Contrastive Variational Autoencoders

Alexander Lin
Vanderbilt University
alexander.l.lin@vanderbilt.edu

Evelyn Zhang
Vanderbilt University
evelyn.zhang@vanderbilt.edu

Arnav Chahal
Vanderbilt University
arnav.chahal@vanderbilt.edu

Abstract

In this work, we explore the impact of incorporating SimCLR-style contrastive learning objectives into Variational Autoencoders (VAEs). We compare standard VAEs with Contrastive Variational Autoencoders (CVAEs) on the CIFAR-10 and MNIST image datasets, which are commonly used for classification tasks and are expected to exhibit a clustered structure in latent space. CVAEs aim to map similar inputs to nearby regions in the latent space, promoting more disentangled and meaningful representations. To assess the quality and structure of the learned embeddings, we use dimensionality reduction-based visualizations and evaluate the performance of a logistic regression classifier trained on the latent representations. We also investigate the effect of the KL divergence weight on the generative performance of both models. Ultimately, our goal is to understand the trade-offs and advantages of contrastive objectives in generative models, and how latent space structure can impact both generative fidelity and downstream discriminative performance.

1 Introduction

Deep generative models such as Variational Autoencoders (VAEs) have emerged as powerful tools for learning compact, meaningful representations of complex data. By combining reconstruction objectives with regularization via Kullback-Leibler (KL) divergence, VAEs aim to organize the latent space in a way that facilitates both data generation and structure-aware learning [1]. However, the latent representations learned by standard VAEs may not focus on learning discriminative features, which could their utility in downstream tasks such as classification or clustering.

Recent advances in self-supervised learning—particularly contrastive methods like SimCLR [2]—offer a potential approach for enhancing representation learning. These methods encourage embeddings of similar inputs to cluster together while pushing dissimilar ones apart, leading to more structured and separable latent spaces. In this work, we explore the integration of contrastive learning into the VAE framework, resulting in Contrastive Variational Autoencoders (CVAEs).

We conduct a comparative analysis of VAEs and CVAEs on the MNIST [3] and CIFAR-10 [4] datasets to investigate how adding a contrastive objective affects the structure and utility of the learned latent spaces. Our evaluation includes visual analysis via dimensionality reduction, reconstruction quality, and performance on a downstream logistic regression classifier trained on latent encodings. Furthermore, we consider how adjusting the KL divergence weight influences the generative capacity and representation quality in both models. Through this study, we aim to shed light on the trade-offs and potential benefits of incorporating contrastive objectives into the traditional VAE framework.

2 Methods

2.1 Data Curation and Processing

We evaluated our models on two standard benchmark datasets: MNIST and CIFAR-10. These datasets were selected to assess performance on both simpler (MNIST) and more complex (CIFAR-10) visual domains. Using these two datasets also allows us to assess the the impact of our methods across tasks of varying difficulty.

MNIST is a dataset of handwritten digit images, commonly used for benchmarking image classification and generative models. It contains 70,000 grayscale images of digits from 0 to 9, split into 60,000 training examples and 10,000 test examples. To enable model selection and early stopping, we further split the original training set into 90% training and 10% validation using a random split. Each image is 28×28 pixels in size with a single channel. The pixels were each divided by 256 to place them in the range [0,1].

CIFAR-10 presents a more visually complex dataset, consisting of 60,000 natural images spanning 10 object categories such as airplane, automobile, bird, cat, and more. Unlike MNIST's centered, high-contrast digits, CIFAR-10 images include significant variability in color, texture, and background, making the task of representation learning more challenging. The dataset is divided into 50,000 training images and 10,000 test images. A similar 90/10 training-validation split was also used for CIFAR-10. Each image is 32×32 pixels with three RGB color channels. The pixels were normalized against a mean of 0.5 and standard deviation of 0.5 to place them approximately in the range of [-1,1].

2.2 Variational Autoencoders

Variational Autoencoders (VAEs) extend traditional autoencoders by introducing a probabilistic framework for learning latent variables. While standard autoencoders map inputs to fixed latent codes, VAEs learn a distribution over the latent space, enforcing structure and building relationships between latent encodings. The distributions provide more information about the latent space and enable sampling, allowing VAEs to generate new, coherent data points by drawing from the learned latent space.

