AMS 598: Big Data Analysis (Fall 2024)

Assignment #5
Due Dec 6th, 2024

---

1. You need to submit (1) a report in PDF and (2) your code file, both to Brightspace.
2. Your PDF report should include (1) answers to the non-programming part, and (2) results and analysis of the programming part. For the programming part, your PDF report should at least include the results you obtained, for example the accuracy, training curves, parameters, etc. You should also analyze your results as needed.
3. Please put all your files (PDF report and code file) into a compressed file named "Assi5_FirstName_LastName.zip"
4. Unlimited number of submissions are allowed on Brightspace and the latest one will be timed and graded.
5. All students are highly encouraged to typeset their reports using Word or LaTeX. In case you decide to hand-write, please make sure your answers are clearly readable in scanned PDF.
6. Only write your code between the following lines. Do not modify other parts.

   ### YOUR CODE HERE

   ### END YOUR CODE
7. Please read and follow submission instructions. No exception will be made to accommodate incorrectly submitted files/reports.

---

1. (10 points) You'd like to train a fully-connected neural network with 5 hidden layers, each with 10 hidden units. The input is 20-dimensional and the output is a scalar. What is the total number of trainable parameters in your network (include bias in your calculation)?
2. (10 points) A single $(15 \times 15 \times 3)$ image is passed through a convolutional layer with 28 filters, each of size $(3 \times 3 \times 3)$. The padding size is 1 (1 unit at top, bottom, left, and right) and the stride size is 2. What is the size of the output feature map volume? What is the number of parameters in this layer (including bias)?
3. (10 points) A feature map X is passed to a convolutional layer with 3 $\times$ 3 filter K (i.e. operation $*$) with no padding and stride being 1, followed by a 2 $\times$ 2 max pooling layer (i.e. operation max), resulting in a scalar output. The process is shown below. Please give the final scalar with all intermediate steps.

$$\textbf{max}\left(\begin{array}{cccc} 4 & 6 & 3 & 4 \\ \underline{1} & 7 & 3 & 4 \\ 5 & 8 & 4 & 4 \\ 2 & 7 & 4 & 4 \end{array} \right)_{X} * \begin{array}{ccc} 0 & -1/4 & 0 \\ -1/4 & 1 & -1/4 \\ 0 & -1/4 & 0 \end{array}_{K}$$

4. (25 points) You come up with a CNN classifier. For each layer, calculate the number of weights, number of biases, and the size of the associated feature maps. The notation follows the convention:
   • CONV-K-N denotes a convolutional layer with N filters, each them of size K×K. Padding and stride are always 0 and 1 respectively.
   • POOL-K indicates a K×K pooling layer with stride K and padding 0.
   • FC-N stands for a fully-connected layer with N neurons.

   Please fill out the form below. Please also provide all intermediate steps after the table.

Layer	Activation map dimensions	Number of weights	Number of biases
INPUT	$128 \times 128 \times 3$	0	0
CONV-9-32			
POOL-2			
CONV-5-64			
POOL-2			
CONV-5-64			
POOL-2			
FC-3			

5. (45 points) (Coding Task) **Autoencoder:** In this assignment, you will apply the autoencoder (AE) to a collection of handwritten digit images from the USPS dataset, which is the same as the data used in Q4 of Assignment 4. The data file is stored in the "AE/data" folder as "USPS.mat". Please check the starting code in folder "AE/code" and follow the instructions. The whole dataset is already loaded and stored in the matrix A with shape 3000×256. Each row of matrix A represents a 16×16 handwritten digit image (between 0 and 9), which is flattened to a 256-dimensional vector. Note you need to use PyTorch in this assignment. **Please read the "Readme" file carefully before getting started.** You are expected to implement the solutions based on the starting code. The files you need to modify are "solution.py" and "main.py". You will test your solution by modifying and running the "main.py" file.

   (a) (10 points) In the ***class AE()***, complete the ***_network()*** and ***_forward()*** function. Note that for problems (a) and (c), the weights need to be shared between the encoder and the decoder with weight matrices transposed to each other.

   (b) (5 points) In the ***class AE()***, complete the ***reconstruction()*** function to perform data reconstruction. Please test your function using three different dimensions for the hidden representation d that $d = 10, 50, 100, 200$.

   (c) (10 points) Obtain reconstruction errors for AE, and then compare them with the reconstruction errors from PCA in Q4 of Assignment 4. Note that you need to set $k = d$ for comparisons. Please evaluate the errors using $k = d = 10, 50, 100, 200$. Report the reconstruction errors and provide a brief analysis.

(d) (10 points) Please modify the ***_network()*** and ***_forward()*** function so that the weights are **not** shared between the encoder and the decoder. Report the reconstructions errors for $d = 10, 50, 100, 200$. Please compare with the sharing weights case and briefly analyze you results.

(e) (10 points) Please modify the ***_network()*** and ***_forward()*** function to include more network layers and nonlinear functions. Please set $d = 64$ and explore different hyperparameters. Report the hyperparameters of the best model and its reconstruction error. Please analyze and report your conclusions.