ECE239AS: Project Guidelines

Instructor: Prof. Lin Yang

TAs: Zhengqi Wu, Zeyu Wang, Joris Kenanian (MSOL)

April 6, 2021

1 Overview

Doing a project gives us the chance to explore RL in more depth. In this document, we suggest a list of topics and papers on which you can base your project. You are welcome to come up with novel research ideas but we do not require that for a full credit grade. Please keep in mind that the Spring quarter is too short for a big complete project. There might be cases that your project/ideas do not work out. In such cases, we accept a careful illustration (using theoretical proofs and/or experimental results, plus a discussion) of why your ideas do not work. However we do not accept reasons like "the coding is not enough", "there is not much time to work on it".
While open to novel ideas, we also provide some default projects. Students who choose to do the default projects can just indicate they are completing the default project/assignment in their proposal.

2 Evaluation and Grade Breakdown

For the project, we expect three phases. First, students design the project and submit a proposal. Then, after a few weeks (see the precise timeline below), we require students to submit a milestone report to report their progress. Finally, students need to prepare a poster, a 2-minute video presentation, and a final report. We will select and highlight good presentations.

Grade Scheme for Course Projects

- Initial Project Proposal (5%), Due on May 3rd.
- Milestone Report (10%), Due on May 31st.
- Presentation (20%), Due on Jun 6.

- Final Report (4 pages, NeurIPS template, 65%), Due on Jun 10.

More detailed requirements are listed as follows.

Proposal The project proposal should be about 1-2 page long and typed in LaTeX(using NeurIPs template), and should include the names of the project team members. We encourage students to seek feedback and help from either TAs, instructors, or someone outside the class. If this is the case, the proposal shall include a project mentor as well.

We will follow a similar guideline from project proposal of Prof. Emma Brunskill's RL course. The proposal should contain an overview of the proposed project and answer the following questions:

- What is the problem that you will be investigating? Why is it interesting?
- If relevant, what data, simulator or real world RL domain will you be looking at? If you are collecting new datasets, how do you plan to collect them?
- What method, algorithm or theoretical analysis are you proposing? If there are existing implementations, will you use them and how ? How do you plan to improve or modify such implementations ? If you are addressing a theoretical question, how do you plan to make progress ?
- What literature have you already surveyed or will be examining to provide context and background?
- How will you evaluate your results? Qualitatively, what kind of results do you expect (e.g. plots or figures)? Quantitatively, what kind of analysis will you use to evaluate and/or compare your results (e.g. what performance metrics or statistical tests)?

Milestone Report We plan to use the Milestone Report to check the progress of projects. The format of the milestone is a two-page long (NeurIPs template) extended abstract. It should contain two sections: "introduction" and "preliminary results". "Introduction" introduces the project and explains "why it is important", "what are the methods you are using", etc. "Preliminary results" shall demonstrate how things are going on. Any figures, working code, pseudo-code, theorem statements should suffice.

Presentation Students should first prepare a poster (no need for printing, just a PDF version is fine) and prepare a 2-minute video recording to present the poster. We will post the presentation online, so that you can look at other groups' presentations.

Final Report The final report should be between 4 and 8 pages using the NeurIPS template. It should follow a similar guideline of writing a NeurIPS paper. We plan to post all the final reports online. If you do not want your report to be published, please indicate so when you are submitting the report. You should include a brief statement on the contributions of different members of the team in the report. Team members should receive the same grades but we reserve the right to differentiate in egregious cases.

Submission Instructions will be communicated to you later.

3 Default Projects

You can also get project ideas from `http://web.stanford.edu/class/cs234/project.html`. We are adding a list of seminal papers in the field of reinforcement learning. A tentative project would be a paper survey and summary of these papers (a "survey and summary" project). If you choose to do a "survey and summary" project, you need to complete the following requirements:

- Describe clearly what the problem is and why it is important;
- Give a brief survey of the history of the problem of interests; describe results of key papers and their methods;
- Give a full description of the methods in the paper you have selected;
- Explain why these methods work. If it is a theory paper, present a proof sketch of the main results, i.e., main theorem and main lemmas. If it is an empirical paper, describe the intuition of the methods and discuss why it works and why they provide better results than previous methods.
- Provide pseudo-code and explanations of the algorithms. If the paper is an empirical study, implement the algorithms and test on at least one environment (e.g. Open-AI gym). Compare the results with existing algorithms on the same environment. Explain the results.

Note that for theory papers, you need to give **proof sketch** of key theorems and lemmas. For empirical/methodology papers, you need to **implement the algorithms** and test them on simple environments. You can also propose additional papers to survey and summary. However, you need to send the papers to the instructor and TA for approval.

Methodology/Empirical Papers

1. Deep Reinforcement Learning with Double Q-learning, Hasselt et al 2015
2. Proximal Policy Optimization Algorithms, Schulman et al, 2017
3. VIME: Variational Information Maximizing Exploration, Houthooft et al, 2016
4. Progressive Neural Networks, Rusu et al, 2016
5. Data-Efficient Hierarchical Reinforcement Learning, Nachum et al, 2018

6. Generative Adversarial Imitation Learning, Ho and Ermon, 2016
7. Sample-Efficient Reinforcement Learning with Stochastic Ensemble Value Expansion, Buckman et al, 2018
8. DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills, Peng et al, 2018
9. MOPO: Model-based Offline Policy Optimization Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, Tengyu Ma (2020)

Theory Papers

1. Finite-time bounds for fitted value iteration, R Munos, C Szepesvári (2008)
2. Near-optimal regret bounds for reinforcement learning, T. Jaksch, R, Ortner, P. Auer, 2015
3. Reinforcement learning in finite MDPs: PAC analysis, AL Strehl, L. Li, ML Littman, (2009)
4. Is q-learning provably efficient? Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, Michael I. Jordan, (2018)
5. Optimality and approximation with policy gradient methods in markov decision processes, A Agarwal, SM Kakade, JD Lee, G Mahajan, (2019)
6. Is a Good Representation Sufficient for Sample Efficient Reinforcement Learning? SS Du, R Wang, M Wang, LF Yang, (2020)