

## Exercise 2

### Machine Learning in Graphics & Vision

Driton Goxhufi   4233242   driton.goxhufi@student.uni-tuebingen.de  
 Damir Ravlija   5503184   damir.ravlija@student.uni-tuebingen.de

#### 1 Task 1

- (a) Classification accuracy of the initialized model on the test dataset is 0.5
- (b) Loss of the initialized model is 0.7149616252170096.
- (c) In the first step of derivation we use the chain rule and the fact that  $f'_w(x) = f_w(x)(1 - f_w(x))$ .

$$\begin{aligned}
 \frac{\partial L(\mathbf{x}, t, \mathbf{w})}{\partial \mathbf{w}} &\stackrel{(1)}{=} \frac{1}{N} \sum_{n=1}^N \left[ -t_n \frac{1}{f_w(\mathbf{x}_n)} f_w(\mathbf{x}_n)(1 - f_w(\mathbf{x}_n)) \mathbf{x}_n + (1 - t_n) \frac{1}{1 - f_w(\mathbf{x}_n)} f_w(\mathbf{x}_n)(1 - f_w(\mathbf{x}_n)) \mathbf{x}_n \right] \\
 &= \frac{1}{N} \sum_{n=1}^N [-t_n(1 - f_w(\mathbf{x}_n)) \mathbf{x}_n + (1 - t_n) f_w(\mathbf{x}_n) \mathbf{x}_n] \\
 &= \frac{1}{N} \sum_{n=1}^N [(-t_n + t_n f_w(\mathbf{x}_n) + f_w(\mathbf{x}_n) - t_n f_w(\mathbf{x}_n)) \mathbf{x}_n] \\
 &= \frac{1}{N} \sum_{n=1}^N [f_w(\mathbf{x}_n) - t_n] \mathbf{x}_n
 \end{aligned}$$

(1)

$$\begin{aligned}
 \frac{\partial f_w(x)}{\partial x} &= \frac{\partial}{\partial x} \left( \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}} \right) = \frac{e^{-\mathbf{w}^T \mathbf{x}}}{(1 + e^{-\mathbf{w}^T \mathbf{x}})^2} = \frac{1 + e^{-\mathbf{w}^T \mathbf{x}} - 1}{(1 + e^{-\mathbf{w}^T \mathbf{x}})^2} = \frac{1}{(1 + e^{-\mathbf{w}^T \mathbf{x}})} - \frac{1}{(1 + e^{-\mathbf{w}^T \mathbf{x}})^2} \\
 &= \frac{1}{(1 + e^{-\mathbf{w}^T \mathbf{x}})} \left( 1 - \frac{1}{(1 + e^{-\mathbf{w}^T \mathbf{x}})} \right) = f_w(\mathbf{w}^T \mathbf{x})(1 - f_w(\mathbf{w}^T \mathbf{x}))
 \end{aligned}$$

After 1,000 iterations the loss and accuracy of the model are:

loss = 0.3868595564299156

accuracy = 0.83

- (d) Since  $f_w$  represents the class posterior probability  $p(c_1|\mathbf{x})$ , the loss is minimal for  $t = 1$  when this probability is the highest (i.e.  $f_w = 1$ ). As the confidence of the model decreases, the loss function grows up to infinity which corresponds to the case when the model is confident in the wrong class, i.e. assigns  $f_w = 0$  (see Figure (2)).

On the other hand, if the label  $t = 0$ , the loss is minimized when  $p(c_2|\mathbf{x}) = 1 - p(c_1|\mathbf{x}) = 1 - f_w$  is maximized which is achieved when  $f_w = 0$ . The loss is higher when the model assigns higher probability to the other class, and grows to infinity as  $f_w$  nears 1.

