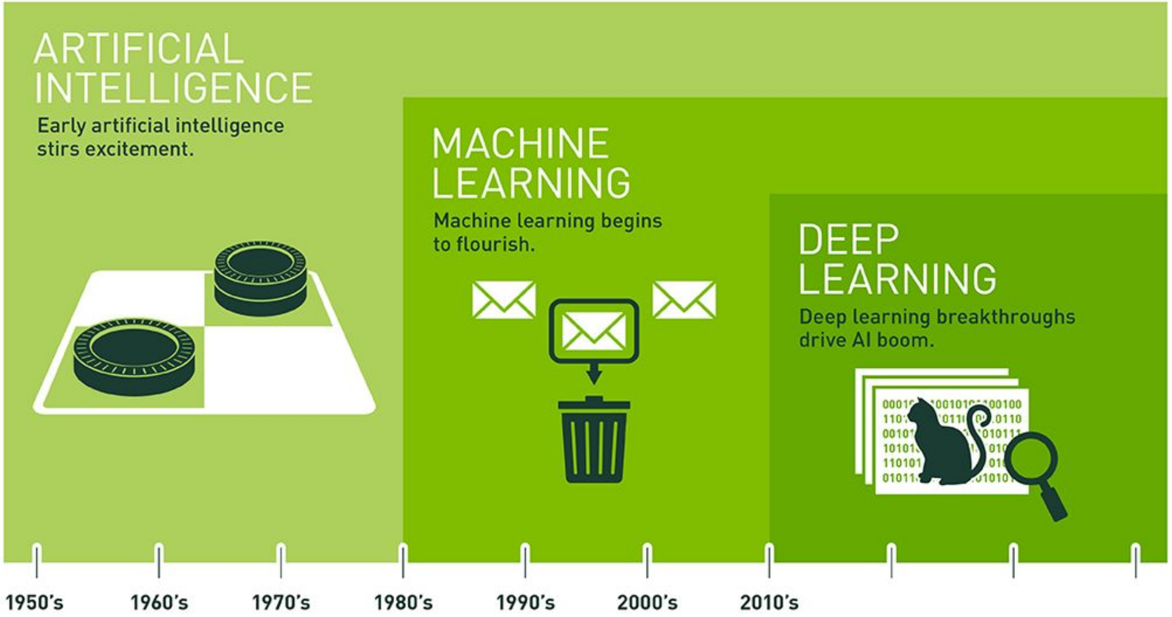
Machine Learning

Tech Club

What is it?

- Machine Learning (ML) algorithms have the ability to learn patterns from input data, and make predictions without explicitly programmed to do so.
- The same ML algorithm can be used for very different use cases.
- For instance, you can use “Neural Networks” to predict the possibility of cancer, or use it to see whether an image is a hot dog or not.
- The algorithms were built using statistics and probability theories.

Buzzwords



ARTIFICIAL INTELLIGENCE

Early artificial intelligence stirs excitement.

MACHINE LEARNING

Machine learning begins to flourish.

DEEP LEARNING

Deep learning breakthroughs drive AI boom.

Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

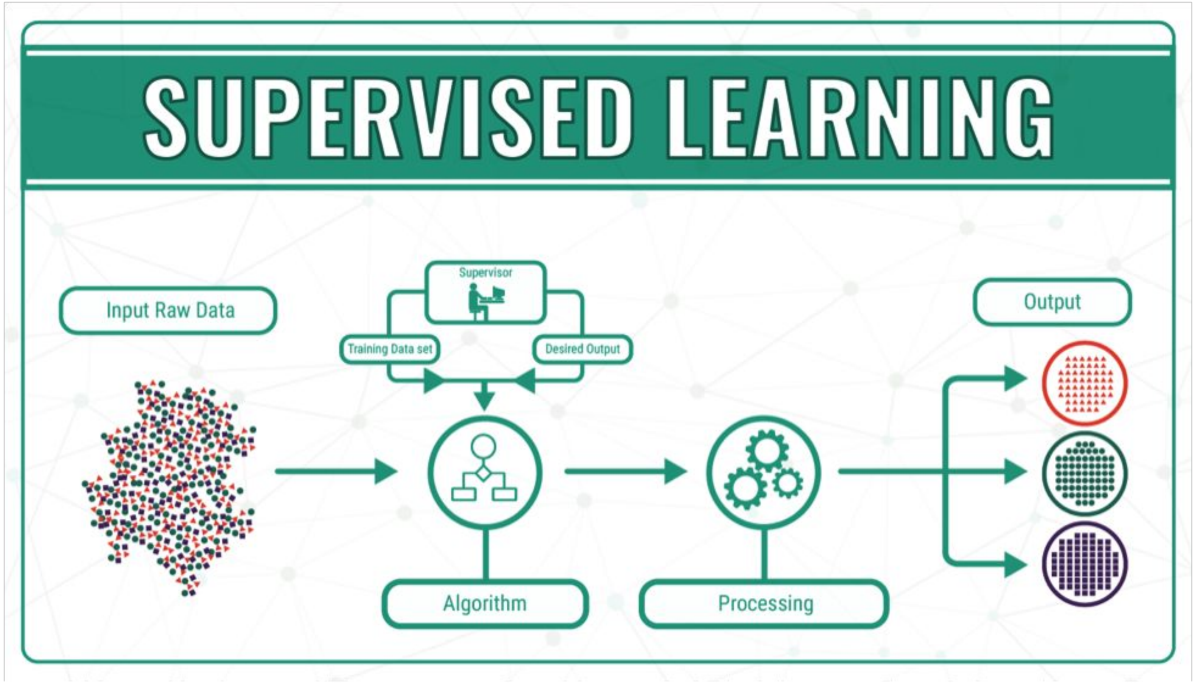
Buzzwords

- **Artificial Intelligence:** Algorithm + Heuristics
- **Machine Learning:** Algorithm + Heuristics + Statistics
- **Deep Learning:** Algorithm + Heuristics + Statistics + Data-Driven
- **Data Analysis:** Plotting Data + Tabulation + Writing Reports
- **Data Science:** Data Analysis + ML
- **Big Data:** Data Analysis at a large scale

ML in the news

- Google Duplex, Lens
- OpenAI's 5v5 Dota Bot
- Portrait mode on mobile phones
- Prisma and FaceApp
- AMD Ryzen 7 (Hardware level AI)
- Netflix Recommender System
- ...and tons more.

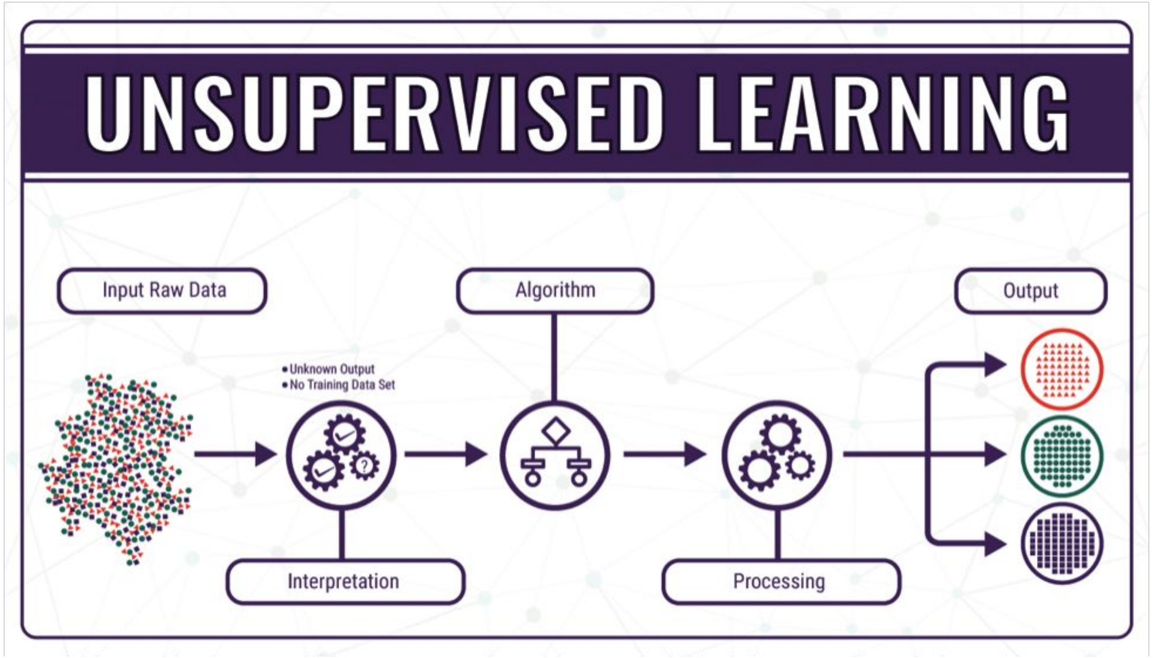
Types - Supervised Learning



Use Cases:

