Télécom Physique Strasbourg - Université de Strasbourg - SDIA/SDSC 2A

TP2: Unsupervised Deep Learning Visualisation

1 Introduction

Objectives of the TP: feature embedding and deep clustering.

We will use Python with TensorFlow ($\geq$ version 2.0) to define and train the models, scikit-learn for embedding, and Matplotlib for visualisation, and datasets to easily load the data.

You will visualise the embeddings of the autoencoders that you implemented in the last TP.

2 Feature Distribution Visualisation

Download and open `tp2_skeleton.py`, understand the existing code.

2.1 MNIST Distributions

Use scikit-learn's `TSNE` and `PCA` functions to complete the `get_tsne_pca` function in order to plot the distributions of the MNIST dataset (using the defined `tsne_perplexity`) in 3D, it is best to use the subsampled version in `X_test_fl`.

2.2 SAE MNIST Distributions

Next copy the SAE from Part 2.2 of TP1 into the relevant place, and

- create an additional `Model` (named `encoder_model`) that has the encoding (middle) layer, i.e. the layer with 32 neurons, as its output (keep the original model as it is needed for training – `encoder_model` is only a references to the relevant layers of the existing network object).
- Train the SAE as in TP1 (using the original model) and use the `encoder_model`'s predict function to get the embedded representation of the MNIST dataset.
- Use `get_tsne_pca()` to plot their distributions.

2.3 Extra

If you have time (or are interested further), here are some more tasks:

- Repeat Part 1 with the other autoencoders from TP1 and compare their distributions.
- Play around with perplexity in Part 1 and relate what you see to https://distill.pub/2016/misread-tsne/.