Generalizing Variational Autoencoders with Hierarchical Empirical Bayes

Wei Cheng  Gregory Darnell  Sohini Ramachandran  Lorin Crawford
Brown University  Providence, RI 02912

Abstract

Variational Autoencoders (VAEs) have experienced recent success as data-generating models by using simple architectures that do not require significant fine-tuning of hyperparameters. However, VAEs are known to suffer from over-regularization which can lead to failure to escape local maxima. This phenomenon, known as posterior collapse, prevents learning a meaningful latent encoding of the data. Recent methods have mitigated this issue by deterministically moment-matching an aggregated posterior distribution to an aggregate prior. However, abandoning a probabilistic framework (and thus relying on point estimates) can both lead to a discontinuous latent space and generate unrealistic samples. Here we present Hierarchical Empirical Bayes Autoencoder (HEBAE), a computationally stable framework for probabilistic generative models. Our key contributions are two-fold. First, we make gains by placing a hierarchical prior over the encoding distribution, enabling us to adaptively balance the trade-off between minimizing the reconstruction loss function and avoiding over-regularization. Second, we show that assuming a general dependency structure between variables in the latent space produces better convergence onto the mean-field assumption for improved posterior inference. Overall, HEBAE is more robust to a wide-range of hyperparameter initializations than an analogous VAE. Using data from MNIST and CelebA, we illustrate the ability of HEBAE to generate higher quality samples based on FID score than existing autoencoder-based approaches.

1 Introduction

Generative modeling has achieved tremendous success in recent years by enabling unsupervised learning of different data distributions, as well as interpretation of data via low-dimensional representations. There two popular approaches in this space: Generative Adversarial Networks (GANs) [1] and Variational Autoencoders (VAEs) [2]. In GANs, one plays a min-max game between a discriminator and a generator where the generator is trained to produce high quality samples that fool the discriminator. GANs suffer from a lack of theoretical support that produces problems like “mode collapse” and makes training difficult [3–5]. VAEs, on the other hand, use the neural network architecture of autoencoders and further draw on variational inference. Instead of simply encoding input features into isolated variables in the latent (or hidden) space and reconstructing them using decoders, VAEs further impose a standard normal prior distribution over the latent variables. This prior smooths the regularized latent space during training, thereby enabling the generation of meaningful samples. VAEs leverage well-established theory and are easier to train than GANs. However, VAEs have been known to generate lower quality samples than GANs, and can also result in over-regularization problems such as posterior collapse [6–8]. Previous studies have used manual tuning of hyperparameters to prevent over-regularization and have made various attempts to...
We begin with reviewing the modeling assumptions underlying the variational autoencoder (VAE) framework. An autoencoder has two key components: an encoder which compresses the original inputs $x_i$ to a lower $k$-dimensional latent variable $z$, and a decoder which takes those latent variables $z$ and attempts to reconstruct the original data (often denoted by $x'_i$). Intuitively, a successfully trained model aims to minimize the loss function $\sum_i ||x_i - x'_i||^2$. The VAE framework can be viewed as a probabilistic version of an autoencoder where the posterior distribution of the latent variable $z$ is imposed to match a prior distribution $p_\theta(z)$. By matching the targeted distribution, VAEs can learn a smooth latent space such that it will not just encode isolated data points but produces a generative model over the underlying latent variables $\theta$. Typical VAEs assume a standard Gaussian prior structure between the latent variables. More specifically, the encoder portion of a VAE aims to find a "best" approximation to the prior $q_\phi(z | x)$ based on the data and a set of free parameters $\phi$. The decoder then aims to construct a likelihood $p_\theta(x | z)$ conditioned on the latent variables $z$. The goal of variational inference is to maximize the marginal log-likelihood for each example $x_i$ in the batch, which takes the following form

$$\log p_\theta(x_i) = KL(q_\phi(z | x_i) \parallel p_\theta(z | x_i)) + \mathbb{E}_{q_\phi(z | x_i)}[\log p_\theta(x_i, z) - \log q_\phi(z | x_i)]$$

where the first term measures the Kullback-Leibler (KL) divergence between the approximate and true posterior distribution of the latent $z$ given the data $x_i$.

Using Jensen’s equality, one can formulate a lower bound to the marginal log-likelihood in Eq. (1), and then iteratively adjust the free parameters $\phi$ so that this bound becomes as tight as possible. It can be shown that finding the “best” approximation in the encoder amounts to finding the free parameters $\phi$ that minimizes the KL divergence between $q_\phi(z | x)$ and $p_\theta(z)$ (2). Taking a stochastic gradient variational Bayes (SGVB) (2) view on VAEs yields the following general expression for the lower bound of the log-likelihood

$$\log p_\theta(x_i) \geq \mathcal{L}_{\text{VAE}}(\theta, \phi; x_i) = \frac{1}{L} \sum_{l=1}^{L} \log p_\theta(x_i | z^{(l)}) - KL(q_\phi(z | x_i) \parallel p_\theta(z))$$

