BTVAE: A Contrastive Learning Approach Towards Expressive Latent Space in Variational Autoencoders

Ruiqi Xu, Yichen Wei, Brenna Yin

Abstract

Purpose: This project aims to design a variational autoencoder model that has simple architecture and more expressive latent space. The importance of our model is twofold, as it has a better generative power and requires no extra training resources compared to the baseline model.
Hypothesis: We propose a novel variational autoencoder model incorporating Barlow Twins loss into the optimization objective. We compare the performance of our model against the baseline variational autoencoder model and measure the quality of generated images of the two models with varying latent variable dimensions.
Evaluation Plan: We demonstrate the effectiveness of our model on MNIST, CIFAR-10, and CelebA datasets as they are widely used benchmarks in the machine learning community. They also have different characteristics and pose different challenges for VAE models. We measure the quality of generated images through both automatic metrics (FID) and human evaluations.
Conclusions: We present a simple and unique approach to increase the expressivity of the latent space in variational autoencoders. Through this project, we have learnt that sampling can easily become a training bottleneck for BDL models, as our initial model required many sampling steps and dramatically slowed the training process. We have also learnt the importance to understand the statistical meaning of our research ideas, as it helps identifying potential biases or limitations in our model and is essential for interpreting and evaluating our experiment results.

1 Introduction

Variational autoencoders (VAEs) (Kingma & Welling, 2014) are a popular class of likelihood-based generative models that have been widely used in tasks such as image generation (Brock et al., 2018; Razavi et al., 2019), music synthesis (Dhariwal et al., 2020), speech generation (Ping et al., 2019; Oord et al., 2016), and semi-supervised learning (Kingma et al., 2014; Izmailov et al., 2020). VAEs consist of an encoder network that maps input data to a latent space, and a decoder network that maps the latent space back to the original data space. The goal of VAEs is to learn a compact and meaningful latent representation of the data, such that the decoder can reconstruct the original data with high fidelity.

The expressiveness of the latent space in VAEs is often limited by the choice of the encoder architecture and the prior distribution over the latent variables. The current state-of-the-art VAEs (Vahdat & Kautz, 2020; Child, 2020) successfully increase the flexibility of the prior via hierarchical priors. Recent studies have also shown promising results by replacing the prior with energy-based models (Xiao et al., 2020), or training an additional binary classifier to produce a reweighting factor over the prior (Aneja et al., 2020). However, all the mentioned works either require running iterative MCMC during training or training additional models on top of VAEs, which are computationally expensive.

Lately, contrastive learning has been gaining increasing popularity in representation learning, especially for vision tasks (Oord et al., 2018; Frosst et al., 2019; Chen et al., 2020; Grill et al., 2020).

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Contrastive learning uses the principle of contrasting samples against each other to learn an embedding space in which similar sample pairs stay close to each other while dissimilar ones are far apart (Weng, 2021). Among the contrastive learning techniques, Barlow Twins (Zbontar et al., 2021a) feeds two distorted versions of samples into the same network to extract features and learns to make the cross-correlation matrix between these two groups of output features close to the identity.

In this work, we propose a novel VAE model based on a VAE and Barlow Twins (BTVAE) that combines the best of both. BTVAE calculates the Barlow Twins loss of the embedding of a ground truth input x_{gt} and the embedding of the reconstructed input x_{recon}, and incorporates the Barlow Twins loss into the evidence lower bound (ELBO) training objective of a VAE. The intuition behind our model is that a raw input and its reconstructed counterpart produced by a VAE should possess similar features, and can be viewed as distorted versions of each other. Using Barlow Twins will hence help a VAE learn better representations of inputs and lead to a more expressive latent space in VAE.

In the following sections, we first provide a brief overview of VAEs and Barlow Twins loss. We then describe our proposed model in detail, and present experimental results on MNIST, CIFAR-10, and CelebA datasets to show the effectiveness of our approach. Finally, we conclude with a discussion of the implications and future directions of our work.

