Assignment 1 Report

Etienne Gaucher, Benedikt Blumenstiel

Task 1

Task 1a)

We first use the chain rule.

$$\frac{\partial C^n(w)}{\partial w_i} = \frac{\partial C^n(w)}{\partial f(x^n)} \frac{\partial f(x^n)}{\partial w_i}$$

$$\frac{\partial C^n(w)}{\partial w_i} = \frac{\partial C^n(w)}{\partial \hat{y}^n} \frac{\partial f(x^n)}{\partial w_i}$$

because $f(x^n) = \hat{y}^n$

Then, we compute $\frac{\partial C^n(w)}{\partial \hat{y}^n}$.

$$\frac{\partial C^n(w)}{\partial \hat{y}^n} = \frac{\partial}{\partial \hat{y}^n}(-(y^n \ln(\hat{y}^n) + (1-y^n)\ln(1-\hat{y}^n)))$$

$$\frac{\partial C^n(w)}{\partial \hat{y}^n} = -(\frac{y^n}{\hat{y}^n} + \frac{y^n - 1}{1-\hat{y}^n}) = -(\frac{y^n - \hat{y}^n}{\hat{y}^n(1-\hat{y}^n)})$$

We obtain

$$\frac{\partial C^n(w)}{\partial w_i} = -x_i^n f(x^n)(1-f(x^n))(\frac{y^n - \hat{y}^n}{\hat{y}^n(1-\hat{y}^n)})$$

$$\frac{\partial C^n(w)}{\partial w_i} = -(y^n - \hat{y}^n)x_i^n$$

because $f(x^n) = \hat{y}^n$

Task 1b)

We first use the chain rule.

$$\frac{\partial C^n(w)}{\partial w_{ij}} = \frac{\partial C^n(w)}{\partial z_i} \frac{\partial z_i}{\partial w_{ij}}$$

$$\frac{\partial C^n(w)}{\partial z_i} = -\sum_{k=1}^{K} y_k^n \frac{\partial \ln(\hat{y}_k^n)}{\partial z_i}$$

$$\frac{\partial C^n(w)}{\partial z_i} = -\sum_{k=1}^{K} \frac{y_k^n}{\hat{y}_k^n} \frac{\partial \hat{y}_k^n}{\partial z_i}$$

If $k \neq i$,

$$\frac{\partial \hat{y}_k^n}{\partial z_i} = \frac{0 - e^{z_k} e^{z_i}}{(\sum_{k'}^{K} e^{z_k'})^2} = -\hat{y}_k^n \times \hat{y}_i^n$$

If $k = i$,

$$\frac{\partial \hat{y}_i^n}{\partial z_i} = \frac{e^{z_i} \sum_{k'}^{K} e^{z_k'} - e^{z_i} e^{z_i}}{(\sum_{k'}^{K} e^{z_k'})^2} = \hat{y}_i^n - (\hat{y}_i^n)^2 = \hat{y}_i^n (1 - \hat{y}_i^n)$$

Then,

$$\frac{\partial C^n(w)}{\partial z_i} = \sum_{k=1, k \neq i}^{K} \frac{y_k^n}{\hat{y}_k^n} \times \hat{y}_k^n \times \hat{y}_i^n - \frac{y_i^n}{\hat{y}_i^n} \hat{y}_i^n (1 - \hat{y}_i^n)$$

$$\frac{\partial C^n(w)}{\partial z_i} = \sum_{k=1, k \neq i}^{K} y_k^n \times \hat{y}_i^n - y_i^n \times (1 - \hat{y}_i^n)$$

$$\frac{\partial C^n(w)}{\partial z_i} = \sum_{k=1}^{K} y_k^n \times \hat{y}_i^n - y_i^n$$

$$\frac{\partial C^n(w)}{\partial z_i} = \hat{y}_i^n \sum_{k=1}^{K} y_k^n - y_i^n$$