The first term on the right hand side of Eq. (2) is normally referred to as the “reconstruction loss” (2,4,8) and resembles the regular loss function in autoencoders. The second term on the right hand side of Eq. (2) can then be viewed as a “regularized loss” function where the variational posterior distribution $q_\phi(z | x_i)$ is being adjusted to approximate the prior $p_\theta(z)$ (2,4,8). Lastly, we use $z^{(l)}$ to denote an empirically sampled latent variable using the re-parameterization trick where

$$z^{(l)} = \mu(x_i) + \sigma(x_i) \odot \varepsilon^{(l)}, \quad \varepsilon \sim \mathcal{N}(0, I)$$
To compute an $L$ number of realizations of $z$, the encoder outputs a $k$-dimensional mean vector $\mu(x_i) = [\mu(x_{i1}), \ldots, \mu(x_{ik})]$ and a $k$-dimensional vector of standard deviations $\sigma(x_i) = [\sigma(x_{i1}), \ldots, \sigma(x_{ik})]$ for each $x_i$. By sampling random noise to determine each $z^{(1)}, \ldots, z^{(L)}$, we are able to analytically compute the reconstruction loss and the regularized loss in Eq. (2).

2.2 Challenges and Disadvantages of the VAE Framework

There are two key challenges and disadvantages with the VAE framework. First, in practice to train a VAE, one has to balance between minimization of the reconstruction loss and the regularized loss via the KL divergence [9, 5]. Concentrating effort on the latter can result in over-regularization of the approximate posterior distribution $q_\phi(z|x_i)$ [4]. A related issue is posterior collapse [8, 10, 11], where a local optimum of the VAE objective is obtained such that $q_\phi(z|x_i) = p_\theta(z)$ and $\text{KL}(q_\phi(z|x_i) \| p_\theta(z)) \to 0$. In an extreme case, if the KL divergence dictates that the encoder always outputs $\mu(x_i) = 0$ and $\sigma(x_i) = 1$ for all samples, the decoder will face the impossible task of reconstructing different samples from completely random noise $z^{(l)} = \epsilon^{(l)}$. To effectively balance this trade-off, VAEs require manually fine-tuning the weight of the KL component and model-specific hyper-parameters [6, 9]. Moreover, finding the optimal value of the KL divergence remains an open question [12]. Since the optimal trade off is unclear, VAEs often generate low quality samples even when the model is fine-tuned.

The second key disadvantage of the VAE framework lies within the assumption that the variational posterior distribution follows an isotropic Gaussian (e.g., Eq. (4)). In this case, the VAE cannot completely guarantee that the inference algorithm will converge onto the standard Gaussian prior if the $k$ latent neurons $z = (z_1, \ldots, z_k)$ are in fact correlated. To see this, notice that the practical lower bound for the joint log-likelihood across all $m$ examples in the training batch is given as

$$L_{\text{VAE}}(\theta, \phi; x_{1:m}) \approx \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i \| z^{(l)}) - \lambda \sum_{i=1}^{m} \sum_{j=1}^{k} \left[ \log \sigma_{ij}^2 + 1 - \sigma_{ij}^2 - \mu_{ij}^2 \right] \tag{4}$$

where $\lambda \geq 0$ is a practical weight parameter for the KL term to balance the trade-off with reconstruction [4, 13]; $\mu_{ij}$ and $\sigma_{ij}$ denotes the $j$-th term of $\mu(x_i)$ and $\sigma(x_i)$, and the last term is the closed form of the KL divergence between two Gaussian distributions $q_\phi(z \| x_i) = \mathcal{N}(\mu(x_i), D(x_i))$ where $D(x_i) = \text{diag}(\sigma(x_i))$ and $p_\theta(z) = \mathcal{N}(0, I)$. While the VAE will impose that each $\sigma_{ij}^2 \rightarrow 1$ and $\mu_{ij} \rightarrow 0$, in the event that the latent variables are correlated, the model will not approach an optimal fit without also imposing that the covariance $\mathbb{V}[z_{ij}, z_{ij'}] = 0$ for every $j \neq j'$ combination.

In this work, we will show that placing a flexible prior distribution over the mean function of the encoder enables us to efficiently achieve the optimal trade-off between the reconstruction and regularization loss function in the VAE framework. In our framework assume a general covariance structure for the latent variables without sacrificing model efficiency. With our hierarchical prior specification, our model results in better matching posteriors and higher quality generated samples.

2.3 Wasserstein Autoencoders and Maximum Mean Discrepancy

A major goal of the generative model within autoencoders is to derive a smoothed latent space. Recently, there have been many efforts that aim to improve this portion of the VAE framework. Here, we briefly review one popular approach. The Wasserstein autoencoder (WAE) starts from an optimal transport point-of-view and aims to force the “aggregated” posterior distribution $q_\phi(z)$ to match the standard normal prior [14]. One major difference between the WAE and VAE is that the WAE uses a deterministic mapping function between the training inputs $x$ and the latent variables $z$. As stated in the original paper [3], one can write the WAE in terms of the VAE framework by regarding $q_\phi(z \| x_i)$ as a delta mass function $\delta(\mu(x_i))$ such that $z = \mu(x_i)$. The WAE then imposes that the aggregated posterior $q_\phi(z) = \phi_\theta(\mu(x_i))$ matches the standard normal by replacing the KL divergence in the regularized loss function with a maximum mean discrepancy (MMD) distance between $q_\phi(z)$ and the standard normal prior. Similar to VAE, the WAE does not explicitly penalize the covariance between latent variables and, thus, cannot guarantee independence between features at convergence.
2.4 Disadvantages of the WAE Framework.

