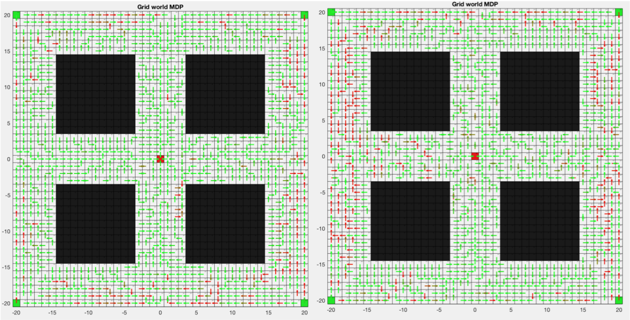

Learning from noisy demonstrations : a exploration/exploitation tradeoff

Louis Faury

Advisors : Mahdi Khoramshahi & Andrew Sutcliffe
Semester Project at LASA

April 17, 2017

Plan

Plan

- For long and complex tasks : common machine learning algorithm are usually very slow to converge
- Accelerate learning via prior knowledge of the environment or task : provide a **demonstration** of the task
- Framework of *learning from demonstration* (LfD)[1]

$\longrightarrow$ Ex. : robotic arm grabbing a cup

: maze solver

[1] Aude G. Billard, Sylvain Calinon, and Rüdiger Dillmann. "Learning from Humans". *Springer Handbook of Robotics*. Ed. Bruno Siciliano and Oussama Khatib. Cham: Springer International Publishing, 2016. 1995–2014. Web.

- How to take the teacher's demonstration into account?
  - Exactly reproduce the teacher's actions
  - Use demonstration data to build a representation of the environment's dynamics
  - **Use the teacher demonstration as an exploration baseline**

- Child learning to dance : first follow its dance teacher moves, before trying out new ones once he feels he has exploited the teacher's recommandation

$\Rightarrow$ notion of **compliance** w.r.t the teacher.

■ **Goal** :

- Introduce a theoretical framework for compliance-based learning
- Grasp ideas and intuition about how such an approach can
  - Speed up the learning
  - Overcome some possible mentor's sub-optimality.
  - Generalize to *transfer learning*

in a **reinforcement learning framework**.

■ **Approach** :

- Create a simple but generic environment and task
- Solve it using classical RL method
- Implement compliant-based learning method
- Compare them with classical methods, evaluate their pros and cons

Plan

RL :

- Framework in which an agent (or a learner) learns its actions from interaction with its environment
- The environment generates scalar values called rewards, that the agent is seeking to maximize over time.

Under a Markovian asumption for the dynamics and reward system, the reinforcement learning problem can be formulated as a *Markov Decision Process* :

$$(\mathcal{S}, \mathcal{A}(\mathcal{S}), \mathcal{P}^a_{ss'}, \mathcal{R}^a_{ss'}) \tag{1}$$

where :

$$\begin{aligned} \mathcal{P}^a_{ss'} &= \underbrace{\mathbb{P}(s_{t+1} = s' \mid s_t = s,\, a_t = a)}_{dynamics} & \mathcal{S} &: \text{ state space} \\ \mathcal{R}^a_{ss'} &= \underbrace{\mathbb{E}\left[r_t \mid s_{t+1} = s',\, s_t = s,\, a_t = a)\right]}_{immediatereward} & \mathcal{A}(\mathcal{S}) &: \text{ action space} \end{aligned} \tag{2}$$

RL :

- Define state value and action value function under a policy (probabilistic decision rule) $\pi : \mathcal{S} \rightarrow \mathcal{A}$:

$$\begin{aligned} V^{\pi}(s) &= \mathbb{E}_{\pi}\left[\sum_{i} \gamma^{i} r_{t+i+1} \,|s_t = s\right] \\ Q^{\pi}(s,a) &= \mathbb{E}_{\pi}\left[\sum_{i} \gamma^{i} r_{t+i+1} \,|s_t = s, a_t = a\right] \end{aligned} \tag{3}$$

- All algorithm computing optimal policies rely on various mix of a *Generalized Policy Iteration* :
  1. Evaluate the current policy (DP,..)
  2. Improve the current policy (greedization)
  3. Repeat

Plan