Model-Free Deep Reinforcement Learning algorithms applied to autonomous systems

Candidate: Piero Macaluso - s252894

Supervisors:

Prof. Pietro Michiardi — EURECOM, France

Prof. Elena Baralis — Politecnico di Torino, Italy

November 21, 2019

This master thesis was developed at EURECOM (Sophia Antipolis, Biot, France)
in collaboration with

Prof. Pietro Michiardi (EURECOM)
Prof. Elena Baralis (Politecnico di Torino)

Table of contents

Reinforcement Learning Background

Beyond supervised and unsupervised learning

Supervised Learning

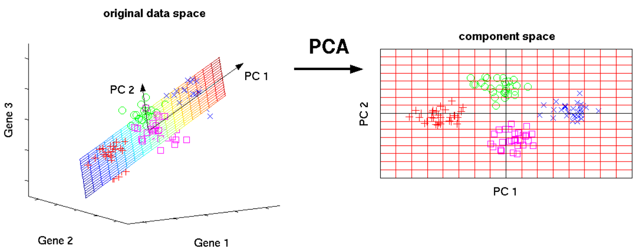
- **Data**: (x, y) where x is data, y is label
- **Goal**: Learn a function $f : x \rightarrow y$
- **Examples**: **Classification**, object detection, semantic segmentation, image captioning, ...

 $\longrightarrow$ CAT

Beyond supervised and unsupervised learning

Unsupervised Learning

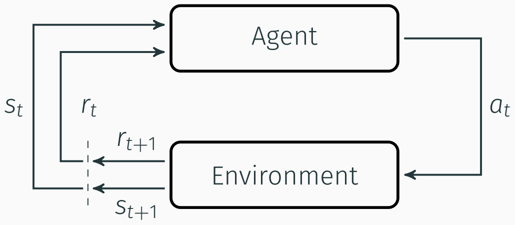
- **Data**: No more labels, just data.
- **Goal**: Learn some underlying hidden structure of the data.
- **Examples**: Clustering, **dimensionality reduction**, feature learning, density estimation, ...



Scholz, "Approaches to analyse and interpret biological profile data".

Problems involving an **agent** interacting with an **environment**, which provides numeric **reward signals**.

Goal: Learn how to take actions in order to maximize reward



Sutton and Barto, *Reinforcement learning: An introduction*.

Reinforcement Learning involves

- Optimization
- Delayed Consequences
- Exploration
- Generalization

Components of the Agent

- **Policy**: agent's behaviour function

$$\textbf{Deterministic: } \pi(s) = a$$

$$\textbf{Stochastic: } \pi(a|s) = \mathbb{P}[A_t = a|S_t = s]$$

- **Value Function**: agent's behaviour function

$$\textbf{State Value: } V^{\pi}(s) = \mathbb{E}\left[\sum_{t\geq 0} \gamma^k r_t | s_0 = s, \pi\right]$$

$$\textbf{Action Value: } Q^{\pi}(s,a) = \mathbb{E}\left[\sum_{t\geq 0} \gamma^k r_t \middle| s_0 = s, a_0 = a, \pi\right]$$

- **Model**: agent's representation of the environment

Categorizing Reinforcement Learning agents

- Value Based
  - No Policy (implicit)
  - Value Function
- Policy Based
  - Policy
  - No value function
- Actor Critic
  - Policy
  - Value function

- Model Free
  - Policy and/or value function
  - No Model
- Model Based
  - Policy and/or value function
  - Model

Reinforcement Learning aim

Learn to make **good sequences of decisions**.

Fundamental challenge in artificial intelligence and machine learning is learning to make good decisions under uncertainty.

Let’s go Deep!