In the next section, we argue that completely abandoning a probabilistic framework and relying solely on deterministic point estimates can lead to a non-smooth latent space and result in a generative model that produces unrealistic samples. Indeed, previous studies have shown that the aim of the WAE is equivalent to maximizing a “looser” lower bound \[12, 3\]. In this study, we propose a new framework that combines the merits of the WAE and VAE. We retain the probabilistic nature of the VAE so we can smooth out the latent space, and we further regularize the aggregated posterior so we can relax the trade-off in the loss function much like the WAE.

3 Proposed Method

3.1 Hierarchical Empirical Bayes Autoencoders (HEBAE)

In this section, we present the Hierarchical Empirical Bayes Autoencoder (HEBAE) framework. For convenience, we will follow previous work and treat each sample index \(i\) as a random variable \[12\]. To this end, we will denote distributions as \(q_\phi(z_i | x_i)\) and \(p_\theta(z_i)\) with subscripts corresponding to the \(i\)th sample. The two key components of the HEBAE framework is that (i) it assumes a Gaussian process (GP) prior over the encoder function that takes input features to the compressed latent space \(\mu : x_i \rightarrow z_i\), and (ii) it assumes a general covariance structure between the latent variables. For the first component, assume a hierarchical prior where \(\mu(\cdot)\) is completely specified by its mean function and positive definite covariance (kernel) function, \(\beta(\cdot)\) and \(\Sigma(\cdot, \cdot)\), respectively. In practice, since we have a finite number of examples in a given training batch, we can take a weight-space view on the Gaussian process and equivalently say

\[
\mu(x_i) \sim \mathcal{N}(\beta, \Sigma)
\]  

(5)

where the encoder function \(\mu\) is assumed to follow from a multivariate normal distribution with mean vector \(\beta\) and general variance-covariance structure \(\Sigma\). For the second key component in the HEBAE framework, we choose non-isotropic Gaussian distributions as our approximating posterior

\[
q_\phi(z_i | x_i) = \mathcal{N}(\mu(x_i), \sigma_i^2 \Sigma)
\]  

(6)

where, similar to the traditional VAE framework, \(\mu(x_i)\) denotes the mean output for the \(i\)th sample from the encoder. Unlike the traditional VAE, we assume that the latent variables also have a correlation structure that is proportional to what is mapped by the encoder scaled by a sample-specific variance component parameter. One could be fully bayesian and also place priors on \(\beta\) and \(\Sigma\); however, to keep the simple structure of original VAEs and save computational time, we derive empirical estimators for \(\beta\) and \(\Sigma\) simply using a batch of \(m\) training samples.

3.2 Model Training via Variational Inference

Our goal is to find the ideal trade-off between the reconstruction error and regularization term in the VAE framework loss function. Similar to the logic posed within the WAE, we propose maximizing the lower bound to the marginal likelihood by imposing the aggregated posterior \(q_\phi(z)\) to match a standard normal distribution instead of regularizing each of the independent conditional posteriors \(q_\phi(z_i | x_i)\). In this section, we will use variational inference to show that this target can be achieved by minimizing the divergence between \(q_\phi(\mu(x_i))\) and the standard normal distribution. We begin with the form of the lower bound within the HEBAE framework.

**Theorem 1** Minimizing \(\text{KL}(q_\phi(\mu(x_i)) \| \mathcal{N}(0, I))\) is equivalent choosing a general isotropic Gaussian as the prior distribution such that \(p_\theta(z_i) = \mathcal{N}(\mu(x_i), \sigma_i I)\) with the constraint that \(\beta^T \beta = 0\). This yields the lower bound to optimize in the HEBAE framework.

\[
\mathcal{L}(\theta, \phi; x_1:m) = \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda_1 \sum_{i=1}^{m} \text{KL}(q_\phi(z_i | x_i) \| p_\theta(z_i)) + \lambda_2 \| \beta \|^2
\]  

\[
= \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda \sum_{i=1}^{m} \text{KL}(q_\phi(\mu(x_i)) \| \mathcal{N}(0, I)),
\]  

