On the Variational Posterior of Dirichlet Process Deep Latent Gaussian Mixture Models

Amine Echraibi [1 2] **Joachim Flocon-Cholet** [1] **Stéphane Gosselin** [1] **Sandrine Vaton** [2]

Abstract

Thanks to the reparameterization trick, deep latent Gaussian models have shown tremendous success recently in learning latent representations. The ability to couple them however with nonparametric priors such as the Dirichlet Process (DP) hasn't seen similar success due to its non parameterizable nature. In this paper, we present an alternative treatment of the variational posterior of the Dirichlet Process Deep Latent Gaussian Mixture Model (DP-DLGMM), where we show that the prior cluster parameters and the variational posteriors of the beta distributions and cluster hidden variables can be updated in closed-form. This leads to a standard reparameterization trick on the Gaussian latent variables knowing the cluster assignments. We demonstrate our approach on standard benchmark datasets, we show that our model is capable of generating realistic samples for each cluster obtained, and manifests competitive performance in a semi-supervised setting.

1. Introduction

Nonparametric Bayesian priors, such as the Dirichlet Process (DP), have been widely adopted in the probabilistic graphical community. Their ability to generate an infinite amount of probability distributions using a discrete latent variable makes them ideally suited for automatic model selection. The most famous applications of the DP have been however limited to classical probabilistic graphical models such as Dirichlet Process Mixture Models and Hierarchical Dirichlet Process Hidden Markov Models (Blei et al., 2006; Fox et al., 2008; Zhang et al., 2016).

[1]Orange Labs, Lannion, France [2]Institute Mines-Telecom Atlantique, Brest, France. Correspondence to: Amine Echraibi <amine.echraibi@orange.com>, Sandrine Vaton <sandrine.vaton@imt-atlantique.fr>, Joachim Flocon-Cholet <joachim.floconcholet@orange.com>, Stéphane Gosselin <stephane.gosselin@orange.com>.

Second workshop on *Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models* (ICML 2020), Virtual Conference

Recently, deep generative models such as Deep Latent Gaussian Models (DLGMs) and Variational AutoEncoders (VAEs) (Kingma & Welling, 2013; Rezende et al., 2014) have shown huge success in modeling and generating complex data structures such as images. Various proposals to generalize these models to the mixture and nonparametric mixture cases have been made (Nalisnick et al., 2016; Nalisnick & Smyth, 2016; Dilokthanakul et al., 2016; Jiang et al., 2016). Introducing such priors on top of the deep generative model can improve its generative capabilities, preserve class structure in the latent representation space, and offer a nonparametric way of performing model selection with respect to the size of the generative model.

The main challenge posed by such models lies in the inference process. Deep generative models with continuous latent variables owe their success mainly to the reparameterization trick (Kingma & Welling, 2013; Rezende et al., 2014). This approach provides an efficient and scalable method for obtaining low variance estimates of the gradient of the variational lower bound with respect to variational posterior parameters. Applying this approach directly to the variational posterior of the DP is not straightforward, due to the fact that a reparameterization trick for the beta distributions is hard to obtain (Ruiz et al., 2016). One approach to bypass this issue have been proposed by (Nalisnick & Smyth, 2016), where the authors used the Kumaraswamy distribution (Kumaraswamy, 1980) as a higher entropy alternative for the beta distribution in the variational posterior. However, by deriving the nature of the variational posterior directly from the variational lower bound, we can show that the appropriate distribution is in fact the beta distribution.

In this paper we provide an alternative treatment of the variational posterior of the DP-DLGMM, where we combine classical variational inference to derive the variational posteriors of the beta distributions and cluster hidden variables, and neural variational inference for the hidden variables of the latent Gaussian model. This leads to gradient ascent updates over the parameters present in nonlinear transformations where the reparameterization trick can be applied knowing the cluster assignment. As for the remaining parameters, closed-form solutions can be obtained by maximization of the evidence lower bound.

2. Dirichlet Process Deep Latent Gaussian Mixture Models

Generalizing deep latent Gaussian models to the Dirichlet process mixture case can be obtained by adding a Dirichlet process prior on the hidden cluster assignments. We denote these cluster assignments by $\mathbf{z}$. Following the assignment of a cluster hidden variable, a deep latent Gaussian model is defined for the assigned cluster similar to (Rezende et al., 2014). We adopt the stick-breaking construction of the Dirichlet Process (Sethuraman, 1994). The generative process of the model (figure 1) is given by:

$$
\begin{aligned}
\beta_k &\sim \text{Beta}(\cdot; 1, \eta) \\
\pi_k &= \beta_k \prod_{l=1}^{k-1} (1 - \beta_l) \\
\mathbf{z}_n|\pi &\sim \text{Cat}(\cdot|\pi) \\
\boldsymbol{\epsilon}_n^{(l)} &\sim \mathcal{N}(\cdot; \mathbf{0}, \mathbf{I}) \quad \forall l \\
\mathbf{h}_n^{(L)} &= m_{\mathbf{z}_n}^{(L)} + s_{\mathbf{z}_n}^{(L)} \odot \boldsymbol{\epsilon}_n^{(L)} \\
\mathbf{h}_n^{(l)} &= f_{W_{\mathbf{z}_n}^{(l)}} \left(\mathbf{h}_n^{(l+1)} \right) + s_{\mathbf{z}_n}^{(l)} \odot \boldsymbol{\epsilon}_n^{(l)} \\
\mathbf{x}_n|\mathbf{h}_n^{(1)}, \mathbf{z}_n &\sim p_X \left(\cdot \, | f_{W_{\mathbf{z}_n}^{(0)}} \left(\mathbf{h}_n^{(1)} \right) \right)
\end{aligned}
$$

where $\mathbf{h}_n^{(l)} \in \mathbb{R}^{p_l}$ is the l^{th} layer hidden representation constructed using a nonlinear transformation $f_{W_{\mathbf{z}_n}^{(l)}}$ represented by a neural network for the cluster assignment $\mathbf{z}_n$. For simplicity, we consider diagonal covariance matrices for each layer where the diagonal elements are $\left[(s_{\mathbf{z}_n,j}^{(l)})^2 \right]_{1 \leq j \leq p_l}$, hence $\odot$ represents the element-wise product. The generalization to full covariance matrices is straightforward using the Cholesky decomposition.

