Desperately Seeking Sutton- RL&DL Project 1

Mehmet Oguz Kazkayasi
OMSCS Student
Georgia Institute of Technology
Atlanta, GEORGIA
mkazkayasi3@gatech.edu

Git Hash value:

***Abstract*—This document is a replication of Richard S. Sutton's "Learning to Predict by the Methods of Temporal Learning" paper, dated 1988. The algorithms described in the paper are implemented and existing figures are recreated with some numerical variation. The purpose of this document is to report the causes of differences between Sutton's paper and the replication.**

***Keywords*—TD Learning, Richard S. Sutton. Incremental Learning, Prediction**

I. INTRODUCTION TO TD LEARNING

In his paper, Richard Sutton introduces Temporal Difference Learning procedure for prediction, uses it to predict the outcome of Random Walk Example. Also, he claims that TD Learning methods uses the limited data more efficient than supervised learning methods.

Whereas a supervise learning method uses the pairs of states and outcomes to make prediction, TD Learning method, uses the following states as well as the final outcome of a state. This property lets TD Learning method to learn more using one episode. If the training data is unlimited, both TD Leaning and supervised learning methods would produce the same results. But, when there is limited data, it is suggested that TD Learning methods make better predictions outside of the training data.

TD methods are sensitive to the differences of predictions of successive states, as its name 'temporal difference' defines. This enables a faster learning than using overall error between states and the final outcome.

Moreover, TD methods does not necessarily wait for the end of an episode to make the correction on the prediction. The correction can be made after seeing a state. But, there are variations that will make the accumulated correction in the end of an episode or even after a collection of episodes. Small enough learning rates makes the convergence possible for all cases.

II. RANDOM WALK EXAMPLE

Even though TD methods are used for much complex domains before the paper is written, Sutton shows the effectiveness of it using a simple Bounded Random Walk example.

The Random Walk example consists of a line and the agent take random steps towards its right or left. It is bounded, so that it ends whenever the agent, which started at center state D ends up at state A or state G. A state sequence is generated using random variables

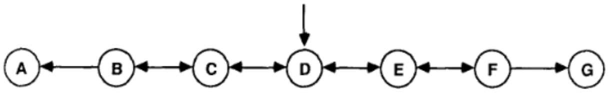

Figure 1. A generator of bounded random walk

All episodes start at state D and there is 50% chance of moving right or left on states B, C, D, E, and F. Whenever state A or G is reached, the episode ends.

In Sutton's paper, it is stated that the only requirement that makes TD Learning better than supervised learning methods is that the system, of which outcome will be predicted, needs to be a dynamic one. To prove this statement, the experiments must show that TD Learning will make better predictions using the same training data.

A random walk has two possible outcomes. In case it ends at state A, its outcome is 0. If it ends at state G, the outcome is 1. TD Learning and supervised learning methods will make prediction on nonterminal states' probability of ending at state G.

As it can be calculated analytically, the outcomes of the nonterminal states are 1/6, 1/3, 1/2, 2/3, 5/6 for state B, C, D, E, and F, respectively

III. COMPUTATIONAL EXPERIMENTS

In Sutton's paper, two different computational experiments are performed using Bounded Random Walk sequences that is described above. The training data that is used by all learning procedures consists of 100 training set, each consisting of 10 sequences of random walk states.

A. Average Error on the Random Walk Problem Under Repeated Presentations

In this first experiment, the correction of states (Δw) is accumulated until end of a batch of sequences, which is a training set, before updating the weights. Each one of the training set is presented to the learner until there is no more for the learner method to improve using that training set. It is stated that, using a small enough alpha value the weights will converge on the same values after repeated presentation no matter what initial values are.

In Sutton's paper, the initial values are not described. Because the weights will converge at the same values, I supposed this missing parameter won't make much difference. So that I have tried different initial values for nonterminal states such as 0, 0.5, and 1.

Interestingly, for different alpha values and epsilon values, initial values have made a significant difference on RMSE values. I think the main reason is the epsilon value that can't be small enough for time efficiency. If small enough alpha and epsilon values are used in the experiment, different initial weights would not make a difference for the prediction using repeated presentation method.

