

Artificial Neural Network

Description:

Artificial neural networks (ANNs) are computer systems that are modelled after the biological neural networks that make up animal brains. It processes data and creates patterns for use in decision-making in the same way that the human brain does. Using the most up-to-date frameworks, you'll learn Artificial neural networks, Transfer Learning, and more.

Instructors:

Sunny Bhaveen Chandra

Duration:**Language:**

English

Price:

25000

Requirements:

Basic Programming Knowledge, A System with a decent internet connection, Your dedication

Features:

Source code, Roadmap, Quizzes , Assignments, Downloadable resources, Completion certificate

Learn:

Neural Network, Perceptron, Evaluation of Neural Network, Maths behind concepts of Neural Networks, Back Propagation, Problems faced while training Neural Network and its solution, Building solutions

Curriculum:

- Introduction:
- AI | Deep Learning | Evolution of ANNs:
 - Introduction
 - Introduction
- Perceptron:
- Perceptron Implementation:
- Perceptron Implementation | Python scripting and packaging | Modular coding:
- Python logging basics in previous codes, docstrings:
- Python packaging | Github Actions | PyPI:
- Neural Network:
- ANN Derivation:
- ANN implementation using tf.keras:
- ANN implementation using python scripting:
- ANN implementation using python scripting continued:
- Callbacks in Tensorflow:
- ANN with Callbacks | Tensorboard | Early Stopping | Model Checkpointing:
- Mathematics in DL:
- THEORY: Vectors:
- THEORY Differentiation | Partial Diff | Gradients | Ascent and Descent:
- THEORY Problems in training NN | Vanishing and Exploding gradients:
- Tensorflow Framework:
- TF 2.x low-level API:
- TF 2.x low-level API PART 2:
- Activation Function:
- Activation Function - Started:

- Activation Function -continued:
- Activation function final:
- Weight initialization, Transfer learning, Batch Normalization:
- Weight initialization and Transfer learning:
- Batch Normalization: Theory and Practical:
- MLFlow:
- Optimizers, Regularization and Loss function:
- Fast Optimizers | Momentum Optimization:
- NAG:
- AdaGrad:
- RMS Prop | Adam:
- Regularization | Dropout | Loss function: