CSI 5340 Homework Exercise 2

Note that to complete the homework, you must submit

- All Python codes for both questions
- A concise PDF file to include the required experimental results, descriptions, or other related information, for both questions.

Question 1 (40 points) *Suppose we have compositions of the following functions, where the input variable $x \in \mathbb{R}^K$; A, B, C are all $K \times K$ matrices:*

$$\begin{aligned} y &:= Ax && (1)\\ u &:= \sigma(y) && (2)\\ v &:= Bx && (3)\\ z &:= (u+v) && (4)\\ w &:= Cz && (5)\\ \mathcal{L} &:= \|w\|^2 && (6) \end{aligned}$$

where σ is the sigmoid function. For a given input vector x and a configuration of (A, B, C), write a program using back-propagation to compute the gradients $\frac{\partial \mathcal{L}}{\partial A}$, $\frac{\partial \mathcal{L}}{\partial B}$ and $\frac{\partial \mathcal{L}}{\partial C}$. **Note: computing the gradients by deriving the closed form solution is not back-propagation.** *Based on the gradients computed by back-propagation, write another program that finds*

$$(\widehat{A}, \widehat{B}, \widehat{C}) := \arg \min_{(A,B,C)} \sum_{i=1}^{N} \mathcal{L}(x_i; A, B, C)$$

using gradient descent, for any N points $x_1, x_2, \ldots, x_N$ in $\mathbb{R}^K$. **Note: The programs must be written in Python. You are only allowed to use the Python library "random" to generate random numbers and the library "math" to compute the sigmoid function. Except for these, any other packages or libraries are NOT allowed to use in your program (e.g., Numpy, Pytorch, etc, are not allowed). That is, you must manually implement the forward propagation, back-propagation, and also operations such as matrix-vector multiplication.** *In the PDF file, write a brief description of the steps of your back-propagation algorithm and exhibit the execution results of your program. After these, write a program to compute the gradients using an automatic differentiation library (e.g., torch.autograd). Compare this result with the result computed by the back-propagation that is implemented by yourself. (If you have different results, that means your implementation is wrong). Also include this result in your PDF file.*

Question 2 (60 points) *MNIST dataset is a simple and popular dataset for image classification. You can download the dataset from* `http://yann.lecun.com/exdb/mnist/` *and get more information about the dataset. In this exercise, you will need to do the following.*

- *develop three classifiers for this dataset using three different models: soft-max regression, MLP, and CNN.*
- *For each model, investigated its behaviour with and without dropout.*
- *For each model, investigated its behaviour with and without batch normalization.*

You may freely explore any design freedom in each model (e.g., width/depth of MLP, kernel size/number of kernels/depth in CNN). You need to submit your code together with a report documenting your observations. In your report, you may feel free to include anything interesting you observe and remark on the lessons learned.