# What are Artificial Intelligence, Machine Learning, Deep Learning and Generative AI?

Most people often get confused over these three terms. To begin with, Artificial intelligence refers to techniques, methodologies, and algorithms that includes those of Machine Learning, Deep Learning, and generative AI. In a general overview, AI equip computers to simulate human behaviour, enabling them to learn, make decisions, recognize patterns, and solve complex problems in a manner like human intelligence. Therefore, it's objective is to create a intelligent agents, which get the percpetion of the real world through the using of sensors, learn using some of the advanced ML tecniques, make decisions and finally implement through acutators.
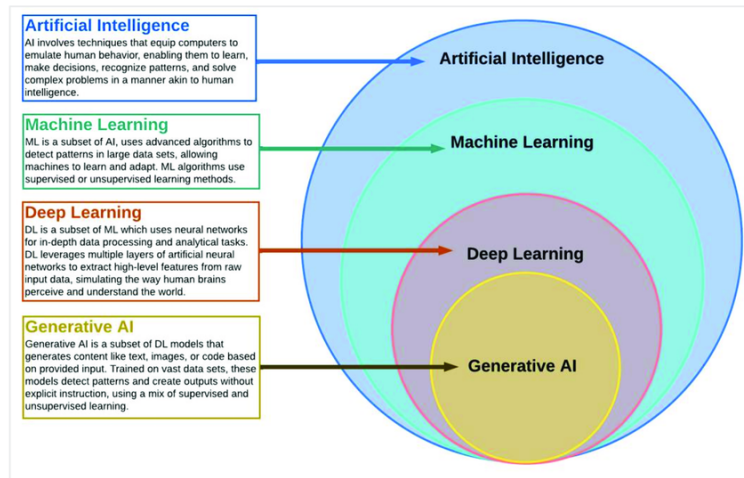
Hierrachical Relationship

Machine Learning: It is a subset of the AI which is the science of programming a machine to make it learn to perform a task from the data, rather than through programming.

Deep Learning: It is again the subset of the AI, but also further subset of the Machine Learning field which is an ensemble of tecniques and alogorithm that exploit neural networks for in depth data processing and analytical tasks. And it involves multiple layers of artificial neural netowkrks to extract high-level of features from raw input data, simulating the way human brains, percieve and understand the world.

Generative AI: This is again the subset of deep learning models which is more advanced form of AI, that has the capacity to generate content like text, images, or code based on provided input.

Short and Suit example:

An Autonomous car:

Artificial in this system, collects data from real world using sensors(road, cars, road signs, speed, objects), possesses a set of rules of behaviour that has its intelligence embeded. Some of the intelligence are: If a car is about to collide with an object, brake preventively, with the rules: a set of actuators implementing relative decisions(brakes, steering, ABS).

In this AI system of the car has a sub module of Machine Learning that allows it to improve over time, as the car "sees" different roads and situtations and increase its performance compared to some metric(safety, reaction time to braking, consumption).

Addditionally, sub-modules can use Deep Learning techniques such as neural networkds, trained to recognize the route of the road, the shapers of cars, distinguish them from those of trees and pedestrians, and so on.

Traditional vs Modern Programming: Machine Learning

Traditional Programming
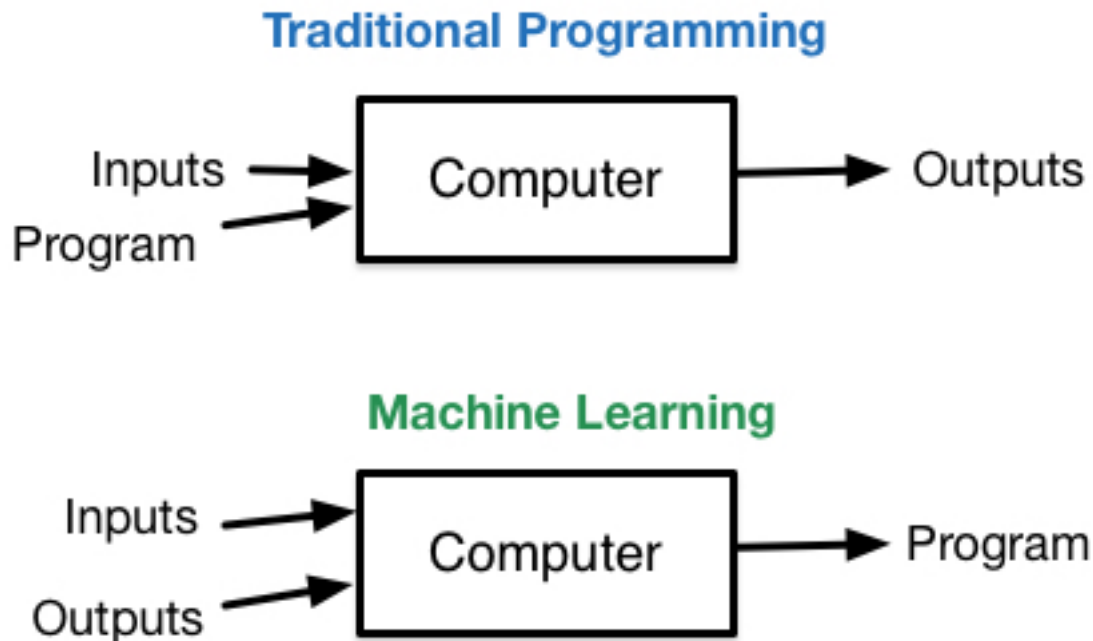
Machine Learning