(7)
Notice that under the KKT conditions, the constraint in the first half of Eq. (7) can be achieved by incorporating an $L_2$-penalty on $\beta$ into the objective \cite{15, 16}. One can interpret the weight $\lambda$ outside the KL term in the second half of Eq. (7) as a regularization parameter similar to the VAE framework in \cite{4}. The full derivation of the HEBAE lower bound in Eq. (7) can be found in the appendix. Our framework also employs the multivariate reparameterization trick, which yields

$$z_i^{(l)} = \mu(x_i) + \sigma_i \odot \text{Re}^{(l)}, \quad \varepsilon \sim \mathcal{N}(0, I)$$

with $\Sigma = RR^\top$ derivred from the Cholesky decomposition of the covariance matrix, where $R$ is a lower triangular matrix with real and positive diagonal entries. The variational inference algorithm will impose that $R \to I$. In traditional VAEs, there are two key issues. First, $R = I$ exactly because of the assumed independence among the latent $z$. Second, there exists a conflicting issue in the framework where the standard normal priors will impose that $\sigma_{2}^2 \rightarrow 1$, while a typical reconstruction loss of the form $\|g(z) - x\|^2 = \|g(\mu(x) + \sigma \odot \varepsilon) - x\|^2$ will force these parameters to tend toward zero. As a result, there is a need to balance the weight between the reconstruction loss and the KL term in the VAE model. However, in the HEBAE framework, since we regularize according to $q_\phi(\mu(x_i))$ in Eq. (5) which does not depend on any variance component hyperparameters, thus the value of each $\sigma_{2}^2$ are fully dictated by the reconstruction.

### 3.3 Model Interpretations and Theoretical Comparisons

In this section, we give intuition behind why the HEBAE framework is better able to avoid posterior collapse and improve upon the performance of the WAE framework. Under the hierarchical model assumptions in Eqs. (5) and (6), the aggregated posterior is

$$q_\phi(z) = \int q_\phi(z | x_i)p_\theta(x_i) \, dx_i \approx \mathcal{N}(\beta, [1 + U(\sigma^2)]\Sigma)$$

where $U(\sigma^2)$ is an averaged estimate of the variance component over all $m$ samples in the training batch. The objective of Eq. (7) will impose that $\Sigma \rightarrow I$ and the reconstruction loss will force $U(\sigma^2)$ to go to small. In this optimal case, $q_\phi(z) \approx q_\phi(\mu(x_i))$ and converge to a standard normal distribution. The reason that we pursue this approximate aggregated posterior by incorporating variational inference instead of deterministic updates is that it provides a natural probabilistic way for sampling from the conditional posterior and injecting noise into the decoder so as to smooth out the latent space. More specifically, the variational inference framework enables the HEBAE to pursue a tighter bound than the WAE. As previously shown \cite{12}, the ELBO for the VAE can be rewritten as

$$\mathcal{L}_{\text{VAE}}(\theta, \phi; x_{1:m}) = \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda \sum_{i=1}^{m} \text{KL}(q_\phi(z) \| p_\theta(z))$$

$$- \left\{ m \log N - \sum_{i=1}^{m} \mathbb{E}_{q_\phi(z)} [\mathbb{H}[q_\phi(i | z_i)] \right\}$$

where the first term is the reconstruction; the second term is the KL divergence between the aggregated posterior and the standard normal prior; and the third term is the “index-code mutual information” where $\mathbb{H}[q_\phi(i | z_i)] = \mathbb{E}[\log q_\phi(i | z_i)]$ denotes the entropy of variational posterior over the sample indices in the training batch. In the WAE setting, the objective only has the reconstruction and a regularizer that minimizes the distance between the aggregated posterior and the standard normal distribution. Thus, as stated in the original paper, the difference between WAE and VAE can be viewed as dropping the mutual information term from the objective, which is effectively pursuing a “looser” lower bound. Because we maintain the variational inference set up in Eq. (7), our model lies between the traditional VAE and WAE. To see the advantages of HEBAE, we now provide the closed form of our objective Eq. (7) as the following

$$\mathcal{L}_{\text{HEBAE}}(\theta, \phi; x_{1:m}) \approx \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda \text{tr}(\Sigma) - k - \log |\Sigma| - \beta^T \beta.$$  

There are three main benefits from our choice to maximize the above. The first directly improves upon the VAE framework. Notice that the the sample specific variance components $\sigma_{2}^2$ do not influence the
KL divergence terms which can effectively help balance the trade-off between the reconstruction and regularized loss functions. By comparison, in traditional VAEs, these variance components will be pushed towards one which is against the goal of the reconstruction. This creates a conflict as one needs to creatively balance the objectives between the two loss functions. As discussed earlier, this benefit is also related to the goal in WAE that aims to match the aggregated posterior to standard normal prior and drop the mutual information from the objective. Indeed, previous work has shown that by accounting for the mutual information term in VAEs, one can prevent over-regularization and posterior collapse [11, 7].

The second benefit is that our hierarchical assumption with an empirical estimator enables direct penalization of the covariance matrix \( \Sigma \) to converge onto the identity matrix \( \mathbf{I} \) (see Eq. (11)) versus Eq. (4). Thus, our posterior is expected to be more consistent with the independence assumption compared to a traditional VAE.

