Model Building
Semi-Supervised Learning

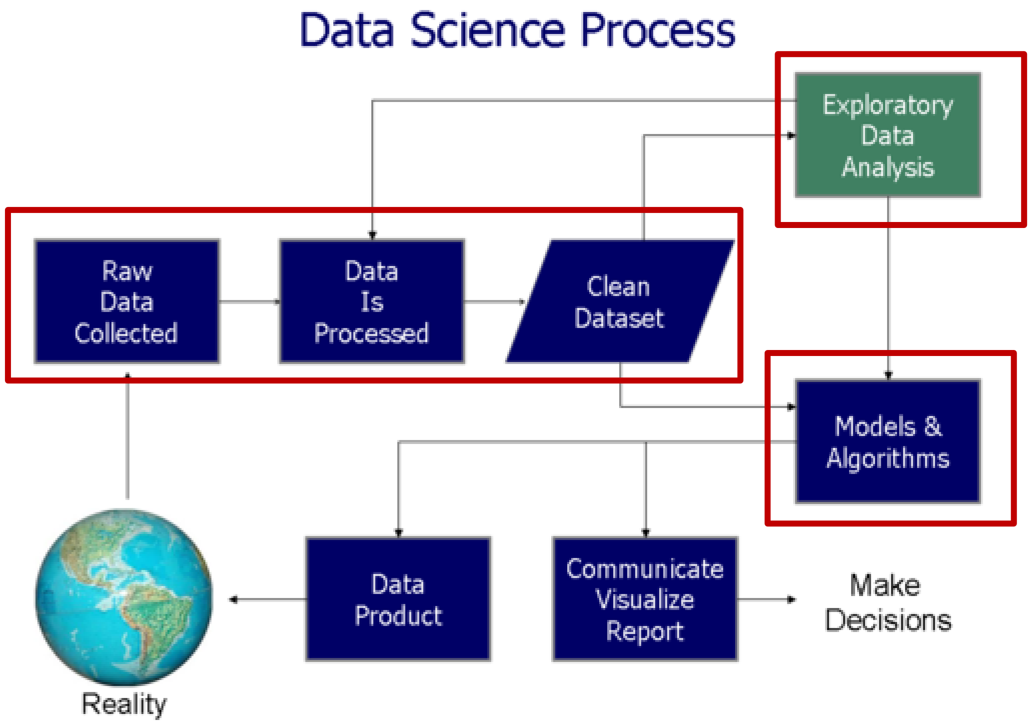
BY:- SHOHIT TYAGI

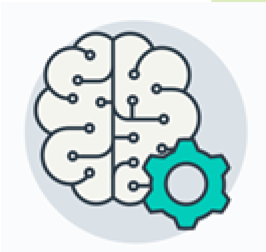
Contents

- Recap
- Semi-supervised Learning
- Reinforcement Learning

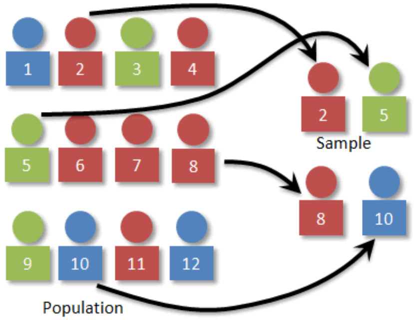
Recap

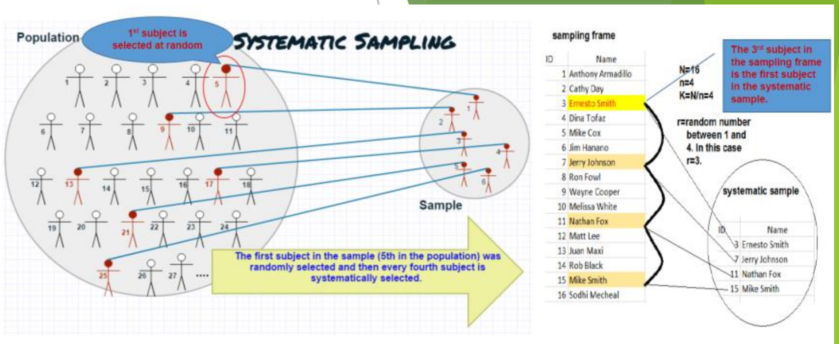
