FITTING DATA NOISE IN VARIATIONAL AUTOENCODER

Anonymous authors
Paper under double-blind review

ABSTRACT

Why does variational autoencoder(VAE) suffer from bad reconstruction, and what influence the disentanglement of VAE? This paper tries to address those issues through a noise modelling perspective. On one fold, the paper proposes the adaptive noise learning algorithms of Gaussian noise and mixture Gaussian noise assumption which empirically contributes to a better reconstruction than original VAE noise assumptions. On other fold, several generating factor properties in the idealistic VAE case are discussed and several performance indicators regarding the disentanglement and generating influence are subsequently raised to evaluate the performance of VAE model and to supervise the used factors. Theoretical analysis is reflexed in the experiment results.

1 INTRODUCTION

From the theory of artificial intelligence perspectives, Variational AutoEncoder(VAE)s, raised by Kingma & Welling (2013) and Rezende et al. (2014), have been attracting much research attention in the recent years due to their powerful human-like abilities in extracting disentangled factors/representation(Bengio et al. (2013)) underlying data in a purely unsupervised manner and generating signals with abundant diversities in a "latent-factor-controllable" way. On the one hand, beyond most of the current machine learning regimes, VAEs are capitalized on its approaching a causal modelling and disentangled generating factor learning capability, which finely simulating human abilities, emphasized by Lake et al. (2016), of the knowledge transferring through shared causes/factors among different tasks/experiences. On the other hand, the generalization capability possessed by VAEs also well comply with the ideal mental imagery mechanism in memory and thinking.

Due to its strong capabilities on latent representation learning, signal reconstruction and new sample generation, VAE and its variants have been widely applied to wide range of applications, including disentangled representations learning of images(Higgins et al. (2016),Kulkarni et al. (2015), Mathieu et al. (2016)) and time series(Fabius & van Amersfoort (2014)), zero/one/few-shot learning(Rezende et al. (2016),Higgins et al. (2017b)) and transfer learning in reinforcement learning(Higgins et al. (2017a)), causal relationships modeling(Louizos et al. (2017)), pixel trajectory predicting(Walker et al. (2016)), joint multi-modal inference learning(Suzuki et al. (2016)), increasing diversity in imitation learning(Wang et al. (2017)), generation with memory(Li et al. (2016)) and etc.

However, the implementations of VAE always ignore the importance of noise on its reconstruction and representation learning ability and the applications of VAE has suffered from the blurred reconstructions/generations and the unstable disentangled quality (Higgins et al. (2016)). In particular, most of them assume a heuristic fixed noise that disables the model to flexible adapt to the real noise. These issues, as results, sometimes make VAE have to be an auxiliary part to only alleviate the generation shortcomings of generative adversarial network(GAN)(Larsen et al. (2015),Wang et al. (2017)).

From the perspective of VAE assumption, noise modelling places an indispensable part. Concretely, on one hand, the major factors are learnt and inferred and the noise enables the optimization. On the other hand, more importantly, since the whole learning procedure is through an unsupervised manner, the prior for major factors and noise modelling together form the core inductive bias of VAE.

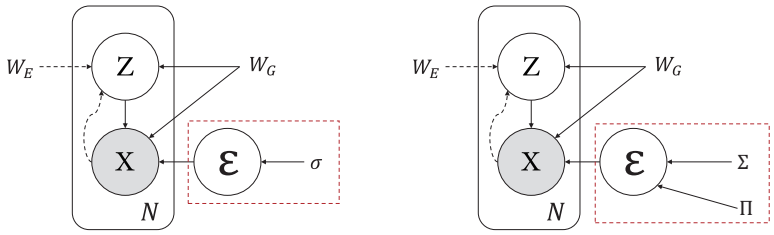


Figure 1: The directed graph model of noise modeling VAE. (a) is under gaussian noise assumption. (b) is under mixture of gaussian noise assumption. Solid lines denote the generative model $p_{model}(z; W_G)p_{model}(x|z; W_G, \sigma)$ in (a) and $p_{model}(z; W_G)p_{model}(x|z; W_G, \Sigma, \Pi)$ in (b). Dashed black lines denote the variational approximation $q(z|x; W_E)$ to the intractable posterior $p_{model}(z|x; W_G, \sigma)$ in (a) or $p_{model}(z|x; W_G, \Sigma, \Pi)$ in (b).

From the perspective of real implementation, different datasets may have drastically different noise and improper modeling for noise distribution inclines to lead to incorrect model hypothesis and tend to influence the distribution learning consequence. The superiority brought by proper consideration on noise modeling has been verified in various applications like denoising(Chen et al. (2017)), background substraction(Yong et al. (2017)), derain(Wei et al. (2017)) and medical image reconstruction and ect.

There are also lack of effective disentanglement performance metrics for VAE in real application. The traditional performance metric raised by Higgins et al. (2016) is intractable to be computed in real data and hard to provide direct feedback of the disentanglement of VAE in used currently.

To alleviate the aforementioned issues on VAE, in this study, we attempt to address the aforemention issues through a noise modelling perspective regarding VAE. This perspective provides us a reinterpretation that the generation process of VAE can be viewed as a noise adaption process. Since the derivation of the VAE is constructed based on the maximum likelihood principle, the supplemental parameters deduced by noise distribution amelioration can be directly embedded into and learnt through optimizing the objective of VAE and general end-to-end learning algorithms can be easily deduced. In this work, we use two modeling regime for fitting noise to data: Gaussian (with its variance adaptively learnt from data) and mixture of Gaussian. In such way, we expect our VAE embedded with noise modeling regime capable of better deliver the intrinsic generalization mechanism underlying data and achieve better performance in both representation and reconstruction. Furthermore, in order to guarantee a better disentanglement of representation, the auxiliary constraints (Higgins et al. (2016)) are introduced to the objectives. Further, the factors properties regarding the Gaussian-prior-VAE are discussed mathematically and indicators for quantitatively assessing the factor disentanglement ability, factor influence degree and inference mutual information raised. Beyond the previous metrics for performance assessment which could be hardly used in practical cases, such raised indicators can help easily quantify VAE model performance w or w/o noise modelling and auxiliary constraints in real data. By qualitative (in visualization) and quantitative (in terms of the presented performance indicators), we show that overall noise modelling w & w/o auxiliary constraints are superior to original VAE by experiments on datasets, including CelebA(Liu et al. (2015)), Extended Yale Face B(Georghiades et al. (2001),Lee et al. (2005)) and MNIST(Lcun et al. (1998)). The proposed metrics can also facilitate effective discovering most "influential" latent factors to help generating and traversal in the latent space.

In summary, the contribution of this paper can be mainly summarized as follows.

- We propose the VAE model embedded with noise fitting component, which is expected to better adapt practical noise configurations in real data. Two noise modeling regimes are considered to construct the noise learning VAE model, Gaussian (with variance learned from data) and mixture of Gaussian. Such amelioration facilitates the VAE capable of always reducing the artificial intervention due to more proper guiding of noise learning. Further, auxiliary constraints can be introduced to guarantee a better disentanglement.

- We propose multiple quantitative indicators for VAE as well as their estimation methods to supervise the degree of disentanglement, to quantify the mutual information of codes/factors and original signal regarding inference/encoder network and to determine the influential factors, and provide a bunch of theorems/definitions to illustrate the idealistic properties of gaussian factors VAE. Different from the previous metrics for assessing VAE performance, these indicators can be calculated by directly implementing our proposed algorithms on any given data.

- We substantiate the effectiveness of the proposed VAE-with-noise-modeling algorithm, as well as the proposed indicators on various datasets, and show the importance of such noise modeling consideration in both the reconstruction and disentanglement performance of VAE model as compared with the previous enumerated pre-specified noise VAEs. The mixture of gaussian(MoG) noise model with proper specified components number can further help achieve a more elaborate noise component decomposition. Also, the Gaussian and MoG noise assumption alleviate the blurry effect issue of the generated images generally encountered by previous VAE methods. The proposed indicators also facilitates an appropriate extraction on intrinsic latent factors underlying data.

The paper is organized as the following: The related work is briefly reviewed in Section 2. The proposed VAE model with noise modeling, together with its theoretical support and implementation algorithms, is introduced in Section 3. The proposed indicators for assessing the factor disentanglement ability, factor influence degree and inference mutual information, as well as their insightful theories, are given in Section 4. The experimental results are demonstrated in Section 5, and finally we provide conclusion and discussions on this work.

2 Related Work

VAE was proposed by Kingma & Welling (2013) and Rezende et al. (2014) to implement the efficient learning and inference in directed probabilistic models regarding continuous latent variables with intractable posterior distributions and in scalable datasets. They introduced a network inference/recoginition model to represent the approximate posterior distribution and utilized reparameterization trick for stochastic joint optimization of a variational lower bound containing the parameters of both the generative/decoder and inference/recoginition/encoder models. While they also designed a network for parameterizing the noise for Gaussian MLP decoder, the capability of noise rectifying based on data has not been specifically emphasized, just similar to most of the latter VAE applications, which easily specify a fixed Gaussian noise (with preset variance parameter) but more or less underestimate the role of noise learning.

After being raised, many VAE variations have been proposed to boost VAE's capabilities in generation quality and/or disentanglement of the learned representation. In these methods, multiple efforts were made by improving the generative and inference network structures. Typical works along this line include the convolution/de-convolution structure raised by Kulkarni et al. (2015) and ladder structure raised by Zhao et al. (2017)). Some other works advanced the mechanism under the VAE generation/inference processes. Typical works include the iterative attention generation/inference mechanism raised by Gregor et al. (2015), normalizing flow proposed by (Rezende & Mohamed (2015)) that enhanced the expressive ability of the approximate posterior and its variants (Kingma et al. (2016)).

Despite the improvement to the VAE itself, some other efforts were made by the ensemble between GAN with VAE. E.g., Larsen et al. (2015) unified GAN and VAE to obtain a better reconstruction and a high-level abstracts visual features embedding. Mathieu et al. (2016) also unified GAN and VAE but put emphasis on disentangling factors of variation. GANs without auxiliary design would learn the data distribution disregarding its noise level though suffer from unstable training and mode collapsing(Salimans et al. (2016)) while VAEs would assume a decomposition of the noise and oracle clean datapoint regarding the noise data with an auxiliary prior on the distribution regarding the factors. If the wrong specified noise in VAE is combined into GAN which inclines to conduce unexpected images with corruptions, it's reasonable that the ensemble model may achieve a better hypothesis of true data distribution.

Besides, many efforts were made by regularization on the factor distribution or factor generating effect. E.g., Makhzani et al. (2015) introduced an adversarial loss into the latent space of the autoencoder which in idealistic case could learn any kind factor/lantent distribution including those contributing to the disentangled factors/representation. InfoGAN, raised by Chen et al. (2016), introduced the infomax principle to GAN by adding an auxiliary mutual information regularization which enabled the inference of GANs and led to a better disentangled representation as well.

However, most of these current VAE models need to pre-specify the noise parameter before VAE training, which inclines to make them perform not stably well especially under complex noises especially different from the subjective pre-specified noise configuration. This issue thus inspires us to introduce noise learning component into the VAE model (as depicted in Fig. 1 to ameliorate the model better fitting the real data and further boost the VAE performance in both latent factor representation and signal reconstruction.

Recently, there is a new VAE variation is proposed by Higgins et al. (2016) who introduced the β-VAE framework which enhanced the constraints regarding the KL-divergence of the posterior and prior distribution of VAE and showd a novel disentanglement performance. This method has obtained a better performance as compared with conventional VAE methods, especially on its flexible tuning a compromising a parameter beta between the KL-divergence term and the likelihood term (the variational lower bound). We thus select this VAE model to embed our noise modeling regime to improve its capability of adapting real noise in data. It should be noted that β-VAE still assumes a Gaussian noise with pre-specified variance parameter, and the noise modeling regime is thus expected to further improve its learning capability and ameliorate its performance to be more stable and robust to real data noise.

In the other perspective of the decomposition of the noise and oracle clean data, deep denoising models, including denoising autoencoder(Goodfellow et al. (2016)), stacked denoising autoencoders(Vincent et al. (2010)), denosing variational encoder(Im et al. (2017)) etc., have also shown similar ability to the proposed VAE model with noise modeling. However, those models are trained by using paired/supervised data (both corrupted and clean data) while our model just learn in a purely unsupervised manner without supervision regarding the noise/or oracle data. Just similar to traditional VAE, such implementation paradigm is more similar to human learning process, and can be better generalized into real applications lacking supervision knowledge.

3 VAE WITH NOISE MODELING

In traditional VAE applications, the noise is generally fixed as a Gaussian with fixed variance. In this section, we further extend such noise-specification implementation as a automatic noise adapting regime. Specifically, we take the Gaussian variance parameter as an optimization variable and integrate it into the VAE model to make the noise fitted by data, and furthermore, we ameliorate the noise as mixture of Gaussian[1] to further enhance its noise modeling capability. The β-VAE is employed to integrate such noise modeling mechanism.