We denote by η the concentration parameter of the Dirichlet process which is a hyperparameter to be tuned manually. The term p_X represents the emission distribution of the observable $\mathbf{x}_n$, usually chosen to be a normal distribution for continuous variables or the Bernoulli distribution for binary variables. We denote the parameters of the generative model by:

$$
\Theta = \{ m_{1:\infty}^{(L)}, s_{1:\infty}^{(L)}, W_{1:\infty}^{(0:L-1)}, s_{1:\infty}^{(1:L-1)} \}
$$

The model thus has an infinite number of parameters due to the Dirichlet process prior. Furthermore, the posterior distribution of the hidden variables cannot be computed in closed-form. In order to perform inference on the model we need to use approximate methods such as Markov Chain Monte Carlo (MCMC) or Variational Inference. MCMC methods are not suitable for high dimensional models such as the DP-DLGMM, where convergence of the Markov chain to the true posterior can prove to be slow and hard to diagnose (Blei et al., 2017).

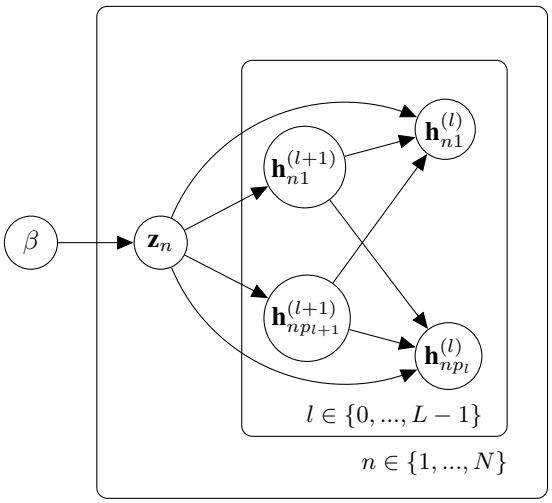

Figure 1. The graphical representation of the generative process of the model, with the convention $\mathbf{x} = \mathbf{h}^{(0)}$.

In the next section, we develop a structured variational inference algorithm for DP-DLGMM. We show that by choosing a suitable structure for the variational posterior, closed-form solutions can be obtained for the updates of the truncated variational posteriors of the beta distributions, the variational posteriors of the cluster hidden variables, and the optimal prior parameters $\{m^{(L)}, s^{(L)}\}$ maximizing the evidence lower bound.

3. Structured Variational Inference

For a brief review of variational methods, we denote by $\mathbf{x}_{1:N}$ the N samples present in the dataset supposed to be independent and identically distributed. The log-likelihood of the model is intractable due to the required marginalization of all the hidden variables. In order to bypass this marginalization, we introduce an approximate distribution q_Φ and use Jensen's inequality to obtain a lower bound (Jordan et al., 1999):

$$
\begin{aligned}
l(\Theta) &= \ln p_\Theta(\mathbf{x}_{1:N}) \\
&= \ln \left[\sum_{\mathbf{z}_{1:N}} \int p_\Theta(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta) d\mathbf{h}_{1:N}^{(1:L)} d\beta \right] \\
&\geq \mathbb{E}_{\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta \sim q_\Phi} \left[\ln \frac{p_\Theta(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta)}{q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta | \mathbf{x}_{1:N})} \right] \\
&\triangleq \mathcal{L}(\Theta, \Phi). \qquad (1)
\end{aligned}
$$

We can show that if the distribution q_Φ is a good approximation of the true posterior, maximizing the evidence lower bound (ELBO) with respect to the model parameters Θ is equivalent to maximizing the log-likelihood. For deep generative models, most state-of-the-art methods use inference networks to construct the posterior distribution (Rezende

et al., 2014; Nalisnick & Smyth, 2016). For deep mixture models with discrete latent variables, this approach leads to a mixture density variational posterior where the reparameterization trick requires additional investigation (Graves, 2016). Our approach combines standard variational Bayes and neural variational inference. We approximate the true posterior using the following structured variational posterior:

$$q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta|\mathbf{x}_{1:N}) = \prod_{n=1}^{N}\prod_{l=1}^{L} q_{\psi_{\mathbf{z}_n}^{(l)}}(\mathbf{h}_n^{(l)}|\mathbf{x}_n, \mathbf{z}_n) \times q_{\phi_n}(\mathbf{z}_n|\mathbf{x}_n)\prod_{t=1}^{T} q_{\gamma_t}(\beta_t|\mathbf{x}_{1:N}), \quad (2)$$

where T is a truncation level for the variational posterior of the beta distributions obtained by supposing that $q(\beta_T = 1) = 1$ (Blei et al., 2006). We assume a factorized posterior over the hidden layers $\mathbf{h}_n^{(1:L)}$, where the intra-layer dependencies are conserved.