Data Science Process Flow

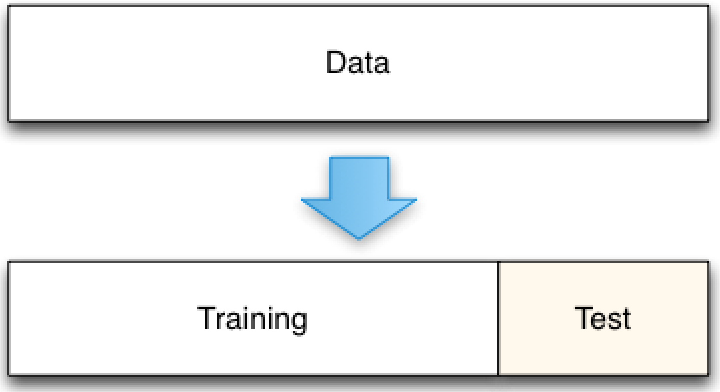


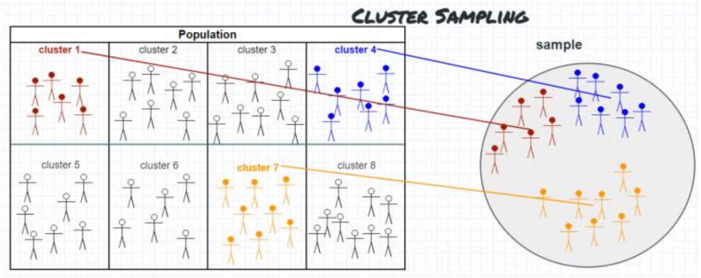


Split the data









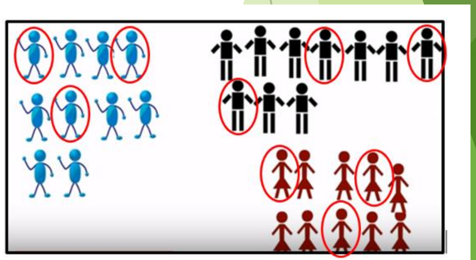
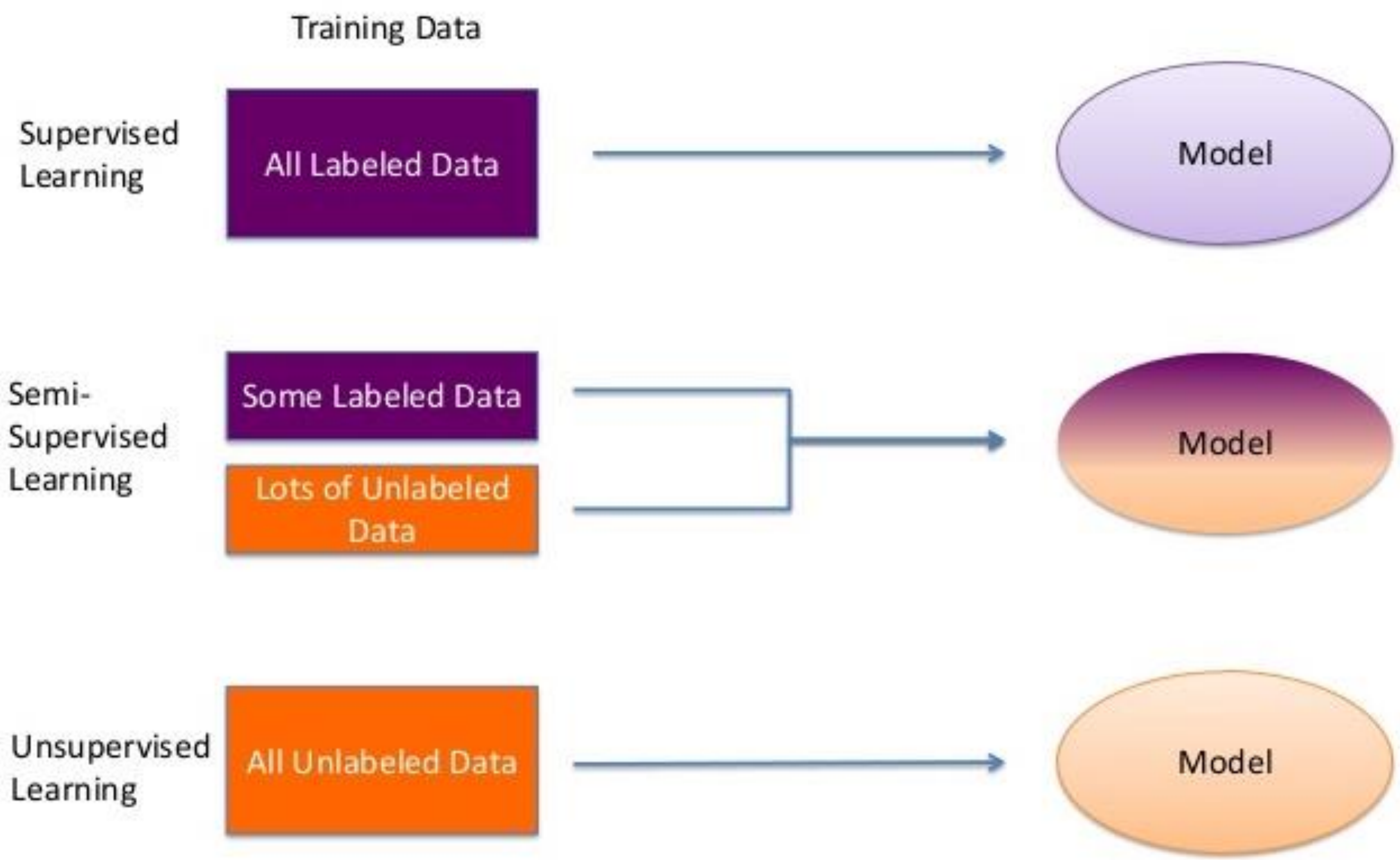


Fig. 3 Selecting datapoint from each subgroup

Modelling: Semi-Supervised Learning

Uses both Labelled and Unlabelled data for building models.



Need For Semi-supervised Learning

- Abundance of the Unlabeled Data
- High cost incurred in labelling data
    - Human annotation is boring and time taking
    - Labelling data may require domain knowledge

