**Week 1: Foundations of Deep Learning**

**Day 1-3: Introduction to Neural Networks**

* **Theory**: Read about the basics of neural networks, perceptrons, and MLPs.
* **Task**: Implement a simple perceptron from scratch using **NumPy**.
* **Dataset**: Start exploring the **Iris dataset** (scikit-learn).
* **Objective**: Understand basic neural network concepts.

**Day 4-5: Forward and Backpropagation**

* **Theory**: Study how forward and backpropagation work in neural networks.
* **Task**: Implement forward and backward passes in a multi-layer perceptron (MLP) using **NumPy**.
* **Dataset**: Work with the **Iris dataset** (train a simple MLP).
* **Objective**: Understand how neural networks learn by adjusting weights.

**Day 6-7: Activation Functions and Loss Functions**

* **Theory**: Study activation functions (ReLU, Sigmoid, Tanh) and loss functions (MSE, Cross-Entropy).
* **Task**: Implement different activation functions and loss functions in Python.
* **Dataset**: Continue working with the **Iris dataset**.
* **Objective**: Learn about the role of activation functions and loss functions in training.

**Week 2: Training Neural Networks Efficiently**

**Day 8-9: Gradient Descent & Optimizers**

* **Theory**: Learn about gradient descent and its variants (SGD, Adam, RMSProp).
* **Task**: Implement gradient descent for a simple linear regression problem.
* **Objective**: Understand how optimization algorithms work in deep learning.

**Day 10-11: Weight Initialization & Regularization**

* **Theory**: Learn about Xavier and He initialization, dropout, and batch normalization.
* **Task**: Implement weight initialization techniques and compare them.
* **Dataset**: Continue training MLP on **Iris dataset** with different regularization methods.
* **Objective**: Improve training speed and prevent overfitting.

**Day 12-13: Hyperparameter Tuning**

* **Theory**: Learn about hyperparameter tuning (learning rates, batch sizes, epochs).
* **Task**: Use **GridSearchCV** or **RandomizedSearchCV** to tune hyperparameters of your MLP.
* **Objective**: Get better model performance with optimal hyperparameters.

**Day 14: Model Evaluation**

* **Theory**: Learn evaluation metrics (Precision, Recall, F1-score, AUC-ROC).
* **Task**: Evaluate your model performance on the **Iris dataset**.
* **Objective**: Understand how to evaluate classification models.

**Week 3: Convolutional Neural Networks (CNNs)**

**Day 15-16: Basics of CNNs**

* **Theory**: Study CNN architecture (Convolution, Pooling, Fully Connected layers).
* **Task**: Implement a basic CNN using **Keras/TensorFlow**.
* **Dataset**: Train on **MNIST dataset** (digit classification).
* **Objective**: Get familiar with CNN building blocks.

**Day 17-18: Advanced CNN Architectures**

* **Theory**: Learn about popular CNN models (AlexNet, VGG, ResNet).
* **Task**: Build a simple CNN and train it on **CIFAR-10 dataset**.
* **Dataset**: **CIFAR-10**.
* **Objective**: Learn how CNN architectures work for image classification.

**Day 19-20: Transfer Learning & Fine-Tuning**

* **Theory**: Study transfer learning and how to fine-tune pre-trained models.
* **Task**: Fine-tune a **pre-trained ResNet** model on **CIFAR-10**.
* **Objective**: Learn how to leverage pre-trained models for faster convergence.

**Day 21: Data Augmentation**

* **Theory**: Learn about data augmentation techniques (rotation, flipping, zoom).
* **Task**: Apply data augmentation to your **CIFAR-10** CNN model.
* **Objective**: Increase the robustness of your model.

**Week 4: Recurrent Neural Networks (RNNs) & LSTMs**

**Day 22-23: Introduction to RNNs**

* **Theory**: Study Recurrent Neural Networks (RNNs) and their applications.
* **Task**: Implement a simple RNN from scratch.
* **Dataset**: Train on a small **sequence prediction task** (e.g., sine wave prediction).
* **Objective**: Understand the architecture and use of RNNs.

**Day 24-25: LSTMs and GRUs**

* **Theory**: Learn about LSTM and GRU networks.
* **Task**: Implement an LSTM model for **sequence classification**.
* **Dataset**: Train on **IMDb Movie Reviews dataset** for sentiment analysis.
* **Objective**: Learn how to use LSTM for sequence data.