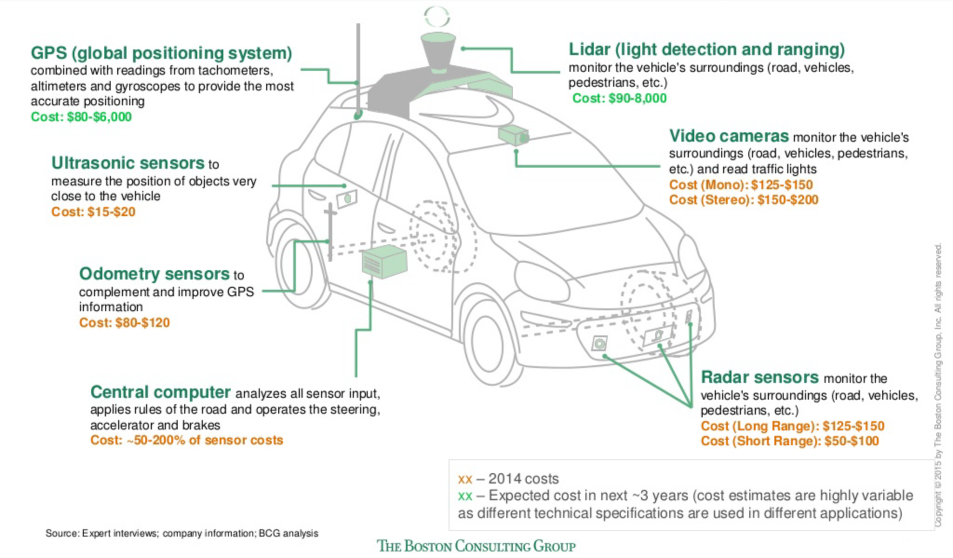
Google DeepMind’s Deep Q-learning playing Atari Breakout

`https://www.youtube.com/watch?v=V1eYniJ0Rnk`

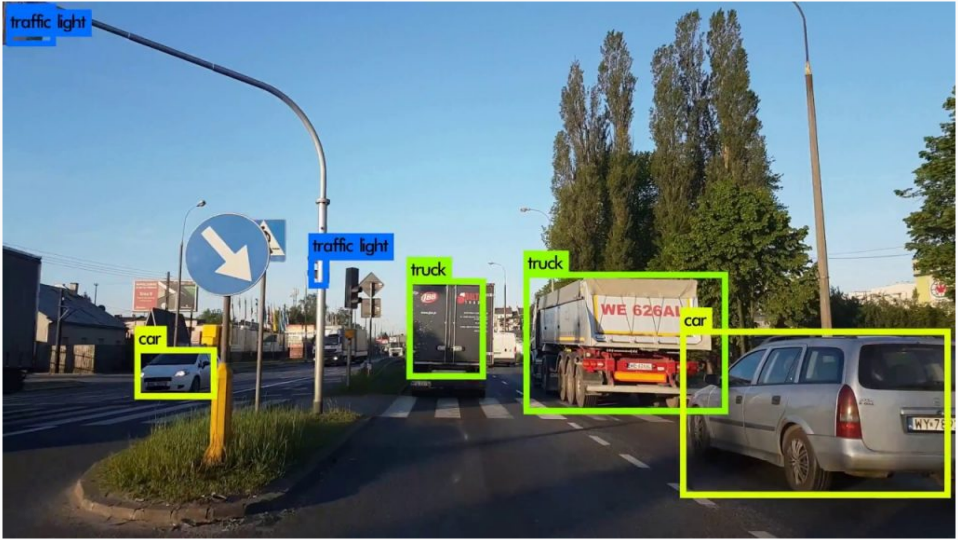
Mnih et al., “Playing atari with deep reinforcement learning”.

Reinforcement Learning for Autonomous Systems

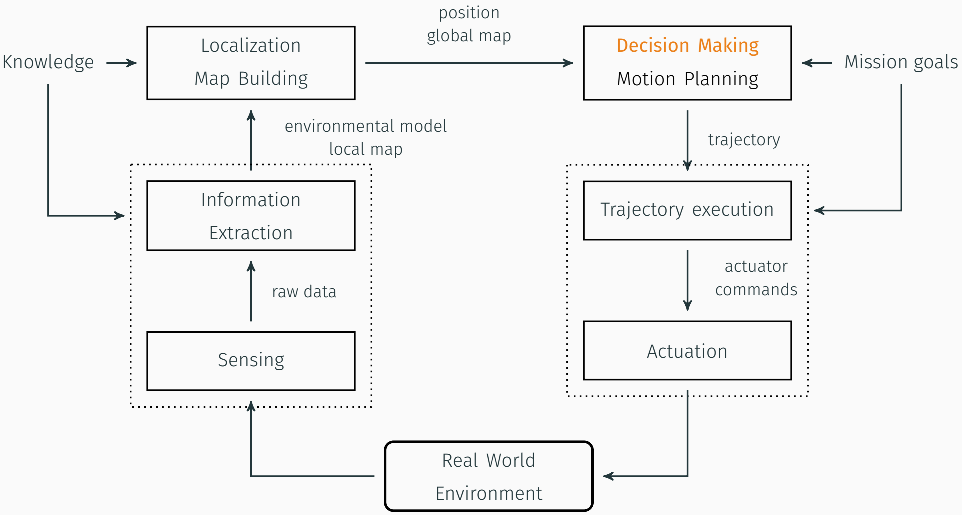
State-of-the-art Autonomous Driving Systems