Semi-supervised Learning

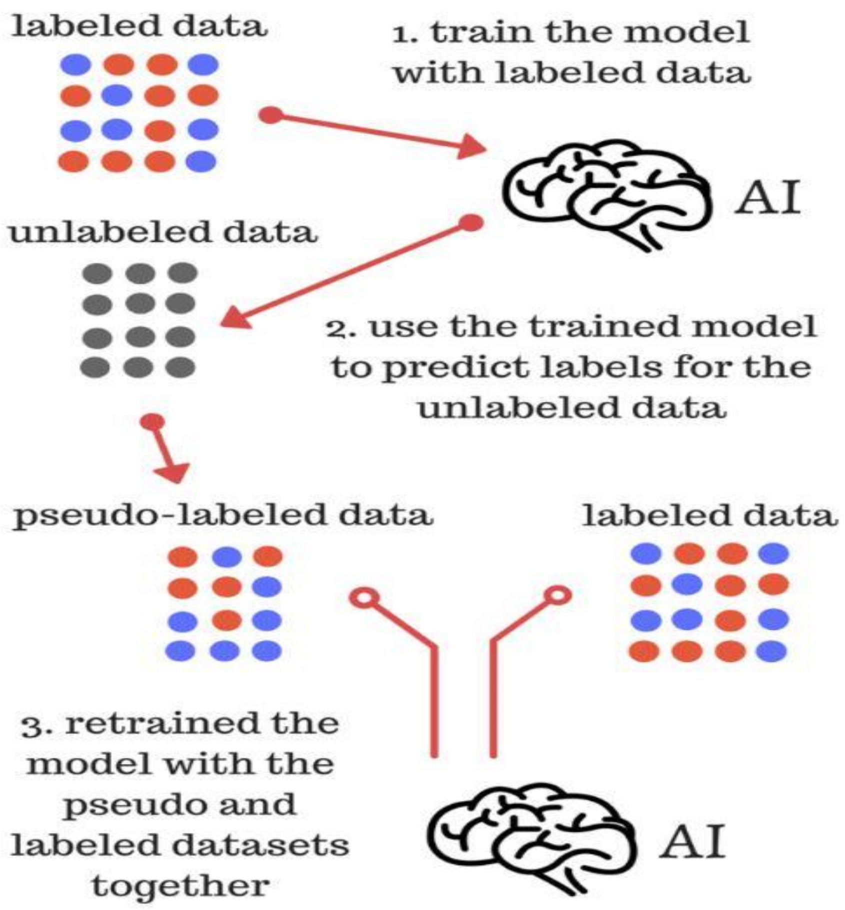
- It is a class of supervised learning tasks and techniques that also make use of unlabeled data for training.

- A small amount of labeled data with a large amount of unlabeled data.

Semi-supervised: Approaches

- Modelling Approach.
- Clustering Approach.

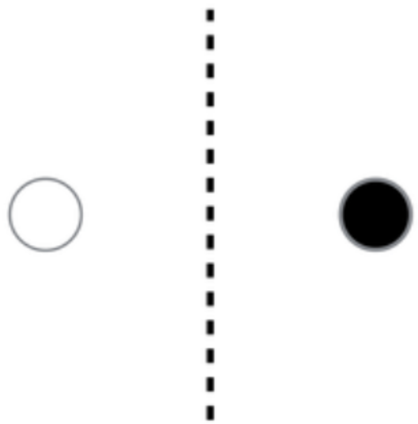
Semi-supervised: Modelling Process



Pseudo Labelling

Semi-supervised: Clustering

Two data points with labels

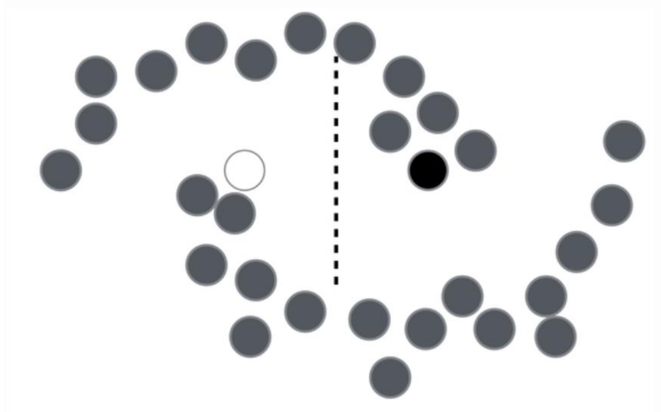


How would you label this point?

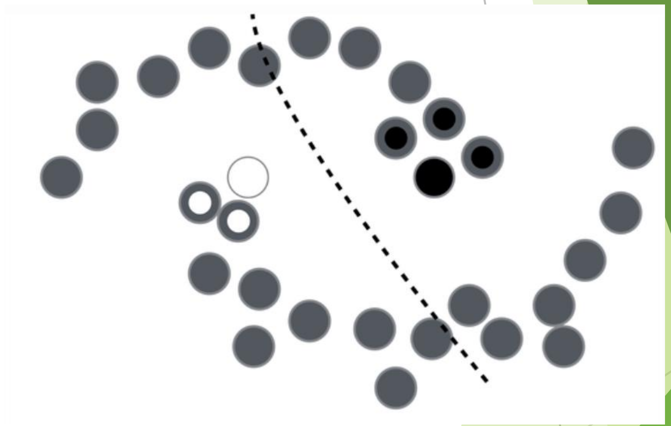
https://users.ics.aalto.fi/praiko/papers/ladder2.pdf

Semi-supervised: Clustering Contd.

