entropy

MDPI

Article

Data-Dependent Conditional Priors for Unsupervised Learning of Multimodal Data †

Frantzeska Lavda [1,2,*], Magda Gregorová [2] and Alexandros Kalousis [2]

[1] Faculty of Science, Computer Science Department, University of Geneva, 1214 Geneva, Switzerland
[2] Geneva School of Business Administration (DMML Group), HES-SO, 1227 Geneva, Switzerland; magda.gregorova@hesge.ch (M.G.); alexandros.kalousis@hesge.ch (A.K.)
* Correspondence: frantzeska.lavda@etu.unige.ch
† This paper is an extended version of our paper published in the European Conference on Artificial Intelligence, ECAI2020.

Received: 26 July 2020; Accepted: 11 August 2020; Published: 13 August 2020

Abstract: One of the major shortcomings of variational autoencoders is the inability to produce generations from the individual modalities of data originating from mixture distributions. This is primarily due to the use of a simple isotropic Gaussian as the prior for the latent code in the ancestral sampling procedure for data generations. In this paper, we propose a novel formulation of variational autoencoders, conditional prior VAE (CP-VAE), with a two-level generative process for the observed data where continuous **z** and a discrete **c** variables are introduced in addition to the observed variables **x**. By learning data-dependent conditional priors, the new variational objective naturally encourages a better match between the posterior and prior conditionals, and the learning of the latent categories encoding the major source of variation of the original data in an unsupervised manner. Through sampling continuous latent code from the data-dependent conditional priors, we are able to generate new samples from the individual mixture components corresponding, to the multimodal structure over the original data. Moreover, we unify and analyse our objective under different independence assumptions for the joint distribution of the continuous and discrete latent variables. We provide an empirical evaluation on one synthetic dataset and three image datasets, FashionMNIST, MNIST, and Omniglot, illustrating the generative performance of our new model comparing to multiple baselines.

Keywords: VAE; generative models; learned prior

1. Introduction

Variational autoencoders (VAEs) [1,2] are deep generative models for learning complex data distributions. They consist of an encoding and decoding network parametrizing the variational approximate posterior and the conditional data distributions in a latent variable generative model.

Though powerful and theoretically elegant, the VAEs in their basic form suffer from multiple deficiencies that stem from the mathematically convenient yet simplistic distributional assumptions. Multiple strategies have been proposed to increase the richness or interpretability of the latent code [3–12]. These mostly argue for more flexible posterior inference procedure or for the use of more complex approximate posterior distributions to facilitate the encoding of non-trivial data structures within the latent space.

In this paper, we reason that for generating realistic samples of data originating from complex distributions, it is the prior that lacks expressiveness. Accordingly, we propose a new VAE formulation, conditional prior VAE (CP-VAE), with two-level hierarchical generative model combining categorical and continuous (Gaussian) latent variables.

The hierarchical conditioning of the continuous latent variable on the discrete latent component is particularly suitable for modelling multimodal data distributions, such as distributional mixtures. Importantly, it also gives us better control of the procedure for generating new samples. Unlike in the standard VAE, we can sample data from specific mixture components at will. This is particularly critical if the generative power of VAEs shall be used in conjunction with methods requiring the identification of the distributional components, such as in continual learning [13,14].

As recently shown [12,15], without supervision (as in our setting), enforcing independence factorization in the latent space does not guarantee recovering meaningful sources of variation in the original space. Therefore, in our CP-VAE formulation, we let the model fully utilize the capacity of the latent space by allowing for natural conditional decomposition in the generative and inference graphical models.

We formulate the corresponding variational lower bound on the data log-likelihood and use it as the optimization objective in the training. In the spirit of empirical Bayes, we propose estimating the parameters of the conditional priors from the data together with the parameters of the variational posteriors in a joint learning procedure. This ensures that the inferred structure of the latent space can be exploited in data generations.

2. From Variational Inference (VI) Objective to VAE Objective

Variational autoencoders (VAEs) [1,2] are deep Bayesian generative models that rely on the principals of amortized variational inference to approximate the complex distributions $p(\mathbf{x})$ from which the observed data $\mathcal{S} = \{\mathbf{x}_i\}_{i=1}^N$ originate.

In their basic form, they model the unknown ground-truth $p(\mathbf{x})$ by a parametric distribution $p_\theta(\mathbf{x})$ with a latent variable generative process

$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})dz \ . \tag{1}$$

Computing $p_\theta(\mathbf{x})$ is difficult and usually turns out to be an intractable distribution. However, we can learn a surrogate loss to the original likelihood $p_\theta(\mathbf{x})$ while using Variational Inference principles.

2.1. Variational Inference

Variational Inference involves the optimization of an approximation to the intractable posterior. In Variational Inference, we specify a family of tractable distributions $q_\phi(\mathbf{z}|\mathbf{x})$. The goal is to find the best variational parameters ϕ, such that the approximation $q_\phi(\mathbf{z}|\mathbf{x})$ is as close as possible to the intractable posterior $p_\theta(\mathbf{z}|\mathbf{x})$, i.e., $q_\phi(\mathbf{z}|\mathbf{x}) \sim p_\theta(\mathbf{z}|\mathbf{x})$. We do that by minimizing the KL divergence of the approximation $q_\phi(\mathbf{z}|\mathbf{x})$ from the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$,

$$\phi^* = argmin_\phi \, \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) \tag{2}$$

where the KL divergence is equal to:

$$\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})}\right] \tag{3}$$

Reordering the terms of Equation (3), we have:

$$\begin{aligned} \log p_\theta(\mathbf{x}) &= \underbrace{\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right)}_{KL} + \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{z},\mathbf{x}) - \log q_\phi(\mathbf{z}|\mathbf{x})\right]}_{ELBO} \\ &= \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) + \mathcal{L}(\mathbf{x};\theta,\phi) \end{aligned} \tag{4}$$

The first term is the initial KL divergence we want to minimize in VI. Since the Kullback-Leibler divergence is always greater than or equal to zero, the second term $\mathcal{L}$, called the Evidence Lower BOund, ELBO, is the variational lower bound on the marginal log likelihood $\log p_\theta(\mathbf{x})$. The closer KL $\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})\right)$ is to 0, the closer $\mathcal{L}(\mathbf{x};\theta,\phi)$ will be to $\log p(\mathbf{x})$.

$$\log p_\theta(\mathbf{x}) \geq \mathcal{L}(\mathbf{x};\theta,\phi) \tag{5}$$

2.2. Variational Autoencoders

In VAEs, the typical assumptions are of a simple isotropic Gaussian prior $p(\mathbf{z})$ for the latent variable $\mathbf{z}$ and, depending on the nature of the data $\mathbf{x}$, factorized Bernoulli or Gaussian distributions for the data conditionals $p_\theta(\mathbf{x}|\mathbf{z})$. These per-sample conditionals are parametrized by a deep neural network, a decoder . Once the decoder network is properly trained, we can sample new data examples from the learned data distribution $p_\theta(\mathbf{x})$ by ancestral sampling procedure: sample the latent $\mathbf{z}$ from the prior $p(\mathbf{z})$ and pass it through the stochastic decoder $p_\theta(\mathbf{x}|\mathbf{z})$ to obtain the sample $\mathbf{x}$.

The VAEs employ the strategy of amortized variational inference. They approximate the intractable posteriors $p_\theta(\mathbf{z}|\mathbf{x})$ by factorized Gaussian distributions $q_\phi(\mathbf{z}|\mathbf{x})$ and infer the variational parameters ϕ of the approximate per-sample posteriors through a deep neural network, an encoder.

The encoder and decoder networks are trained end-to-end by stochastic gradient-based optimization maximizing the sample estimate of The Evidence Lower Bound $\mathcal{L}_{\theta,\phi} = \mathbb{E}_{p(\mathbf{x})}\mathcal{L}_{\theta,\phi}(\mathbf{x})$ on the data log-likelihood.

$$\frac{1}{N}\sum_i^N \log p_\theta(\mathbf{x}_i) \geq \frac{1}{N}\sum_i^N \mathcal{L}_{\theta,\phi}(\mathbf{x}_i) \approx \mathcal{L}_{\theta,\phi}$$

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z})}_{A} - \underbrace{\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right)}_{B} \tag{6}$$

The first term A in Equation (6) can be seen as a negative reconstruction cost, term B penalizes the deviations of the approximate posterior from the fixed prior $p(\mathbf{z})$ and it has a regularizing effect on the model learning. The term A encourages the latent variable $\mathbf{z}$ to contain meaningful information in order to reconstruct x and at the same time, the term B penalizes the approximate posterior for deviating from the prior, preventing the model from simply memorizing each data point.

The gradients of the lower bound with respect to the model parameters θ can be obtained streighforwardly through Monte Carlo estimation. For the posterior parameters ϕ, the gradients are estimated by stochastic backpropagation while using a location-scale transformation known as the reparametrization trick.

2.3. Posterior Collapse and Mismatch between the True and the Approximate Posterior

In Equation (4), we see that, in order to improve the variational lower bound, the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ should match the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$. In other words, the ELBO is tight when $q_\phi(\mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{z}|\mathbf{x})$. As we mentioned above, the choice of $q_\phi(\mathbf{z}|\mathbf{x})$ is often a factorized Gaussian distribution for simplicity and efficiency. In this way, the approximate posterior is simplified and it is hard for it to match the possible complex true posterior.

Moreover, by minimizing the KL-term in Equation (6), we encourage the approximate posterior to be close to the simple isotropic Gaussian prior $p(\mathbf{z})$, an even simpler distribution. This may cause the main issue with VAE, called posterior collapse, where the model learns to ignore the latent variable and the approximate posterior mimics the prior, $q_\phi(\mathbf{z}|\mathbf{x}) \approx p(\mathbf{z})$ [6,16]. This reduces the capacity of the generative model, making it impossible for the decoder to use all the information of all of the latent dimensions or even not use, at all, the latent variable. This problem is more common when the decoder $p_\theta(\mathbf{x}|\mathbf{z})$ is parametrised as an autoregressive model [6].

The posterior collapse and, consequently, the mismatch between the true and the approximate posterior motivates a direct improvement of variational inference by assuming/learning a more flexible posterior approximation for variational inference [3,5], or an indirect improvement assuming a more flexible prior [6,17]. Moreover a range of heuristic approaches in the literature have attempted to diminish the effect of the KL term in the ELBO to alleviate posterior collapse [18], or propose new regularizers [9].

2.4. Optimal Prior

Even though the prior in the VAEs is usually modelled by a simple isotropic Gaussian distribution, this assumption is a source of over-regularization, and is one of the causes of the poor density estimation performance [19].

To derive the optimal prior, we reformulate the VAE objective Equation (4). By maximizing the ELBO, we force the approximate posterior to be close to the true one and the marginal likelihood $p_\theta(\mathbf{x})$ to be close to the data distribution, $p_D(\mathbf{x})$, as we see in Equation (7).

$$\begin{aligned}\mathcal{L}_{\theta,\phi} &= \mathbb{E}_{p_D(\mathbf{x})}\left[\log p_\theta(\mathbf{x}) - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))\right] \\ &= \mathbb{E}_{p_D(\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x})}{p_D(\mathbf{x})} p_D(\mathbf{x})\right] - \mathbb{E}_{p_D(\mathbf{x})}\left[\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))\right] \\ &= -\mathrm{KL}(p_D(\mathbf{x})\|p_\theta(\mathbf{x})) - \mathbb{H}(p_D(\mathbf{x})) - \mathbb{E}_{p_D(\mathbf{x})}\left[\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))\right] \end{aligned} \tag{7}$$

The maximizing solution is equal to the negative entropy of the data distribution, $-\mathbb{H}(p_D(\mathbf{x}))$ and it is reached when the two KL divergence terms are equal to zero, meaning that the approximate posterior becomes equal to the true one, $q_\phi(\mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{z}|\mathbf{x})$ and the data distribution equal to the true distribution, $p_D(\mathbf{x}) = p_\theta(\mathbf{x})$.

In this optimal case, the marginal approximate posterior $q_\phi(\mathbf{z})$ matches the prior, $q_\phi(\mathbf{z}) = \int_x q_\phi(\mathbf{z}|\mathbf{x})p_D(\mathbf{x})d\mathbf{x} = \int_x p_\theta(\mathbf{z}|\mathbf{x})p_\theta(\mathbf{x})d\mathbf{x} = p(\mathbf{z})$. This indicates that the optimal prior for maximizing the ELBO is the marginal approximate posterior.

$$p(\mathbf{z}) \leftarrow q_\phi(\mathbf{z}) = \frac{1}{N}\sum_{i=1}^{N} q_\phi(\mathbf{z}|\mathbf{x}_i)$$

where the summation is performed over all training samples x_i; $i = 1, \cdots, N$. The marginal posterior is the average of the approximate posterior with as many components as data points in the sample S, and it can been seen as Mixture of Gaussians (MoG) over all the data. However, this extreme case leads to over-fitting as this prior essentially memorizes the training set. Moreover, it is computationally inefficient, since it is very expensive to compute at every training iteration.

A natural approximation of the marginal approximate posterior prior can be a Mixture of Gaussian (MoG) prior in a random subset of the data, $p(\mathbf{z}) = \frac{1}{K}\sum_{k=1}^{K} p_\phi(\mathbf{z}|\mathbf{x}_k)$ with $K < N$ components.

Alternatively, marginal approximate posterior can be modelled while using a mixture of posteriors over learned virtual observations (pseudo-inputs) with a fixed number of components $p(\mathbf{z}) \simeq \frac{1}{K}\sum_k q_\phi(\mathbf{z}|u^{(k)})$ [17]. Hence, the original standard Gaussian prior is replaced by a flexible multi-modal distribution.

3. Related Work

Since their introduction in 2014 [1,2], variational autoencoders have become one of the major workhorses for large-scale density estimation and unsupervised representation learning. Multitudes of variations on and enhancements of the original design have been proposed in the literature. These can broadly be categorized into four large groups (with significant overlaps as many methods mix multiple ideas to achieve the best possible performance).

First, it has been argued that optimizing the variational bound Equation (6) instead of the intractable likelihood $p_\theta(\mathbf{x})$ inhibits the VAEs to learn useful latent representations for both data reconstructions and downstream tasks. Methods using alternative objectives aim to encourage the learning towards representations that are better aligned with the data (measured by mutual information), e.g., InfoVAE [9,10], or which separate important factors of variations in the data (disentangling), e.g., [18,20,21]. Although these methods report good results on occasions, there seem to be little evidence that breaking the variational bound brings systematical improvements [12,15].

For our model, the analysis presented in Section 4.3.1 suggests that our objective (which is a proper lower bound on the likelihood) encourages the encoding of the major source of variation, that of the originating mixture component, through the categorical variable without any extra alterations. At the same time, it should be noted that our goal is not the interpretability of the learned representations or their reuse outside the VAEs models. Our focus is on generations reflecting the underlying multi-modal distribution over the original data space.

Second, the simplifying conditional independence assumptions for the data dimensions factored into the simple Gaussian decoder $p_\theta(\mathbf{x}|\mathbf{z})$ have been challenged in the context of modelling data with strong internal dependencies. More powerful decoders with autoregressive architectures have been proposed for modelling images, e.g., PixelVAE [22], or sequentially dependent data such as speech and sound, e.g., VRNN [23]. In our model, we use a hierarchical decoder $p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{c})$ corresponding to the cluster-like structure we assume for the data space. However, in this work, we stick to the simple independence assumption for the data dimensions. Augmenting our method with stronger decoder should, in principle, be possible and it is open for future investigation.

Third, the insufficient flexibility of the variational posterior $q_\phi(\mathbf{z}|\mathbf{x})$ to approximate the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$ has led to proposals for more expressive posterior classes. For example, a rather successful approach is based on chaining invertible transformations of the latent variable [3,5]. While increasing the flexibility of the approximate posterior improves the modelling objective through better reconstructions, without accompanied enhancements of the prior it does not guarantee better generations.

This has been recognised and addressed by the fourth group of improvements that focuses on the model prior and that our method pursues. These build on the observation that overly-simple priors can be source of excessive regularization, limiting the success of the VAE models [6,19]. For example, the authors in [11,24] replace the distributional class of the prior (together with the posterior) by von Mises–Fisher distributions with potentially better characteristics for high-dimensional data with hyperspherical latent space.

More related to ours are methods that suggest to learn the prior. The VLVAE [6] uses the autoregressive flows in the prior that are equivalent to the inverse autoregressive flows in the posterior [5]. The increased richness of the encoding and prior distributions leads to higher quality generations; however, the prior cannot be used to generate from selected parts of the data space, as our model can.

The VampPrior [17] proposes constructing the prior as a mixture of the variational posteriors over a learned set of pseudo-inputs. These could be interpreted as learned cluster prototypes of the data. However, the model does not learn the importance of the components in the mixture, and it does not align the prior and posteriors at an individual component level as our model does. Instead, it pushes the posteriors to align with the overall prior mixture that diminishes the models ability to correctly generate from the individual components of multimodal data. In [25], they use the aggregated posterior as the prior by directly estimating the KL divergence without modeling the aggregated posterior explicitly, while using a kernel density trick . However, because their prior is implicit, they cannot sample from the prior directly. Instead, they sample from the aggregated posterior. Moreover, the model simillarly to VampPrior does not learn the importance of the components in the mixture.