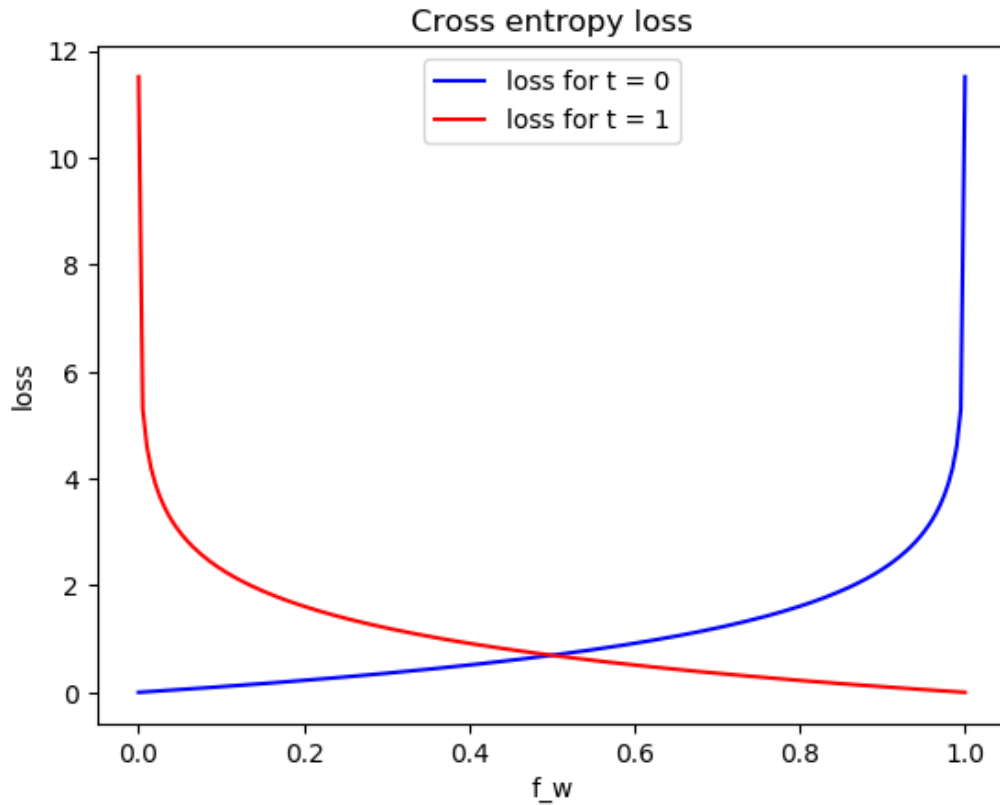


Figure 1: Plot of results from task (d)

- (e) We observe that the lower the learning rate is, the increase in accuracy and loss decrease are slower but more stable over time. Even after 1,000 iterations, smaller learning rates show increasing accuracy and decreasing loss trends, which implies that they would benefit from more iterations. Optimal learning rate that shows the best combination of stability and convergence speed is 0.1. Both loss and accuracy achieve the best values when the learning rate is set to 0.1.

In the extreme case of learning rate set to 1, both loss and accuracy are extremely unstable. This instability probably arises from overly large steps that the gradient descent makes and which causes it to overstep the minima. Conversely, in case of the smallest learning rate 0.0001 the accuracy of the model starts increasing only after around 200 iterations.

- (f)
- (g) Inference step of this learning approach is more efficient than the nearest neighbors search which has to find the  $k$  nearest neighbors. Another advantage of this learning approach is that it returns the probability of how confident it is that the value belongs to a certain class.

