

INTRODUCTION

---

**REINFORCEMENT LEARNING**

**ALL MACHINE LEARNING IS  
NOTHING BUT GLORIFIED  
CURVE FITTING.**

**Andrew Ng**

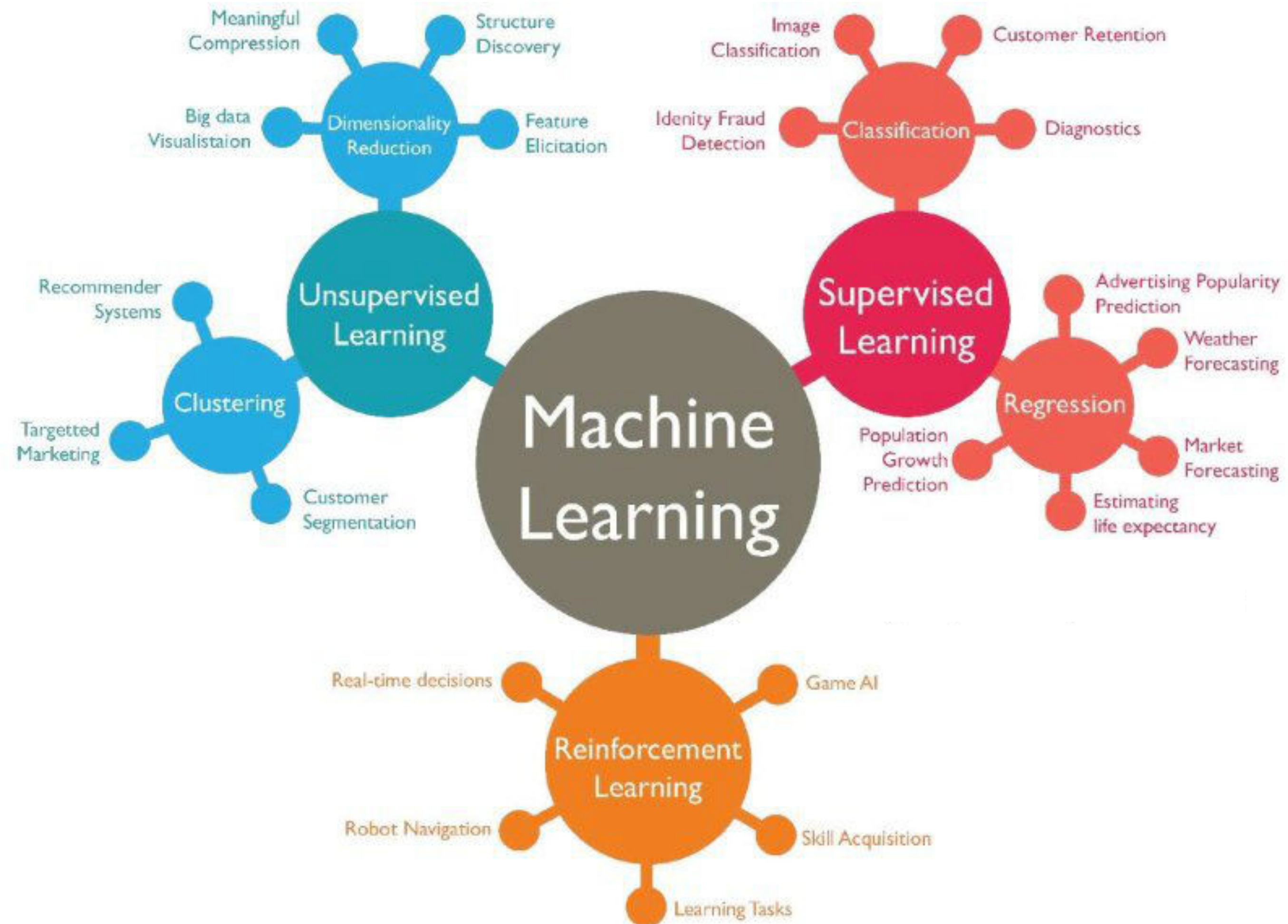
A Venn diagram consisting of four concentric ellipses. The innermost ellipse is black and labeled 'Deep Learning e.g. MLPs'. The next ellipse is red and labeled 'Representation Learning e.g. Shallow Autoencoders'. The third ellipse is blue and labeled 'Machine Learning e.g. Logistic Regression'. The outermost ellipse is green and labeled 'AI e.g. Knowledge Bases'. The ellipses are nested, indicating that Deep Learning is a subset of Representation Learning, which is a subset of Machine Learning, which is a subset of AI.