The continuous-discrete decomposition of the latent space similar to ours have been used for data clustering through generative model presented in [7,26]. The first combines the VAE with a Gaussian mixture model through two stage procedure mimicking the independence assumptions in their inference model. The latter assumes (conditional) independence in the generative and inference models and extends to a full Bayesian formulation through the use of hyper-priors. Their complex model formulation exhibits some over-regularization issues that, to the authors acknowledge, are challenging to control.

Options for freeing the distributional class of the latent representations through Bayesian non-parametrics have been explored, for example, in [8,27,28]. The learned structures in the latent representations greatly increase the generative capabilities, including also the (hierarchical) clustering ability. However, this comes at a cost of complex models that are tricky to train in a stable manner. In contrast, our model is elegantly simple and easy to train.

4. VAE with Data-Dependent Conditional Priors

The mathematically and practically convenient assumption of the factorial Gaussian approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ has been previously contested as one of the major limitations of the basic VAE architecture. For complex data distributions $p(\mathbf{x})$, the simple Gaussian $q_\phi(\mathbf{z}|\mathbf{x})$ may not be flexible enough to approximate well the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$.

Even though various methods have been proposed for enriching the posterior distributions, as we mention in Section 3, by learning latent representations more appropriate for the complex data structures they cannot guarantee better generations. In order to achieve this a closer match between the posterior and prior distributions used for sampling the latent variables during inference and data generations, respectively, is required.

We propose a new VAE formulation, conditional prior VAE (CP-VAE), with a conditionally structured latent representation that encourages a better match between the prior and the posterior distributions by jointly learning their parameters from the data.

4.1. Two-Level Generative Process

We consider a two-level hierarchical generative process for the observed data where two latent variables $\mathbf{c}$ and $\mathbf{z}$ are introduced in addition to the observed variables $\mathbf{x}$. Variable $\mathbf{c}$ is a $K-$way categorical latent variable, and $\mathbf{z}$ is a $D-$dimensional continuous latent variable. To generate $\mathbf{x}$, we first sample $\mathbf{c}$ sample from its prior, $p(\mathbf{c})$, and then a continuous latent variable $\mathbf{z}$ is sampled from the learned conditional distribution $p_\varphi(\mathbf{z}|\mathbf{c})$. Finally, a sample is drawn from $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c})$, parameterized by the decoder network. The joint probability can be written as:

$$p(\mathbf{x},\mathbf{z},\mathbf{c}) = p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c})p_\varphi(\mathbf{z},\mathbf{c}) \tag{8}$$

where, the joint prior distribution is equal to $p_\varphi(\mathbf{z},\mathbf{c}) = p_\varphi(\mathbf{z}|\mathbf{c})p(\mathbf{c})$.

We assume a uniform categorical as a prior distribution for the discrete component $\mathbf{c}$, so that, for each of the K categories $p(c_k) = 1/K$, $k = 1,\ldots,K$, which encourages every component to be used. The conditionals of the continuous component are factorised Gaussians with learnable means and variances.

$$p_\varphi(\mathbf{z}|c_k) = \prod_i p_\varphi(z_i|c_k) = \prod_i \mathcal{N}\left(z_i \,|\, \mu_{ik}, \sigma^2_{ik}\right),\ k = 1,\ldots,K\ . \tag{9}$$

The compositional prior we propose is well suited for generations of new samples from multimodal data distributions mixing multiple distributional components. In contrast to sampling from a simple isotropic Gaussian prior that concentrates symmetrically around the origin, we can sample the latent code from discontinuous parts of the latent space. These are expected to represent data clusters corresponding to the originating distributional mixing.

In addition, the variations encoded into the continuous part of the latent space are also sampled conditionally and therefore are better adapted to represent the important factors of data variations within the distributional clusters. This is in contrast to the single common continuous distribution of the basic VAE (Section 2.2) or VAEs with similar continuous-discrete composition of the latents as ours, which, however, assume independence between the two parts of the latent representation [21], which we discuss in detail in Section 5.1.

The data conditional $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c})$ is parametrised by a decoder network $d_\theta(\mathbf{z},\mathbf{c})$ as a Bernoulli$(\mathbf{x}\,|\,d_\theta(\mathbf{z},\mathbf{c}))$ or a Gaussian $\mathcal{N}\left(\mathbf{x}\,|\,d_\theta(\mathbf{z},\mathbf{c}),\sigma^2 I\right)$ distribution, depending on the nature of the data $\mathbf{x}$.

Data-Dependent Conditional Priors

There is no straightforward way to fix the parameters $\varphi = (\mu, \sigma)$ in the distributions Equation (9) for each of the conditioning categories c_k a priori. Instead of placing hyper-priors on the parameters and expanding to full hierarchical Bayesian modelling, we estimate the prior parameters from the data through a relatively simple procedure that resembles the empirical Bayes technique [29].

As explained in Section 4.3, the conditional $p_\varphi(\mathbf{z}|\mathbf{c})$ enters our objective function through a KL divergence term. Therefore, the prior parameters φ can be optimized by backpropagation together with learning the encoder and decoder parameters ϕ and θ. Once the model is trained, all of the parameters are fixed and the learned prior $p_\varphi(\mathbf{z}|\mathbf{c})$ can be used in the ancestral sampling procedure that is described above to generate new data samples, Figure 1.

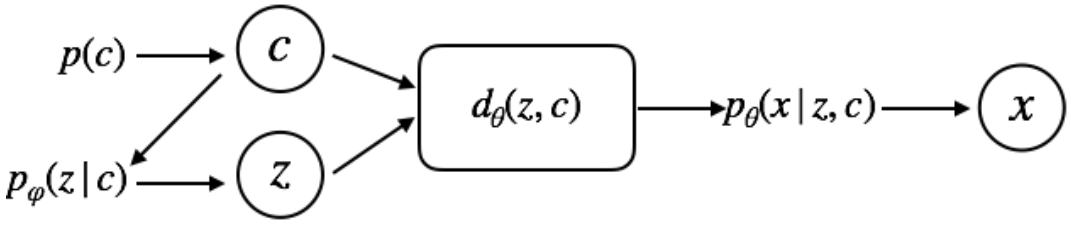


Figure 1. To generate new examples from the learned data distribution $p_\theta(\mathbf{x})$, we sample the discrete and continuous latent variables from the two-level prior and pass those through the decoder.

4.2. Inference Model

As in standard VAEs, we employ amortized variational inference to learn the unknown data distribution. We use the approximate posterior distribution

$$q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x}) = q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})q_\phi(\mathbf{c}|\mathbf{x}) \tag{10}$$

in place of the intractable posterior $p_\theta(\mathbf{z},\mathbf{c}|\mathbf{x})$.

Our approximate posterior replicates the two-level hierarchical structure of the prior. In this way, we ensure that the latent samples are structurally equivalent both during inference and new samples generations. This is not the case in other hierarchical latent models that rely on simplifying mean field assumptions for the posterior inference [7,26].

We use encoder network with a gated layer $e_\phi(\mathbf{x}) = (\pi_\phi(\mathbf{x}), \mu_\phi(\mathbf{x},\pi), \sigma_\phi(\mathbf{x},\pi))$ for the amortized inference of the variational approximate posteriors, Figure 2

$$\begin{aligned} q_\phi(\mathbf{c}|\mathbf{x}) &= \text{Cat}\left(\pi_\phi(\mathbf{x})\right) \\ q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c}) &= \mathcal{N}(\mu_\phi(\mathbf{x},\pi), \text{diag}(\sigma^2_\phi(\mathbf{x},\pi))) \ . \end{aligned}$$

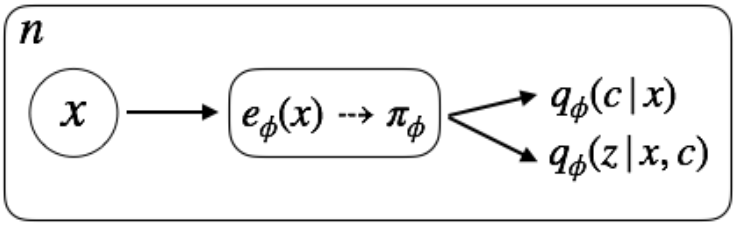


Figure 2. The encoder infers the parameters of the discrete and continuous approximate posteriors using a gated layer for the hierarchical conditioning. First, it outputs the parameters of the discrete latent variable, π_ϕ. Subsequently, there is an extra layer that takes as input π_ϕ concatenated with the last hidden layer of the encoder and infers the parameters of the continuous latent variable.

4.3. Optimization Objective

As customary in variational inference methods, our optimization objective is the maximization of the lower bound on the data log-likelihood

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c})}_{A} - \underbrace{\mathrm{KL}\left(q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})||p_\varphi(\mathbf{z},\mathbf{c})\right)}_{B} \quad . \tag{11}$$

This is a straightforward adaptation of the bound from Equation (6) to the compositional latent code $(\mathbf{z},\mathbf{c})$ with similar interpretations for the A and B terms. Using the prior and posterior distribution decompositions from Equation (10) the KL term in B can be rewritten as a sum of two KL divergences that are more amenable to practical implementation: $B1$ for the continuous conditional distributions and $B2$ for the discrete.

$$\underbrace{\mathrm{KL}\left(q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})||p_\varphi(\mathbf{z},\mathbf{c})\right)}_{B} = \underbrace{\mathbb{E}_{q_\phi(\mathbf{c}|\mathbf{x})}\, \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right)}_{B1} + \underbrace{\mathrm{KL}\left(q_\phi(\mathbf{c}|\mathbf{x})||p(\mathbf{c})\right)}_{B2} \tag{12}$$

The first term $B1$ can be seen as a weighted average of the KL divergences between the posterior and prior conditionals. The weights are the probabilities of the posterior categorical distribution, so that the two conditionals are pushed together more strongly for those observations $\mathbf{x}$ and latent categories c_k to which the model assigns high probability. The KLs can be conveniently evaluated in a closed form as both the posterior and the prior conditionals are diagonal Gaussians.

The minimization of the KL divergence between the categorical posterior and the fixed uniform prior in the second term $B2$ is equivalent to maximizing the entropy of the categorical posterior $\mathbb{H}\left(q_\phi(\mathbf{c}|\mathbf{x})\right)$ (up to a constant).

$$\underbrace{\mathrm{KL}\left(q_\phi(\mathbf{c}|\mathbf{x})||p(\mathbf{c})\right)}_{B2} = -\mathbb{H}\left(q_\phi(\mathbf{c}|\mathbf{x})\right) + \log K \tag{13}$$

We train the model by a stochastic gradient-based algorithm (Adam [30]). As the gradients of the variational lower bound $\mathcal{L}_{\theta,\phi}$ with respect to the model parameters are intractable, we use the usual well-established Monte-Carlo methods for their estimation.

For the decoder parameters θ, the gradient is estimated as the sample gradient of the conditional log-likelihood with the latent $\mathbf{z}$ and $\mathbf{c}$ sampled from the approximate posterior.

$$\nabla_\theta \mathcal{L}_{\theta,\phi}(\mathbf{x}) \approx \nabla_\theta \log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c}), \quad (\mathbf{z},\mathbf{c}) \sim q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x}) \tag{14}$$

For the encoder parameters ϕ, we use the pathwise gradient estimators [31] based on the standard location-scale $\mathbf{z} = f_\phi(\tilde{\mathbf{z}})$ and Gumbel–Softmax [32] $\mathbf{c} = g_\phi(\tilde{\mathbf{c}})$ reparametrizations with the auxiliary $\tilde{\mathbf{z}} \sim \mathcal{N}(0,1)$ sampled from the standard normal and $\tilde{\mathbf{c}}$ sampled from the Gumbel(0,1) distribution.

$$\nabla_\phi \mathcal{L}_{\theta,\phi}(\mathbf{x}) \approx \nabla_\phi \log p_\theta(\mathbf{x}, f_\phi(\tilde{\mathbf{z}}), g_\phi(\tilde{\mathbf{c}})) - \nabla_\phi \log q_\phi(f_\phi(\tilde{\mathbf{z}}), g_\phi(\tilde{\mathbf{c}})|\mathbf{x}), \quad \tilde{\mathbf{z}} \sim \mathcal{N}(0,1),\ \tilde{\mathbf{c}} \sim \text{Gumbel}(0,1) \tag{15}$$

Finally, the gradients with respect to the parameters φ of the conditional prior are estimated alongside the gradients of the decoder under the same sampling of the latents.

$$\nabla_\varphi \mathcal{L}_{\theta,\phi}(\mathbf{x}) \approx -\nabla_\varphi \log p_\varphi(\mathbf{z}|\mathbf{c}), \quad (\mathbf{z}, \mathbf{c}) \sim q_\phi(\mathbf{z}, \mathbf{c}|\mathbf{x}) \tag{16}$$

4.3.1. Analysis of the Objective

The KL divergence in term B of the objective Equation (11) has important regularization effects on the model learning. We expand on the discussion of these in the standard VAE objective Equation (6) from [9] to analyse our more complex model formulation.

There are two major issues that optimizing the reconstruction term A of the objective Equation (11) in isolation could cause. First, the model could completely ignore the categorical component of the latent representation $\mathbf{c}$ by encoding all of the data points $\mathbf{x}$ into a single category with a probability $q_\phi(c_k|\mathbf{x}) = 1$ for all $\mathbf{x}$. All of the variation in the data $\mathbf{x}$ would then be captured within the continuous component of the latent representation through the single continuous posterior $q_\phi(\mathbf{z}|\mathbf{x}, c_k)$. While this would not diminish the ability of the model to reconstruct the observed data and, therefore, would not decrease the reconstruction part of the objective A, it would degrade the generative properties of our model. Specifically, with all of the data clusters pushed into a single categorical component and distributed within the continuous latent space, we would have no leverage for generating samples from the individual data distributional components, which is one of the major requirements for our method. This pathological case is essentially equivalent to learning with the standard VAE.

Second, maximizing the log-likelihood in A naturally pushes the continuous posteriors to be concentrated around their means in disjoint parts of the continuous latent space with variances tending to zero, as discussed in [9]. For such posteriors, the model could learn very specific decoding, yielding very good reconstructions with very high log-likelihoods $p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{c})$. However, the generations would again suffer as the prior used for the ancestral sampling would not cover the same areas of the latent space as used during the inference.

To analyse the reguralization effect of term B in the objective Equation (11) on the learning, we decompose the expected KL divergence into three terms and a constant (see proof in Appendix A.1):

$$\begin{aligned} \mathbb{E}_{p(\mathbf{x})} \text{KL}\left(q_\phi(\mathbf{z}, \mathbf{c}|\mathbf{x})||p_\varphi(\mathbf{z}, \mathbf{c})\right) &= \mathbb{I}_q\left((\mathbf{z}, \mathbf{c}), \mathbf{x}\right) \\ &+ \mathbb{E}_{q_\phi(\mathbf{c})} \text{KL}\left(q_\phi(\mathbf{z}|\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right) \\ &- \mathbb{H}\left(q_\phi(\mathbf{c})\right) + \log K \ . \end{aligned} \tag{17}$$

The first is the mutual information of the composite latent variable $(\mathbf{z}, \mathbf{c})$ and the data $\mathbf{x}$ under the posterior distribution q. Minimization of the KL divergence in Equation (11) pushes the mutual information between the two to be low and, therefore, prevents the overfitting of the latent representation to the training data described in the second point above.

The third term is the negative entropy of the marginal categorical posterior whose empirical evaluation over the data sample $\mathcal{S} = \{\mathbf{x}_i\}_{i=1}^n$ is often referred to as the aggregated posterior [17,33].

$$q_\phi(\mathbf{c}) = \mathbb{E}_{p(\mathbf{x})} q_\phi(\mathbf{c}|\mathbf{x}) \approx \frac{1}{N} \sum_i^N q_\phi(\mathbf{c}|\mathbf{x}_i) \tag{18}$$

The regularizer maximizes the entropy of this distribution, thus encouraging the model to use evenly all of the categories of the discrete latent code counteracting the pathological case of the first point above.

Finally, the middle term pushes the marginalized conditional posteriors of the continuous latent variable $\mathbf{z}$ to be close to the priors conditioned on the corresponding categories. It helps to distribute the variations in the data into the continuous component of the latent space in agreement between the inferential posteriors and the learned generative priors. It does so for each latent category c_k separately, putting more or less weights on the alignment, as per the importance of the latent categories established through the categorical marginal posterior $q_\phi(\mathbf{c})$. It is this term in the objective of our VAE formulation that safeguards the generative properties of the model by matching the inferential posteriors and the learned generative priors used in the ancestral sampling procedure for new data examples.

5. VAE with Continuous and Discrete Components

We unify and analyse the objective under different assumptions for the joint distribution of continuous and discrete latent variables $p_\phi(\mathbf{z}, \mathbf{c})$ and $q_\phi(\mathbf{z}, \mathbf{c}|\mathbf{x})$ in order to justify our decisions for the inference and generative model.

