Assignment 1 Report

This is an outline for your report to ease the amount of work required to create your report. Jupyter notebook supports markdown, and I recommend you to check out this cheat sheet. If you are not familiar with markdown.

Before delivery, **remember to convert this file to PDF**. You can do it in two ways:

1. Print the webpage (ctrl+P or cmd+P)
2. Export with latex. This is somewhat more difficult, but you'll get somehwat of a "prettier" PDF. Go to File -> Download as -> PDF via LaTeX. You might have to install nbconvert and pandoc through conda; `conda install nbconvert pandoc` .

Task 1

task 1a)

Having that:

$$w_{j,i} =: w_{j,i} - \alpha \frac{\partial C}{\partial w_{j,i}}$$

$$\frac{\partial C}{\partial w_{j,i}} = \frac{\partial C}{\partial z_j} \frac{\partial z_j}{\partial w_{j,i}} = \delta_j \frac{\partial}{\partial w_{j,i}} \sum_{i=0}^{I} w_{j,i} x_i = \delta_j x_i$$

Which shows that:

$$w_{j,i} =: w_{j,i} - \alpha \delta_j x_i$$

Next we want to show that:

$$\delta_j = f'(z_j) \sum_k w_{k,j} \delta_k$$

Starting from that $\delta_j = \frac{\partial C}{\partial z_j}$. This is the effect on the cross-entropy loss from á spesific node in the hidden layer. This effekt is propagated through the output layer, meaning each output node will contain some of this effect, thus:

$$\delta_j = \frac{\partial C}{\partial z_j} = \sum_k \frac{\partial C}{\partial z_k} \frac{\partial z_k}{\partial z_j} = \sum_k \delta_k \frac{\partial z_k}{\partial z_j}$$

$$z_k = \sum_j w_{k,j} a_j + b_k$$

The above formula says that the output of a node is the weighted sum of effects from the previous layer. Moving on from the result of the expanded equation for δ_j we have:

$$\frac{\partial z_k}{\partial z_j} = \frac{\partial z_k}{\partial a_j}\frac{\partial a_j}{\partial z_j} = w_{k,j} f'(z_j)$$

Combining the "components" that was found above we get that:

$$\delta_j = f'(z_j)\sum_k w_{k,j}\delta_k$$

task 1b)

From the hidden layer (j) to the output layer (k), we have:

$$\boldsymbol{W}^{l+1} := \boldsymbol{W}^{l+1} - \alpha(\delta^{l+1})^T \cdot \boldsymbol{a}^l$$

Where $l =$ first layer, and $l+1 =$ next layer.

For input layer (i) to the hidden layer (j) we have:

$$\boldsymbol{W}^{l} := \boldsymbol{W}^{l} - \alpha(\delta^{l})^T \cdot \boldsymbol{x}$$

where:

$$\delta^l = f'(\boldsymbol{z}^l)(\boldsymbol{W}^{l+1})^T \cdot \delta^{l+1}$$

The dimentions of the matrixes and vectors are as follows:

$$\boldsymbol{x} \; \epsilon \; \mathbb{R}^I$$

$$\boldsymbol{W}^l \; \epsilon \; \mathbb{R}^{JxI}$$

$$\boldsymbol{\delta}^l \; \epsilon \; \mathbb{R}^J$$

$$\boldsymbol{a}^l \; \epsilon \; \mathbb{R}^J$$

$$\boldsymbol{W}^{l+1} \; \epsilon \; \mathbb{R}^{KxJ}$$

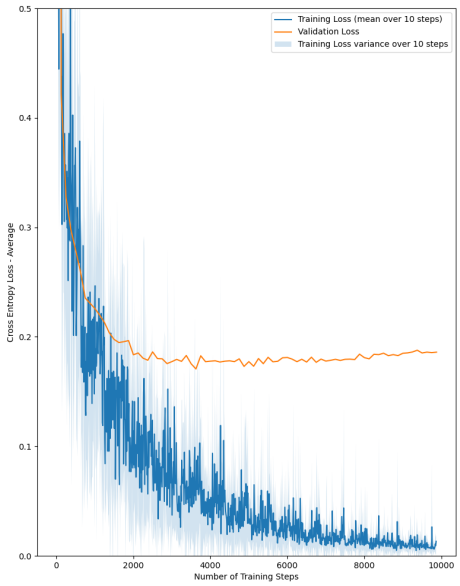
$$\boldsymbol{\delta}^{l+1} \; \epsilon \; \mathbb{R}^K$$

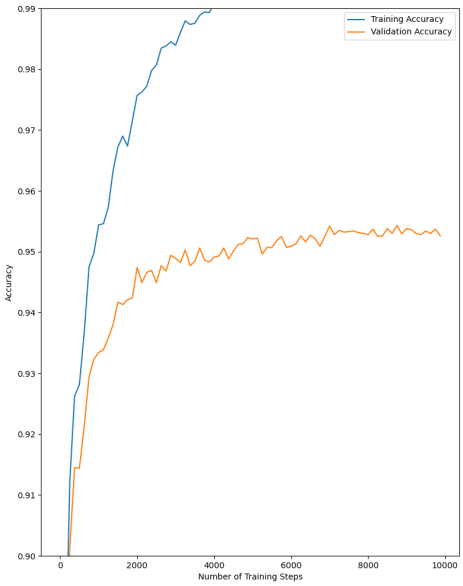
Task 2

Task 2a)