On the other hand, using same initial values (0.5 for nonterminal states), only changing alpha value made significant difference as well. As far as my computational experiences shows, the smaller alpha value is the better predictions.

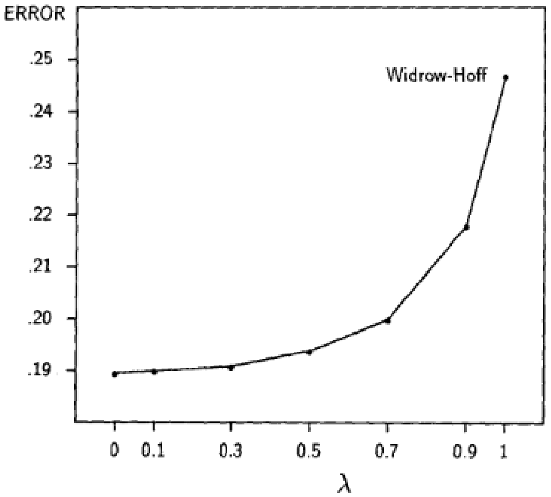

Figure 2: Sutton's Repeated Presentation Figure

Sutton has demonstrated his implementation of the first experience with the figure above. In my replications, the RMSE values are much smaller, depending on the learning rate used.

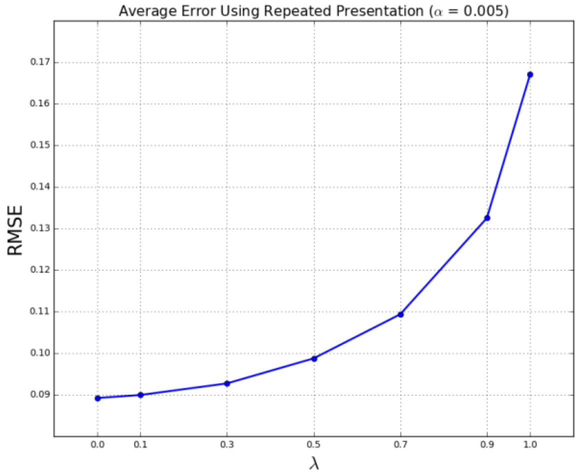

Figure 3: Repeated Presentation Figure

As it is seen in the graphs, my implementation of repeated presentation algorithm has almost half RMSE value of Sutton's implementation for low lambda values. There are a few reasons that might cause this difference.

First one is the randomness. My experience used a completely different dataset than what Sutton had used. Even though this sounds like to have a significant effect, my repeated experiences with different training datasets has shown that it has little to do with the results. This is because we use 100 training set, each consisting of 10 sequences, and this is large enough to remove the effect of the randomness.

The second effect is caused by the learning rate. As I have mentioned above, learning rate has a significant effect on the RMSE values of learning procedures. I have tried 4 different learning rate and the results are significantly different. In Sutton's paper, there is no learning rate specified for this experience, so it is hard to test and say that the difference is just caused by the difference of the learning rate.

The last effect might be caused by the epsilon value, which is the difference between the last weights and the updated weights. It should be smaller than epsilon value for the algorithm to stop. It is also not stated in Sutton's paper. I have used a total sum of 0.01 difference for the epsilon value in my experiments. But, to get better results I believe smaller values can be used.

In his paper, Sutton's purpose is not to implement the TD learning algorithm, along with TD(1), to get the lowest RMSE values. His purpose is to show the difference of efficiency between TD methods and supervised learning methods. Therefore, I believe he used small enough learning rate and epsilon values.

The figure I had after tweaking these values has a very similar pattern with Sutton's figure. So, it shows the difference of efficiency between TD methods and supervised learning methods, as well. And I think this is what matters for the paper.

B. *Average Error on Random Walk Problem After Experiencing Ten Sequences*

As described above, the experiments are done using 100 training sets, consisting of 10 sequences. This experience concerns the effect of the learning rate and it differs with the first one on the following properties.

