SKILLS

A GUIDE TO self-studying Machine Learning

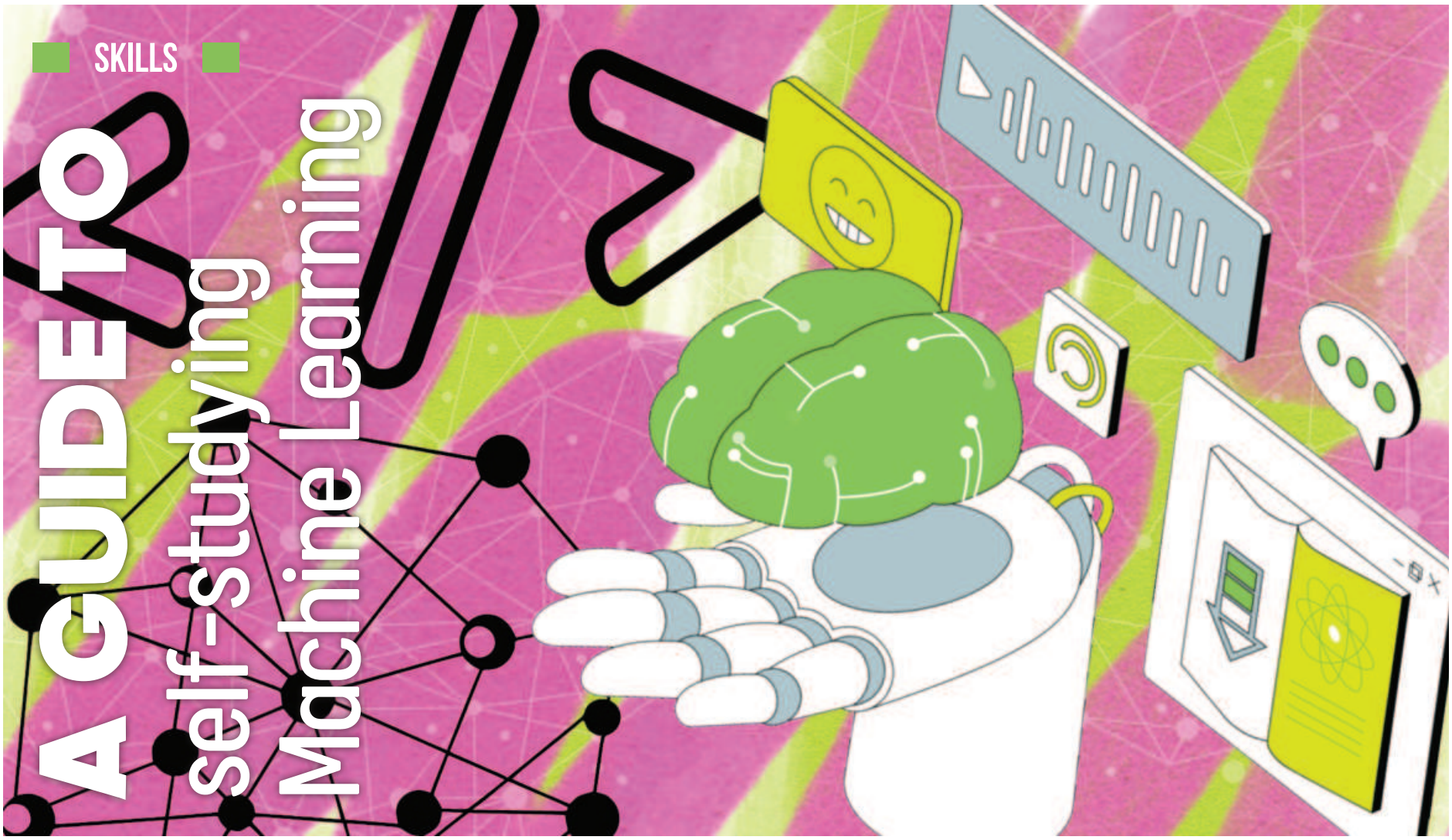

ILLUSTRATION: **ABIR HOSSAIN**

BIPRA PRASUN DAS

Over the past decade or so, few fields have grown as explosively as Artificial Intelligence (AI). One of the most important domains of AI is Machine Learning (ML) – a branch of AI that involves enabling computers to learn patterns and make decisions from data, without being explicitly programmed for each task.

