Report of the project 1

Introduction

In Richard Sutton's 1988 paper "Learning to Predict by the Methods of Temporal Differences", he proposed two computational experiments of 5-steps random walk problem to demonstrate the efficacy and advantages of TD(lambda) algorithm. The main purpose of this report is to describe and discuss how I implemented these two experiments, and replicated the results shown in Figure 3, 4 and 5 in the original paper.

The 5-steps random walk problem is a typical Markov process. It has two boundary states (A and G) and five intermediate states (B, C, D, E and F) (see Figure 1 as below):

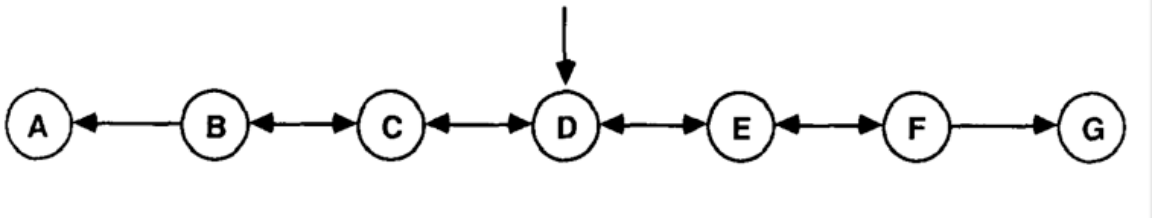

Figure 1. The diagram of 5-steps random walk problem.

The agent always starts at state D and could randomly move to left or right with equal probability (0.5) until the boundary state is reached. Once the agent arrives at the boundary state, the walk is terminated and the agent receives the reward, z = 1 for ending at state G and z = 0 for ending at state A. The theoretical probability of a walk ending in state G from non-terminal state is: T = {1/6, 2/6, 3/6, 4/6, 5/6}, for states {B, C, D, E, F}, respectively. The primary goal of this research is to produce a serials of predictions, each of which is an estimation of the final reward z, over different random-walk episodes using the TD(lambda) method.

Methods

For computational experimental one, I generated 100 training sets and each set includes 10 episodes of random walk on the basis of binomial distribution of each step. These 100 training sets then will be employed to train seven TD learning procedures with lambda = {0.0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0}, respectively. According to Sutton's 1988 paper, the learning process could be transformed to update the weighting factors of each state of the random walk at time t. I followed the paper's rules concerning with updating weighting factors in experimental one, which means the weighting factor changes were summed up over episodes and the weighting factors will not be updated until the whole training set was presented to a give learning procedure. Each training set will present to each learning procedure repeatedly until the weighting factors were converged. Here I defined the convergence of weighting factors as the weighting factor changes less than epsilon where epsilon = 0.0001 in my codes. The performance of the

learning procedure was evaluated by the RMSE between the converged weighting factors and the theoretical probabilities of a random walk ending in state G from different non-terminal states. To accelerate the convergence of weighting factors, after the whole training set was presented to a given learning procedure, I calculated the mean of accumulated weighting factor changes and added it to the weighting factor. The learning rate alpha was set to 0.05.

For computational experimental two, I used the same training set as described in experimental one. Yet the strategy of updating weighting factor was changed and I exactly followed the way of that in Sutton's 1988 paper page 22, under Figure 5.

One tricky part of this 5-steps random walk problem is the reward of random walk between non-terminal states. Sutton's paper did not mention that and in my implementation, I assume such reward is 0.