- Each training set is presented to the algorithm just one, instead of repeated presentation.
- The same data is presented to the learning procedures for several lambda values (0, 0.3, 0.8, 1)
- The weights are updated after each sequence instead of each training set.
- Training set is presented to each learning procedure using different alpha values from 0 to 0.6, with 0.05 intervals.
- All initial weights for nonterminal states are set to 0.5

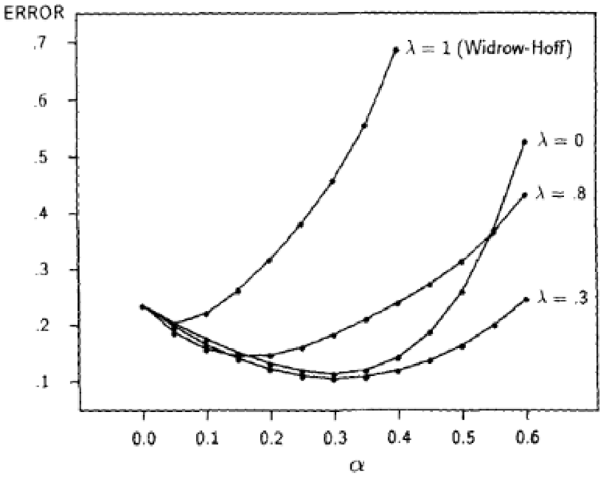

Figure 4: Sutton's Average Error on Random Walk Experience

What Sutton is demonstrating in this graph is that the intermediate values are doing a better job than supervised learning methods. It can be seen that lambda value of 0.3 is giving the lowest RMSE values.

This figure above was particularly difficult for me to recreate. The reason is that all the parameters are described in the paper. The lambda value and learning rate are already stated in the graph and there is no epsilon value because this is not a repeated presentation procedure. Even though randomness in this experiment makes some difference, it was not significant enough, either. There is not much to tweak to get a different graph than what I was getting. And without any restriction I was getting a graph as shown at Figure 5, which actually kind of shows that supervised learning can't give the best outcome comparing TD methods.

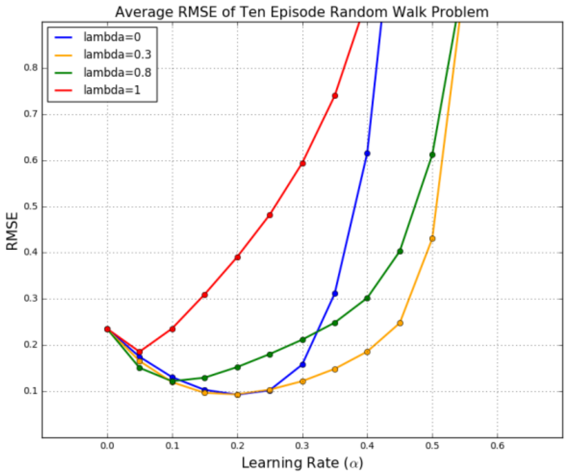

Figure 5: Average RMSE of Ten Episode Random Walk (No Restriction)

After careful inspection of the Sutton's paper, I have read that he mentions a maximum value of episode length for a random walk. So, I have tried to restrict the length of random walks to see if I get any closer. I just ignored the produced random walks that takes more than M steps. After seeing that will take me closer to Sutton's graph, I have tweaked the M value for the best and decided to keep it at 16.