3.1 INTERPRET THE ORACLE GENERATION OF VAE IN NOISE MODELING PERSPECTIVE

When modelling the real-valued generation process, VAEs often assume the conditional distribution to be

$$p_{model}(x|z) = \mathcal{N}(x|G(z), \sigma^2 I_d),$$

where $x \in \mathbb{R}^D$ is the random vector variable corresponding to input data and $z \in \mathbb{R}^H$ represents the latent factors for implicitly generating the data. G is the generating/decoder function parameterized by neural networks and σ^2 is the variance parameter of the Gaussian distribution.

Such VAE model can be equivalently interpreted in perspective of noise modeling. The data are formulated by the ideal generation $G(z)$ (clean data, where z follows a prior $p(z) = \mathcal{N}(z; 0, I_H)$) corrupted by an additional element-wise Gaussian noise ε with variance σ^2. mathematics, this

[1]The mixture of Gaussian is of a strong universal approximation capability to general probability distributions(**?**).

saying can be expressed as the following[2]

$$x = G(z) + \varepsilon. \tag{1}$$

In VAE setting, the approximate inference[3] method is applied to maximizing the variational lower bound of $p_{model}(x) = \int p_{model}(x|z)p(z)dz$,

$$\mathcal{L}(q) = \mathop{\mathbb{E}}_{z\sim q(z|x)} \log p_{model}(x|z) - D_{KL}(q(z|x)||p(z)) \leq \log p_{model}(x). \tag{2}$$

In most practical applications of VAE, the noise level, σ^2, is generally heuristically pre-specified(Makhzani et al. (2015),Walker et al. (2016),Higgins et al. (2017a)). Since the objective of the log-likelihood needs to negotiate with the KL-divergence term in the optimization process for VAE, if the noise level cannot guarantee to comply with the real noise level underlying data, the properness of the above VAE model could not be satisfied, which naturally tends to lead to the instable performance of the method in practice (i.e. suffer from Severely Wrong Model Assumption). Such an issue has been observed in β-VAE(Higgins et al. (2016)), which enforces a tunable parameter β to more or less deliver noise variation knowledge of data in the VAE model[4], while such β still needs to be manually pre-specified before VAE optimization process, which is generally still a challenging task for real data.

3.2 Noise Modelling With Auxiliary Constraint

The entangled representation can be caused by the over-large of searching space of $q(z|x)$. The posterior distribution searching space determined by the neural structure could always be two large in VAE application. If the learned $q(z) = \int q(z|x)p_{data}(x)dx$ doesn't factorize, then the VAE model in the perspective of inference network just tends to learn the entangle representation. Actually, in the VAE model, what we want is to search in the space that $q(z)$ is possibly similar to $p(z)$ [5]. By implementing this ideal, we add auxiliary upper bound $\mathbb{E}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||p(z))$ (detailed in Theorem 5) of $D_{KL}(q(z)||p(z))$ to the original objective. This equivalently leads to the approach of β-VAE raised by Higgins et al. (2016),

$$\sup_{q(z|x)} \mathop{\mathbb{E}}_{x\sim p_{data}(x)} \mathcal{L}(q(z|x)) - (\beta - 1)D_{KL}(q(z|x)||p(z))$$

[2]This setting is quite common used in many other fields, and normally leads to L^2 loss during the optimization regarding the factor if there is no other term have to be negotiated with negative-log likelihood

$$\arg\inf_G -\log p(x|z) = \frac{\|x - G(z)\|_2^2}{2\sigma^2} + \frac{1}{2}\log(2\pi)^m|\sigma^2 I_m| \longleftrightarrow \arg\inf_G \|x - G(z)\|_2^2.$$

[3]The approximate posterior distribution $q(z|x)$ follows $\mathcal{N}(En(x); \Sigma_{z|x}(x))$ where En denotes the Encoder and $\Sigma_{z|x}$ represents the covariance matrix parameterized by the neural network. The reparameterization trick, $z|x = En(x) + \Sigma_{z|x}^{1/2}(x)e$ where $e \sim \mathcal{N}(0, I_H)$, enables the calculation of coefficient gradient of the first part of VAE through sampling, that is

$$\mathop{\mathbb{E}}_{z\sim q(z|x)} \log p_{model}(x|z) = \mathop{\mathbb{E}}_{e\sim\mathcal{N}(0,1)} \log p_{model}(x|En(x)+\Sigma_{z|x}^{1/2}(x)e) \approx \frac{1}{M}\sum_{m=1}^{M} \log p_{model}(x|En(x)+\Sigma_{z|x}^{1/2}(x)e^m).$$

[4] Objective of σ^2 pre-specified VAE:

$$\mathop{\mathbb{E}}_{z\sim q(z|x)} \frac{\|x - G(z)\|_2^2}{2\sigma^2} - D_{KL}(q(z|x)||p_{model}(z)).$$

Objective of (σ^2 pre-specified as σ_{pre}^2) β-VAE:

$$\mathop{\mathbb{E}}_{z\sim q(z|x)} \|x - G(z)\|_2^2 - 2\beta\sigma_{pre}^2 D_{KL}(q(z|x)||p_{model}(z)),$$

where we call $\beta\sigma_{pre}^2$ the normalized variance.

[5] Jensen Shannon Divergence and other integral probability metric which can be good choice and directly be optimized through a adversarial format (Makhzani et al. (2015)) as well. However, in practice, we were defeated by the unstability of training GAN-like model.

$$= \mathop{\mathbb{E}}_{x\sim p_{data}(x)} \mathop{\mathbb{E}}_{z\sim q(z|x)} \log p_{model}(x|z) - \beta D_{KL}(q(z|x)||p(z)) \tag{3}$$

where $\beta > 1$.

3.3 Integrating Noise Fitting and VAE Parameter Learning

We can easily integrate the noise parameters into the VAE objective to make the noise as a learnable part in VAE to compensate a better performance on both representation and reconstruction of VAE.

3.3.1 Gaussian Case

The VAE objective (i.e., the variational lower bound) can be treated as function of the noise variance parameters σ^2 and networks parameters W(which parameterizes the factor W_G, the encoder and the posterior distribution W_E),

$$\mathcal{L}(W, \sigma, x^m) = \mathop{\mathbb{E}}_{z\sim q(z|x^m;W_E)} \log p_{model}(x^m|z;\sigma,W_G) - D_{KL}(q(z|x^m;W_E)||p_{model}(z)). \tag{4}$$

Here the SGVB estimator in (Kingma & Welling (2013)), $\tilde{\mathcal{L}}^B(W,\sigma,x^m) = [\frac{1}{L}\sum_{l=1}^{L} \log p_{model}(x^m|z^l;\sigma,W_G)] - D_{KL}(q(z|x^m;W_E)||p_{model}(z))$ is used. Note that the noise variance σ is also taken as an optimization variable in the model, making the model capable of better adapting noise variation of data in practical cases in a totally automatic way, instead of a manually set manner.

Given multiple data points from a dataset X, we can construct an estimator of the mean marginal likelihood lower bound of the full dataset, based on minibatches

$$\tilde{\mathcal{L}}^M(W,\sigma,X^M) = \frac{1}{M}\sum_{m=1}^{M} \tilde{\mathcal{L}}^B(W,\sigma,x^m), \tag{5}$$

where the minibatch $X^M = \{x^m\}_{m=1}^M$ is a randomly drawn sample set of M datapoints from the full dataset X. Such a lower bound also constitutes an important indicator for model evidence in latter experiment. We call it the empirical variational lower bound (EVLB) in the following.

Note that $\tilde{\mathcal{L}}^B(W,\sigma,x^m) \simeq \mathcal{L}(W,\sigma,x^m)$ and we can deduce that

$$\tilde{\mathcal{L}}^M(W,\sigma,X^M) \simeq \mathop{\mathbb{E}}_{x\sim p_{data}(x)} \mathcal{L}(W,\sigma,x)$$

$$\leq \mathop{\mathbb{E}}_{x\sim p_{data}(x)} \log p_{model}(x;W_G,\sigma) \leq \mathop{\mathbb{E}}_{x\sim p_{data}(x)} \log p_{data}(x). \tag{6}$$

The last inequality holds due to $D_{KL}(p_{data}(x)||p_{model}(x;W_G,\sigma)) \geq 0$.

The alternative optimization strategy can be readily utilized to design the algorithm for solving the model by iteratively updating the noise parameter and the network ones. During the optimization process, the objective can be monotonically increasing, and thus the algorithm can be guaranteed to be convergent.

The algorithm is summarized as follows:

Optimization for W: gradient method for W in regard to $\tilde{\mathcal{L}}^M(W,\sigma,X^M)$.

Optimization for σ: $\sigma^2 = \frac{\sum_{l=1}^{L}\sum_{m=1}^{M}\|x^m - G(z^{m,l})\|_2^2}{dML}$. (Close form solution in regard to $\tilde{\mathcal{L}}^M(W,\sigma,X^M)$.)

Direct gradient method to the transformed variable $log\_simga = \log\sigma \in \mathbb{R}$ can be implemented to lift the lower bound $\tilde{\mathcal{L}}^M$ as a result to increase the likelihood as well.

3.3.2 Mixture of Gaussian Case

The noise ε in Eq. (1) in real situation might be more complex than a simple Gaussian, like that existed in real photographs(Plotz & Roth (2017)). We thus try to further ameliorate the noise setting

as a mixture of gaussian(MoG) noise. Such noise modeling strategy has been widely verified to be effective in applications, like matrix factorization (Meng & Torre (2014)) and robust principal component analysis (Zhao et al. (2014)). That is, we assume that

$$\varepsilon \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(0, \sigma_k^2). \tag{7}$$

Let $c_d \in \{0,1\}^K$ be the latent indicator random one-hot variable, $\sum_{k=1}^{K} c_{dk} = 1$, for the MoG-noise component of pixel indexed by d. Let $\Pi = [\pi_1, \cdots, \pi_K]$ and $\Sigma = [\sigma_1^2, \cdots, \sigma_K^2]$ be the ratio and variance of each component, respectively. Let $W_N = [\Pi, \Sigma]$. The conditional joint distribution turns to be

$$p_{model}(c_d, x_d|z, W_N, W_G) = \prod_{k=1}^{K} \pi_k^{c_{dk}} \mathcal{N}(x_d|G(z)_d, \sigma_k)^{c_{dk}}. \tag{8}$$

The posterior distribution $q(z, c|x)$ can be factorized as $q(z|x)q(c|x, z)$, where $q(z|x)$ will be direct learnt and the alternative of $q(c|x, z)$, $q(c|x, e)$ will be set to the last step $p_{model}(c|x, e)$ in regard to EM procedure. The lower bound of $\log p_{model}(x)$ is then reformulated as follows:

$$\mathcal{L}(q(z, c|x)) = \mathop{\mathbb{E}}_{z\sim q(\tilde{z}|x)} \mathop{\mathbb{E}}_{c\sim q(\tilde{c}|x,\tilde{z}=z)} \log p_{model}(x, c|z) + H(q(c|x, z)) - D_{KL}(q(z|x)||p_{model}(z)). \tag{9}$$

Similar to the Gaussian case, the reparamerization trick is implemented,

$$\begin{aligned} &\mathcal{L}(q(c|x, e), W_N, W_G, W_E, x^m) \\ &= \mathop{\mathbb{E}}_{e\sim\mathcal{N}(0,1)} \mathop{\mathbb{E}}_{c\sim q(\tilde{c}|x,e)} \log p_{model}(x, c|\tilde{z}) + \mathcal{H}(q(c|x, \tilde{z})) - D_{KL}(q(z|x)||p_{model}(z)), \end{aligned} \tag{10}$$

where $\tilde{z} = En(x) + {\Sigma_{z|x}}^{1/2}(x)e$.

By utilizing the SGVB estimator, we get,

$$\begin{aligned} \tilde{\mathcal{L}}^B(q(c|x, e), W_N, W_G, W_E, x^m) = [\frac{1}{L} \sum_{l=1}^{L} \mathop{\mathbb{E}}_{c\sim q(\tilde{c}|x^m,e^{(l)})} \log p_{model}(x^m, c|z^{m,l}) \\ + \mathcal{H}(q(c|x^m, e^{(l)}))] - D_{KL}(q(z|x^m)||p_{model}(z)). \end{aligned} \tag{11}$$

Given an input dataset X, we can then construct an estimator to the mean marginal likelihood lower bound of the full dataset, based on minibatches, as follows:

$$\tilde{\mathcal{L}}^M(q(c|x, e), W_N, W_G, W_E, X^M) = \frac{1}{M} \sum_{i=1}^{M} \tilde{\mathcal{L}}^B(q(c|x, e), W_N, W_G, W_E, x^m), \tag{12}$$

where $z^{m,l} = En(x^m) + {\Sigma_{z|x}}^{1/2}(x^m)e^{(l)}$ and the minibatch $X^M = \{x^m\}_{i=1}^{M}$is a randomly drawn sample of M datapoints from the full dataset X.

Then let

$$p_{model}^{old}(c_d, x_d|z_{old}, W_N^{old}, W_G^{old}) = \prod_{k=1}^{K} {\pi_k^{old}}^{c_{dk}} \mathcal{N}(x_d|G^{old}(z_{old})_d, \sigma_k^{old})^{c_{dk}}, \tag{13}$$

where $z_{old} = En^{old}(x) + {\Sigma_{z|x}^{old}}^{1/2}(x)e$, and we can get

$$p_{model}^{old}(c_d|x, z_{old}, W_N^{old}, W_G^{old}) = \frac{p_{model}^{old}(c_d, x_d|z_{old}, W_N^{old}, W_G^{old})}{\sum\limits_{c_d} p_{model}^{old}(c_d, x_d|z_{old}, W_N^{old}, W_G^{old})}. \tag{14}$$

The EM algorithm can be naturally employed to solve the model. The implementation steps are listed as follows:

Step 1. **Expectation Step**.

Set $q(c|x^m, e^{(l)}) = p^{old}_{model}(c|x^m, En^{old}(x^m) + \Sigma^{old}(x^m)e^{(l)})\ i = 1, \cdots, m, l = 1, \cdots, L.$

Calculate the expectation of the latent variable c:

$$E(c_{dmlk}) = \gamma_{dmlk} = \frac{\pi_k \mathcal{N}(x^m_d | G(z^{m,l})_d, \sigma^2_k)}{\sum_{l=1}^{L}\sum_{m=1}^{M} \pi_k \mathcal{N}(x^m_d | G(z^{m,l})_d, \sigma^2_k)}, \tag{15}$$

where $z^{m,l} : z^{m,l}_{old} = En^{old}(x^m) + \Sigma^{old}(x^m)e^{(l)}$.

The Objective in Maximization Step is obtained as the following,

$$\tilde{\mathcal{L}}^M(q(c|x,e), W_N, W_G, W_E, x^m) = \frac{1}{M}\sum_{i=1}^{M} -D_{KL}(q(z|x^m)||p_{model}(z))$$

$$+\frac{1}{L}\sum_{l=1}^{L}\mathcal{H}(q^{old}(c|x^m, e^{(l)})) + \sum_{k=1}^{K}\sum_{d=1}^{D}\gamma_{dmlk}[\frac{(x^m_d - G(z^{m,l})_d)^2}{2\sigma^2_k} + \frac{1}{2}\log(2\pi)\sigma^2_k + \ln\pi_k]. \tag{16}$$

Step 2. **Maximization Step:**

Fix: $q(c|x,e)$ determined in the Expectation Step.

$$\frac{1}{M}\sum_{i=1}^{M} -D_{KL}(q(z|x^m)||p_{model}(z)) + \frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K}\sum_{d=1}^{D}\gamma_{dmlk}[\frac{(x^m_d - G(z^{m,l})_d)^2}{2\sigma^2_k} + \frac{1}{2}\log(2\pi)\sigma^2_k + \ln\pi_k]. \tag{17}$$

Update $[W_N]$,$[W_G, W_E]$ by alternative optization strategy.

Update Π, Σ: note here $z^{m,l} : z^{m,l}_{old} = En^{old}(x^m) + \Sigma^{old}(x^m)e^{(l)}$, and we can easily get the closed-form updating formula for these parameters:

$$N_k = \sum_{d,m,l}\gamma_{dmlk} \qquad \pi_k = \frac{N_k}{\sum_{k=1}^{K} N_k} \qquad \sigma^2_k = \frac{1}{N_k}\sum_{d,m,l}\gamma_{dmlk}(x^m_d - G(z^{m,l}_{old})_d)^2. \tag{18}$$

Update W_G, W_E: gradient methods with respect to W_G, W_E. Note here $z^{m,l} = En(x^m) + \Sigma(x^m)e^{(l)}$.

The algorithm can then be summarized as follows:

1. Initialize the coefficient of W_G,$W_E = [Encoder, \Sigma_{z|x}]$ and the coefficient of noise ε: Π,Σ.
2. Sample e from $\mathcal{N}(0, I_H)$ to obtain $e_1, \cdots, e_M$ [One for each element sample in the mini batch in the next step (L here is set to 1)].
3. Sample a mini batch X^M from $p_{data}(x)$.
4. Implement EM algorithms as aforementioned (approximate inference for $q(c, z|x)$):
   Expectation: calculate γ_{dmk}.
   Maximization: update W_N , Update W_G,W_E with gradient methods.
5. Goto 3: Until Trigger End-Criterion.

3.3.3 Network Parameterized Gaussian Case

Here we want to give a short discussion on the difference of the proposed VAE model with noise modeling with that raised by Kingma & Welling (2013) , in which they model the noise as the following parameterized network structure:

$$x \sim \mathcal{N}(G(z), \sigma^2(z))$$

$$G(z) = W_u h(z) + b_u$$
$$\log \sigma^2(z) = W_\sigma h(z) + b_\sigma, \tag{19}$$

where h represents the mapping induced by the previous network layers; $\sigma^2(z)$ represents the diagonal of diagonal covariance matrix; $\{W_u, W_\sigma, b_u, b_\sigma\}$ are the weights and biases of the last layer of network.

It can be observed that this model assumes that each pixel noise indexed by d has its own level determined by $b_{\sigma\, d}$ and is also influenced by the deterministic part $W_\sigma h(z)$. When the noise level is shared among all data points and not influenced by the deterministic part, then it degenerates to the Gaussian assumption; if the noise has several discrete level, it then tends to degenerate to MoG assumption on noise. However, on the one hand, the assumption used in the model inclines to make the optimization for the model difficult due to the numerical instability caused even by one zero-variance residual pixel. This is possibly why most applications on VAE have not employed such noise assumption while prefer to fix and manually set a noise level before VAE training. On the other hand, the over-parameterized noise could significantly increase the difficulty to find a better deterministic part since the model might be inclined to fit the noise hypothesis rather than learn a good G. These limitations have been empirically verified by all our experiments and can be observed in the experimental results as listed in Section 5.

4 Generating Factor Properties and Performance Indictors

In order to evaluate the performance of our model, we propose multiple new indictors. All these indicators can be approximately calculated from input data, which ameliorates the issue that the previous performance metric cannot be easily computed from real data (as discussed in Section 4.2.1). Roughly speaking, we try to show that idealistic VAE model[6] is hard to properly learn excess/extra factors by information conservation theorem(Theorem 1); factors that are possible to be learnt by idealistic VAE tend to form an equivalence class under the orthogonal transformation by Gaussian factor equivalence theorem(Theorems 2 and 3) and therefore even idealistic VAE cannot learn "semantic disentangle" representation. Subsequently, multiple meaningful performance indicators are raised: the estimation for $D_{KL}(q(z)||p(z))$ is used for quantifying the disentanglements, and the estimation of $\mathcal{I}_{encoder}(x; z_h)$, used for quantifying the influential("used") factors.

4.1 Information Conservation

Whether VAE can learn the real factors or just some fantasies that model itself makes up is an important issue for a generative model. We try to address this issue by disregarding the training procedure and direct considering the idealistic VAE's behavior through the following theorem.

Theorem 1 (Information Conservation)**.** *Suppose that $z = (z_1, \cdots, z_H)$ and $y = (y_1, \cdots, y_P)$ are sets of H and P ($H \neq P$) independent unit Gaussian random variables, respectively, then these two sets of random variables can not be the generating factor of each other. That is, there are no continuous functions $f : \mathbb{R}^H \to \mathbb{R}^P$ and $g : \mathbb{R}^P \to \mathbb{R}^H$ such that*

$$z = g(y) \quad and \quad y = f(z).$$

The principle of the theorem is visually illustrated in Fig. 2.This theorem roughly demonstrates that the number of the learnt "used" factors of VAE can be the same as the true factors number under some assumptions such as the learnt $q(z)$ should equal $p(z)$ and decode/encode process is continuous and reversible. Empirically, only a small amount of unit gaussian variables regarding the factors of well disentangled VAE have been used in practical VAE application and this theorem helps provide an interpretation to explain this phenomenon. Suppose that the observed data, denoted by random variable x, is generated by y (with P independent unit gaussian random variables) with a homeomorphism mapping $x = \phi(y)$. VAE will be forced to learn the factor z (with H independent unit gaussian random variables) that generates the x with a homeomorphism mapping $x = \psi(z)$. It yields $z = \psi^{-1} \circ \phi(y)$ and $y = \phi^{-1} \circ \psi(z)$. Then according to the information conservation theorem, it must hold that $H = P$.

[6] An idealistic VAE model means that it can perfectly encode the signal into "used" factors and perfectly decode the "used" factors to original input signal and the factors follows i.i.d unit gaussian distribution.

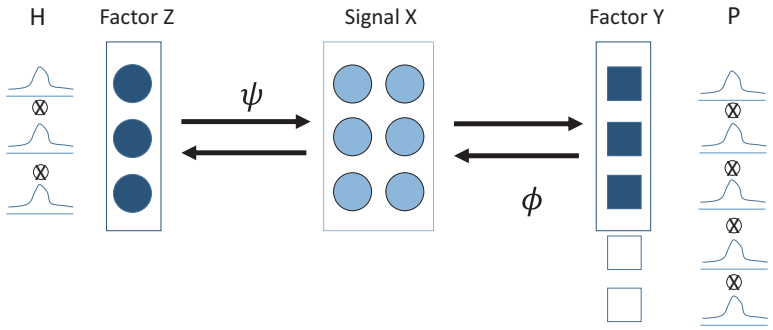


Figure 2: The illustration of the information conservation theorem

4.1.1 Efficient Representation and Clarification on Disentanglement

According to the information conservation theorem, the independent unit gaussian distribution assumption regarding the factor of the model facilitates the model incline to achieve most efficient coding.(i.e., the number of functional latent factors extracted from the model should be same as that of the intrinsic latent factors underlying the model). That is, no auxiliary factor tend to be learnt, though the number latent of factors sometimes is pre-specified larger than that in the idealistic VAE setting.

Here, in order to avoid the ambiguity of the terminology of disentanglement, we make the following clarification.

- The disentanglement of the learnt representation/factors in this literature refers to two parts depicted in Theorem 1:
  - the factors are closer to be independent with each other,
  - the factors incline to be able to generate the oracle signal and to be inferred perfectly from the oracle signal through a continuous procedure/mapping.
- The "disentanglement" refers to the closeness of the learnt factors to the pre-specified independent factors/concpets that can generate the oracle signal and be perfect inferred through a continuous procedure/mapping such as the independent semantic/visual factors.

Therefore, the estimation for $D_{KL}(q(z)||p(z))$ that reflects the divergence of the learnt factor distribution and the i.i.d. unit gaussian prior can be good a indicator to supervise the independence of the factors can be served to quantitatively assess the disentanglement of each extracted factor.

The "disentanglement" will be shown that is hard to be obtained in a unsupervised manner. Concretely, even in the idealistic cases, the extracted factors tend to possess the intrinsic number of latent factors of the model, while there are still possibly large variations of these factors due to it can be obtained in as proved in the next subsection.

4.2 Generator Equivalence

Theorem 2 (Gaussian Factor Equivalence)**.** *Suppose that $z = (z_1, \cdots, z_H)$ is a set of H independent unit gaussian random variables. Let $Q \in \mathbb{R}^{H\times H}$ be an orthogonal matrix and then $y = Qz$ is also a set H independent unit gaussian random variables. Besides, z and y can generate each other through a linear homeomorphism mapping.*

This theorem implies that there are a class of unit Gaussian random variables which can generate each other and have equivalent conservation information, as indicated by the following theorem.

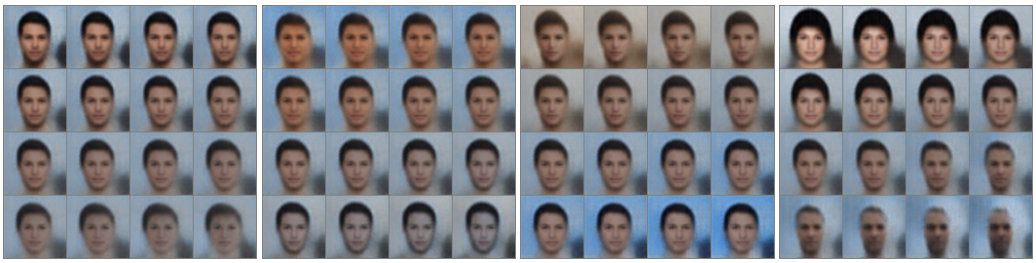
Theorem 3 (Linear Gaussian Factor Equivalence Class)**.**

$$[z] = \{y|y = Qz, \quad Q \in \mathbb{R}^{H\times H} \textit{be the orthogonal mapping.}\}$$

Then $\forall y \in [z]$, y is a set of H independent unit gaussian random variables and can generate z through an linear homeomorphism mapping.

The theorem clarifies that if VAEs have an linear matrix multiplication freedom degree of learning the factors, then the factors in the equivalence class can all be possibly learnt.

The empirically results tally with the above analysis(see Fig. 3). Suppose the visual semantic concepts can be viewed as a set of independent Gaussian variables ($z = (z_{rotation}, z_{gender}, z_{with-glass}, \cdots)^T$) which are desired to be captured and learnt by VAEs, while the model is also possible to learn the independent factor set $y = (y_1, \cdots)^T = Qz$ in the equivalence class $[z]$. This explains why changing one factor like y_1 always empirically results in change in multiple visual concepts.



