**ST590 Project Abstract**

Robert Sawyer and Shawn Harris

In reinforcement learning, an agent interacts within an environment, periodically receiving rewards. The agent knows where it currently is in the environment, the state, and knows which actions it is allowed to take from that state. The agent aims to learn an optimal policy, or mapping from states to actions, to maximize the discounted future rewards received from the environment. Reinforcement Learning methods aim to convert the agent’s experiences with the environment into an optimal policy [1], which can then be followed by the agent to maximize its rewards in the environment. When an agent follows its optimal policy, the agent is said to be *exploiting* the environment. Conversely, an agent is *exploring* the environment when it takes actions it does not believe to be optimal in order to learn more about the environment. The agent must balance this tradeoff between *exploration* and *exploitation* to maximize its overall reward while ensuring it is not missing better action trajectories through the environment.

The reinforcement learning problem is often modeled as a Markov Decision Process, which is a model defined by states, actions, transition probabilities, start state probabilities, and a reward function. This is a form of model-based reinforcement learning, where an explicit model of the environment is maintained by the agent, estimating the transition probabilities and reward function through its interactions with the environment. In a frequentist setting, these quantities, such as transition probabilities, are estimated through their maximum likelihood estimate, such as the ratio of times transition into the next state by the total times transitioned. In a Bayesian setting, these quantities, such as transition probabilities, can be treated as random variables with probability distributions, such as a Beta-Binomial posterior or Dirichlet-Multinomial posterior when there are more than two states to transition to. The advantages of a Bayesian treatment are twofold: (1) it allows quantification of the uncertainty in the model parameters, which are typically assumed to be known in RL problems (such as Dynamic Programming methods) and (2) it provides an elegant approach to the *exploration-exploitation* problem [2]. The solution to the exploration/exploitation problem involves sampling the transition probabilities from the posterior distribution in order to calculate the best action under the expected reward from the sampled transition probabilities.

In this work, we compare several popular frequentist *exploration-exploitation* action selection strategies with the Bayesian approach (known as Thompson Sampling [2]). We perform this comparison on a reinforcement learning problem known as the K-Armed Bandit problem, in which the agent must learn which of K slot machines (bandits) gives a binary reward at the highest rate. We compare several reinforcement learning metrics such as cumulative reward, regret (difference from optimal policy), convergence to the optimal policy, and computational efficiency. We also propose an addition to the Thompson Sampling algorithm to determine when the Bayesian agent should switch to an *exploitation* mindset utilizing the posterior distributions maintained by the agent, and give an empirical comparison of this agent with the Thompson Sampling agent.

The simulation environment will be performed in Python, in code developed by Rob and Shawn. The results will be visualized in several ways to completely describe the differences between the frequentist and Bayesian approaches to this problem.

**References**

[1] Sutton, Richard S., and Andrew G. Barto. *Reinforcement learning: An introduction*. Vol. 1. No. 1. Cambridge: MIT press, 1998.

[2] Ghavamzadeh, Mohammad, et al. "Bayesian reinforcement learning: A survey." *Foundations and Trends® in Machine Learning* 8.5-6 (2015): 359-483.