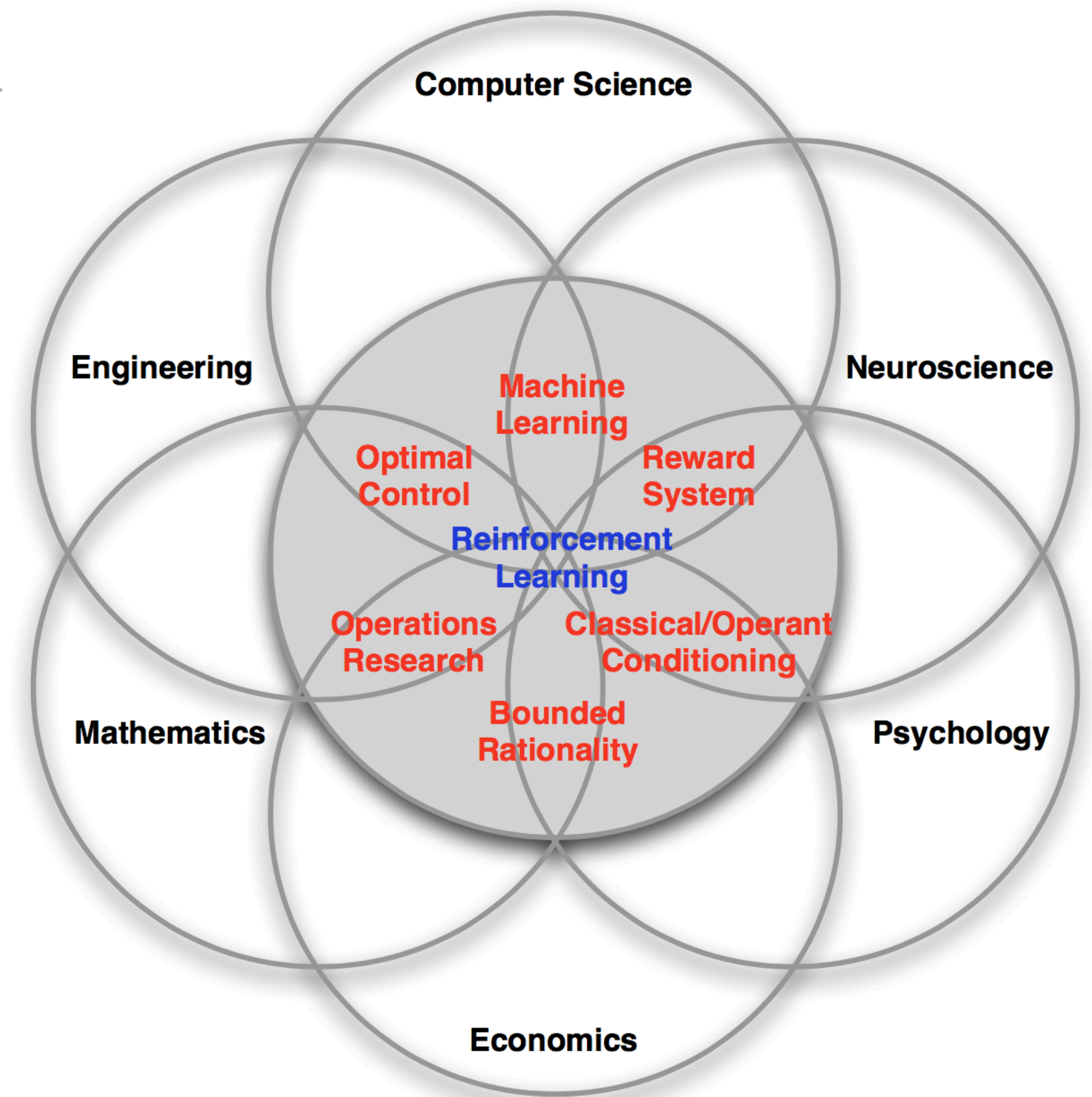
**Deep Learning**  
e.g. MLPs

**Representation Learning**  
e.g. Shallow Autoencoders

**Machine Learning**  
e.g. Logistic Regression

**AI**  
e.g. Knowledge Bases





## WHAT IS REINFORCEMENT LEARNING?

---

- ▶ There is no supervisor, only a reward signal.
- ▶ Feedback is delayed, not instantaneous.
- ▶ Time really matters - sequential.
- ▶ Agent's action affect the subsequent data it receives.