(a) (b) (c) (d)

Figure 3: One-shot traversal & generating factor equivalence class demonstration. The images are generated by MoG-2 β(=40) VAE trained on CelebA. The seed image is obtained out from the datasets. Each block represents the traversal of the generating factor from $[-3+z_{seed_h}, +3+z_{seed_h}]$. (a) corresponds to face color white-yellow & female-male change. (b) corresponds to face color white to yellow change. (c) corresponds to background yellow to blue change. (d) corresponds to hair color white to black & face width change. It can be seen that changing one factor results in multiple semantic factor change in a comprehensible manner which reflexed analysis regarding generating factor equivalence.

This perspective also suggests that it's actually hard to obtain the disentangled representation that exactly "one-to-one" corresponds to the "independent semantic representation" even though they are in the same equivalence class. As a result, the idealistic VAE model just tend to learn the "entangle representation" if we do preset a "oracle generating factor" belonging to the equivalent class.

However, though those conclusions might be upsetting, it seems not be biology impossible. Many biological evidences have proved that actually a neuron in the brain of animals could combinationally possesses several functional capabilities. E.g., Aronov et al. (2017) found that some neurons in rat's hippocampus involved in representing sound frequencies also were involved in spatial representation after training rats by a tasks that required them to use a joystick to manipulate sound in frequency continuously. We thus expect that even with such "entangling mechanism" the extracted factors by VAE could also possess rational representation capabilities and be finely interpreted.

If we assume that the most of the visual concept/factors follow the condition regarding the "disentanglement", it is rational to qualitatively measure the interpretability of extracted latent factors to infer the disentanglement. Besides, both the biological and previous empirical evidences of VAE applications(Higgins et al. (2016),Higgins et al. (2017b),Larsen et al. (2015),Mathieu et al. (2016)) have shown that such extracted factors located in the equivalent class can also finely reveal the interpretable representations underlying data.

4.2.1 Deficiency of the Existing Disentanglement Metric

Higgins et al. (2016) proposed an "simulated factor" based disentanglement metric on the simulation datasets. However, this metric could be hardly calculated in the real datasets to provide direct feedback of the disentanglement of the model since it needs to pre-know the generating factors of the VAE model by default, which yet are generally hardly to know in practice. Besides, according to Gaussian generator equivalence theorem(Theorem 2) that even idealistic VAE will still learn the factors in the equivalence class, their metric can suffer severely instability to evaluate the VAE in different trials (detailed in Appendix 7.2).

4.3 INFORMATION CHANNEL

The mutual information[7] regarding the factors learned by the inference/encoder network and the signal x can be a good quantity for evaluating the generating influence[8]. That is,

$$\mathcal{I}_{encoder}(x;z) = \mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||q(z)). \tag{20}$$

In order to understand and estimate which factor of the VAE was learnt and influenced the generating process, $\mathcal{I}_{encoder}(x;z_h)$ can be taken as a rational indicator[9]. If we assume that $z_1, z_2, \cdots, z_H$ is conditional independent given x[10], it can yield a useful result as the following.

Theorem 4 (Separation of the Mutual Information)**.** *Suppose $z_1, \cdots, z_H$ be independent unit gaussian distribution, and $z_1, z_2, \cdots, z_H$ be conditional independent given x. Then*

$$\mathcal{I}(z_1, \cdots, z_H; x) = \sum_{h=1}^{H} \mathcal{I}(z_h; x). \tag{21}$$

This theorem suggests that if the learnt $q(z)$ can factorize and the $q(z|x)$ can factorize, then the consideration of each $\mathcal{I}(z_h;x)$ won't be excess or lose information.

4.4 INDICATORS

In order to quantify the disentanglement performance as well as the $\mathcal{I}_{encoder}(x;z)$. We assume that $q^*(z)$ is a factorized zero mean gaussian estimation for $q(z)$. We first propose the following relevant theorem and then provide the indicators.

Theorem 5. *The terminology follows the aforemention definitions and if the involved KL-divergence and mutual information is well defined, then*

$$\mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||p(z)) = \mathcal{I}_{encoder}(x;z) + D_{KL}(q(z)||p(z)). \tag{22}$$

The theorem demonstrates that the second term in variation lower bound in Eq. (3.2) capable of controlling both the mutual information of x and z induced by the encoder network as well as the similarity of the learnt $q(z)$ and the prior $p(z)$(disentanglement performance). We can then list the indicators for assessing latent factor disentanglement:

Definition 1 (Estimation for $\mathbb{E}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||p(z))$)**.**

$$\tilde{D}_{KL}(q(z|x)||p(z)) = \frac{1}{M}\sum_{m=1}^{M} D_{KL}(q(z|x_m)||p(z)). \tag{23}$$

Definition 2 (Estimation for $I_{encoder}(x;z)$)**.**

$$\tilde{I}_{encoder}(x;z) = \frac{1}{M}\sum_{m=1}^{M} D_{KL}(q(z|x_m)||q^*(z)). \tag{24}$$

Definition 3 (Estimation for $I_{encoder}(x;z_h)$ which quantifies the influence of each factor)**.**

$$\tilde{I}_{encoder}(x;z_h) = \frac{1}{M}\sum_{m=1}^{M} D_{KL}(q(z_h|x_m)||q^*(z_h)). \tag{25}$$

[7] This term can be seen as a lower bound of the channel capacity (defined in Cover & Thomas (2012)) of the inference/encoder network

[8] Maximizing the mutual information of the encoder networks can also be viewed as the original objective of the autoencoder (Vincent et al. (2010)).

[9] If $\mathcal{I}_{encoder}(x;z_h) = 0$, it yields x and z_h are independent with each other. The bigger $\mathcal{I}_{encoder}(x;z_h)$, the more information z_h conveys regarding x.

[10] A special case is that z_i will be either determined by x or be the unit gaussian distribution which is independent of x. It follows the real implementation assumption that $\Sigma_{z|x}(x) = diag(\sigma_{z1}(x), \cdots, \sigma_{zH}(x))$.

Definition 4 (Estimation for $D_{KL}(q(z)||p(z))$)**.**

$$\tilde{D}_{KL}(q(z)||p(z)) = \tilde{D}_{KL}(q(z|x)||p(z)) - \tilde{I}_{encoder}(x; z). \tag{26}$$

Note that the above indicators 2-4 need the value of $q^*(z)$, we then introduce how to calculate this term based on Theorem 5. Through the minimization equivalence, we know taht

$$\min_Q \mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||Q(z)) \Leftrightarrow \min_Q D_{KL}(q(z)||Q(z))dz, \tag{27}$$

the $q*(z)$ can then be obtained from solving the following optimization problem which can be calculated by gradient method.

$$q^*(z) = \arg\min_Q \frac{1}{M}\sum_{m=1}^{M} D_{KL}(q(z|x_m)||Q(z)). \tag{28}$$

5 EXPERIMENT

In this section, we will show experimental results to both quantitatively demonstrate the superiority of a VAE model with embedded noise modeling component as compared with that without this part, and qualitatively show the better reconstruction capability and meaningful-latent-factor-extraction capability of the ameliorated VAE model with noise modeling. The functional effects of the proposed indicators can also be verified.

The comparison method is employed as the recently proposed β-VAE(Higgins et al. (2016)), which has been proved to have a good reconstruction and representation capabilities as compared with traditional VAE methods due to its involvement of a tunable compromising parameter β between the likelihood and KL-divergence terms in VAE objective. Besides, the network parameterized Gaussian noise learning VAE model is also considered for comparison in Extended Yale B dataset. We will show the performance amelioration taken by integrating noise modeling component in these methods.

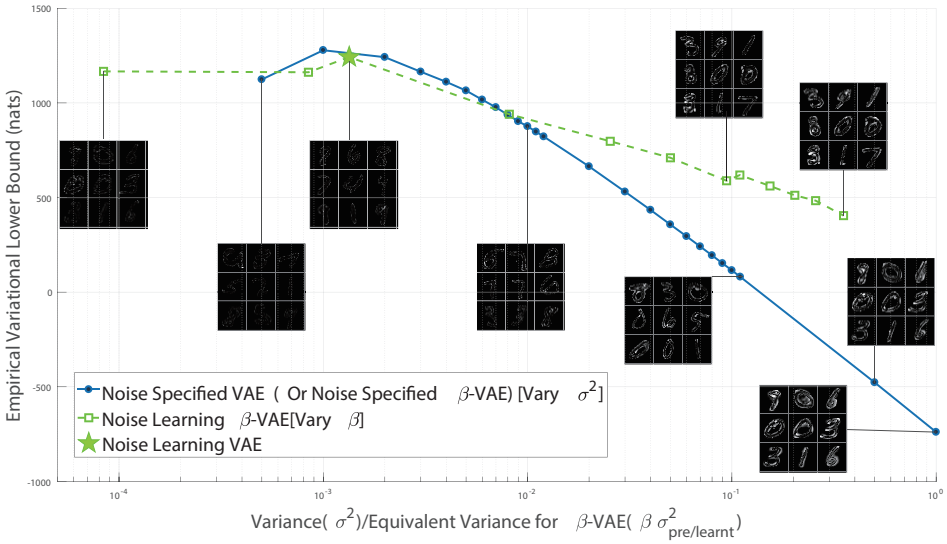
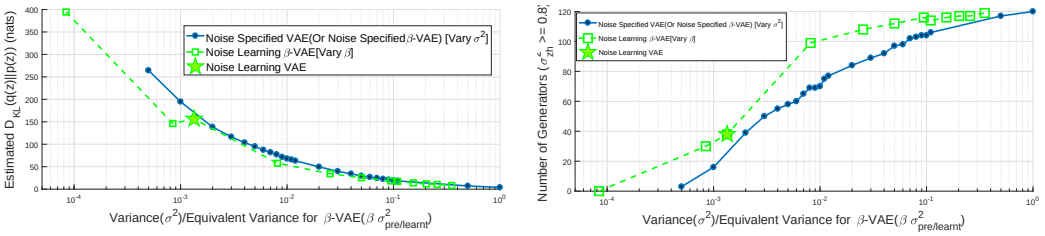


Figure 4: Noise specification influence the learned hypothesis and the reconstruction. Blue Line: the EVBL of different specified σ^2 VAE [correspond to pre-specified σ^2_{pre} β-VAE illustrated at footnote 4 with equivalent $\sigma^2 = \beta\sigma^2_{pre}$] . Green Line: the EVLB of noise learning β-VAE with different specified β [normalized to $\sigma^2 = \beta\sigma^2_{learnt}$ for convenient comparing] . Red Plus: the EVLB of noise learning VAE. Other Figures: residual (=abs(original-reconstruct)) on the testing set.

5.1 EXPERIMENTS ON MNIST

MNIST is a database of handwritten digits. By setting β as different values, we compare the performance of β-VAE with and without considering noise modeling components on this dataset. We specifically listed the result of β(=1)-VAE in all cases. More details can be referred to in Appendix 7.3.1.

The noise specifications significantly influence the quality of final method performance in both quantity and quality, as clearly shown in Fig. (4). However, different datasets have its own noise and the relative optimal specification of noise level in practice might be really hard to be obtained. It can be seen that in most cases the β-VAE model with noise modeling is superior in learning a relatively better hypothesis with higher EVLB to that without this component. This can be easily interpreted by the fact that each dataset has its own level of noises, and noise modeling regime in VAE model tends to help the model better fit such noise and naturally conduct a better reconstruction result.



(a) (b)

Figure 5: (a) $\tilde{D}_{KL}(q(z)||p(z))$ of different VAE models & (b) Number of normal-variance generators of different VAE models (with 128 factors.)

The noise specifications significantly influence the disentanglement and noise learning β-VAE with noise modeling achieves a better disentanglement quantitatively based on the proposed indicators in the perspective of $D_{KL}(q^*(z)||p(z))$ [11] and the number of normal variance factors. In regard to the same normalized variance, according to figure 5a, the factor distributions of the β-VAEs with noise learning are closer to the prior distribution, which means it is more likely to be independent. The learnt factors with an estimated normal variance $\sigma^2_{z_h} \geq 0.8$ are counted to convey more information regarding the factor distribution. According to Fig. (5b), the β-VAEs with noise learning learn significantly more normal variance factors, in regard to the same normalized variance. The values of these indicators quantitatively show that the factors of β-VAE with noise learning are more likely/closer to be independent with each other while also guarantee to be a good distribution hypothesis in regard to maximum likelihood principle as depicted in Fig. (4).

The β-VAE with noise learning also achieves a better disentanglement qualitatively. As shown in Fig. (6) and 7, in regard to normalized variance, the β-VAE with noise learning learns more interpretable factors as well as more normal variance factors. Also, according to the estimation of mutual information, the influence of factor is also more balanced than that of traditional β-VAE. It can also be found that β-VAE has the ability to automatically suppress the auxiliary factors and learn the intrinsic factor dimension, already suggested by the information conservation theorem 1.