Deep Learning for autonomous vehicles



State-of-the-art Autonomous Driving Systems



Learning to drive in a Day

Learning to drive in a day

`https://www.youtube.com/watch?v=eRwTbRtnT1I`

Kendall et al., “Learning to Drive in a Day”.

Urban Driving with End-to-End Deep Learning

`https://www.youtube.com/watch?v=26Or4QbLbMM`

WAYVE, *Learning to Drive like a Human.*

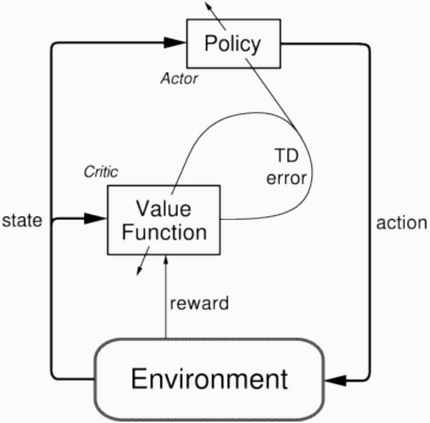
Model-Free Actor Critic methods

Critic Network

Estimates the value function. This could be the action value Q or state value V.

Actor Network

Updates the policy distribution in the direction suggested by the Critic (such as with policy gradients).



Sutton and Barto, *Reinforcement learning: An introduction*.

Model-Free algorithms exploited

Deep Deterministic Policy Gradient (DDPG)

- DDPG is an off-policy algorithm.
- Ornstein–Uhlenbeck process noise for exploration
- Countinuous action spaces

Soft Actor-Critic (SAC)

- SAC is an off-policy algorithm which exploits entropy-regularized reinforcement learning
- Auto-tune parameters: Less hyper-parameters, less tuning
- Suitable for Real-World Experiments

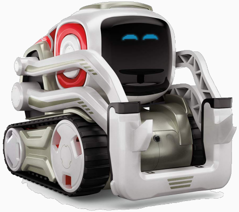
Lillicrap et al., "Continuous control with deep reinforcement learning".
Haarnoja et al., "Soft actor-critic algorithms and applications".

Outline of the Project

Main Objectives

- Building a **control system** and an **interface** between Cozmo robot and algorithms using OpenAI Gym.
- **Real World** Reinforcement Learning experiments.
- Comparison between DDPG and SAC.
- Strengths and Weaknesses of Reinforcement Learning.

Anki Cozmo - Not just a toy robot



Why Cozmo?

- Small and portable
- 30fps VGA Camera
- Powerful mechanics
- Python SDK and interfaces

The Reinforcement Learning Control System Stack

- Human Level Control through a WebApp (**Flask**, **Python** and **Javascript**)
- Algorithm written in **Python**
- **PyTorch** as Deep Learning Framework
- **OpenAI Gym** Framework for Reinforcement Learning
- **Cozmo SDK**

Human Control Panel

COZMO Reinforcement Learning Dashboard

Reinforcement Flow

Commands to manage episode and enable human remote control

Enter	Start/Stop Episode
Backspace	Stop and Forget last episode
Q	Toggle Test Phase
C	Toggle Save'n'Close Phase

Commands to restore the correct position of Cozmo

W / A S D	Drive Forwards Left / Back / Right
R T / F G	Move LIFT/HEAD up and down
Shift	Hold to Move Faster (Driving, Head and Lift)
Alt	Hold to Move Slower (Driving, Head and Lift)

Info

Phase	Train
Episode	Started
Discarded	FALSE
Save and Close	FALSE

Other Info

Phase	Mark
Episode	Jacob
Discarded	Larry

A Study of Reinforcement Learning

A Master Thesis by Piero Macaluso.