Traditionally, the classic approach to the problem consists in designing a process(algorithm) that is then realized in a programming language implmentation), thus solving the problem iin a detterministic fashion.

The approach of Machine Learning consists instead in seeing a lot of input examples with the output considered correct, and programming the machine to detect the recurring patterns that occur in the data. So, what happens is, in this ML method, there involves corrected input and output together, and programmed the machine to find the recurring patterns that occur in a dataset/business model. Then pattern is used by machine as "knowledge", and according to that knowledge, machine tries to return the correct output provided new input(seen or unseen).

And here comes the important terms, **training**, which is the process of feeding with data and its result is a model: a set of parameters properly 'adjusted' so they could recongnize the patterns that interest us for the given problem. Once a model is obtained, it can be used to predict the correct output to inputs that it has never "seen".

Some of the great examples of using Machine Learning is applied in the real world:

1. Spam email filter.
2. Hand drawn figures or symbols, such as numbers, letters, or figures recognition
3. Recommeded system: A system of customization of content suggested based on what you've already seen such as Netflix recommends the next movie to watch, Facebook suggesting groups releated to our interest, Amazon shows prdoucts often purchased togther, youtube suggesting the related videos often watched.
4. Sentiment analsis: This is analysis of extracting important information from tweets or other data generated by humans about a particular event or character. For example, the post that I tweet in the twitter can be analysed whether the people like or hate the tweets(negative, positive, neutral)
5. Prediction of next words: A system of writing assistance that, as the text in composed, it foresees the next words and suggests them. Such as automatic completion of smartphone or the facebook search bar. This has already been implemented in copilot and notebook python.
6. A computer vision system that recognize the model and brand of a car by seeing it with a sensor.

Below are not a Machine learning examples:

1. Search engines: this system in not a machine learning application as it does not improve with experience, and has not got "intelligent", instead have to program explicitly and is totally based on keywords.

2. A sensor programmed detecting car number plates. As it may seem like that it is related to the concept of computer vision but there is not involvement of improving with experience, and its performance which are totally based on programming only at the design stage, where it is explicitly calibrated to detect a standard object(the license plate).

In summary, Machine learning are appropriate for:

1. Fluid environments, where the problem constantly changes and in an unprdictable way.
2. Complex problems where the traditional approach would be impracticable or would lead to "frankenstein" of code with high maintenance cost. The ML approach can significantly simplify the code and achieve better performance.
3. Get information about complex problems and discover patterns through large amounts of data.

What do I need to do Machine Learning?

Today, every area of science and economy has access to a quatity of data unimaginable even a decade ago. Additionally, there is abundant and inexpensive way of computing capacity which is possible and available through the powerful concept cloud provided by cloud service providers. Uncounted research on machine learning has been taken place and been going on expeditiously in the development of increasignly effective techniques. Similarly, the engineering world creates the ML toolbox by ajusting itself accordingly, and has already developend wonderful frameworks and high-level interfaces to apply ML techniques quickly, which is going towards more user-friednly 'enablers'.

Keras, which is a high-level framework of Pyhon allows to train a deep neural netowork in just 15 lines of code. Cloud providers offer the possibility to host on their machines our ML models, with powerful logic that allows us to buiild a complete AI/ML startup from our laptop.This is a perfect example of the above phonemenon.

Now let's jump in to the elements that we need to get started.

The three elements

Data

Data comes from the latin 'Datus', and it's meaning is fact, happened. Building ML programs means exploiting something that has already happened in the past(the data) to discover recurring patterns that suggest how to act in the future.

ML program is something that we don't write ourselves, but rather something that we understand by observing and processing data. The programmer tries to make the the data do the bulk of the work and engineers combine data with algorthims to grow prorgams. And it is often required to understand that if the data is to do bulk of the work, the quality (and quantity) of the data is strictly necessary for a model to make accurate predictions, or to extract knowledge effectively from them.