We find that β-VAE with noise modeling suffers from the suppression on $I_{encoder}(x; z)$, as depicted in Fig. (8a) and that is comprehensible since β-VAE is minimizing the auxiliary constraints both $I_{encoder}(x; z) + D_{KL}(q(z)||p(z))$ based on Theorem Theorem 5.

5.2 EXPERIMENTS ON EXTENDED YALE FACE DATABASE B

The extended Yale Face Database B contains images of several human subjects under different poses and different illumination conditions. In this series We compare the $\beta-$VAEs with noise Gaussian and MoG components, and network parameterized Gaussian noise and as well as different β balancing the representation ability. The Table 1 quantitatively compare the performance indicators of different methods.

[11] This term estimates the similarity of the learnt factors distribution to the unit i.i.d. Gaussian distribution.

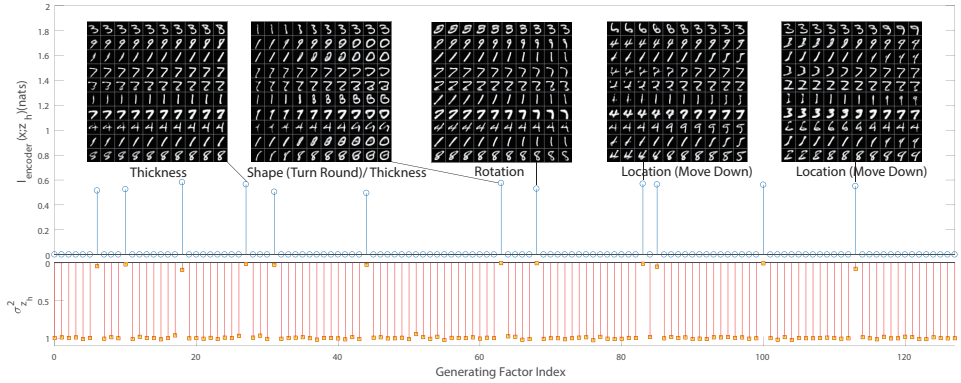


Figure 6: Noise learning β-VAE ($\beta = 8$, equivalent $\sigma^2 = 0.0944$): estimation of $I_{encoder}(x; z_h)$, σ_h^2 and qualitatively influential factor traversals. **The top pulse subgraph**: the estimated mutual information $I_{encoder}(x; z_h)$ of each factor. **The bottom reverse pulse subgraph**: the estimated variance $\sigma_{z_h}^2$ of each factor. **The montages:** influential factor traversals. In all figures of factor traversal each montage corresponds to the traversal of a single factor while keeping others fixed to their inferred (VAE, β-VAE). Each row corresponds to a different seed image used to infer latent factor value in the VAE-based models. β-VAE and VAE traversal is over [-3, 3]. Note that all the factors with $I_{encoder}(x; z_h)$ not close to zero (> 0.1) can be visually tell the existence of their generation effect. We select those factor traversals with visually most interpretable/comprehensive effects to present. Due to the limitation of space, the whole influential factor traversals are listed in appendix 7.4.2 The mutual information of "used" factor learnt by noise learning β-VAE can be found relatively balanced. It's interesting that whether the learnt estimation of $\sigma_{z_h}^2$ takes 1 or small value is strongly correlated with whether the estimation of $I_{encoder}(x; z_h)$ be near zero or not. The phenomenon of the multiple semantic change induced by the same learnt factor and the encoding of same semantic among different learnt factor tallies with factor equivalence class theorem 2.

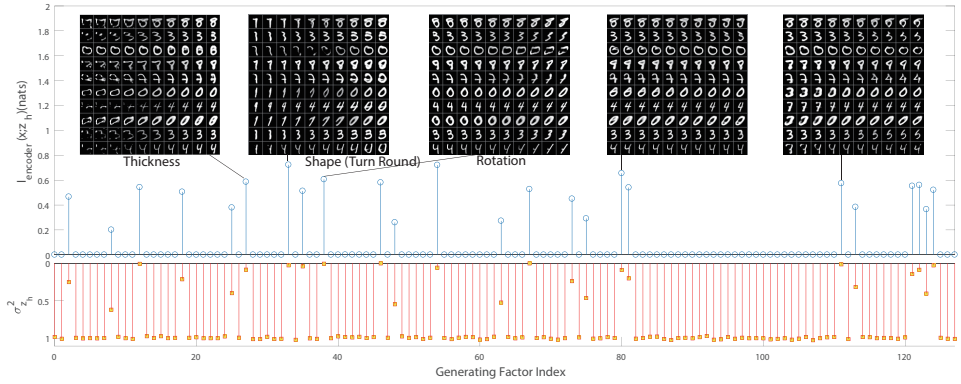
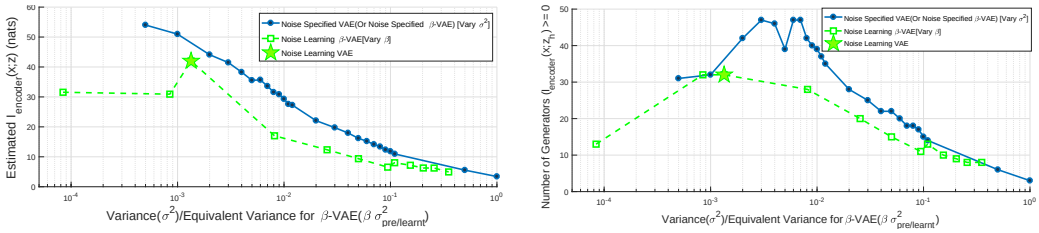


Figure 7: Noise specified (β-)VAE with equivalent $\sigma^2 = 0.1$: estimation of $I_{encoder}(x; z_h)$, σ_h^2 and qualitatively influential factor traversals. Note that we select those factor traversals with visually most interpretable/comprehensive effects to present. Due to the limitation of space, the whole influential factor traversals are listed in appendix 7.4.1. The mutual information of "used" factor learnt by noise specified β-VAE can be found more diverse than that in figure 6. The $\sigma_{z_h}^2$ and $I_{encoder}(x; z_h)$ value correlation is also significant. However, the effect of factor learnt by the noise specified VAE is hard to be intepreted (They maybe not independent with each other.).

β-VAE with MoG noise modeling ($\beta = 1$) learns an evidently better distribution hypothesis compared with the Gaussian one. It's comprehensive that MoG-VAE learns two different noise level component according to Fig. (9). One of them can be interpreted as the intrinsic physical Gaussian



(a) (b)

Figure 8: (a) $\tilde{I}_{encoder}(x;z)$ of different VAE models & (b)Number of influential generators of different VAE models

Table 1: Yale Face Database B Model Comparison [with 128 Latents (H=128)]

β	Noise	σ_k^2	π_k^2	EVLB	$\tilde{D}_{KL}(q(z)\|\|p(z))$	$\#(\sigma^2_{z_h} > 0.8)$	$\#(\mathcal{I}(z_h;x) > 0.5)$(updated)
1	G	0.00040	1	76892	127.2	0	128*
	Network	-	-	90506	127.4	0	128
	Network(MoG-2)	[0.00011 0.0053]	[0.728 0.272]	72778	127.4	0	128
	Network(G)	0.0016	1	57195	127.4	0	128
	MoG-2	[0.0001 0.0029]	[0.783 0.217]	81200	127.9	0	128
40	MoG-2	[0.00012 0.0043]	[0.719 0.281]	72346	89.28	73	46
80	MoG-2	[0.00012 0.0049]	[0.697 0.302]	69074	44.25	105	23
120	MoG-2	[0.00013 0.0054]	[0.664 0.336]	64439	27.27	114	14
160	MoG-2	[0.00013 0.0059]	[0.660 0.340]	63447	23.41	116	12

The result can be influenced by the specification of the initialization of the σ_k^2 and π_k. Variance was clipped to 0.0001 to guarantee the stable optimization. The results of Network(MoG-2) and Network(G) are derived using the fixed $G(z)$ but changing the noise hypothesis and recalculating the noise parameters. The * result is calculated by $\tilde{D}_{KL}(q(z|x)||p(z))$-$\bar{D}_{KL}(q(z)||p(z))$ and the others in the same column are calculated by $\tilde{D}_{KL}(q(z)||p(z))$.

noise and the other might be the part hard to be reconstructed. However, if the noise is assumed to be one Gaussian, then it can hardly decompose such an elaborate description of noise configurations, as shown in Fig. (9).

For the parameterized VAE, although the network parameterized noise VAE achieves the highest distribution hypothesis, the qualitatively reconstruction of the network was not as good as its EVLB. The model generates the more blurred reconstruction, which can be observed on two typical faces shown in Fig. (10). We further plug the generator $G(z)$ into MoG-2 noise hypothesis, according to the Table 1, its EVLB decreases significantly. Besides, the network parameterized noise VAE suffers severely from the numerical instability such that we could always not finish a complete training(2002 epoch) due to the its objective collapsed illustrated in Section 3.3.3. All of the aforemention evidences suggest that the model learns a relatively dedicate hypothesis for the noise rather than the deterministic part(oracle signal).

5.3 Experiments on CelebA

CelebA is a large-scale celebfaces attributes datasets and only its images are used in our experiments. We compare the VAEs of different specification of number of latent factors with both Gaussian and MoG noise modeling and as well as different β balancing the representation ability. The following Table 2 of the performance indicators shows the comparison of the model.

Table 2: CelebA Model Comparison

β	Noise	σ_k^2	π_k^2	EVLB	$\tilde{D}_{KL}(q(z)\|\|p(z))$	$\#(\sigma^2_{z_h} > 0.8)$	$\#(\bar{I}(z_h;x) > 0.5)$	# latents
40	MoG-2	[0.0030 0.029]	[0.628 0.372]	10552	24.22	94	27	128
30	MoG-2	[0.0027 0.027]	[0.637 0.363]	11324	29.72	82	32	128
1	G	0.011	1	10015	31.94	0	32	32

The $\bar{(}I(x;z_h))$ is calculated by $\tilde{D}_{KL}(q(z|x)||p(z))$-$\bar{D}_{KL}(q(z)||p(z))$ and $\tilde{D}_{KL}(q(z)||p(z))$ is better.

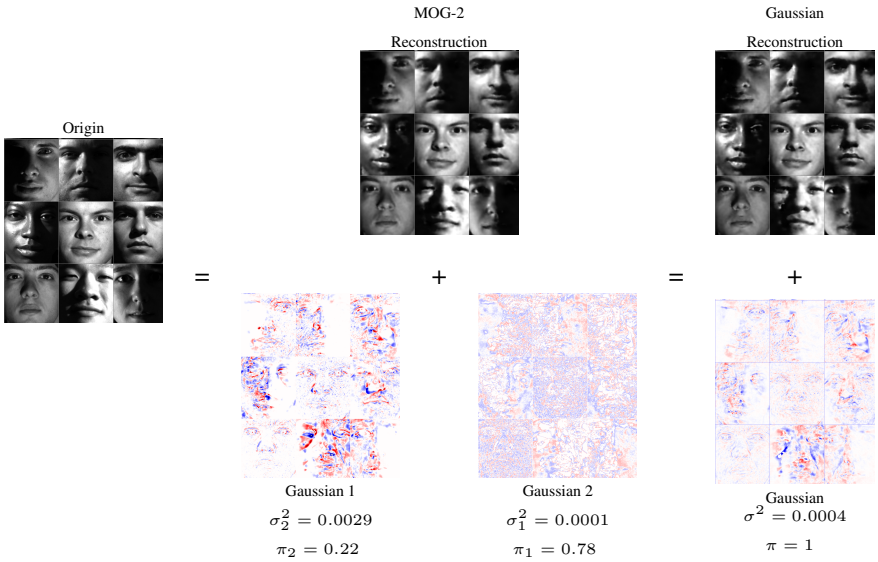


Figure 9: MoG-2/Gaussian VAE Reconstruction and Residual Gaussian Components Membership

Figure 10: Reconstruction Visual Comparison

The table shows that as compared with carefully specification of the generating factors number, noise modelling and auxiliary constraints make the model capable of learning both better hypothesis and disentangled representation.

The generating equivalence property is again well demonstrated by seeing Fig. (11): "Blue to Yellow" background change $\sim$ factor 13, 96,40,45,118. "Black to White" background change $\sim$ factor 7. Height $\sim$ factor 37,45. Mouth Open to Close background $\sim$ factor 8. Face direction change $\sim$ factor 26,31. "Male to Female" change $\sim$ factor 28. "Big to Small" face change $\sim$ factor 63,77,82. Lighting $\sim$ factor 73,90. Face lighting $\sim$ factor 120,110. Glass $\sim$ factor 73. Neck length $\sim$ factor 102. "White to Yellow" skin color change $\sim$ factor 102,96,28,63,82. Hair color $\sim$ factor 120. "Half Bright Half Gloomy" background change $\sim$ factor 110.

6 Conclusion and Perspective

In summary, the paper obtains the following conclusions:

- Integrating noise modeling component into a VAE model tends to evidently ameliorate the reconstruction quality of VAE and disentanglement performance from the evidence of the indictors and interpretability of the extracted latent factors obtained by.

