Achref Jaziri, Konrad Wartke, Prof. Dr. Visvanathan Ramesh

Pattern Analysis & Machine Intelligence Praktikum: MLPR-SS/23

Week 1: part 1: Introduction
part 2: Naive Bayes classifier + Logistic Regression

Lecture requirements

- Sessions per week, 8 credit points
- Final class group projects. Will start at the end of the lecture period with presentations at the end of the semester (we will choose a date together later).

- Each Monday session consists of a traditional lecture, followed by an introduction to the practical assignment.
- Practical assignments are online notebooks that need to be filled with code. Solutions will be uploaded by the end of week and discussed on Friday.
- The Friday session will also include presentations from our lab members to prepare the students for the projects.
- We will use Google's Collaboratory for cloud computation. If you have a Gmail, android etc. account, then it is the same account you already have.
- If you don't want to use Colab (because you don't want a google account), you can execute the notebooks locally, we advise a Linux or Mac OS (and will provide no Windows support). However, the later deep learning content will be close to impossible to run on your local laptops.

- **All lecture materials will be shared on the Moodle Page:**

Schedule

17.04 - week 1:
General introduction to machine learning and course utilities. + Classification with Naive Bayes + Logistic regression and gradient descent
Practical assignments: (optional) python review notebook. Spam message identification + Gradient descent variants and Titanic survivor prediction

24.04 - week 2:
Introduction to neural networks: multi-layer perceptron (MLP), convolutional neural network (CNN). + ML Frameworks
Practical assignment: (Optional): MLP classification of fashion images with pure Numpy + Fashion image classification and reading traditional Japanese character PyTorch.

01.05: week 3. Holiday
Practical assignment: No Practical Assignment

08.05 - week 4:
Random Forests.
Practical assignment: Revisiting Titanic survivor prediction + an additional challenge (San Francisco Crime).

15.05 - week 5:
Neural sequence models, recurrent neural networks.
Practical assignment: Shakespeare poetry text generation.

22.05 - week 6:
Transformers and Attention Mechanisms (
Practical assignment: Text Generation with Transformers

29.05 - week 7
Lecture: Holiday
Practical Assignment: No Practical Assignment

05.06 week 8:
Lecture: Time Series Forecasting with NN vs Classical Methods.
Practical assignment: TBD

12.06 - week 9:
Lecture: Introduction to unsupervised learning. Clustering with k-means. Principal and independent component analysis (PCA and ICA).
Practical assignment: Finding clusters in generated distributions and signal source separation.

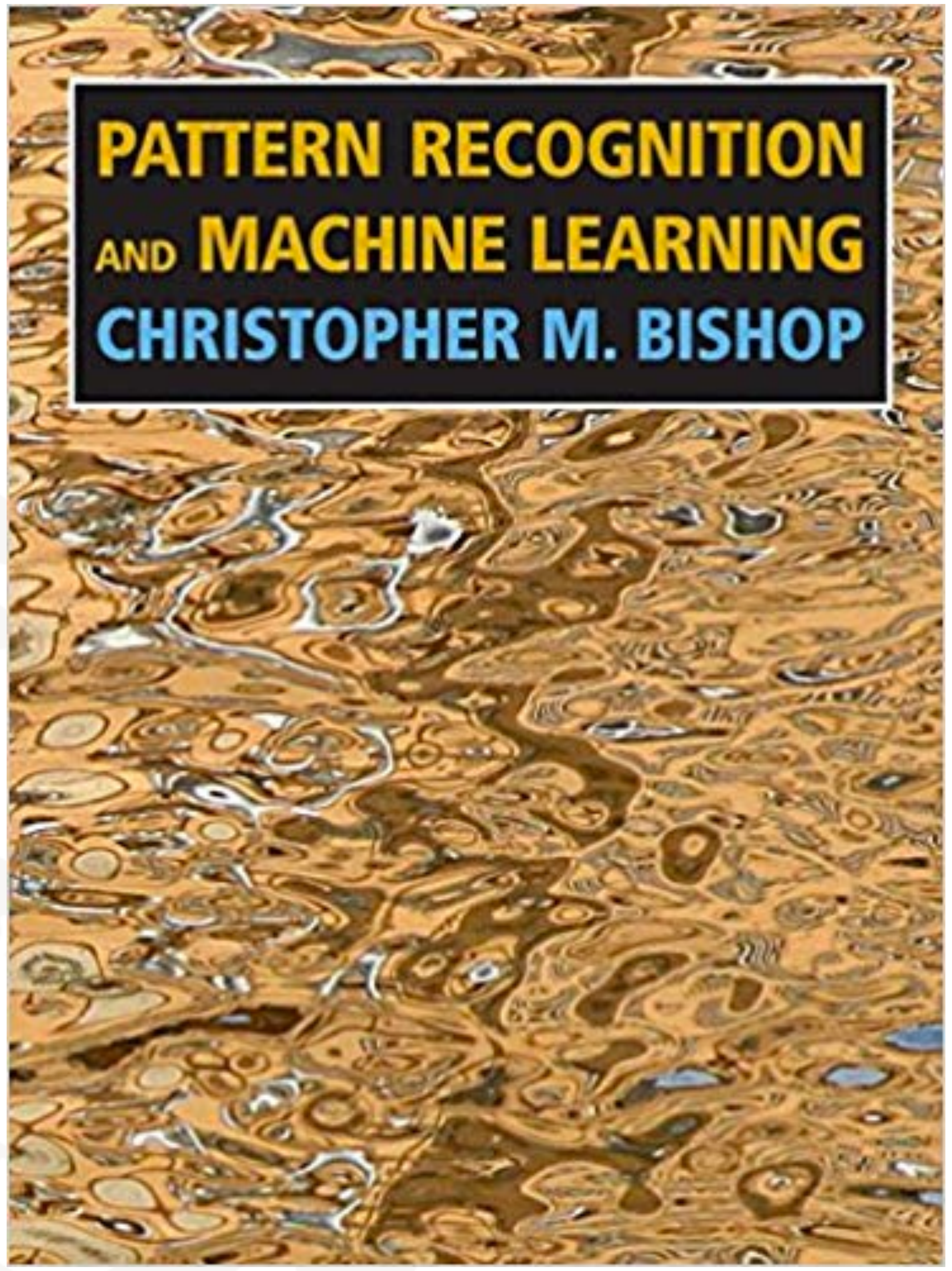
19.06 - week 10:
Unsupervised neural networks: representation learning through variational auto-encoding (VAE). And Generative adversarial networks (GAN).
Practical assignment: Revisiting fashion and Kuzushiji for unsupervised pre-training and image generation + Face generation using GANs

26.06 - week 11:
Self Supervised Learning: Contrastive Learning and similar methods.
Practical assignment: SimCLR and BYOL for image classification.