- Classification
- Regression
- Segmentation
- Localisation
- Translation

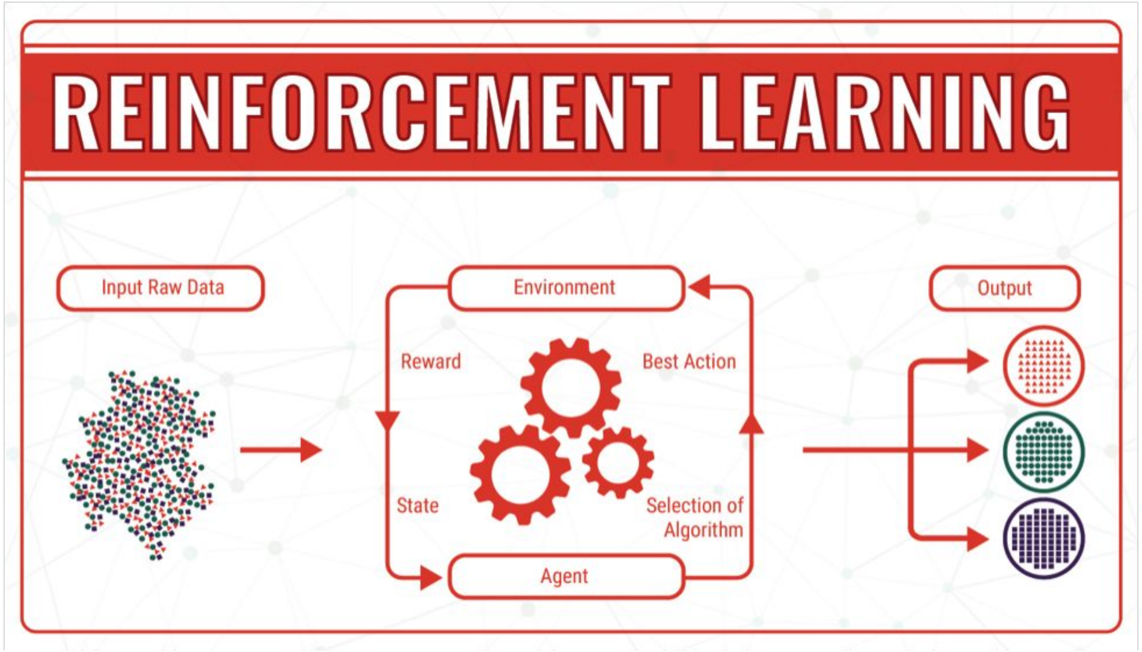
Types - Unsupervised Learning



Use Cases:

- Dimensionality Reduction
- Pattern Finding
- Clustering

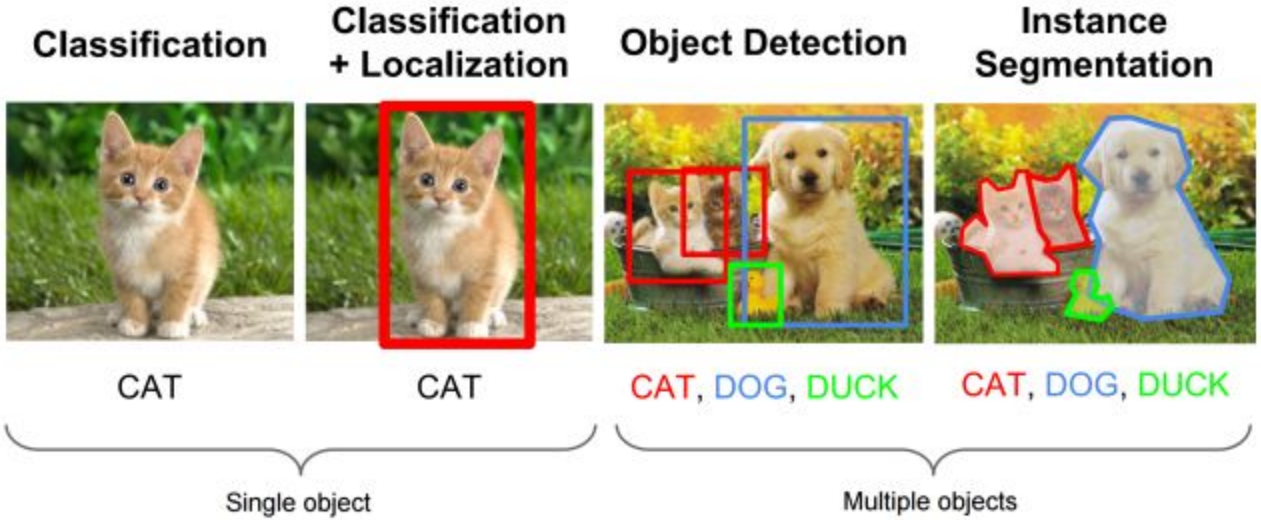
Types - Reinforcement Learning



Use Cases:

