Artificial Intelligence

Chapter 3

INTERIM SEMESTER 2021-22
BPL
CSE3007-LT-AB306
FACULTY: SIMI V.R.

Machine Learning

What is Learning

- Learning denotes changes in a system that enable the system to do the same task more efficiently next time.
- Learning is an important feature of “Intelligence”.

Definition

A computer program is said to learn from experience **E** with respect to some class of tasks **T** and performance measure **P**, if its performance at tasks in **T**, as measured by **P**, improves with experience **E**. (Mitchell 1997)
This means :

Given : A task **T,** A performance measure P, Some experience **E** with the task

Goal : Generalize the experience in a way that allows to improve your performance on the task.

Why do you require Machine Learning ?

- Understand and improve efficiency of human learning.
- Discover new things or structure that is unknown to humans.
- Fill in skeletal or incomplete specifications about a domain.

LEARNING

An agent is **learning** if it improves its performance on future tasks after making observations about the world.

FORMS OF LEARNING

Any component of an agent can be improved by learning from data. The improvements, and the techniques used to make them, depend on four major factors:

- Which *component* is to be improved.
- What *prior knowledge* the agent already has.
- What *representation* is used for the data and the component.
- What *feedback* is available to learn from.

Components to be learned

1. A direct mapping from conditions on the current state to actions.
2. A means to infer relevant properties of the world from the percept sequence.
3. Information about the way the world evolves and about the results of possible actions the agent can take.
4. Utility information indicating the desirability of world states.
5. Action-value information indicating the desirability of actions.
6. Goals that describe classes of states whose achievement maximizes the agent’s utility.

Well-Posed Learning Problems : Examples

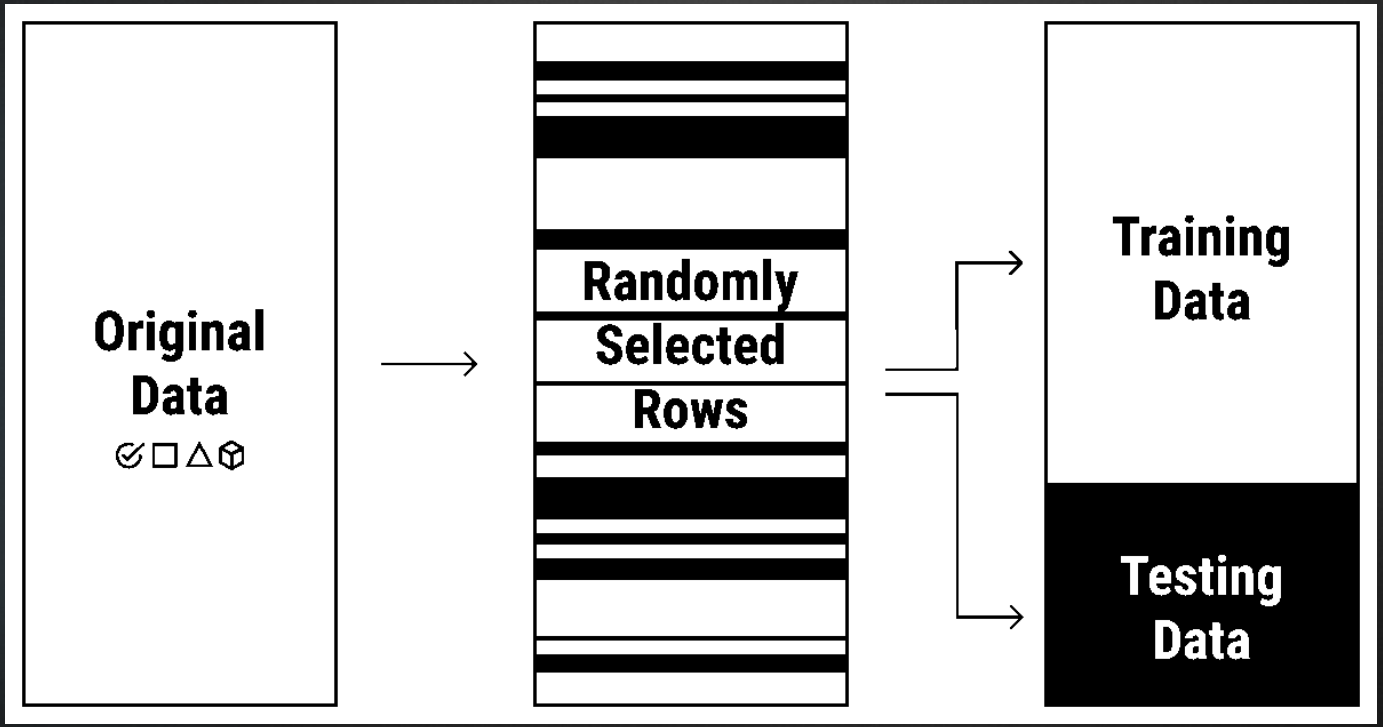
- A checkers learning problem
  - Task T : playing checkers
  - Performance measure P : percent of games won against opponents
  - Training experience E : playing practice games against itself

