1. **Mathematical Fundamentals:**

**Linear Algebra:**

* Course: [18.06 Linear Algebra](https://web.mit.edu/18.06/www/)
* Vectors, matrices, eigenvalues, SVD, matrix calculus

**Stats & Probability:**

* Course: [Introduction to Probability and Statistics | Mathematics | MIT OpenCourseWare](https://ocw.mit.edu/courses/18-05-introduction-to-probability-and-statistics-spring-2022/)
* Distributions, Bayes' theorem, hypothesis testing, statistical inference

**Calculus:**

* Course: [3Blue1Brown - The Essence of Calculus](https://www.3blue1brown.com/lessons/essence-of-calculus)
* Partial derivatives, gradients, optimization basics

1. **Python Foundations:**

**Python:**

* Course: [Python Data Science Handbook | Python Data Science Handbook](https://jakevdp.github.io/PythonDataScienceHandbook/)
* NumPy, Pandas, Matplotlib for data manipulation and visualization
* Look into basics of tenserflow, pytorch, sci-kit learn

**Data Structures & Algorithms Review:**

* Sources: GeeksforGeeks for theory and explanations, LeetCode for questions
* Refresh trees, graphs, dynamic programming.

1. **Classical ML:**

**Theory:**

* Course: [Stanford Engineering Everywhere | CS229 - Machine Learning](https://see.stanford.edu/Course/CS229)
* Supervised learning, unsupervised learning, and model evaluation

**Implementation:**

* Course: [Hands on Machine-Learning with Scikit : O'Reilly - Aurelien Geron : Free Download, Borrow, and Streaming : Internet Archive](https://archive.org/details/handson-machine-learning-with-scikit-2-e-1)
* Regression, classification, clustering, dimensionality reduction

1. **Deep Learning:**

**Neural Network Fundamentals:**

* Course: [Deep Learning](https://www.deeplearningbook.org/)
* Implement basic neural networks, understand backpropagation
* Create projects using TensorFlow/Pytorch

1. **AI Specialized Areas:**

**Computer Vision:**

* Course: [Stanford University CS231n: Deep Learning for Computer Vision](https://cs231n.stanford.edu/)
* Image classification, object detection, segmentation
* OpenCV, advanced CNN architectures

**Natural Language Processing (LLMs):**

* Course: [Stanford CS 224N | Natural Language Processing with Deep Learning](https://web.stanford.edu/class/cs224n/)
* Transformer architecture, attention mechanisms
* BERT, GPT understanding, fine-tuning pre-trained models

**Reinforcement Learning:**

* Course: [CS 285](https://rail.eecs.berkeley.edu/deeprlcourse/)
* David Silver's RL course or Berkeley CS285
* Q-learning, policy gradients, actor-critic methods