2 Methods

2.1 Variational Autoencoders

We now re-introduce the baseline variational autoencoder model in the setting of image tasks (Pu et al., 2016). Let $x_i \in \mathbb{R}^{C\times H\times W}$ be the ith ground truth image where C is the number of channels, H is the height, and W is the width of the image. Let $z_i \in \mathbb{R}^D$ be the embedding vector for the ith image where D is the dimension of the embedding space.

VAE aims to learn a generative distribution $p(x, z) = p(z)p(x|z)$ where $p(z)$ is chosen to be a Gaussian prior distribution over the latent variable: $p(z) = \mathcal{N}(0, 1)$. The Likelihood and posterior are derived from our decoder and encoder models.

2.1.1 Decoder

The decoder part of our model is a Deep "Deconvolutional" Generative Model, made up of one fully connected layer and multiple transposed convolution layers (Zeiler et al., 2010).

Firstly, we define the convolution operation. For each element in the ith row and jth column in the result of the convolution, we have

$$(X * Y)_{(i,j)} = \sum_m \sum_n X_{(m,n)} Y_{(i+m,j+n)} \tag{1}$$

where the subscript (m, n) mean the mth row and the nth column.

The encoded vector first goes through a linear fully connected layer and gets unflattened into a matrix. Then it gets passed into the transposed convolution layers.

The transposed convolution layers use the same convolution operations as normal convolution layers, but with one key change: one transformation is done before the convolution operation, which enables the convolution to effectively upsample the image:

1. $n - 1$ Rows and Columns of columns are added around the image ("padding"), where n is the dimension of the kernel/weight matrix V.
2. Insert m rows/columns of 0's in between existing rows and columns, where m is the stride size of the convolution operation.

Namely, for each of the k-th 2D Convolution layer we define, recursively, that

$$u_i^k = \text{ConvTranspose}^k(u_i; V, b) = \sigma\left(b^k + \sum_{k=0}^{C-1} V^k * u_i^{k-1}\right)$$

where u_i^k is the encoded vector of ith image after k layers of operations, σ is the activation function, which is ReLU, and * is the convolution operation defined above. Here we define base case $u_i^0 = V^{\text{fc}} z_i + b^{\text{fc}}$ be the output of the initial fully connected linear layer.

Then we can compute

$$\hat{x_i} \leftarrow \text{d}(u_i; \phi) \quad \text{d}(u_i; V, b) = \text{ConvTranspose}^k(\sigma(V^{\text{fc}} z_i + b^{\text{fc}}); V, b)$$

Then we can define the likelihood of the operation to be

$$p(x_i|z_i) = \mathcal{N}(\hat{x_i}, \sigma_l)$$

where σ_l is a hyperparameter set as 1.

2.1.2 Encoder

Since the posterior

$$p(z_i|x_i) = \frac{p(z_i)p(x_i|z_i)}{\int_z p(z)p(x_i|z)dz}$$

is intractable, we need an alternative model that approximates this distribution.

We define the encoder architecture to be multiple layers of 2D Convolutional layers (Le Cun et al., 1989) with a final fully connected layer, that predicts both mean μ_p and std σ_p for the encoding layer.

For each of the k-th 2D Convolution layer we define, recursively, that

$$u_i^k = \text{Conv}^k(x_i; V, b) = \sigma\left(b^k + \sum_{k=0}^{C-1} V^k * u_i^{k-1}\right)$$

where u_i^k is the encoded vector of ith image after k layers of operations, σ is the activation function, which is ReLU, and * is the convolution operation defined above. Here we define $u_i^0 = x_i$.

The output is one linear layer applied on top of k layers of the convolution layers. That is,

$$\mu_p, \sigma_p \leftarrow \text{e}(z_i; \theta) \quad \text{e}(z_i; V, b) = \sigma(V^{\text{fc}}\text{Conv}^k(x_i; V, b) + b^{\text{fc}})$$

where V^{fc} and b^{fc} are the weights and biases for the final linear layer. μ_p and σ_p each have dimension D, the length of the latent vector. For the fully connected layer, we define the activation function to be the sigmoid function.

