Dauphine | PSL
UNIVERSITÉ PARIS

Project 2 : Learning latent space representations

Generative model : Variational Autoencoders (Group: MH_VAE)

Authors: Yassine Abou Hadid, Ismail El Hadrami, Mohammed Hlel

November 27, 2022

Keywords:
VAE, VQ-VAE, VAE-GAN

Conducted under the supervision of:

- Benjamin Negrevergne
- Alexandre Vérine

1. Introduction

1.1. Context and problem statement

In machine learning, there are mainly two sub-categories of models: discriminative learning models and generative learning models.
Most of the models used are discriminative. Whether for classification, segmentation or regression, these are machine learning models whose basic principle is to associate an input record (x) with an output value (y). For example, a model where an input image x is classified by the model into a predetermined class (of cats, dogs, vehicles, etc.). Regardless of the learning technique (using an artificial neural network or a decision tree), the operating principle of discriminative models is the same: associate an input variable with a specific output label.
Generative models represent a separate category. Contrary to discriminative models which associate input values to output labels, their objective is to generate new data according to specific rules and conditions.
The goal of this project is to become familiar with state of the art generative models and compare them, for this purpose. To achieve this task we will use the models to generate images from the dataset MNIST, and we will evaluate the quality of the generated images with the Fréchet Inception Distance.

1.2. Overview of the project

Our project can be divided in three steps :

- Implementation and training of the generative models
- Implementation of the FID score
- Benchmark of the different models implemented

2. Solution & Implementation

2.1. CVAE

The variational autoencoder or VAE is a directed graphical generative model which has obtained excellent results and is among the state-of-the-art approaches to generative modeling. It assumes that the data is generated by some random process, involving an unobserved continuous random variable z. it is assumed that the z is generated from some prior distribution P(z) and the data is generated from some condition distribution P(X—Z), where X represents that data. The z is sometimes called the hidden representation of data X. We use convolutional layers since it is more adapted to image problems. Like any other autoencoder architecture, it has an encoder and a decoder. The encoder part tries to learn the hidden representation of data X or encoding the X into the hidden representation (probabilistic encoder). The decoder part tries to learn the decoding of the hidden representation to input space. The model is trained to minimize the objective function: The first term in this loss is the reconstruction error or expected negative log-likelihood of the data point. The expectation is taken with respect to the encoder's distribution over the representations by taking a few samples. This term encourages the decoder to learn to reconstruct the data when using samples from the latent distribution. A large error indicates the decoder is unable to reconstruct the data.

$$\underbrace{-\mathbb{E}_{\mathbf{z}\sim q(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x}|\mathbf{z})\right]}_{\text{reconstruction error}} + \underbrace{\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right)}_{\text{regularization}}$$

Figure 2.1: VAE-loss function

The second term is the Kullback-Leibler divergence between the encoder's distribution and p(z). This divergence measures how much information is lost when using q to represent a prior over z and encourages its values to be Gaussian.

2.2. VQ-VAE

The Vector Quantised-Variational AutoEncoder (VQ-VAE) [1], differs from VAEs in two key ways: the encoder network outputs discrete, rather than continuous, codes; and the prior is learnt rather than static.
Vector quantisation (VQ) is a method to map K-dimensional vectors into a finite set of "code" vectors. The process is very much similar to KNN algorithm. The optimal centroid code vector that a sample should be mapped to is the one with minimum Euclidean distance.