VAEs are typically implemented with an encoder-decoder structure. The encoder maps the input x to the parameters of the variational posterior $q_\phi(z|x)$, which typically include the mean μ and the log-variance $\log(\sigma^2)$ of a Gaussian distribution. The decoder them samples from this variational posterior to reconstruct the original data. The ultimate goal of VAEs is to approximate the intractable posterior $p(z|x)$ using a variational distribution $q_\phi(z|x)$. The optimization objective is to maximize the Evidence Lower Bound (ELBO):

$$\log p(x) \geq \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - \text{KL}\left(q_\phi(z|x) \,\|\, p(z)\right), \tag{1}$$

where:

- $q_\phi(z|x)$ is the variational posterior, parameterized by the encoder,
- $p_\theta(x|z)$ is the likelihood function, parameterized by the decoder,
- $p(z)$ is the prior distribution over the latent variables, $\mathcal{N}(0, I)$ for this study.
- KL represents the Kullback–Leibler (KL) divergence.

The ELBO combines two key components:

1. **Reconstruction Loss:** Encourages the decoder to accurately reconstruct x from z.
2. **KL Divergence:** Regularizes $q_\phi(z|x)$ to align with the prior $p(z)$.

The weight of the two components can be adjusted, motivating the model to prioritize either reconstruction or latent structure.

2.3 SimCLR

SimCLR is a self-supervised learning framework introduced by Chen et al. (2020) that learns useful latent representations without data labels or any enforcement of a prior. It implements contrastive learning by encouraging representations of augmented views of the same image to be similar, while pushing apart representations of different images. The goal of this formulation is for SimCLR to capture semantically meaningful features that can ultimately be used for downstream tasks.

SimCLR learns representations by maximizing the similarity between encodings of different augmented views of the same input using a contrastive loss. Each image x is transformed into two views x_i and x_j via random data augmentation. These are encoded into representations $h_i = f(x_i)$ and projected to $z_i = g(h_i)$ using a projection head. The training objective is the NT-Xent (Normalized Temperature-scaled Cross Entropy) loss, defined for a positive pair (i, j) as:

$$\ell_{i,j} = -\log \frac{\exp\left(\frac{\text{sim}(z_i, z_j)}{\tau}\right)}{\sum_{k=1}^{2N} 1_{[k \neq i]} \exp\left(\frac{\text{sim}(z_i, z_k)}{\tau}\right)} \tag{2}$$

- $\text{sim}(z_i, z_j) = \frac{z_i^\top z_j}{\|z_i\| \|z_j\|}$ is the cosine similarity
- τ is a temperature parameter, which controls how sharply the model distinguishes between similar and dissimilar pairs. A low τ encourages the model to favor the most similar pairs, while a high τ treats more pairs as equally similar.
- $2N$ is the total number of augmented examples in a batch (each of the N items is augmented once, doubling the batch size)

2.4 Contrastive Variational Autoencoders

The Contrastive Variational Autoencoder (CVAE) utilized in our study combines the prior regularization of VAEs with the representation learning strengths of contrastive learning introduced by SimCLR. Specifically, CVAEs enforce that representations of similar inputs (e.g. augmentations) remain close in latent space and dissimilar inputs are repelled. The goal is for the model to learn more semantically meaningful embeddings while maintaining generative capacity.

The CVAE architecture extends a convolutional VAE by introducing a contrastive objective over latent representations. The encoder maps input images x and their augmentations x_{aug} to latent means and variances, which are used to sample latent vectors z and z_{aug} via the reparameterization trick. These are decoded to reconstruct the input, and compared using MSE loss. The VAE loss $\mathcal{L}_{\text{CVAE}}$ is the same as Equation 1.

To encourage consistency between latent codes of augmented pairs, a contrastive loss is added. Latent vectors z and z_{aug} are first projected into a smaller space using a simple MLP, after which they are L2-normalized and concatenated. The cosine similarity matrix is computed and scaled by a temperature parameter τ. A cross-entropy loss is applied such that the positive pair (original and augmented) is pulled together while other pairs are pushed apart. The formula for the contrastive loss $\mathcal{L}_{\text{contrastive}}$ is the average of the losses in Equation 2 over the samples, where we fixed the temperature parameter (τ) to be 0.5.