Today there is a large amout of data available to start practicing machine learning and discover its potential.

Free avaialble data

1. Kaggle
2. UCI datasets
3. US Government open data
4. EU open data
5. [A very long list from Forbes] (https://www.forbes.com/sites/bernardmarr/2016/02/12/big-data-35-brilliant-and-free-data-sources-for-2016/)
6. UK Government open data

Collect and label your own data

Data collection is a huge and critical topic, which technical details go outside the scope. So, to collect huge quantities of data we can do through scrapping and other sources. Other resources: Digital Methods.

Nowadays, many and more organizations feeling like they are on top of goldmine. As they have years and years of historical data, from machines, conversation logs, transactions, or sensor measurements. There might be temendous value to be unlocked.

There is a technique of "five-whys", which is a critical thinking process. So, to deal with or get the data part right, let us know our data before performing any actions or tasks or transformations. For detail, go to github

know-how

Doing data science is not simple and will never be if you want to fully understand the mathematical processes behind the elegant models offered by high-level interfaces such as keras.

The process of developing a DS project consists of:

1. Business Understanding or domain knowledge
2. Data collection
3. Data preparation
4. Exploratory Data Analysis
5. Modelling
6. Model Evaluation
7. Deploy model

1. Business understanding: This is the first step before going through the analytical process which is basically mean to understand the domain knowledge of the particular company what they are desire to get from the data. Proposing of relevant questions and define objectives for the problem involves in this first step. Data science cycle usually consists of experimenting with solutions in an iterative way by collecting data, shaping the problem and developing ML models.

2. Data Collection: This is the process of gathering and scraping the data necessary for the project and consists of putting together data sources, building pipelines to pre-process them, and having a place to put them on hold for analysis.

3. Data Preparation: This process has got two more steps: data cleaning and data pre-processing/data wrangling. This consists of feature engineering(combining existing data in a meaningful way), training models, and evaluating their performance.

4. Exploratory Data Analysis: EDA is the phase of exploring and visualizing the data to get to know the relationships and features and helps in selection of the features for model building. Formation of hypotheses about the problem that has been defined in the first step and analyze the data by visualizing.

_ to confirm later:_ Inside EDA: **data exploration**: forming hypothesis about problem by visually analysing the data and **feature engineering**: selecting the important features and construct more meaningful ones using the raw data that you have.

5. Modelling: This is the phase where the data is used to train a machine learning (or deep learning) model that will be able to predict on never seen data.

6. Model Evaluation: It consists of evaluating the performance of different models and selecting the best one in the real-life scenario of the problem and use them to make predictions.

7. Model deployment: This is the last step where trained model make available to consumers.

8. Data visualization: Communicate the findings with key stakeholders using plots and interactive visualizations.

After all the steps above, regularly monitoring of the performances of the model is required to do and to change as per requirement.

Note: It's better to know the agile methodologies in data science

Computational Power:

The third element that is computational power. The most of the time, you will be doing the calculations between matrices while doing experimenting with Machine learning. Similarly, Deep Learning and neural netowrks are by no means of exception for calculations. So, for these calculations, we need high computational power capability computers. In the past, such calculations were made directly using the CPU of computers, which is ofcourse not sufficient to train the most complex networks.

Fortunately, to overcome the above case, there is GPUs developed which is GRAPHICAL PROCESS UNIT, and its task is exactly to do matrix calculation. Moreover, due to the internet connection and cloud providers, it is becoming much more convenient to use machines virtually like Amazon, Google and Mircorsoft.

With a few clicks, files and files of high performance machines can be deployed, and managed in a transparent way especially with on-demand capabilities. This is done all from a dedicated web interface or client, which drastically reduces the cost and complexity of developing complex softwares and solutions. It also allows us to serve solutions quickly, flexibly and without the classic problems of large-scale computing that today's distributed systems require.

Recently, all large tech companies have started to provide services related to AI and ML. Therefore, data engineer and data scientist are able to achive incredible levels of proudctivity and business through end-to-end platforms that cover the entire development cycle of AI and ML solutions. Some are AWS Sagemaker and Azure Machine Learning Studio.

There are other simple way to experiment or to do data analysis task such google colab or jupyter notebooks, which are free.

Do you really need machine learning?

Although AI and ML offers immense business and product opportunities, they are not always the recipe for all evils. You can't solve all kind of problem with this. Deep learning is a type of artificial intelligence where a computer learns to recognize patterns and make decisions by mimicking how the human brain works. Imagine it as teaching a computer to learn and understand things by itself, just like how we learn from our experiences.