What if you see all the unlabeled data?



Labels are homogeneous in densely populated space

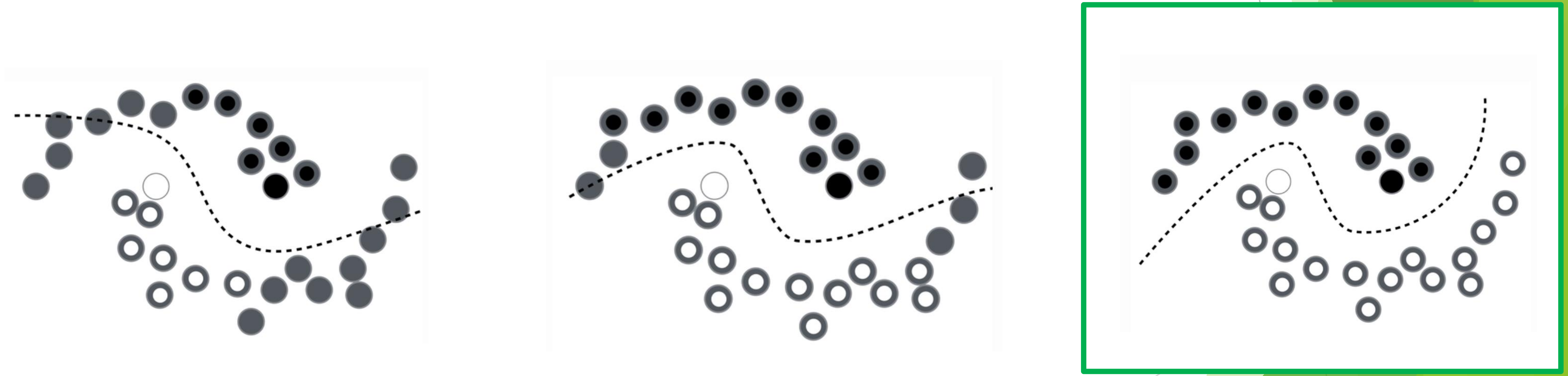


Data points close to each other belong to same class

*Assume some structure to the underlying distribution of data

Semi-supervised: Clustering Contd.

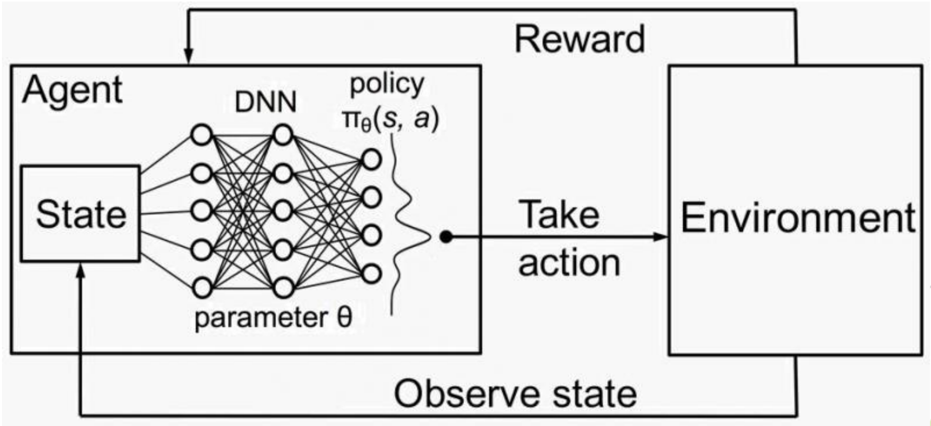
Iterating this assumption over the data points until all data points are assigned a label,