As in the vanila VAE, the different variations of VAE with continuous and discrete latent variables jointly optimize the generative and the inference model. Using discrete latent variables we impose a categorical distribution as the output of the encoder. We first perform a decomposition of the objective given by Equation (11) and then apply different independence assumptions about the inference and generative models.

$$\mathcal{L}(\theta, \phi) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{c})\right]}_{A} - \underbrace{\mathbb{I}_q\left((\mathbf{z}, \mathbf{c}), \mathbf{x})\right)}_{B} - \underbrace{\mathrm{KL}(q_\phi(\mathbf{z}, \mathbf{c})\|p_\varphi(\mathbf{z}, \mathbf{c}))}_{C} \tag{19}$$

5.1. Comparing the Alternative Models

To better understand the various modifications of the VAE objective with continuous and categorical latent variables, we review the possible independence assumptions for the inference and generative models as summarized in Table 1. For the marginal posterior we assume the same decomposition as for the corresponding prior, i.e., $p_\varphi(\mathbf{z}|\mathbf{c})p(\mathbf{c}) = q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})$ and $p(\mathbf{z})p(\mathbf{c}) = q_\phi(\mathbf{z})q_\phi(\mathbf{c})$.

As we show in Appendix A.2, Equation (19) can be rewritten in the general form of Equation (20) for all the models considered in Table 1.

Table 1. Independence assumptions for discrete-continuous latent variable models and the corresponding decomposition of the B and C terms in Equation (19).

Model	$q_\phi(\mathbf{z}, \mathbf{c}\|\mathbf{x})$	$p_\varphi(\mathbf{z}, \mathbf{c})$	$B1$	$C1$	Refs.
CP-VAE	$q_\phi(\mathbf{z}\|\mathbf{x}, \mathbf{c})q_\phi(\mathbf{c}\|\mathbf{x})$	$p_\varphi(\mathbf{z}\|\mathbf{c})p(\mathbf{c})$	$\mathbb{I}_q(\mathbf{z}, \mathbf{c}\|\mathbf{c})$	$\mathbb{E}_{q(c)}[\mathrm{KL}(q_\phi(\mathbf{z}\|\mathbf{c})\|p_\varphi(\mathbf{z}\|\mathbf{c}))]$	
INDq	$q_\phi(\mathbf{z}\|\mathbf{x})q_\phi(\mathbf{c}\|\mathbf{x})$	$p_\varphi(\mathbf{z}\|\mathbf{c})p(\mathbf{c})$	$\mathbb{E}_{q_\phi(\mathbf{c},\mathbf{x})}[\mathrm{KL}(q_\phi(\mathbf{z}\|\mathbf{x})\|q_\phi(\mathbf{z}\|\mathbf{c})]$	$\mathbb{E}_{q(c)}[\mathrm{KL}(q_\phi(\mathbf{z}\|\mathbf{c})\|p_\varphi(\mathbf{z}\|\mathbf{c}))]$	[7]
INDp	$q_\phi(\mathbf{z}\|\mathbf{x}, \mathbf{c})q_\phi(\mathbf{c}\|\mathbf{x})$	$p(\mathbf{z})p(\mathbf{c})$	$\mathbb{I}_q(\mathbf{z}, (\mathbf{c}, \mathbf{x}))$	$\mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z}))$	[26,34]
INDqp	$q_\phi(\mathbf{z}\|\mathbf{x})q_\phi(\mathbf{c}\|\mathbf{x})$	$p(\mathbf{z})p(\mathbf{c})$	$\mathbb{I}_q(\mathbf{z}, \mathbf{x})$	$\mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z}))$	[13,21]

$$\mathcal{L}(\phi, \theta) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{c})\right]}_{A} - B1 - \underbrace{\mathbb{I}_q(\mathbf{c}, \mathbf{x})}_{B2} - C1 - \underbrace{\mathrm{KL}(q_\phi(\mathbf{c})\|p(\mathbf{c}))}_{C2} \tag{20}$$

The terms A, $B2$ and $C2$ remain the same in all of the models, terms $B1$ and $C1$ way, as per the independence assumptions listed in Table 1. A is the negative reconstruction cost. Term $B2$, is the mutual information in the inference model between the discrete latent variable and the observed data. Through minimizing this mutual information we encourage $\mathbf{x}$ to be independent from the discrete latent variable. Term $C2$ matches the discrete marginal posterior $q_\phi(\mathbf{c})$ to the prior $p(\mathbf{c})$.

CP-VAE

In the proposed model where we do not make any independence assumption about the approximate posterior, $q_\phi(\mathbf{z}, \mathbf{c}|\mathbf{x}) = q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{c})q_\phi(\mathbf{c}|\mathbf{x})$ and the prior, $p_\varphi(\mathbf{z}|\mathbf{c}) = p_\varphi(\mathbf{z}|\mathbf{c})p(\mathbf{c})$.

The term $B1$ is the mutual information between the continuous latent variable $\mathbf{z}$ given the discrete latent variable $\mathbf{c}$ and the data $\mathbf{x}$ given the discrete latent variable $\mathbf{c}$. Inferring the continuous latent variable $\mathbf{z}$ from $\mathbf{x}$ and $\mathbf{c}$ could result in only using the information from $\mathbf{x}$ ignoring the discrete latent variable $\mathbf{c}$. By minimizing $B1$ term, we encourage $\mathbf{z}|\mathbf{c}$ and $\mathbf{x}|\mathbf{c}$ to be decoupled by removing the information of the data distribution given a category from the continuous latent variables. In this way, we ensure that, when inferring the continuous latent variable $\mathbf{z}$, the discrete latent variable will be used. Moreover, minimizing this term penalizes the first term, the negative reconstruction error, helping to avoid over-fitting.

The term $C1$ matches the marginalized conditional posteriors of the continuous latent variable $\mathbf{z}$, $q_\phi(\mathbf{z}|\mathbf{c})$ to the priors conditioned on the corresponding categories, $p_\varphi(\mathbf{z}|\mathbf{c})$ (see also Section 4.3.1) .

INDq model

In INDq model, we assume conditional independence between the continuous and discrete latent variables, $q_\phi(\mathbf{z}, \mathbf{c}|\mathbf{x}) = q_\phi(\mathbf{z}|\mathbf{x})q_\phi(\mathbf{c}|\mathbf{x})$ without making any independence assumption about the prior, $p_\varphi(\mathbf{z}, \mathbf{c}) = p_\varphi(\mathbf{z}|\mathbf{c})p(\mathbf{c})$. The continuous latent variable $\mathbf{z}$ is inferred from the observed data, while, in our model, it is inferred from the observed data and the discrete latent variable $\mathbf{c}$.

$B1$ term encourages the approximate continuous posterior, $q_\phi(\mathbf{z}|\mathbf{x})$, to be close to the conditional distribution of the continuous latent variable $\mathbf{z}$ given the discrete latent variable $\mathbf{c}$, $q_\phi(\mathbf{z}|\mathbf{c})$. This means that, even if the discrete latent variable $\mathbf{c}$ is not used to infer the continuous $\mathbf{z}$, the continuous latent variable is encouraged to contain information for the corresponding category, but it is not ensured that it will used it like in our model. The term $C1$ is the same as in the CP-VAE with the same effect.

These assumptions are made by the Variational Deep Embedding (VaDE) paper [7], where the authors proposed a clustering framework.

INDp model

In INDp model, we do not make any independence assumption about the approximate posterior $q_\phi(\mathbf{z}, \mathbf{c}|\mathbf{x}) = q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{c})q_\phi(\mathbf{c}|\mathbf{x})$, but we assume marginal independent priors $p_\varphi(\mathbf{z}, \mathbf{c}) = p(\mathbf{z})p(\mathbf{c})$.

In this model, similarly to our model, the continuous latent variable $\mathbf{z}$ is inferred from the observed data and the discrete latent variable $\mathbf{c}$.

$B1$ term is the mutual information between the continuous latent variable $\mathbf{z}$ and $(\mathbf{c}, \mathbf{x})$ pair governed by the joint distribution $q_\phi(\mathbf{c}, \mathbf{x})$. Minimizing this mutual information, we encourage $\mathbf{z}$ and $(\mathbf{c}, \mathbf{x})$ to become independent, discouraging $\mathbf{z}$ to contain any information about the discrete latent variable $\mathbf{c}$ and the data $\mathbf{x}$, even though the discrete latent variable $\mathbf{c}$ is used to infer the continuous $\mathbf{z}$.

$C1$ is the KL divergence between the marginalized continuous posterior $q_\phi(\mathbf{z})$ and the prior $p(\mathbf{z})$. This helps to produce realistic samples without relying on any information regarding the corresponding category.

In this model, none of the terms ensure that the discrete latent variable $\mathbf{c}$ will not be ignored while inferring the continuous latent variable $\mathbf{z}$. This, in combination with the non-appearance of the discrete latent variable $\mathbf{c}$ in the KL term $C1$, makes it infeasible to generate samples from a specific category, in contrast to our proposed model.

The INDp assumptions are used in the semi-supervised model by Kingma et al. in [34], where the discrete label is treated as a latent variable when missing. Their model is augmented with a discriminative loss in order to learn better the categorical approximate posterior while using the labelled data. Without the use of supervision, there is no guarantee that it would be able to generate samples from specific categories. Gaussian Mixture Variational Autoencoder (GMVAE) [26] is built upon the semi-supervised model [34] adding an extra latent variable.

INDqp model

INDqp model assumes conditional independence between the continuous and discrete latent variables and marginal independent priors. In this case, the continuous latent variable $\mathbf{z}$ is only inferred from the observed data, the same as in the INDq model.

$B1$ term minimizes the mutual information between the continuous latent variable $\mathbf{z}$ and $\mathbf{x}$. Encouraging $\mathbf{z}$ and $\mathbf{x}$ to become independent, we help to avoid over-fitting by preventing the learning of a unique $\mathbf{z}$ for each $\mathbf{x}$ (also see Section 4.3.1). The $C1$ term is the same as in the INDp model. It matches the marginalized continuous posterior $q_\phi(\mathbf{z})$ to the prior $p(\mathbf{z})$.

In contrast to our proposed model, in INDqp, none of the terms in the objective prevent the model from ignoring the discrete latent variables or guarantees samples from a specific category.

This was also experimentally found in [21], where the same independence assumptions are used in order to learn disentangled representations in an unsupervised manner. To overcome this issue, they added weights to control the capacities of the discrete and continuous latent variables. These weights are modified separately during the training (like an annealing procedure) forcing the model to encode information both in the discrete and continuous variables. Moreover, the same model is also used under the setting of continual learning [13], where a mutual information regularizer is added in order to overcome this issue.

5.2. Assuming Uniform Approximate Categorical Posterior

In this section, we examine the special case where instead of inferring the categorical posterior as in the models above, we assume that it follows a uniform distribution over K components $q_\phi(\mathbf{c}|\mathbf{x}) \sim \frac{1}{K}$. We show that the vanilla VAE is a special case of the INDqp model.

Assuming that the categorical posterior follows the uniform distribution, the marginal categorical posterior $q_\phi(\mathbf{c})$ is equal to $\frac{1}{K}$, $(q_\phi(\mathbf{c}) = \sum_n^N q_\phi(\mathbf{c}, \mathbf{x}) = \sum_n q_\phi(\mathbf{c}|\mathbf{x})p(\mathbf{x}) = \frac{1}{K}\sum_n p(\mathbf{x}) = \frac{1}{K} \times 1 = \frac{1}{K})$ and the terms $B2$ and $C2$ in Equation (20) are equal to zero $(\mathbb{I}_q(\mathbf{c}, \mathbf{x}) = 0 \,\text{and}\, \mathrm{KL}(q_\phi(\mathbf{c})\|p(\mathbf{c})) = 0)$. The objective of the INDqp model becomes

$$\begin{aligned}\mathcal{L}(\phi, \theta) &= \underbrace{\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{c})\right]}_{A} - \underbrace{\mathbb{I}_q(\mathbf{z}, \mathbf{x})}_{B1} - \underbrace{\mathrm{KL}(q(\mathbf{z})\|p_\theta(\mathbf{z}))}_{C1} \\ &= \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{c})\right] - \mathrm{KL}(q(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right] \end{aligned} \tag{21}$$

which is the VAE objective with an extra $\mathbf{c}$ in the A term. Given that we do not infer the categorical posterior, this extra $\mathbf{c}$ does not influence the model.

6. Empirical Evaluation

We validate our new conditional prior (CP-VAE) model through experiments (the implementation of our method together with the settings for replication of our experiments is available from our Bitbucket repository https://bitbucket.org/dmmlgeneva/cp-vae/) over synthetic data and three image datasets (MNIST [35], FashionMNIST [36] and Omniglot [37]). We compare the results with those produced by standard VAE (VAE), VAE with Mixture of Gaussian prior (MoG), and VAE with VampPrior (VP) [17], and the three combinations of discrete and continuous latent variable models discussed in Section 4.

We use the same structure of the encoder and decoder networks for all the methods in all our experiments not to obfuscate the analysis of the benefits of our method by various tweaks in the model architecture.

We set the dimensions of the continuous latent variable to 40, we use simple feed-forward networks with two hidden layers of 300 units each for both the encoder and the decoder, we initialise the weights according to Glorots method [38], and we utilize the gating mechanism of [39] as the element-wise non-linearity.

We trained all of the models while using ADAM optimizer [30] with learning rate 5×10^{-4} and early stopping based on the stability of the objective over a validation-set. We use a linear annealing/warm-up scheme of 100 epochs to avoid pathological local minima and numerical issues during training [16], during which the KL regularization in the objective is annealed from 0 to 1 during training.

For generating new data examples, we use the ancestral sampling strategy with the latent variables being sampled from the respective prior distributions of each method. In the simple VAE, this is from the standard normal Gaussian $\mathbf{z} \sim p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|0, I)$. In the MoG model it is from the set of learned Gaussian components $\mathbf{z} \sim p_\varphi(\mathbf{z}) = \frac{1}{K}\sum_i^K \mathcal{N}(z|\mu_k, \text{diag}(\sigma_k))$ with equal weighting. For VP, it is from the mixture of variational posteriors $\mathbf{z} \sim p_\varphi(\mathbf{z}) = \frac{1}{K}\sum_i^K q_\phi(\mathbf{z}|\mathbf{u}_k)$ over the learned set of pseudo-inputs $\mathcal{U} = \{\mathbf{u}_k\}_{i=1}^K$, which first have to be passed through the encoder network. For INDqp and INDp, we sample the continuous latent component from the standard normal Gaussian $\mathbf{z} \sim p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|0, I)$ and from the empirical aggregated posterior Equation (18) for the discrete component $\mathbf{c} \sim q_\phi(\mathbf{c})$ For our method, we follow the two level-generative process described in Section 4.1, where we use the learned conditional priors for each of the categories for sampling the continuous latent component $\mathbf{z} \sim p_\varphi(\mathbf{z}|\mathbf{c} = c_k) = \mathcal{N}(z|\mu_k, \text{diag}(\sigma_k))$ and the empirical aggregated posterior Equation (18) for the discrete component $\mathbf{c} \sim q_\phi(\mathbf{c})$. We follow a similar procedure for the INDq model.

6.1. Synthetic Data Experiments

In this section, we demonstrate the effectiveness of the CP-VAE method through experiments over synthetic data. We use a toy dataset with 50,000 examples $\mathbf{x} \in \mathbb{R}$ generated from a Gaussian mixture with two equally weighted components $\mathbf{x} \sim p(\mathbf{x}) = \frac{1}{2}\left(N(0.3, 0.05) + N(0.7, 0.05)\right)$.

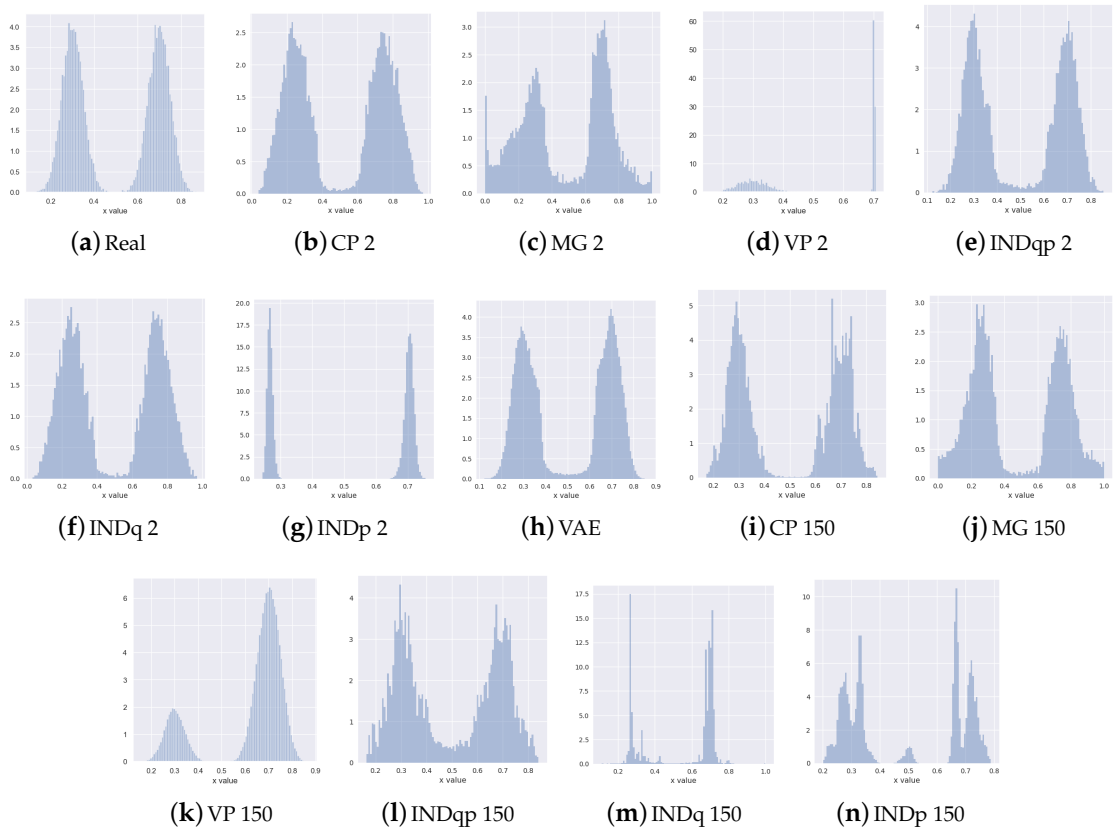
This simple set-up allows us to better understand the strengths and weaknesses of the method in terms of its density estimation performance for a known and rather simple ground-truth data distribution.

