

AI / ML Roadmap in 6 steps

Do them in order. Each step lists what to learn and free places to learn it.

1

Math & Stats basics

The foundation of every model. You don't need a degree, just the working basics.

Linear Algebra

Probability

Statistics

Calculus

Free: 3blue1brown.com (Essence of Linear Algebra) · khanacademy.org · StatQuest on YouTube

2

Python + its libraries

Learn Python first, then the three libraries you'll use daily.

NumPy

Pandas

Matplotlib

Free: kaggle.com/learn (Python, Pandas) · numpy.org · pandas.pydata.org

3

Machine Learning

Understand the main types, then build and evaluate models with scikit-learn, because this is what's used in the real world.

Supervised

Unsupervised

Regression

Classification

Clustering

Free: scikit-learn.org (user guide) · kaggle.com/learn (Intro to ML) · Andrew Ng's ML Specialization (Coursera, audit for free)

4

Deep Learning

Neural networks, then images and sequences, using TensorFlow or PyTorch, so you can work with images, text and audio.

Neural Networks

CNN

RNN

TensorFlow / PyTorch

Free: pytorch.org/tutorials · tensorflow.org/tutorials · fast.ai

5

Pick a specialization

Specializing helps you stand out in the industry.

NLP

GenAI

Computer Vision

Free: huggingface.co/learn (NLP, LLM and Diffusion courses) · pytorch.org/vision tutorials

6

Build projects

Practice builds confidence. Put every project on GitHub with a proper README: what it does, how to run it, results and screenshots.

ML models

Chatbots

Image classifiers

Practice: kaggle.com (datasets and competitions) · github.com

PROJECT IDEAS (EASY → HARD)

1. House price predictor (regression)
2. Spam / not-spam classifier
3. Customer segmentation (clustering)
4. Handwritten digit recognizer (CNN)
5. A chatbot over your own notes (GenAI)
6. An image classifier web app you deploy

A GOOD README HAS

Problem in one line · dataset and where it came from · approach (model + why) · results (metrics) · how to run it · screenshots or a demo link

COMMON MISTAKES

- Jumping to deep learning before basic ML
- Only watching tutorials, never building
- Skipping evaluation (always hold out test data)
- Projects without a README