Discussions

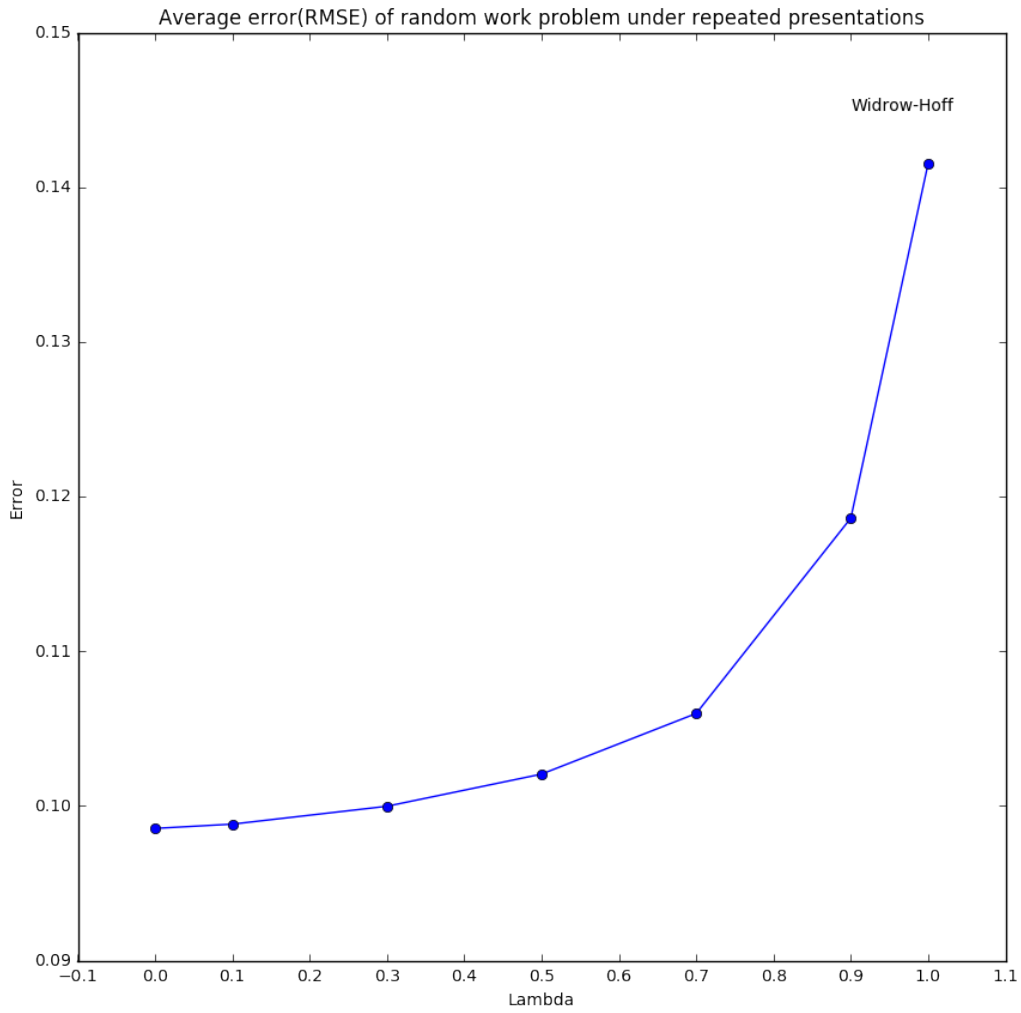

Figure 2. Average RMSE under repeated presentations of training set

Figure 2 is my attempt to replicate the Figure 3 in Sutton's 1988 paper. We could see the trend of these two figures are similar. The TD(0) method has the lowest average error between the converged weighting factors and the theoretical probabilities. And TD(1) method has the highest average RMSE. The differences between these two figures is the average RMSEs of Sutton's Figure 3 at each lambda were higher than mine. Since Sutton's paper did not report the alpha and epsilon used in Figure 3, the parameters I choose (alpha = 0.05 and epsilon = 0.0001) might be not as same as the ones used in the paper.