- A handwriting recognition learning problem
  - Task T : recognizing and classifying handwritten words within images
  - Performance measure P : percent of words correctly classified
  - Training experience E : a database of handwritten words with given classifications

- A robot driving learning problem
  - Task T : driving on public four-lane highways using vision sensors
  - Performance measure P : average distance traveled before an error (as judged by human overseer)
  - Training experience E : a sequence of images and steering commands recorded while observing a human driver

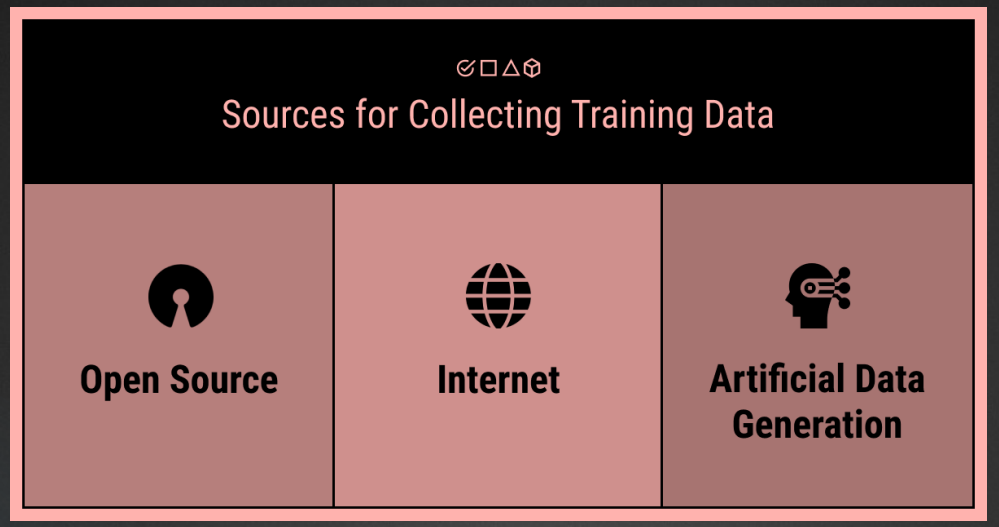
What Is Training Data in Machine Learning?

Training data is a set of samples (such as a collection of photos or videos, a set of texts or audio files, etc.) with assigned relevant and comprehensive labels (classes or tags) used to fit the parameters (weights) of a machine learning model with the goal of training it by example.



80%-20%
70%-20%

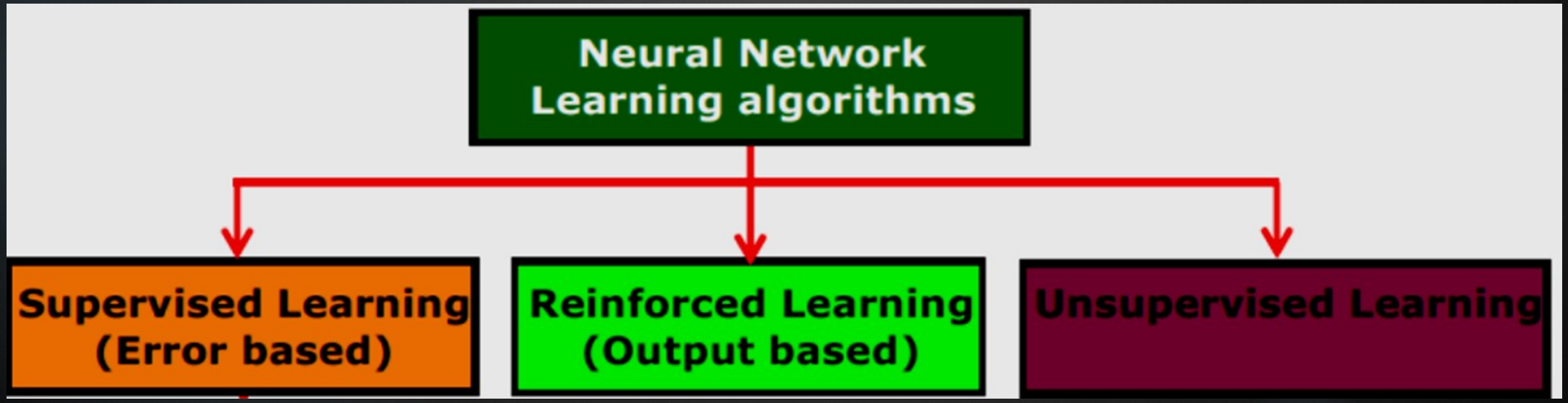
Where to get training data?



Open-source training data sets: The great benefit of this option is that it's free and it's already collected.
Web and IoT: This means that you use the Internet to collect the pieces of data. Alternatively, sensors, cameras, and other smart devices may provide you with the raw data that you will later need to annotate by hand.
Artificial training data sets: You have to create an ML model that will generate your data. This is a great way if you need large volumes of unique data to train your algorithm.