We use two experimental set-ups because, in real-life problems, the number of distributional clusters in the data (the number of mixture components) may not be known or even easy to estimate:

- **known** number of components: discrete latent variable $\mathbf{c}$ with two categories (corresponding to the ground-truth two mixture components)
- **unknown** number of components: discrete latent variable $\mathbf{c}$ with 150 categories

In Figure 3, we present histograms of data generated from the ground truth and the learned distributions. As we can see, our method (CP) correctly recovers the bi-modal structure of the data for both set-ups. This is important for practical utility of the method in situations where the domain knowledge does not provide an indication on the number of underlying generative clusters. With high enough number of categories within the discrete latent, our method can recover the correct multi-modal structure of the data. INDq has similar behaviour to our model when we use a discrete latent variable $\mathbf{c}$ with two categories, which confirms the importance of learning the conditional prior $p_\varphi(\mathbf{z}|\mathbf{c})$ instead of assuming marginal independent prior. When the number of categories is 150, it has difficulties to recover the structure of the data in contrast to our model.

Because of the simplicity of this set-up, even methods that do not adjust their priors to the disjoint learned representation, such as the simple VAE is able to recover the multimodal structure of the data at generation time. However, VAE in contrast to CP-VAE, Figure 4, because of the nature of the model, is not able to conditionally generate samples. MoG and VP have difficulties to recover the structure of the data when a small number of components/pseudo-inputs is used. This seems to improve when the number of components/pseudo-inputs is increased to 150. In contrast, INDqp and INDp have difficulties to recover the structure of the data when a large number of components is used, but this is improved when the exact number of components is used. This can be problematic in practice when the number of mixture components is not known or difficult to estimate.

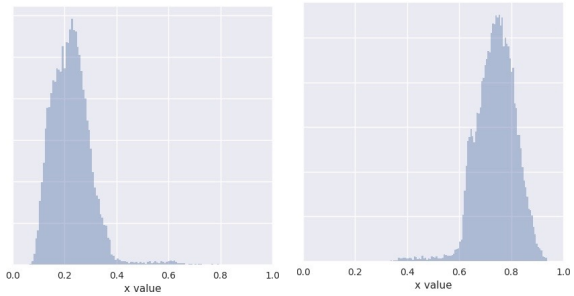


(**a**) Real (**b**) CP 2 (**c**) MG 2 (**d**) VP 2 (**e**) INDqp 2

(**f**) INDq 2 (**g**) INDp 2 (**h**) VAE (**i**) CP 150 (**j**) MG 150

(**k**) VP 150 (**l**) INDqp 150 (**m**) INDq 150 (**n**) INDp 150

Figure 3. Histograms of the data generated from the ground-truth $\mathbf{x} \sim p(\mathbf{x}) = \frac{1}{2}\left(N(0.3, 0.05) + N(0.7, 0.05)\right)$ and the learned distributions using CP-VAE, MoG, VampPrior INDq, INDp, INDqp, with 2 and 150 categories and VAE. Our CP method can recover the bi-modal structure of the data correctly, irrespective of the number of categories used for the latent categorical component.



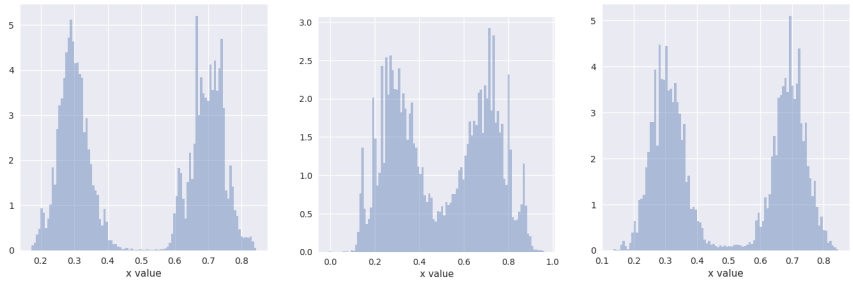
(**a**) CP (**b**) CP

Figure 4. Histograms of conditionally generate samples using our conditional prior variational autoencoder (CP-VAE) model with latent discrete variable with two categories. In subfigure **a** we conditionally generate samples from the first category and in subfigure **b** we conditionally generate samples from the first category

We further explore how our model handles the excess capacity within the categorical latent variable. For this, we focus on the 150-category case and generate data by sampling the discrete latent variable (a) from the marginal posterior $\mathbf{c} \sim q_\phi(\mathbf{c})$, (b) from the uniform prior $\mathbf{c} \sim p(\mathbf{c}) = \frac{1}{K}$.

When comparing the two in Figure 5, we see that, unlike the generations sampled from the marginal posterior, the generations from the uniform prior display some mixing artifacts. This suggests that our model learns to ignore the excess capacity by assigning low marginal probabilities $q_\phi(c_k) \approx 0$ to some of the categories. The continuous latent representations that correspond to these

parts of the disjoint latent space are irrelevant for both the reconstructions and the generations due to our weighted KL formulation in $B1$ of Equation (12).



(**a**) $\mathbf{c} \sim q_\phi(\mathbf{c})$ (**b**) $\mathbf{c} \sim p(\mathbf{c}) = \frac{1}{K}$ (**c**) reconstruction

Figure 5. CP-VAE (150-category case) generations sampled from the marginal posterior (**a**) and from the uniform prior (**b**) and CP-VAE reconstructions (**c**). The CP-VAE learns to ignore the excess capacity of the disjoint latent space by assigning near-zero probability to some of the categories in the discrete latent space. These parts of the latent space are ignored for the reconstructions and by sampling the categorical variable from the marginal posterior $q_\phi(c)$ can correctly be ignored also for the generations.

6.2. Real Data Experiments

For the real-data experiments, we use three image datasets, MNIST [35], FashionMNIST [36] and Omniglot [37], commonly used for the evaluation of generative models. We use the dynamically binarized versions of the datasets, as in [40], with the following train-validation-test splits: for MNIST and FashionMNIST 50,000–10,000–10,000, for Omniglot 23,000–1345-8070.

We examine the ability of our model to generate new examples from the underlying distributional clusters. For this, we trained our model (CP-VAE) over the FashionMNIST data with 150 categories in the discrete latent variable and compared it to the INDq, INDp, and INDqp models.

Figure 6 illustrates FashionMNIST generations using our model, Figure 7 generations using INDq model and Figure 8 sample generations using INDp model (1st row) and INDqp model (second row). For all the models the examples in each of the subplots were generated by fixing the discrete latent variable to one category and sampling the continuous latent from the corresponding learned prior for the CP-VAE and INDq models and the standard normal distribution for the INDp and INDqp. Moreover, in all the cases, we only consider the categories of the discrete latent variable with probability higher than 1/150 (this is the probability assuming the categorical marginal posterior follows the uniform distribution).

We show (Figure 6) that the learned discrete encoding in our model accurately captures the main source of variation of the data without any supervision. The unsupervised categories achieved by the model through the learned conditional prior correspond well to what a human annotator would do. Not only there are ten main categories (e.g., dresses, sandals), but our model also discovers subcategories among each main category (e.g., long and short sleeve dresses, hight heel, and flat sandals). In contrast, INDq (Figure 7) does not always capture the categories that generate a mix of images. This difference in the two models is because of two reasons. Firstly, our model learns the categorical posterior $q(\mathbf{c})$ with many more categories having non-zero probability ($q(c_k \rightarrow 0)$) compared to the INDq, Figure 9. Secondly, our model also assigns high probability to different categories for each label, while, in INDq, some categories are assigned with high probability for more than one label, Figures 10 and 11. Although INDp and INDqp are able to generate decent samples, Figure 8, none of them are able to accurately capture the categories of the data, confirming our theoretical analysis in Section 5.1.

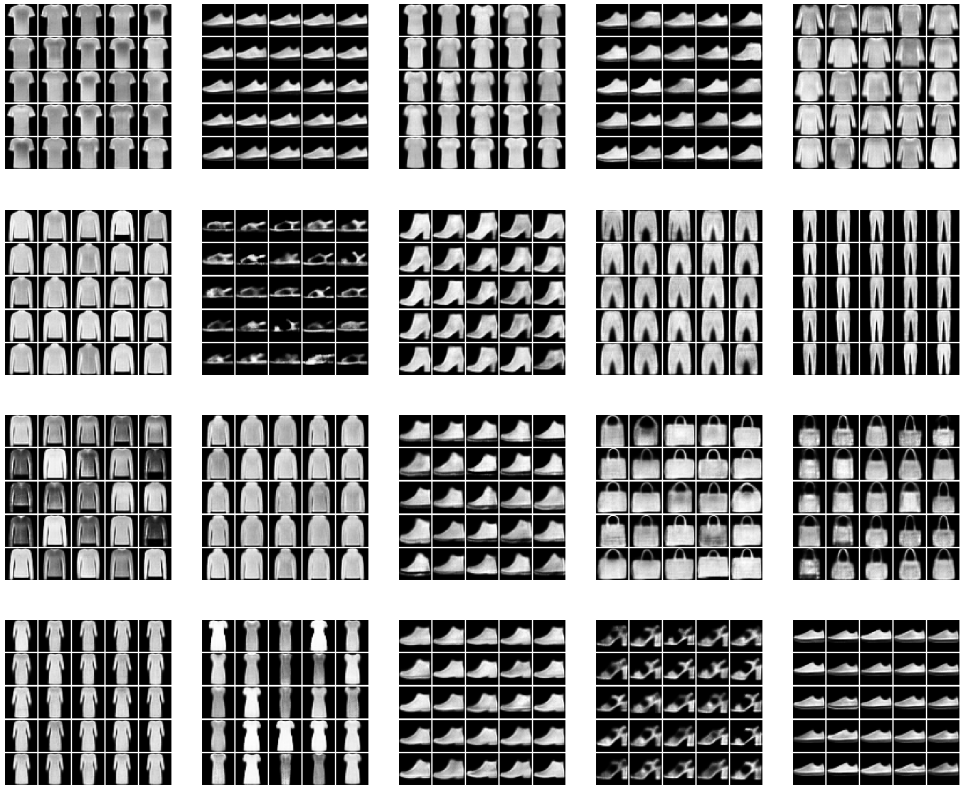


Figure 6. New data examples from the FashionMNIST generated by our CP-VAE model with latent discrete variable with 150 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples we randomly use 20 categories with probability higher than 1/150. CP-VAE accurately captures not only the main source of variation of the data, but can also find subcategories among each main category, in a totally unsupervised manner. For example we condition on category 61 and we can see in the 2nd subplot of the second row that it generates flat sandals while when we condition on category 34 in the fourth subplot of the fourth row it generates sandals with heels.

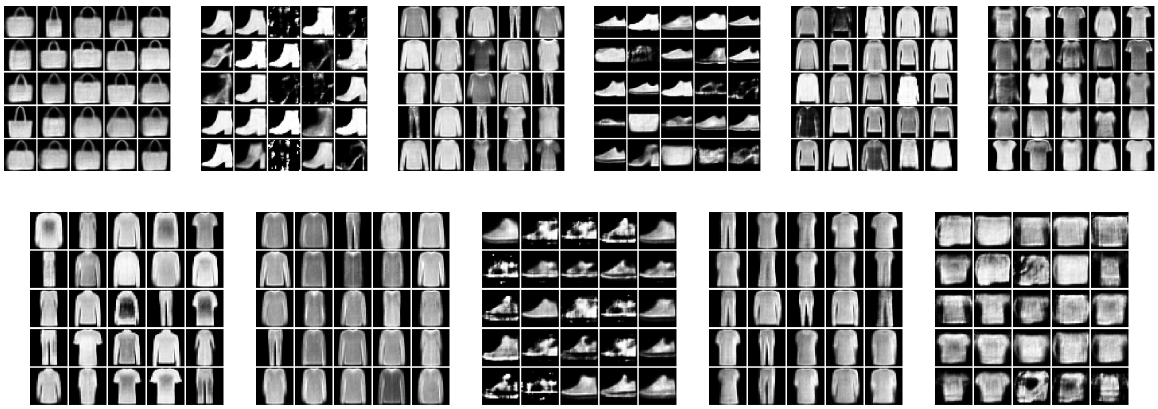


Figure 7. New data examples from the FashionMNIST generated by INDq model with a latent discrete variable with 150 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples, we use all of the categories with probability higher than 1/150. INDq model is not always able to conditionally generate new samples from the individual modes of the underlying distribution but generates a mix of images from different modes in some subplots. For example in the 1st subplot of the second row it mixes t-shirts, dresses, pullovers, and trousers, and in the third subplot of the second row it mixes flat sandals with sneakers and ankle boots.

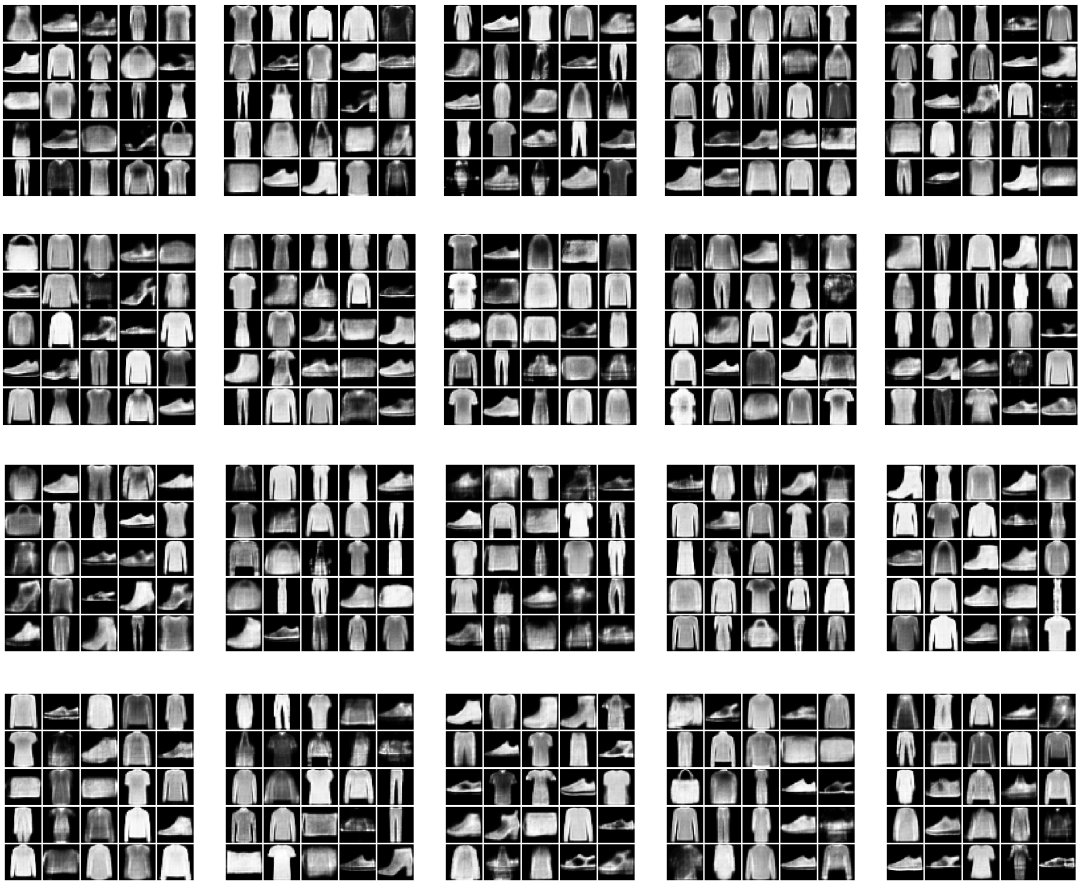
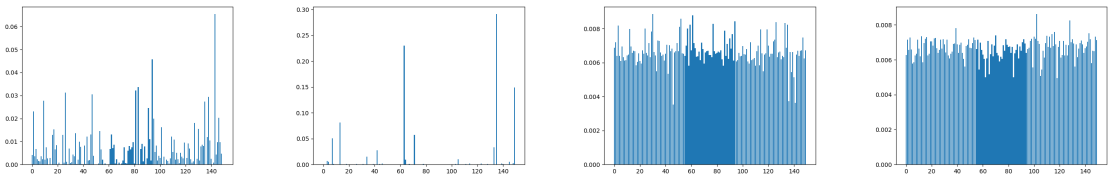


Figure 8. New data examples from the FashionMNIST generated by INDp model (1st–2nd row) and INDqp model (3rd–4th row) with a latent discrete variable with 150 categories. Samples in the same subplot were generated from the same discrete category. For both models we randomly pick 10 categories with probability higher than 1/150 in order to generate the samples. INDp and INDqp models are both not able to conditionally generate new samples from the individual modes of the underlying distribution, but generate a mix of images from different modes.