- Advanced maze solving
- Robot movements
- Playing games

Examples


Classification
Classification + Localization
Object Detection
Instance Segmentation
CAT
CAT
CAT, DOG, DUCK
CAT, DOG, DUCK
Single object
Multiple objects

How2ML

- A lot of these algorithms are available in libraries, especially in Python. Python is the most widely supported language (and preferred) for Machine Learning.
- As developers, we are interested in using ML for some application. So as much as possible, do not reinvent the wheel. Use the algorithms available in the libraries.
- With that being said, it is also important to know the theory and working of the algorithms. Blindly using the algorithms may not give you the best results.

How2ML

- The best way to get started is to checkout any popular ML course. You can also gauge your interest level based on the content. The next few slides will list out some popular courses.
- Once you are done with the course, implement your algorithm on a toy dataset (Like MNIST or Titanic). If that worked out well, try using it for some unique application.
- Courses may not teach you all the useful algorithms. With that in mind, the next few slides will list out some popular algorithms and libraries as well so that you don't miss out.

Popular ML algorithms

- Linear Regression/Classification
- k-Means Clustering
- Random Forest Regression/Classification
- Support Vector Machines
- XGBoost and LightGBM (Advanced algorithms)
- Neural Networks or Multi Layered Perceptrons (MLP)
- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN/LSTM/GRU)
- Object Detection and Localisation (Faster RCNN / SSD / YOLO)

Popular Python Frameworks (Libraries)

- Data Manipulation
  - Numpy (For matrices)
  - Pandas (For tables)
- Image Manipulation
  - OpenCV
  - Scikit-Image
- Plotting data
  - Matplotlib
  - Seaborn

Popular Python Frameworks (Libraries)

- Machine Learning
  - Scikit-Learn
- Deep Learning
  - TensorFlow
  - PyTorch
  - Keras
  - Caffe, Theano, Mxnet
- Misc
  - Nltk (For NLP)
  - os, glob, collections, time, re, argparse, json etc. (Inbuilt libraries)

Popular Courses

1. Machine Learning - Coursera
2. https://www.deeplearning.ai/ - Coursera
3. https://www.fast.ai
4. CS 231N - Stanford
5. Udemy Courses - (A-Z, Lazy Programmer etc.)
6. Read tech blogs on Medium/OReilly or other sources

- **Note**: It's okay to do free courses without a certificate. The projects you make with the skills you've learnt will speak louder than a piece of paper.

Keras or TensorFlow?

- Keras is a higher level API that can run on top of other Deep Learning frameworks, including TensorFlow.
- Coding in Keras is much **much** simpler. Popular deep learning algorithms can be implemented with nearly no effort at all.
- It is excellent for beginners, and to try out prototypes.
- The simplicity of Keras comes at the cost of flexibility. It is hard to experiment with custom algorithms. Hence, it is not recommended for hardcore research or deployment.

Keras or TensorFlow?

- Coding in TensorFlow is moderately difficult to start with.
- It’s difficult because it uses “symbolic programming” and stuff like sessions, scopes and graphs.
- A 100 line code using TensorFlow may only be 20 lines using Keras.
- But TensorFlow is widely used for research owing to its great flexibility.
- It also has support over different platforms and edge devices (Phones, Raspberry Pi’s etc.)

What to do after the courses?

- Try playing with “toy datasets” such as MNIST, Titanic, Iris etc.
- Once you’re confident with implementing basic algorithms, try doing a hobby project. It can be as simple as classifying oranges from apples.
- Read research papers in ML if it interests you. You can work on more complex problems with professors. If you are confident enough, you can try tackling them yourself too.
- Curate a github profile with all your projects.
- Attend hackathons and online contests.

Work environment

- Usually, software updates are rolled out on Linux systems first, as they are very flexible.
- Hence, it is recommended but not mandatory to use Ubuntu for ML purposes.
- Deep Learning models are pretty compute intensive, so we often run them on GPUs. If your system does not have a GPU, you can try Google Colaboratory or Kaggle’s kernels. They allow you to use their GPU for 12hrs, free of cost.