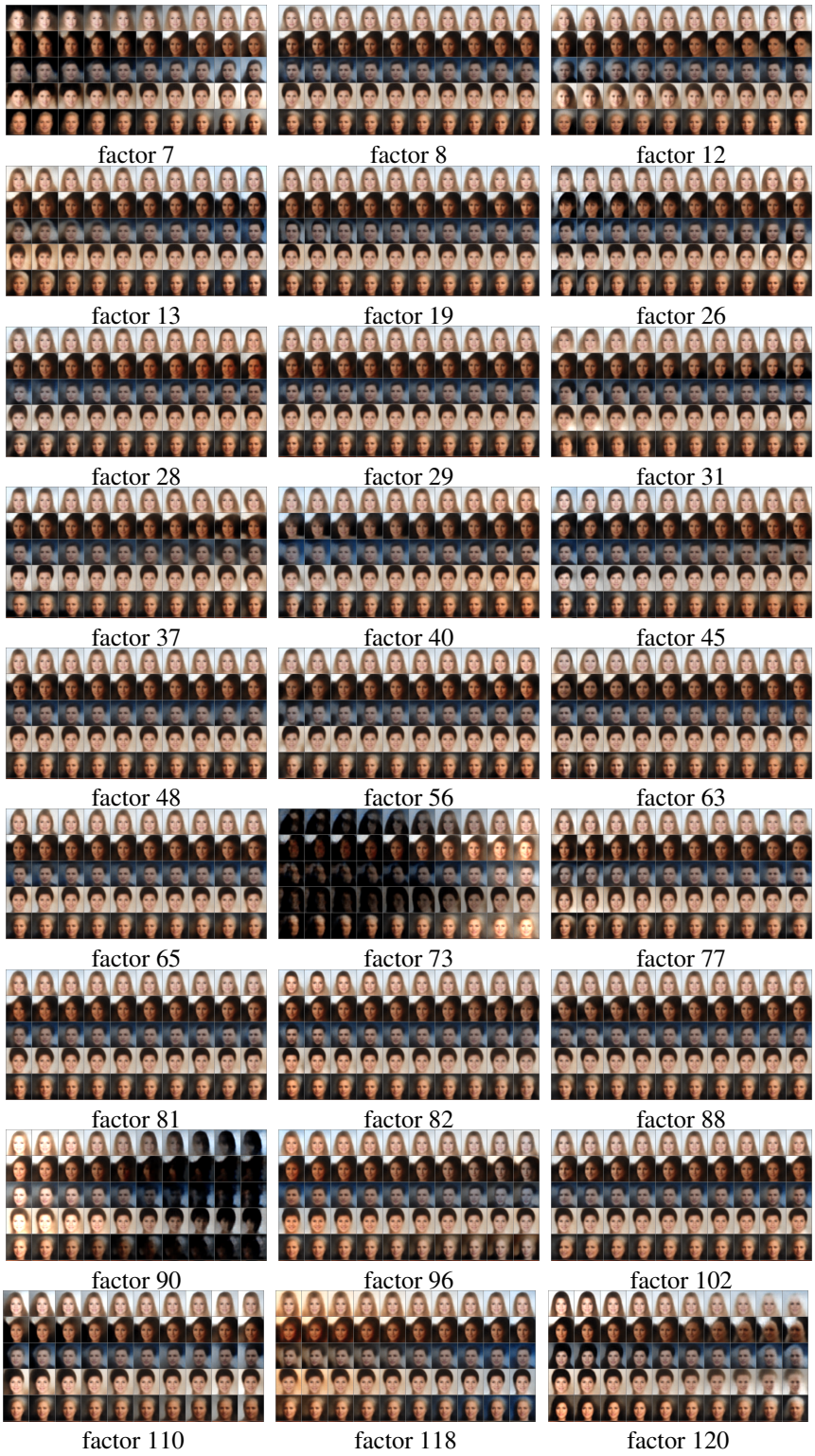


Figure 11: CelebA: Generating Factors Traversal

- β-VAE with noise modelling is able to automatically attain a relatively better distribution hypothesis and help achieve a better disentanglement performance as compared to the manipulation on the pre-specified noise (β-)VAE.

- Further, MoG β-VAE and can learn a better hypothesis distribution than the β-VAE with Gaussian noise modeling when the data noise distribution is complex.
- Network parameterized noise VAE learns a more blurred generation and tends to suffer from the numerical instability though it can learn a good distribution hypothesis.

- The Gaussian prior assumption contributes to the efficient coding of VAE model, and the idealistic VAE won't learn auxiliary dimension of generating factors.
- The learned factors of the idealistic VAE exist an equivalence class under an orthogonal linear transformation, though the semantic factors can generate the data.
- The mutual information $I_{encoder}(x; z_h)$ is a good indicator to help determine the "used" generating factors.

We further try to give some discussions which should be beneficial to our future works on this work.

Firstly, from the perspective of noise modeling:

- The physical noise in different practical scenarios, such as medical image processing/generating, can be taken into consideration while implementing the VAE model.
- The noise modelling for other generative model and deep model is also an interesting direction.

From the perspective of representation learning:

- It is interesting that the topology properties of oracle signal are used to obtain the proof for the information conservation theorem. Other situation including the data has several connected components can be further considered and would uncover the efficient coding properties of discrete factors.
- The learnt factors' variance still exists a gap to the unit Gaussian prior, and it is unsatisfactory that the auxiliary constraint suppresses the $I_{encoder}(x; z)$. A better mechanism that is innocuous to other part of VAE but complies $q(z)$ to follow the prior $p(z)$ is still required to be investigated.

REFERENCES

Dmitriy Aronov, Rhino Nevers, and David W Tank. Mapping of a non-spatial dimension by the hippocampal/entorhinal circuit. *Nature*, 543(7647):719, 2017.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *CoRR*, abs/1606.03657, 2016. URL http://arxiv.org/abs/1606.03657.

Yang Chen, Xiangyong Cao, Qian Zhao, Deyu Meng, and Zongben Xu. Denoising hyperspectral image with non-iid noise structure. *arXiv preprint arXiv:1702.00098*, 2017.

Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

Otto Fabius and Joost R van Amersfoort. Variational recurrent auto-encoders. *arXiv preprint arXiv:1412.6581*, 2014.

A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

Irina Higgins, Arka Pal, Andrei A Rusu, Loic Matthey, Christopher P Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. Darla: Improving zero-shot transfer in reinforcement learning. *arXiv preprint arXiv:1707.08475*, 2017a.

Irina Higgins, Nicolas Sonnerat, Loic Matthey, Arka Pal, Christopher P Burgess, Matthew Botvinick, Demis Hassabis, and Alexander Lerchner. Scan: Learning abstract hierarchical compositional visual concepts. *arXiv preprint arXiv:1707.03389*, 2017b.

Daniel Jiwoong Im, Sungjin Ahn, Roland Memisevic, Yoshua Bengio, et al. Denoising criterion for variational auto-encoding framework. In *AAAI*, pp. 2059–2065, 2017.

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

Diederik P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, pp. 4743–4751, 2016.

Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems*, pp. 2539–2547, 2015.

Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, pp. 1–101, 2016.

Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.

Yann Lcun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

K.C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 27(5):684–698, 2005.

Chongxuan Li, Jun Zhu, and Bo Zhang. Learning to generate with memory. In *International Conference on Machine Learning*, pp. 1177–1186, 2016.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738, 2015.

Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. *arXiv preprint arXiv:1705.08821*, 2017.

Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. Adversarial autoencoders. *CoRR*, abs/1511.05644, 2015. URL http://arxiv.org/abs/1511.05644.

Michael F Mathieu, Junbo Jake Zhao, Junbo Zhao, Aditya Ramesh, Pablo Sprechmann, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, pp. 5040–5048, 2016.

Deyu Meng and Fernando De La Torre. Robust matrix factorization with unknown noise. In *IEEE International Conference on Computer Vision*, pp. 1337–1344, 2014.

Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. *CoRR*, abs/1707.01313, 2017. URL http://arxiv.org/abs/1707.01313.

Danilo Rezende, Ivo Danihelka, Karol Gregor, Daan Wierstra, et al. One-shot generalization in deep generative models. In *International Conference on Machine Learning*, pp. 1521–1529, 2016.

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pp. 2234–2242, 2016.

Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. Joint multimodal learning with deep generative models. *arXiv preprint arXiv:1611.01891*, 2016.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408, 2010.

Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *European Conference on Computer Vision*, pp. 835–851. Springer, 2016.

Ziyu Wang, Josh Merel, Scott Reed, Greg Wayne, Nando de Freitas, and Nicolas Heess. Robust imitation of diverse behaviors. *arXiv preprint arXiv:1707.02747*, 2017.

Wei Wei, Lixuan Yi, Qi Xie, Qian Zhao, Deyu Meng, and Zongben Xu. Should we encode rain streaks in video as deterministic or stochastic? In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Wikipedia. Mental image — wikipedia, the free encyclopedia, 2017. URL https://en.wikipedia.org/w/index.php?title=Mental_image&oldid=798962875. [Online; accessed 6-October-2017].

Hongwei Yong, Deyu Meng, Wangmeng Zuo, and Lei Zhang. Robust online matrix factorization for dynamic background subtraction. *arXiv preprint arXiv:1705.10000*, 2017.

Qian Zhao, Deyu Meng, Zongben Xu, Wangmeng Zuo, and Lei Zhang. Robust principal component analysis with complex noise. In *International Conference on Machine Learning*, pp. 55–63, 2014.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. Learning hierarchical features from generative models. *arXiv preprint arXiv:1702.08396*, 2017.

7 APPENDIX

7.1 PROOF

Proof. For theorem 1. Proof by Contradiction. Suppose those two function exist, and we will show that they will be inverse mapping of each other and the homeomorphism mapping of $\mathbb{R}^H$ and $\mathbb{R}^P$. Since $\mathbb{R}^H$ and $\mathbb{R}^P$ have different topology structures ($P \neq H$), the homeomorphism mapping will not exist.

$$z = g(y) = g(f(z)) \; \forall z \in \mathbb{R}^H \Rightarrow g \circ f = I_H$$

$$y = f(z) = f(g(x)) \; \forall y \in \mathbb{R}^P \Rightarrow f \circ g = I_P$$

Since both f and g are continuous, there is a homeomorphism mapping between $\mathbb{R}^H$ and $\mathbb{R}^P$ and it leads to the contradiction. □

Proof. For theorem 2. We only need to test the mean and variance of y.

$$\mathbb{E}(y) = \mathbb{E}(Qz) = Q\,\mathbb{E}(z) = 0$$
$$Cov(y, y) = QCov(z, z)Q^T = QIQ^T = I$$

Therefore, y is another set of H independent unit gaussian random variables. Since $x = Q^T y$, z and y can generate each other with an linear homeomorphism mapping. □

Proof. For theorem 4,

$$\mathcal{I}(z_1, \cdots, z_H; x) = \int p(z_1, \cdots, z_H, x) \log \frac{p(z_1, \cdots, z_H, x)}{p(z_1, \cdots, z_H)p(x)} dz_1 \cdots dz_H dx$$
$$= \int p(z_1, \cdots, z_H, x) \log \frac{\prod_{h=1}^{H} p(z_h|x)}{\prod_{h=1}^{H} p(z_h)} dz_1 \cdots dz_H dx = \sum_{h=1}^{H} \int p(z_h, x) \log \frac{p(z_h|x)}{p(z_h)} dz_h dx$$
$$= \sum_{h=1}^{H} \mathcal{I}(z_h; x).$$

□

Proof. For theorem 5.

$$\mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||p(z)) = \int q(z|x)p_{data}(x) \frac{q(z|x)p_{data}(x)}{p(z)p_{data}(x)} dx$$
$$= \int q(z|x)p_{data}(x) \frac{q(z|x)p_{data}(x)}{q(z)p_{data}(x)} \frac{q(z)}{p(z)} dx = \mathcal{I}_{encoder}(x; z) + D_{KL}(q(z)||p(z)). \tag{29}$$

□

7.1.1 Auxiliary Explanations for Indicators

Corollary 1. *The terminology follows the aforemention definitions and if the involved KL-divergence and mutual information be well defined then*

$$\mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||q^*(z)) = \mathcal{I}_{encoder}(x; z) + D_{KL}(q(z)||p(z)). \tag{30}$$

The proof of corollary 1 is the same as that of theorem 5. This corollary suggests that the estimation in definition 2 provides another upper bound for the capacity of the encoder network. Empirically, this estimation is a much tighter estimation than using the estimation in definition 1.

Corollary 2. *The terminology follows the aforemention definitions and if the involved KL-divergence and mutual information be well defined then*

$$\mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||p(z)) - \mathop{\mathbb{E}}_{x\sim p_{data}(x)} D_{KL}(q(z|x)||q^*(z))$$
$$= D_{KL}(q(z)||p(z)) - D_{KL}(q(z)||q^*(z)) \leq D_{KL}(q(z)||p(z)). \tag{31}$$

The corollary is an direct result of theorem 5 and corollary 1. It suggests that the estimation in definition 4 is a lower bound for $D_{KL}(q(z)||p(z))$.