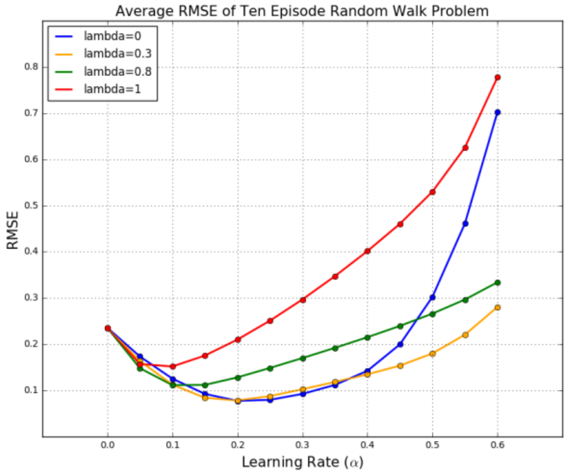

Figure 6: Average RMSE of Ten Episode Random Walk (Restricted Episode Side)

Figure 6 is much close to Sutton's figure. So that, I think Sutton also put a restriction on the episode side. On the other hand, I

C. *Average Error at Best Learning Rate on Random-Walk Problem*

The third figure uses the same learning procedure with the second one. But, it plots the best learning rate values for each lambda value. In the experiment, also in Figure 4, it can be seen that different lambda values brings up the best outcomes at different learning rates.

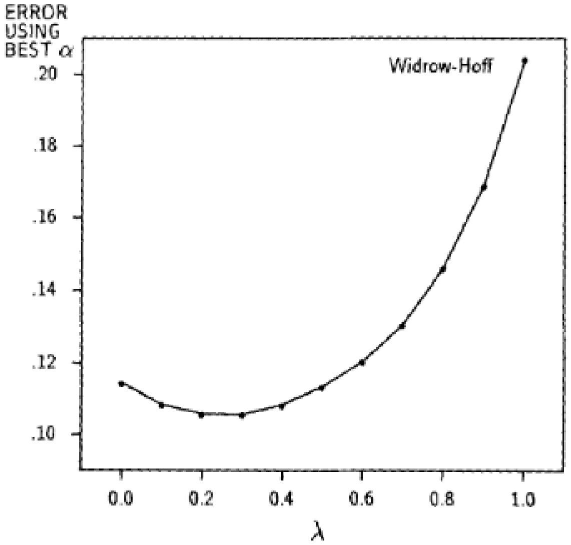

Figure 7: Sutton's Average Error at Best Learning Rate Experience

Sutton shows that similar to repeated presentation method, the best error levels are achieved under lambda value 1.

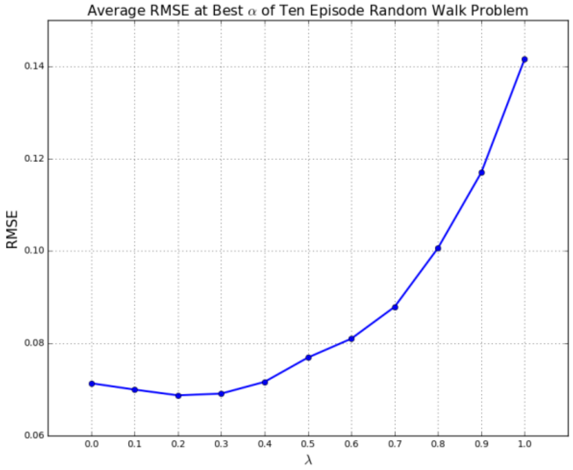

Figure 8: Average Error at Best Learning Rate Experience

In this experiment, I have used 13 different learning rates, from 0 to 0.6, for each lambda value and pick the best among each of them. The results are very similar to Sutton's experience, but my experiment ended up at lower RMSE values.

I have used exhaustive search for the best learning rate for each lambda value and get the best among all.

The small difference at RMSE values between Sutton's and my implementation might be caused by the possibility that Sutton decided the best alpha value on some other training dataset and used it on this one. And, my best learning rates can be overfit to this training set so they won't give the best results for other data.

So, I have chosen the best learning rates for each lambda value and put them in to action at a different dataset. It gave similar results to my implementation above and didn't explain the difference.

I have tried different seeds to check the effect of the randomness and seen that randomness doesn't close the difference, as well.

So, there must be some other answer to higher RMSE values of Sutton's experiment. Other than this small difference, the best value is reached at 0.2 to 0.3 at both graphs. Also, the patterns of the graphs match each other.

References

[1] R. Sutton, "Learning to Predict by the Methods of Temporal Difference", February 1988.