Supervisors:
Prof. Elena Baralis, Politecnico di Torino (Torino, Italy)
Prof. Pietro Michiardi, Eurecom (Biot, France)

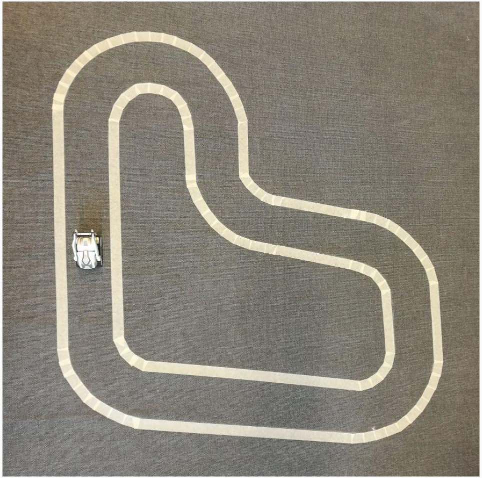
Anki Cozmo

Developed using Anki Cozmo Robot and its Open Source SDK

#SaveAnki | #SaveVector | #SaveCozmo

The Track

- Contrast between lane and asphalt.
- Lane width comparable to the real one.
- Fewer Reflections.
- Easily Repeatable.



First Experiment Results

Results - Training Phase

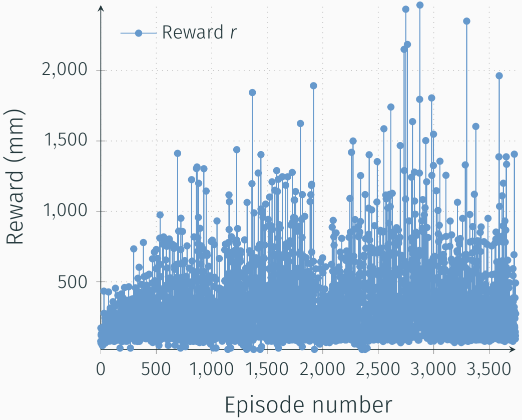


Figure 1: Total reward for each episode. The maximum value of almost 3 meters between episode 2500 and 3000.

Results - Test Phase

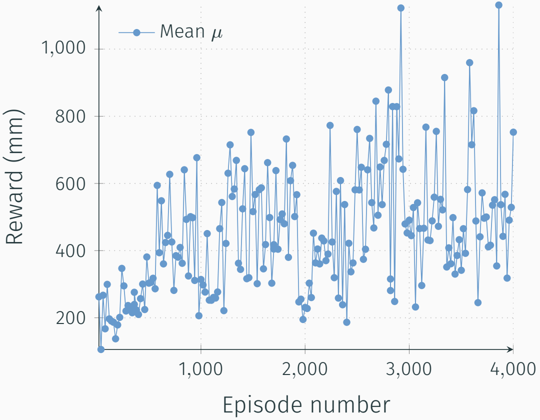


Figure 2: Test Phase every 20 episodes of learning. Mean Reward over 5 episode of test.

Best Episodes - Episode 2748 ans 2876

Reinforcement Learning Training Episode with Anki Cozmo

`https://pieromacaluso.github.io/episode`

Considerations

- These results might appear not so extraordinary.
- In reality, it is like teaching a baby how to drive a car!
- It is a process which starts from scratch. **From Zero to Hero!**

Reflections and possible developments

Issues

- Hunger for data.
- Human Bias.
- Narrow view of the camera.

Possible improvements

- Increase the number of epochs for each episode.
- Apply gradient clipping.
- Prioritized Experience Replay.
- Improve Fault Recovery System.

Possible developments

- Increase the number of data (e.g sensors).
- Overcome the limitations of Cozmo.
  - Anki Vector
  - Donkey Car
- Neural Network for object detection.

Questions?

Thank you!

References

- Haarnoja, Tuomas et al. “Soft actor-critic algorithms and applications”. In: *arXiv preprint arXiv:1812.05905* (2018).
- Kendall, Alex et al. “Learning to Drive in a Day”. In: *arXiv preprint arXiv:1807.00412* (2018).
- Lillicrap, Timothy P et al. “Continuous control with deep reinforcement learning”. In: *arXiv preprint arXiv:1509.02971* (2015).
- Mnih, Volodymyr et al. “Playing atari with deep reinforcement learning”. In: *arXiv preprint arXiv:1312.5602* (2013).

References ii

- Scholz, Matthias. “Approaches to analyse and interpret biological profile data”. In: (2006).
- Sutton, Richard S and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- WAYVE. *Learning to Drive like a Human*. https://wayve.ai/blog/driving-like-human. 2019. (Visited on 04/03/2019).