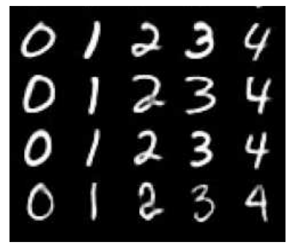
Neural Networks for Images - Exercise #2

Submission Date - 14/4/2022

Programing Task: Auto-Encoding and transfer-learning over the MNISTdigit dataset



The MNIST dataset consists of 60,000 (+10,000 test) images of scanned hand-written digits (0-9). The dataset contains the digits values as labels. The original images are of size 28-by-28 pixels (but better be resized to 32x32 using zero padding for easier encoding-decoding). The images are monochromatic, i.e., have a single brightness channel.

In this exercise we will explore aspects of auto-encoding and its uses for dimensionality reduction and transfer-learning.

Practical Tasks:

1. **Auto-Encoding**. Define a convolutional autoencoder to encode (and decode) the images into a small dimensional latent space d (around d=10) containing no spatial structure (1x1xd tensors). Explore the reconstruction error over the test set when (i): using low and high latent space dimension d, and (ii) using a fixed d yet different architecture with more and less layers (/weights). Report these tests and the best score obtained (i.e. plot your results as a function of the different experiments performed). The best practice of implementing this code is by defining an Encoder and a Decoder as separate Pytorch modules.

2. **Interpolation.** Once you have trained the AE, you can use it to interpolate between two digits in latent space. That is, let I_1 and I_2 be two *different* digits, perform the interpolation $D((E(I_1) * \alpha) + (E(I_2) * (1 - \alpha)))$ for $a = 0 \ldots 1$ where D denotes the decoder and E the encoder. (i) Include the gradual images obtained. (ii) Try different pairs of digits. (iii) Try repeating this operation with an AE trained using a higher embedding dimension $d \sim 20$. (iv) Which is better? Provide an explanation why the quality increased or decreased.

3. **Decorrelation.** In the following experiment we will investigate the connection between dimensional reduction and dependencies (/redundancies) in the representation. Carry

this out by computing the [Pearson correlations] between different coordinates of the latent codes (based on a few thousands encoded images), and use them to come up with a single value (that you came up with!) for measuring the overall correlation. Plot this value with respect to the latent space dimension d (over at least 4 values of d). Explain your choices and the trend in correlation versus d that you observe. Provide an explanation in your report.

4. **Transfer Learning**. Use a pre-trained encoder as a fixed network (i.e. excuse its weights from the following training optimization), and attach it to a small MLP (in latent space) and train only the latter to classify the digits (recall that MNIST is labeled). Do this with a small fraction of the labels in the dataset (~tens of images). Compare the performance of this solution next to the option of training the entire network (i.e., allow the encoder weights to train as well as the classification MLP) over this small number of training examples. Note that both training schemes used the same number of labels. Which option operated best? Report all the choices made (e.g., latent space dimension, MLP arch., and number of labels used, etc.), and the results obtained by each of the two training approaches.

We are expecting you to report and elaborate on every practical task in the PDF, using your own words and analysis of what you've done. Include everything that you think is crucial for us to understand your way of thinking.

Theoretical Questions:

1. Show that the composition of linear functions is a linear function. Show that the composition of affine transformations remains an affine function.

2. The calculus behind the Gradient Descent method:
   a. What is the stopping condition of this iterative scheme,

   $$\theta^{n+1} = \theta^n - \alpha \nabla f_{\theta^n}(x)$$

   b. Use the second-order multivariate Taylor theorem,

$$f(x+dx) = f(x) + \nabla f(x) \cdot dx + dx^T \cdot H(x) \cdot dx + O(\|dx\|^3),$$
$$H_{ij}(x) = \frac{\partial^2 f}{\partial x_i \partial x_j}(x)$$

to derive the conditions for classifying a stationary point as local maximum or minimum.

3. Assume the network is required to predict an angle (0-360 degrees). How will you define a prediction loss that accounts for the circularity of this quantity, i.e., the loss between 2 and 360 is not 358, but 2 (since 0 is 360..). Write your answer in a pytorch-like pseudo-code.

4. Chain Rule. Differentiate the following terms (specify the points of evaluation of each function):
   a.
   $$\frac{\partial}{\partial x} f(x+y, 2x, z)$$
   b.
   $$f_1\Big(f_2\big(...f_n(x)\big)\Big)$$
   c.
   $$f_1\Big(x, f_2\big(x, f_3(...f_{n-1}(x, f_n(x)))\big)\Big)$$
   d.
   $$f\Big(x + g\big(x + h(x)\big)\Big)$$

Submission Guidelines:

The submission is in **<u>singles</u>.** Please submit a single zip file named “ex2_ID.zip”. This file should contain your code, along with an ”ex2.pdf” file which should contain your answers to the theoretical part as well as the figures/text for the practical part. Furthermore, include in this compressed file a README with your name, ID, and CSE username.
Please write readable code, with documentation where needed, as the code will also be checked manually.