Applications

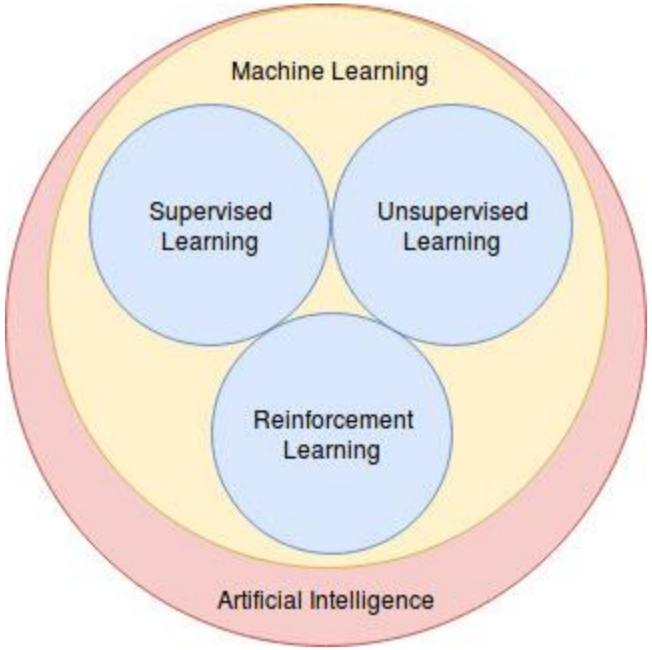
- **Speech Analysis**
- **Web Content Classification**

Reinforcement Learning



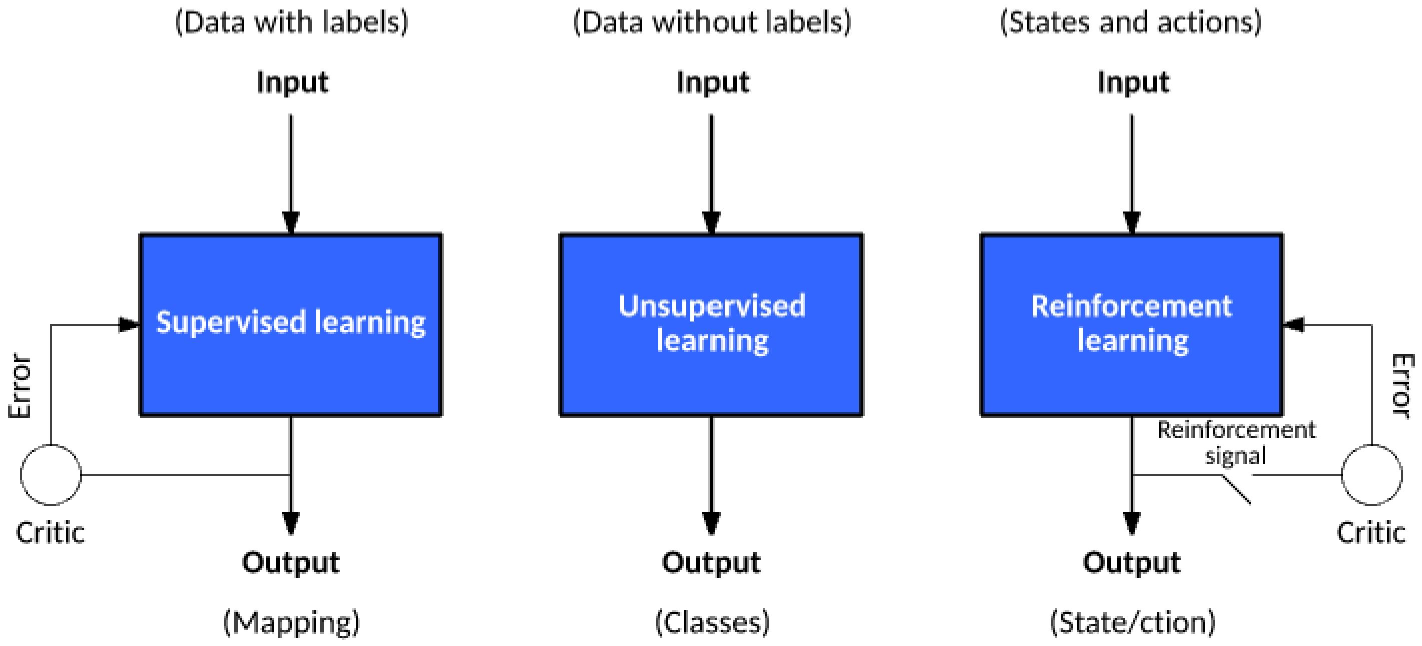
What is Reinforcement Learning?