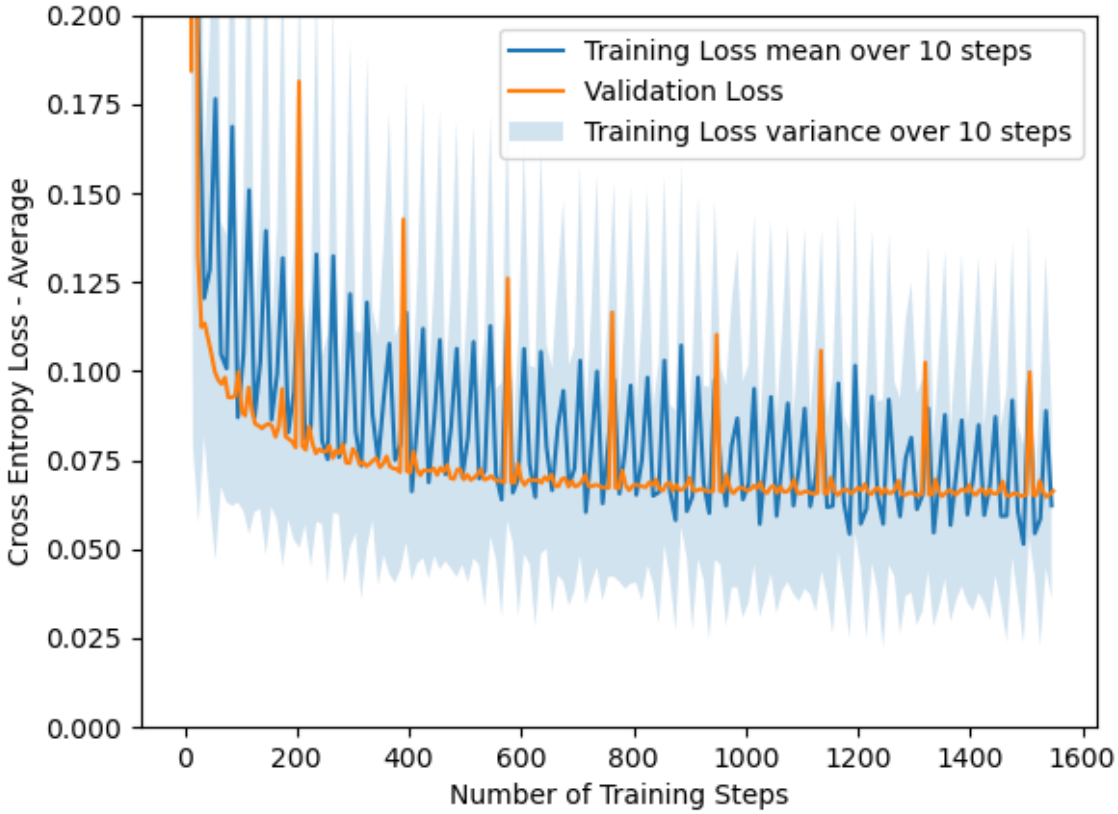
$$\frac{\partial C^n(w)}{\partial z_i} = \hat{y}_i^n - y_i^n$$