03.07 - week 12:
Lecture :Classic tabular q-learning. + Deep reinforcement learning, QNN
Practical assignment: (Optional) Cart pole balancing. + Taxi driver with DeepRL

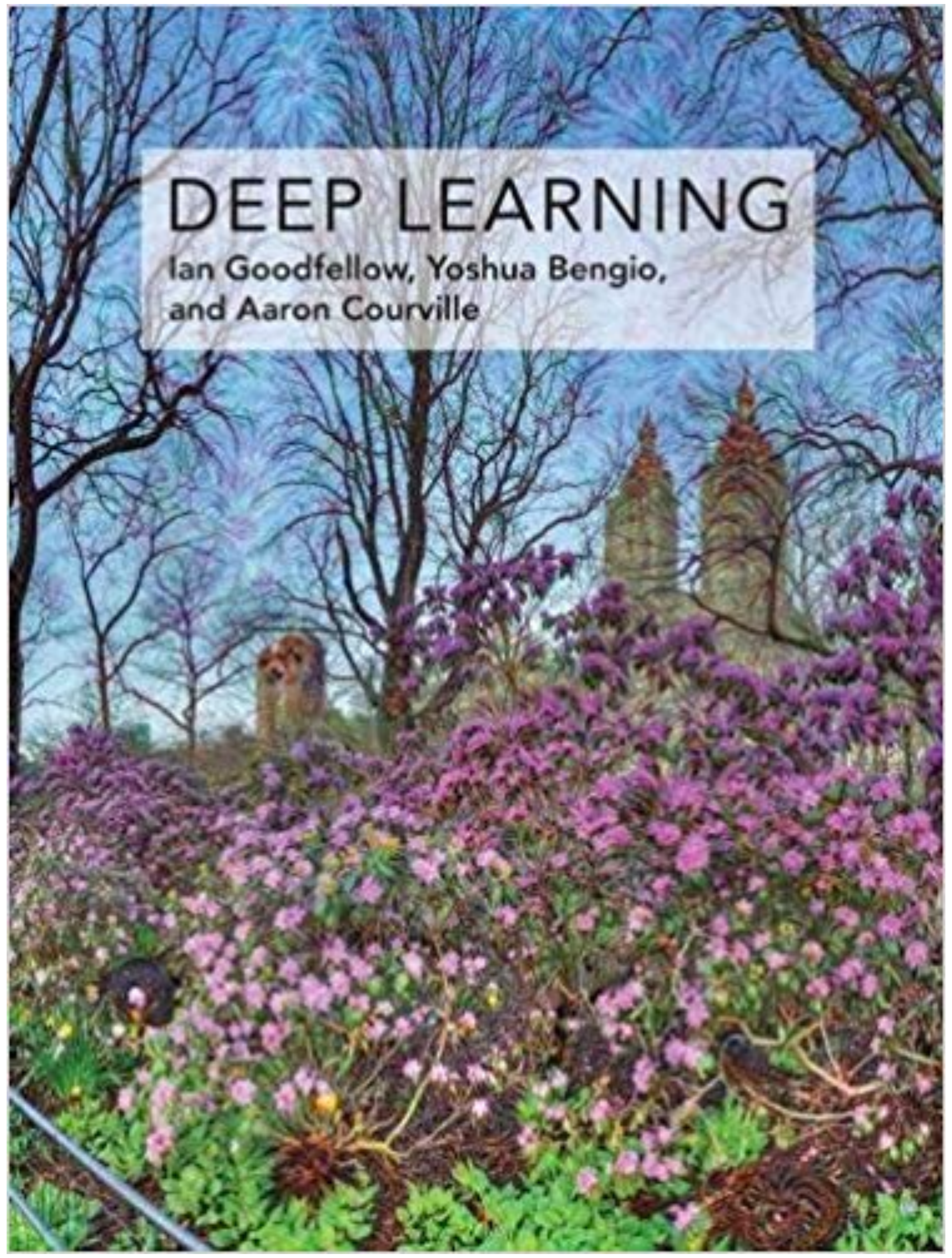
10.07 - week 13:
Meta-learning.
Practical assignment: neural architecture search using the reinforce algorithm.

Literature



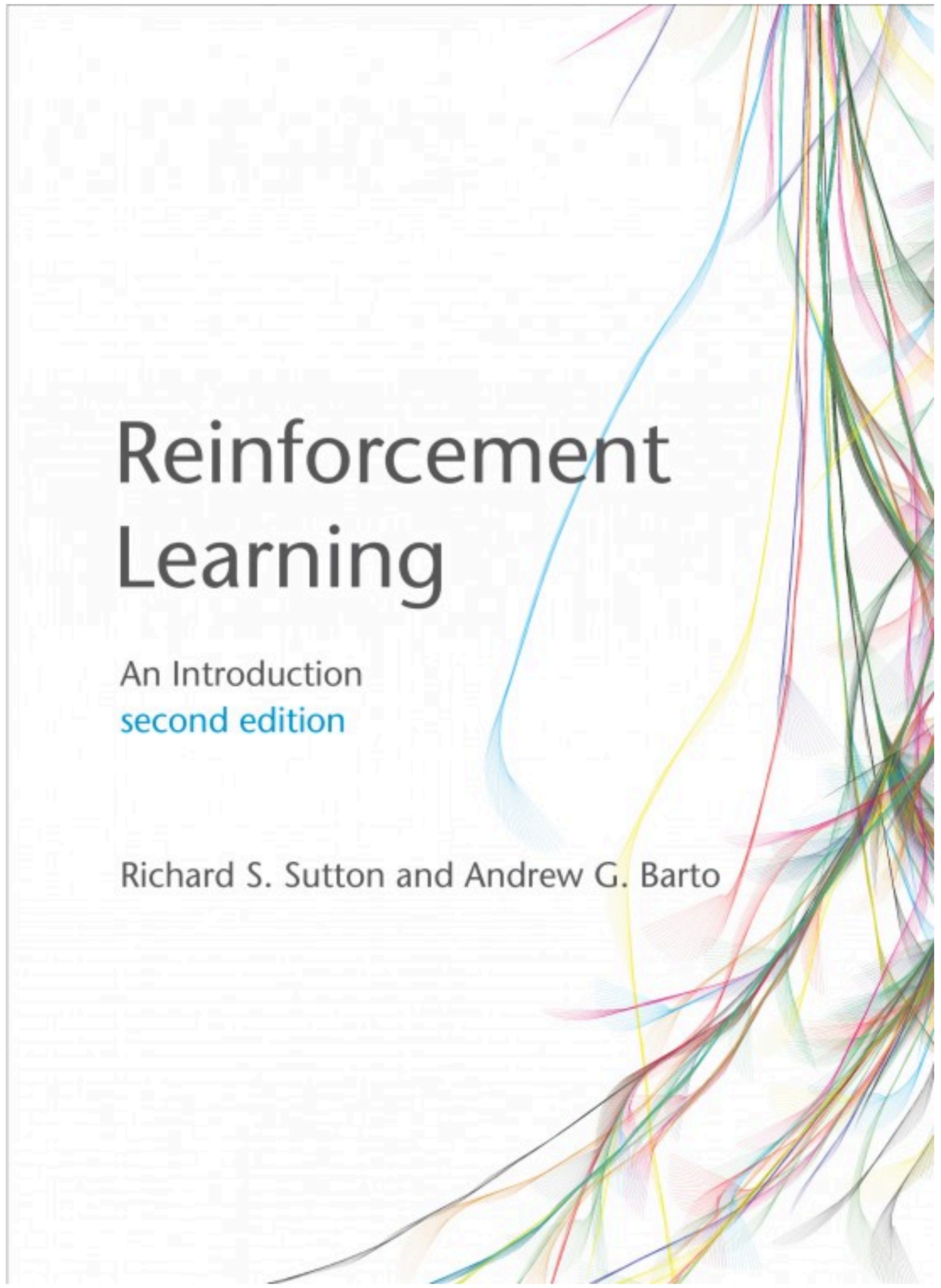
PATTERN RECOGNITION AND MACHINE LEARNING