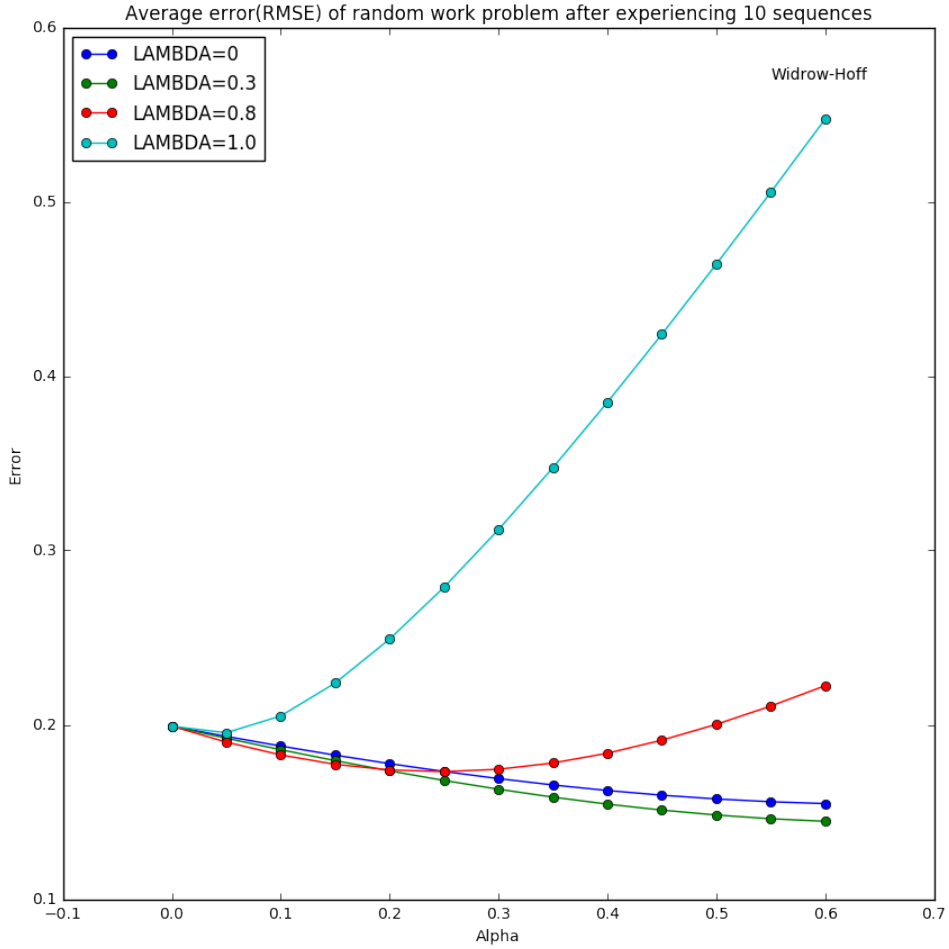

Figure 3. Average RMSE after experiencing 10 sequences.

Figure 3 is my attempt to replicate the Figure 4 in Sutton's 1988 paper. The difference between these two figures was the curve of lambda = 0. In the Figure 4 of Sutton's 1988 paper, the lambda = 0 curve has higher average RMSE value than lambda = 0.8 curve at alpha = 0.6. One possible explanation of this phenomenon is the training data sets were randomly generated, and caused different outcomes even using the same learning procedures.

Figure 4 is my attempt to replicate the Figure 5 in Sutton's 1988 paper. The average RMSEs were calculated at the best alpha for each lambda. And these two figures have

the same pattern of the results, especially my figure shows the TD(0) did not have the lowest average RMSE, which was mentioned in the paper. Also my figure shows the best lambda value is around 0.4, not 0.3 as reported in the paper. Again, this mismatch may be due to the difference between randomly generated training sets.

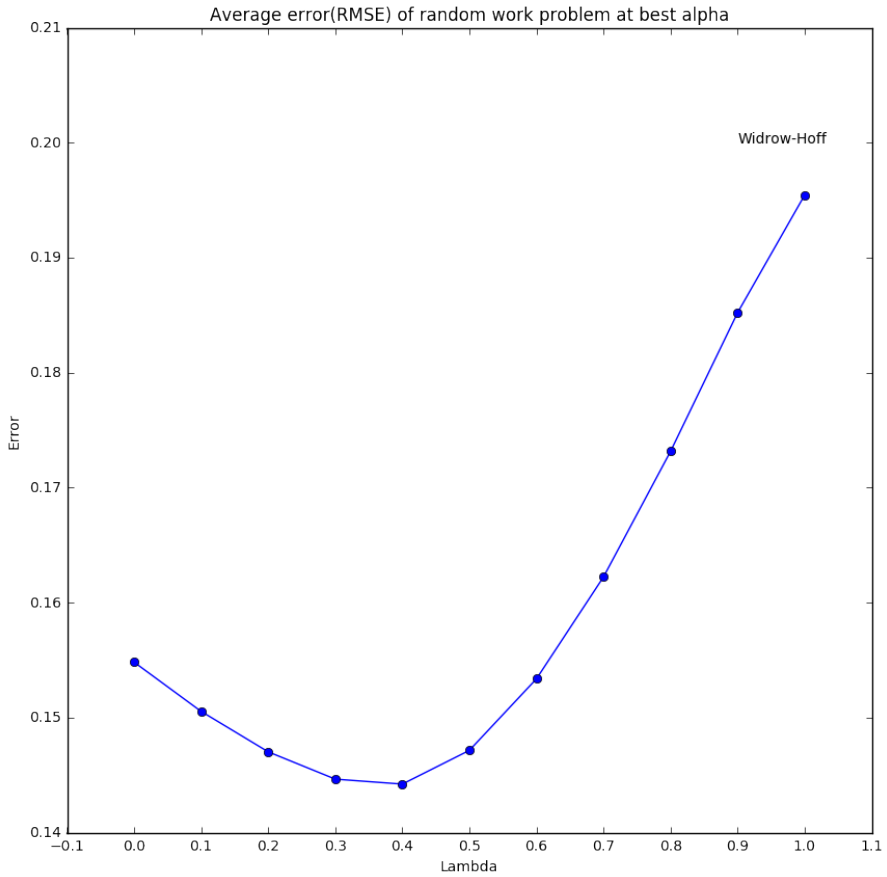

Figure 4. Average RMSE at best alpha.

Summary

The replications of Sutton's 1988 paper results found in Figure 3, 4 and 5 were remarkable on the basis of my implementations. The differences between my outcomes and that of paper are mainly due to the unreported parameters and randomly generated training sets.

Reference

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. Machine learning, 3(1), 9-44.