Figure 9 illustrates the categorical marginal posterior of our model, INDq, INDp, and INDqp models. As we can see in Figure 9b, in INDq the vast majority of the categories of the discrete latent variable have very low probability nearing zero. This indicates that the model learns the distributions over a small numbers of categories making it almost impossible to generate samples from different subgroups. In contrast, our model not only learns the distributions over many more categories of the discrete latent variable, Figure 9a, but also the majority of them has probability higher than 1/150. For the INDp and INDqp models, the majority of the categories are used, Figure 9c,d, resulting an almost uniform marginal categorical posterior.



(**a**) CP (**b**) INDq (**c**) INDp (**d**) INDqp

Figure 9. FashionMNIST: Marginal categorical posterior of CP-VAE (**a**), INDq (**b**), INDp (**c**) and INDqp (**d**) with discrete latent variable with 150 categories.

The discrete latent variables seems to discover the true labels in an unsupervised manner as the major source of variability and therefore we confirm this by examining the conditional marginal categorical posteriors. This is implemented by training our model without any supervision and, at the end, we use the the true labels to compute the marginal categorical posterior condition on each label. In Figure 10 and Table 2, we can see that our model uses with high probability different categories of the discrete latent variable for each label. This makes it feasible to generate new images conditioned on each label avoiding mix image generations. Moreover, our model for each label learns more than one category with high probability allowing to capture the different subgroups among the labels. In contrast, INDq, INDp and INDqp models, Figures 11 and 12, Table 2, use the same categories of the discrete latent variable in more than one label, resulting in a mix of images.

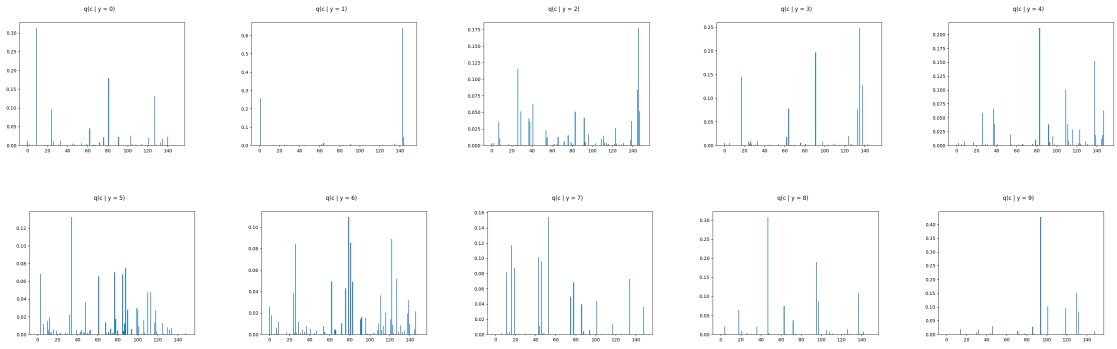


Figure 10. FashionMNIST: Marginal categorical posterior conditioned on each label of CP-VAE with discrete latent variable with 150 categories.

Table 2. FashionMNIST: first, five categories with higher probability for each label based on the marginal categorical posterior condition on each label of CPVAE and INDq with discrete latent variable with 150 categories. With bold, we mark the categories that appear in more than one label.

	CPVAE					**INDq**				
label 0	9	80	127	24	62	135	133	149	42	64
label 1	143	1	144	64	135	135	149	71	64	3
label 2	146	25	145	41	29	135	149	71	42	3
label 3	135	91	17	138	64	135	149	133	3	148
label 4	83	138	109	37	147	135	71	149	42	114
label 5	34	88	77	3	85	63	13	34	39	31
label 6	79	122	81	26	127	149	135	42	133	71
label 7	53	16	43	46	19	63	34	13	31	126
label 8	47	95	137	97	63	7	63	105	123	145
label 9	94	130	101	119	132	13	63	34	31	145

If the true class label is available at the training data, then CP-VAE is also able to generate samples from specific labels. This can be done by computing the marginal categorical posterior for each class, $q(\mathbf{c}|x \in class\, i)$ and then for each class fixing the discrete latent variable to the categories with the highest probabilities and sampling the continuous latent from the corresponding learned priors. In this way, we can generate samples form a specific label, but we can also generate samples from different subcategories of this label, by conditioning on different categories c_k Figure 13. This is just a theoretical exercise meant to show the power of our model. If the labels were truly available, they should be better used for training in a supervised manner. However, this is not the setting that we consider in the unsupervised learning problem that our CP-VAE is developed for.

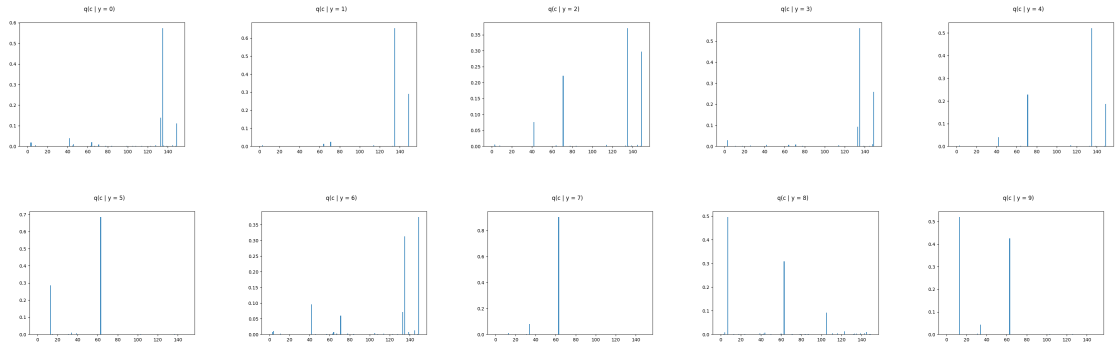


Figure 11. FashionMNIST: marginal categorical posterior conditioned on each label of INDq with discrete latent variable with 150 categories.

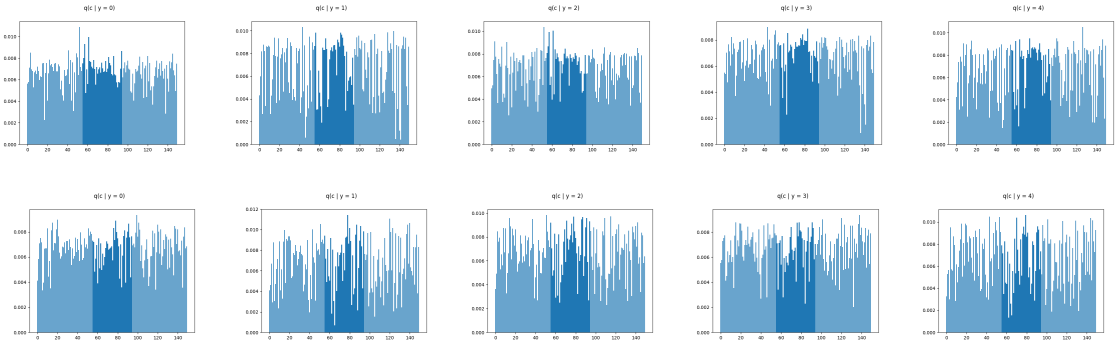


Figure 12. FashionMNIST: Marginal categorical posterior conditioned on the 5 first labels of INDp (1st row) and INDqp (2nd row) with discrete latent variable with 150 categories.

Repeating the analysis (for better flow of the text the corresponding Figures are left for the Appendix) using the MNIST data set we observe the same behaviour for our model. It uses the vast majority of the discrete latent variable with high probability allowing to discover a lot of different clusters among the data, Figure A1a in Appendix B. Furthermore, different categories are activated with high probability for each label allowing to discover important factors of data variations within each label, Figures A2 and A3. The INDq model, due to the simplicity of MNIST dataset, uses more categories of the discrete latent variable with higher probability, Figure A1b, and discovers more subgroups as compared to FashionMNIST dataset. Generating samples using the 20 categories of the discrete latent variable with the highest probability, Figure A4, we can see that the model is able to generate few samples from different subgroups but also generates mix of images for most subplots. This can also be confirmed from the marginal categorical posterior conditioned on each label, Figure A5. There are a few categories that are used only in one label, resulting in samples only from a specific subgroup, while some categories appear in more than one label with high probability, causing a generation of mixed images. The INDp and INDqp models are not able to capture the possible underlying clusters of the data, even in this relatively simple dataset Figure A6.

Unlike MNIST and FashionMNIST, which have a small number of labels with many images of each label and a large amount of data, the Omniglot dataset [37] consists of 105×105 binary images across 1628 labels with only 20 images per label. This data set allows for demonstrating that our model is able to capture some structure of the data even in regimes with limited amounts of data within a big number of categories. Our model uses the vast majority of the discrete latent variable with high probability allowing to discover a lot of different clusters among the data, Figure A7a. As Figure A8 illustrates, our model seems to recognise the modes over the original data and it is able to conditionally generate new samples from the underlying multi-modal distribution even in this more challenging dataset. INDq seems also able to discover some structure, Figure A9, but again it mostly generates a

mix of images. As in the previous data sets the INDp and INDqp models are not able to capture the possible underlying clusters of the data, Figure A10 in Appendix C.

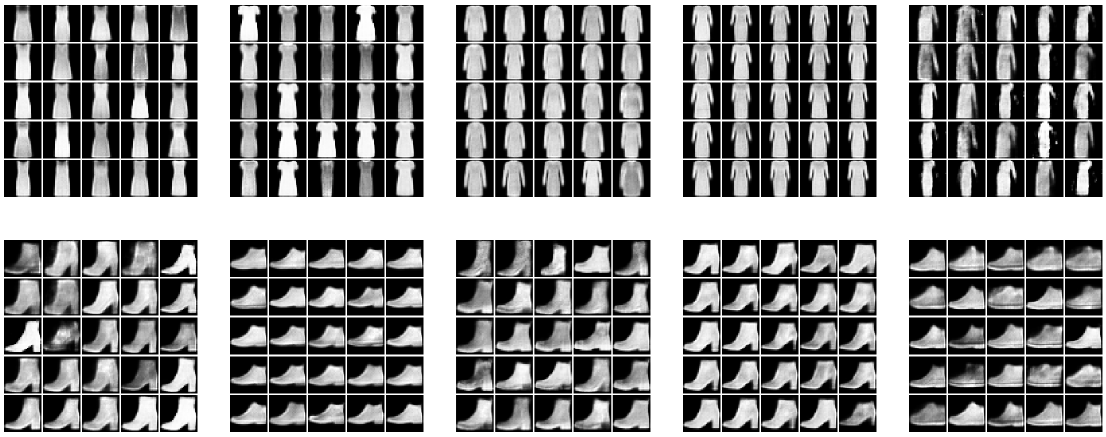


Figure 13. New variations of label specific individual FashionMNIST generated by CP-VAE model with a latent discrete variable with 150 categories. Samples in the same row belong to the same class label and samples in the same subplot were generated from the same discrete category.

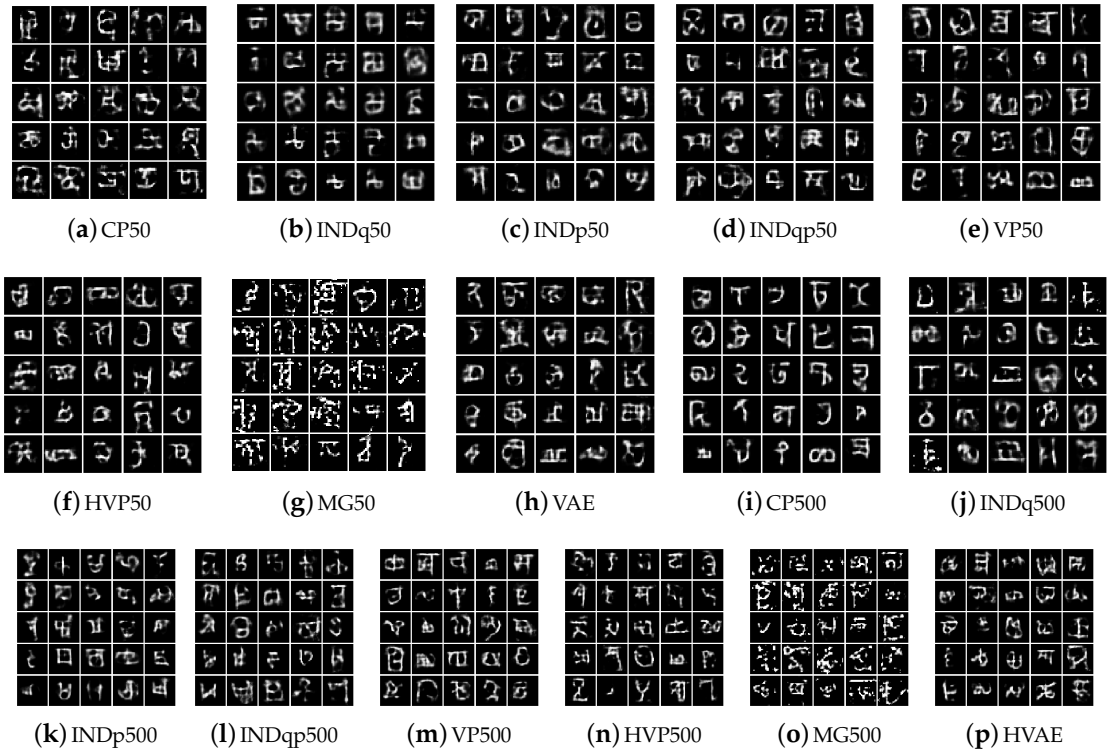
Finally, we compare the performance of our CP-VAE to a number of standard baselines varying also the size of the categorical variable. We experiment with $\{10, 150, 500\}$ categories for MNIST and FashionMNIST and $\{50, 500\}$ categories for Omniglot. The methods that we compare to are the simple VAE (VAE), the three combinations of continuous and discrete latent variable models INDq, INDp, INDqp and the following methods from [17]: VAE with Mixture of Gaussians prior (MG), VAE with VampPrior (VP), hierarchical two-layerd VAE with VampPrior (HVP), and hierarchical two-layerd VAE with simple fixed prior (HVAE). For the VP and MG methods, we use the same numbers of pseudo-inputs and mixture components as the number of the latent categories. For the two layers models we use 40 latent variables at each layer.

We summarize the numerical results in terms of the negative variational lower bound calculated over the test data in Table 3. Our model achieves better results when compared to INDq model, the other method with learned prior, in all of the cases. The INDp and INDqp seem to perform slightly better and VampPrior and especially the hierarchical VampPrior method consistently perform the best. However, this numerical evaluation should be treated with care and considered in the context. As explained in Section 3, good values of the variational lower bound objective do not guarantee good generations and certainly not good control over the distributional clusters, which is the goal of our CP-VAE.

We present the new data examples generated by the various methods in Figures 14–16 for the Omniglot, MNIST, and FashionMNIST data, respectively. Our model is able to consistently generate good quality new samples for all of the datasets, irrespective of the number of latent categorical components. The other three combinations of continuous and discrete latent models that we examine (INDq, INDqp, and INDp) are also able to generate decent samples. However, as previously explained, our model has a critical advantage, since these cannot generate conditionally. The other methods (all VampPrior variations, including the two-layer hierarchical, and the MG) fail to generate quality examples with only 10 components within the prior. They also seem to collapse to generating examples only from a few digits (items, symbols), which suggest an important lack of flexibility available for the generations. As the number of components (pseudo-inputs) in the prior mixture increases, the VampPrior generations tend to improve, with the hierarchical version of the method systematically outperforming the simple VP version.

Table 3. Comparison of negative variational lower bounds for the different methods over the test data sets.