CHRISTOPHER M. BISHOP



DEEP LEARNING

Ian Goodfellow, Yoshua Bengio, and Aaron Courville



Reinforcement Learning

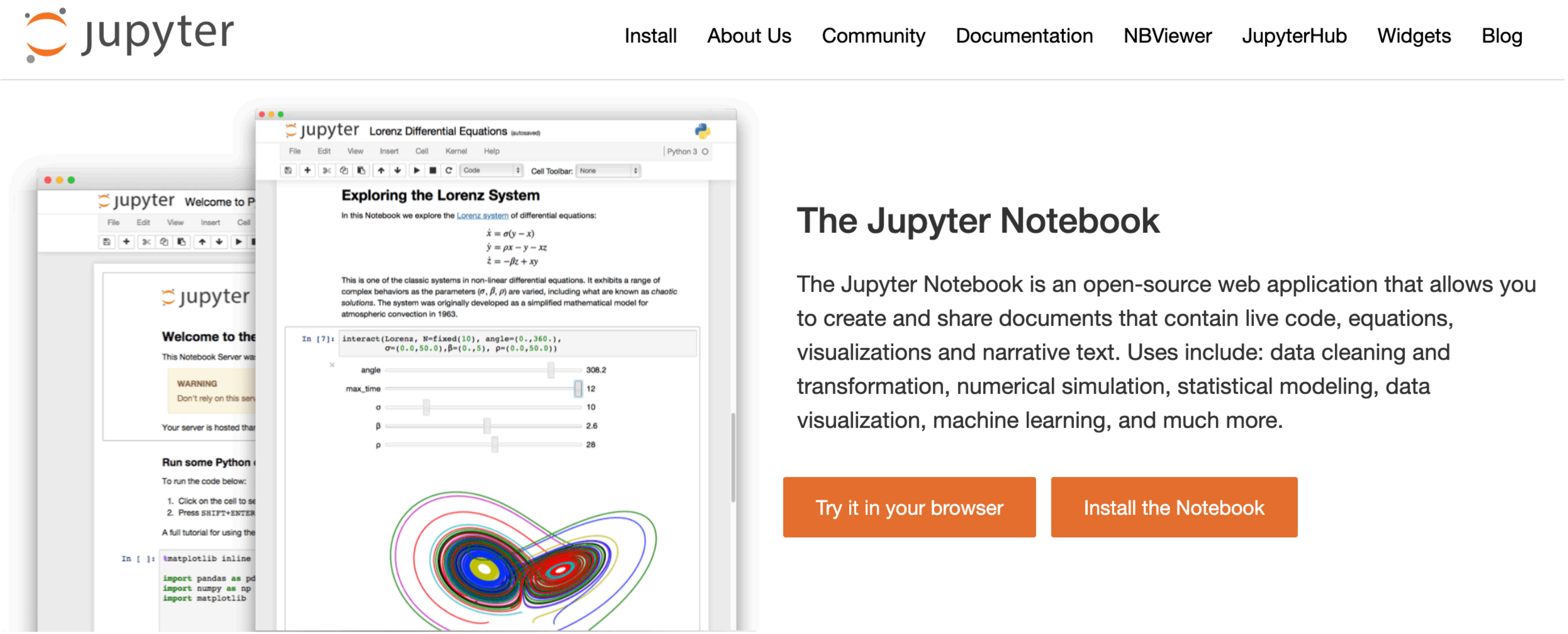
An Introduction

second edition

Richard S. Sutton and Andrew G. Barto

Jupyter notebooks: https://jupyter.org/

- Execution of python code in cells
- We will provide step-by-step notebooks where sections are missing and need to be filled in



```
%matplotlib inline

import pandas as pd
import numpy as np
import matplotlib
```

Exploring the Lorenz System

In this Notebook we explore the Lorenz system of differential equations:

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters (σ, β, ρ) are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

```
In [7]: interact(Lorenz, N=fixed(10), angle=(0.,360.),
                 σ=(0.0,50.0),β=(0.,5), ρ=(0.0,50.0))
```