Finaly,

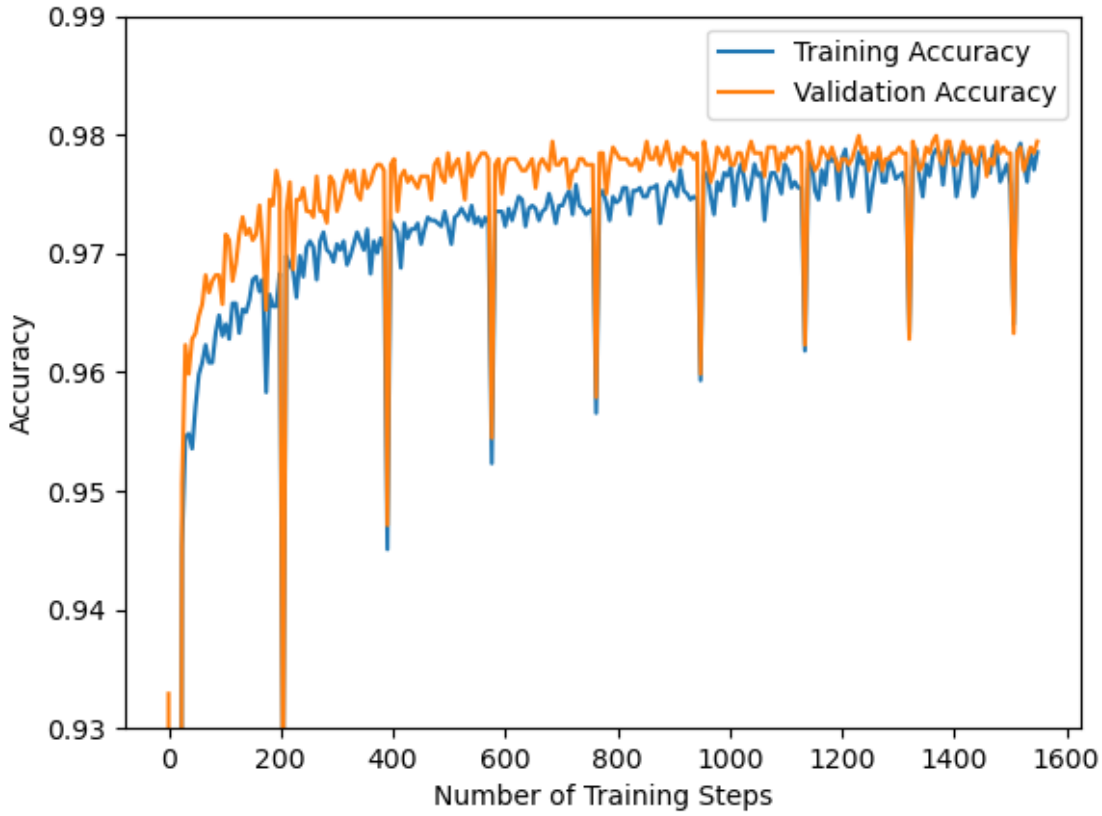
$$\frac{\partial C^n(w)}{\partial w_{kj}} = \frac{\partial C^n(w)}{\partial z_k} \frac{\partial z_k}{\partial w_{kj}} = (\hat{y}_k^n - y_k^n) x_j^n = -x_j^n (y_k^n - \hat{y}_k^n)$$

Task 2

Task 2b)



Task 2c)



Task 2d)

