DataLab Preparation (Week 3, DataLab II, Thursday)

**2. Gradient Descent in Neural Networks**

**2a Describe in your own words how the gradient descent is used to minimize the loss function in a neural network.**

Gradient descent works by repeatedly computing the gradient of a loss function for each layer. This will give us the direction of the steepest increase of the function. Then, it goes in the opposite direction, therefore 'decreasing' the function, and it also indicates by how much it should go in that direction. How fast this change in direction is depends on the learning rate.

**2b Describe the effect of the learning rate on the speed and stability of convergence in gradient descent.**

The higher the learning rate, the more chances there are for the cost function to be higher than 0 (fast convergence). However, if it is too low, then the changes in direction would be tiny and the gradient descent algorithm would move very slowly (slow convergence).

On the contrary, the higher the learning rate, it can cause the training to go haywire, bouncing around and not finding the best solution (unstable convergence). If it is low, the learning rate might actually make the training too slow or get stuck in a not-so-great solution (too stable convergence).

**2c Consider a scenario where a high learning rate causes divergence in the gradient descent process. How this situation can be identified and mitigated?**

If a high learning rate causes divergence in the gradient descent process, it means that the optimisation algorithm overshoots the minimum of the loss function, causing the loss to increase rather than decrease over time.

A telltale sign of this situation is training loss that starts to increase rather than decrease over successive epochs. Another way to identify this situation is plotting the learning curves of the training and validation loss. If there is a sharp increase in the loss values over time instead of a decrease or stabilisation, that means divergence is occurring.

To mitigate the situation, the learning rate can be reduced or a schedule can be implemented so the function takes smaller steps during optimisation. Additionally, regularisation techniques such as L1 and L2 can help to stabilise the process.

**2d Discuss the impact of the weight initialization on the convergence of gradient descent in neural networks.**

Weight initialisation plays a significant role in the convergence of gradient descent. If the initial weights are too small or too large, it can lead to slow convergence or even divergence during training, because the optimisation algorithm may struggle to find the right direction to move towards minimising the loss function.

The choice of weight initialisation can also influence the behaviour of activation functions.

Therefore, improper weight initialisation can easily lead to divergence.

**2e Explain how a stochastic gradient descent works.**

A stochastic gradient descent finds the combination of weight values that yields the smallest possible loss function, and then moves the parameters in the opposite direction from the gradient. Stochastic refers to the fact that each batch of data is drawn at random.

1. A batch is drawn of training samples, x, and their corresponding targets, y\_true.

2. Running the model on x to obtain predictions y\_pred

3. Computing the loss of the model on the batch (the mismatch between y\_pred and y\_true)

4. Computing the gradient of the loss with regard to the model's parameters

5. Moving the parameters in the opposite direction from the gradient, thus reducing the loss on the batch a bit. The learning rate is used here as a scalar factor that modulates the 'speed' of the gradient descent process.