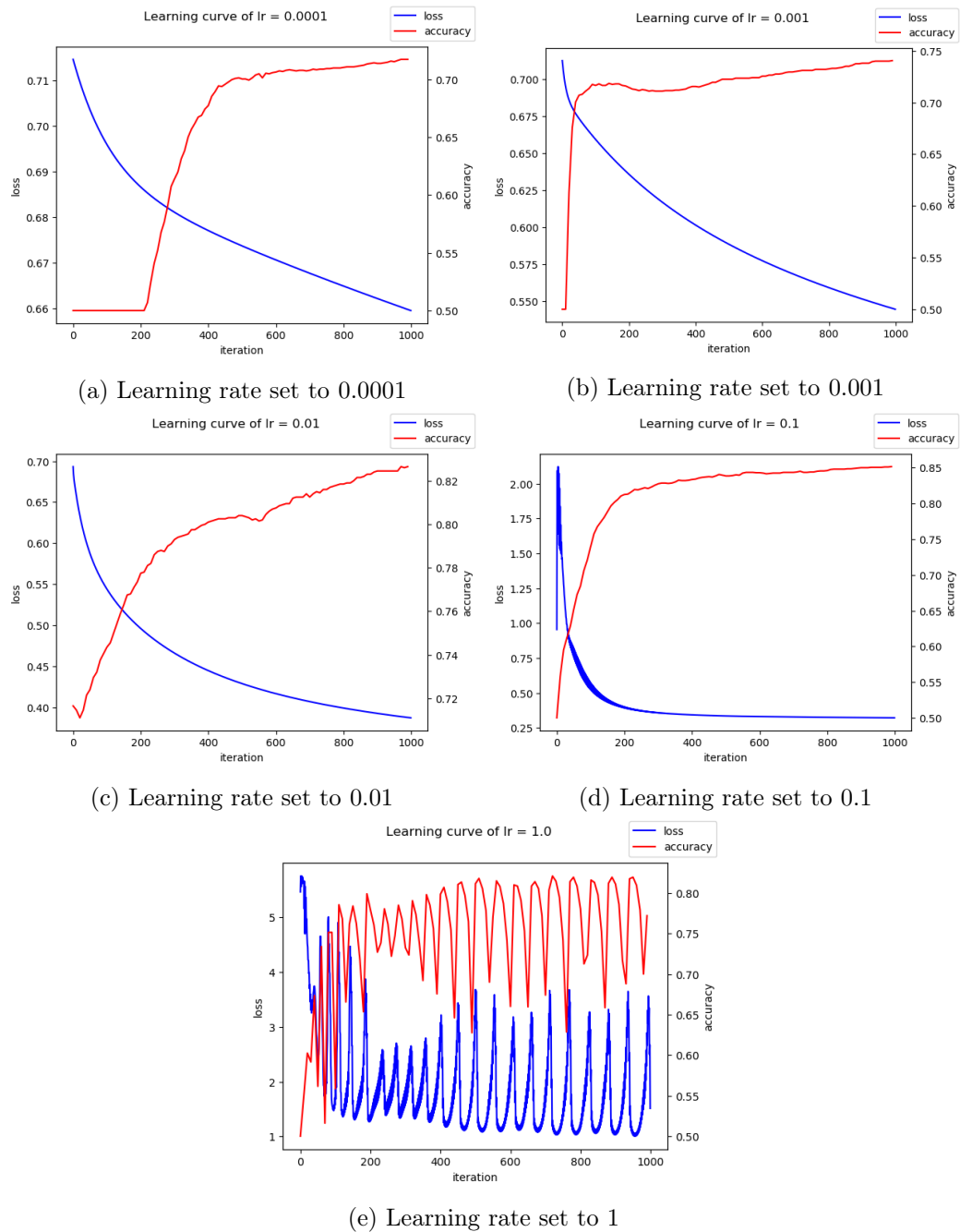


Figure 2: Plot of results from task (e)