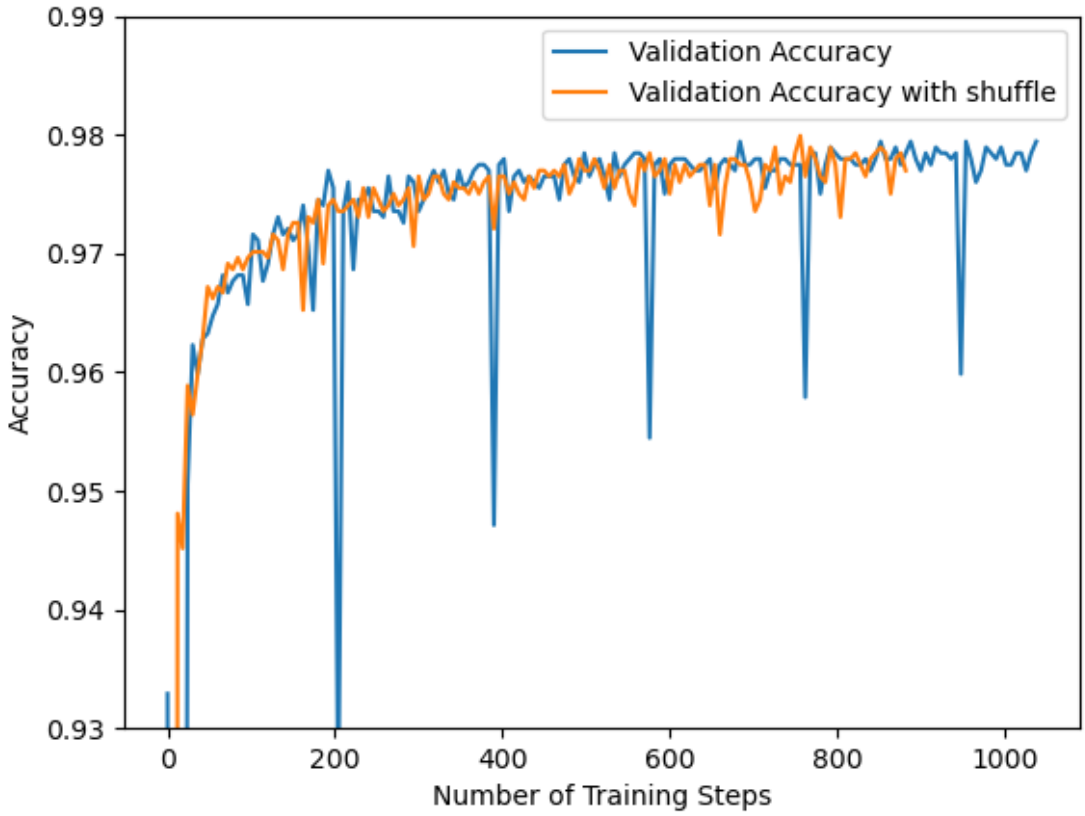
After how many epochs does early stopping kick in?

The training is stopped after 33 epochs in epoch 34 with validation loss of 0.0658.

Task 2e)

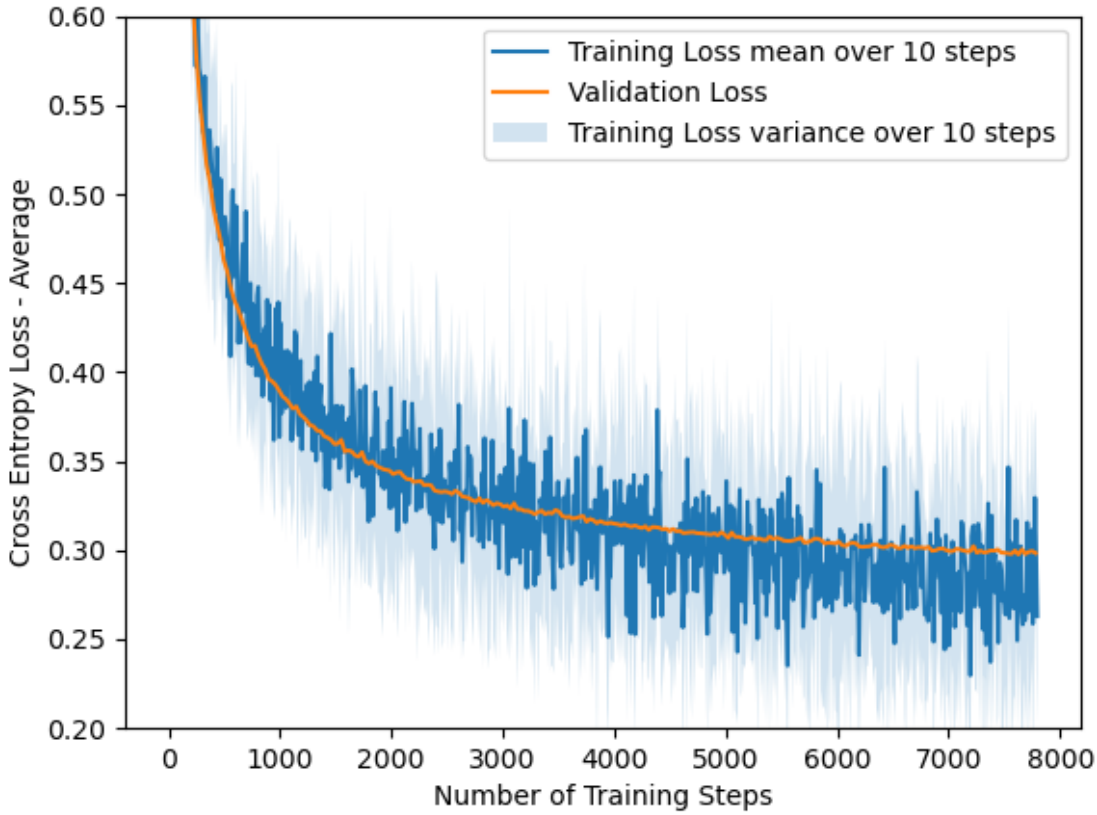
In the dataset some images are harder to predict than others. In some batches there may be an above-average number of these harder to predict images. If they are not shuffled, they are processed at regular intervals and can be seen in the plot as "spikes".

