Projects List for EE-568 Reinforcement Learning @ EPFL

April 10, 2025

1 Theory Projects

Below you can find 12 possible project directions for the *theory track*. Pick **one** of them (e.g., "Policy Gradient") and read the 2–3 papers listed for it. The steps of a project should be the following:

- Quickly scroll through the papers to understand the general idea. Skip the proofs and the appendix for the moment, and try to understand what problem each of the papers is solving.
- Understand how the results of the papers we suggested compare with each other. (For example, they might consider different setups, or they might build on top of each other.)
- Understand whether some of the papers have limitations that other papers in the list solve.
- Understand how the papers could be improved. For example try to think which research questions remains open or seem within reach.
- Write a report of 6–8 pages answering the above questions. Please do include technical writing (i.e., include relevant citations, theorem statements, definitions, etc.).

Theory Project 1: Policy Gradient

Papers: Agarwal et al. [1], Mei et al. [20]

Theory Project 2:Online Learning in Bandits

Papers: Lattimore and Szepesvári [17][Chapters 6,7,11]

Theory Project 3:Online Learning in Adversarial MDP

Papers: Even-Dar et al. [7], Zimin and Neu [35], Jin et al. [13]

Theory Project 4:Reinforcement Learning with Linear Function Approximation

Papers: Jin et al. [12], Cai et al. [2], Wang et al. [32]

Theory Project 5:Offline Reinforcement Learning

Papers: Gabbianelli et al. [9], Hong and Tewari [11]

Theory Project 6:Offline Imitation Learning

Papers: Agarwal et al. [1][Chapter 15.1, 15.2, 15.3], Zeng et al. [34], Rajaraman et al. [22]

Theory Project 7:Query-Based Imitation Learning

Papers: Ross and Bagnell [26], Ross et al. [25], Rajaraman et al. [23]

Theory Project 8:Dataset-based Imitation Learning

Papers: Shani et al. [28], Xu et al. [33], Viano et al. [31]

Theory Project 9:Identifiability in Inverse Reinforcement Learning

Papers: Cao et al. [3], Kim et al. [15], Rolland et al. [24]

Theory Project 10:Multi-Agent Reinforcement Learning

Papers: Liu et al. [19], Jin et al. [14], Leonardos et al. [18]

Theory Project 11:Reinforcement Learning in Constrained MDP

Papers: Efroni et al. [6], Ding et al. [5], Vaswani et al. [30]

Theory Project 12:Robust Reinforcement Learning

Papers: Derman et al. [4], Tessler et al. [29], Kumar et al. [16]

2 Applied Project 1: Deep RL

For the *practical track* you are required to re-implement four famous RL algorithms among the ones listed below. You should also test them on some of the following `OpenAI-Gym` environments: `Cartpole`, `MountainCar`, `MountainCarContinuous`, `Acrobot` and `Pendulum`.

In your report of 6–8 pages you are required to compare the **following 4 algorithms of your choice** based on of your empirical observation. This means providing appropriate plots and score statistics of your algorithms based on fair comparison between them.

Possible algorithms:

- DQN by Mnih et al. [21]
- PPO by Schulman et al. [27]
- SAC by Haarnoja et al. [10]
- TD3 by Fujimoto et al. [8]

You can also add a qualitative discussion about the two algorithms building around the following questions:

- Which algorithm is more computationally expensive per iteration ?
- Which algorithm store the policy more compactly ?
- Which one scales better for continuous actions ?
- Which algorithm makes efficient use of off-policy data ?

Finally, view the report as diary in which you can keep track of the observations made during the implementation process. We are interested in knowing which small details in the implementation you found are crucial to make the algorithm work in practice! For example, if you had a bug that took you you a long time fix, write it down. If you found that the algorithm's performance is very sensitive to certain hyperparameter tuning, write it down. Take also note if you find out that an hyperparameter affects the performance only minimally, and think about possible reasons. Corroborate your claims by showing plots that compare the algorithms when run for the different hyperparameters (i.e., do not only report the final, good hyperparameter choices that made it work eventually).

