

Lab 3 Exercise (Due Friday Feb 7th 2025 10:00 AM CAT)

This lab is based on [chapter 3](#) of the book Modern Computer Vision with Pytorch. You have premium access to this book and other O'Reilly books through your Andrew email. This exercise is based on the [notebook](#) that was already shared with you. You can access it with your Andrew email.

Note: You can access free GPU on Colab, which can be helpful if you want to run more epochs & explore more complex models in this exercise. Alternatively, you can use Kaggle by downloading the colab notebook and uploading it to Kaggle.

Image Representation

- 1) How is the representation of a color image different from that of a color image? Can you print some tensors representing these images? (share your results)

Training a Neural network

- 1) Write a brief comment on the dataset class FMNISTDataset
- 2) Why do we shuffle data in the Dataloader
- 3) When we have batch size of 32, what is the shape of the input to the Neural network and the shape of the output?
- 4) How many parameters (weights & biases) does the given model have? You can calculate by hand or use torch_summary
- 5) What is a loss function? Why do we use cross-entropy loss, and what other losses are there?
- 6) What is the role of the optimizer? Give a list of optimizers you could use in Deep learning.
- 7) Why do we need the learning rate parameter?
- 8) Why do we do batching? What is the difference between mini-batch and multiple-batches
- 9) How often do we do gradient updates if our dataset has 60,000 data points and uses a batch size of 32? What about using a batch size of 1,000?

Scaling

Your dataset is initially made of tensors with numbers between 0 -255; you can scale them (as is done in the notebook) by dividing by 255 so that it's between 0 and 1.

- What is the difference in the results you see?
- Can you explain why we see that difference?

Different batch sizes

Now that we have a batch size of 32, can you plot the results of training & Validation loss and accuracy? Explain your plot.

Try a different batch size:

- What happens when we increase batch size? take note of the training results and the performance of the model (accuracy and loss)

Different Optimizers

- From the optimizers you listed above. Describe 3 of them; how do they work? How do you expect them to affect your model's performance in training? Try those optimizers & show the results, and explain them.

Model Depth & size

- You found the number of parameters of the model in the second section. Now, you will experiment with different model sizes.
- Explore models of different depths that change the number of layers and the number of hidden layers. Run at least four different architectures.
- What is the impact on the training time and performance of the model?

Batch normalization

- What is Batch normalization?
- Using this tutorial (https://github.com/PacktPublishing/Modern-Computer-Vision-with-PyTorch/blob/master/Chapter03/Batch_normalization.ipynb), explore the impact of batch normalization on your training, plot and explain the histograms of the weight values at different layers of your model.