Hands-on / Programming Question

In this assignment, your will implement and train a Gaussian VAE, and run experiments using the MNIST datasets (containing $28 * 28$ gray-scale hand written digit images). (100')

Tasks

1) Implement the `forward()` function for the `ConvnetBlock` found in `models.py`
2) Train a Gaussian VAE to minimize the negative evidence lower-bound:

$$\mathcal{L}(x;\theta,\varphi) = \mathop{\mathbb{E}}_{z\sim q_\varphi(\cdot|x)}\left[\frac{1}{2\sigma^2}\left\|x - g_\theta(z)\right\|^2\right] + D\left(q_\varphi(z \mid x)\|r(z)\right)$$

Implement this lower bound as `gaussian_elbo` in `losses.py` using the closed-form expression for the KL divergence $D(q(z \mid x)\|p(z)) = \frac{1}{2}\sum_{i=1}^{k}\left(\sigma_i^2(x) + \mu(x)_i^2 - 1 - \log\sigma_i^2(x)\right)$. Report visual samples from your optimized model.

Notes

You can download the datasets and extract them to the `data/` directory in the root of this repository). Code for loading and processing this data into minibatches is provided in `mnist.py`.

Scaffolding for the VAE model is provided in `models.py`, and you will implement the loss function for Gaussian VAE's in `losses.py`.

Framework code for training your model is found in `gaussian_vae.ipynb`. It is recommended to use Google Colab or Kaggle to execute these notebooks (with a GPU accelerator attached). For debugging, you may find it helpful to modify the default hyper-parameters to build smaller models that are faster to train.