Ahmet Akman 2442366
Middle East Technical University
Electrical and Electronics Engineering

April 4, 2024

HOMEWORK 1 — Report

1 Question 1

1.1 Question 1.1 - Preliminaries

The partial derivaive calculation steps for Tanh, Sigmoid and ReLU activation functions are shown in Figure 1.

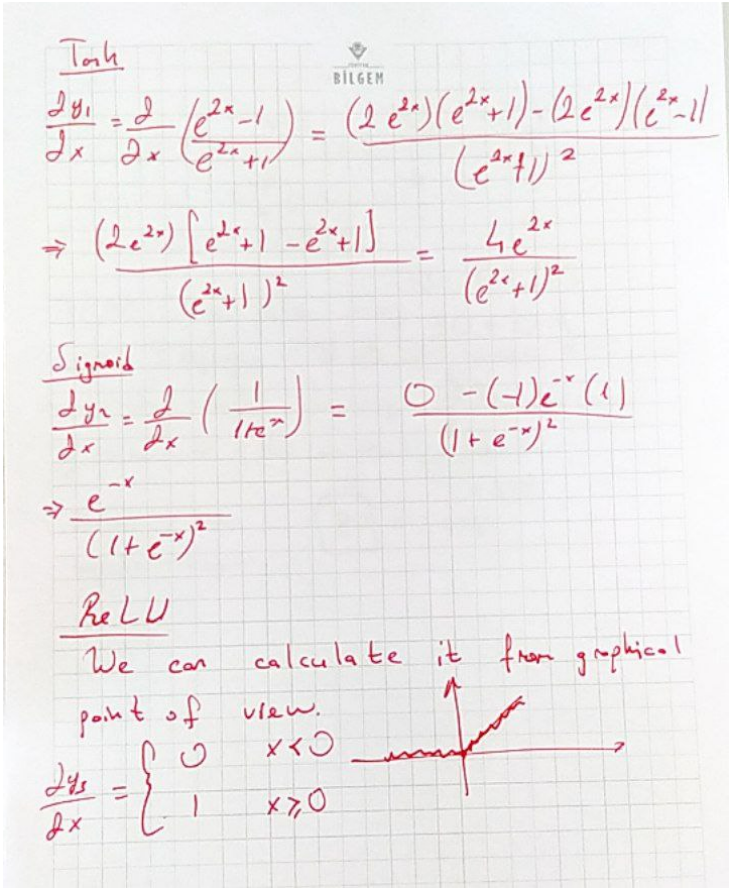


Figure 1: Partial derivative calculation steps for Tanh, Sigmoid and ReLU activation functions.

Figure 2 illustrates the activation functions' response between -2 and 2. The plots are obtained using matplotlib library as instucted.

Figure 3 indicates the gradients of those functions in the same range. The gradients are calculated using the partial derivatives derived in Figure 1.

1.2 Question 1.2

In this part, utilizing the given template of code, MLP with one hidden layer is implemented. The input-output pairs are fetched from the XOR data provided in utils.py. Note that for all networks learning rate