The full loss becomes:

$$\mathcal{L}_{\text{CVAE}} = \mathcal{L}_{\text{VAE}} + \lambda \cdot \mathcal{L}_{\text{contrastive}}$$

where λ is manually modified to adjust the priority of the contrastive objective. This design enables learning latent representations that support both reconstruction and discriminative tasks.

2.5 Model Architecture

We adopt a consistent encoder–decoder model structure for both the VAE and the CVAE, introducing only the minimal changes required by each dataset and the contrastive objective.

2.5.1 MNIST Dataset

- **Input:** 28×28 grayscale images ($c = 1$).
- **Encoder:**
  - Four convolutional layers, each with 4×4 kernels, stride 2, padding 1.

  - Channel depths: $1 \to 8 \to 16 \to 32$.
  - Final feature map size: $32 \times 7 \times 7 = 1568$ units.
  - Two parallel linear layers map the 1568 units to a 10-dimensional mean and log-variance.
- **Decoder:**
  - Linear layer reshapes the 10-dimensional vector back to $32 \times 7 \times 7$.
  - Three transposed-conv layers mirror the encoder to reconstruct the 28×28 image.

2.5.2 CIFAR-10 Dataset

- **Input:** 32×32 RGB images ($c = 3$).
- **Encoder:**
  - Four convolutional layers with 4×4 kernels, stride 2, padding 1.
  - Channel depths: $3 \to 32 \to 64 \to 128 \to 256$.
  - Final feature map size: $256 \times 2 \times 2 = 1024$ units.
  - Two parallel linear layers map the 1024 units to a 128-dimensional mean and log-variance.
- **Decoder:**
  - Linear layer reshapes the 128-dimensional vector back to $256 \times 2 \times 2$.
  - Four transposed-conv layers mirror the encoder to reconstruct the 32×32 image.

VAE vs. CVAE The only architectural difference between the VAE and the CVAE is the small projection head appended to the latent sample in the CVAE. After sampling z, the CVAE passes it through a lightweight MLP (e.g. 10→8→5 for MNIST or 128→64→32 for CIFAR-10), normalizes the output, and applies the contrastive loss. All other layers, kernel sizes, and activation functions remain identical across both models.

The augmentations performed for the contrastive loss include a random slight resize crop, random 15 degree rotation, and random affine transformations for the MNIST dataset. For the CIFAR-10 dataset, we used random moderate resize crop, random horizontal flip, and color jitter.

2.6 Model Training

We split each dataset into training and validation sets, use minibatches (128 for MNIST, 256 for CIFAR-10), and run for a fixed number of epochs (700 on MNIST, 2000 on CIFAR-10).

All models are trained with the Adam optimizer (learning rate 1×10^{-4}) and use a combined reconstruction and Kullback–Leibler divergence loss. The CVAEs have an additional contrastive loss. To control the trade-off between reconstruction fidelity and latent regularization, we sweep the KLD weight β on MNIST over $\{10^{-3}, 10^{-5}, 10^{-8}\}$ and fix $\beta = 10^{-3}$ for the VAE and $\beta = 10^{-2}$ for the CVAE on CIFAR-10. The weight on the contrastive loss λ is set to 10^{-1} for all CVAEs.

Models are saved every 100 epochs to track performance over training and to enable later analysis of how different β, λ values influence convergence.

3 Experiments

3.1 Image Reconstruction

We first evaluated the model's ability to accurately reconstruct input images from their latent representations. By comparing original and reconstructed images, we assessed how well the encoder-decoder pipeline preserves key visual features. Additionally, we compare reconstruction quality between the standard VAE and the CVAE to assess how contrastive learning impacts fidelity of the reconstructed images.