Simple Explanation with Examples:

1. **Learning Simple to Complex Concepts**:
   - Deep learning models are like a series of building blocks stacked on top of each other, where each block represents a different layer of learning.
   - The first layers learn basic, simple features, like edges in a picture.
   - As you go higher up the layers, the model combines these simple features to recognize more complex patterns, like shapes.
   - Finally, the top layers understand very complex concepts, like recognizing a face in a photograph.

Example 1: Recognizing a Cat in a Photo

- **Initial Layers**:
  - The first layers might learn to detect edges and lines (simple features).
  - These layers look for things like horizontal, vertical, and diagonal lines.

- **Middle Layers**:
  - These layers combine edges to detect shapes and textures (more complex features).
  - They might recognize patterns like fur, eyes, and whiskers.

- **Final Layers**:
  - The highest layers put all these shapes and textures together to recognize the entire object.
  - At this point, the model can identify that the combination of fur, whiskers, eyes, and specific shapes means it's looking at a cat.

Example 2: Understanding Spoken Language

- **Initial Layers**:
  - The first layers might learn to detect sounds and phonemes (basic units of sound).
  - They recognize individual sounds like "b," "a," and "t."

- **Middle Layers**:
  - These layers combine sounds to form syllables and words.
  - They might recognize the word "bat."

- **Final Layers**:
  - The highest layers understand the context and meaning of sentences.
  - They can take the word "bat" and understand if it refers to a flying mammal or a piece of sports equipment, based on the surrounding words.

Summary

Deep learning models start by learning simple elements and gradually build up to understand more complex structures. They are designed in layers, where each layer processes and learns from the output of the previous one, just like how we understand the world by first recognizing small details and then combining them to grasp the bigger picture.

Deep learning requires three elements to function, as we have discuss previously. After taking consideration above all, the next thing is to evaluate opportunites.

Evaluate Opportunites

Having above all the three elements, it is likely that ML and DL can offer incredible possibilites.

How can we be sure whether if the data have hidden pearls? in general, we can use ML methodds to predict the value(Continuous) or the class(discrete) associacted with a row from having the right amount of data(for example, rows of a database). Or in other words, we can predict the value of a database column.

Example

Let's suppost we have an e-commerce database and contains for each line the record of the purchase of a product. Some of the columns are: "name", "color", "price", "price range". The column "price range" has got a label that identifies if an item has a low price or high, and this range is provided by the company that produces the product. Sometimes, however, the company that supplies the product is careless and forgets to assign a price range to the object.

Solving above problem

The problem is associated with binary classification task which takes as input the price of the object(some features, such as color, do not affect the price in this case) and return output the label of the price range("high" or "low").

In this situation, we can use a deep learning model to predict the "price range" label for products that are missing this information. Similarly, can try more general ML technique and also a deterministic program(traditional programming approach).

With neural network, in which it takes input all the features and returns the value of the price range as output. **But it would probably be an overkill**. We know that data must be cleaned, prepared to be inserted into the network, the network must be trained and adjusted until it reaches satisfactory evaluation metrics. In addition, if tomorrow the requirements of the forecast change(for example if new lable added), all the process need to be done again, and a new network should be trained and evaluted.

This problem can be solved using just simply Machine Learning generic Linear model that maps the input to the ouptut without the need to introduce "non-linearity". **Linear classifier**(classical ML) can easily solve the problem, and probably get better performance.

This problem is so simple that the use of heurisic rules such as "if.....then....." could solve it. In fact, we could notice that simply all the objects are a certain N values are classified as "cheap" and those above that N as "very expensive".

___Deep learning is an extremly powerful tool, and for this reason, we should avoid using it for tasks that are too simple because overengineering is never good. Which means that they are not always the simplest solution, and sometimes the effort that we apply could be useless.___For better understanding the concept click When not to use Neural Networks

Hence, it is higly advisable to ask question yourself first: is my problem quite complex and do I have enough data to make useful machine learning techniques or even deep learning?

The techniques and tools of ML and DL are primarily to solve the business problems. In particular, make sure that data exist, are easily available, are transformable to represent the problem you want to solve.

Some of the reason for the failure of ML projects. 1. unfulfilled expectations 2. 9 Reasons why your machine learning project will fail. Highley recommended to read.

No ML without data

You are within an organization, or have a problem of your own that you want to solve, and you have propelry understand what oppourtunities the ML offers. As from above discussed, we already knew that we need data, computing power and know-how.

If you cannot get the data for technical, confidentiality or relational reasons, forget about the ML. Without data, we can do nothing, just we can't grow a plant without a seed. So, success of an ML project must be sponsored "from the top", by those have access the data, by those who can retrieve it and make it available.