**Day 26-27: Sequence Modeling**

* **Theory**: Learn about time-series forecasting and NLP applications.
* **Task**: Train an LSTM model on **stock market prediction** or **NLP** tasks.
* **Dataset**: **Yahoo Finance API** or **IMDb dataset**.
* **Objective**: Understand sequence modeling in deep learning.

**Day 28: Sequence to Sequence Models**

* **Theory**: Learn about sequence-to-sequence (Seq2Seq) models for tasks like translation.
* **Task**: Implement a simple Seq2Seq model using LSTMs or GRUs.
* **Objective**: Build models for tasks requiring variable-length inputs and outputs.

**Week 5: Transformers & Attention Mechanisms**

**Day 29-30: Introduction to Attention**

* **Theory**: Study attention mechanisms in deep learning.
* **Task**: Implement basic attention mechanisms in an NLP task.
* **Dataset**: **IMDB dataset** for sentiment analysis.
* **Objective**: Understand how attention improves model performance.

**Day 31-32: Transformer Architecture**

* **Theory**: Learn about the Transformer model, self-attention, and multi-head attention.
* **Task**: Implement the **Transformer model** for sequence-to-sequence tasks.
* **Objective**: Deep dive into the core of modern NLP models.

**Day 33-34: BERT, GPT, and Transformers in NLP**

* **Theory**: Study BERT, GPT, and transformer-based models for NLP tasks.
* **Task**: Fine-tune **BERT** on **Fake News Dataset**.
* **Dataset**: **Fake News Detection Dataset**.
* **Objective**: Learn how to use BERT for NLP tasks.

**Day 35: Hugging Face and Pre-trained Models**

* **Theory**: Learn about Hugging Face and how to use pre-trained models.
* **Task**: Fine-tune a pre-trained transformer model using Hugging Face.
* **Objective**: Implement state-of-the-art NLP models.

**Week 6: Generative Models (GANs & VAEs)**

**Day 36-37: Introduction to GANs**

* **Theory**: Study the basics of GANs (Generative Adversarial Networks).
* **Task**: Implement a GAN to generate handwritten digits using **MNIST dataset**.
* **Objective**: Learn the concept of adversarial training.

**Day 38-39: Variational Autoencoders (VAEs)**

* **Theory**: Learn about VAEs and how they are used for generative tasks.
* **Task**: Implement a VAE for generating new samples from the **MNIST dataset**.
* **Objective**: Understand the generative capabilities of VAEs.

**Day 40-41: Advanced GAN Architectures**

* **Theory**: Study advanced GAN architectures like **StyleGAN** and **CycleGAN**.
* **Task**: Experiment with **CycleGAN** for image-to-image translation.
* **Objective**: Understand how GANs can be used for creative applications.

**Day 42: Review and Fine-Tuning**

* **Theory**: Review GAN and VAE concepts and their applications.
* **Task**: Fine-tune your GAN or VAE model for better performance.
* **Objective**: Gain hands-on experience with fine-tuning generative models.

**Week 7: Deep Learning for Real-World Applications**

**Day 43-44: Object Detection (YOLO, Faster R-CNN)**

* **Theory**: Study object detection using YOLO and Faster R-CNN.
* **Task**: Implement YOLO or Faster R-CNN for object detection on **COCO dataset**.
* **Objective**: Learn object detection and localization.

**Day 45-46: Speech Recognition**

* **Theory**: Learn about deep learning models for speech recognition (e.g., DeepSpeech).
* **Task**: Implement a basic speech-to-text model.
* **Dataset**: **LibriSpeech dataset**.
* **Objective**: Understand how deep learning can be applied to speech data.

**Day 47-48: Recommendation Systems**

* **Theory**: Study deep learning-based recommendation systems (Neural Collaborative Filtering).
* **Task**: Build a recommendation system using **MovieLens dataset**.
* **Objective**: Learn the principles of recommendation systems.

**Day 49-50: Review and Final Projects**

* **Theory**: Review all deep learning concepts.
* **Task**: Start your **Capstone Project** (e.g., object detection, NLP, or recommendation system).
* **Objective**: Build and deploy a complete deep learning application.

**Week 8: Capstone Project**

* **Complete a Capstone Project** based on your interest area (Image Processing, NLP, Recommendation, etc.).
* **Task**: Implement the project, test it on real-world data, and document your work.