If the batches are shuffled, the incorrectly predicted images are distributed more evenly across all batches and there are no negative "spikes".

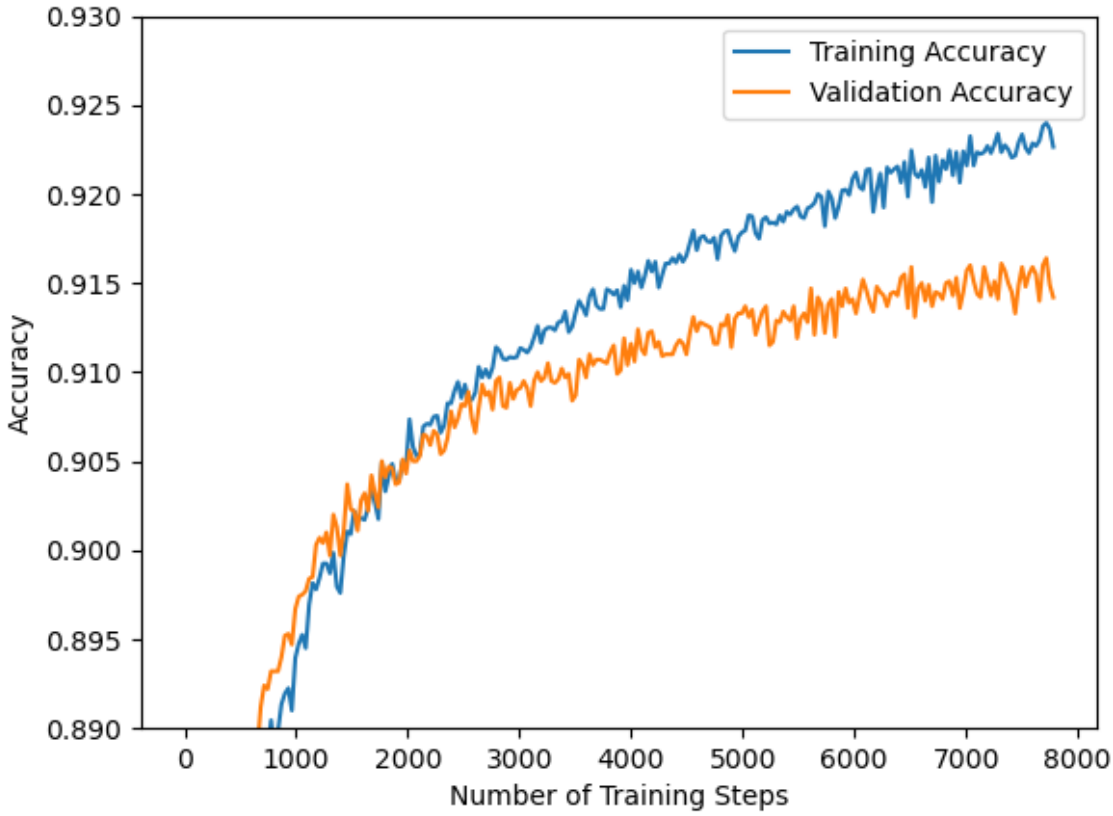


Task 3

Task 3b)



Task 3c)



Task 3d)

Do you notice any signs of overfitting?