MNIST

In Figure 1 and Figure 2, the top row displays the original image and the bottom row displays the images reconstructed from the latent space encodings. With both the VAE and CVAE we see that the

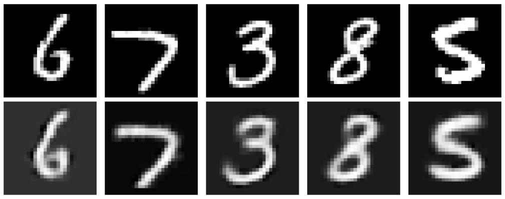

Figure 1: MNIST VAE Image Reconstruction

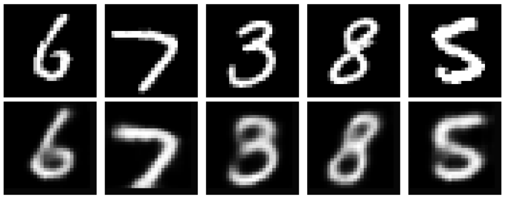

Figure 2: MNIST CVAE Image Reconstruction

latent space reconstructions are not only identifiable as their class, but also strongly resemble their original images, albeit somewhat blurred with grayer backgrounds.

CIFAR-10

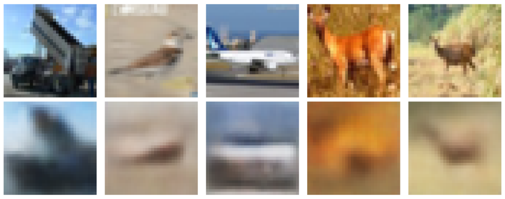

Figure 3: CIFAR VAE Image Reconstruction

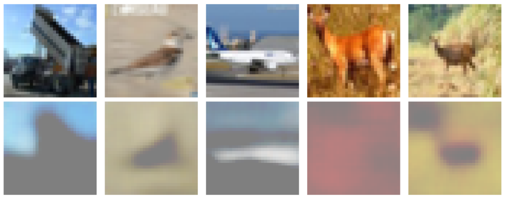

Figure 4: CIFAR CVAE Image Reconstruction

Similarly, in Figure 3 and Figure 4 the top row contains the original images and the bottom row the reconstructed. Compared to MNIST, the reconstructions for the CIFAR-10 dataset are visibly worse. The reconstructions by just the VAE appear to be a blurred version of the original images, while the CVAE only vaguely resembles the original images with the colors and locations of the features. We note that the CVAE has a lower weight on reconstruction in comparison to the reconstruction weight of the VAE because the CVAE needs to consider the contrastive loss in addition to the two losses already accounted for. This results in the poorer reconstructions that we observe.

3.2 Latent Space Visualization

We then visualized the 2D projections of latent embeddings using UMAP dimensionality reduction [5] to better understand how each model organizes the data. Comparing the VAE and contrastive VAE reveals differences in latent structure — for example, whether classes are more clearly clustered or separable. This helps assess how contrastive objectives influence the geometry of the learned space.

MNIST Latent Space

We can see from the MNIST latent space visualizations that the addition of a projection head and contrastive loss to the objective function did not greatly impact the separability within the latent space. For both types of models, similar looking digits (such as 3 and 8 or 4 and 9) would form clusters relatively close together, while other digits would be more spread out in the space. Since the MNIST discrimination problem is much simpler, we expect to see that both types of models can create semantically meaningful latent spaces.

For a simple dataset like MNIST, it was important to carefully consider the random augmentations applied to these images. Since certain modifications like flips and rotations can change one digit to another, the addition of certain augmentations could cause different digits to be mapped to similar places in the latent space.

CIFAR-10 Latent Space

Analyzing the CIFAR-10 latent spaces, we can see that the inclusion of contrastive loss adds separability to the latent space. The vanilla VAE produces a latent space that is jumbled, where there is little clustering in the UMAP. However, we can see that the UMAP for the CVAE shows more separation, where different classes migrate to defined sections of the plot. This suggests that our introduction of a contrastive loss regularization creates more discriminate and potentially informative latent spaces.