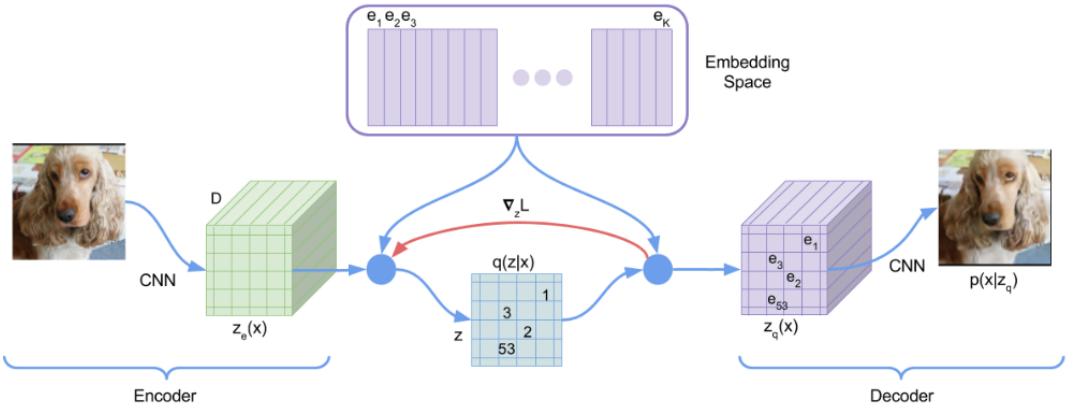


Figure 2.2: VQ-VAE architecture

$$L = \log p\left(x \mid z_q(x)\right) + \left\|\text{sg}\left[z_e(x)\right] - e\right\|_2^2 + \beta \left\|z_e(x) - \text{sg}[e]\right\|_2^2 \tag{2.1}$$

where sg[.] is the stop_gradient operator.

The loss function described in equation 2.1 has three components that are used to train different parts of VQ-VAE:

- **The reconstruction loss** : optimizes the decoder and the encoder.
- **The codebook loss** : since gradients bypass the embedding, we use a dictionary learning algorithm which uses an $l2$ error to move the embedding vectors e towards the encoder output
- **The commitment loss** : since the volume of the embedding space is dimensionless, it can grow arbitrarily if the embeddings e do not train as fast as the encoder parameters, and thus we add a commitment loss to make sure that the encoder commits to an embedding.

We remark that for the loss function the Kullback-Leibler (KL) term is absent while it usually appears in the ELBO, it can be shown that the KL divergence term from the VAE loss becomes a constant and therefore can be dropped for VQ-VAE (Since we assume a uniform prior for z)

2.3. VAE-GAN

While a VAE learns to encode the given input and then reconstructs it by the encoder, a GAN aims to generate new data which can't be distinguished from real data. The VAE-GAN model [2] was introduced for simultaneously learning to encode, generating and comparing dataset samples.The motivation behind this combined model is to replace the VAE reconstruction error term (expected log likelihood) with a reconstruction error expressed in the GAN discriminator. Since both the decoder of a VAE and generator of a GAN operate on the latent space z to produce the image x, a decoder is used instead of a generator.

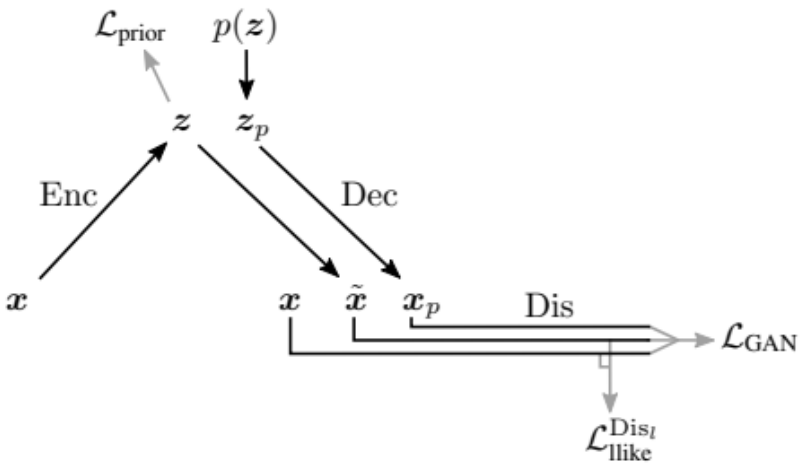


Figure 2.3: VAE-GAN architecture

The combined model is trained with the triple criterion:

$$\mathcal{L} = \mathcal{L}_{\text{prior}} + \mathcal{L}^{\text{Dis}_l}_{\text{llike}} + \mathcal{L}_{\text{GAN}}$$

Figure 2.4: VAE-GAN loss function

Where the GAN model loss is defined as:

$$\mathcal{L}_{\text{GAN}} = \log(\text{Dis}(\boldsymbol{x})) + \log(1 - \text{Dis}(\text{Gen}(\boldsymbol{z})))$$

Figure 2.5: GAN loss function

3. Results

3.1. Fréchet Inception Distance (FID)

It is one of the most popular metrics for measuring the distance between feature points in real and generated images. The Fréchet distance is a measure of similarity between several curves that takes into account the location and order of points along these curves. It can also be used to measure the distance between two distributions.