## EXAMPLES OF REINFORCEMENT LEARNING

---

- ▶ Fly stunt helicopters.
- ▶ Make two Neural Network Agents play Scotland Yard.
- ▶ Make a fully autonomous pizza delivery drone.
- ▶ Defeat the world champion at Backgammon.
- ▶ Train Skynet, take over the world.
- ▶ Defeat Dendi at Dota2.



# LET'S WATCH SOME VIDEOS.

- ▶ Flappy Bird.
- ▶ Doom.
- ▶ Santi's walking video.
- ▶ Helicopter manoeuver.
- ▶ PacMan.



# REWARDS

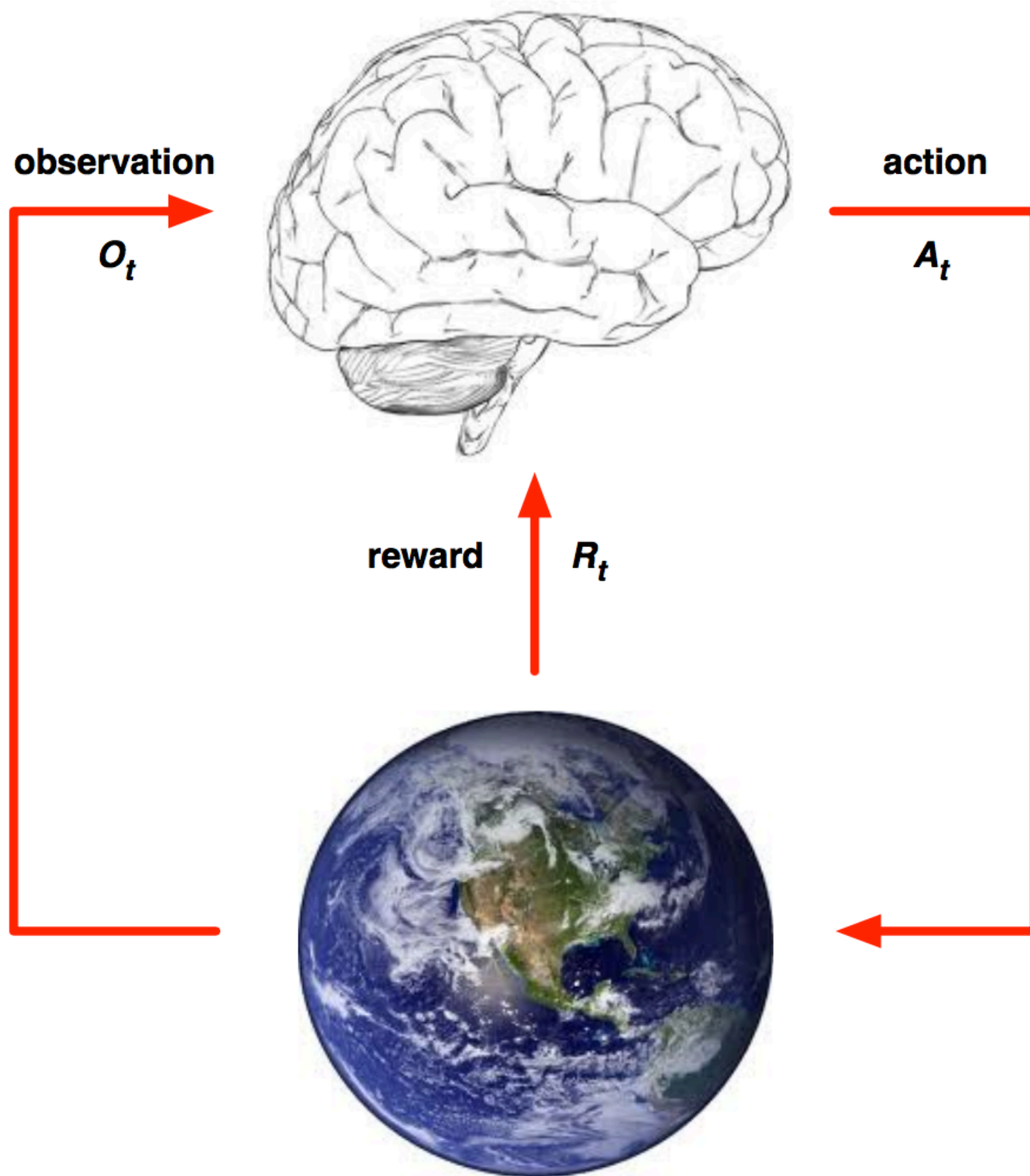
- ▶ A reward  $R(t)$  is a scalar feedback signal.
- ▶ Indicates how well an agent is doing at time step  $t$ .
- ▶ The agent's job is to maximise the cumulative reward. Or minimise if it is modelled as 'cost.'
- ▶ Reinforcement learning is based on the reward hypothesis.