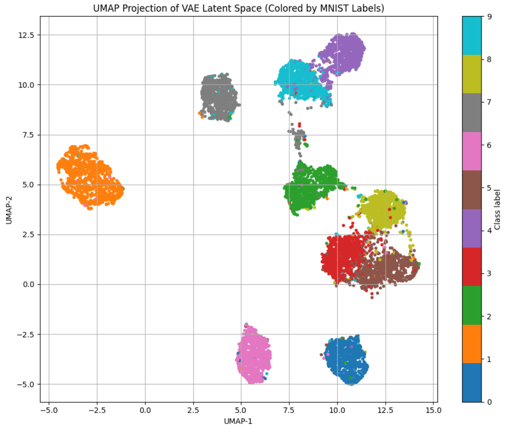

Figure 5: MNIST UMAP Projection of VAE Latent Space

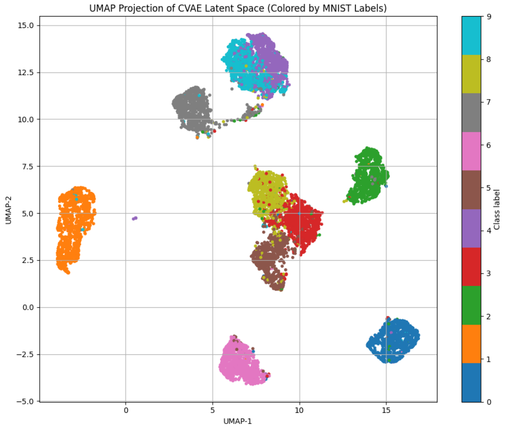

Figure 6: MNIST UMAP Projection of CVAE Latent Space

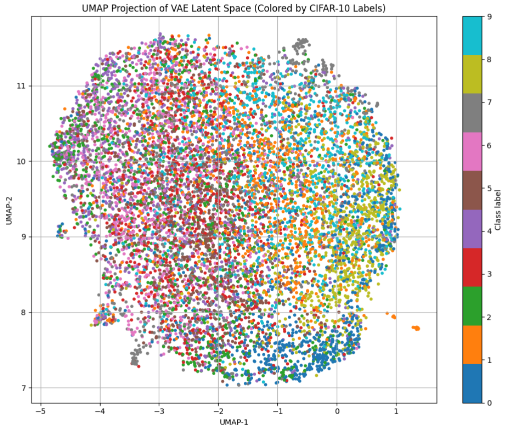

Figure 7: CIFAR-10 UMAP Projection of VAE Latent Space

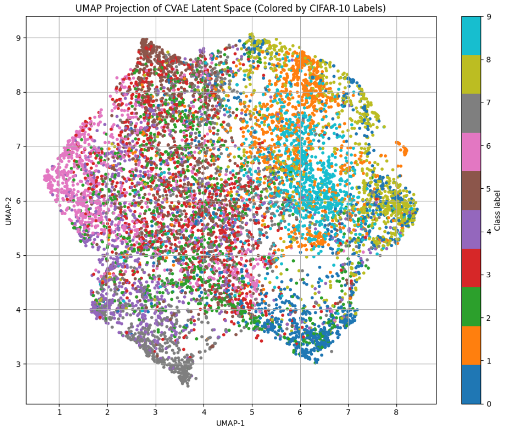

Figure 8: CIFAR-10 UMAP Projection of CVAE Latent Space

3.3 Latent Space Classification

To quantitatively evaluate the informativeness of the learned latent space, we trained a downstream logistic regression classifier on the latent vectors from each model. Higher classification accuracy would suggest that the latent space encodes more class-relevant information. This experiment compared how well VAEs and contrastive VAEs retain discriminative features after compression into latent encodings.

Dataset	VAE	CVAE
MNIST	91.80%	93.05%
CIFAR-10	38.05%	60.15%

Table 1: Comparison of VAE and CVAE latent space classification accuracy across MNIST and CIFAR datasets.

The classifier performed significantly better on the MNIST dataset than the CIFAR-10 dataset. This is because the handwritten digits are clear and distinct compared to the more abstract classes of CIFAR-10. This is also reflected in the clustering and separation of the latent spaces discussed in the previous experiment. Furthermore, the dataset for CIFAR-10 has 3 channels and 32x32 images, compared to 1 channel and 28x28 of the MNIST dataset, significantly increasing the amount of information the model needs to learn from the dataset. The MNIST digits are also high-contrast with solid black backgrounds, making the digits much more distinguishable compared to the objects in the CIFAR-10 dataset that vary in color and have noiser backgrounds.