	MNIST			**FashionMNIST**			***Omniglot***	
	c **= 10**	c **= 150**	c **= 500**	c **= 10**	c **= 150**	c **= 500**	c **= 50**	c **= 500**
CP	87.15	88.53	89.91	232.52	233.79	234.89	117.51	120.64
INDq	89.67	92.15	93.38	232.71	234.41	234.93	125.28	124.48
INDp	88.20	88.77	88.53	229.83	230.41	231.34	120.92	121.83
INDqp	87.93	88.21	88.98	228.65	230.98	231.18	119.99	120.82
VAE	88.75	—	—	231.49	—	—	115.06	—
MG	89.43	88.96	88.85	267.07	272.60	274.55	116.31	116.12
VP	87.94	86.55	86.07	230.87	229.82	270.83	114.01	113.74
HVAE	86.7	—	—	230.10	—	—	110.81	—
HVP	85.90	85.09	85.01	229.67	229.36	229.62	110.50	110.16

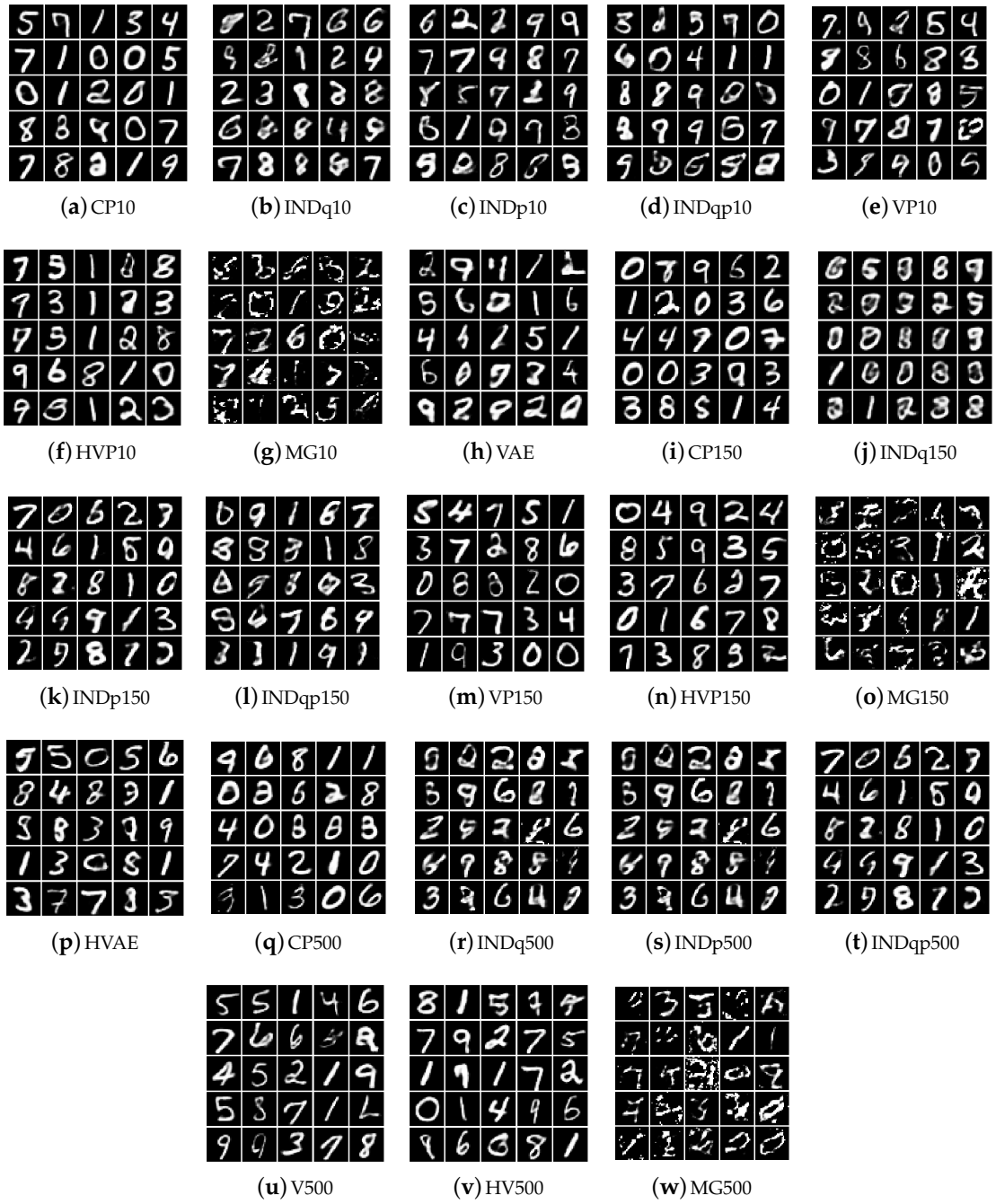


(**a**) CP50 (**b**) INDq50 (**c**) INDp50 (**d**) INDqp50 (**e**) VP50

(**f**) HVP50 (**g**) MG50 (**h**) VAE (**i**) CP500 (**j**) INDq500

(**k**) INDp500 (**l**) INDqp500 (**m**) VP500 (**n**) HVP500 (**o**) MG500 (**p**) HVAE

Figure 14. New data examples from the Omniglot dataset generated by the various methods with increasing number of the prior components.



(**a**) CP10 (**b**) INDq10 (**c**) INDp10 (**d**) INDqp10 (**e**) VP10

(**f**) HVP10 (**g**) MG10 (**h**) VAE (**i**) CP150 (**j**) INDq150

(**k**) INDp150 (**l**) INDqp150 (**m**) VP150 (**n**) HVP150 (**o**) MG150

(**p**) HVAE (**q**) CP500 (**r**) INDq500 (**s**) INDp500 (**t**) INDqp500

(**u**) V500 (**v**) HV500 (**w**) MG500

Figure 15. New data examples from the MNIST dataset generated by the various methods with increasing number of the prior components.

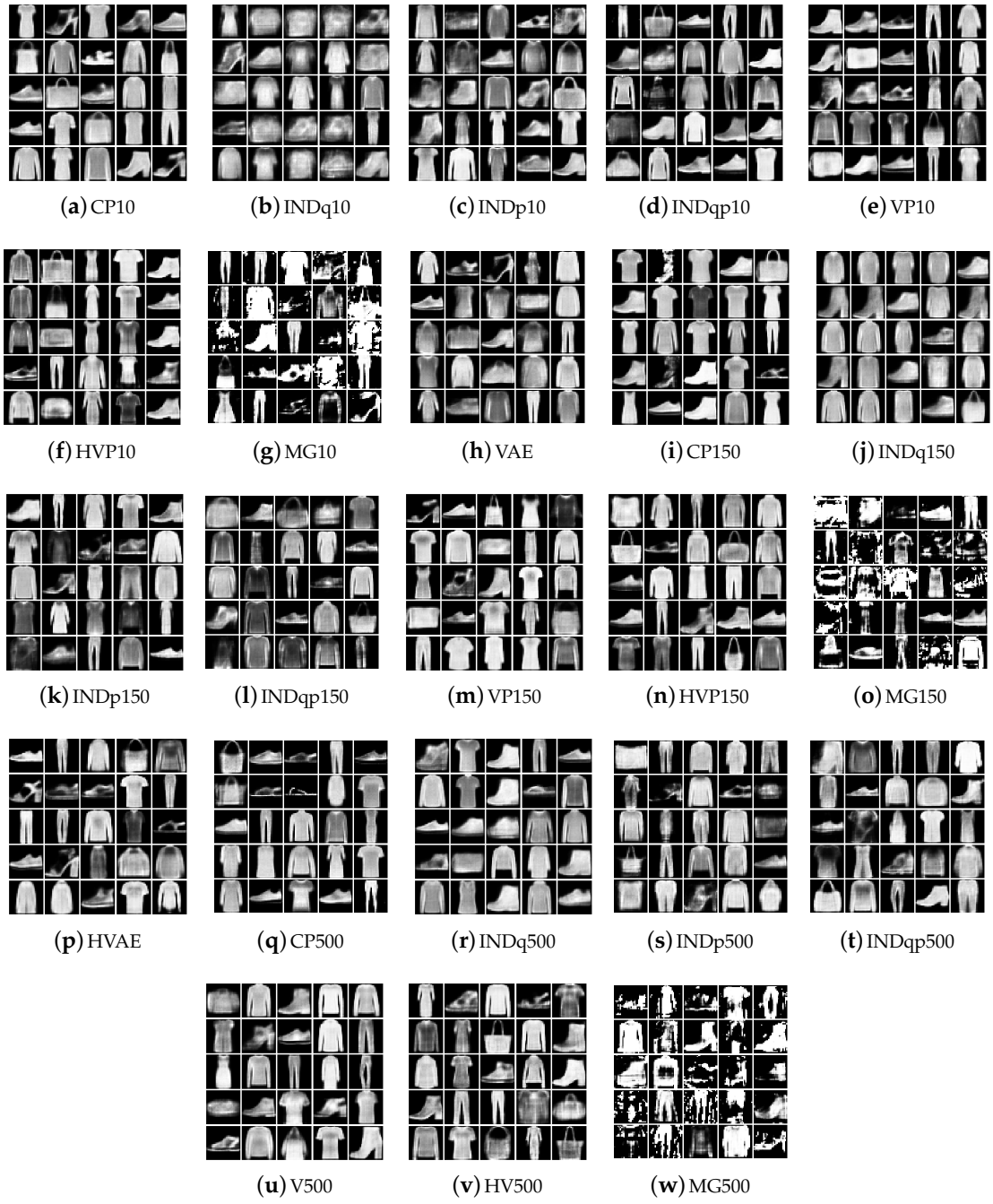


Figure 16. New data examples from the FashionMNIST dataset generated by the various methods with increasing number of the prior components.

7. Conclusions

In this paper, we introduce CP-VAE, an unsupervised generative model that is able to learn the multi-modal probabilistic structure of the data. We propose a conditionally structured latent representation that enables our model to discover the modes in the training data distribution. This is achieved by decomposing the latent representation into a continuous and a discrete component and through a better matching between prior and posterior distributions by jointly learning their parameters from the data. The experimental results demonstrate that our approach is able to recover the modes over the original data in an unsupervised manner with a performance similar to that of a human annotator and that CP-VAE is able to conditionally generate new samples from the individual

modes of the underlying distribution. In addition, we conduct a theoretical and experimental analysis of various independence assumptions on the continuous and discrete latent representations adopted in the related literature and argue in favour of our more general model formulation.

Author Contributions: Formal analysis, F.L.; Investigation, F.L., M.G. and A.K.; Methodology, F.L., M.G. and A.K.; Supervision, M.G. and A.K.; Visualization, F.L.; Writing—original draft, F.L.; Writing—review & editing, F.L., M.G. and A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been in part supported by the Swiss National Foundation grant 177179: Modelling pathological gait resulting from motor impairments.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proofs

Appendix A.1. Proofs of Section 4

Proof of Equation (12).

$$\underbrace{\mathrm{KL}\left(q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})||p_\varphi(\mathbf{z},\mathbf{c})\right)}_{B} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})q_\phi(\mathbf{c}|\mathbf{x})}\log\frac{q_\phi(\mathbf{c}|\mathbf{x})}{p(\mathbf{c})}\frac{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})}{p_\varphi(\mathbf{z}|\mathbf{c})}$$
$$= \underbrace{\mathrm{KL}\left(q_\phi(\mathbf{c}|\mathbf{x})||p(\mathbf{c})\right)}_{B1} + \underbrace{\mathbb{E}_{q_\phi(\mathbf{c}|\mathbf{x})}\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right)}_{B2}$$

□

Proof of Equation (13).

$$\underbrace{\mathrm{KL}\left(q_\phi(\mathbf{c}|\mathbf{x})||p(\mathbf{c})\right)}_{B1} = \mathbb{E}_{q_\phi(\mathbf{c}|\mathbf{x})}\left[\log q_\phi(\mathbf{c}|\mathbf{x}) - \log K^{-1}\right] = -\mathbb{H}\left(q_\phi(\mathbf{c}|\mathbf{x})\right) + \log K$$

□

Appendix A.2. Proofs of Section 5

Proof of Equation (17).

$$\begin{aligned}
&\mathbb{E}_{p(\mathbf{x})}\mathrm{KL}\left(q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})||p_\varphi(\mathbf{z},\mathbf{c})\right)\\
&\quad= \mathbb{E}_{p(\mathbf{x})}\mathrm{KL}\left(q_\phi(\mathbf{c}|\mathbf{x})||p(\mathbf{c})\right) + \mathbb{E}_{p(\mathbf{x})}\mathbb{E}_{q_\phi(\mathbf{c}|\mathbf{x})}\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right)\\
&\quad= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{c})}\left[\log\frac{q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c})} + \log\frac{q_\phi(\mathbf{c})}{p(\mathbf{c})}\right] + \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}\left[\log\frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c}|\mathbf{x})q_\phi(\mathbf{z}|\mathbf{c})} + \log\frac{q_\phi(\mathbf{z}|\mathbf{c})}{p_\varphi(\mathbf{z}|\mathbf{c})}\right]\\
&\quad= \mathrm{KL}\left(q_\phi(\mathbf{c})||p(\mathbf{c})\right) + \mathbb{E}_{q_\phi(\mathbf{c})}\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right) + \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}\log\frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})}\\
&\quad= \log K - \mathbb{H}(q_\phi(\mathbf{c})) + \mathbb{E}_{q_\phi(\mathbf{c})}\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right) + \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}\log\frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z},\mathbf{c})}\\
&\quad= \log K - \mathbb{H}(q_\phi(\mathbf{c})) + \mathbb{E}_{q_\phi(\mathbf{c})}\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{c})||p_\varphi(\mathbf{z}|\mathbf{c})\right) + \mathbb{I}_q\left((\mathbf{z},\mathbf{c}),\mathbf{x}\right)
\end{aligned}$$

□

Proof of Equation (19).

$$\begin{aligned}\mathcal{L}(\theta,\phi) &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{p_\theta(\mathbf{x},\mathbf{z},\mathbf{c})}{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}+\log\frac{p_\varphi(\mathbf{z},\mathbf{c})}{p_\varphi(\mathbf{z},\mathbf{c})}+\log\frac{q_\phi(\mathbf{z},\mathbf{c})}{q_\phi(\mathbf{z},\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{p_\theta(\mathbf{x},\mathbf{z},\mathbf{c})}{p_\varphi(\mathbf{z},\mathbf{c})}\right]-\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z},\mathbf{c})}\right]-\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z},\mathbf{c})}{p_\varphi(\mathbf{z},\mathbf{c})}\right]\\ &= \underbrace{\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c})\right]}_{A}-\underbrace{\mathbb{I}_q\left((\mathbf{z},\mathbf{c}),\mathbf{x}\right)}_{B}-\underbrace{\mathrm{KL}(q_\phi(\mathbf{z},\mathbf{c})\|p_\varphi(\mathbf{z},\mathbf{c}))}_{C}\end{aligned}$$

□

The staring point for all the models is Equation (19). Here we decompose the terms B and C as per the independence assumptions listed in Table 1.

Proof of CP-VAE objective:

$$\begin{aligned}B:\quad I_q((\mathbf{z},\mathbf{c}),\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z},\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})}+\log\frac{q_\phi(\mathbf{x}|\mathbf{c})}{q_\phi(\mathbf{x}|\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z},\mathbf{x}|\mathbf{c})}{q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{x}|\mathbf{c})}\right]+\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c})}\right]\\ &= \mathbb{I}_q(\mathbf{z}|\mathbf{c},\mathbf{x}|\mathbf{c})+\mathbb{I}_q(\mathbf{c},\mathbf{x})\\ &= B1+B2\end{aligned}$$

$$\begin{aligned}C:\quad \mathrm{KL}(q_\phi(\mathbf{z},\mathbf{c})\|p_\varphi(\mathbf{z},\mathbf{c})) &= \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})\|p_\varphi(\mathbf{z}|\mathbf{c})p(\mathbf{c}))\\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})}\left[\log\frac{q_\phi(\mathbf{z}|\mathbf{c})}{p_\varphi(\mathbf{z}|\mathbf{c})}\right]+\mathbb{E}_{\log q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})}\left[\frac{q_\phi(\mathbf{c})}{p(\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{c})}[\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{c})\|p_\varphi(\mathbf{z}|\mathbf{c}))]+\mathrm{KL}(q_\phi(\mathbf{c})\|p(\mathbf{c}))\\ &= C1+C2\end{aligned}$$

□

Proof of INDq objective:

$$\begin{aligned}B:\quad I_q((\mathbf{z},\mathbf{c}),\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z},\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z}|\mathbf{x})q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{c})q_\phi(\mathbf{z}|\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{c})}\right]+\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log\frac{q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c})}\right]\\ &= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{c})}\,\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}))\|q_\phi(\mathbf{z}|\mathbf{c}))+\mathbb{I}_q(\mathbf{c},\mathbf{x})\\ &= B1+B2\end{aligned}$$

C term is the same as in CP-VAE. □

Proof of INDp objective:

$$\begin{aligned} B: \quad I_q((\mathbf{z},\mathbf{c}),\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z},\mathbf{c})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z})q_\phi(\mathbf{c})} + \log \frac{q_\phi(\mathbf{x},\mathbf{c})}{q_\phi(\mathbf{x},\mathbf{c})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}{q_\phi(\mathbf{z})q_\phi(\mathbf{x},\mathbf{c})}\right] + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c})}\right] \\ &= \mathbb{I}_q(\mathbf{z},(\mathbf{x},\mathbf{c})) + \mathbb{I}_q(\mathbf{c},\mathbf{x}) \end{aligned}$$

$$\begin{aligned} C: \quad \mathrm{KL}(q_\phi(\mathbf{z},\mathbf{c})\|p_\varphi(\mathbf{z},\mathbf{c})) &= \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{c})q_\phi(\mathbf{c})\|p_\varphi(\mathbf{z}|\mathbf{c})p(\mathbf{c})) \\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z})q_\phi(\mathbf{c})} + \log \frac{q_\phi(\mathbf{x},\mathbf{c})}{q_\phi(\mathbf{x},\mathbf{c})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}{q_\phi(\mathbf{z})q_\phi(\mathbf{x},\mathbf{c})}\right] + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c})}\right] \\ &= \mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z})) + \mathrm{KL}(q_\phi(\mathbf{c})\|p(\mathbf{c})) \\ &= C1 + C2 \end{aligned}$$

