks ranging from 1922 to 2011, with a peak in the year 2000 and onwards. Apart from the year, the dataset provides 90 additional representative features; of these 90 attributes, 12 are timbre average and 78 are timbre covariance, all extracted from the timbre features. We use these 90-dimensional feature vectors as the observations presented to the evaluated methods.

In our experiments, our goal is to differentiate between songs written in the 1980s, 1990s, and 2000s. For this purpose, we randomly select 10% of the training set songs released in these decades as our *labeled* training data; the remainder of the available training data pertaining to these decades is used as our *unlabeled* training dataset (i.e., with their labels considered missing). Subsequently, all methods are evaluated on the grounds of correctly classifying the test set tracks (included in the dataset) that pertain to these three decades. The obtained results are provided in Table 2. As we observe, the proposed approach outperforms all its competitors, yielding notable and statistically significant performance differences.

Table 2
Song classification experiments: Test error (%) of the evaluated methods (means and standard deviations over multiple repetitions).

Method	Performance
M1+TSVM	38.12 (± 0.12)
M1+M2	36.49 (± 0.13)
VAT	37.44 (± 0.13)
SDGM	33.16 (± 0.11)
AsyDGM	28.30 (± 0.10)

Table 3
Image classification benchmarks: Test error (%) of the evaluated methods (means and standard deviations over multiple repetitions).

Method #Training labels	MNIST 100	NORB 1000
M1+TSVM	11.82 (± 0.25)	18.79 (± 0.05)
M1+M2	3.33 (± 0.14)	–
VAT	2.12	9.88
SDGM	1.32 (± 0.07)	9.40 (± 0.04)
AsyDGM	1.34 (± 0.08)	9.03 (± 0.02)

4.4. Image classification benchmarks

Finally, we evaluate our method on two popular benchmark datasets dealing with image classification, namely MNIST and *small* NORB. The popularity of these datasets facilitates transparency in our comparisons with the existing literature. MNIST comprises a total of 60,000 training samples, which constitute images of handwritten digits, with size 28× 28. On the other hand, the small NORB dataset comprises 24,300 training samples and an equal amount of test samples; these constitute images of size 32× 32, and are distributed across 5 classes: animal, human, plane, truck, car.

In Table 3, we report the obtained performance of our method, alongside the number of retained training data labels in each case. We also cite the performances of related methods reported in the recent literature. As we observe, our method turns out to yield re-

sults merely comparable to SDGM in the case of the MNIST dataset. This outcome is probably reasonable, since MNIST is a rather easy dataset, with clear underlying structural patterns, and absence of artifacts such as skewness or outliers. Therefore, one would not expect substantial room for improvement obtained by means of a method designed to account for such artifacts.

The obtained comparative empirical outcome changes in the case of the NORB dataset, where our method does yield a statistically significant performance improvement over the second best performing method. Indeed, one could claim that this performance difference is not as high as in the previously considered experimental scenarios. We argue though that this outcome could be easily expected: The nature of NORB, which comprises images of some simple objects without significant clutter, is much less likely to give rise to modeling problems related with skewness, atypical data, and outliers. Such problems though can become extremely prominent when dealing with noisy signals such as music, as well as when dealing with activity recognition in video sequences, where such artifacts are much more common.

4.5. A note on computational complexity

We underline that the extra computational costs of our method are solely associated with learning of the skewness vectors $\boldsymbol{\lambda}$. These costs are only limited to the training algorithm of the model, and do not constitute a significant complexity increase, due to our approximation (43). Hence, computational complexity for the training algorithm of our method is comparable to SDGM; indeed, we have experimentally observed requirements of the same order of magnitude in computational time. Note also that training algorithm convergence has been empirically found to be similarly fast in both the cases of our model and of its main competitor, i.e. SDGM, in all the conducted experiments. On the other hand, the computational performance of our method in test time is (almost) identical to SDGM, since both approaches essentially require the same set of *feedforward* computations.