The final benefit is based upon improving the deterministic approach in the WAE framework. It is clear that the mutual information is bounded between \([0, \log N]\). In WAE framework, the index \( i \) and latent variable \( z \) have a deterministic relationship such that \( q(i \mid z) \) is set to be a delta function. This will result in the entropy \( \mathbb{E}_{q_i(z)} \mathbb{E}[q_i(i \mid z)] = 0 \) and the mutual information is fixed at \( \log N \). Though this does help prevent over-regularization in autoencoders and improves performance when generating new data compared to VAEs, completely removing the mutual information from the objective is maximizing a looser bound on the marginal likelihood. Our framework, on the other hand, enables adaptive learning of each \( \sigma_i^2 \) based on the reconstruction. Since the resulting estimate of \( \sigma_i^2 \) in the HEBAE framework is not exactly 0, the index-code mutual information will be large enough to prevent over-regularization (i.e., close to \( \log N \)) but will simultaneously introduce enough uncertainty to \( q(i \mid z) \) such that we are able to sample and estimate a smooth latent space.

## 4 Experiments

In this section, we empirically assess our model against the traditional VAE and WAE frameworks. Our aim is to evaluate the following metrics: (i) whether our hierarchical framework can better balance the trade-off between the regularization and reconstruction losses than the traditional VAE; (ii) whether our variational posteriors better converge onto the standard normal prior assumptions; and (iii) whether our variational inference algorithm results in a decoder that can generate better quality samples than VAEs and WAEs.

![Figure 1](image-url)

Figure 1: HEBAE outperforms VAE and WAE on all three metrics measured. (a) Top row shows that the ELBO of HEBAE converges faster to a better optimum than VAE in all experiments with different latent dimension \( k \). Bottom row shows that HEBAE is less sensitive to different KL divergence weights (\( \lambda \)) while VAEs are susceptible to over-regularization. Results are based on the MNIST dataset. (b) Comparison of FID scores for HEBAE, VAE, and WAE on the CelebA dataset. HEBAE is less sensitive to \( \lambda \) and has the lowest FID score.
Figure 2: The estimated posterior of the HEBAE framework is more consistent with the standard normal prior compared to the VAE and WAE frameworks, in both MNIST and CelebA analyses. (a, b) Top row shows the absolute value of the variance-covariance matrices. Bottom row shows the correlation matrices. Results are based on MNIST dataset. (c, d) Averaged mutual information measurements: Maximal Information Coefficient (MIC) and Total Information Coefficient (TIC) [17] computed using index $i$ and each dimension of latent variable $z$. HEBAE maintains higher mutual information than VAEs, but slightly smaller than the WAE.

We evaluate each approach using two datasets: MNIST [18] and CelebA [19]. The MNIST dataset contains 60,000 training images and 10,000 test images, while CelebA contains 202,599 images in total. For MNIST, all models were trained using the same simple two dense layer architectures for both the encoder and decoder. We then carried out experiments with different $k$-dimensions for the latent variables $z$. Specifically, we examine the common values $k = \{2, 5, 10, 20\}$ and report the results for $k = 20$ in the main text. For the CelebA analyses, we adopted the convolution architectures from Tolstikhin et al. (2017) [3] for all the models with the common choice $k = 64$-dimensional space for the latent variables [3, 4]. All the models are trained with an Adam optimizer [20]. More details about architectures and training procedures could be found in Appendix B.

Since HEBAE utilizes variational inference, we can directly compare its maximized ELBO with the ELBO from a VAE. In Fig. [1], we plot these ELBOs against training epochs for different dimensionality of $k$ in the MNIST data and find that our model consistently converges to a higher ELBO faster than VAEs through all experiments. Next, we plot the maximized ELBO against various weight parameter values ($\lambda$) for the KL loss (see the bottom panels of Fig. [1]). VAEs are generally sensitive to the choice of the KL weight and can easily over-regularize which leads to a much lower ELBO. Our model, on the other hand, is much less sensitive. To further investigate model sensitivity to hyperparameter tuning, in the CelebA dataset, we vary either the KL weight (in the VAE and HEBAE) or the MMD weight (in the WAE) and plot the converged Fréchet Inception Distance (FID) introduced by [21] in Fig. [1]. As the weight parameter gets closer to 0, each model should behave like a regular autoencoder and performs worse in all three frameworks (i.e., have a higher FID score). Again, we can see that HEBAE is generally less sensitive to the regularize loss weight.

Next, we investigate how well the variational posterior distribution of the latent variables converges onto the standard normal within the context of each of these three frameworks. In Fig. [2] we plot the variance-covariance and correlation matrices at convergence for both the MNIST ($k = 20$) and CelebA ($k = 64$) datasets. Namely, at convergence, a successful model will have an independent covariance structure between the latent variables. We can see specific instances where the VAE experiences posterior collapse: this is illustrated on the top row of Fig. [2] where the variance terms on the diagonals reduce close to 0. This is a phenomenon that is not experienced in within the HEBAE and WAE frameworks. Altogether, Fig. [2] illustrates the point that matching the aggregate posterior
can remedy the over-regularization problem. Furthermore, the hierarchical structure in the HEBAE framework allows it to converge onto the target standard normal prior more consistently than VAE and WAE (i.e., see the second and third rows of Fig. 2). Specifically, from the correlation plot, we can directly see that our aggregated posteriors converge onto the independent assumption better than the other two approaches.