Mean = 33.55274553571429 Standard deviation = 78.87550070784701

Task 2c)





Task 2d)

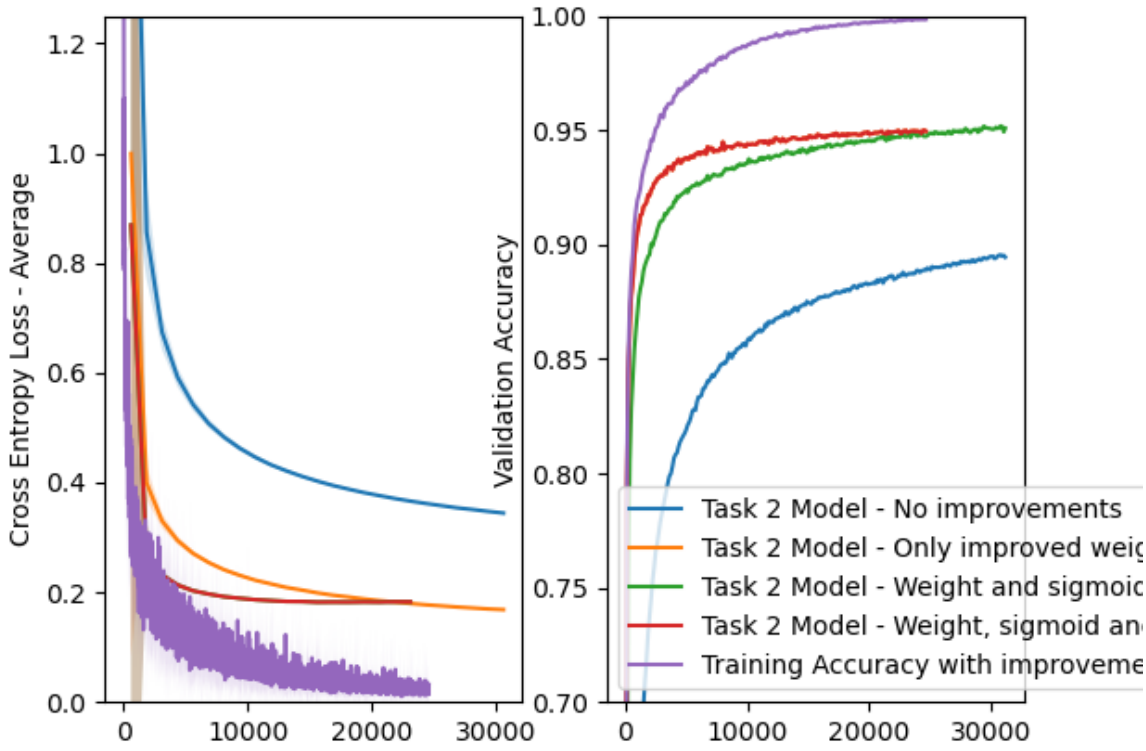
The total number of parameters is the sum of the total number of weights and the total number of biases. This is given by: $784 \cdot 64 + 64 \cdot 10 + 784 + 64 = 50816$

Task 3

Epochs:

Without improvements: 49 epochs \ With improved weight init: 49 epochs \ With improved weight and sigmoid: 39 epochs \ With improved weight and sigmoid, and momentum: 39 epochs \

