Notes, Ideas, and Advice on Your Final Project

Should not involve new issues. Rather, should further develop an issue explored in the course, or the combination or intersection of two issues explored individually in the course.

Should not involve a lot of writing. Just enough to mention the extant issues and what new is being done with them.

Should not involve a new domain that does not fit neatly into the MDP framework. Should not involve a new domain that involves extensive explanation.

Should not involve multiple novelties. Just one new aspect please.

The real rule is that you want your project to have some meaning, to answer some question and not just bring up new ones without significant progress having been made. You want to close off some questions.

A good project says, here is an interesting issue that has come up, and here is a straightforward experiment or two that could be used to possibly resolve it. Here, Ive done that experiment, here is what happened, this is what i conclude from it with regard to the interesting issue. In doing such a project you should learn (or at least get some experience with) how to frame a scientific question, how to reduce it to an experiment, and how to assess your results, extracting the maximum of meaning from them without going beyond what has actually been shown.

Each one of the experiments in the book are examples of good projects. They are all simple, clear, and unambitious, yet they are instructive; try to make you project like these. In fact, a good project could be made based on most of the examples in the book. First replicate the given results, then extend them in some instructive way, perhaps by using more algorithms, or a variation on the task, or both.

A bad project says, what about this? This was not covered in the book! I will make a new project about it. But then the project will not build on what you have learned in any way in the course. I dont want to see this.

A bad project says, here is a new domain or problem. What if you wanted to solve it in this sort of way? How could that be fit into the RL/MDP framework, even if it does not look very suitable? Then the project will be all about the domain and again not really build on what you learned in the course.

A good project need not be unique to an individual. It should address something of recognizable, arguable interest to many, to a generic reader. It should be of interest such that we care about the results, that we want to get them right and we are genuinely unsure in advance of how they will

turn out. A good project can be explored by multiple teams of students. This, in fact, is my ideal—if we were to have multiple students are student teams working on the same project. Do not go to lengths to ensure that your project is specific to you!

Some possible good projects have been suggested as we went through the course:

1. Non-stationary bandits. See description in the dropbox.

2. I think there are things to be done with blackjack and the off-policy MC methods using importance sampling. To begin with, what if we did the full control problem (rather than just prediction for one state as in example 5.4) with ordinary and weighted importance sampling. The target policy would be the optimal policy and the behavior policy might be the equi-probable random policy (hit and stick 50-50). There will be a little challenge in figuring out how to report the results, as they may be qualitatively different in different states.

3. Sections 5.8 and 5.9 discuss two extensions to importance sampling that attempt to take advantage of the special structure of returns to reduce variance. However, no example is given in which these techniques speed learning. Blackjack is not good for this because the potential improvement only arise in the presence of discounting (5.8) or rewards along the way (5.9). So, a new simple problem is needed to show/test these advantages. Devise such a problem and show one of the advantages. Concentrate on one or the other of the two issues until it is fully understood. Ideally you will make me an example that i can include in the second edition before it is finalized in january.

4. Sums of TD errors. See description in the dropbox.

5. Sections 7.4 and 7.5 discuss the tree-backup and Q(sigma) algorithms, and sing their praises, but again give no illustrations. Remedy this with an example in your project.

6. The idea of this project is to assess whether the extra term in the update of the parameter vector in true online TD(λ) actually helps or hurts performance. The effect seems to be small, so a careful experiment is needed, with enough runs to get statistically significant results. You will have to select an appropriate test problem, and ideally, more than one. It is probably not necessary to vary λ.