

1-Month Deep Learning Plan (30 minutes/day)

This plan takes you from being proficient with NumPy, SciPy, and scikit-learn to being able to build, train, and evaluate deep learning models using PyTorch, with a focus on applications in medical imaging. Each day is designed for approximately 30 minutes of focused learning and coding.

Week 1 — PyTorch Basics & Tensors

- Day 1: Install PyTorch + setup (check GPU availability)
- Day 2: Tensors vs NumPy (creation, indexing, reshaping)
- Day 3: Autograd (requires_grad, backward, grad)
- Day 4: Simple Linear Regression (manual gradient descent)
- Day 5: Dataset & DataLoader (batching)
- Day 6: Build a feedforward network with nn.Sequential
- Day 7: Review + project (regression on sklearn dataset)

Week 2 — Computer Vision & CNNs

- Day 8: Image Tensors (torchvision.datasets, transforms)
- Day 9: Convolution Layers (nn.Conv2d)
- Day 10: Pooling & Activation (MaxPool2d, ReLU)
- Day 11: Build a CNN (MNIST or CIFAR-10)
- Day 12: Training Loops (loss tracking)
- Day 13: Evaluation & Inference (model.eval)
- Day 14: Review + tweak CNN architecture

Week 3 — Medical Imaging Orientation

- Day 15: DICOM basics (pydicom)
- Day 16: Preprocessing medical images (normalisation, resizing)
- Day 17: Custom Dataset for medical slices
- Day 18: Train a CNN on medical images
- Day 19: Data Augmentation (rotations, flips)
- Day 20: Evaluation metrics (accuracy, sensitivity, specificity)
- Day 21: Review + integrate into one notebook

Week 4 — Segmentation & Transfer Learning

- Day 22: Intro to segmentation (pixel-wise classification)
- Day 23: U-Net basics (architecture diagram)
- Day 24: U-Net mini implementation (lung masks)
- Day 25: Transfer learning (pretrained ResNet)
- Day 26: Fine-tuning for custom dataset
- Day 27: Saving & loading models (torch.save/load)
- Day 28: Deployment basics (ONNX, TorchScript)
- Day 29: Final project start
- Day 30: Final project wrap-up (evaluation & notes)

Extra Tips

- Keep all your code in a GitHub repo to show progression to employers/supervisors.
- Use Kaggle Notebooks for free GPU training time.
- After this month, explore 3D CNNs for volumetric CT/MRI data.
- Combine deep learning with classical medical physics algorithms for niche expertise.