Machine Learning, Deep Learning & Generative AI Bootcamp

Duration: 15 Days (120 Hours)
DC: Theory + Labs (Open Source + Koenig DC)
Prerequisite: Python fundamentals (mandatory)

Course Objectives

The objectives of this course are to:

1. Build a **strong foundation in Machine Learning concepts and workflows** required for real-world problem solving.
2. Enable learners to **analyze, preprocess, and model data** using Python and standard ML libraries.
3. Introduce **Deep Learning fundamentals**, including neural networks and modern training techniques.
4. Provide conceptual and hands-on understanding of **Natural Language Processing and Transformer architectures**.
5. Develop practical skills in **Generative AI and Large Language Models (LLMs)**.
6. Teach effective **Prompt Engineering techniques** to improve accuracy and reliability of LLM outputs.
7. Introduce **Retrieval-Augmented Generation (RAG)** for building knowledge-grounded AI systems.
8. Familiarize learners with **basic Agentic AI concepts** and how agents differ from traditional pipelines.
9. Prepare participants for **industry-relevant AI roles** by combining theory, labs, and applied use cases.

Course Learning Outcomes

By the end of this course, participants will be able to:

1. Apply **Python programming** for data preprocessing, analysis, and machine learning tasks.
2. Build and evaluate **supervised and unsupervised machine learning models** for regression, classification, and clustering problems.
3. Perform **feature engineering, model tuning, and performance evaluation** using industry best practices.
4. Design and train **deep learning models** using neural networks for real-world datasets.
5. Understand and implement **NLP pipelines**, including text preprocessing and vectorization.

6. Explain the working principles of **Transformers and attention mechanisms**.
7. Develop **Generative AI applications** using Large Language Models.
8. Create effective prompts using **prompt engineering strategies** such as few-shot and chain-of-thought prompting.
9. Build **RAG-based applications** that combine LLMs with external knowledge sources.
10. Understand **Agentic AI workflows**, including planning, tool usage, and basic agent reasoning.

DAY 1 – Python for Machine Learning (Foundations)

- Python basics for ML
- Data types, control flow
- Functions, file handling
- NumPy fundamentals
- Lab: Python for ML workflows

DAY 2 – Data Handling & EDA

- Pandas for data manipulation
- Data preprocessing techniques
- Missing values & outliers
- Data visualization
- Lab: EDA on real dataset

DAY 3 – Machine Learning Fundamentals

- What is Machine Learning
- Supervised vs unsupervised learning
- ML workflow & lifecycle
- Bias–variance tradeoff
- Lab: ML pipeline overview

DAY 4 – Supervised Learning (Regression)

- Linear & Polynomial Regression
- Loss functions
- Model evaluation (MSE, RMSE, R^2)
- Overfitting & regularization
- Lab: Regression use case

DAY 5 – Supervised Learning (Classification)

- Logistic Regression
- KNN, Decision Trees
- Confusion matrix, accuracy, precision, recall
- ROC & AUC
- Lab: Classification problem

DAY 6 – Unsupervised Learning

- Clustering concepts
- K-Means & Hierarchical clustering
- Dimensionality reduction (PCA – intuition)
- Lab: Customer segmentation

DAY 7 – Model Tuning & ML Best Practices

- Feature engineering
- Hyperparameter tuning
- Cross-validation
- Model comparison
- Lab: Improving ML performance

DAY 8 – Introduction to Deep Learning

- Why Deep Learning?
- ML vs DL
- Neural network fundamentals

- Perceptron & MLP
- Activation functions
- Lab: Build first neural network

DAY 9 – Deep Learning with TensorFlow / Keras

- ANN architecture
- Loss functions & optimizers
- Training & validation
- Overfitting in DL
- Lab: ANN for classification

DAY 10 – Introduction to NLP

- NLP overview & applications
- Tokenization & text preprocessing
- POS tagging & NER
- Vectorization basics
- Lab: Text analytics with NLP

DAY 11 – Transformers & Attention

- Limitations of classical NLP
- Encoder–Decoder architecture
- Attention mechanism
- Transformer overview
- Lab: Transformer-based task (T5)

DAY 12 – Introduction to Generative AI

- What is Generative AI
- Generative vs predictive models
- Text generation concepts
- Ethical concerns & hallucination

- Lab: Text generation using LLM

DAY 13 – Prompt Engineering for LLMs

- Prompt anatomy
- Zero-shot & few-shot prompting
- Chain-of-thought reasoning
- Prompt grounding
- Lab: Prompt optimization exercises

DAY 14 – Retrieval-Augmented Generation (RAG)

- Why RAG is required
- RAG architecture
- Document loading & chunking
- Embeddings & vector databases
- Lab: Build a simple RAG system

DAY 15 – Introduction to Agentic AI

- What is Agentic AI
- RAG vs Agentic RAG
- Agent lifecycle
- Tool calling basics
- Simple agent workflow
- Lab: Mini agentic demo