

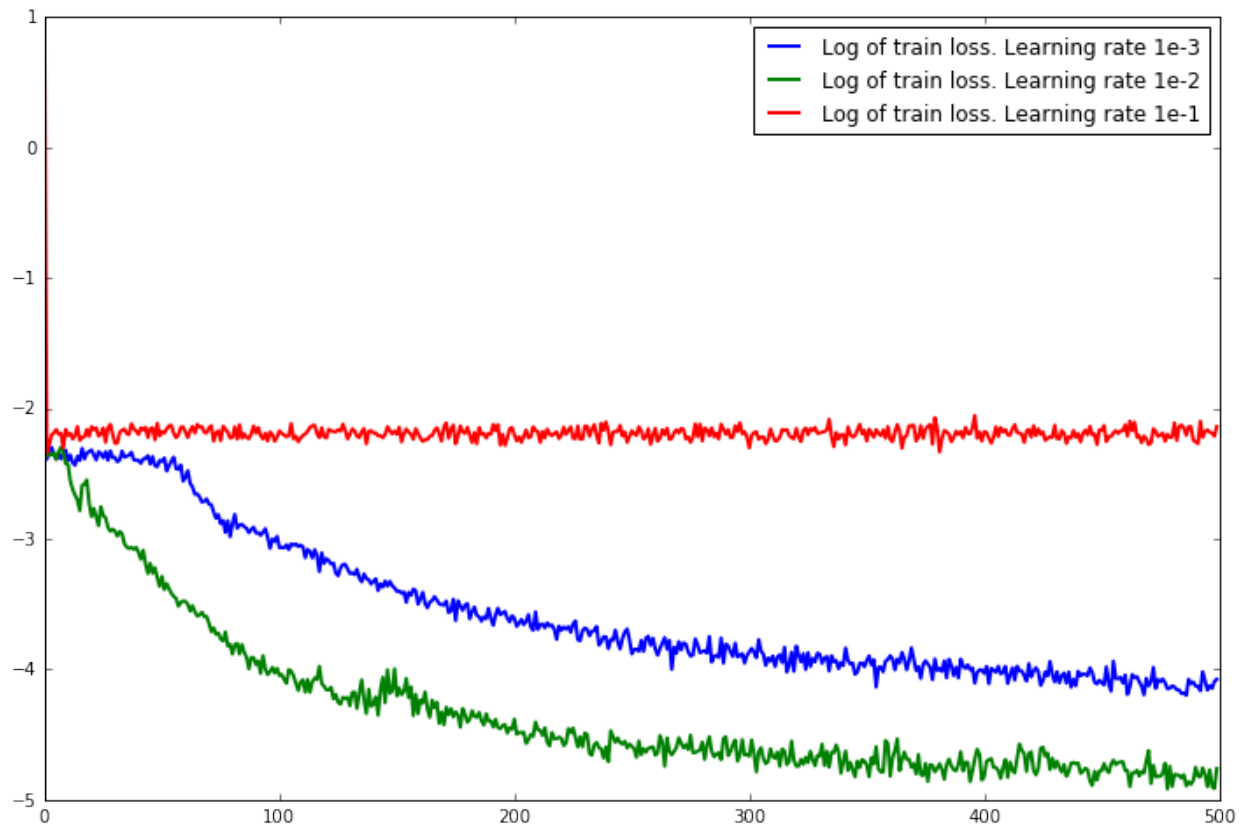
Deep Learning Lab Course 2017

Assignment 3.

Olesya Tsapenko

I have Implemented auto-encoder as it was described at the assignment.

After that I played with learning rate. Results you can see below (I have used logarithm of training loss for ease of understanding which learning rate is the best; 500 epochs)







Conclusions from this graphics:

- learning rate 0.1 is too big that we do not have any convergence at all;
- learning rate 0.001 is still too small however we have convergence, but it may take long time (particularly, at the first 50 epochs we do not see converges at all);
- learning rate 0.01 is the best.

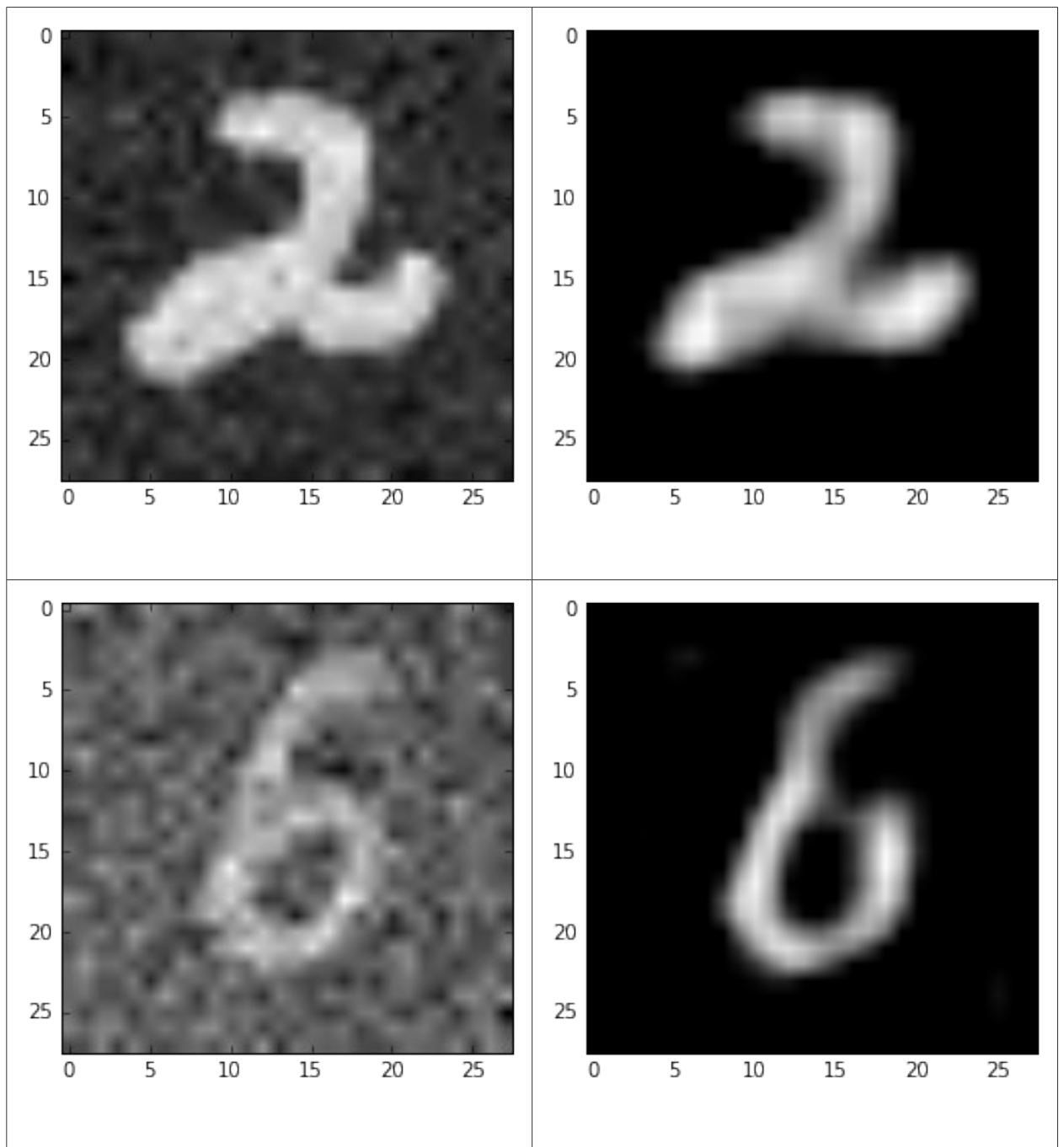
Based on this graph, I had decided to do all subsequent calculations with learning rate 0.01.

Some examples of auto-encoder:

Input images	Auto-encoded images
 <p>A 28x28 grayscale plot of the handwritten digit '7'. The x and y axes are labeled from 0 to 25 in increments of 5. The digit is drawn with a thick, slightly blurred white stroke on a black background.</p>	 <p>An auto-encoded version of the digit '7'. It is very similar to the input image but appears slightly more blurred and has a slightly different intensity distribution, particularly in the curved part of the digit.</p>
 <p>A 28x28 grayscale plot of the handwritten digit '9'. The x and y axes are labeled from 0 to 25 in increments of 5. The digit is drawn with a thick, slightly blurred white stroke on a black background.</p>	 <p>An auto-encoded version of the digit '9'. It is very similar to the input image but appears slightly more blurred and has a slightly different intensity distribution, particularly in the loop of the digit.</p>

After that I had added small random Gaussian noise to several images and look at the auto-encoded outputs (auto-encoder was trained only with “pure” images without any noise). Results can be seen below:

Input noisy images	Auto-encoded images
--------------------	---------------------



Conclusions from this assignment :

- auto-encoder works pretty fast (25 seconds for 500 epochs on the same computer which I used for the second assignment);
- even these 25 seconds are enough for having quite good results;
- auto-encoder for noisy images returns not-noisy results because it was trained with “pure” images and uses weight-matrices which correspond features of “pure” pictures. When we unwrapped our image back from weight-matrices we do not have information about noise . It still works good enough even with really noisy images.