3.1. Deriving the variational posteriors q_{ϕ_n} and q_{γ_t}

Deriving the nature of the posterior distributions of the hidden layers $\mathbf{h}_n^{(1:L)}$ using the variational approach is intractable due to the nonlinearities present in the model. Thus, we take a similar approach to (Rezende et al., 2014), and we assume that the variational posterior is specified by an inference network, where the parameters of the distribution are the outputs of deep neural networks $\mu_{\psi_t^{(l)}}$ and $\Sigma_{\psi_t^{(l)}}$ of parameters $\psi_t^{(l)}$ for the l^{th} layer and the t^{th} cluster:

$$q_{\psi_t^{(l)}}(\mathbf{h}_n^{(l)}|\mathbf{x}_n, \mathbf{z}_n = t) = \mathcal{N}\left(\mathbf{h}_n^{(l)}; \mu_{\psi_t^{(l)}}(\mathbf{x}_n), \Sigma_{\psi_t^{(l)}}(\mathbf{x}_n)\right).$$

In contrast to the hidden layers, we can use the proposed variational posterior of equation (2) to derive closed-form solutions for q_{ϕ_n} and q_{γ_t}. Let us consider the Kullback-Leibler definition of the ELBO $\mathcal{L}$:

$$\mathcal{L}(\Theta, \Phi) = -\mathbb{D}_{KL}\left[q_\Phi(\cdot|\mathbf{x}_{1:N})||p_\Theta(\cdot, \mathbf{x}_{1:N})\right].$$

By plugging the variational posterior and isolating β_t terms and $\mathbf{z}_n$ terms, we can analytically derive the optimal distributions q_{γ_t} and q_{ϕ_n} maximizing $\mathcal{L}$:

$$\begin{aligned} q_{\gamma_t}(\beta_t|\mathbf{x}_{1:N}) &= \text{Beta}(\beta_t; \gamma_{1,t}, \gamma_{2,t}) \\ q_{\phi_n}(\mathbf{z}_n|\mathbf{x}_n) &= \text{Cat}(\mathbf{z}_n; \phi_n), \end{aligned}$$

where the fixed point equations for the variational parameters ϕ_n and γ_t are:

$$\gamma_{1,t} = 1 + \sum_{n=1}^{N}\phi_{n,t} \quad (3)$$

$$\gamma_{2,t} = \eta + \sum_{n=1}^{N}\sum_{r=t+1}^{T}\phi_{n,r} \quad (4)$$

$$\begin{aligned} \ln \phi_{n,t} = \text{const} &+ \mathbb{E}_{\beta\sim q}[\ln \pi_t] \\ &+ \mathbb{E}_{\mathbf{h}_n^{(1:L)}\sim q_{\psi_t^{(1:L)}}}\left[\ln p_X(\mathbf{x}_n, \mathbf{h}_n^{(1:L)}|\mathbf{z}_n = t)\right] \\ &+ \sum_l \mathbb{H}\left[q_{\psi_t^{(l)}}(\cdot|\mathbf{z}_n = t, \mathbf{x}_n)\right] \\ \text{s.t.} \quad &\sum_{t=1}^{T}\phi_{n,t} = 1, \end{aligned} \quad (5)$$

The fixed point equation of $\phi_{n,t}$, requires the evaluation of the expectation over the hidden layers, this can be performed by sampling from the variational posterior of each hidden layer and then forwarding the sample using the generative model:

$$\begin{aligned} &\mathbb{E}_{\mathbf{h}_n^{(1:L)}\sim q_{\psi_t^{(1:L)}}}\left[\ln p_X(\mathbf{x}_n, \mathbf{h}_n^{(1:L)}|\mathbf{z}_n = t)\right] \\ &\approx \frac{1}{S}\sum_{s=1}^{S}\ln p_X\left(\mathbf{x}_n, \mathbf{h}_{n,t}^{(1:L)(s)}|\mathbf{z}_n = t\right) \\ &\text{where:} \quad \mathbf{h}_{n,t}^{(l)(s)} \sim q_{\psi_t^{(l)}}(\mathbf{h}_n^{(l)}|\mathbf{x}_n, \mathbf{z}_n = t). \end{aligned} \quad (6)$$

A key insight here is the following: if a cluster t is incapable of reconstructing a sample $\mathbf{x}_n$ from the variational posterior, this will reinforce the belief that $\mathbf{x}_n$ should not be assigned to that cluster. Furthermore, the estimation of the expectation can be performed using the same reparameterization trick that we will develop in section 3.3.

3.2. Closed-Form updates for $m_{1:T}^{(L)}$ and $s_{1:T}^{(L)}$

In addition to the variational posteriors of the beta distributions and the cluster assignments, closed-form solutions can be obtained for the updates of $m_{1:T}^{(L)}$ and $s_{1:T}^{(L)}$. Let us reconsider the evidence lower bound of equation (1), where we isolate only terms dependent on the prior parameters. We have:

$$\begin{aligned} &\mathcal{L}(m_{1:T}^{(L)}, s_{1:T}^{(L)}) = \text{const} - \sum_{n,t}\phi_{n,t} \\ &\times \mathbb{D}_{KL}\left[\mathcal{N}(\mu_{\psi_t^{(L)}}(\mathbf{x}_n), \Sigma_{\psi_t^{(L)}}(\mathbf{x}_n))||\mathcal{N}(m_t^{(L)}, V_t^{(L)})\right]. \end{aligned}$$

where $V_t^{(L)} = \text{diag}\left[(s_{t,j}^{(L)})^2\right]_{1\le j\le p_L}$ represents the covariance matrix of the L^{th} layer. By setting the derivative of $\mathcal{L}$ with respect to the parameters to zero, we obtain:

$$m_t^{(L)} = \frac{1}{N_t}\sum_{n=1}^{N}\phi_{n,t}\mu_{\psi_t^{(L)}}(\mathbf{x}_n) \quad N_t = \sum_{n=1}^{N}\phi_{n,t} \quad (7)$$

$$V_t^{(L)} = \frac{1}{N_t}\sum_{n=1}^{N}\phi_{n,t}\mathbf{I}\odot\Big\{\Sigma_{\psi_t^{(L)}}(\mathbf{x}_n) + (\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)})(\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)})^{\mathrm{T}}\Big\}, \quad (8)$$

where to extract the diagonal elements we perform an elementwise multiplication by the identity matrix $\mathbf{I}$. The update rules obtained are similar to the M-Step of a classical Gaussian Mixture Model, except in this case the updates are performed on the last hidden layer of the generative model, and the E-step of equation (5) takes into account all the hidden layers. Detailed derivation of the previous equations are presented in the supplementary material.