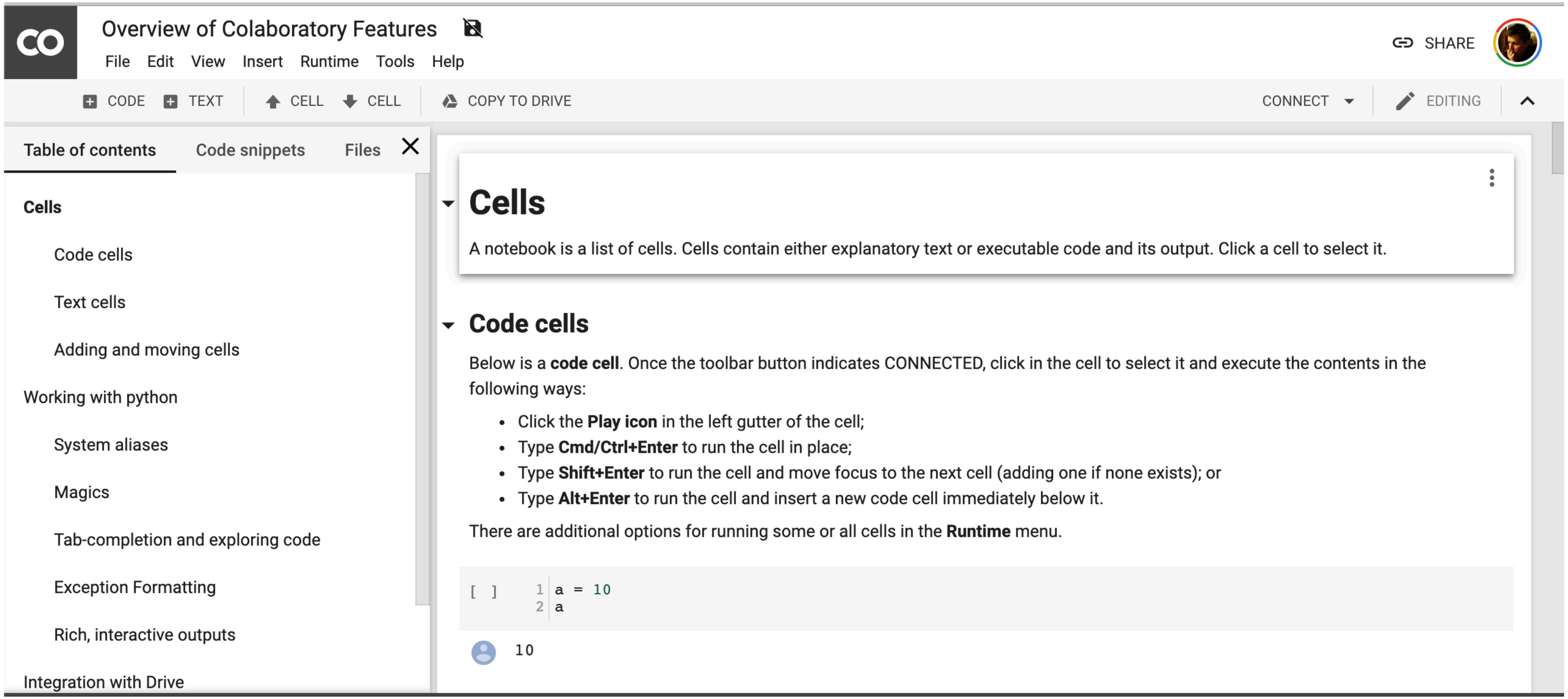
angle 308.2
max_time 12
σ 10
β 2.6
ρ 28

The Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

Google Colab: https://colab.research.google.com/

- Google Colab allows execution of notebooks in the cloud from a browser
- GPU/TPU instances are free with an execution time-out after 1 hour (enough for our purposes)



Cells

A notebook is a list of cells. Cells contain either explanatory text or executable code and its output. Click a cell to select it.

Code cells

Below is a **code cell**. Once the toolbar button indicates CONNECTED, click in the cell to select it and execute the contents in the following ways:

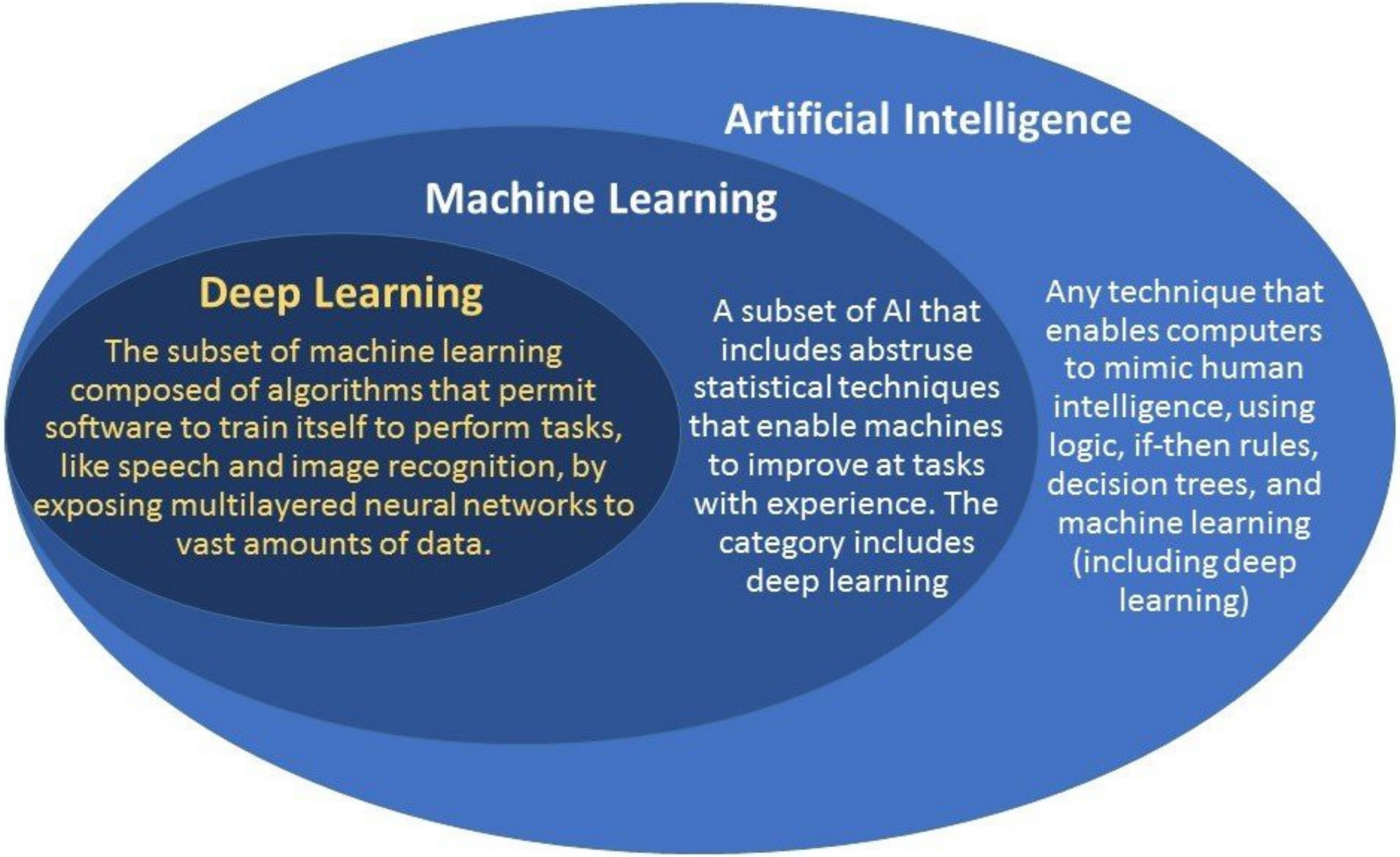
- Click the **Play icon** in the left gutter of the cell;
- Type **Cmd/Ctrl+Enter** to run the cell in place;
- Type **Shift+Enter** to run the cell and move focus to the next cell (adding one if none exists); or
- Type **Alt+Enter** to run the cell and insert a new code cell immediately below it.