With that architecture, we can define the likelihood of the code given the data to be

$$q(z_i|x_i) = \mathcal{N}(\mu_p, I_C \sigma_p),$$

where μ_p and σ_p are the predicted elements from the Deep Convolutional Neural Network.

2.1.3 Training Objective

The training objective for the baseline Variational Autoencoder is to maximize the evidence lower bound (ELBO) $L_{VAE}(x)$ on the log-likelihood $\log p(x) \geq L_{VAE}(x)$, where

$$L_{VAE}(x) = \text{MSELoss}(x, \hat{x}) + \lambda_1 \, \mathbb{E}_{q(z|x)}[KL(q(z|x)||p(z))] \tag{2}$$

Note that here we replaced the negative Log-likelihood with Mean Squared Error since they only differ by some coefficients and a square root in the case of Gaussian Distribution. By using MSE, we were able to speed up the computation.

2.1.4 Hyperparameters

The hyperparameters, in this case, include the dimension of the latent vector, the structure of the encoding network (the decoding network is the same structure, but just in reverse), and the coefficient λ_1 for the KL divergence in the training objective.

In our experiment, we will only tune the dimension of the latent space due to constraints of time and computing power. We will do a grid search of different latent spaces.

2.2 Upgraded Model: Barlow Twins VAE

2.2.1 Barlow Twins

Neuroscientist H. Barlow's redundancy-reduction principle implies that human's perception of two distorted versions of a sample should be similar. Inspired by this principle, the Barlow Twins loss(Zbontar et al., 2021b) measures the difference between the empirical cross-correlation matrix $\mathcal{C} = z_{\mathrm{A}}^{\mathsf{T}} z_{\mathrm{B}} \in \mathbb{R}^{D \times D}$ of the embeddings. Given two sets of images x_A and $x_B \in \mathbb{R}^{N \times C \times H \times W}$ within the same data class and matrices of their corresponding embedding vectors z_A and $z_B \in \mathbb{R}^{N \times D}$ given by the encoder, we define the loss as:

$$\textsc{BarlowTwinsLoss}(z_A, z_B) = \sum_{0 \le i < D} (1 - \mathcal{C}_{ii})^2 + \delta \sum_{0 \le i < D} \sum_{0 \le j < D,\, j \ne i} \mathcal{C}_{ij}^2 \tag{3}$$

where D is the dimension of the embedding space, and $\mathcal{C}_{ij}$ stands for the elements in the empirical cross-correlation matrix between z_A and z_B.

Here, the first term will be minimized when the diagonal of the cross-correlation matrix is all 1, and the second term will be minimized when the non-diagonal elements of the matrix is all 0. δ is a non negative hyper-parameter to allow tuning in different situations where it might be more important to require the diagonal to be all 1 than for the non-diagonal elements to be zero, or vice versa.

By measuring this difference, the Barlow Twins loss is able to evaluate the similarity between the embedding vectors, and encourages different features of the embedding vectors to be less correlated. It encourages the embeddings to capture the invariance among different images in the same data class. In this project, we find it intuitive to apply Barlow Twins to VAE, as an input image and its reconstructed counterpart should possess similar attributes and can be viewed as distorted versions of each other which belong in the same data class. The embeddings in Barlow Twins just refer to the latent space, which can be easily obtained by feeding the images to the encoder in VAE. We describe the details of our model in the following section.

2.2.2 BTVAE (Barlow Twins VAE)

In this project, we integrate the Barlow Twins loss into the variational autoencoder architecture. Similar to using embedding vectors computed from two images from the same data class, we use the embedding vectors of a ground truth image and its reconstructed image.

