**Stage 1: Python & Math Foundations**

**Goal:** Programming, math, and ML-ready skills

* **Python Basics**: Loops, Functions, OOP, List/Dict comprehension
* **NumPy & Pandas**: Arrays, Broadcasting, DataFrame ops, GroupBy, Merge
* **Data Cleaning & Feature Prep**: Missing values, Outliers, Encoding, Scaling
* **Data Visualization**: Matplotlib, Seaborn, Pairplots, Histograms, Heatmaps
* **Math in Code**:
  + Linear Algebra essentials (Matrices, Eigenvalues, Dot Products)
  + Calculus essentials (Derivatives, Gradients)
  + Probability & Statistics basics (Mean, Variance, Correlation, Bayes theorem)
* **Colab/Kaggle Workflow**: GPU usage, Data upload/download, Version control

**Stage 2: Core Machine Learning**

**Goal:** Classical ML algorithms and their intuition

**Supervised Learning**

* Regression: Linear, Multiple, Polynomial, Regularization (Ridge, Lasso, ElasticNet)
* Classification: Logistic Regression (Binary/Multiclass), KNN, Decision Tree, Random Forest
* Advanced Algorithms: SVM, Gradient Boosting, AdaBoost, XGBoost, LightGBM, CatBoost
* Model Evaluation: Accuracy, Precision, Recall, F1, ROC-AUC, Confusion Matrix
* Model Validation: Train/Test split, K-Fold, Hyperparameter tuning

**Unsupervised Learning**

* Clustering: K-Means, Hierarchical, DBSCAN
* Dimensionality Reduction: PCA, t-SNE, LDA
* Evaluation & Interpretability: Silhouette Score, Davies-Bouldin, SHAP, LIME

**Semi & Self-Supervised Learning (Intro)**

* Autoencoders, Contrastive Learning

**Stage 3: Applied ML / Projects**

* End-to-End ML Projects: Titanic, House Prices, Heart Disease, Customer Segmentation
* Feature Engineering & Selection
* Hyperparameter Tuning & Cross-Validation
* Model Deployment Basics (Streamlit / Flask / API Serving)

**Stage 4: Deep Learning Foundations**

**Goal:** Neural Networks, optimization, and training techniques

* Neural Networks basics & Backpropagation
* Activation Functions (ReLU, Sigmoid, Tanh), Loss Functions (MSE, Cross-Entropy)
* Optimization: Gradient Descent, Adam, RMSProp
* Regularization: Dropout, L2, Early Stopping
* Normalization & Data Augmentation (BatchNorm, Image Augmentation)
* Transfer Learning: Pretrained CNNs (VGG, ResNet, EfficientNet)
* Projects: MNIST, CIFAR-10 Classification

**Stage 5: Computer Vision**

* Image Processing: OpenCV, Histograms, Edge Detection, Augmentation
* CNN Foundations: Conv Layers, Pooling, Flattening, Fully Connected Layers
* Object Detection Concepts: YOLO, SSD, R-CNN
* Advanced Vision Models: Swin Transformer, CvT, CNN + Transformer comparison
* Hands-on Projects: Image Classification, Object Detection

**Stage 6: NLP (Natural Language Processing)**

* Text Preprocessing: Tokenization, Lemmatization, Stopwords, Padding
* Word Embeddings: Word2Vec, GloVe, FastText
* Sequence Models: RNN, GRU, LSTM, seq2seq, Attention
* Transformers: BERT, GPT, Fine-tuning, LoRA
* NLP Applications: Sentiment Analysis, Translation, Summarization, QA, NER

**Stage 7: Advanced AI Concepts**

* Foundation Models Overview: BERT, GPT, ViT
* Responsible AI: Bias detection, Safety, Explainability
* Deployment & Serving: APIs, Demos, Model monitoring
* End-to-End Projects integrating CV + NLP

**Stage 8: Capstone / Real-World Projects**

* Multi-modal AI (Vision + Text)
* Kaggle Competitions
* Building Portfolio: GitHub + Deployment + Blog explanation

**Suggested Learning Sequence**

1. **Stage 1:** Python + Math + Data Prep (2–4 weeks)
2. **Stage 2:** Core ML (Supervised + Unsupervised + Semi-Supervised) (4–6 weeks)
3. **Stage 3:** ML Projects (2–4 weeks)
4. **Stage 4:** Deep Learning Foundations (4–6 weeks)
5. **Stage 5:** Computer Vision (3–5 weeks)
6. **Stage 6:** NLP & Transformers (4–6 weeks)
7. **Stage 7:** Advanced AI Concepts & Responsible AI (2–4 weeks)
8. **Stage 8:** Capstone Projects & Kaggle Competitions (Ongoing)