There are additional options for running some or all cells in the **Runtime** menu.

```
[ ]   a = 10
      a
```

```
10
```

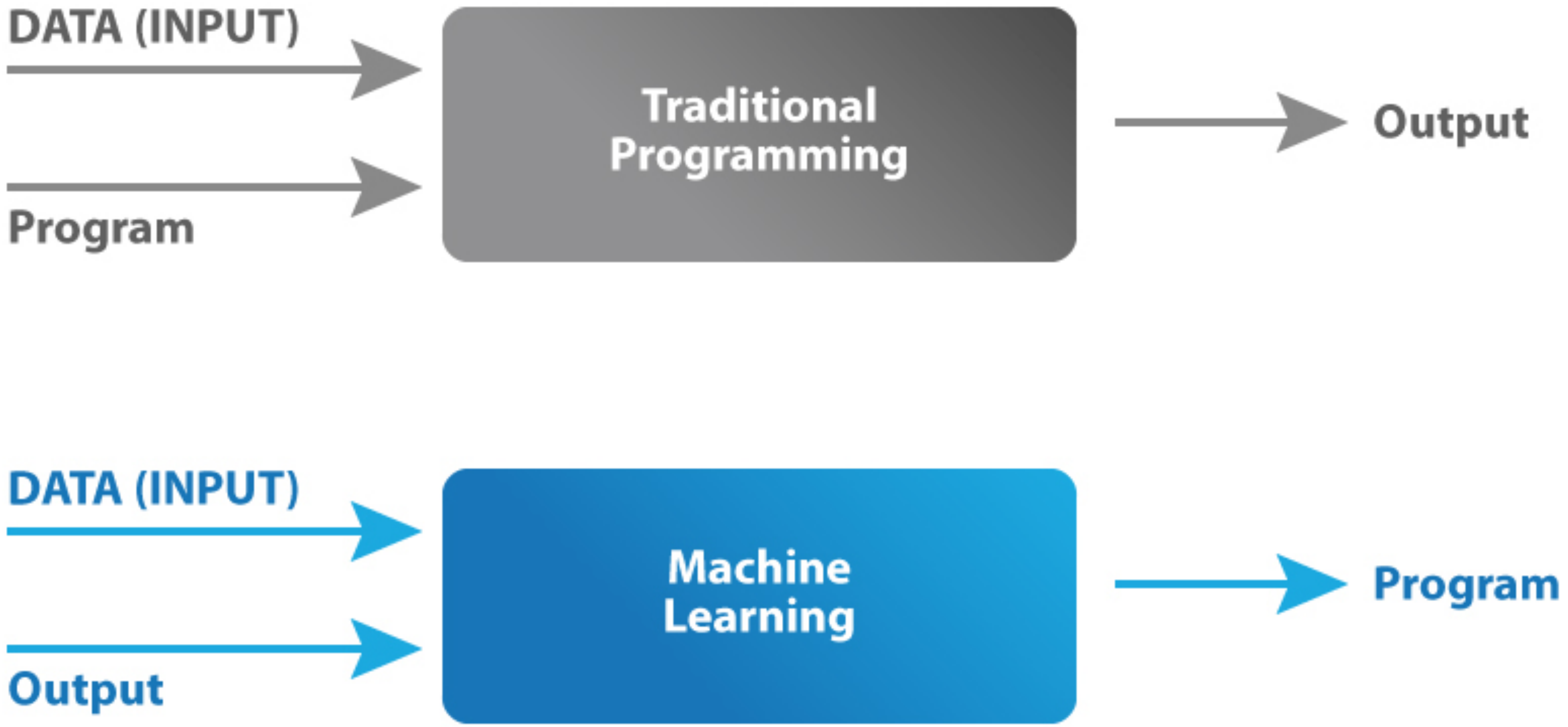
What is Machine Learning?



Artificial Intelligence
Machine Learning
Deep Learning
The subset of machine learning composed of algorithms that permit software to train itself to perform tasks, like speech and image recognition, by exposing multilayered neural networks to vast amounts of data.
A subset of AI that includes abstruse statistical techniques that enable machines to improve at tasks with experience. The category includes deep learning
Any technique that enables computers to mimic human intelligence, using logic, if-then rules, decision trees, and machine learning (including deep learning)

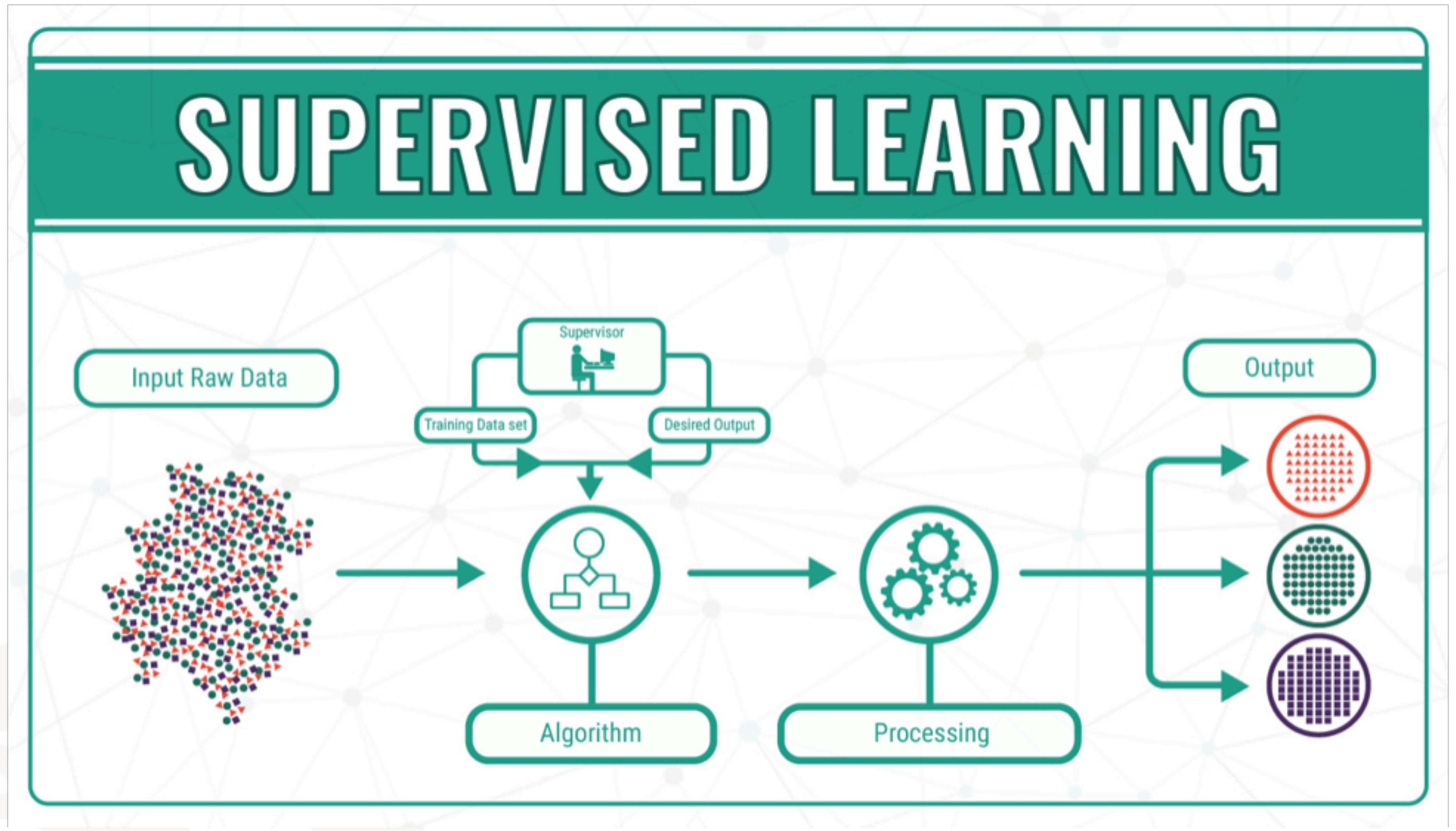
https://medium.com/@harish_6956/what-is-machine-learning-deep-learning-7788604004da