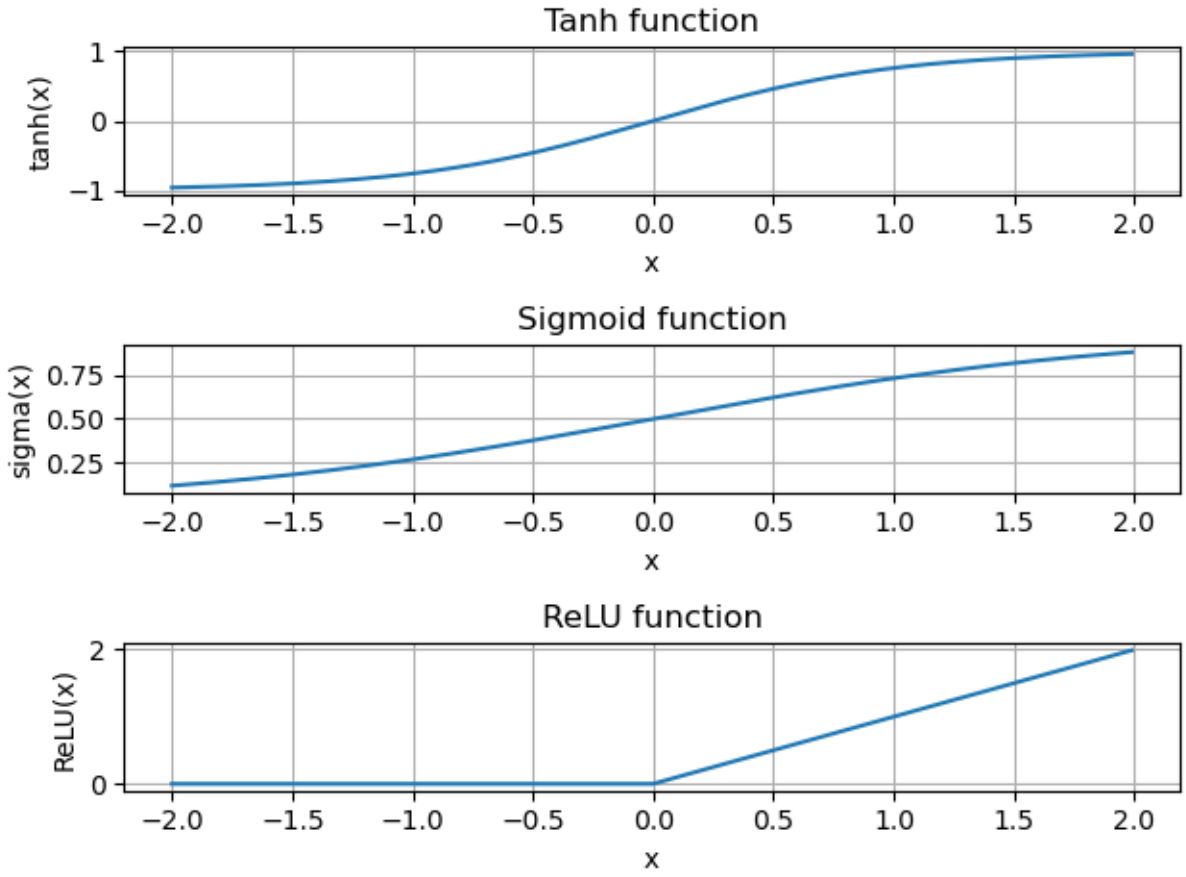


Figure 2: Activation functions plot.

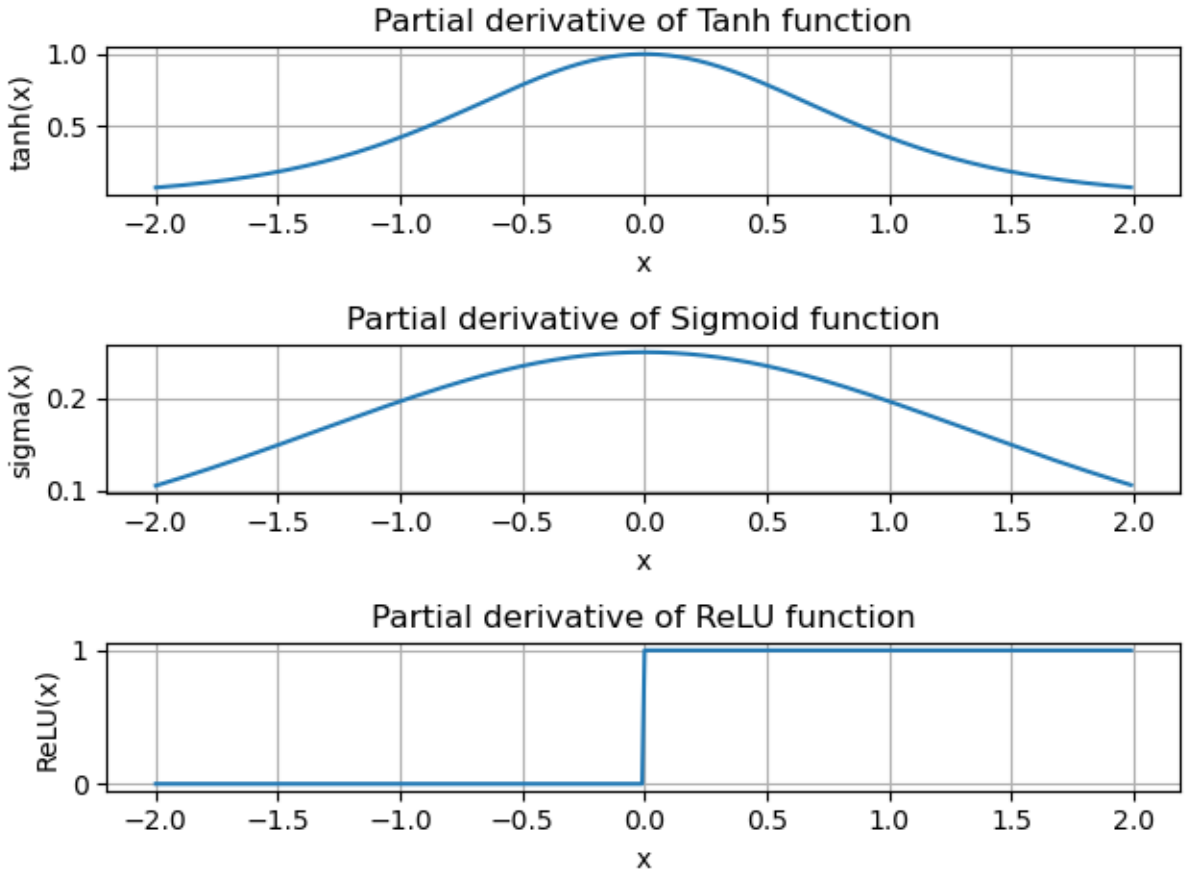


Figure 3: Gradients of the activation functions plot.

is fixed to 0.00001 and seed is utilized. Figure 4 shows the decision boundary for the sigmoid activated network. Similarly, Figure 5 is the decision boundary for the tanh activated network and Figure 6 is the