- It is a type of machine learning that has an agent (like a robot) that learns how to behave in an environment by taking actions and quantifying the results.



Comparison of Machine Learning Methodologies

Unsupervised learning find the underlying patterns rather than the mapping



Supervised learning where feedback provided to the agent is correct set of actions for performing a task

Reinforcement learning uses rewards and punishment as signals for positive and negative behavior.

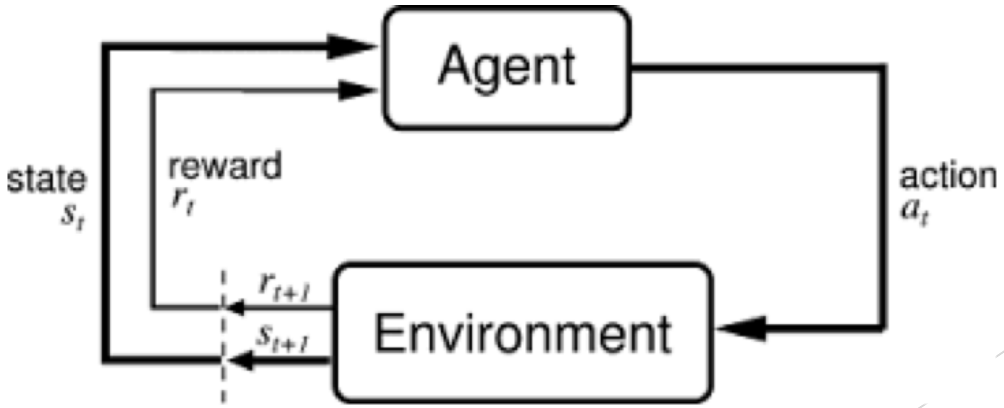
Terminologies

- **Agent**:
  - Agent takes actions, like a drone making a delivery, or Super Mario navigating a video game.
  - Algorithm is the agent.
- **Action:**
  - Action is the set of all possible moves the agent can make.
  - Agents choose among a list of possible actions like in video games, the list might include running right or left, jumping high or low, crouching or standing still.
- **Environment**:
  - World through which the agent moves.
  - Environment takes the agent’s current state and action as input and returns as output the agent’s reward and its next state.

Terminologies

- **State** :
  - State is a concrete and immediate situation in which the agent finds itself i.e. a specific place and moment.
  - It can the current situation returned by the environment, or any future situation.

- **Reward :**
  - Reward is the feedback by which we measure the success or failure of an agent's actions.
  - Example, in a video game, when Mario touches a coin, he wins points.



Example - PacMan

Goal: PacMan has to eat the food in the grid while avoiding the ghosts on its way