□

Proof of INDqp objective:

$$\begin{aligned} B: \quad I_q((\mathbf{z},\mathbf{c}),\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z},\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z},\mathbf{c})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{z})q_\phi(\mathbf{c})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z})}\right] + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{c},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{c}|\mathbf{x})}{q_\phi(\mathbf{c})}\right] \\ &= \mathbb{I}_q(\mathbf{z},\mathbf{x}) + \mathbb{I}_q(\mathbf{c},\mathbf{x}) \qquad\qquad = B1 + B2 \end{aligned}$$

C term is the same as in INDp model. □

Appendix A.3. Proofs of Section 5.2

Proof. $\mathbb{E}_{p(\mathbf{x})}\left[KL(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))\right] = \mathbb{I}_q(\mathbf{z},\mathbf{x}) + \mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z}))$:

$$\begin{aligned} E_{p(x)}\left[\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))\right] &= E_{q_\phi(\mathbf{z},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})}\right] \\ &= E_{q_\phi(\mathbf{z},\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})}\frac{q_\phi(\mathbf{z})}{q_\phi(\mathbf{z})}\right] \\ &= E_{q_\phi(z,x)}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z})}\right] + E_{q_\phi(z)}\left[\log \frac{q_\phi(\mathbf{z})}{p(\mathbf{z})}\right] \\ &= \mathbb{I}_q(\mathbf{z},\mathbf{x}) + \mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z})) \end{aligned}$$

□

Appendix A.4. Maximizing the Negative RE is Equivalent to Maximizing a Lower Bound on MI between the Latent Variables $(\mathbf{z}, \mathbf{c})$ *and* $\mathbf{x}$

Proof. The non-negativity of the KL divergence implies:

$$\begin{aligned} \mathrm{KL}(q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})\|p(\mathbf{x}|\mathbf{z},\mathbf{c})) &= E_{\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})}[\log p(\mathbf{x}|\mathbf{z},\mathbf{c})] - E_{\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})}[\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})] \geq 0 \\ &\Rightarrow E_{\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})}[\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})] \geq E_{\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})}[\log p(\mathbf{x}|\mathbf{z},\mathbf{c})] \end{aligned} \tag{A1}$$

This leads to a lower bound on the mutual Information $I_q((\mathbf{z},\mathbf{c}),\mathbf{x})$

$$\begin{aligned} I_q((\mathbf{z},\mathbf{c}),\mathbf{x}) &= \mathrm{KL}(q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})\|p_D(\mathbf{x})q_\phi(\mathbf{z},\mathbf{c})) \\ &= E_{\log q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}[\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})] - E_{\log q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}[p_D(\mathbf{x})] \\ &= E_{\log q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}[\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})] + H(p_D(x)) \\ &= E_{\log q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}[\log q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})] + H(p_D(x)) + \mathrm{KL}(q_\phi(\mathbf{x}|\mathbf{z},\mathbf{c})\|p(\mathbf{x}|\mathbf{z},\mathbf{c})) \\ &\geq E_{\log q_\phi(\mathbf{x},\mathbf{z},\mathbf{c})}[\log p(\mathbf{x}|\mathbf{z},\mathbf{c})] + H(p_D(x)) \end{aligned} \tag{A2}$$

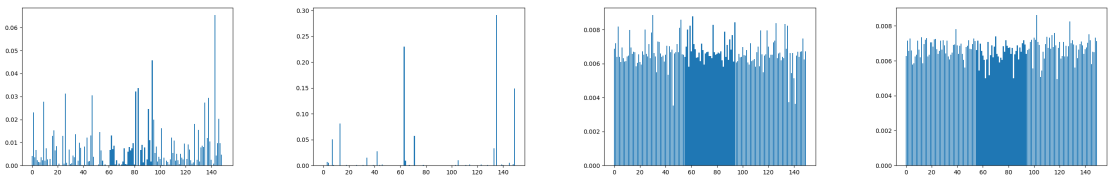
where the first term is the negative reconstruction error and the second, $H(p(x))$ is the entropy. This means that maximizing the negative reconstruction error we maximize a lower bound on $I_q((\mathbf{z},\mathbf{c}),\mathbf{x})$. □

Appendix B. MNIST

Results from training CP-VAE, INDq, INDp and INDqp over the MNIST [35] data with 150 categories in the discrete latent variable (Table A1).

Table A1. MNIST: First five categories with higher probability for each label based on the marginal categorical posterior condition on each label of CPVAE and INDq with discrete latent variable with 150 categories. With bold we mark the categories that appear in more than one labels.

	CPVAE					**INDq**				
label 0	22	102	17	59	134	67	145	56	18	32
label 1	32	48	21	9	20	94	97	85	124	67
label 2	29	55	45	95	88	31	5	48	140	65
label 3	129	5	13	146	125	122	135	125	149	99
label 4	53	40	47	122	74	107	87	120	132	80
label 5	66	60	117	63	10	43	141	56	102	149
label 6	41	35	127	130	149	109	103	111	141	98
label 7	52	0	148	108	106	135	67	125	2	99
label 8	132	103	27	100	85	1	23	15	63	40
label 9	75	42	81	144	0	67	99	18	34	79



(**a**) CP (**b**) INDq (**c**) INDp (**d**) INDqp

Figure A1. MINST: Marginal categorical posterior of CPVAE (**a**), INDq (**b**), INDp (**c**) and INDqp (**d**) with discrete latent variable with 150 categories.

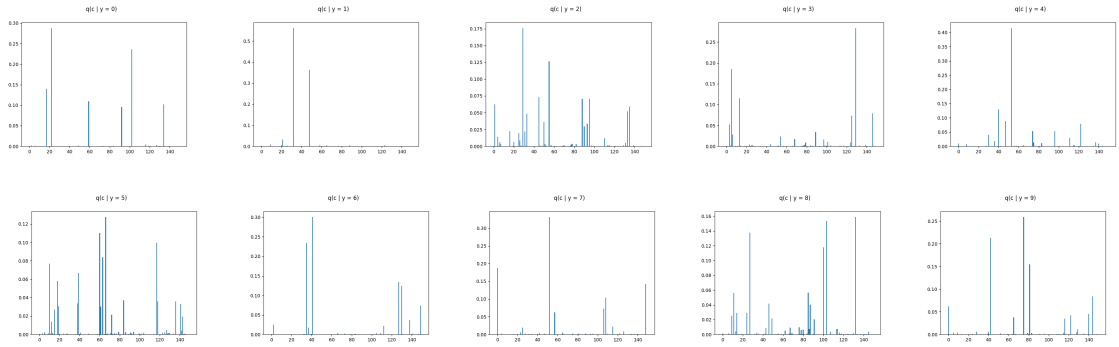


Figure A2. MINST: Marginal categorical posterior condition on each label of CPVAE with discrete latent variable with 150 categories.

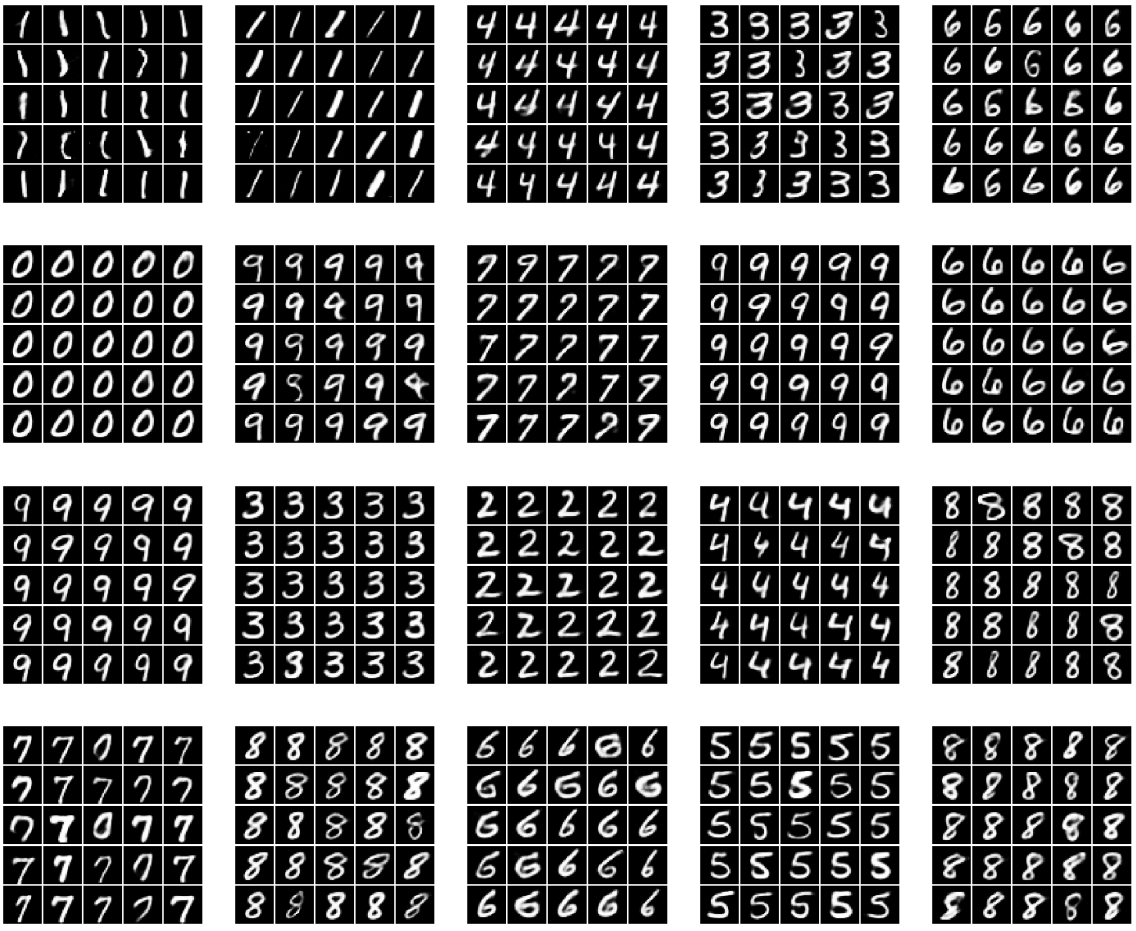


Figure A3. New variations of individual MNIST digits generated by our CP-VAE model with a latent discrete variable with 150 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples we randomly use 20 categories with probability higher than 1/150.

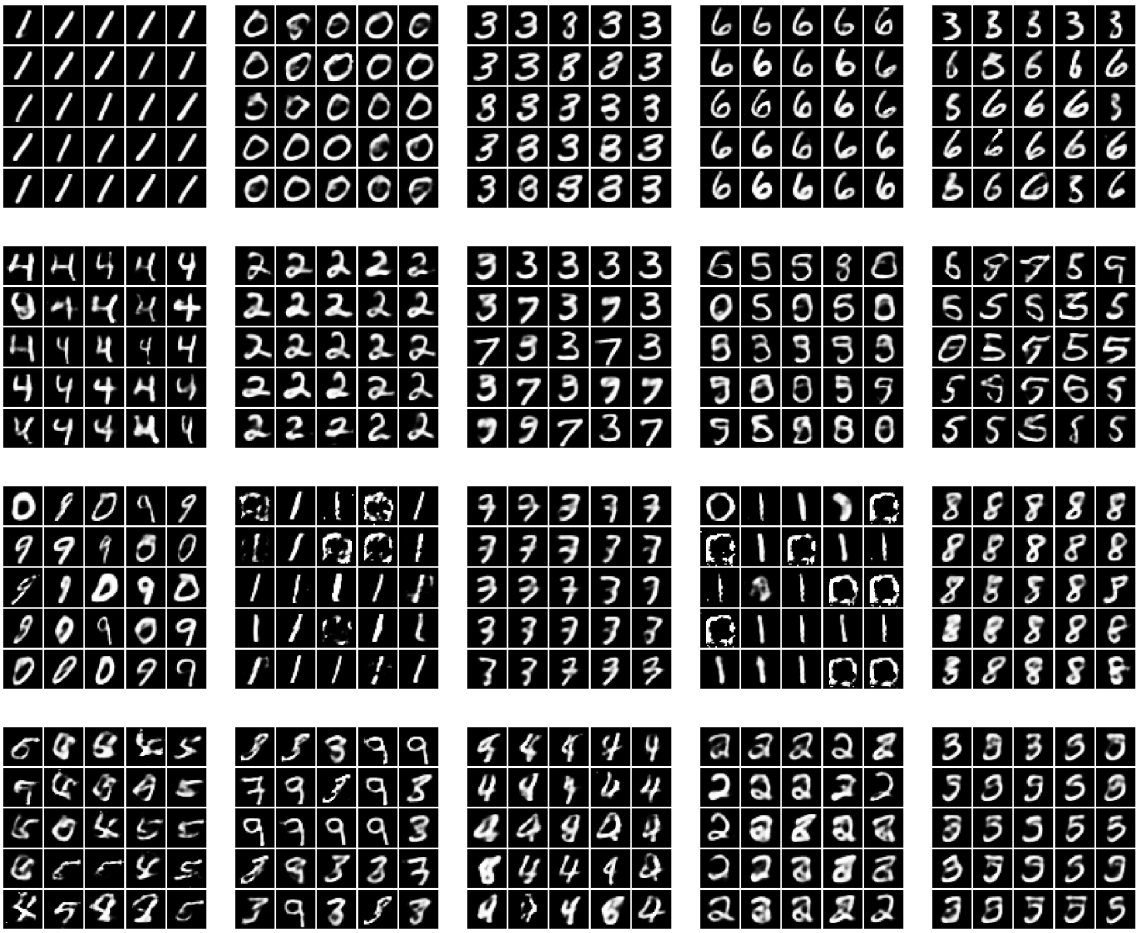


Figure A4. New variations of individual MNIST digits generated by INDq model with a latent discrete variable with 150 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples we randomly use 20 categories with probability higher than 1/150.

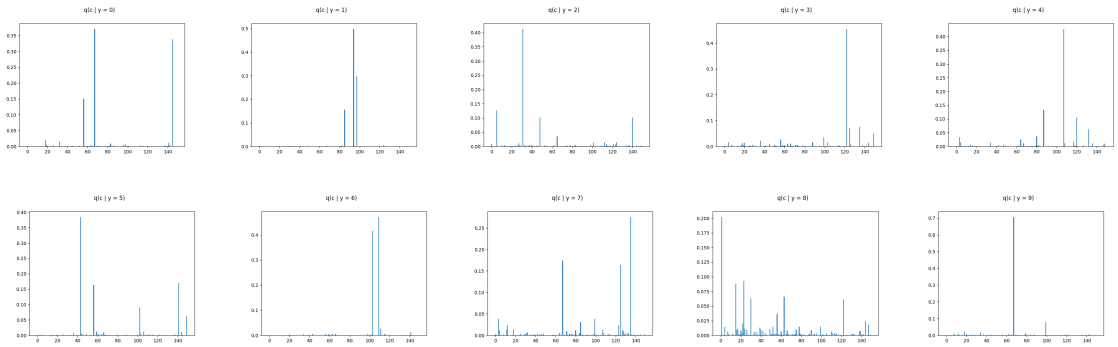


Figure A5. MINST: Marginal categorical posterior condition on each label of INDq with discrete latent variable with 150 categories.

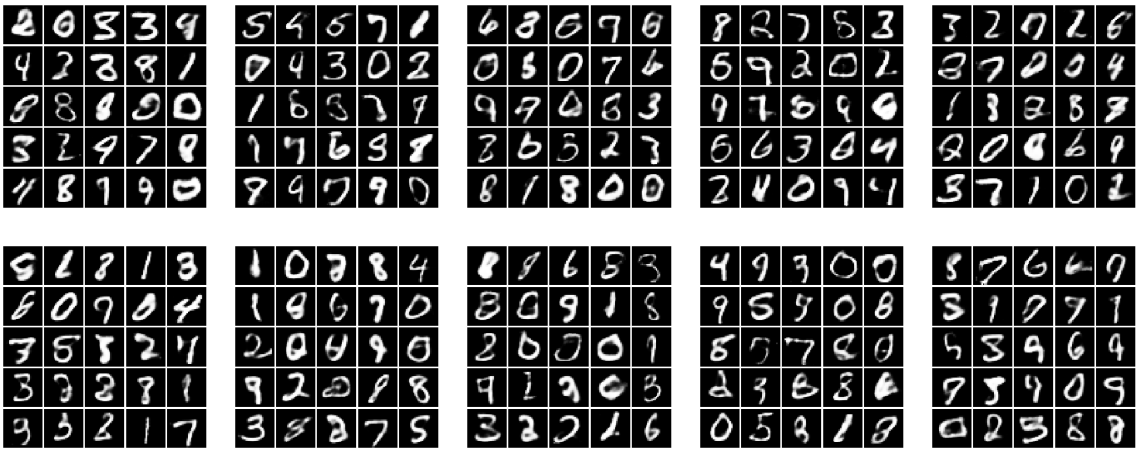
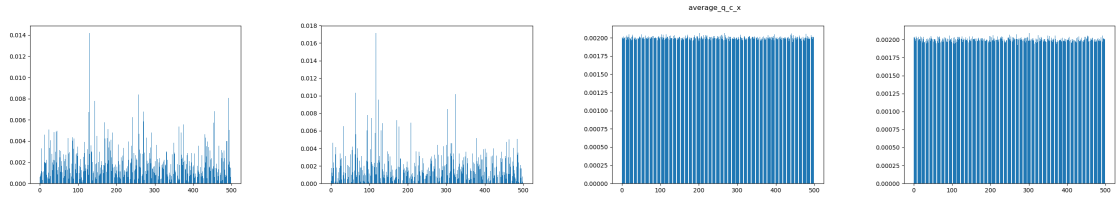


Figure A6. New variations of individual MNIST generated by INDp model (1st–2nd row) and INDqp model (3rd–4th row) with a latent discrete variable with 150 categories. Samples in the same subplot were generated from the same discrete category. For both models we randomly pick 10 categories categories with probability higher than 1/150 in order to generate the samples.