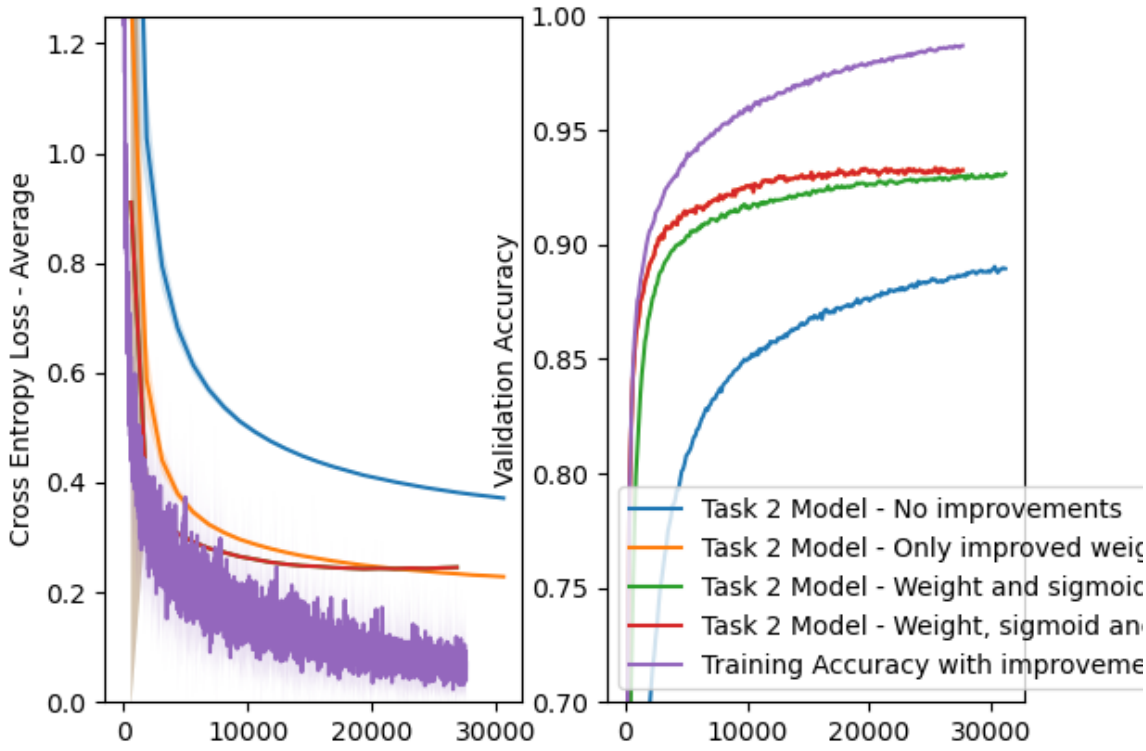
As seen from the plot, there is no signs of overfitting of the training data. When implementing the sigmoid function, early stopping kicked in at epoch 39, ti epochs earlier than without it. Improved weights initialization and sigmoid function improved convergence properties, but by included the momentum function, convergence was sped up considerably.



Task 4

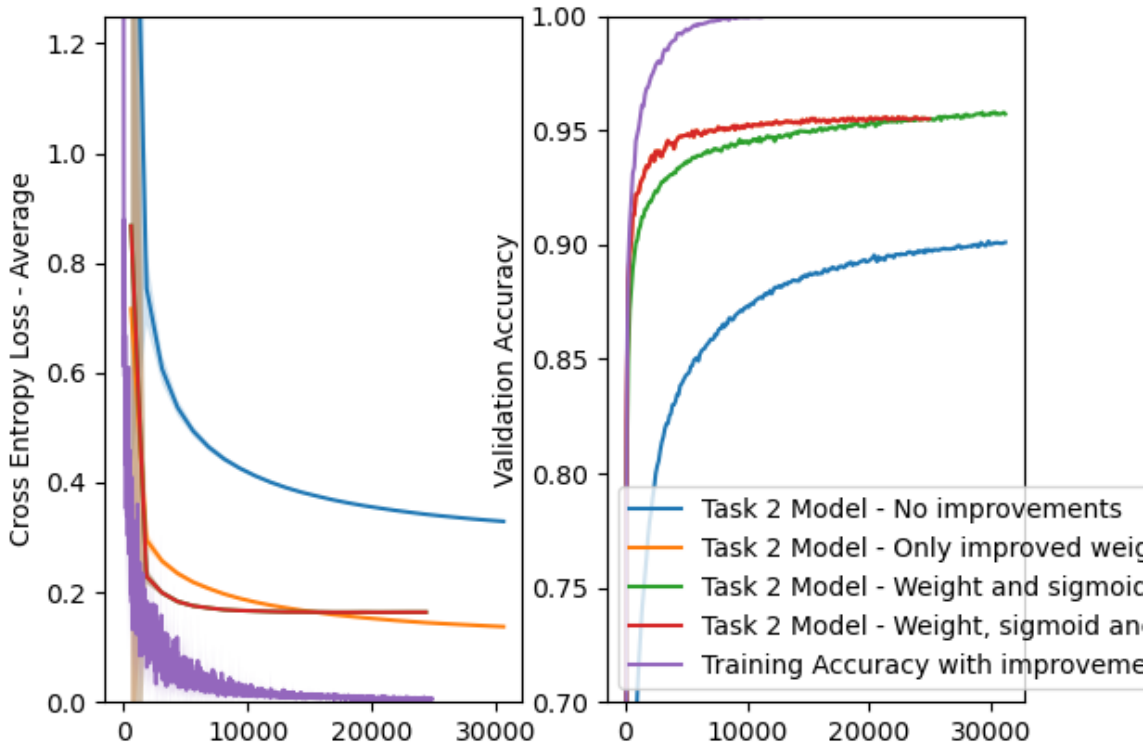
Task 4a)

As seen from the plot, cross-entropy loss and validation accuracy were somewhat worse when using 32 neurons in the hidden layer. In addition, early stopping kicked in at epoch 44 instead of 39.



Task 4b)

Using 128 neurons in the hidden layer, the final cross-entropy loss and validation accuracy actually improved. Also, although early stopping kicked in at 39 epochs, the increased complexity increased calculation time.



Task 4d)

FILL IN ANSWER

Task 4e)

FILL IN ANSWER