The total cumulative reward is PacMan winning the game

States
{location of PacMan in the grid world}

Reward
{PacMan receives a reward for eating food}

Punishment
{killed by the ghost i.e. lose the game}

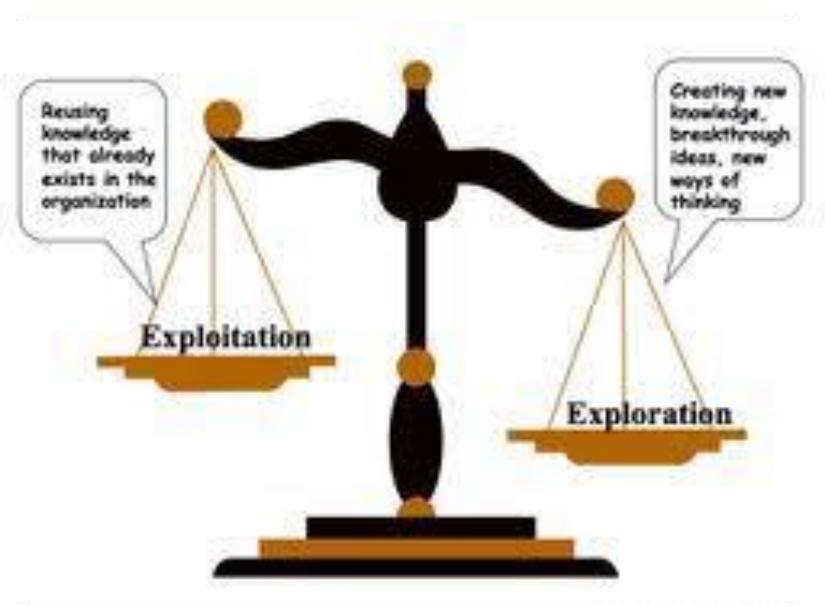
Agent
{PacMan}

Environment {Grid}

Exploration vs Exploitation trade-off

- In order to build an optimal policy, the agent faces the dilemma of exploring new states while maximizing its reward at the same time

Reusing knowledge that already exists

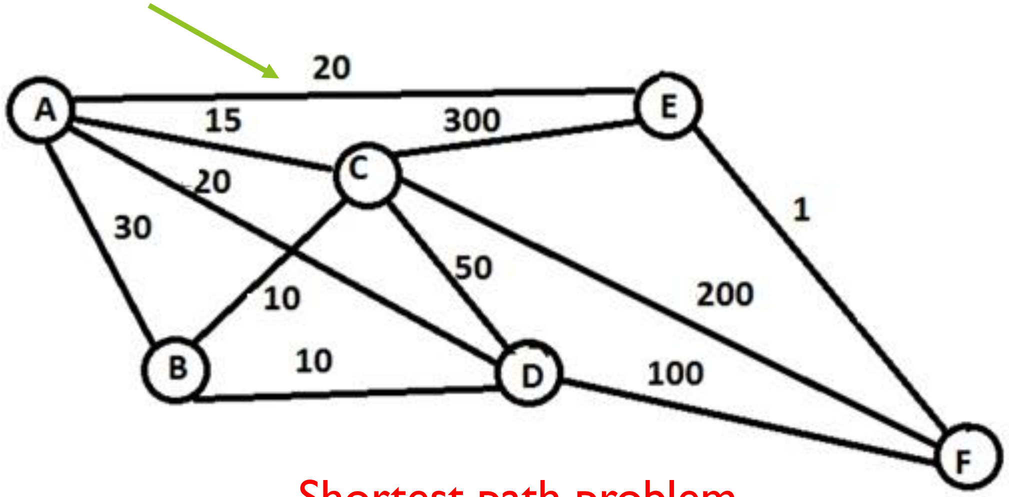


Creating new knowledge, breakthrough idea

Reinforcement Learning Framework

Markov Decision Process - Our task here is to maximize our rewards by choosing the correct policy

The numbers at each edge between two places represent the cost taken to traverse the distance.



Shortest path problem

- set of **States** are the nodes, viz {A, B, C, D, E, F}
- **Action** to take is to go from one place to other, {A -> B, C -> D}
- **Reward** function is the value represented by edge, i.e. cost
- **Policy** is the "way" to complete the task, viz {A -> E -> F}

Goal – Go from place A to place F, with as low cost as possible

Techniques in Reinforcement Learning

- Q-Learning
- SARSA (State-Action-Reward-State-Action)
- Deep Q-Networks (DQNs)

Thank you