3.3. Stochastic Backpropagation

We next show how to perform stochastic backpropagation in order to maximize $\mathcal{L}$ with respect to the parameters ψ and $\Lambda = \{W_{1:T}^{(0:L-1)}, s_{1:T}^{(1:L)}\}$. Similarly to the previous section, we isolate the terms in the evidence lower bound dependent on ψ and Λ. We have:

$$\mathcal{L}(\psi,\Lambda) = \text{const} + \sum_{n,t}\phi_{n,t}\Big\{\sum_{l}\mathbb{H}\left[q_{\psi_t^{(l)}}(\cdot|\mathbf{z}_n = t, \mathbf{x}_n)\right] + \mathbb{E}_{\mathbf{h}_n^{(1:L)}\sim q_{\psi_t^{(1:L)}}}\left[\ln p_X(\mathbf{x}_n, \mathbf{h}_n^{(1:L)}|\mathbf{z}_n = t)\right]\Big\}. \quad (9)$$

By taking the expectation over the hidden cluster variables $\mathbf{z}_n$, we obtain conditional expectations over the hidden layers $\mathbf{h}_n^{(1:L)}$ knowing the cluster assignment. In order to backpropagate gradients of Λ and ψ, it suffices to perform a reparameterization trick for each cluster assignment at each hidden layer (proof in Appendix A). We can achieve this by sampling:

$$\epsilon_{n,t}^{(l)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

a sample from the posterior of the l^{th} hidden layer can then be obtained by the following transformation:

$$\mathbf{h}_{n,t}^{(l)} = \mu_{\psi_t^{(l)}}(\mathbf{x}_n) + \epsilon_{n,t}^{(l)}\sqrt{\Sigma_{\psi_t^{(l)}}(\mathbf{x}_n)},$$

where $\Sigma_{\psi_t^{(l)}}(\mathbf{x}_n)$ is supposed to be a diagonal matrix for simplicity. Following the previous analysis, we can derive an algorithm to perform inference on the proposed model, where between iterations of the fixed point update steps, E epochs of gradient ascent are performed to obtain a local maximum of the ELBO with respect to Λ and ψ. Algorithm 1 summarizes the process.

Algorithm 1 Variational Inference for the DP-DLGMM

Input: $\mathbf{x}_{1:N}, T, \eta, \alpha$
Initialize ϕ, Λ, ψ
while not converged **do**
  update: $\gamma_t \quad \forall t$ {(3), (4)}
  update: $m_t^{(L)} \quad \forall t$ {(7)}
  update: $V_t^{(L)} \quad \forall t$ {(8)}
  for each epoch **do**
    $\Lambda \leftarrow \Lambda + \alpha\partial_\Lambda\mathcal{L}$
    $\psi \leftarrow \psi + \alpha\partial_\psi\mathcal{L}$
  end for
  update: $\phi_{n,t} \quad \forall n, \forall t$ {(5)}
end while

4. Semi-Supervised Learning (SSL)

4.1. SSL using the DP-DGLMM

In this section, similarly to (Kingma et al., 2014) we consider a partially labeled dataset $\mathbf{x}_{1:N} = D_l \cup D_u$, where $D_l = \{\mathbf{x}_n, \mathbf{y}_n\}_n$ is the labeled part, $\mathbf{y}_n$ represents the label of the sample $\mathbf{x}_n$, and D_u represents the unlabeled part. The log likelihood can be divided for the labeled and unlabeled parts as:

$$\begin{aligned} l(\Theta) &= \ln p_\Theta(\mathbf{x}_{1:N}) \\ &= \sum_{\mathbf{x}_n\in D_l}\ln p_\Theta(\mathbf{x_n}) + \sum_{\mathbf{x}_n\in D_u}\ln p_\Theta(\mathbf{x}_n) \\ &= \sum_{\mathbf{x}_n\in D_l}\ln p_\Theta(\mathbf{x}_n, \mathbf{z}_n = \mathbf{y}_n) + \sum_{\mathbf{x}_n\in D_u}\ln p_\Theta(\mathbf{x}_n). \end{aligned}$$

The last equation follows from the fact that $p_\Theta(\mathbf{x}_n|\mathbf{z}_n \neq \mathbf{y}_n) = 0$. By dividing the labeled and unlabeled parts of the dataset, we can follow the same approach presented in section 3 in order to derive a variational inference algorithm. In this case, the fixed point updates and the gradient ascent steps remain unchanged if we set $\phi_{n,\mathbf{y}_n} = 1$ for a labeled $\mathbf{x}_n$ sample.