For the MNIST dataset, we see that adding contrastive learning improves the classification accuracy by just 1.25%. The base performance of classification on MNIST latent encodings with just a VAE is already high at 91.8%, and accuracy is difficult to improve with a latent dimension of only 10. However, the CVAE still performs better as expected because forcing different classes apart using contrastive learning results in more distinctive latent encoding compared to those of the VAE.

For the CIFAR-10 dataset, we see that the classification accuracy is poor in comparison to the accuracy achieved on the MNIST dataset. This is partially influenced by the complexity of the CIFAR-10 dataset and limitations (see Section 3.5) when it comes to achieving a model with a representative latent space. This aligns with the results found in the previous section where MNIST latent representations are dispersed across the space in clusters, rather than the entangled class encodings in CIFAR-10.

In the CIFAR-10 latent space classification task, the CVAE shows a marked improvement over the VAE, achieving more than 50% higher accuracy (see Table 1). This performance boost reflects the CVAE's ability to produce considerably more distinct and separable latent encodings compared to those generated by the VAE. Contrastive loss provides a larger improvement for this dataset compared to the improvement between models for the MNIST dataset, since the model needs help distinguishing between different classes when the images themselves are more entangled. Again, this reflects what we observed in the previous experiment. The UMAP projection of the CVAE latent space for the CIFAR-10 dataset is visibly more separable than the UMAP project of the VAE latent space, confirming the results we achieve in this experiment with classification accuracy.

3.4 Image Generation

Finally, we assessed the generative capabilities of each model by sampling from the prior distribution and decoding to image space. This allowed us to compare the visual quality and diversity of samples produced by VAEs and CVAEs. We do this by comparing the impact of the KL Divergence weight on our image reconstruction. The KL divergence term in our error for a VAE helps us to enforce a prior distribution on our latent space, in this case a standard normal distribution. This then allows us to sample from our prior and generate new images. As such, we intended to analyze the impact of lowering a KL divergence term in a CVAE and a normal VAE. We use three values for the KL Divergence term our high KL term is $1 * 10^{-3}$, $1 * 10^{-5}$,$1 * 10^{-8}$. We hypothesize that a CVAE will be able to generate better images with a lower KL divergence term because we still enforce our contrastive loss term on our latent space. While we may not be strictly a normal distribution in our latent space, our generation still should be relatively strong because of our contrastive loss.

The FID scores for each model at three settings are shown in Table2. FID scores are a commonly used metric to evaluate the quality of generated images [6]. As β decreases (we decrease the KL penalty), sample quality degrades for both models, indicated by higher FID. Contrary to our hypothesis, the standard VAE consistently outperforms the CVAE in FID across all β values. For comparison, the FID scores for the original data for MNIST were around 10.

Model	High (β)	Mid (β)	Low (β)
VAE	48.06	87.04	199.75
CVAE	79.51	162.71	225.79

Table 2: FID scores for VAE and CVAE models at different KL-weight (β) settings.

From Table 2, we see a clear trend: as the KL weight β decreases, both VAE and CVAE models produce higher (worse) FID scores, indicating that relaxing the KL penalty degrades sample quality. The plain VAE achieves the best performance at all three β settings (48.06 → 87.04 → 199.75), whereas the CVAE lags behind (79.51 → 162.71 → 225.79). This outcome runs counter to our hypothesis that the contrastive term would bolster generation when β is small; instead, the CVAE's additional projection head and contrastive loss appear to disrupt the match between the approximate posterior and the Gaussian prior, leading to poorer sampling.

One possible explanation is that enforcing contrastive structure in the latent space conflicts with the VAE's requirement for an standard Gaussian latent distribution—particularly when the KL penalty is weak. As β is lowered, the VAE itself begins to ignore the prior and focuses on reconstruction, yet its simpler latent pathway still supports reasonable sampling. In contrast, the CVAE's dual objectives pull the latent representations in two directions, which may harm diversity and fidelity of generated images.