Given a ground truth input image $\mathbf{x}_{\text{gt}}$, the workflow diagram of BTVAE is illustrated in Fig. 1

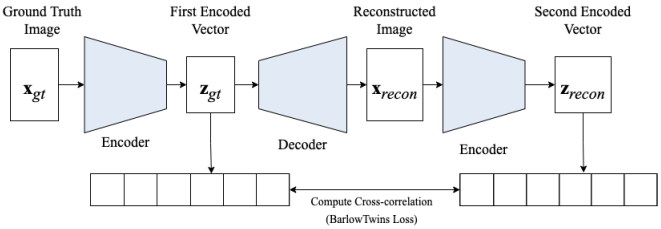

Figure 1: Workflow diagram of BTVAE.

where $\mathbf{z}_{\text{gt}}$ is the embedding vector for the input image, $\mathbf{x}_{\text{recon}}$ is the reconstructed image of $\mathbf{x}_{\text{gt}}$ and $\mathbf{z}_{\text{recon}}$ is the re-encoded embedding vector of the reconstructed image.

The first part of the training objective in our VAE-BT model retains the ELBO in Eq. 4. During training, BTVAE first runs $\mathbf{x}_{\text{gt}}$ through the encoder and gets the embedding vector $\mathbf{z}_{\text{gt}}$. Then, BTVAE derives the reconstructed image $\mathbf{x}_{\text{recon}}$ by passing $\mathbf{z}_{\text{gt}}$ to the decoder. Finally, BTVAE passes $\mathbf{x}_{\text{recon}}$ to the encoder to get the embedding vector $\mathbf{z}_{\text{recon}}$. This process constitutes the second part of the training objectives, which is to minimize the Barlow Twins loss between $\mathbf{z}_{\text{gt}}$ and $\mathbf{z}_{\text{recon}}$. Combining the two parts, the training objective becomes

$$L_{VAE}(x) = \text{MSELoss}(x, \hat{x}) + \lambda_1 \, \mathbb{E}_{q(z|x)}[KL(q(z|x)||p(z))] + \lambda_2 \cdot \textsc{BTLoss}(\mathbf{z}_{\text{gt}}, \mathbf{z}_{\text{recon}}). \tag{4}$$

where λ_1, λ_2 are a hyper-parameters. Here, λ_1 is not the focus of our experiment today and we fix it as 1 since it's the same for the base and the upgraded model, but λ_2 is an important hyper-parameter.

In practice, we find that the lambda value λ_2 is an important hyper-parameter to tune, with the best value to be 0.025 in most cases.

2.2.3 Hyperparameters

Our upgrade introduced two extra hyperparameters, the weight of the BarlowTwins Loss in the optimization target, λ_2, and the weight δ inside the BarlowTwins loss specifying the precedence of diagonal loss versus non-diagonal loss.

In our experiment, we will only tune the BarlowTwins Loss coefficient λ_2 and the dimension of the latent space due to constraints of time and computing power. We will tune the model through a grid search of different latent spaces.

3 Experiments

In this section, we examine BTVAE on several image datasets. We present the main quantitative results in Sec 3.1, qualitative results in Sec 3.2, and implementation details in Sec 3.3.

3.1 Main Quantitative Results

We examine BTVAE on MNIST (LeCun, 1998), CIFAR-10(Krizhevsky et al., 2009), and CelebA 64×64 (Liu et al., 2015) datasets. All the datasets are commonly used for evaluating likelihood-based generative models. Frechet Inception Distance (FID) (Heusel et al., 2017) with 10,000 samples is adopted as the quantitative metric to measure the sample quality. We fix the latent space dimension and encoder/decoder architecture for BTVAE and the baseline VAE to make the results are comparable with each other, and report the performances of BTVAE with different λ values.

Model	MNIST	CIFAR-10	CelebA
VAE	**0.0187**	0.3465	1.173
BTVAE, $\lambda = 0.00025$	0.0482	0.3478	1.137
BTVAE, $\lambda = 0.0025$	0.1984	0.3194	1.107
BTVAE, $\lambda = 0.025$	0.03245	0.2838	1.028
BTVAE, $\lambda = 0.25$	0.03286	**0.2235**	**0.9283**
BTVAE, $\lambda = -0.00025$	0.01939	0.3394	1.089

Table 1: FID results on MNIST, CIFAR-10, and CelebA. Lower FID stands for better sample quality.