4.2. The predictive distribution

In order to make predictions using the model, we need to evaluate the predictive distribution. Given a new sample $\mathbf{x}_{N+1}$, the objective is to evaluate the following quantity $p(\mathbf{z}_{N+1} = k|\mathbf{x}_{1:N+1})$. This task requires an intractable marginalization over all the other hidden variables. However, similarly to (Blei et al., 2006), we can use the variational posterior to approximate the true posterior, which in turn leads to simpler expectation terms:

$$\begin{aligned} p(\mathbf{z}_{N+1} = k|\mathbf{x}_{1:N+1}) &\propto p(\mathbf{z}_{N+1} = k, \mathbf{x}_{N+1}|\mathbf{x}_{1:N}) \\ &\propto\!\!\propto \mathbb{E}_{\beta\sim q}\left[\pi_k(\beta)\right] \\ &\times \mathbb{E}_{\mathbf{h}_{N+1}^{(1:L)}\sim q_{\psi_k^{(1:L)}}}\left[p_X\left(\mathbf{x}_{N+1}|f_\Lambda(\mathbf{h}_{N+1}^{(1:L)}), \mathbf{z}_n = k\right)\right] \end{aligned} \quad (10)$$

where $f_\Lambda(\cdot)$ represents the forward pass over the generative model. The expectation with respect to the beta terms can

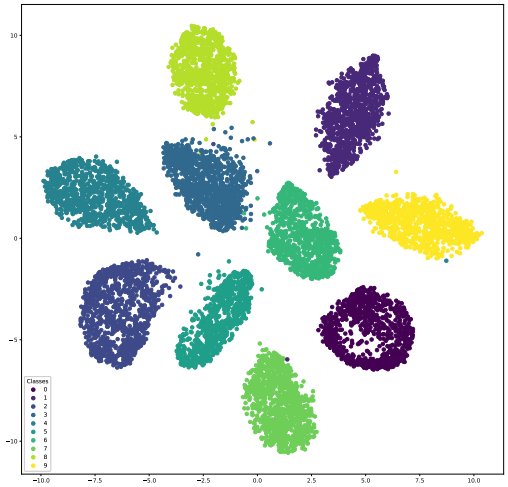

Figure 2. t-SNE plot of the second stochastic hidden layer on the MNIST test set for the semi-supervised (10% labels) version of the DP-DLGMM.

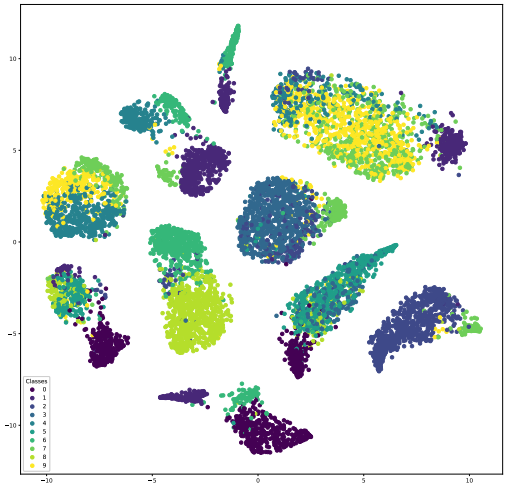

Figure 3. t-SNE plot of the second stochastic hidden layer on the MNIST test set for the unsupervised version of the DP-DLGMM.

be computed in closed-form as a product of expectations over the beta posteriors. The second expectation can be evaluated using the Monte-Carlo estimator of equation (6).

5. Experiments

5.1. Evaluation of the semi-supervised classification

We evaluate the semi-supervised classification capabilities of the model. We train our DP-DLGMM model on the MNIST dataset (LeCun & Cortes, 2010) with train-valid-test splits equal to $\{45000, 5000, 10000\}$ similarly to (Nalisnick & Smyth, 2016), with 10 % labelisation randomly drawn. We run the process for 5 iterations, and we evaluate our model on the test set. We report the mean and standard deviation of the classification error in percentages in Table 1. Our method produces a competitive score with existing state-of-the art methods: Deep Generative Models (DGM) (Kingma et al., 2014) and Stick-Breaking Deep Generative Models (SB-DGM) (Nalisnick & Smyth, 2016). Unlike the previous approaches, the loss was not up-weighted for the labeled samples. Figure 2 shows the t-SNE projections (Maaten & Hinton, 2008) obtained with 10 % of the labels provided. We notice that by introducing a small fraction of labels the class structure was highly preserved in the latent space.

kNN (k=5)	DGM	SB-DGM	DP-DLGMM
$6.13 \pm .13$	$4.86 \pm .14$	$3.95 \pm .15$	$\mathbf{2.90 \pm .17}$

Table 1. Semi-supervised classification error (%) on the MNIST test set with 10 % labelisation. Comparison with (Nalisnick & Smyth, 2016)

.

5.2. Data generation and visualization

To further test our model, we generate samples for each cluster from the models trained on both the MNIST and SVHN (Netzer et al., 2011) datasets. The MNIST model is trained in an unsupervised manner, and the SVHN model is trained with semi-supervision where we provide 1000 randomly generated labels. The samples obtained are represented in figure 4. For the unsupervised model, we notice that the clusters are representative of the shape of each digit. We plot the t-SNE projections of the MNIST test set of the unsupervised model in Figure 3. We notice that the digits belonging to the same true class tend to group with each other. However, two groups of the same class can be very separated in the embedding space. The interpretation we can draw from this effect is that the DP-DLGMM tends to separate the latent space in order to distinguish between the variations of hidden representations of the same class. The clusters obtained are not always representative of the true classes which is a common effect with infinite mixture models. In a full unsupervised setting, data can be explained

Figure 4. Generated samples from the DP-DLGMM model for the unsupervised version on the MNIST dataset (left) and the semi-supervised version on the SVHN dataset (right).

by multiple correct clusterings. This effect can simply be countered by adding a small supervision (figure 2).

6. Conclusion

In this paper, we have presented a variational inference method for Dirichlet Process Deep Latent Gaussian Mixture Models. Our approach combines classical variational inference and neural variational inference. The algorithm derived is thus a standard variational inference algorithm, with fixed point updates over a subset of the parameters presenting linear dependencies. The parameters present in nonlinear transformations are updated using standard gradient ascent where the reparameterization trick can be applied for the variational posterior of the stochastic hidden layers knowing the cluster assignments. Our approach shows promising results both for the unsupervised and semi-supervised cases. In future work, stochastic variational inference can be explored to speed-up the training procedure. Our approach can also be generalized to other types of deep probabilistic graphical models.