<u>Important</u>: *Each plot you present should report an algorithm's performance averaged across at least 3 seeds.*

3 Applied Project 2: RLHF

For the *practical track* you are required to test different RLHF algorithms on some of the following `OpenAI-Gym` environments: `Cartpole`, `MountainCar`, `MountainCarContinuous`, `Acrobot` and `Pendulum`.
You have to proceeded as follow.

- **Trajectory generation:** Use an RL algorithm of your choice to train a good policy π_1 that achieve quite reliably the highest reward in the environment. Also save a checkpoint during training for a policy π_2 which achieves a reward more or less equal to half the maximum attainable total reward. At this point, generate the preference dataset generating K pairs of preferred and rejected trajectory. To generate each pair, generate one trajectory with π_1 (denoted by τ_1) and one with π_2 (denoted by τ_2). Letting $R(\tau)$ being the total reward of the trajectory τ, let τ_1 be the preferred trajectory with probability
$$\frac{\exp\left(R(\tau_1)\right)}{\exp\left(R(\tau_1)\right) + \exp\left(R(\tau_2)\right)}$$
- **Run RLHF algorithms** Compare DPO and PPO-RHLF for different sizes of the preference dataset and for at least two of the aforementioned environments

Important: Each plot you present should report an algorithm's performance averaged across at least 3 seeds.

4 Applied Project 3: Imitation Learning

For the *practical track* you are required to test different imitation learning algorithms on some of the following `OpenAI-Gym` environments: `Cartpole`, `MountainCar`, `MountainCarContinuous`, `Acrobot` and `Pendulum`.
You have to proceeded as follow.

- **Trajectory generation:** Use an RL algorithm of your choice to train a good policy π_1. At this point, generate the expert dataset generating K trajectories rolling out π_1.
- **Run imitation learning algorithms** Compare IQ-Learn, one algorithm among (CSIL, HyPE, f-IRl, ML-IRL) and its SOAR enhanced version for different sizes of the expert dataset and for at least two of the aforementioned environments.

Important: Each plot you present should report an algorithm's performance averaged across at least 3 seeds.

5 Interdisciplinary projects

Some other labs at EPFL offer *interdisciplinary projects* that target the application of RL algorithms to other scientific problems. We collected a list of labs that are open towards supervising you on such a project.

If you choose this option you can reach out to **one** of the labs in this list and express your interest in the project. You are expected to hand in a report of 6–8 pages. It will be evaluated in collaboration with the external lab that hosted your project.

References

[1] Alekh Agarwal, Sham M. Kakade, Jason D. Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift, 2020.

[2] Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning (ICML)*, 2020.

[3] Haoyang Cao, Samuel Cohen, and Lukasz Szpruch. Identifiability in inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 34:12362–12373, 2021.

[4] Esther Derman, Yevgeniy Men, Matthieu Geist, and Shie Mannor. Twice regularized markov decision processes: The equivalence between robustness and regularization. *arXiv preprint arXiv:2303.06654*, 2023.

[5] Dongsheng Ding, Kaiqing Zhang, Tamer Basar, and Mihailo Jovanovic. Natural policy gradient primal-dual method for constrained markov decision processes. *Advances in Neural Information Processing Systems*, 33:8378–8390, 2020.

[6] Yonathan Efroni, Shie Mannor, and Matteo Pirotta. Exploration-exploitation in constrained mdps. *arXiv preprint arXiv:2003.02189*, 2020.

[7] Eyal Even-Dar, Sham M Kakade, and Yishay Mansour. Online markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009.

[8] Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods, 2018.

[9] Germano Gabbianelli, Gergely Neu, Nneka Okolo, and Matteo Papini. Offline primal-dual reinforcement learning for linear mdps. *arXiv preprint arXiv:2305.12944*, 2023.

[10] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning (ICML)*, 2018.