From diagnosing diseases based on medical scans, predicting stock trends, filtering spam emails, to training chatbots like ChatGPT, ML has a wide range of applications. Self-learning ML can sound like a very daunting challenge. With a bit of patience and consistent effort, however, it is very much possible. It must be noted, though, that everyone's learning pattern is different, and it is important to figure out what works best for you.

Firstly, you must become fluent in Python. Begin by installing the latest version of Python and setting up a code editor such as Visual Studio Code. Familiarise yourself with fundamental concepts such as loops, functions, data structures (lists, tuples, dictionaries), and list comprehensions by working through concise tutorials on resources like W3Schools.

Once you feel confident enough, you can spend some time building small projects to apply your knowledge. A good place to look for beginner projects is the GeeksforGeeks website. On a side note, it is important to have a solid grasp of some maths concepts. For probability and statistics, you can refer to the book *Probability and Statistics for Engineers and the Scientists* by Anthony J Hayter and videos from *The Organic Chemistry Tutor* on YouTube.

Next, for Neural Networks, *Khan Academy's* "Intro to AI" playlist and *3Blue1Brown's* "Neural Network" playlist should cover everything you need to know. For Linear Algebra, the book *Elementary Linear Algebra* by Howard Anton and Chris Rorres is a good resource, alongside *3Blue1Brown's* "Linear Algebra" playlist on YouTube if you are more of a visual learner.

Now that we have covered the fundamentals, there are some libraries that will make your coding experience easier. Libraries are already written pieces of code that you can take and use as your own. Install packages via pip and then shift your focus to the following. Pandas for manipulating tabular data; practice loading, cleaning (removing empty rows), reshaping, and summarising datasets. Additionally, you can opt for NumPy for numerical operations and matrix manipulations, which is the backbone of most ML computations. And finally, you may use Matplotlib to generate informative graphs, turning abstract results into visual stories.

A good way to begin is by following hands-on tutorials. Daniel Bourke's TensorFlow YouTube series is beginner-friendly and highly practical. Andrej Karpathy's "Neural Networks: Zero to Hero" breaks down how neural networks work and helps you build them from scratch. Start with basic models like a simple neural network.

A good start is revisiting W3Schools or other concise tutorials for basic syntax. Then, download a classic dataset (for example, the Iris flower dataset on Kaggle) and practise a full pipeline: load the data, clean it, compute summary statistics (mean, mode), slice and reshape arrays, and produce clear visualisations. Give yourself some time to become proficient at this. After some time, you should be comfortable reshaping real-world data with Pandas, performing array operations in NumPy, and making charts that provide insights about your data. When you can handle these tasks without guidance, you'll be poised to tackle the core of machine learning.

To do that, you'll need to use a machine learning framework. The two most popular options are TensorFlow and PyTorch. These are powerful libraries that make it easier to create and train machine learning models without writing every algorithm from scratch. Both frameworks are excellent, and they offer the same core functionality.

A good way to begin is by following hands-on tutorials. *Daniel Bourke's* TensorFlow YouTube series is beginner-friendly and highly practical. *Andrej Karpathy's* "Neural Networks: Zero to Hero" breaks down how neural networks work and helps you build them from scratch. Start with basic models like a simple neural network. As you go through the code, pause and look up what each line does. This might mean asking ChatGPT, reading documentation, or watching quick videos to understand the maths and logic behind each concept. Other good resources are: Andrew Ng's Machine Learning on Coursera (Free), Andrew Ng's "deeplearning.ai" course on Coursera (Paid), Fast.ai Deep Learning Course (part 1) (Free), Fast.ai Deep Learning Course (part 2) (Free).

As for books, the most popular ones are *Artificial Intelligence – A Modern Approach* and *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.*

Once you have made it this far, you can start by applying your knowledge to various projects. You may find ideas for projects in GitHub's project-based-learning repository. Other resources are Huggingface and Google Colab.

Once you have learnt basic to intermediate ML, you can now choose to specialise in domains like Natural Language Processing (NLP), Generative AI, Computer Vision, etc. However, all of that is beyond the scope of this article. Learning ML is a process that can be both challenging and rewarding. With the right approach and understanding of what works for you, it is possible to self-study Machine Learning.

Bipra Prasun Das studies Computer Science and Engineering at North South University