Table 1 shows the FID results of BTVAE and the baseline VAE models on MNIST, CIFAR-10, and CelebA datasets. There are several things to note. First, the performance of BTVAE is sensitive to λ, and the relationship between λ and FID score is non-linear. This shows that λ is an important hyper-parameter that requires careful fine-tuning for BTVAE and indicates one limitation of this project that more experiments are required to provide a clear interpretation on the relationship between λ and sample quality. Secondly, BTVAE outperforms the baseline VAE on CIFAR-10 and CelebA datasets, but underperforms on MNIST. Thirdly, we tested the performance of BTVAE with $\lambda = -0.00025$ as a short ablation analysis to check the effectiveness of incorporating Barlow Twins loss. Table 1 supports our hypothesis that BTVAE with negative λ should have worse performances than those with positive λ.

3.2 Qualitative Results

3.2.1 Sample images

In this section, we show the qualitative samples in Fig. 2. It is notable that the sampled images on CIFAR-10 are of low-quality for both baseline VAE and BTVAE. This is potentially due to the fact that CIFAR-10 is more complicated and challenging compared to MNIST and CelebA, given that its data classes have very different attributes already, instead of just similar, hard-to-classify images like

Figure 2: Sample images on MNIST, CIFAR-10, and CelebA. The left side is the result of baseline VAE, and the right side is the result of BTVAE.

those in CelebA. One immediate future work of this project is to run the experiments on CIFAR-10 for baseline VAE and BTVAE with more powerful encoder/decoder architecture to get more reliable results.

While our model currently underperforms on the FID metric on the MNIST dataset, it is important to note that these metrics do not fully capture the perceptual quality of the generated images. On MNIST and CelebA, both baseline VAE and BTVAE seem to generate samples at similar levels of clarity. Furthermore, upon careful examination of the sample images on CelebA, we make the observation that BTVAE seems to produce images with more diverse attributes in terms of background, lighting, and the face's alleged skin color, gender, and hair styles. This supports our hypothesis that BTVAE learns a more expressive latent space compared to the baseline VAE model. To further validate our observation, we also conduct human evaluation that is discussed in detail in the following section.

3.2.2 Human Evaluation

We conduct human evaluation as a complement to automatic metrics and validation that BTVAE learns a more expressive latent space. We sample 144 examples on the CelebA dataset with the best performing baseline VAE and BTVAE, which are shown as the bottom two subfigures in Fig. 2. We then ask thirty annotators (ten per author) to evaluate the degree of diversity for the two sample groups. We ask the annotators to rank the two sample groups with the following prompt:

> "Please rank the two groups of sample images in terms of diversity. Diversity includes image attributes such as background and lighting, and face attributes such as alleged skin color, gender, hair style, and shape. Rank the more diverse group 1 and the other 2. If you think the two groups are equally diverse, rank both 1.5."

Table 2 shows the average ranking of VAE and BTVAE based on the responses from annotators. It supports our hypothesis from a qualitative perspective that BTVAE learns a more expressive latent space than the baseline VAE.

Model	Diversity
VAE	1.9
BTVAE	1.1

Table 2: Human Evaluation Metrics.

3.2.3 Embedding Space Visualization on MNIST

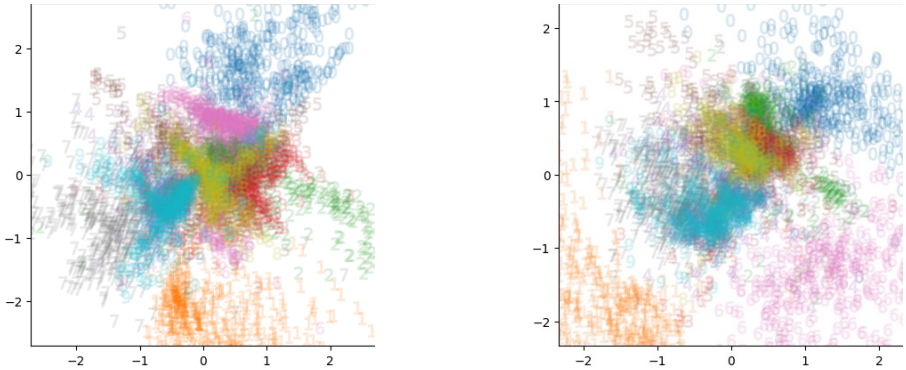

Figure 3: Visualization of the 2D latent space of VAE (left) and BTVAE (right). Axes correspond to normalized values of the latent variables.