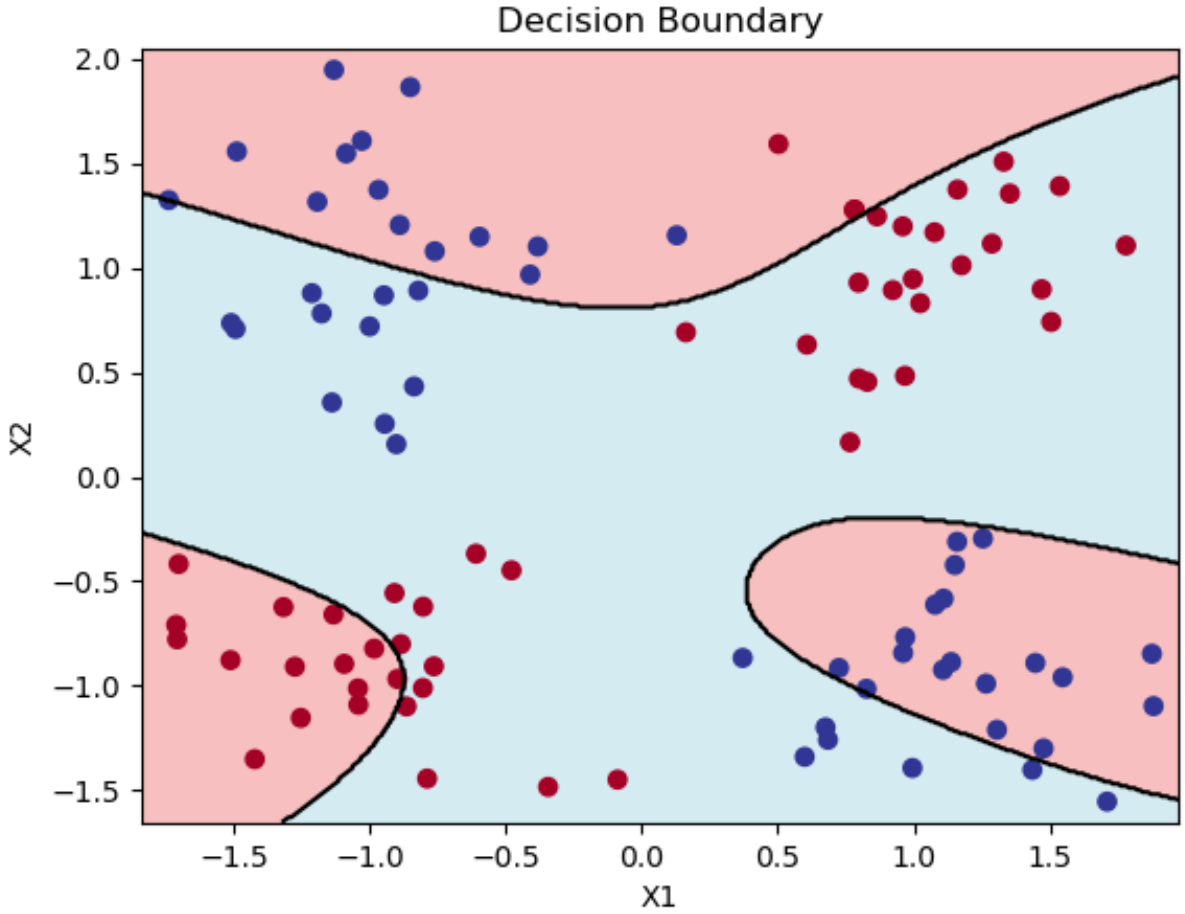


Figure 4: Sigmoid activated XOR problem output.

decision boundary for the ReLU activated network.

1.3 Question 1.3

1. All of those activation functions provide a smooth transition from one state to another. The advantage of Tanh and Sigmoid is, they are limited in the range of -1 to 1 and 0 to 1, respectively. This property can be beneficial in some cases. However, the ReLU function is not limited in the range. It is also computationally cheaper than the other two. The disadvantage of ReLU is that it is not smooth at the origin. This can cause some problems in the optimization process. Another advantage of ReLU is negative gradients are zero. This may be helpful in some cases. **2.**

XOR problem is two input decision problem where inputs have to be different than each other to obtain 1. Since the output is always with respect to the state of two variables, decision boundary is not linear. Therefore, a single layer perceptron can not solve since in one step two case evaluations can not be done. Yet, by adding a hidden layer, the problem is solved. The activation functions are used to introduce non-linearity to the network. The decision boundary of the XOR problem is shown in Figures 4, 5 and 6. As can be seen from the figures, the MLP's with activation functions can solve the XOR problem at some extent.

3. The boundaries change in each run, since the initial weights and the data points are randomly generated. Therefore, the decision boundaries are dependent on the training process where initial randomness

