# Exercise Sheet 7: Variational Autoencoders

Due on 14.06.2024, 10:00

Felix Krause (felix.krause@lmu.de)

**Important Notes:**

1. **Questions:** Please use the Moodle platform and post your questions to the forum. They will be answered by us or your fellow students.

2. **Submission:** Your submission should consist of one single ZIP file which includes a PDF file and the corresponding codes. Both the PDF file and ZIP file should contain your surname and your matriculation number (Surname-MatriculationNumber.zip) for grading purposes. You may use Jupyter 1 for exporting your python notebooks as PDF, but you still have to hand in your .ipynb or .py files for us to test your code. For this exercise, please export the .ipynb as a PDF file and include that in the ZIP file. Submissions that fail to follow the naming convention or missed PDF/code files will not be graded.

3. **Deadline:** The due date for this exercise is the 14th of June.

**Task 1: Training a Variational Autoencoder on MNIST (10P)**

You are going to train a VAE on the MNIST dataset. When training a VAE on a dataset with classes, the encoder network not only receives a data point as input, but also the corresponding class label. Similarly, the decoder not only receives a latent variable as input, but also the corresponding class label. This is illustrated in Fig. 1. The loss function does not change when using class labels, so you will still use the following loss:

$$L(x^{(i)}, \theta, \phi) = -\mathbb{E}z \sim q_\phi(z|x^{(i)})[\log p_\theta(x^{(i)}|z)] + D_{KL}(q_\phi(z|x^{(i)}) \parallel p_\theta(z)) \quad (1)$$

To get you started, we included an implementation of a simple VAE network in PyTorch (see network.py). You can either use it, change it, or create your own implementation. Using the provided implementation will not affect your points. Your report should include:

1. Write PyTorch code for the loss function as well as a complete training routine for your model.

2. Plot a sample for each class after every epoch.

3. Plot the training curve (number of steps on the x-axis and the loss on the y-axis)

Hints:

- You should balance the two terms of the loss function by multiplying the KL term with something like 0.0001.
- Use torch.optim.Adam with a learning rate of around 0.001.
- You should see good results after around 5 training epochs.

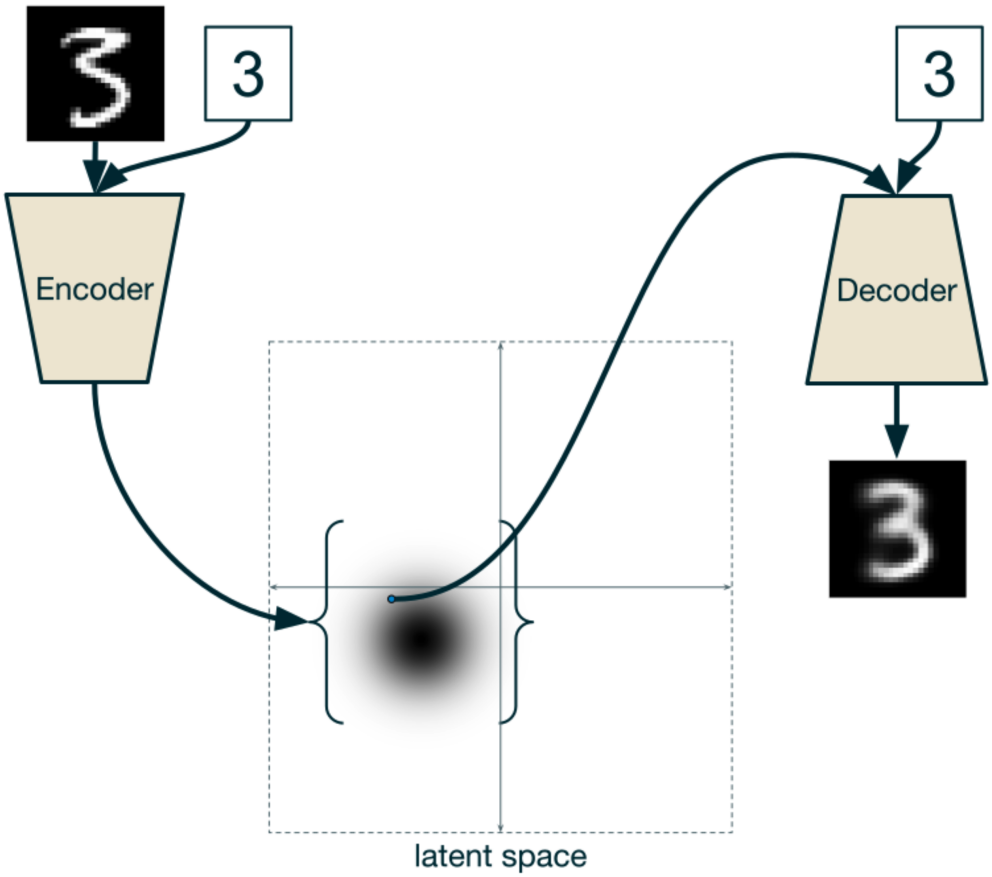


Figure 1: Variational Autoencoder with classes. Source: ttps://ijdykeman.github.io/assets/cvae_figures/cvae_diagram.svg

**Task 2: Vsiaulize the latent space** **(6P)**

The loss function in Eq. 1 consists of two terms. The first term maximizes the likelihood of the original input being reconstructed and the second term penalizes any difference between the posterior modeled by the encoder network and the prior. To get some insight into the contribution of both terms, you are going to visualize the latent space in 2D.

1. Train your VAE again using only the first term of Eq. 1 (the reconstruction loss). Set the dimension of the latent variable $z$ to 2.
2. Train your VAE again using only the second term of Eq. 1 (the KL divergence). Set the dimension of the latent variable $z$ to 2.
3. Train your VAE again using only the full loss function. Set the dimension of the latent variable $z$ to 2.
4. For all three models, embed the MNIST test set into the latent space (for each image you should get a 2D vector) and create a scatter plot of all embedded images in the test set. All points in the scatter plot should be colored according to the corresponding class (all 0 digits have one color, all 1 digits have another color, etc). You should make three separate scatter plots for each model.

**Task 3: Anomaly Detection using a Variational Autoencder** **(4P)**

Anomaly detection identifies data points that deviate significantly from the norm. This can be applied to detect fraud, system failures, or rare occurrences in general. In this task we will use A VAE to do anomaly detection, however, this is not a special trait of VAEs but can be done using normal AEs as well.
A VAE trained on a dataset containing only object $A$ will have a higher reconstruction loss when given samples containing object $B$ and can therefore be used for anomaly detection. For this task, use the VAE you have trained for the longest.

1. Download the Fashion MNIST dataset
2. Use samples from MNIST handwritten digit dataset and the MNIST. fashion dataset to showcase the difference in reconstruction loss. Plot 10 samples of each dataset and their corresponding reconstruction loss.