HYPERPARAMETER PARTY

The goals of this challenge are:

- ✓ Experiment with different neural network topologies.
- ✓ Get experience with (automated) hyperparameter tuning.
- ✓ Get experience with neural network technologies like Pytorch or Keras/Tensorflow.
- ✓ Get acquainted with the Avans AI-lab (AAI-lab).

1. Choose a multi-class or multi-label classification problem of your liking, using the internet.
   a. Don't choose a problem that is too big or too complex, as the focus of this challenge is not the problem itself.
   b. For the same reason, ensure that little effort needs to go into data exploration, cleansing and wrangling.
2. Based on logical reasoning choose hyperparameter values for the neural network that you'll use to solve the classification problem.
   a. Hyperparameters are neural network type, neural network depth and width, activation functions, cost function, optimizer, number of epochs, batch size, learning rate, amount of regularization, …
   b. This also involves choosing which hyperparameter will be kept at their default values. For example, only consider fully connected neural networks.
   c. It should be a trade-off between performance, learning time, memory usage. The biggest network might be the best but might be an unnecessary overkill.
3. Train the network and evaluate its performance.
   a. For evaluation, choose appropriate evaluation metric(s).
   b. Learning speed (== convergence speed) is also part of the performance of the neural network. So also choose appropriate metrics to assess this (e.g., graph with loss vs n_epochs). Tensorboard is your good friend.
   c. Part of the evaluation is to ensure that the model is not under/overfitting. You might for example want to experiment with 'early stopping'. Conclusion might be that you need to train longer or shorter, or that you need a simpler or more complex neural network.
4. Perform (automated) hyperparameter tuning.
   a. "Automated" by means of nested for-loops or by means of Optuna to sweep a part of the hyperparameter space.
   b. Go beyond the values that you based on logical reasoning. This means you try some illogical choices, values, or intervals. You might be surprised by the results. If so, try to explain the results and improve your general understanding of neural networks.
   c. Evaluate the performance using the metrics chosen in 3.

Based on the results of step 4, you might need to update your logical reasoning from step 2. Iterate through steps 2 and 4 a couple of times. It will be a journey where your understanding gets better and better.

Write-up your journey in a Jupyter notebook and finish with a short reflection on what you've learned. You can make this challenge as simple or challenging as you want. Don't exaggerate.