Definition 5 (Another Estimation for $D_{KL}(q(z)||p(z))$)**.**

$$\bar{D}_{KL}(q(z)||p(z)) = D_{KL}(q^*(z)||p(z)). \tag{32}$$

Empirically, $\bar{D}_{KL}(q(z)||p(z))$ and $\tilde{D}_{KL}(q(z)||p(z))$ shown the same estimation results on MNIST.

Definition 6 (Another estimation for $I_{encoder}(x; z)$)**.**

$$\bar{I}_{encoder}(x; z) = -D_{KL}(q^*(z)||p(z)) + \frac{1}{M} \sum_{m=1}^{M} D_{KL}(q(z|x_m)||p(z)). \tag{33}$$

Definition 7 (Another estimation for $I_{encoder}(x; z_h)$ which quantifies the influence of each factor)**.**

$$\bar{I}_{encoder}(x; z_h) = -D_{KL}(q^*(z_h)||p(z_h)) + \frac{1}{M} \sum_{m=1}^{M} D_{KL}(q(z_h|x_m)||p(z_h)). \tag{34}$$

7.2 ANALYSIS ON THE DISENTANGLEMENT METRIC RAISED IN β-VAE (HIGGINS ET AL. (2016))

The terminology inherits those in the β-VAE paper. The main idea of that disentanglement metric is to create a statistic point z_{diff} relevant to the model for each simulated factor respectively and then to use a linear classifier to project the statistic point to the corresponding index of the simulated factor. If the statistic points induced by the model are easy to be separated then the model is thought to learn disentangle representation.

Here, we will argue that even for the idealistic VAE model that already learnt i.i.d unit Gaussian factors may receive bad score under that performance metric.

Suppose that the true simulated factors v be with distribution $\mathcal{N}(0, I_H)$. Then the learnt "used"[12] factor z can be in the equivalence class $[v]$ according to theorem 2. Concretely, there exists an orthogonal transformation Q such that $z = Qv$.

Suppose that the simulated factors with index y of v is fixed. Suppose $v^1_{y-fixed}$ and $v^2_{y-fixed}$ are two random variable representing the samples from the y-fixed v. Then the factors inferred by the idealistic VAE turns to be $z^1 = Qv^1_{y-fixed}$ and $z^2 = Qv^2_{y-fixed}$.

In order to calculate the statistic point $z_{diff}(y) = \mathbb{E}\,|z^1 - z^2|$, we first calculate the mean and variance of $(z^1 - z^2)$.

$$\mathbb{E}(z^1 - z^2) = Q\,\mathbb{E}(v^1_{y-fixed} - v^2_{y-fixed}) = 0 \tag{35}$$

$$\begin{aligned} Var(z^1 - z^2) &= Var(Q(v^1_{y-fixed} - v^2_{y-fixed})) = QCov(v^1_{y-fixed} - v^2_{y-fixed}, v^1_{y-fixed} - v^2_{y-fixed})Q^T \\ &= Qdiag(2, \cdots, 2, \underline{0}_y, 2, \cdots, 2)Q^T = 2I - Qdiag(0, \cdots, 0, \underline{2}_y, 0, \cdots, 0)Q^T = 2I - 2q_y q_y^T. \end{aligned} \tag{36}$$

Therefore, $z_{diff}(y)$ can be calculated through the diagonal value of $2I - 2q_y q_y^T$. That is,

$$z_{diff}(y) = \mathbb{E}(|z^1 - z^2|) = 2\sqrt{\frac{2}{\pi}}(\sqrt{(1 - q^2_{y1})}, \cdots, \sqrt{(1 - q^2_{yH})})^T. \tag{37}$$

From the above equation, the location of statistic point is unique determined by $(q^2_{y1}, \cdots, q^2_{yH})$. When $(q^2_{y1}, \cdots, q^2_{yH})$ is close to the vertex of the unit cubic for each y then all the statistic points turn to be easily separated.

However, from the perspective of the objective, all the orthogonal Qs are with the same potential to be learnt. It seems not to be with a small probability that statistic points of different indexes take similar location. For instance, if $H = 2$ and $Q = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$ then $z_{diff}(1) = z_{diff}(2)$ cannot be separated while the learnt factors $z = (z_1, z_2)^T$ are independent with each other and are already able to perform an efficient disentangled representation.

Among different trials, the Q might contribute to that disentanglement metric but also might not. That explains the reason why that metric is unstable.

7.3 EXPERIMENT DETAILS

We set L to 1, and minibatch size M to be 100 in all practices. All the pixels value have been linear normalized in to [0,1].

7.3.1 MNIST

We split randomly 7000 datapoints according to ratio $[0.6 : 0.2 : 0.2]$ into training set, validation set (no use), testing set. All the indicators and $q^*(z)$ are evaluated/calculated on 10000 datapoints

[12] The auxiliary unused factor is innocuous for the subsequent analysis.

belonging to the testing set. All the seed images used to infer latent code and to draw the traversal come from the testing set.

In all figures of latent code traversal each block corresponds to the traversal of a single latent variable while keeping others fixed to either their inferred (β-VAE, VAE). Each row represents a different seed image used to infer the latent values in the VAE-based models. β-VAE and VAE traversal is over the $[-3, 3]$ range.

The assumed variance σ^2 of noise specified Gaussian of VAE models is enumerated from $[0.0005, 0.001 : 0.001 : 0.012, 0.02 : 0.01 : 0.11]$. The β setting for the noise learning β-VAE is enumerated from $[0.1, 0.5, 1, 2 : 2 : 18]$.

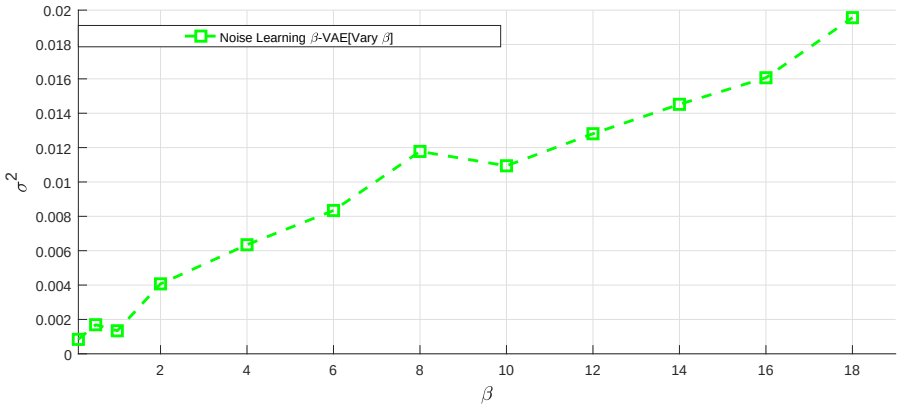


Figure 12: Learned σ^2 of different β setting

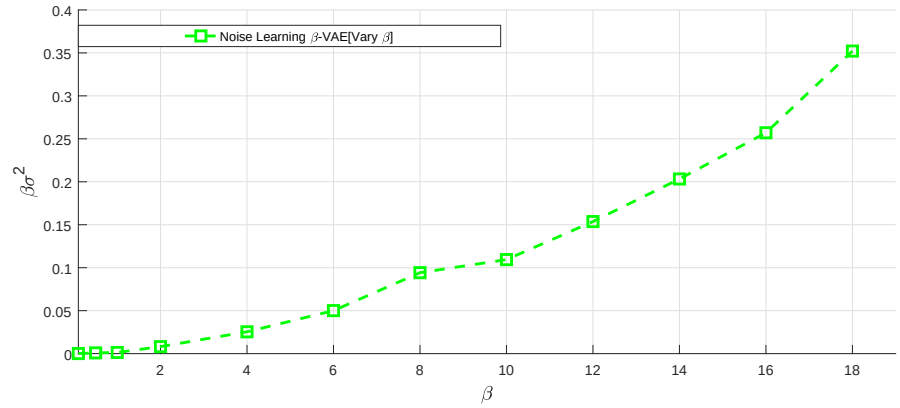


Figure 13: $\beta\sigma^2$ of different β setting

Empirically, we find that the different KL-divergence estimations definition 5 and definition 2 show the same results on this datasets. We calculate the estimation for $\mathcal{I}(x; z_h)$ in figure 7 and 6 through $\tilde{D}_{KL}(q(z|x)||p(z))$-$\bar{D}_{KL}(q(z)||p(z))$. It's better to use $\tilde{D}_{KL}(q(z)||p(z))$ but it's innocuous on this datasets.

7.3.2 Extended Yale Face B

We split randomly 2424 datapoints according to ratio $[0.8 : 0.1 : 0.1]$ into training set, validation set (no use), testing set. The model is training on the training set. All the indicators and $q^*(z)$ are evaluated/calculated on 400 datapoints selected from entire datasets. All the seed images used to infer latent code and to draw the traversal come from the 100 datapoints from the testing set.

In all figures of latent code traversal each block corresponds to the traversal of a single latent variable while keeping others fixed to either their inferred (β-VAE, VAE). Each row represents a different seed image used to infer the latent values in the VAE-based models.

β-VAE and VAE traversal is over the $[-3, 3]$ range.

The β setting for the noise learning β-VAE is enumerated from $[1, 40, 80, 120, 160]$.

7.3.3 CELEBA

We split randomly roughly 200000 datapoints according to ratio $[0.8 : 0.1 : 0.1]$ into training set, validation set (no use), testing set. The model is training on the training set. All the indicators and $q^*(z)$ are evaluated/calculated on 10000 datapoints selected from testing set. All the seed images used to infer latent code and to draw the traversal come from the 100 datapoints from the testing set.

In all figures of latent code traversal each block corresponds to the traversal of a single latent variable while keeping others fixed to either their inferred (β-VAE, VAE). Each row represents a different seed image used to infer the latent values in the VAE-based models.

β-VAE and VAE traversal is over the $[-3, 3]$ range.

The β setting for the noise learning β-VAE is enumerated from $[1, 30, 40]$.

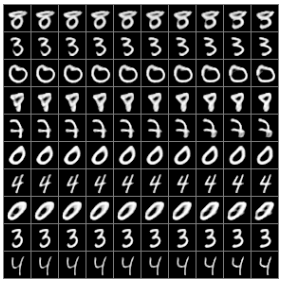
7.3.4 NETWORK STRUCTURE

Dataset	**Optimiser**	**Architecture**	
Mnist	Adam $1e-4$ Epoch 200	Input Encoder Latents Decoder	28x28x1 Conv 32x4x4,32x4x4 (stride 2). FC 256. ReLU activation. 128 FC 256. Linear. Deconv reverse of encoder. ReLU activation.
CelebA	Adam $1e-4$ Epoch 20	Input Encoder Latents Decoder	64x64x3 Conv 32x4x4,32x4x4,64x4x4,64x4x4 (stride 2). FC 256. ReLU activation. 128/32 FC 256. Linear. Deconv reverse of encoder. ReLU activation.
Extended Yale Face B	Adam $1e-4$ Epoch 2002	Input Encoder Latents Decoder	192x168x1 Conv 32x4x4,32x4x4,64x4x4,64x4x4 (stride 2). FC 256. ReLU activation. 128 FC 256. Linear. Deconv reverse of encoder. ReLU activation.
Extended Yale Face B (Network Parameterized Noise)	Adam $1e-4$ Epoch 1460	Input Encoder Latents Decoder	192x168x1 Conv 32x4x4,32x4x4,64x4x4,64x4x4 (stride 2). FC 256. ReLU activation. 128 FC 256. Linear. Deconv reverse of encoder. ReLU activation.