A. Proof of the reparameterization trick knowing the cluster assignment

The evidence lower bound of our model can be written in its general form as:

$$\begin{aligned}\mathcal{L}(\theta,\psi) &= \sum_{t=1}^{T}\phi_t \times \mathbb{E}_{h\sim\mathcal{N}(\mu_{\psi_t}(x),\sigma_{\psi_t}(x)^2\mathbf{I})}\left[f_\theta(h)\right]\\ &= \sum_{t=1}^{T}\phi_t \int_h \mathcal{N}(h,\mu_{\psi_t}(x),\sigma_{\psi_t}(x)^2\mathbf{I}) f_\theta(h)dh\\ &= \sum_{t=1}^{T}\phi_t \int_h \mathcal{N}(h,\mu_{\psi_t}(x),\sigma_{\psi_t}(x)^2\mathbf{I})\\ &\times\ f_\theta(h)|\partial_\epsilon h_t|d\epsilon.\end{aligned}$$

By introducing the following transformation:

$$h_t(\epsilon) = \mu_{\psi_t}(x) + \sigma_{\psi_t}(x)\odot\epsilon \quad \epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I}),$$

and using the density transformation lemma:

$$\mathcal{N}(h,\mu_{\psi_t}(x),\sigma_{\psi_t}(x)^2\mathbf{I})|\partial_\epsilon h_t| = \mathcal{N}(\epsilon;\mathbf{0},\mathbf{I}),$$

we have:

$$\begin{aligned}\mathcal{L}(\theta,\psi) &= \sum_{t=1}^{T}\phi_t\mathbb{E}_\epsilon\left[f_\theta(\mu_{\psi_t}(x)+\sigma_{\psi_t}(x)\odot\epsilon)\right]\\ &\approx \sum_{t=1}^{T}\phi_t f_\theta(\mu_{\psi_t}(x)+\sigma_{\psi_t}(x)\odot\hat{\epsilon}).\end{aligned}$$

where $\hat{\epsilon}$ is a sample drawn from $\mathcal{N}(\mathbf{0},\mathbf{I})$, thus we can backpropagate stochastic gradients for each class assignment.

B. Stochastic Variational Inference

Updating the Λ and ψ parameters using E epochs of gradient ascent significantly adds to the complexity of Algorithm 1. One possible approach is to perform stochastic variational inference (Hoffman et al., 2013) for fixed point update equations. This allows for the use of the same batch of data for the gradient ascent steps of Λ and ψ and the stochastic updates of the fixed point equations. Let us consider a batch $\mathbf{x}_{1:B}$, the updates in this case are:

$$\begin{aligned}\gamma_{1,k}^{(t+1)} &= (1-\rho_t)\gamma_{1,k}^{(t)} + \rho_t\frac{N}{B}\hat{\gamma}_{1,k}\\ \gamma_{2,k}^{(t+1)} &= (1-\rho_t)\gamma_{2,k}^{(t)} + \rho_t\frac{N}{B}\hat{\gamma}_{2,k}\\ m_k^{(t+1)} &= (1-\rho_t)m_k^{(t)} + \rho_t\frac{N}{B}\hat{m}_k\\ V_k^{(t+1)} &= (1-\rho_t)V_k^{(t)} + \rho_t\frac{N}{B}\hat{V}_k,\end{aligned}$$

where $\hat{\gamma}_{1,k}$, $\hat{\gamma}_{2,k}$, $\hat{m}_k$, $\hat{V}_k$, are computed for the minibatch $\mathbf{x}_{1:B}$ using equations (3),(4),(7), and (8) respectively. In order to guarantee convergence ρ_t must satisfy:

$$\sum_t \rho_t = \infty \quad \text{and} \quad \sum_t \rho_t^2 < \infty.$$

References

Blei, D. M., Jordan, M. I., et al. Variational inference for dirichlet process mixtures. *Bayesian analysis*, 1(1):121–143, 2006.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Dilokthanakul, N., Mediano, P. A., Garnelo, M., Lee, M. C., Salimbeni, H., Arulkumaran, K., and Shanahan, M. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.

Fox, E. B., Sudderth, E. B., Jordan, M. I., and Willsky, A. S. An hdp-hmm for systems with state persistence. In *Proceedings of the 25th international conference on Machine learning*, pp. 312–319, 2008.

Graves, A. Stochastic backpropagation through mixture density distributions. *arXiv preprint arXiv:1607.05690*, 2016.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Jiang, Z., Zheng, Y., Tan, H., Tang, B., and Zhou, H. Variational deep embedding: An unsupervised and generative approach to clustering. *arXiv preprint arXiv:1611.05148*, 2016.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pp. 3581–3589, 2014.

Kumaraswamy, P. A generalized probability density function for double-bounded random processes. *Journal of Hydrology*, 46(1-2):79–88, 1980.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov): 2579–2605, 2008.

Nalisnick, E. and Smyth, P. Stick-breaking variational autoencoders. *arXiv preprint arXiv:1605.06197*, 2016.

Nalisnick, E., Hertel, L., and Smyth, P. Approximate inference for deep latent gaussian mixtures. In *NIPS Workshop on Bayesian Deep Learning*, volume 2, 2016.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. 2011.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Ruiz, F. R., AUEB, M. T. R., and Blei, D. The generalized reparameterization gradient. In *Advances in neural information processing systems*, pp. 460–468, 2016.

Sethuraman, J. A constructive definition of dirichlet priors. *Statistica sinica*, pp. 639–650, 1994.

Zhang, A., Gultekin, S., and Paisley, J. Stochastic variational inference for the hdp-hmm. In *Artificial Intelligence and Statistics*, pp. 800–808, 2016.

