Python, Machine Learning & Deep Learning Foundations

40-Hour Course — Table of Contents

Target Audience: Beginners to intermediate learners (basic programming logic helpful but not required) **Format:** Hands-on, code-along sessions with mini-exercises after each topic

---

Module 1: Python Programming Fundamentals (8 hours)

Hour	Topic
1	Python setup (Anaconda/Jupyter/Colab), syntax, variables, data types
2	Operators, conditionals (if/elif/else), loops (for/while)
3	Strings, lists, tuples, sets, dictionaries
4	Functions, arguments, lambda functions, scope
5	List/dict comprehensions, error handling (try/except)
6	File handling (read/write CSV, text, JSON)
7	Object-Oriented Programming basics (classes, objects, inheritance)
8	Modules, packages, virtual environments; mini-project (e.g., simple data parser)

Module 2: Python for Data Science — NumPy & Pandas (6 hours)

Hour	Topic
9	NumPy arrays, indexing, slicing, broadcasting
10	NumPy operations: aggregation, reshaping, random module
11	Pandas Series & DataFrames: creation, indexing, selection
12	Data cleaning: handling missing values, duplicates, data types

Hour	Topic
13	GroupBy, merging, joining, pivot tables
14	Reading/writing real datasets (CSV, Excel, SQL basics); hands-on exercise

---

Module 3: Data Visualization (3 hours)

Hour	Topic
15	Matplotlib basics: line, bar, scatter, histogram plots
16	Seaborn: distribution plots, heatmaps, pair plots
17	Building a simple Exploratory Data Analysis (EDA) report on a real dataset

---

Module 4: Machine Learning Fundamentals (10 hours)

Hour	Topic
18	What is ML? Types: supervised, unsupervised, reinforcement (overview)
19	Linear Regression: theory + implementation (scikit-learn)
20	Logistic Regression & classification basics
21	Decision Trees & Random Forests
22	K-Nearest Neighbors (KNN) & Support Vector Machines (SVM)
23	Naive Bayes classifier
24	Ensemble methods: Bagging & Boosting (e.g., XGBoost overview)
25	Unsupervised learning: K-Means clustering, hierarchical clustering
26	Feature engineering: scaling, encoding, feature selection
27	End-to-end mini ML project (e.g., house price or churn prediction)

Module 5: Model Evaluation & Improvement (4 hours)

Hour	Topic
28	Train/test split, cross-validation
29	Evaluation metrics: accuracy, precision, recall, F1, ROC-AUC, RMSE
30	Overfitting/underfitting, bias-variance tradeoff
31	Hyperparameter tuning: GridSearchCV, RandomizedSearchCV

Module 6: Introduction to Deep Learning (7 hours)

Hour	Topic
32	Neural networks basics: neurons, weights, activation functions
33	Forward propagation, backpropagation, gradient descent (intuition, no heavy math)
34	Optimizers (SGD, Adam) and loss functions — intuition-level
35	Building your first neural network with TensorFlow/Keras
36	Introduction to CNNs (image data) — concept + simple demo
37	Introduction to RNNs (sequential/text data) — concept + simple demo
38	Model saving/loading and basic deployment intuition (e.g., simple API/Streamlit demo)

Module 7: Capstone Project & Wrap-Up (2 hours)

Hour	Topic
39	Capstone project: apply ML/DL pipeline end-to-end on a chosen dataset
40	Project presentations, best practices, career guidance, Q&A

Course Outcomes

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

- Write Python programs confidently for data tasks
- Clean, explore, and visualize real-world datasets
- Build, evaluate, and tune a wide range of classical ML models (including ensembles)
- Understand the fundamentals of neural networks and basic DL architectures (CNN/RNN)
- Complete an end-to-end ML/DL project from raw data to results

Suggested Tools/Stack

Python 3.x, Jupyter/Colab, NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn, TensorFlow/Keras