Question 1. 1.3 :

a : Here is a reconstruction image from the test set for the VAE model :

FIGURE 1 – Reconstruction image from the test set for the VAE model

Here are some samples from the VAE model at the end of training :

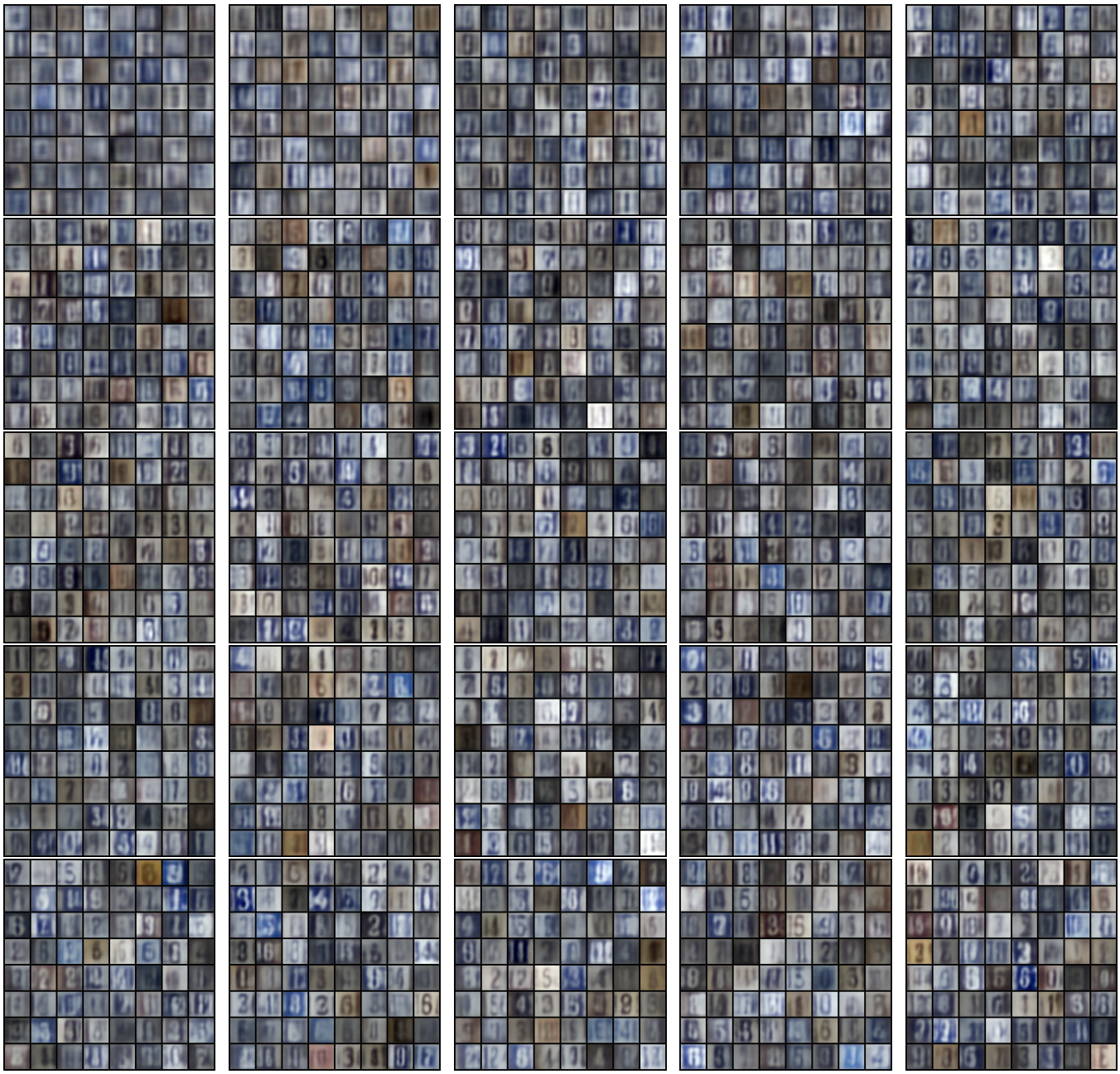

FIGURE 2 – Samples from VAE model at the end of the training

b : The generated samples from the VAE model are very blurry, especially the samples from the first few epochs. As the models is being trained for more epochs, the generated images are becoming less blurry and digit patterns can be seen. It's also worth mentioning that the generated samples are much more blurred than those coming from GANs.

c : Here are images for the interpolation for the VAE model :

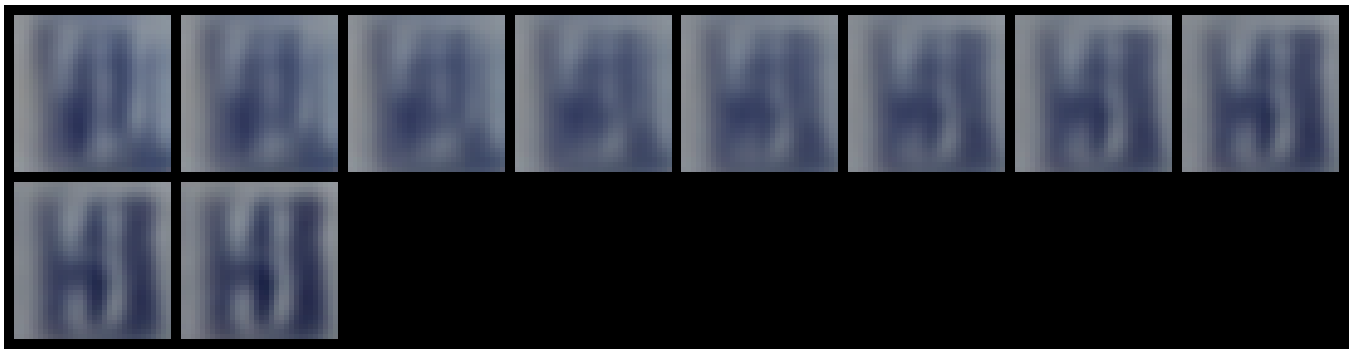

FIGURE 3 – Interpolation images for the VAE model

Yes, as can be seen from the figure above, the transition is smooth between z_1 and z_2 and images are changing smoothly between these two points.

Question 2. 2.2 :

a :

Here are some samples from the GAN model at the end of training :

FIGURE 4 – Samples from GAN model at the end of the training

b :

The quality of the image samples generated by the GAN model is of high quality and compared to the VAE model, the images are better in quality and are a lot less blurred.

c :

Here are images for the interpolation result between two points from the noise distribution :

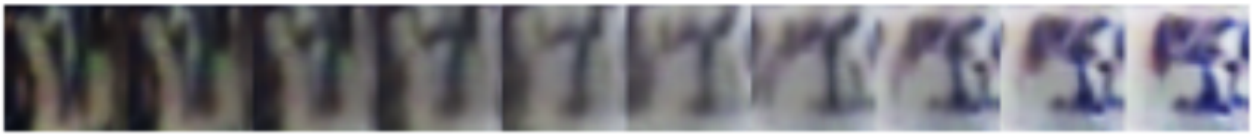

FIGURE 5 – Interpolation images result between two points from the noise distribution for the GAN model

Yes, as can be seen from the figure above, there is a smooth transition between the two images z_1 and z_2.

2.3 :

Yes it is essential to use the detach() method in the discriminator training for the fake generated images.

The detach() method in PyTorch returns a new tensor that does not require a gradient by separating a tensor from the computational graph.

When calculating the discriminator loss, for the fake generated images, the detach() method is used. The reason is that you want to only compute gradients for the discriminator and not the generator. The detach() method here allows you to make sure this happens.

You do not use detach() match when computing generator loss. If you do it for the generator loss, then the generator loss will not contribute to the generator gradient which is not what you want

2.4 :

a) No the GAN model is not good for reconstructing input images. The reason is that for reconstructing images, it's better to have an encoder-decoder based model like VAE, and GAN models are not based on that

b) No, the GAN model is not good for computing the log-likelihood. As opposed to VAE models, the GAN model is not based on optimizing log-likelihood, so they can not approximate or compute log-likelihood correctly.

c) Yes, GAN models are very good at learning representations, and tend to generate much more high quality samples.