Lastly, we demonstrate how HEBAE benefits from turning the optimization goals of a WAE into a probabilistic model like a VAE. Remember that WAE attempt to maximize a “looser” lower bound (see discussion around Eq. (10)). In Fig. 2c and Fig. 2d, we assess the corresponding mutual information between the sample-index \(i\) and latent variables \(z\) at the convergence. As WAEs do not provide a closed-form ELBO, we follow previous work [17] and approximate mutual information using the average Maximal Information Coefficient (MIC) and Total Information Coefficient (TIC) for different dimensions of \(z\). Here, we see that the VAE has much smaller mutual information, while the WAE has the highest scores. The mutual information from the HEBAE sits between the VAE and WAE, respectively. Similar to previous work, Maintaining a high mutual information prevents HEBAE from over-regularization and posterior collapse [3, 7]. Also since we do this in a probabilistic framework, our model is able to sample posterior estimates and smooth out the latent space which improves sample qualities compared with WAE [12]. To see this, we assess the quality of samples generated by HEBAE. In Fig. 3 and Fig. 4, we provide qualitative comparisons using test reconstruction, test interpolation, and random generated samples for the CelebA data. Quantitatively, HEBAE shows to have lower FID scores (with the lowest score being 46). Results for MNIST can be found in Appendix C.

Figure 3: HEBAE produces qualitatively higher-quality images based on the CelebA dataset than the VAE and WAE frameworks. Results on MNIST can be found in the Appendix.

5 Conclusion

In this work, we present Hierarchical Empirical Bayes Autoencoder (HEBAE), a new framework for probabilistic generative modeling. Our theoretical work connects the probabilistic framework of the VAE and the deterministic objective of the WAE. We illustrated the trade-off between reconstruction of samples and regularization of the latent space in the VAE and have shown how HEBAE can prevent over-regularization. We further demonstrate that matching the posterior to a more general prior distribution avoids issues with posterior collapse. In experiments assessing mutual information, it is clear that the sampling mechanism and probabilistic priors of HEBAE yield a smoother latent space to connect the encoder and decoder. The hierarchical assumption present in HEBAE leads to less sensitivity to initial settings of hyperparameters and enables fast algorithmic convergence. It remains to disentangle the precise contribution of performance in our model between estimating an aggregated posterior and the form of the prior itself — which yields a latent space with orthogonal components.
that more strictly conform to mean-field assumptions. Finally, we validate our assumptions and theories with empirical experiments and show that HEBAE can generate higher quality images than its auto-encoder counterparts. GANs still remain state-of-the-art in generative modeling for images, and we expect insights from our probabilistic framework could be transferred into further improvements in GAN architecture.

**Broader Impact**

In this work, we propose a new method named HEBAE for generative modeling. We show that by combining the merits of probabilistic and deterministic autoencoders, we can develop an easily trainable method that generates higher quality data than the state-of-the-art. It is true that we could focus on the negative broader impacts of our work which might include generating more realistic fraudulent images and fake news stories. However, while we acknowledge these possibilities, we will instead focus on the positive. Naturally, HEBAE will benefit the many communities that are seeking methods for photograph generation, attribute interpolation, image-to-image translation, etc. However, our work also has the potential to be used in broader machine learning applications such as semi-supervised and disentanglement learning, as well as within other scientific fields such as natural language processing (NLP), robotics, and genomics. Indeed, one merit of our method is its ability to prevent over-regularization and posterior collapse in traditional variational autoencoders. For example, in NLP, the widely used autoregressive decoder is known to suffer from these same issues. The genomics community is one place that stand to gain great benefit from our new approach. Generating realistic artificial genomes can be used to improve human genome privacy issues and, in rare disease studies where data are far less prevalent, generative models can have a massive impact in helping us derive successful treatment strategies for future patients we have yet to observe. Furthermore, HEBAE is built based on an autoencoder structure which we showed can generate independent lower dimensional representations of data similar to non-linear principal component analysis (PCA); thus, one can also use our method for efficient dimension reduction purposed in studies that aim to analyze high-throughput sequencing assays.

**Acknowledgements**

This research was conducted using computational resources and services at the Center for Computation and Visualization (CCV), Brown University. This work was supported by grants from the US National Institutes of Health (R01 GM118652, P20GM109035, P20GM103645, and 2U10CA180794-06), the National Science Foundation (CAREER award DBI-1452622), and an Alfred P. Sloan Research Fellowship. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any of the funders or supporters.
Appendix

A Derivation of Theorem 1

In the main text, we show that the main advantage of the Hierarchical Empirical Bayes Autoencoder (HEBAE) framework is having the ability to find the ideal trade-off between the reconstruction error and regularization term in the traditional variational autoencoder (VAE) loss function. To do so, we propose maximizing the evidence lower bound (ELBO) with respect to the aggregated posterior distribution \( q_\phi(z_i) \). In other words, instead of regularizing each independent conditional posterior \( q_\phi(z_i | x_i) \) during training, HEBAE instead imposes that \( q_\phi(z) \) matches a standard normal distribution. This led to the following statement about the closed-form for the lower bound within the HEBAE framework.