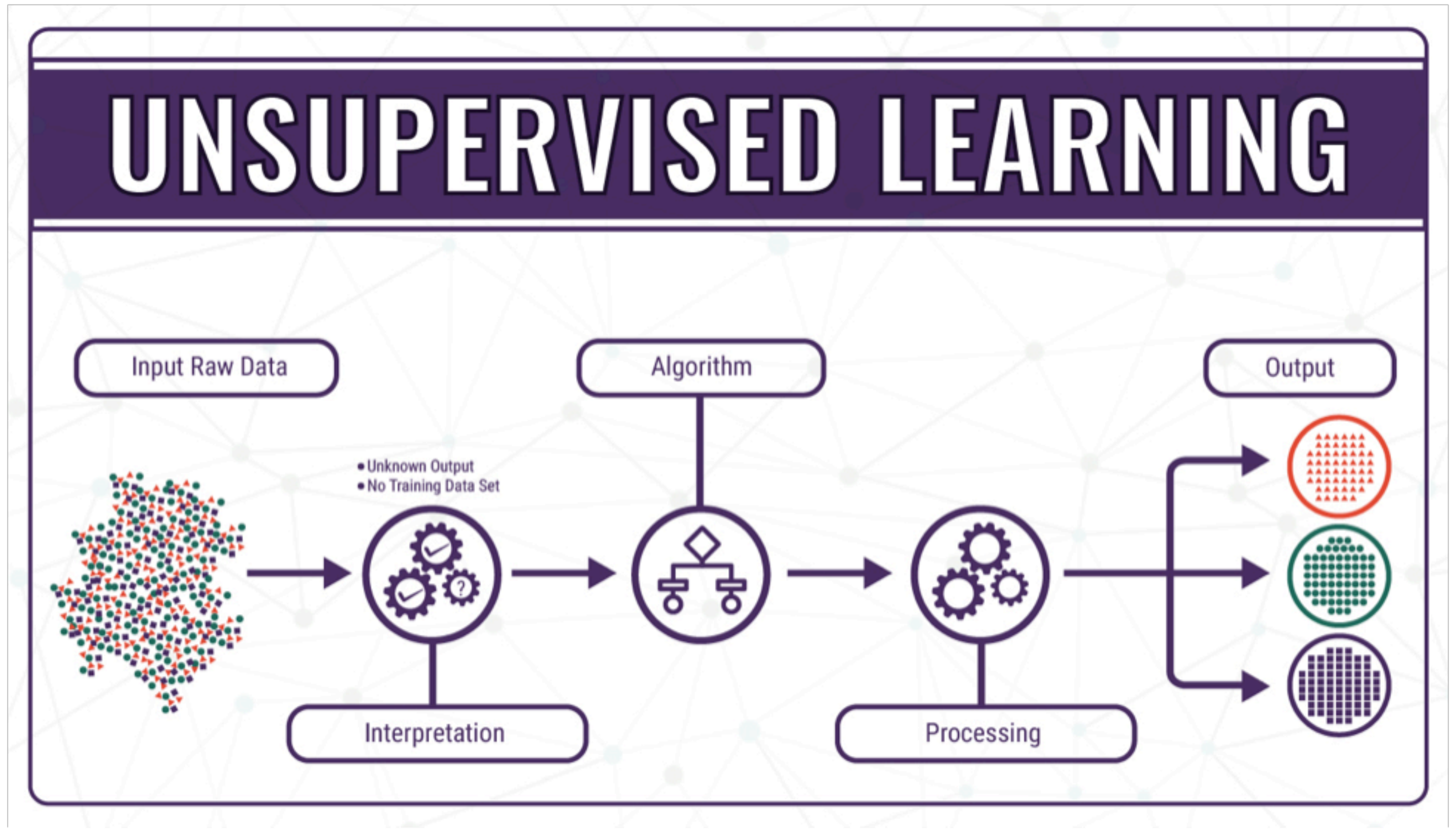
What is Machine Learning?