Question 3. 2.2 :

a :

Here are some samples from the DDPM model at the end of training :

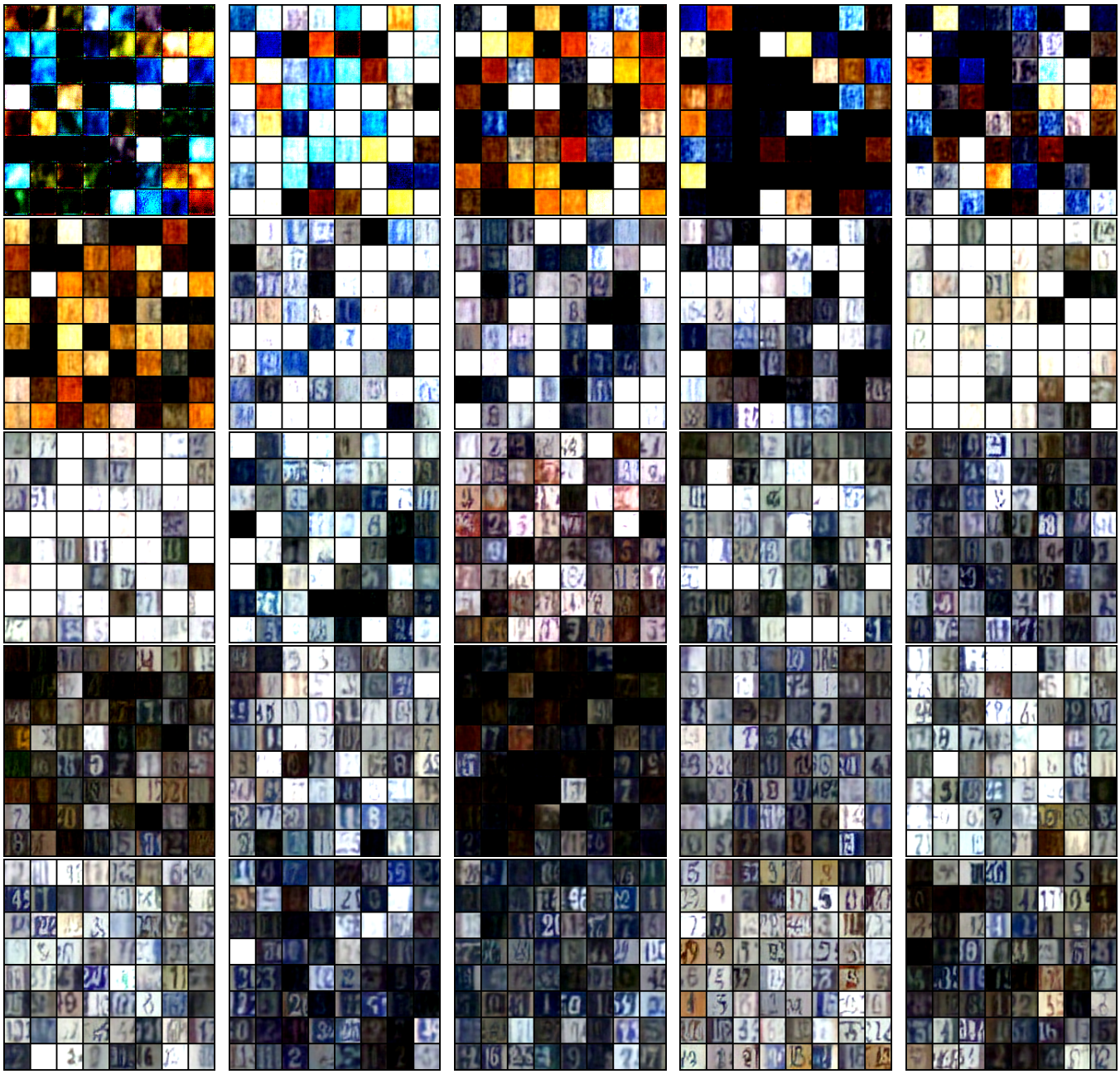

FIGURE 6 – Samples from DDPM model at the end of the training

b :

The samples from the DDPM model have quite good quality. They are better than the VAE Samples (since they were so blurry) but not as good the GAN samples.

Question 4. VAE models :

A draw-back for the VAE models is that the generated samples are much more burred than GAN or DDPM models because of the injected noise and imperfect reconstruction and the decoder ar-

chitecture with factorized distribution.

A strength for the VAEs is that they provide a latent representation, which can be very useful to do things such as imputation and completion.

Another strength for VAEs is that they are based on optimizing the log-likelihood, so there is a clear and recognized way to evaluate the quality of the model.

The computation load for the training and inference of the VAEs are not high compared to GAN and DDPM models

GAN models

GAN models do not have a clear objective like the VAEs to optimize and their training can be unstable. GANs, though powerful, are hard to train (especially compared to VAEs) and have higher computational load at training and inference. However, the quality of the samples generated by GANs are high.

DDPM models

As DDPM models require simulating a markov chain for many steps to produce a sample, they can generate high quality samples. However, their computational load for training and inference is very high compared to GAN models and VAE models.