Signs of overfitting can be seen. After 2500 training steps, the accuracy of the validation set still increases, but it shows a significantly lower value than the accuracy on the training set. This suggests that the model generalises less well and instead memorises patterns in the training set. This leads to overfitting. However, the increasing validation accuracy also shows that generalised patterns are still being learned.

Task 4

Task 4a)

$$\frac{\partial J}{\partial w} = \frac{\partial C(w)}{\partial w} + \lambda \frac{\partial R(w)}{\partial w}$$

where $R(w) = ||w||^2 = w^T w$

$$\frac{\partial J}{\partial w} = \frac{1}{N} \sum_{n=1}^{N} \frac{\partial C^n(w)}{\partial w} + 2\lambda w$$

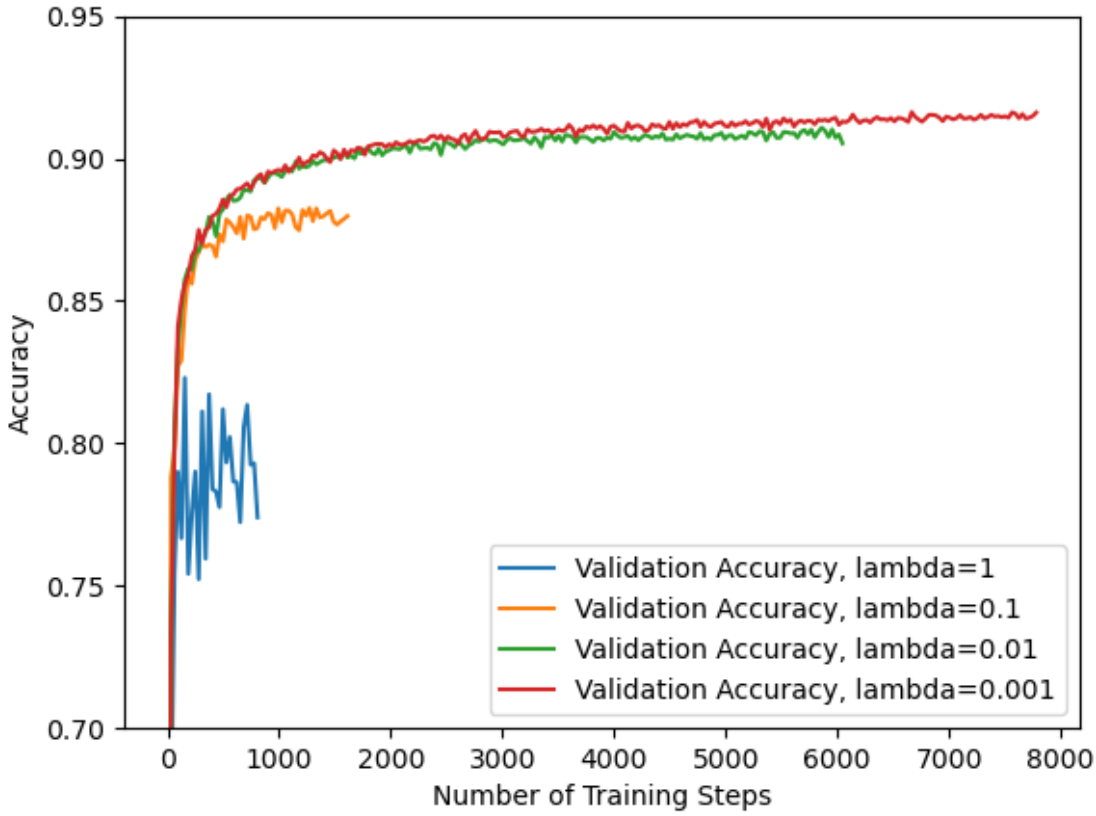
$$\frac{\partial J}{\partial w} = -\frac{1}{N} \sum_{n=1}^{N} x^n (y^n - \hat{y}^n) + 2\lambda w$$

Task 4b)

The weights for the model with λ = 1 are less noisy because of the L_2 penalty. The penalty forces the weights to be smaller, so that the model is less complex. Other than the model without penalty the model is not overfitting and is learning more general information.