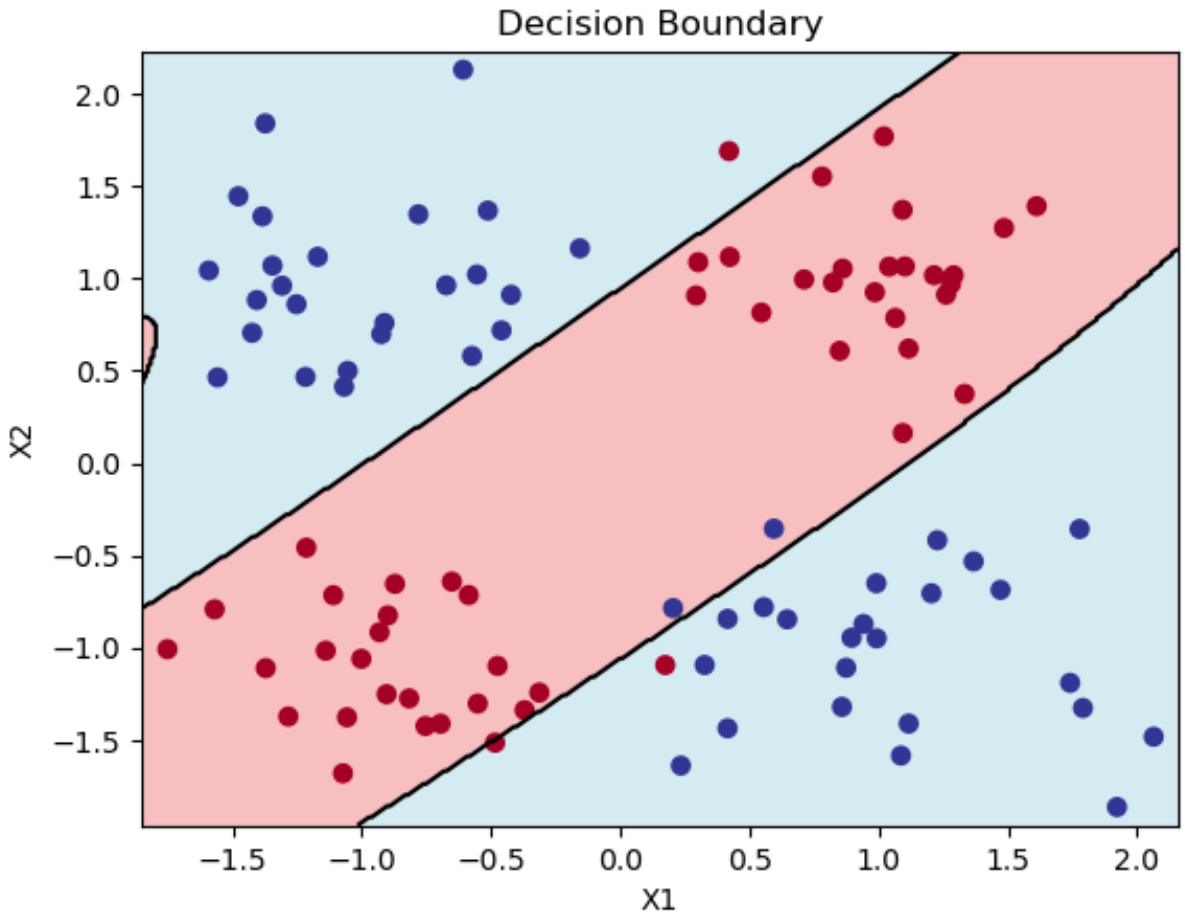


Figure 5: Tanh activated XOR problem output.