**Theorem 1** Minimizing the Kullback-Leibler divergence \( KL(q_\phi(\mu(x_i)) \| N(0, I)) \) is equivalent choosing a general isotropic Gaussian as the prior distribution such that \( p_\theta(z_i) = N(\mu(x_i), \sigma^2 I) \) with the constraint that \( \beta^T \beta = 0 \). This yields the lower bound to optimize in the HEBAE framework,

\[
\mathcal{L}(\theta, \phi; x_{1:m}) = \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda_1 \sum_{i=1}^{m} KL(q_\phi(z_i | x_i) \| p_\theta(z_i)) + \lambda_2 ||\beta||^2
\]

where we use the following multivariate reparameterization trick

\[
z_i^{(l)} = \mu(x_i) + \sigma_i \odot R \varepsilon^{(l)}, \quad \varepsilon \sim N(0, I)
\]

with \( \Sigma = RR^T \) derived from the Cholesky decomposition of the covariance matrix between the latent z variables and R is a lower triangular matrix with real and positive diagonal entries. Notice that both Eq. (12) and Eq. (13) have the same reconstruction loss as the first term. Therefore, in this section, we will focus on the KL divergence terms and their relationship with the extra constraint. Under KKT conditions, the constraint in Eq. (12) can be achieved by incorporating an \( L_2 \)-penalty on \( \beta \) into the objective (15). One can interpret the weight \( \lambda_1 \) outside the KL term in Eq. (12) as a regularization parameter similar to the VAE framework (where \( \lambda_1 = 1 \) in the traditional model). Since both Eq. (12) and Eq. (13) has the reconstruction loss as the first term, then showing their equivalence simply amounts to deriving the relationship between the KL divergence terms and the extra constraint. To begin, we first restate the hierarchical variational family assumption within the HEBAE framework,

\[
q_\phi(z_i | x_i) \sim N(\mu(x_i), \sigma^2_i \Sigma), \quad q_\phi(\mu(x_i)) \sim N(\beta, \Sigma).
\]

Under this model, we can find a closed form expression for the KL term in the objective in Eq. (12)

\[
KL(q_\phi(z_i | x_i) \| p_\theta(z_i)) = KL(N(\mu(x_i), \sigma^2_i \Sigma) \| N(0, \sigma^2 I))
\]

Taking the KL divergence between two Gaussian distributions with some algebraic rearrangement and simplification yields

\[
KL(N(\mu(x_i), \sigma^2_i \Sigma) \| N(0, \sigma^2 I)) = \text{tr}(\Sigma) - k - \log|\Sigma|
\]

We can plug Eq. (16) into the objective in Eq. (12) to find

\[
\mathcal{L}(\theta, \phi; x_{1:m}) = \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda_1 \sum_{i=1}^{m} KL(q_\phi(z_i | x_i) \| p_\theta(z_i)) + \lambda_2 ||\beta||^2
\]

\[
= \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda_1 \sum_{i=1}^{m} \left[ \text{tr}(\Sigma) - k - \log|\Sigma| \right] + \lambda_2 ||\beta||^2
\]

\[
= \frac{1}{L} \sum_{l=1}^{L} \sum_{i=1}^{m} \log p_\theta(x_i | z_i^{(l)}) - \lambda_1 \sum_{i=1}^{m} \text{tr}(\Sigma) - k - \log|\Sigma| + \beta^T \beta
\]

In the settings where \( \lambda_1 = \lambda_2 \), then the objectives in Eq. (12) is equal to the objective in Eq. (13) which concludes the proof.
B Details on Experiments Setup

B.1 MNIST Dataset

We used the following simple dense architectures for the HEBAE, VAE and WAE models. Note that $k$ denotes the dimension for latent variable $z$ and $\text{FC}_{k \times p}$ represents the fully connected layer. Lastly, $p = 2$ for VAE and HEBAE as both need to output variance component terms, while $p = 1$ for WAE. The encoder architectures are then:

$$x \in \mathbb{R}^{784} \rightarrow \text{FC}_{784} \rightarrow \text{ReLU} \rightarrow \text{FC}_{800} \rightarrow \text{ReLU} \rightarrow \text{FC}_{k \times p}$$

For the architecture in the decoder, we use:

$$z \in \mathbb{R}^{k} \rightarrow \text{FC}_{800} \rightarrow \text{ReLU} \rightarrow \text{FC}_{800} \rightarrow \text{ReLU} \rightarrow \text{FC}_{784}$$

We used mini-batches with size = 128 and all the models were trained for 100 epochs. The default KL weight $\lambda$ is set to be 1 for VAEs except for the experiments used to generate the bottom row of Fig. 1(a) in the main text where we evaluated each method based on a grid of $\lambda$ values. For the WAE, we adopted a suggestion from Tolstikhin et al. (2017) [3] and used $\lambda = 10$ for the MMD penalty. We used the Adam optimizer [20] with an initial learning rate of 0.001 and then the learning rate decays at a rate of 0.995 with every epoch.