5. Conclusions

This paper constitutes an attempt to increase the effectiveness and representation power of the learned latent variable posteriors of DGMs in a principled, rather than an *ad hoc*, fashion. To this end, we drew inspiration from recent developments in the field of multivariate analysis: It has been recently shown that shallow, factor analysis-type, latent variable models are capable of yielding a significantly increased representation power by postulating latent variables with skew-elliptical distributions. On this basis, we examined whether similar benefits could be obtained for DGMs, by introducing an asymmetric DGM formulation, based on rMSN-distributed latent variables.

Since in this work we focused on the problem of semi-supervised learning, we exhibited the derivation of our approach in the context of a graphical formulation also adopted by the recently proposed SDGM approach. To allow for the derivation of an elegant inference algorithm for our model, we utilized a three-level hierarchical representation of the rMSN distribution, inspired from [17]. We examined the efficacy of our approach in several experimental scenarios, using benchmark datasets. As we showed, our method proves to be more effective than the competition in terms of modeling and predictive performance when artifacts such as skewness and outliers are prevalent in the observed data. These empirical results corroborate our theoretical claims.

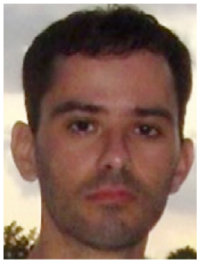
Acknowledgment

We gratefully acknowledge the support of NVIDIA Corporation with the donation of one Tesla K40 GPU used for this research.

References

[1] D. Kingma, M. Welling, Auto-encoding variational Bayes, in: Proceedings of ICLR, 2014.
[2] D.P. Kingma, D.J. Rezende, S. Mohamed, M. Welling, Semi-supervised learning with deep generative models, in: Proceedings of NIPS, 2014.
[3] D.J. Rezende, S. Mohamed, D. Wierstra, Stochastic backpropagation and approximate inference in deep generative models, in: Proceedings of ICML, 2014.
[4] D.J. Rezende, S. Mohamed, Variational inference with normalizing flows, in: Proceedings of ICML, 2015.
[5] R.E. Turner, M. Sahani, Two problems with variational expectation maximisation for time-series models, in: D. Barber, T. Cemgil, S. Chiappa (Eds.), Bayesian Time series models, chapter 5, pp. 109–130, Cambridge University Press, 2011, pp. 109–130. Cambridge University Press, 2011, Ch. 5.
[6] L. Maaløe, C.K. Sønderby, S.K. Sønderby, O. Winther, Auxiliary deep generative models, in: Proceedings of ICML, 2016.
[7] A. Azzalini, A.D. Valle, The multivariate skew-normal distribution, Biometrika 83 (1996) 715–726.
[8] S. Sahu, D. Dey, M. Branco, A new class of multivariate skew distributions with application to Bayesian regression models, Can. J. Stat. 31 (2003) 129–150.
[9] T. Lin, Maximum likelihood estimation for multivariate skew normal mixture models, J. Multivar. Anal. 100 (2009) 257–265.
[10] T.I. Lin, J. Lee, S. Yen, Finite mixture modelling using the skew normal distribution, Stat. Sin. 17 (2007) 909–927.
[11] S. Pyne, X. Hu, K. Wang, E. Rossin, T. Lin, L. Maier, C. Baecher-Allan, G. McLachlan, P. Tamayo, D. Hafler, P.D. Jager, J. Mesirov, Automated high- dimensional flow cytometric data analysis, Proc. Natl. Acad. Sci 106 (2009) 8519–8524.
[12] S. Lee, G. McLachlan, On mixtures of skew normal and skew t-distributions, Adv. Data Anal. Classif. 7 (2013) 241–266.
[13] S. Lee, G. McLachlan, Finite mixtures of multivariate skew t-distributions: some recent and new results, Stat. Comput. 24 (2014) 181–202.
[14] B. Franczak, P. McNicholas, R. Browne, P. Murray, Parsimonious shifted asymmetric Laplace mixtures, IEEE Trans. Pattern Anal. Mach. Intell. 36 (6) (2013) 1149–1157.
[15] C. Tortora, P.D. McNicholas, R.P. Browne, A mixture of generalized hyperbolic factor analyzers, Adv. Data Anal. Classif. 10 (4) (2016) 423–440, doi:10.1007/s11634-015-0204-z.
[16] A. Montanari, C. Viroli, A skew-normal factor model for the analysis of student satisfaction towards university courses, J. Appl. Stat. 37 (2010) 473–487.
[17] T.-I. Lin, G.J. McLachlan, S.X. Lee, Extending mixtures of factor models using the restricted multivariate skew-normal distribution, J. Multiv. Anal. 143 (2016) 398–413.
[18] F. Agakov, D. Barber, An auxiliary variational method, in: Neural Information Processing, in: Lecture Notes in Computer Science, Vol. 3316, 2004, pp. 561–566.
[19] D.M. Blei, M.I. Jordan, J.W. Paisley, Variational Bayesian inference with stochastic search, in: Proceedings of ICML, 2012, pp. 1367–1374.
[20] K. Jarrett, K. Kavukcuoglu, M. Ranzato, Y. LeCun, What is the best multi-stage architecture for object recognition? in: Proceedings of ICCV, 2009.
[21] T. Miyato, S.-i. Maeda, M. Koyama, K. Nakae, S. Ishii, Distributional smoothing with virtual adversarial training, in: Proceedings of ICLR, 2016.
[22] D. Kingma, J. Ba, Adam: a method for stochastic optimization, in: Proceedings of ICLR, 2015.
[23] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceeding sof AISTATS, 2010.
[24] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I.J. Goodfellow, A. Bergeron, N. Bouchard, Y. Bengio, Theano: new features and speed improvements, in: Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.
[25] A. Voulodimos, D. Kosmopoulos, G. Vasileiou, E. Sardis, V. Anagnostopoulos, C. Lalos, A. Doulamis, T. Varvarigou, A threefold dataset for activity and workflow recognition in complex industrial environments, MultiMedia, IEEE 19 (2012) 42–52, doi:10.1109/MMUL.2012.31.
[26] D. Kosmopoulos, S. Chatzis, Robust visual behavior recognition, IEEE Signal Process. Mag. 27 (5) (2010) 34–45.
[27] S.M. Oh, J.M. Rehg, T. Balch, F. Dellaert, Learning and inferring motion patterns using parametric segmental switching linear dynamic systems, Int. J. Comput. Vis. 77 (1–3) (2008) 103–124.
[28] D. T. Bertin-Mahieux, P. Ellis, B. Whitman, P. Lamere, The million song dataset, in: Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011), 2011.

