Artificial Intelligence Methods
Assignment 7
Kacper Multan

You will need to implement the gradient descent algorithm outlined in Equation (19.4) with the loss function in Equation (1) and its derivative derived in Section 22.1.2 Gradients and learning in the textbook.

The algorithm is as follows:

w ← any point in the parameter space
while not converged **do**
  for each w_i **in w do**

$$w_i \leftarrow w_i - \alpha \frac{\partial}{\partial w_i} Loss(\mathbf{w})$$

$$\mathbb{E}[\vec{\boldsymbol{w}}] = \frac{1}{2}\sum_{d\in\mathcal{D}}(t_d - o_d)^2$$

Output layer:

$$\begin{aligned}
\frac{\partial}{\partial w_{3,5}} Loss(h_{\mathbf{w}}) &= \frac{\partial}{\partial w_{3,5}}(y-\hat{y})^2 = -2(y-\hat{y})\frac{\partial \hat{y}}{\partial w_{3,5}} \\
&= -2(y-\hat{y})\frac{\partial}{\partial w_{3,5}} g_5(in_5) = -2(y-\hat{y})g_5'(in_5)\frac{\partial}{\partial w_{3,5}} in_5 \\
&= -2(y-\hat{y})g_5'(in_5)\frac{\partial}{\partial w_{3,5}}\left(w_{0,5} + w_{3,5}a_3 + w_{4,5}a_4\right) \\
&= -2(y-\hat{y})g_5'(in_5)a_3 .
\end{aligned}$$

Hidden layer:

$$\begin{aligned}
\frac{\partial}{\partial w_{1,3}} Loss(h_{\mathbf{w}}) &= -2(y-\hat{y})g_5'(in_5)\frac{\partial}{\partial w_{1,3}}\left(w_{0,5} + w_{3,5}a_3 + w_{4,5}a_4\right) \\
&= -2(y-\hat{y})g_5'(in_5)\,w_{3,5}\frac{\partial}{\partial w_{1,3}} a_3 \\
&= -2(y-\hat{y})g_5'(in_5)\,w_{3,5}\frac{\partial}{\partial w_{1,3}} g_3(in_3) \\
&= -2(y-\hat{y})g_5'(in_5)\,w_{3,5}\,g_3'(in_3)\frac{\partial}{\partial w_{1,3}} in_3 \\
&= -2(y-\hat{y})g_5'(in_5)\,w_{3,5}\,g_3'(in_3)\frac{\partial}{\partial w_{1,3}}\left(w_{0,3} + w_{1,3}x_1 + w_{2,3}x_2\right) \\
&= -2(y-\hat{y})g_5'(in_5)\,w_{3,5}\,g_3'(in_3)x_1 .
\end{aligned}$$

In your report, write down the mean squared error you get on the training and test set using the trained model. Document the hyperparameters you used, such as the learning rate and how parameters are initialized.

To compute the mean squared error for these sets I used the following hyperparameters:

- Learning Rate
- Weights Hidden: Random initialization
- Biases Hidden: Random initialization
- Weights Output: Random initialization
- Biases Output: Random initialization

I did not change the initialization of weights and biases while testing my code, but I have made alterations to the learning rate. I obtained the following results:

```
Learning rate: 0.1
Number of epochs: 100
Epoch 0, Loss: 0.3526427631340674
Epoch 10, Loss: 0.1272636399605748
Epoch 20, Loss: 0.08159641118523814
Epoch 30, Loss: 0.07149795822715556
Epoch 40, Loss: 0.06892868200749529
Epoch 50, Loss: 0.06815186833238375
Epoch 60, Loss: 0.06787713528232937
Epoch 70, Loss: 0.06776844697221544
Epoch 80, Loss: 0.06772195882331652
Epoch 90, Loss: 0.06770054520529942
Average Train: 0.08768662554532393

Epoch 0, Loss: 0.07102940293809101
Epoch 10, Loss: 0.07101781662075311
Epoch 20, Loss: 0.07101037388629455
Epoch 30, Loss: 0.07100476703864275
Epoch 40, Loss: 0.07099997797045948
Epoch 50, Loss: 0.07099555485860932
Epoch 60, Loss: 0.07099129587864382
Epoch 70, Loss: 0.0709871098450604
Epoch 80, Loss: 0.0709829545951412
Epoch 90, Loss: 0.0709788097028904
Average Test: 0.07099725432366853
```

```
Learning rate: 0.05
Number of epochs: 100
Epoch 0, Loss: 0.3526427631340674
Epoch 10, Loss: 0.19877766519814682
Epoch 20, Loss: 0.12919639990517146
Epoch 30, Loss: 0.0973108523849425
Epoch 40, Loss: 0.0824290074577099
Epoch 50, Loss: 0.0753113617170924
Epoch 60, Loss: 0.07179983137085372
Epoch 70, Loss: 0.0700019613690708
Epoch 80, Loss: 0.06904273695176875
Epoch 90, Loss: 0.06850881128870046
Average Train: 0.10687995122721182

Epoch 0, Loss: 0.07157921174413849
Epoch 10, Loss: 0.07138311667693766
Epoch 20, Loss: 0.07125901945317294
Epoch 30, Loss: 0.07117873046580753
Epoch 40, Loss: 0.07112580976570143
Epoch 50, Loss: 0.07109034202377308
Epoch 60, Loss: 0.07106617679403836
Epoch 70, Loss: 0.07104941340569045
Epoch 80, Loss: 0.07103753618954733
Epoch 90, Loss: 0.07102890269768458
Average Test: 0.07115279501190958
```