Appendix C. Omniglot

Results from training CP-VAE, INDq, INDp and INDqp over the Omniglot [37] data with 500 categories in the discrete latent variable.



(**a**) CP (**b**) INDq (**c**) INDp (**d**) INDqp

Figure A7. Omniglot: Marginal categorical posterior of CPVAE (**a**), INDq (**b**), INDp (**c**) and INDqp (**d**) with discrete latent variable with 500 categories.

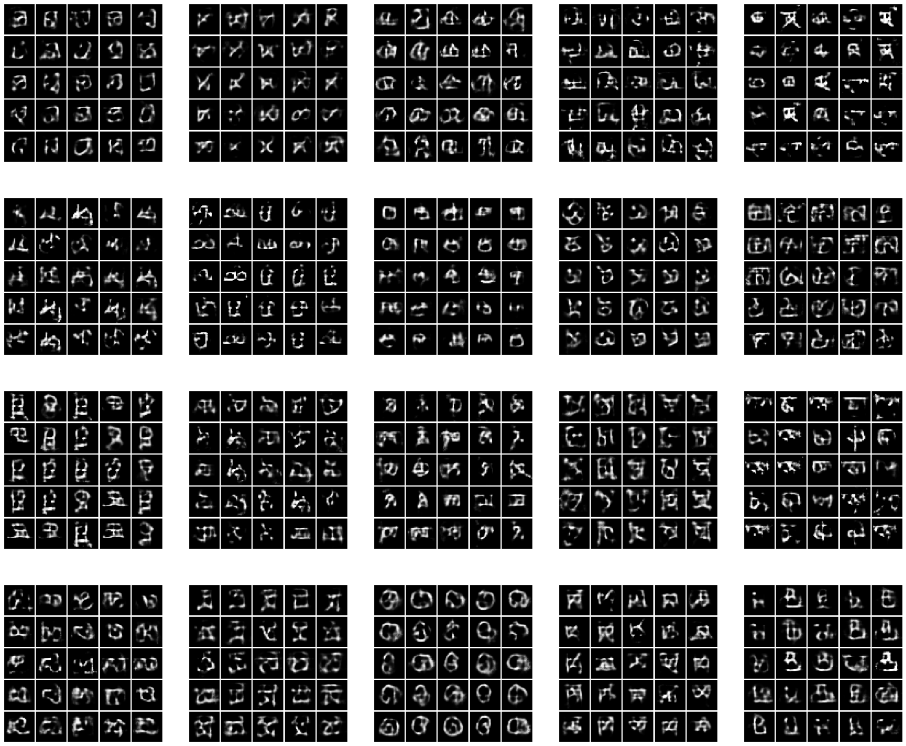


Figure A8. New variations of individual Omniglot symbols generated by our CP-VAE model with a latent discrete variable with 500 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples we randomly use 20 categories with probability higher than 1/500.

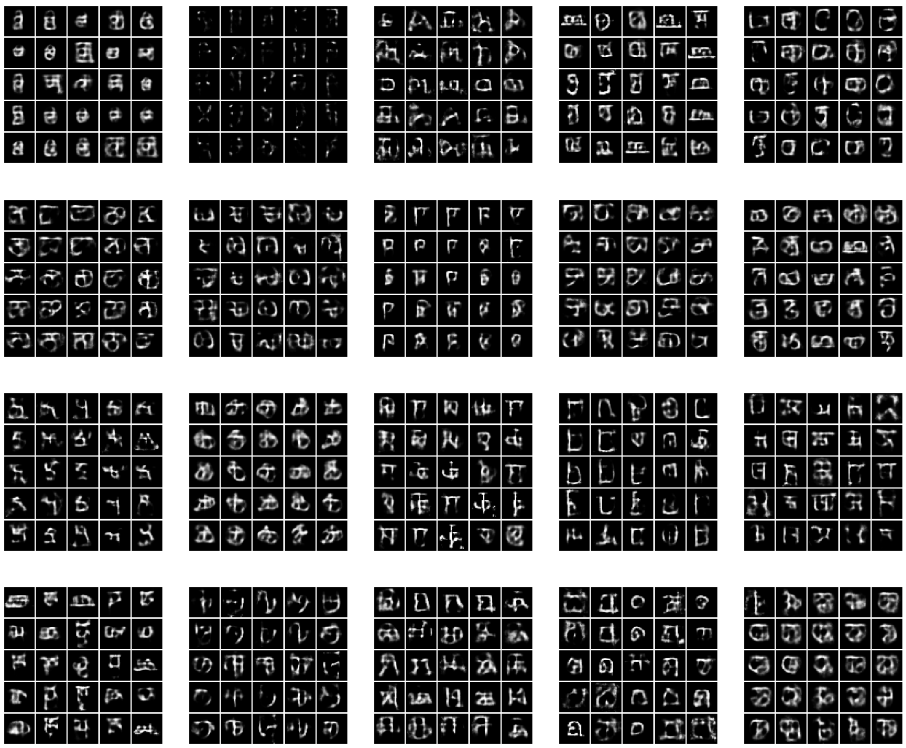


Figure A9. New variations of individual Omniglot symbols generated by INDq model with a latent discrete variable with 500 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples we randomly use 20 categories with probability higher than 1/500.

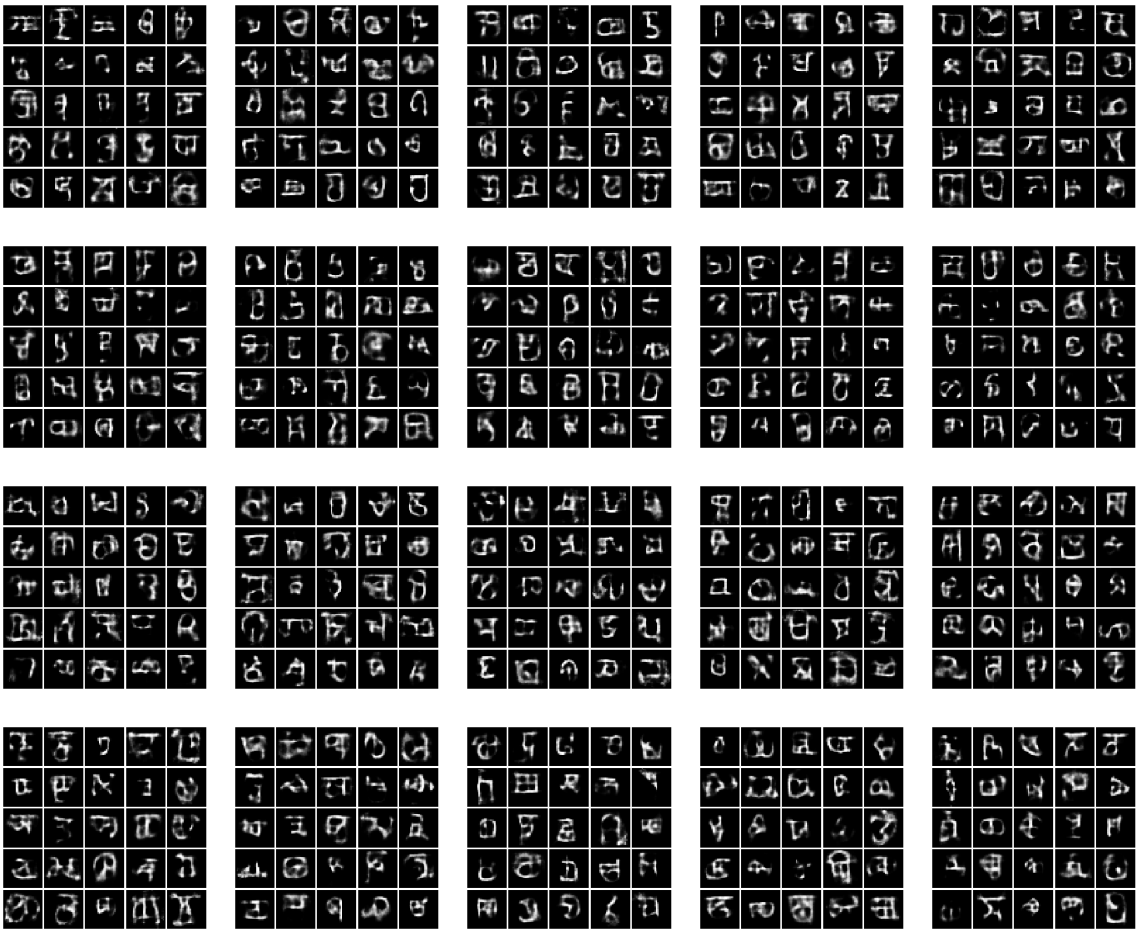


Figure A10. New variations of individual Omniglot symbols generated by INDp model (1st–2nd row) and INDqp model (3rd–4rth row) with a latent discrete variable with 500 categories. Examples in the same subplot were generated from the same discrete category. To generate the samples we randomly use 10 categories with probability higher than 1/500.

References

1. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. In Proceedings of the 2nd International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
2. Rezende, D.J.; Mohamed, S.; Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In Proceedings of the International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 1278–1286.
3. Rezende, D.J.; Mohamed, S. Variational inference with normalizing flows. In Proceedings of the 32 International Conference on Machine Learning, Lille, France, 6–11 July 2015; p. 9.
4. Burda, Y.; Grosse, R.B.; Salakhutdinov, R. Importance weighted autoencoders. In Proceedings of the 4th International Conference on Learning Representations (ICLR), San Juan, Puerto Rico, 2–4 May 2016.
5. Kingma, D.P.; Salimans, T.; Jozefowicz, R.; Chen, X.; Sutskever, I.; Welling, M. Improved variational inference with inverse autoregressive flow. In Proceedings of the Advances in Neural Information Processing Systems 2016, Barcelona, Spain, 5–10 December 2016.
6. Chen, X.; Kingma, D.P.; Salimans, T.; Duan, Y.; Dhariwal, P.; Schulman, J.; Sutskever, I.; Abbeel, P. Variational lossy autoencoders. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017; p. 17.
7. Jiang, Z.; Zheng, Y.; Tan, H.; Tang, B.; Zhou, H. Variational deep embedding: An unsupervised and generative approach to clustering. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence,Melbourne, Australia, 19–25 August 2017.

8. Nalisnick, E.; Smyth, P. Stick-breaking variational autoencoders. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
9. Zhao, S.; Song, J.; Ermon, S. InfoVAE: Information maximizing variational autoencoders. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI) 2017, San Francisco, CA, USA, 4–9 February 2017.
10. Alemi, A.A.; Poole, B.; Fischer, I.; Dillon, J.V.; Saurous, R.A.; Murphy, K. Fixing a broken ELBO. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018.
11. Davidson, T.R.; Falorsi, L.; De Cao, N.; Kipf, T.; Tomczak, J.M. Hyperspherical variational auto-encoders. *arXiv* **2018**, arXiv:1804.00891.
12. Dai, B.; Wipf, D. Diagnosing and enhancing VAE models. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
13. Ramapuram, J.; Gregorova, M.; Kalousis, A. Lifelong generative modeling. *arXiv* **2019**, arXiv:1705.09847.
14. Lavda, F.; Ramapuram, J.; Gregorova, M.; Kalousis, A. Continual classification learning using generative models. Continual learning workshop, Advances in Neural Information Processing Systems 2018, Montreal, QC, Canada, 3–8 December 2018. *arXiv* **2018**, arXiv:1810.10612.
15. Locatello, F.; Bauer, S.; Lucic, M.; Rätsch, G.; Gelly, S.; Schölkopf, B.; Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In Proceedings of the 36th International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019.
16. Bowman, S.R.; Vilnis, L.; Vinyals, O.; Dai, A.; Jozefowicz, R.; Bengio, S. Generating Sentences from a Continuous Space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*; Association for Computational Linguistics: Berlin, Germany, 2016; pp. 10–21. [CrossRef]
17. Tomczak, J.M.; Welling, M. VAE with a VampPrior. In Proceedings of the 21st International Conference on Artificial Intelligence and Statistics (AISTATS), Playa Blanca, Lanzarote, Spain, 9–11 April 2018.
18. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. Beta-VAE: Learning basic visual concepts with a constrained variational framework. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017.
19. Hoffman, M.D.; Johnson, M.J. ELBO surgery: Yet another way to carve up the variational evidence lower bound. In Proceedings of the NIPS Symposium on Advances in Approximate Bayesian Inference, Montreal, QC, Canada, 2 December 2018; p. 4.
20. Kim, H.; Mnih, A. Disentangling by factorising. *arXiv* **2018**, arXiv:1802.05983.
21. Dupont, E. Learning disentangled joint continuous and discrete representations. In Proceedings of the Advances in Neural Information Processing Systems 2018, Montreal, QC, Canada, 3–8 December 2018.
22. Gulrajani, I.; Kumar, K.; Ahmed, F.; Taiga, A.A.; Visin, F.; Vazquez, D.; Courville, A. PixelVAE: A latent variable model for natural images. In Proceedings of the International Conference on Learning Representations (ICLR) 2017, Toulon, France, 24–26 April 2017.
23. Chung, J.; Kastner, K.; Dinh, L.; Goel, K.; Courville, A.C.; Bengio, Y. A recurrent latent variable model for sequential data. In *Advances in Neural Information Processing Systems 28*; Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2015; pp. 2980–2988.
24. Xu, J.; Durrett, G. Spherical latent spaces for stable variational autoencoders. In Proceedings of the Conference on Empirical Methods in Natural Language Processing 2018, Brussels, Belgium, 31 October–4 November 2018.
25. Takahashi, H.; Iwata, T.; Yamanaka, Y.; Yamada, M.; Yagi, S. Variational autoencoder with implicit optimal priors. In Proceedings of the AAAI Conference on Artificial Intelligence 2019, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 5066–5073.
26. Dilokthanakul, N.; Mediano, P.A.M.; Garnelo, M.; Lee, M.C.H.; Salimbeni, H.; Arulkumaran, K.; Shanahan, M. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv* **2017**, arXiv:1611.02648.
27. Goyal, P.; Hu, Z.; Liang, X.; Wang, C.; Xing, E. Nonparametric variational auto-encoders for hierarchical representation learning. In Proceedings of the IEEE International Conference on Computer Vision 2017, Venice, Italy, 22–29 October 2017.
28. Li, X.; Chen, Z.; Poon, L.K.M.; Zhang, N.L. Learning Latent Superstructures in Variational Autoencoders for Deep Multidimensional Clustering. In Proceedings of the International Conference on Learning Representations 2019, New Orleans, LA, USA, 6–9 May 2019.

29. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; Adaptive Computation and Machine Learning Series; MIT Press: Cambridge, MA, USA, 2012.
30. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference for Learning Representations 2015, San Diego, CA, USA, 7–9 May 2015.
31. Mohamed, S.; Rosca, M.; Figurnov, M.; Mnih, A. Monte carlo gradient estimation in machine learning. *arXiv* **2019**, arXiv:1906.10652.
32. Jang, E.; Gu, S.; Poole, B. Categorical reparameterization with gumbel-softmax. In Proceedings of the International Conference on Learning Representations (ICLR) 2017, Toulon, France, 24–26 April 2017.
33. Makhzani, A.; Shlens, J.; Jaitly, N.; Goodfellow, I.; Frey, B. Adversarial autoencoders. *arXiv* **2016**, arXiv:1511.05644.
34. Kingma, D.P.; Mohamed, S.; Jimenez Rezende, D.; Welling, M. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems 27*; Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2014; pp. 3581–3589.
35. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
36. Xiao, H.; Rasul, K.; Vollgraf, R. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. *arXiv* **2017**, arXiv:1708.07747.
37. Lake, B.M.; Salakhutdinov, R.; Tenenbaum, J.B. Human-level concept learning through probabilistic program induction. *Science* **2015**, *350*, 1332–1338. [CrossRef] [PubMed]
38. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; pp. 249–256.
39. Dauphin, Y.N.; Fan, A.; Auli, M.; Grangier, D. Language modeling with gated convolutional networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017.
40. Salakhutdinov, R.; Murray, I. On the quantitative analysis of deep belief networks. In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; pp. 872–879.