7.4 AUXILIARY GENERATING PICTURE

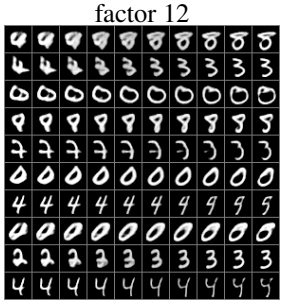
Note that only the factors with $\mathcal{I}_{encoder}(x; z_h) > 0.5$ are shown.

factor 12

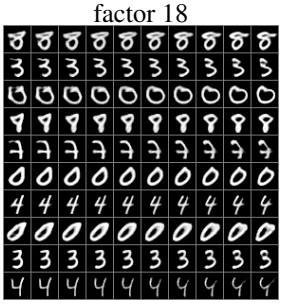


factor 18

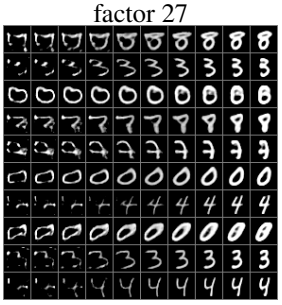
factor 27



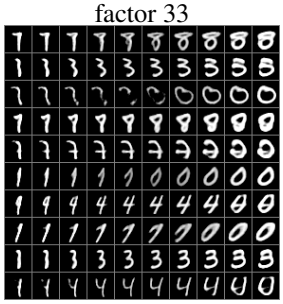
factor 33



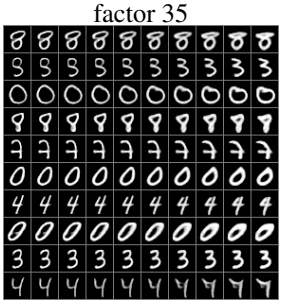
factor 35



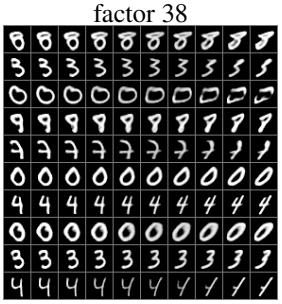
factor 38



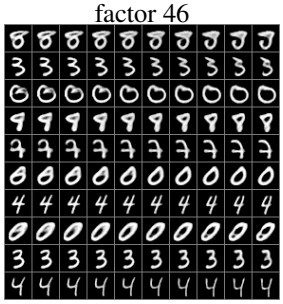
factor 46



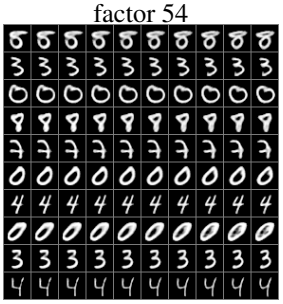
factor 54



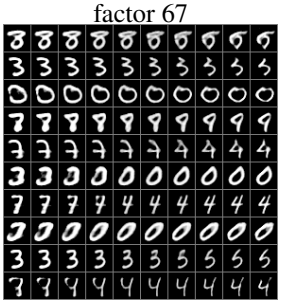
factor 67



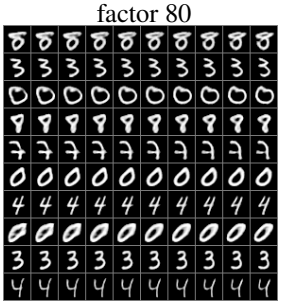
factor 80



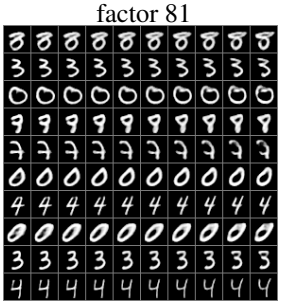
factor 81



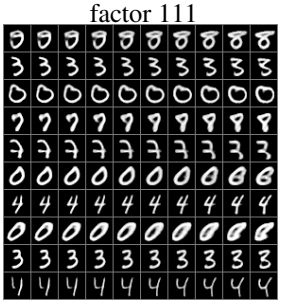
factor 111



factor 121

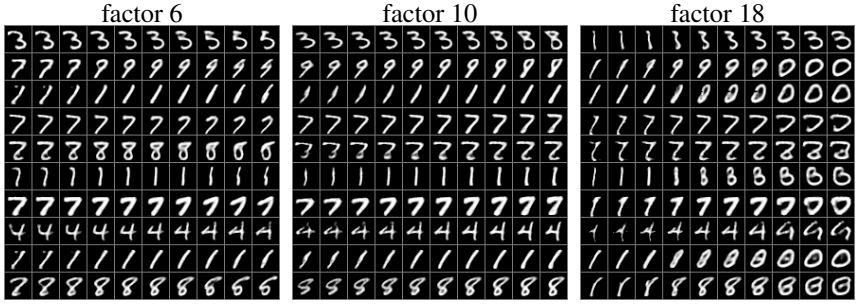


factor 122

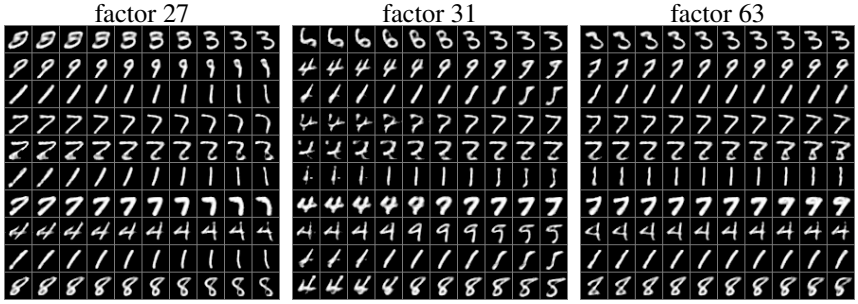


factor 124

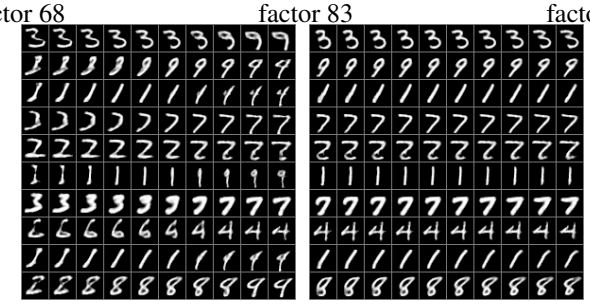
factor 6 factor 10 factor 18



factor 27 factor 31 factor 63



factor 68 factor 83 factor 85



factor 100 factor 113

7.4.1 MNIST: GENERATING FACTOR TRAVERSAL OF $\sigma^2 = 0.1$ PRE-SPECIFIED VAE

7.4.2 MNIST: GENERATING FACTOR TRAVERSAL OF NOISE LEARNING β-VAE

7.4.3 EXTENDED YALE FACE B: GENERATING FACTOR TRAVERSAL OF NOISE LEARNING $\beta(= 120)$-VAE

7.4.4 EXTENDED YALE FACE B: RECONSTRUCTION VISUAL COMPARISON

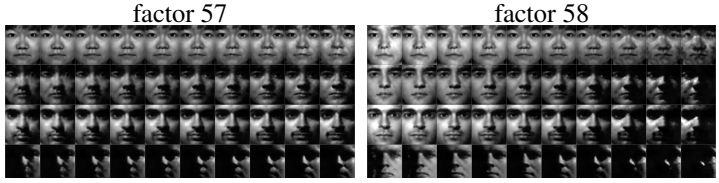
7.4.5 EXTENDED YALE FACE B:NETWORK NOISE LEARNT MEAN WITH MOG-2 NOISE HYPOTHESIS DEMONSTRATION

7.4.6 CELEBA: MOG-2 β(=40)-VAE RECONSTRUCTION AND RESIDUAL GAUSSIAN COMPONENTS MEMBERSHIP

factor 23 factor 24

factor 37 factor 52

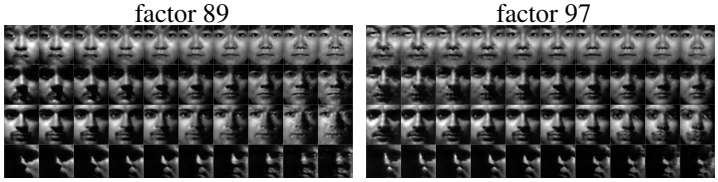
factor 57 factor 58



factor 77 factor 80

factor 85 factor 88

factor 89 factor 97



factor 105 factor 123

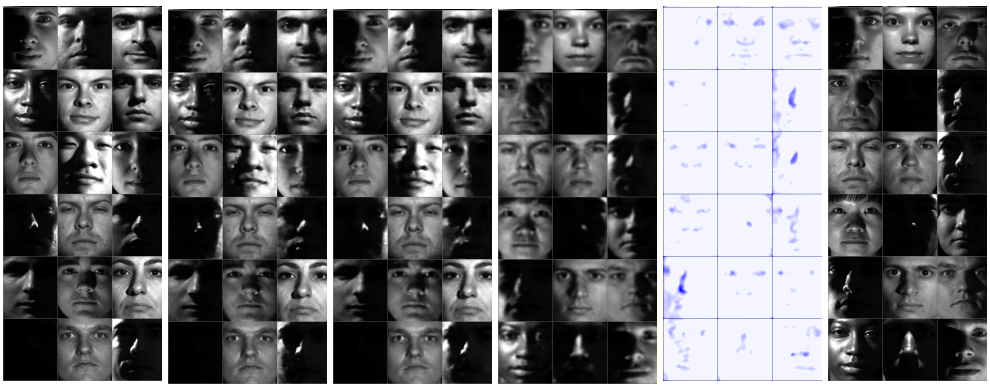


Figure 14: Reconstruction Visual Comparison full

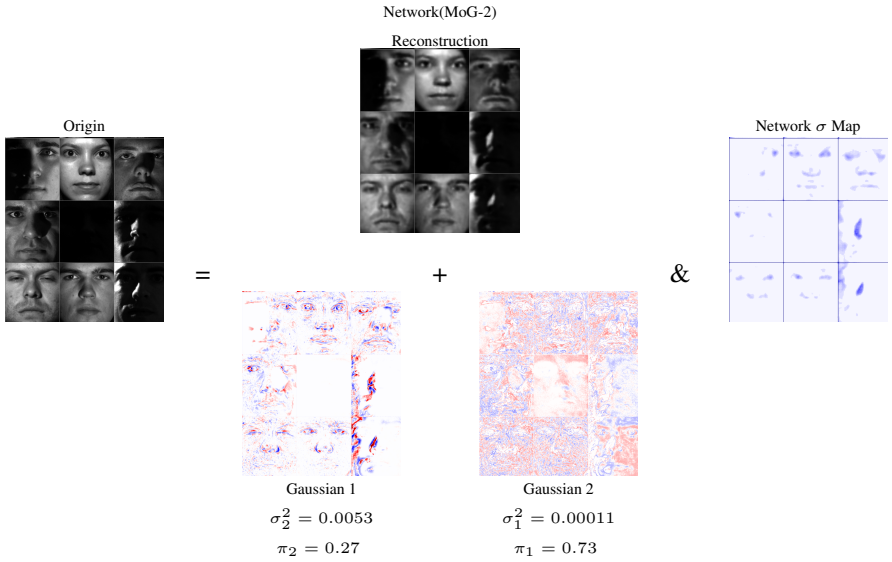


Figure 15: Network (MoG-2) Reconstruction and Residual Gaussian Components Membership

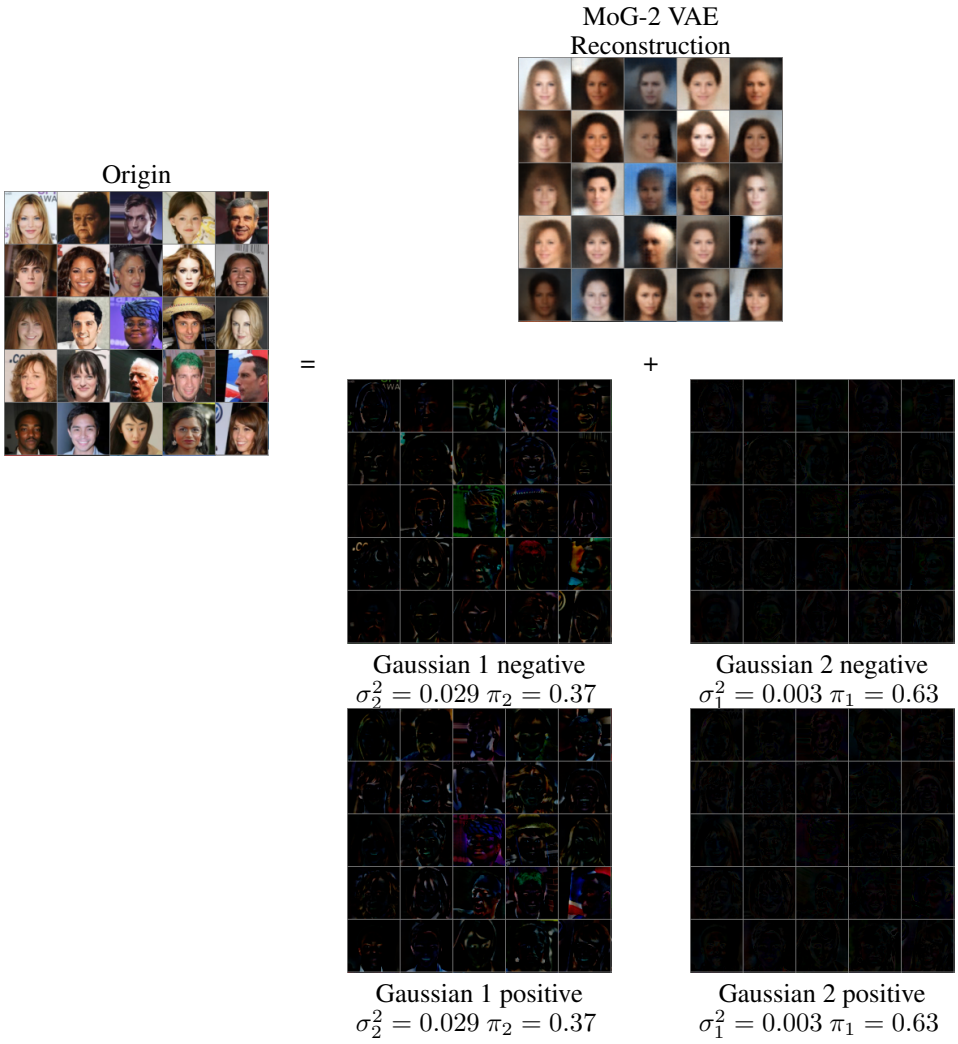


Figure 16: MoG-2 β(=40)-VAE reconstruction and residual Gaussian components membership