Exercise 5

Deadline: 25.06.2019

In this exercise we will implement a variational autoencoder (VAE) and a conditional variational autoencoder (CVAE). We train both type of models on the MNIST dataset, visualize them and use a CVAE as a generative classifier on the MNIST dataset. We provide a Jupyter notebook `cvae.ipynb` on Moodle which contains code stubs for all tasks. Please fill in the missing code at the marked places.

Regulations

Hand-in the completed Jupyter notebook `cvae.ipynb` along with its exported version `cvae.html` and zip both files into a single archive with naming convention (sorted alphabetically by last names)

`lastname1-firstname1_lastname2-firstname2_exercise05.zip`

or (if you work in a team of three)

`lastname1-firstname1_lastname2-firstname2_lastname3-firstname3_exercise05.zip`

and upload it to Moodle before the given deadline. We will give zero points if your zip-file does not conform to the naming convention.

1 VAE and CVAE (15 points)

Implement the CVAE class and the negated ELBO-loss in the Jupyter notebook. Detailed instructions can be found in the notebook. Check whether your implementation works by comparing the reconstruction of the VAE/CVAE with its input. Furthermore, sample latent codes z and use your decoder in order to generate samples from the data distribution. Evaluate experimentally how the number of latent dimensions influences the quality of the reconstruction and generated samples samples.

2 Visualisation VAE (10 points)

Visualization of the latent space is easiest when it has just two dimensions. Use this possibility to create two types of plots:

- Visualize the relationship between latent code positions (e.g. placed on a grid) and the corresponding decoder outputs, i.e. the generated images (see example below).
- Plot embeddings of different training instances in the latent space. Make sure to visualize their labels as well.

Design your plots such that they convey interesting information about the latent space structure. Give an interpretation of the visualizations with regard to this structure (2 points). Can you tell from the plot in which cases the VAE might fail to generate outputs that look like real data? Give a short explanation of possible failure cases and plot a few examples (2 points).

Ullrich Köthe
ullrich.koethe@iwr.uni-heidelberg.de

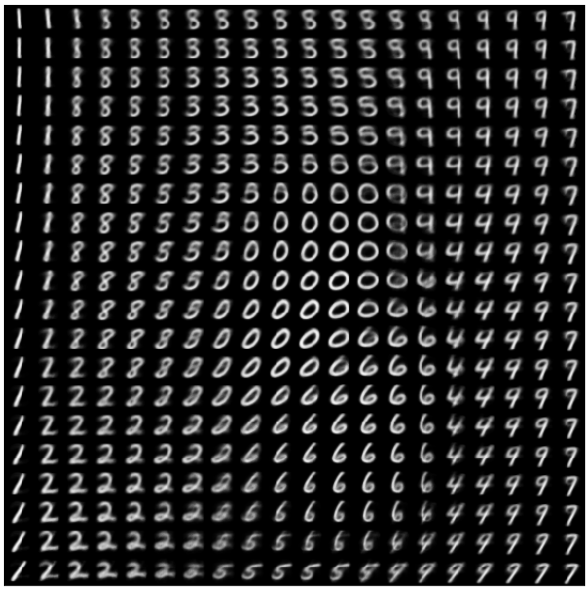

Abbildung 1: Example of how latent space can be illustrated. Output of VAE decoder for different locations of the latent space.

3 Visualization CVAE (5 points)

This task is similar to task 2, but we now use the CVAE with a 2 dimensional latent space. Visualize the relationship between latent code positions and corresponding decoder outputs (see example below) for three different conditions (=digits). Make sure your plot conveys interesting information about the latent space. What latent space structure do you observe (e.g. encoded styles)?

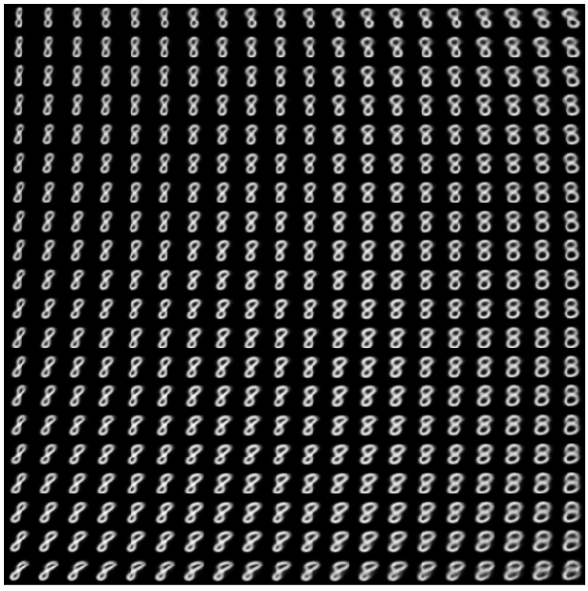

Abbildung 2: Example of how latent space can be illustrated. Output of CVAE decoder, conditioned on label 8, for different locations of the latent space.

Furthermore, plot embeddings of different training instances in the latent space for three conditions/digits. What can you say about the difference between the CVAE and VAE encodings?

4 Classification (10 points)

Train a CVAE on all classes as a generative model (or reuse the CVAE already learned above). Use this model as a generative classifier on MNIST. Determine the test set accuracy of your classifier as a function of the latent space dimension.