Machine Learning Areas

- **Supervised Learning:** Data and corresponding labels are given
- **Unsupervised Learning:** Only data is given, no labels provided
- **Semi-supervised Learning:** Some (if not all) labels are present
- **Reinforcement Learning**: An agent interacting with the world makes observations, takes actions, and is rewarded or punished; it should learn to choose actions in such a way as to obtain a lot of reward



Representation and prior knowledge

Supervised Learning

- A teacher is present during learning process and presents expected output.
- Every input pattern is used to train the network.
- Learning process is based on comparison, between network's computed output and the correct expected output, generating "error".
- The "error" generated is used to change network parameters that result improved performance.

Unsupervised Learning

- No teacher is present.
- The expected or desired output is not presented to the network.
- The system learns of it own by discovering and adapting to the structural features in the input patterns.

Reinforced learning

- A teacher is present but does not present the expected or desired output but only indicated if the computed output is correct or incorrect.
- The information provided helps the network in its learning process.
- A reward is given for correct answer computed and a penalty for a wrong answer.

- **In semi-supervised learning we are given a few labelled examples and must make what we can of a large collection of unlabelled examples. Even the labels themselves may not be the oracular truths that we hope for. Imagine that you are trying to build a system to guess a person's age from a photo. You gather some labeled examples by snapping pictures of people and asking their age. That's supervised learning. But in reality some of the people lied about their age.**

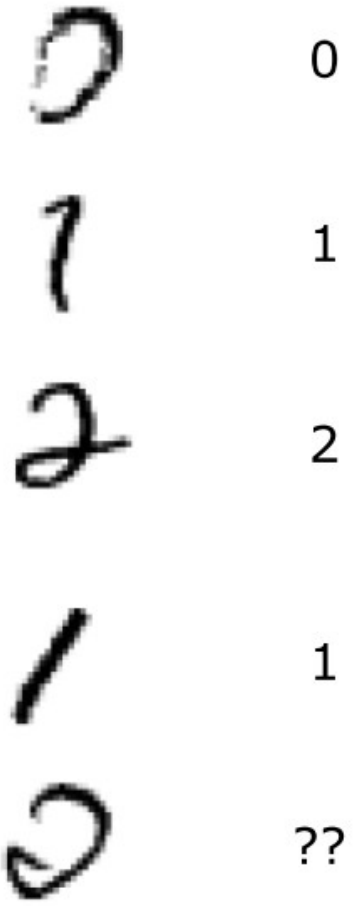
Example: Digit Recognition

Input: images / pixel grids

Output: a digit 0-9

Setup:

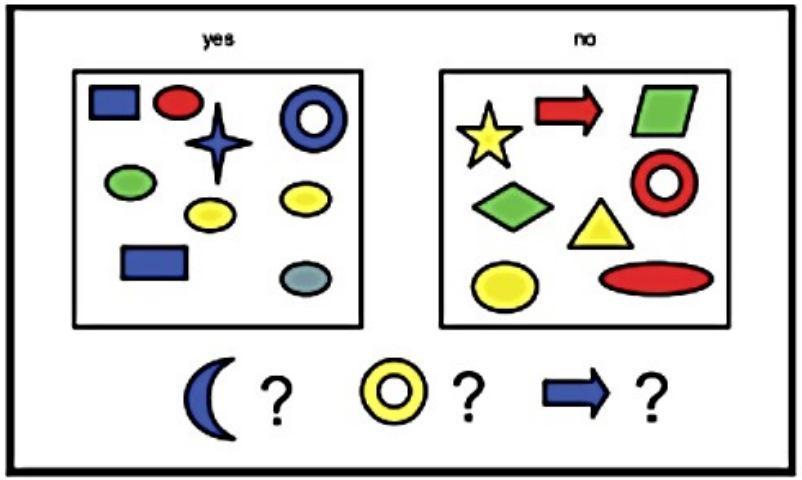
- Get a large collection of example images, each labeled with a digit
- Note: someone has to hand label all this data!
- Want to learn to predict labels of new, future digit images



Features: The attributes used to make the digit decision

- Pixels: (6,8)=ON
- Shape Patterns: NumComponents, AspectRatio, NumLoops
- …

Illustrating Classification Task



(a)

D features (attributes)

N cases

Color	Shape	Size (cm)	Label
Blue	Square	10	1
Red	Ellipse	2.4	1
Red	Ellipse	20.7	0

(b)

(a) Some labeled training examples of colored shapes, along with 3 unlabeled test cases. (b): Representing the training data as an N × D design matrix. Row i represents the feature vector x_i. The last column is the label, $y_i \in \{0, 1\}$.

References

http://www.myreaders.info/html/soft_computing.html

J-S R Jang and C-T Sun, Neuro-Fuzzy and Soft Computing, Prentice Hall, 1997

S. Rajasekaran and G.A. Vijayalaksmi Pai ,Neural Network, Fuzzy Logic, and Genetic Algorithms - Synthesis and Applications, (2005), Prentice Hall.