To demonstrate that BTVAE learns a more expressive latent space, we train a BTVAE with 2D latent space on MNIST, as well as a corresponding baseline VAE with the same encoder/decoder architecture and latent space dimension. Fig. 3 plots the 2D latent space and shows certain number classes are clearly more separated from other classes by BTVAE, compared to the baseline VAE. We do not expect perfect separation as the latent space is fixed to be two-dimensional for display purposes and can not learn sufficient information about the inputs. We also note that the data distribution in latent space is more even out in BTVAE, which translates to fewer "white spots" in the latent space.

3.3 Implementation Details

We implement our model and experiments using PyTorch. We used Barlow Twins loss implementation released by the authors of the original paper at `https://github.com/facebookresearch/barlowtwins`. We built upon an existing VAE implementation located at `https://github.com/waymao/PyTorch-VAE`, and added the new VAE-Barlow Twins model to it.

For the encoder architecture of the model, we use five CNN layers with incremental dimensions 16, 32, 64, 128, and 256. We choose the kernel size to be 3, the stride size to be 2, and the padding to be 1. Each CNN layer is followed by a BatchNorm2d layer and a Leaky ReLu layer. We choose batch size to be 64 and use an Adam optimizer (Kingma & Ba, 2014) with fixed learning rate 0.005 and zero weight decay. For calculating the Barlow Twins loss, we fix the off-diagonal coefficient δ to be $5e-3$. During training, we monitor the validation loss and apply an early stopping with patience of 5. We set the maximum training epoch to be 50.

We implement the FID score with the standard practice of TorchMetrics, with a inceptionv3 feature layer size of 64. For visualizing the latent space on MNIST, we use the fourth assignment material provided for this course (Hughes, 2022).

3.4 Conclusion

In this project, we present BTVAE, a novel variational autoencoder model that incorporates Barlow Twins into the optimization target. We conduct comprehensive experiments on the sample quality and

latent space expressivity of our model against the baseline model on MNIST, CIFAR-10 and CelebA datasets. Both the quantitative and qualitative results indicate that BTVAE outperforms the original VAE model under most conditions. In the future, more experiment efforts are required to solidify the effectiveness of BTVAE as well as interpret the relationship between its hyper-parameters and model performance.

4 Reflection and Outlook

For short-term future work, we plan to use more advanced encoder/decoder architecture and repeat the FID experiment on CIFAR-10 to investigate why BTVAE currently underperforms on this dataset. we also plan to test the performance of BTVAE with different latent space dimensions. In the longer-term, we note that Barlow Twins is a generic learning model and can be integrated into any VAE architecture. We are interested in combining BTVAE with various VAE models to potentially create models with better performance.

This project has been a wonderful journey. At the start of this project, our initial model required many Monte Carlo processes and considerably slowed the training process. We have learnt that sampling can easily become a bottleneck for training, especially for BDL models. We have also learnt the importance to understand the statistical meaning of our research ideas. This helps us identify the potential biases or limitations in our model and is essential for interpreting and evaluating the experiment results.

References

Aneja, J., Schwing, A., Kautz, J., & Vahdat, A. (2020). *A contrastive learning approach for training variational autoencoder priors*. https://doi.org/10.48550/ARXIV.2010.02917

Brock, A., Donahue, J., & Simonyan, K. (2018). *Large scale gan training for high fidelity natural image synthesis*. https://doi.org/10.48550/ARXIV.1809.11096

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). *A simple framework for contrastive learning of visual representations*. https://doi.org/10.48550/ARXIV.2002.05709

Child, R. (2020). Very deep vaes generalize autoregressive models and can outperform them on images. *CoRR*, abs/2011.10650. https://arxiv.org/abs/2011.10650