B.2 CelebA Dataset

For the CelebA analyses, we adopted the convolution architectures from Tolstikhin et al. (2017) [3, 4]. Similarly, images are also center cropped and resized to $64 \times 64$ resolution. Here, note that $\text{Conv}_n$ represents the convolution layer with $n$ filters and $\text{ConvT}_n$ represents the transpose convolution layer with $n$ filters. All convolution and transpose convolution layers have filter sizes of $5 \times 5$ with a stride of size 2, except for the last transpose convolution layer of the decoder which has a stride of size 1. Once again, $\text{FC}_{k \times p}$ denotes the fully connected layer where $p = 2$ for VAE and HEBAE and $p = 1$ for WAE. The encoder architectures are:

$$x \in \mathbb{R}^{64 \times 64 \times 3} \rightarrow \text{Conv}_{128} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{Conv}_{256} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{Conv}_{512} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{Conv}_{1024} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{FLATTEN} \rightarrow \text{FC}_{64 \times p}$$

The decoder architectures are:

$$z \in \mathbb{R}^{64} \rightarrow \text{FC}_{8 \times 8 \times 1024} \rightarrow \text{ConvT}_{512} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{ConvT}_{256} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{ConvT}_{128} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{ConvT}_1$$

Similar to the MNIST analyses, we used mini-batches of size = 100 and all the models are trained up to 100 epochs. We used a learning strategy from Tolstikhin et al. (2017) [3] where we set that the initial learning rate to be $10^{-4}$ and then was decreased it by a factor of 2 after 30 epochs, by a factor of 5 after 50 epochs, and by a 10 after 70 epochs. The choices of $\lambda$ are shown in main text Fig. 1(b).
C Sample Results of MNIST

Figure 4: HEBAE produces qualitatively higher-quality images than the VAE and WAE frameworks. Samples are generated with $k = 10$, $\lambda = 1$ for VAE and HEBAE, and $\lambda = 10$ for WAE.

References

[1] Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al.. Generative Adversarial Networks; 2014.

[2] Kingma DP, Welling M. Auto-Encoding Variational Bayes; 2013.

[3] Tolstikhin I, Bousquet O, Gelly S, Schoelkopf B. Wasserstein Auto-Encoders; 2017.

[4] Ghosh P, Sajjadi MSM, Vergari A, Black M, Schölkopf B. From Variational to Deterministic Autoencoders; 2019.

[5] Dai B, Wipf D. Diagnosing and Enhancing VAE Models; 2019.

[6] Bowman SR, Vilnis L, Vinyals O, Dai AM, Jozefowicz R, Bengio S. Generating Sentences from a Continuous Space; 2015.

[7] Phuong M, Welling M, Kushman N, Tomioka R, Nowozin S. The Mutual Autoencoder: Controlling Information in Latent Code Representations; 2018.
[8] Razavi A, van den Oord A, Poole B, Vinyals O. Preventing Posterior Collapse with delta-VAEs; 2019.

[9] Bauer M, Mnih A. Resampled Priors for Variational Autoencoders; 2018.

[10] van den Oord A, Vinyals O, Kavukcuoglu K. Neural Discrete Representation Learning; 2017.

[11] He J, Spokony D, Neubig G, Berg-Kirkpatrick T. Lagging Inference Networks and Posterior Collapse in Variational Autoencoders; 2019.

[12] Hoffman MD, Johnson MJ. Elbo surgery: yet another way to carve up the variational evidence lower bound; 2016.

[13] Higgins I, Matthey L, Pal A, Burgess C, Glorot X, Botvinick M, et al. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. Iclr. 2017;2(5):6.

[14] Makhzani A, Shlens J, Jaitly N, Goodfellow JJ. Adversarial Autoencoders. CoRR. 2015;abs/1511.05644. Available from: http://arxiv.org/abs/1511.05644

[15] Karush W. Minima of functions of several variables with inequalities as side constraints. M Sc Dissertation Dept of Mathematics, Univ of Chicago. 1939.

[16] Kuhn HW, Tucker AW. Nonlinear programming. In: Traces and emergence of nonlinear programming. Springer; 2014. p. 247–258.

[17] Albanese D, Filosi M, Visintainer R, Riccadonna S, Jurman G, Furlanello C. Minerva and minepy: a C engine for the MINE suite and its R, Python and MATLAB wrappers. Bioinformatics. 2013;29(3):407–408.

[18] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998;86(11):2278–2324.

[19] Liu Z, Luo P, Wang X, Tang X. Deep Learning Face Attributes in the Wild. In: Proceedings of International Conference on Computer Vision (ICCV); 2015.

[20] Kingma DP, Ba J. Adam: A Method for Stochastic Optimization; 2014.

[21] Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in neural information processing systems; 2017. p. 6626–6637.