Mathematically, the Fréchet Distance is used to calculate the distance between two "multivariate" normal distributions. For a "univariate" normal distribution, the Fréchet Distance is given as,

$$d(X, Y) = (\mu_X - \mu_Y)^2 + (\sigma_X - \sigma_Y)^2 \tag{3.1}$$

where μ and σ represent the mean and standard deviation of normal distributions, respectively, and X and Y represent two normal distributions.

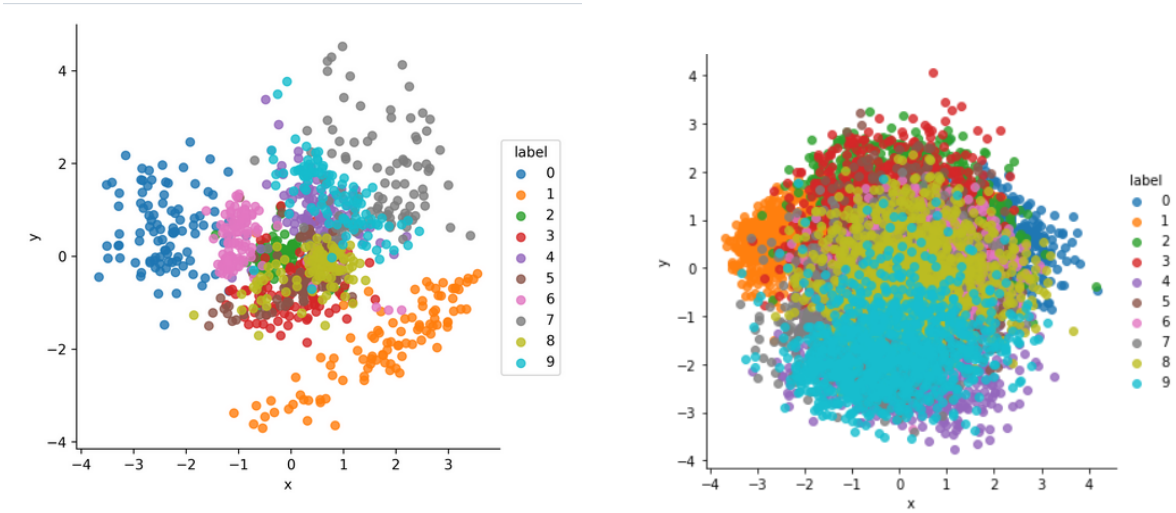
In the context of computer vision, in particular evaluation of generative models, we use the feature distance as described above. We use the pre-trained Inception V3 model. Using the activations of the Inception V3 model to summarize each image gives this score its name "Fréchet Inception Distance"[3]. This activation is taken from the penultimate layer.

Here on the following table, we summarize the results we got during this project :

Model	FID
Real Data	0.005
CVAE	33.11
DeepConv GAN	36.18
VAE-GAN	26.45
Random Gaussian noise	83.06

3.2. Reconstruction & Generation

When we study the representations of the latent spaces of the VAE and the CVAE in figure 3.1, we notice that the CVAE is much more compact, which makes it possible to have better generations when performing a random sampling compared to the VAE which has more chance of computing a garbage output.



(a) VAE (b) CVAE

Figure 3.1: Representation of the latent space

The reason the FID value of the VQ-VAE is not shown in the previous table is that the VQ-VAE is not really suitable for image generation, but more for image reconstruction due to its uniform prior. However, other methods exist to perform image generation using VQ-VAE and a PixelCNN to learn the priors on the discrete latents for image sampling [4]. However, this method has not been implemented due to lack of time. In order to evaluate our VQ-VAE, we only study the reconstruction, and we notice that the CVAE has a better reconstruction.
However, this evaluation is not really fair because the VQ-VAE has fewer parameters than the CVAE. Moreover, according to the community evaluations, the VQ-VAE associated with a PixelCNN is supposed to outperform a CVAE on the MNIST and CIFAR10 datasets

Figure 3.2: Real Data

(a) VQ-VAE Reconstruction (b) CVAE Reconstruction

4. Conclusion

We have taken a journey through the state of the art of generative modeling research, starting out with the basic ideas behind autoencoders, GANs, and building upon these foundations to understand what state-of-the-art models such as CVAE, VQ-VAE and VAE-GAN architectures are capable of achieving. A possible next step could be to test the presented methods on different, perhaps more complex datasets in order to clearly draw the limits of each model.

Bibliography

[1] A. v. d. Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," 2017.

[2] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," 2015.

[3] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," 2017.

[4] A. Razavi, A. v. d. Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," 2019.