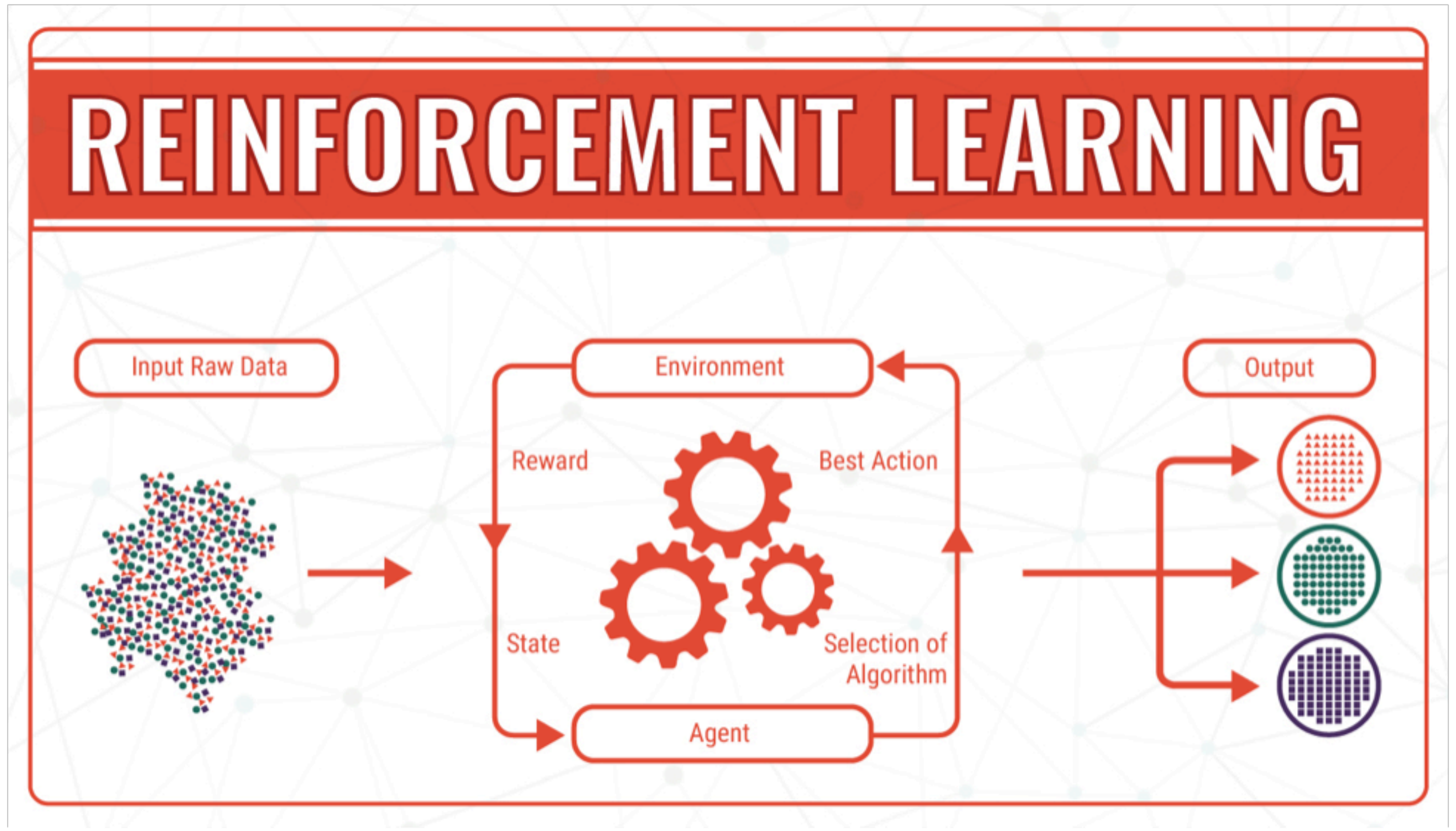


https://www.mygreatlearning.com/blog/what-is-machine-learning/

What is Machine Learning?