[11] Kihyuk Hong and Ambuj Tewari. A primal-dual algorithm for offline constrained reinforcement learning with low-rank mdps. *arXiv preprint arXiv:2402.04493*, 2024.

[12] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I. Jordan. Provably efficient reinforcement learning with linear function approximation, 2019.

[13] Chi Jin, Tiancheng Jin, Haipeng Luo, Suvrit Sra, and Tiancheng Yu. Learning adversarial markov decision processes with bandit feedback and unknown transition. In *International Conference on Machine Learning*, pages 4860–4869. PMLR, 2020.

[14] Chi Jin, Qinghua Liu, Yuanhao Wang, and Tiancheng Yu. V-learning–a simple, efficient, decentralized algorithm for multiagent rl. *arXiv preprint arXiv:2110.14555*, 2021.

[15] Kuno Kim, Shivam Garg, Kirankumar Shiragur, and Stefano Ermon. Reward identification in inverse reinforcement learning. In *International Conference on Machine Learning*, pages 5496–5505. PMLR, 2021.

[16] Navdeep Kumar, Esther Derman, Matthieu Geist, Kfir Y Levy, and Shie Mannor. Policy gradient for rectangular robust markov decision processes. *Advances in Neural Information Processing Systems*, 36, 2024.

[17] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[18] Stefanos Leonardos, Will Overman, Ioannis Panageas, and Georgios Piliouras. Global convergence of multi-agent policy gradient in markov potential games. *arXiv preprint arXiv:2106.01969*, 2021.

[19] Qinghua Liu, Tiancheng Yu, Yu Bai, and Chi Jin. A sharp analysis of model-based reinforcement learning with self-play. In *International Conference on Machine Learning*, pages 7001–7010. PMLR, 2021.

[20] Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International conference on machine learning*, pages 6820–6829. PMLR, 2020.

[21] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning, 2013.

[22] Nived Rajaraman, Lin Yang, Jiantao Jiao, and Kannan Ramchandran. Toward the fundamental limits of imitation learning. *Advances in Neural Information Processing Systems*, 33:2914–2924, 2020.

[23] Nived Rajaraman, Yanjun Han, Lin Yang, Jingbo Liu, Jiantao Jiao, and Kannan Ramchandran. On the value of interaction and function approximation in imitation learning. *Advances in Neural Information Processing Systems*, 34:1325–1336, 2021.

[24] Paul Rolland, Luca Viano, Norman Schürhoff, Boris Nikolov, and Volkan Cevher. Identifiability and generalizability from multiple experts in inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 35:550–564, 2022.

[25] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

[26] Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

[27] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.

[28] Lior Shani, Tom Zahavy, and Shie Mannor. Online apprenticeship learning. *arXiv:2102.06924*, 2021.

[29] Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In *International Conference on Machine Learning*, pages 6215–6224. PMLR, 2019.

[30] Sharan Vaswani, Lin Yang, and Csaba Szepesvári. Near-optimal sample complexity bounds for constrained mdps. *Advances in Neural Information Processing Systems*, 35:3110–3122, 2022.

[31] Luca Viano, Stratis Skoulakis, and Volkan Cevher. Better imitation learning in discounted linear mdp. 2023.

[32] Tianhao Wang, Dongruo Zhou, and Quanquan Gu. Provably efficient reinforcement learning with linear function approximation under adaptivity constraints. *Advances in Neural Information Processing Systems*, 34:13524–13536, 2021.

[33] Tian Xu, Ziniu Li, Yang Yu, and Zhi-Quan Luo. Provably efficient adversarial imitation learning with unknown transitions. *arXiv preprint arXiv:2306.06563*, 2023.

[34] Siliang Zeng, Chenliang Li, Alfredo Garcia, and Mingyi Hong. When demonstrations meet generative world models: A maximum likelihood framework for offline inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[35] Alexander Zimin and Gergely Neu. Online learning in episodic Markovian decision processes by relative entropy policy search. *Advances in neural information processing systems (NeurIPS)*, 2013.