Mr. Harris Partaourides is a PhD student with the Cyprus University of Technology. His research interests are in the fields of Machine Learning and Artificial Intelligence. He holds a 5-year Diploma from the Department of Electrical and Computer Engineering of the University of Patras, Greece. His work mainly focuses on Deep Learning methods, Generative models, and variational Bayesian Inference.



Sotirios Chatzis received the M.Eng. (Hons.) degree in electrical and computer engineering and the Ph.D. degree in machine learning from the National Technical University of Athens, Athens, Greece, in 2005 and 2008, respectively. He was a Post-Doctoral Fellow with the University of Miami, Coral Gables, FL, USA, from 2008 to 2010. He was a Post-Doctoral Researcher with the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K., from 2010 to 2012. He is currently an Assistant Professor with the Department of Electrical Engineering, Computer Engineering and Informatics, Cyprus University of Technology, Limassol, Cyprus. He has authored more than 60 papers in the most prestigious journals and conferences of the research field. His current research interests include machine learning theory and methodologies, specifically hierarchical Bayesian models, Bayesian nonparametrics, and deep hierarchical feature extractors, with a focus on modeling data with temporal dynamics. His Ph.D. research was supported by the Bodossaki Foundation, Greece, and the Greek Ministry for Economic Development. Dr. Chatzis was a recipient of the Dean?s scholarship for Ph.D. studies, being the best performing Ph.D. student of the class.