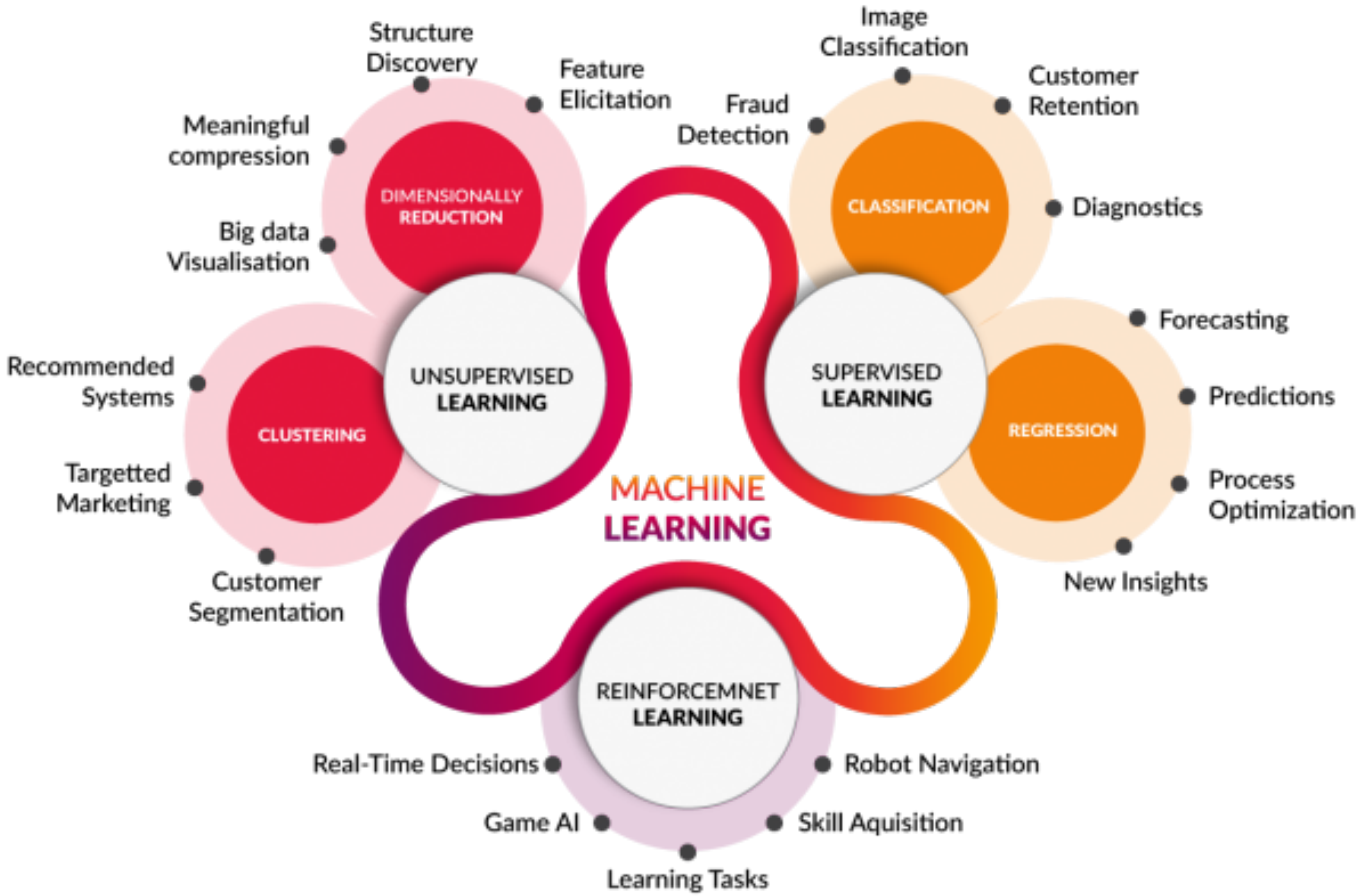






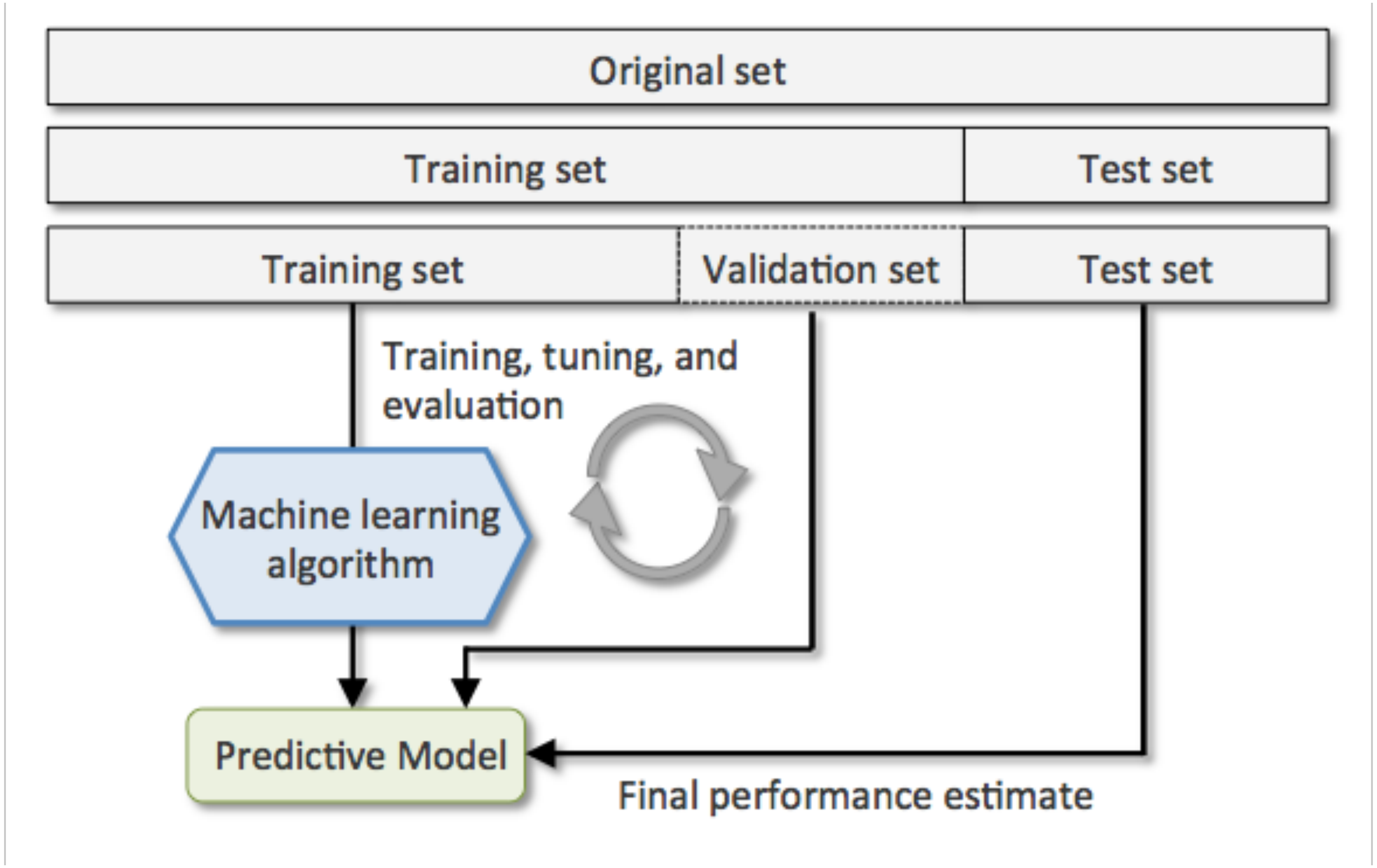
https://bigdata-madesimple.com/machine-learning-explained-understanding-supervised-unsupervised-and-reinforcement-learning/

What is Machine Learning?



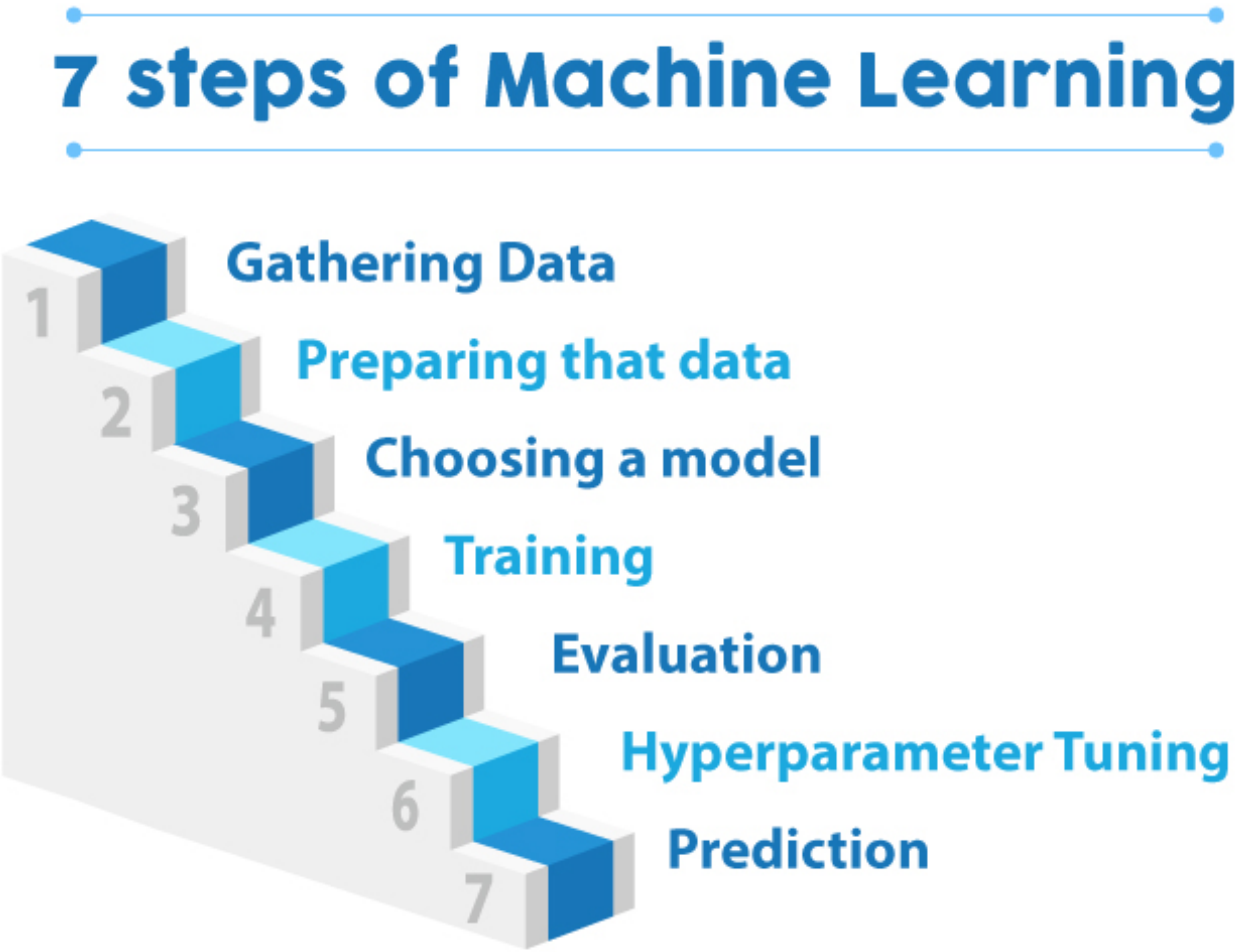
https://www.ironhack.com/en/data-analytics/what-is-machine-learning

How do we train and evaluate?


Original set
Training set
Test set
Training set
Validation set
Test set
Training, tuning, and evaluation
Machine learning algorithm
Predictive Model
Final performance estimate

https://stats.stackexchange.com/questions/410118/cross-validation-vs-train-validation-test

Machine Learning is not just tuning a model



The rigor with which we perform the steps surrounding a machine learning system will majorly impact its usefulness and our gained insights.

https://www.mygreatlearning.com/blog/what-is-machine-learning/