### Definition (Reward Hypothesis)

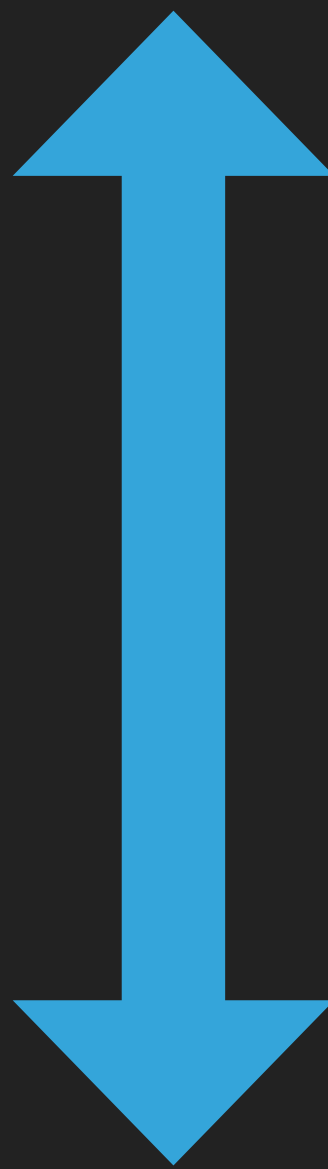
*All* goals can be described by the maximisation of expected cumulative reward

- Fly stunt manoeuvres in a helicopter
  - +ve reward for following desired trajectory
  - -ve reward for crashing
- Defeat the world champion at Backgammon
  - +/-ve reward for winning/losing a game
- Manage an investment portfolio
  - +ve reward for each \$ in bank
- Control a power station
  - +ve reward for producing power
  - -ve reward for exceeding safety thresholds
- Make a humanoid robot walk
  - +ve reward for forward motion
  - -ve reward for falling over
- Play many different Atari games better than humans
  - +/-ve reward for increasing/decreasing score