Figure 1: The visualization of the weights for a model with λ = 0.0 (top row), and λ = 1.0 (bottom row).

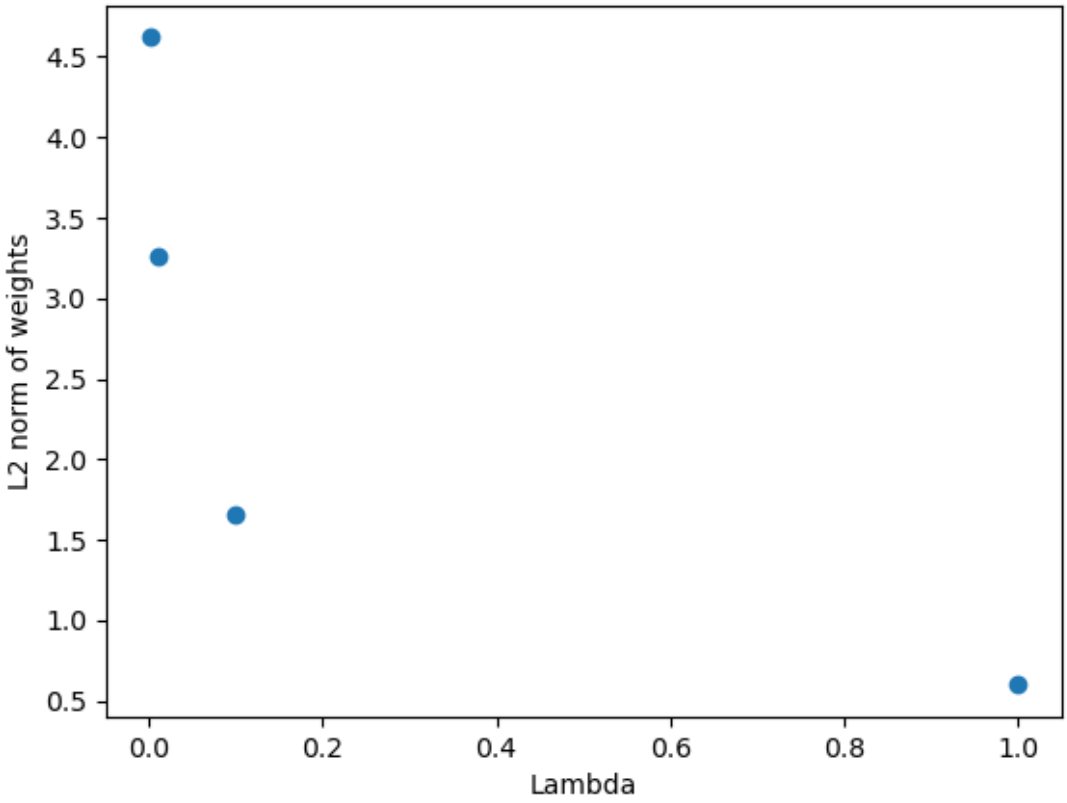
Task 4c)



Task 4d)

The validation accuracy degrades when increasing the amount of regularization since the regularization lowers the complexity of the neural network model during training. Reduced complexity can prevent overfitting. But a high regularization is also limiting the capability of the model to learn general patterns and the model is underfitting. This results in a lower validation accuracy after increasing lambda to 0.1 or even 1.

Task 4e)



We observe that the L_2 norm of the weight vector $||w||^2$ is inverse proportional to λ. Therefore, the information stored in the weights and the resulting complexity of the model is decreasing with higher λ. The low weights with $\lambda = 1$ leads to underfitting of the model because the penalty of the L_2 regularization is too high.