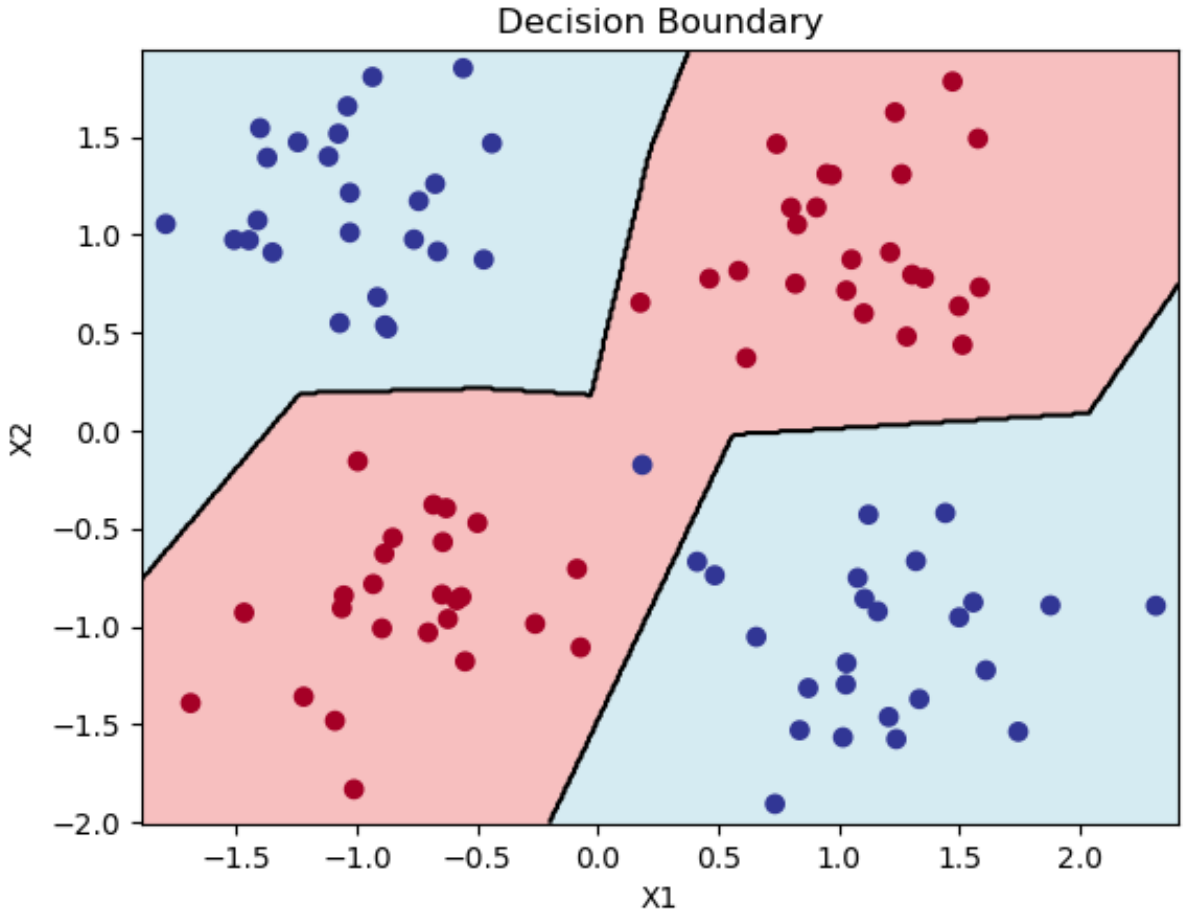


Figure 6: ReLU activated XOR problem output.

lead to different outputs.

2 Question 2

A convolution operator is implemented. Using the provided inputs, outputs provided in Figure 7 is obtained.

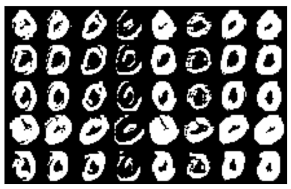
2.1 Question 2.2 - Discussions

1. 2. 3. 4. 5. 6.

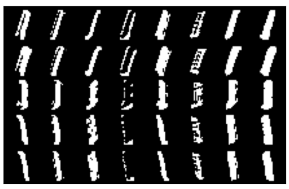
3 Question 3

4 Question 4

5 Question 5



(a) Output 0



(b) Output 1

(c) Output 2

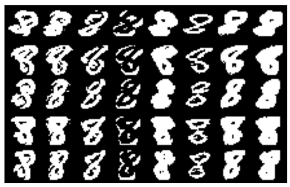
(d) Output 3

(e) Output 4

(f) Output 5

(g) Output 6

(h) Output 7



(i) Output 8

(j) Output 9

Figure 7: Convolution output

6 References

References

[1] M. Stimberg, R. Brette, and D. F. Goodman, "Brian 2, an intuitive and efficient neural simulator," *eLife*, vol. 8, p. e47314, Aug. 2019.

[2] R. Brette, M. Rudolph-Lilith, N. T. Carnevale, M. L. Hines, D. Beeman, J. M. Bower, M. Diesmann, A. Morrison, P. H. Goodman, F. C. Harris, M. Zirpe, T. Natschläger, D. Pecevski, B. Ermentrout, M. Djurfeldt, A. Lansner, O. Rochel, T. Viéville, E. B. Müller, A. P. Davison, S. E. Boustani, and A. Destexhe, "Simulation of networks of spiking neurons: A review of tools and strategies," *Journal of Computational Neuroscience*, vol. 23, pp. 349–398, 2006.

Submitted by Ahmet Akman 2442366 on April 4, 2024.