# NEXT TIME, ON REINFORCEMENT LEARNING.



Agent



Environment

**DOUBTS?**  
**QUESTIONS?**

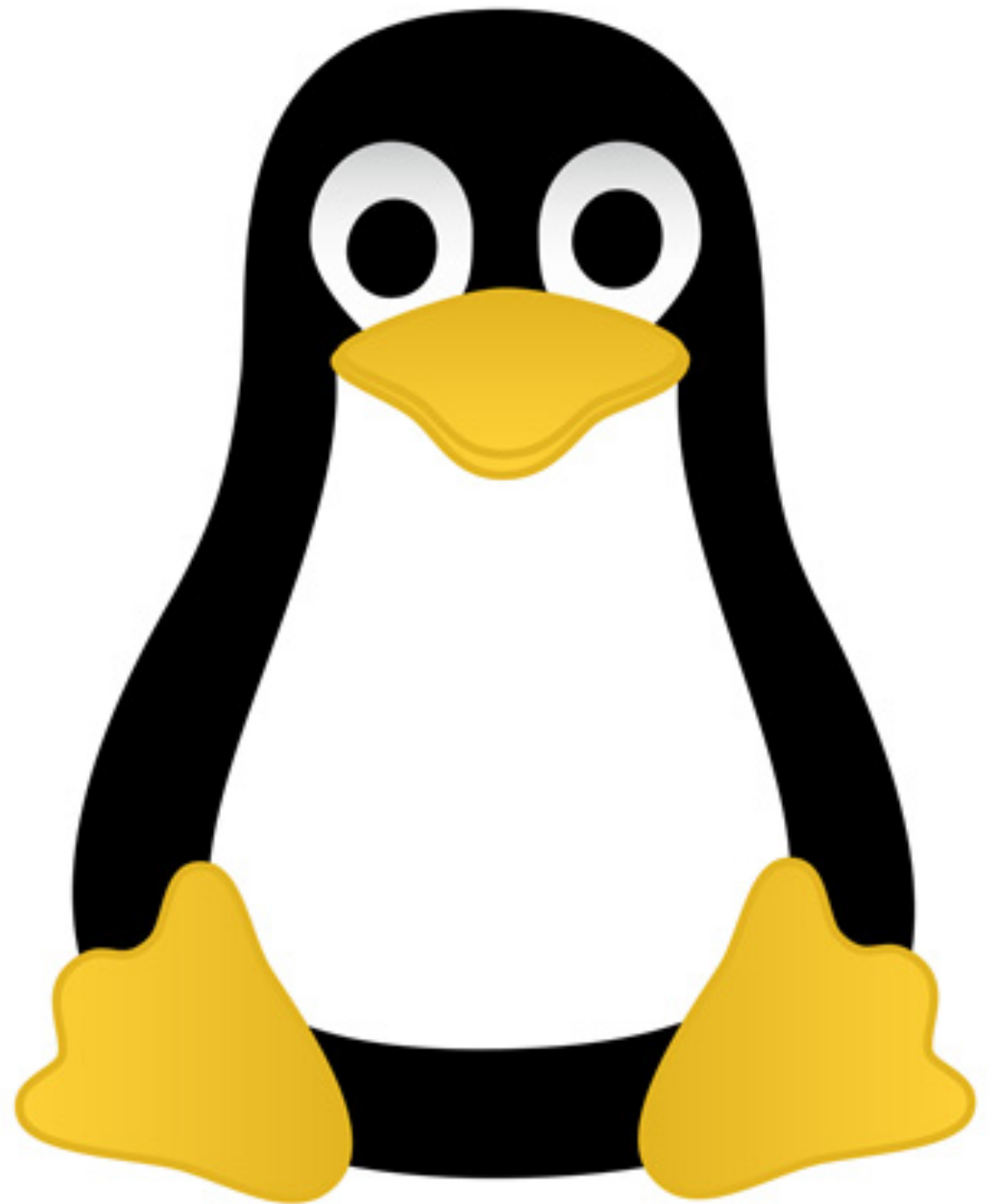
---

**IF YOU KNOW THE SYSTEM  
WELL ENOUGH, YOU CAN DO  
THINGS THAT AREN'T  
SUPPOSED TO BE POSSIBLE.**

**Linus Torvalds**

GET AN UNIX BASED SYSTEM ON YOUR LAPTOP

DUAL BOOT  
SESSION  
<DATE, TIME,  
LOCATION>



HANDS ON

---

**PYTHON3**



## GETTING PYTHON3

- ▶ Once you're on an unix based system, installing python3 will be easy.
- ▶ Just install anaconda.



# ANACONDA

## GETTING PYTHON3

- ▶ You can download the graphical installer or just install it from the terminal. The documentation should guide you.
- ▶ Anaconda will also install a version of python3.



**\$ CONDA**

# DOWNLOAD ANACONDA DISTRIBUTION

Version: 4.4.0 | Release Date: May 31, 2017

Download for:   

## High-Performance Distribution

Easily install 1,000+ [data science packages](#)

## Package Management

Manage packages,  
dependencies and environments  
with [conda](#)

## Portal to Data Science

Uncover insights in your data  
and create interactive  
visualizations

## Download for Your Preferred Platform



Windows



macOS



Linux

## Anaconda 4.4.0 For macOS Graphical Installer

<https://cs231n.github.io/python-numpy-tutorial>

<https://LearnPythonTheHardWay.org/python3/>

---

**LEARN PYTHON THE HARD WAY**

## HOMework

- ▶ Get any unix based system on your laptop. Attend the dual-boot session if you need help.
- ▶ Get python3 on it. GET anaconda cause it has built in package management systems.
- ▶ Revise python3. Download numpy, pandas and matplotlib using anaconda's package manager.
- ▶ If time permits, go through the documentation of numpy, pandas, matplotlib.

**ALL SLIDES AND  
NOTICES WILL BE PUT UP  
ON A GITHUB  
REPOSITORY.  
LINK WILL BE SHARED  
SOON**

## IF YOU GUYS NEED HELP, MAIL US AT:

- ▶ Preetham Reddy: [f2015174@goa.bits-pilani.ac.in](mailto:f2015174@goa.bits-pilani.ac.in)
- ▶ Sahith Dambekodi: [f2015192@goa.bits-pilani.ac.in](mailto:f2015192@goa.bits-pilani.ac.in)
- ▶ Sebastin Santy: [f2015357@goa.bits-pilani.ac.in](mailto:f2015357@goa.bits-pilani.ac.in)
- ▶ WazzerMan: [f2015003@goa.bits-pilani.ac.in](mailto:f2015003@goa.bits-pilani.ac.in)



**DOUBTS?**  
**QUESTIONS?**

**THANK YOU.**