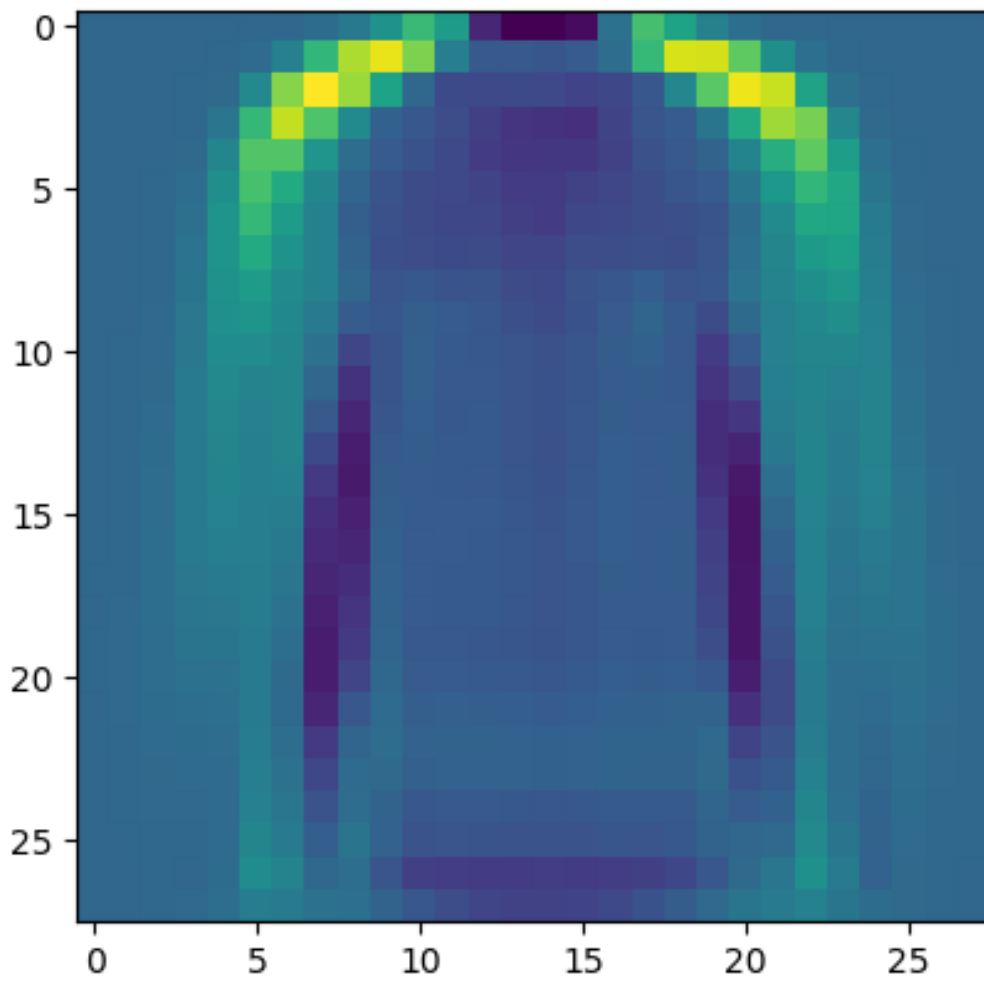


Figure 3: Task f part 1

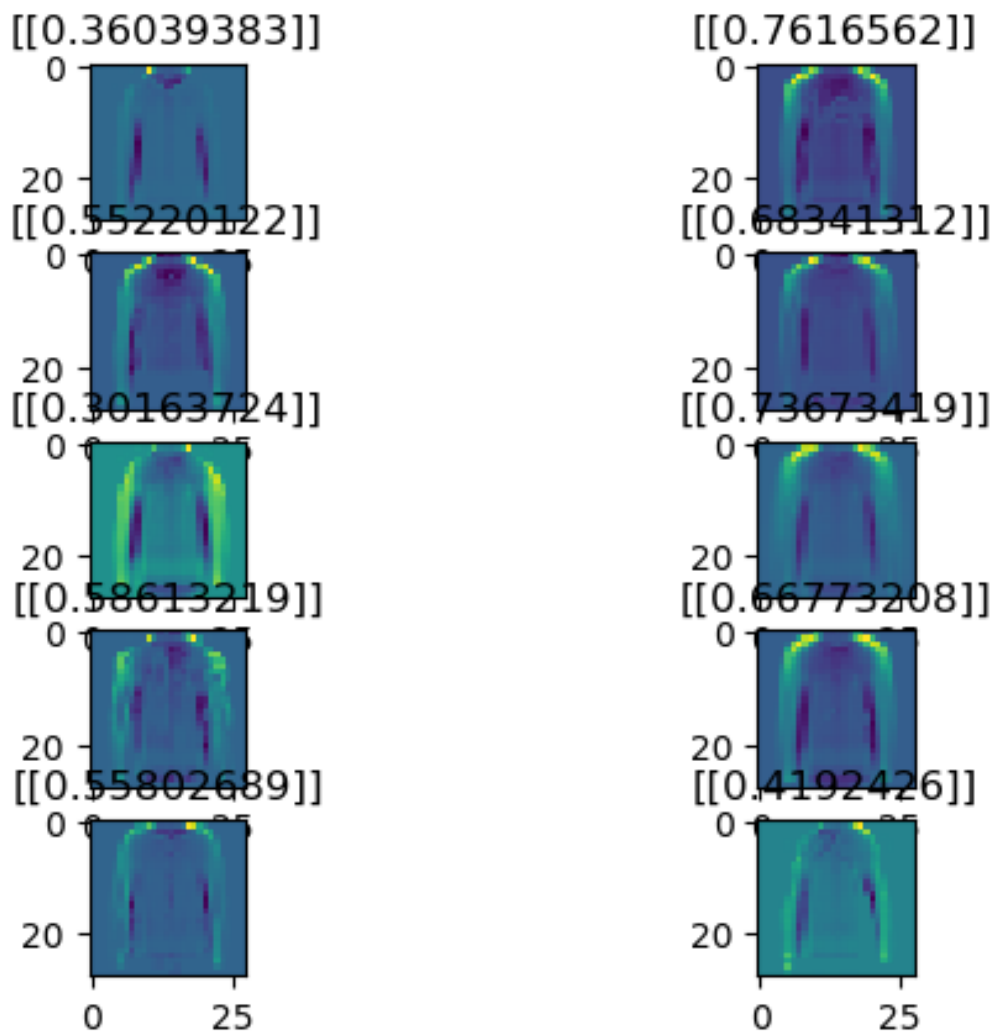


Figure 4: Task f part 2