

ML Engineer: Detailed Master Syllabus (2026)

LEVEL 1: THE FOUNDATION (Weeks 1-8)

> Python Mastery

Focus on OOP (Classes, Inheritance), Decorators, Context Managers, and Type Hinting (Python 3.12+).

> Math for ML

Linear Algebra: Dot products, Eigendecomposition. Calculus: Chain Rule, Gradients. Stats: Distributions (Normal, Bernoulli), Bayes Theorem.

> Data Stack

NumPy: Vectorization over loops. Pandas: Lambda mapping, Joins/Merges. Matplotlib/Seaborn: Heatmaps and Distribution plots.

LEVEL 2: CLASSICAL MACHINE LEARNING (Weeks 9-16)

> Data Preprocessing

Feature Scaling (StandardScaler), Imputation, Encoding, and Feature Selection.

> Supervised Learning

Linear/Logistic Regression, SVMs, Decision Trees, and Gradient Boosted Trees (XGBoost/LightGBM).

> Unsupervised Learning

K-Means, PCA (Dimensionality Reduction), and Anomaly Detection.

> Model Selection

Cross-Validation (K-Fold), Hyperparameter Tuning (GridSearch, Optuna), and Evaluation Metrics (F1, AUC-ROC).

LEVEL 3: DEEP LEARNING & NEURAL NETS (Weeks 17-24)

> Deep Learning Theory

Backpropagation, Stochastic Gradient Descent (SGD), Optimizers (Adam, RMSprop), and Regularization (Dropout, L2).

> PyTorch Framework

Tensors, Autograd, Dataset/DataLoader classes, and Custom Module construction.

> Computer Vision

CNN Architectures (ResNet, EfficientNet) and Data Augmentation.

> NLP & Transformers

Word Embeddings, Attention Mechanisms, and the Transformer Architecture (Attention is All You Need).

LEVEL 4: MLOPS & DEPLOYMENT (Weeks 25-32)

> Model Serving

FastAPI, BentoML, or TorchServe. Building RESTful endpoints for model inference.

> Containership

Docker (Writing Dockerfiles), Kubernetes (Basic orchestration), and CI/CD pipelines.

> LLM Engineering

RAG (Retrieval Augmented Generation), Vector Databases (Pinecone/Milvus), and Fine-tuning via PEFT/LoRA.