Supplementary material

A. Detailed derivation of the variational inference algorithm:

The evidence lower bound of equation (1) can be written as :

$$\mathcal{L}(\Theta, \Phi) = -\mathbb{D}_{KL}\left[q_\Phi(\cdot|\mathbf{x}_{1:N})||p_\Theta(\cdot, \mathbf{x}_{1:N})\right],$$

in the following we develop this equation in order to derive the nature of the variational posteriors and their fixed point updates.

A.1. Computing $q_{\gamma_t}(\beta_t|\mathbf{x}_{1:N})$:

$$\begin{aligned}
\mathcal{L} &= \sum_{\mathbf{z}_{1:N}} \int q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta|\mathbf{x}_{1:N}) \ln \frac{p_\Theta(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta)}{q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta|\mathbf{x}_{1:N})} d\mathbf{h}_{1:N}^{(1:L)} d\beta \\
&= \sum_{\mathbf{z}_{1:N}} \int q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, |\mathbf{x}_{1:N}) q_\Phi(\beta|\mathbf{x}_{1:N}) \left[\ln p(\mathbf{x}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}|\mathbf{z}_{1:N}) + \ln p(\mathbf{z}_{1:N}, \beta)\right] d\mathbf{h}_{1:N}^{(1:L)} d\beta - \mathbb{H}[q_\Phi] \\
&= \underbrace{\text{const}}_{\text{independent of } \beta} + \prod_{t=1}^{T} \int_0^1 q(\beta_t) \left[\sum_n \sum_{\mathbf{z}_n} q(\mathbf{z}_n) \ln p(\mathbf{z}_n|\beta) + \sum_{t'} (\ln p(\beta_{t'}) - \ln q(\beta_{t'}))\right] d\beta_t \\
&= \underbrace{\text{const}}_{\text{independent of } \beta_t} + \int_0^1 q(\beta_t) \left[\sum_n \sum_{\mathbf{z}_n} q(\mathbf{z}_n) \ln p(\mathbf{z}_n|\beta) + \ln p(\beta_t) - \ln q(\beta_t)\right] d\beta_t \\
&= \text{const} + \int_0^1 q(\beta_t) \left[\sum_n \sum_k q(\mathbf{z}_n = k)(\ln \beta_k + \sum_{l<k} \ln(1-\beta_l)) + (\eta - 1)\ln(1-\beta_t) - \ln q(\beta_t)\right] d\beta_t \\
&= \text{const} + \int_0^1 q(\beta_t) \left[(\sum_n q(\mathbf{z}_n = t)) \ln \beta_t + (\eta + \sum_n \sum_{r=t+1}^{T} q(\mathbf{z}_n = r) - 1)\ln(1-\beta_t) - \ln q(\beta_t)\right] d\beta_t \\
&= \text{const} - \mathbb{D}_{KL}\left[q(\beta_t)||\text{Beta}(\beta_t; \gamma_{1,t}, \gamma_{2,t})\right]
\end{aligned}$$

The distribution $q^*(\beta_t)$ maximizing $\mathcal{L}$, minimizes the kullback-leibler term. Hence:

$$q_{\gamma_t}^*(\beta_t|\mathbf{x}_{1:N}) = \text{Beta}(\beta_t; \gamma_{1,t}, \gamma_{2,t})$$

$$\gamma_{1,t} = 1 + \sum_{n=1}^{N} q(\mathbf{z}_n = t) \quad \gamma_{2,t} = \eta + \sum_{n=1}^{N} \sum_{r=t+1}^{T} q(\mathbf{z}_n = r)$$

A.2. Computing $q_{\phi_n}(\mathbf{z}_n|\mathbf{x}_n)$:

By isolating the terms dependent on $\mathbf{z}_n$ in $\mathcal{L}$, we obtain:

$$\begin{aligned}
\mathcal{L} &= \sum_{\mathbf{z}_{1:N}} \int q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta|\mathbf{x}_{1:N}) \ln \frac{p_\Theta(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta)}{q_\Phi(\mathbf{z}_{1:N}, \mathbf{h}_{1:N}^{(1:L)}, \beta|\mathbf{x}_{1:N})} d\mathbf{h}_{1:N}^{(1:L)} d\beta \\
&= \text{const} + \sum_{\mathbf{z}_n} q(\mathbf{z}_n) \left\{ \int q(\mathbf{h}_n^{(1:L)}|\mathbf{x}_n, \mathbf{z}_n) \ln \left[\frac{p(\mathbf{x}_n|\mathbf{h}_n^{(1:L)}) p(\mathbf{h}_n^{(1:L)}|\mathbf{z}_n)}{q(\mathbf{h}_n^{(1:L)}|\mathbf{x}_n, \mathbf{z}_n)}\right] d\mathbf{h}_n^{(1:L)} + \mathbb{E}_{\beta\sim q}[\ln p(\mathbf{z}_n|\beta)] - \ln q(\mathbf{z}_n) \right\} \\
&= \text{const} - \mathbb{D}_{KL}\left[q(\mathbf{z}_n)||\text{Cat}(\mathbf{z}_n; \phi_n)\right]
\end{aligned}$$

Hence, the optimal distribution $q^*_{\phi_n}(\mathbf{z}_n|\mathbf{x}_n)$ maximizing $\mathcal{L}$ satisfies:

$$q^*_{\phi_n}(\mathbf{z}_n|\mathbf{x}_n) = \text{Cat}(\mathbf{z}_n; \phi_n)$$

where,

$$\begin{aligned}
\ln \phi_{n,k} &= \text{const} + \int q(\mathbf{h}_n^{(1:L)}|\mathbf{x}_n, \mathbf{z}_n = k) \ln \left[\frac{p(\mathbf{x}_n, \mathbf{h}_n^{(1:L)}|\mathbf{z}_n = k)}{q(\mathbf{h}_n^{(1:L)}|\mathbf{x}_n, \mathbf{z}_n = k)} \right] d\mathbf{h}_n^{(1:L)} + \mathbb{E}_{\beta\sim q}[\ln \pi_k] \\
&= \text{const} + \mathbb{E}_{\mathbf{h}_n^{(1:L)} \sim q_{\psi_k^{(1:L)}}} \left[\ln p(\mathbf{x}_n, \mathbf{h}_n^{(1:L)}|\mathbf{z}_n = k) \right] + \mathbb{E}_{\beta\sim q}[\ln \pi_k] + \sum_l \mathbb{H}\left[q_{\psi_k^{(l)}}(\cdot|\mathbf{z}_n = k, \mathbf{x}_n) \right]
\end{aligned}$$

A.3. Closed-form solutions for $m_{1:T}^{(L)}$ and $s_{1:T}^{(L)}$:

Let us consider in this case only the terms dependent on $m_{1:T}^{(L)}$ and $s_{1:T}^{(L)}$ present in the evidence lower bound, we have:

$$\begin{aligned}
\mathcal{L} &= \text{const} + \sum_n \sum_t q_{\phi_n}(\mathbf{z}_n = t) \left\{ \underbrace{\mathbb{E}_{\mathbf{h}_n^{(1:L)} \sim q_{\psi_t^{(1:L)}}} \left[\ln p(\mathbf{x}_n|\mathbf{h}_n^{(1:L)}, \mathbf{z}_n = t) \right]}_{\text{independent of } m^{(L)} \text{ and } s^{(L)}} + \mathbb{E}_{\mathbf{h}_n^{(L)} \sim q_{\psi_t^{(L)}}} \left[\frac{p(\mathbf{h}_n^{(L)}|\mathbf{z}_n = t)}{q(\mathbf{h}_n^{(L)}|\mathbf{x}_n, \mathbf{z}_n = t)} \right] \right\} \\
&= \text{const} - \sum_n \sum_t \phi_{n,t} \mathbb{D}_{KL} \left[\mathcal{N}(\mu_{\psi_t^{(L)}}(\mathbf{x}_n), \Sigma_{\psi_t^{(L)}}(\mathbf{x}_n)) || \mathcal{N}(m_t^{(L)}, V_t^{(L)}) \right]
\end{aligned}$$

where, $V_t^{(L)} = \text{diag}\left[(s_{t,j}^{(L)})^2 \right]_{1\leq j\leq p_L}$. By setting the derivative of $\mathcal{L}$ to zero:

$$\begin{aligned}
\partial_{V_t^{(L)-1}} \mathcal{L} &= -\sum_n \phi_{n,t} \left\{ \Sigma_{\psi_t^{(L)}}(\mathbf{x}_n) + (\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)})(\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)})^{\text{T}} \right\} + N_t V_t^{(L)} \\
&= 0 \\
\partial_{m_t^{(L)}} \mathcal{L} &= -\sum_n \phi_{n,t} \left(\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)} \right) V_t^{(L)-1} \\
&= 0
\end{aligned}$$

Hence, the closed-form update equations:

$$m_t^{(L)} = \frac{1}{N_t} \sum_{n=1}^{N} \phi_{n,t} \mu_{\psi_t^{(L)}}(\mathbf{x}_n) \quad N_t = \sum_{n=1}^{N} \phi_{n,t}$$

$$V_t^{(L)} = \frac{1}{N_t} \sum_{n=1}^{N} \phi_{n,t} \mathbf{I} \odot \left\{ \Sigma_{\psi_t^{(L)}}(\mathbf{x}_n) + (\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)})(\mu_{\psi_t^{(L)}}(\mathbf{x}_n) - m_t^{(L)})^{\text{T}} \right\}$$

Given that $V_t^{(L)}$ is a diagonal matrix we extract the diagonal updates by multiplying elementwise by the identity matrix.

B. The predictive distribution:

In order to perform inference for a new sample $\mathbf{x}_{N+1}$, we need to compute the predictive distribution $p(\mathbf{z}_N + 1|\mathbf{x}_{1:N+1})$:

$$\begin{aligned}
p(\mathbf{z}_N + 1|\mathbf{x}_{1:N+1}) &\propto p(\mathbf{z}_{N+1}, \mathbf{x}_{N+1}|\mathbf{x}_{1:N}) \\
&\propto \int p(\mathbf{x}_{N+1}, \mathbf{h}, \mathbf{z}_{N+1}, \beta|\mathbf{x}_{1:N}) d\mathbf{h} d\beta \\
&\propto \int p(x_{N+1}|f_\Lambda(\mathbf{h}), \mathbf{z}_{N+1} = k) p(\mathbf{h}|\mathbf{x}_{1:N}, \mathbf{z}_{N+1} = k) p(\mathbf{z}_{N+1} = k|\beta) p(\beta|\mathbf{x}_{1:N}) d\mathbf{h} d\beta
\end{aligned}$$

By approximating the true posteriors with the variational posteriors:

$$p(\beta|\mathbf{x}_{1:N}) \approx q(\beta) \quad \text{and} \quad p(\mathbf{h}|\mathbf{x}_{1:N}, \mathbf{z}_{N+1} = k) \approx q(\mathbf{h}|\mathbf{z}_{N+1} = k)$$

We conclude:

$$p(\mathbf{z}_{N+1} = k|\mathbf{x}_{1:N+1}) \propto \mathbb{E}_{\beta\sim q}\left[\pi_k(\beta)\right] \times \mathbb{E}_{\mathbf{h}_{N+1}^{(1:L)}\sim q_{\psi_k^{(1:L)}}}\left[p_X\left(\mathbf{x}_{N+1}|f_\Lambda(\mathbf{h}_{N+1}^{(1:L)}), \mathbf{z}_n = k\right)\right]$$