

# Probabilistic Reasoning in AI - Project

This project refers mainly to reinforcement learning and is aimed to give you some practical experience with RL algorithms. You may work by yourself or in a team of 2 students. If you want to propose an alternative project, you may do so by e-mailing Doina with a proposal. note that the project is worth 15% of your grade.

You will be writing a reinforcement learning agent for the game of Yahtzee. Code for this game, as well as for some simple benchmark agents is available at:

<http://rlai.cs.ualberta.ca/RLR/index.html>

You will have to visit the environment shelf and the agent shelf and download the yahtzee archive.

Note that the given code is in C++. However, you do not need to write your own code in C++ in order to use this environment. You need to write code which can communicate with this environment (and you should be able to do this easily in Java or C; Matlab would be tricky).

You may implement any reinforcement learning method of your choice. Detailed information on different methods that we discussed briefly in class is available at:

<http://www.cs.ualberta.ca/~sutton/book/the-book.html>

You may use existing code, as long as you acknowledge your sources. Also, if you use existing methods that have been published in research papers, make sure you reference your source appropriately.

What you should turn in:

1. A report (pdf format) describing:

- The method you decided to use, in enough detail that one could reproduce your code; explain your reasoning for the different design choices (e.g. how to represent the state)
- Any choices of parameters that you are using, and how you decided on those choices
- Experiments showing evidence that your method is learning (e.g. learning curves, where you show performance improving over time)
- A comparison with the benchmark agents (note that numbers summarizing their comparison are provided in the pdf document describing the domain, so you do not have to run these agents).

2. A tar archive with your code, including a README file explaining what is in each file, and any known bugs.

Please e-mail both the report and the code to Doina by Monday April 28, 10am.

Grading:

- Quality of the approach - 20%
- Correctness and evidence of learning - 40%
- Comparison with benchmark agents - 10%
- Writing quality of the report - 20%
- Code organization and legibility - 10%