Additionally, the addition of the contrastive loss term can increase clustering within our latent space, as shown in our previous results. This could potentially lead to holes in our latent space that do not correspond to any particular class of data. Sampling from these sections to generate novel images would lead to the poor quality that we observed in CVAE FID scores.

3.5 Limitations

The main bottleneck in our study was our lack of processing power, resulting in a long training time and inability to train complex models. We did not have access to GPUs and trained all of our models on personal CPUs.

Our choice of running 700 epochs on MNIST and 2000 epochs on CIFAR-10 required multiple hours for each run, and our experimental setup required multiple runs for each model (in order to sweep the KLD weight β). This also limited the number of experiments we were able to run (with a variety of combinations of hyperparameters), and decreased the number of chances we had to tune the hyperparameters (ex: learning rate, layers, normalization method, channel depths, kernel sizes, loss weight) to achieve a well-performing model.

Our insufficient computational power also prevented us from being able to implement complex model architectures that are typically used for image classification tasks, such as networks with a larger number of parameters and layers, or ResNets. This would likely have taken us weeks to train each model, and we would not have been able to obtain clear experimental results in the allotted time.

The contrastive loss computation in the CVAEs added additional overhead. This computation effectively doubles the batch sizes, and requires more forward/backward passes with the projection head. The increase in memory usage and runtime further limited our ability to experiment on top of the long runtime we already experienced from the VAEs.

Finally, this also limited the scope of our experiment. MNIST and CIFAR-10 achieve significantly different results in our experiments, and testing our methods on a greater variety of datasets would help us better understand how different characteristics of datasets affect the impact contrastive loss has on model quality.

4 Conclusion

In this study, we investigated the impact of incorporating a SimCLR-style contrastive learning objective into the Variational Autoencoder (VAE) framework, resulting in a Contrastive VAE (CVAE). Our goal was to understand how this added objective influences both the generative and discriminative qualities of the learned latent space. Through experiments on the MNIST and CIFAR-10 datasets, we performed a comparative analysis between the standard VAE and the CVAE, evaluating both image reconstruction and generation quality, as well as the structure and utility of the latent representations.

Our results reveal a clear trade-off introduced by the contrastive loss. On one hand, the CVAE exhibits a significant decrease in reconstruction accuracy and a deterioration in the visual fidelity of generated samples compared to the standard VAE. These findings suggest that the contrastive objective interferes with the model's ability to retain fine-grained information, which is critical for high-quality generation. This is also reflected in higher FID scores for the CVAE, indicating a larger divergence from the real image distribution.

On the other hand, the latent space learned by the CVAE demonstrates improved structure and separability. Logistic regression classifiers trained on the CVAE's latent embeddings outperformed those trained on VAE embeddings, indicating that the representations are more informative and better aligned with class semantics. This improvement suggests that the contrastive objective encourages the encoder to prioritize features that are discriminative and relevant to the underlying data distribution, even at the expense of accuracy in reconstruction.

These observations emphasize the trade-off between generative and discriminative goals in representation learning. While the VAE aims to capture all factors of variation in a dense and continuous latent space optimized for reconstruction and generation, the addition of a contrastive loss shifts the focus toward learning features that separate similar and dissimilar inputs. This shift enhances the utility of the latent codes for tasks like classification or clustering, but degrades their capacity to support image synthesis.

In summary, adding contrastive learning to VAEs can be a powerful way to promote meaningful and task-relevant latent structures, but it requires careful balancing when generative performance is also a priority. Future work could explore multi-stage training (e.g., contrastive pretraining followed by generative fine-tuning) or dynamic weighting of the contrastive loss to better balance this trade-off.

5 References

[1] Kingma, D. P., & Welling, M. (2022). Auto-Encoding Variational Bayes (arXiv:1312.6114). arXiv. https://doi.org/10.48550/arXiv.1312.6114

[2] Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In International conference on machine learning (pp. 1597-1607). PmLR.

[3] Li Deng. (2012). The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. IEEE Signal Processing Magazine, 29(6), 141–142. https://doi.org/10.1109/MSP.2012.2211477

[4] Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images.

[5] McInnes, L., Healy, J., & Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426.

[6] 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, 30.

6 Code Availability

Code for this project is available at `https://github.com/allx2100/contrastive-encoders`.