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., & Sutskever, I. (2020). *Jukebox: A generative model for music*. https://doi.org/10.48550/ARXIV.2005.00341

Frosst, N., Papernot, N., & Hinton, G. (2019). *Analyzing and improving representations with the soft nearest neighbor loss*. https://doi.org/10.48550/ARXIV.1902.01889

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R., & Valko, M. (2020). *Bootstrap your own latent: A new approach to self-supervised learning*. https://doi.org/10.48550/ARXIV.2006.07733

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, volume 30. https://proceedings.neurips.cc/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf

Hughes, M. (2022). *Bayesian deep learning*. https://github.com/tufts-ml-courses/cs152BDL-22f-assignments

Izmailov, P., Kirichenko, P., Finzi, M., & Wilson, A. G. (2020). Semi-supervised learning with normalizing flows. *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 4615–4630. https://proceedings.mlr.press/v119/izmailov20a.html

Kingma, D. P. & Ba, J. (2014). *Adam: A method for stochastic optimization*. https://doi.org/10.48550/ARXIV.1412.6980

Kingma, D. P., Rezende, D. J., Mohamed, S., & Welling, M. (2014). *Semi-supervised learning with deep generative models*. https://doi.org/10.48550/ARXIV.1406.5298

Kingma, D. P. & Welling, M. (2014). Auto-Encoding Variational Bayes. *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images. https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

Le Cun, Y., Jackel, L. D., Boser, B., Denker, J. S., Graf, H. P., Guyon, I., Henderson, D., Howard, R. E., & Hubbard, W. (1989). Handwritten digit recognition: Applications of neural network chips and automatic learning. *IEEE Communications Magazine*, 27(11), 41–46.

LeCun, Y. (1998). The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. *Proceedings of International Conference on Computer Vision (ICCV)*.

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). *Wavenet: A generative model for raw audio*. https://doi.org/10.48550/ARXIV.1609.03499

Oord, A. v. d., Li, Y., & Vinyals, O. (2018). *Representation learning with contrastive predictive coding*. https://doi.org/10.48550/ARXIV.1807.03748

Ping, W., Peng, K., Zhao, K., & Song, Z. (2019). *Waveflow: A compact flow-based model for raw audio*. https://doi.org/10.48550/ARXIV.1912.01219

Pu, Y., Gan, Z., Henao, R., Yuan, X., Li, C., Stevens, A., & Carin, L. (2016). Variational autoencoder for deep learning of images, labels and captions. *Advances in neural information processing systems*, 29.

Razavi, A., van den Oord, A., & Vinyals, O. (2019). Generating diverse high-fidelity images with vq-vae-2. *Advances in Neural Information Processing Systems*, volume 32. https://proceedings.neurips.cc/paper/2019/file/5f8e2fa1718d1bbcadf1cd9c7a54fb8c-Paper.pdf

Vahdat, A. & Kautz, J. (2020). *Nvae: A deep hierarchical variational autoencoder*. https://doi.org/10.48550/ARXIV.2007.03898

Weng, L. (2021). Contrastive representation learning. *lilianweng.github.io*. https://lilianweng.github.io/posts/2021-05-31-contrastive/

Xiao, Z., Kreis, K., Kautz, J., & Vahdat, A. (2020). *Vaebm: A symbiosis between variational autoencoders and energy-based models*. https://doi.org/10.48550/ARXIV.2010.00654

Zbontar, J., Jing, L., Misra, I., LeCun, Y., & Deny, S. (2021a). *Barlow twins: Self-supervised learning via redundancy reduction*. https://doi.org/10.48550/ARXIV.2103.03230

Zbontar, J., Jing, L., Misra, I., LeCun, Y., & Deny, S. (2021b). Barlow twins: Self-supervised learning via redundancy reduction. *International Conference on Machine Learning*, 12310–12320. https://doi.org/10.48550/arXiv.2103.03230

Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. (2010). Deconvolutional networks. *2010 IEEE Computer Society Conference on computer vision and pattern recognition*, 2528–2535.