A Replication of "Learning to Predict by the Methods of Temporal Differences" (Sutton, 1988)

Aaron M. Buxbaum
Georgia Institute of Technology
me@aaronbuxbaum.com

Abstract This article replicated the experiments presented in Sutton (1988), which describes a novel method of machine learning via temporal differences. The obtained results were not identical to the original article, but reproducibility was confirmed by comparison of core concepts.

***Index Terms* — incremental learning, machine learning, prediction, replication, temporal difference**

I. Introduction

Sutton's article presents a novel method of prediction for problems with dynamical states. Predictions are given in the form of a vector of weights (w), which updates over time steps (t) via the update rule expressed in Equation 1:

$$w \leftarrow w + \sum_{t=1}^{m} \Delta w_t \tag{1}$$

Δw_t is most fully expressed in Equation 4:

$$\Delta w_t = \alpha(P_{t+1} - P_t) \sum_{k=1}^{t} \lambda^{t-k} \nabla_w P_k \tag{2}$$

where α is the learning rate, P_t is the active prediction value at time t, λ represents the trace decay, and $\nabla_w P_k$ is the vector of partial derivatives of P_t with respect to each component of w. (2) has two major advantages to the prototypical supervised-learning update procedure given in Equation 2: it runs iteratively rather than waiting until the final value is realized; and it allows for significantly lower memory usage, since past values of $\nabla_w P_k$ don't need to be maintained. Sutton's given experiment used these equations to solve for reward probabilities in a bounded random-walk problem, described below.

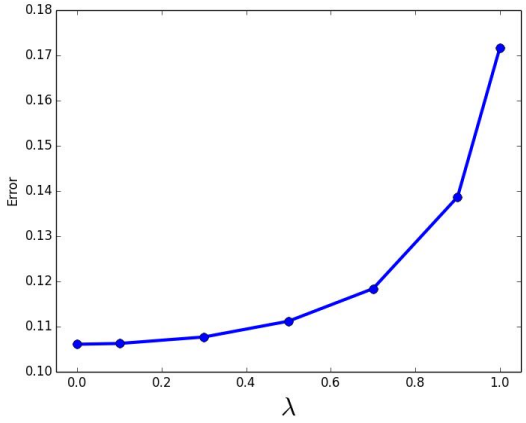
II. Methods

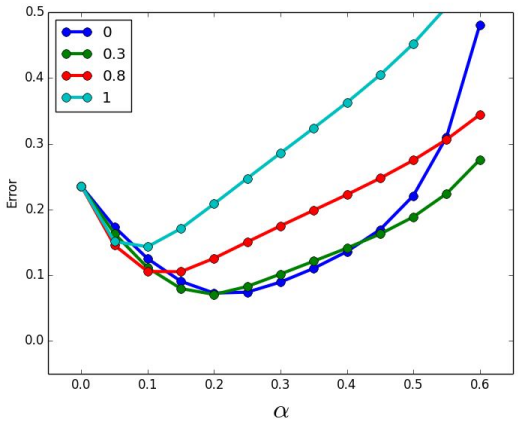
Bounded random-walk is a problem containing a sequence of n linear states with bi-directional edges between each neighboring state ($1 \leftrightarrow 2 \leftrightarrow \ldots \leftrightarrow n$). The states on each end -- that is to say, state 1 and state n -- are absorbing states, where the far state (state n) is the state that we measure probability of access against. Since Figure 2 shows a 7-state problem example, the experiments were assumed to be likewise, though the size of the walk is not technically stated in the article. An example walk sequence is: [4, 3, 4, 5, 6, 7], which ends in the final state of 7. Since state 7 is already in its absorbing state, its probability is by definition 1. Likewise, since state 1 is an absorbing state, but not the "desired" state, its probability is 0. Trivially, the probability of reaching the desired state is $\frac{i-1}{n-1}$ for each state i. In these experiments, the goal was to predict the probabilities for each position of a bounded random walk ending in the desired absorbing

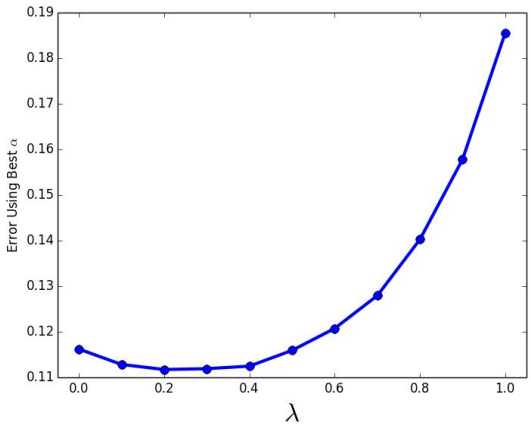
state, when presented with a set of training data.

The experiments described in Section 3.2 were duplicated as closely as possible to Sutton's written descriptions in a Python 2.7 script. 100 sets of 10 sequences of random moves were randomly generated to serve as training data. Two experiments were described:

- Experiment 1 - *Batch Learning*. For any small α, and several λ between 0 and 1, each set of sequences was repeatedly presented, accumulating Δw between sequences and only applying it to w at the end of the set. At the end of each set, Δw was reset to 0. The sets were presented continually until w converges.
- Experiment 2 - *On-Line Learning*. For several α between 0 and 0.6, and several λ values between 0 and 1, each set was presented exactly once, this time immediately applying Δw to w after each sequence. At the end of each sequence, Δw was reset to 0. Instead of repeatedly presenting sets until convergence, each set was presented exactly once.







III. Results

Accuracy was evaluated by calculating the average root-mean-squared error (RMSE) between the actual results and the optimal values for each set. The lower the error, the more accurate the results. Each experiment was performed and then graphed, on the left, corresponding to Sutton's Figures 3-5.

Experiment 1. Average error after repeated set presentations. These results mirror Sutton's Figure 3 nearly perfectly, with one exception: all error values were approximately 40% lower in our implementation.

Experiment 2. Average error after experiencing a set once. These results mostly resemble Sutton's Figure 4. Overall error was again slightly decreased. In addition, λ=1 increased linearly, and therefore more slowly than the original article's low exponential rate of increase.

Experiment 2, Part 2. Average error at best α value after experiencing a set once. This shape very closely matched Sutton's Figure 5. Yet again, error was consistently slightly lower.

Analysis. The difference in data values were within $p \leq 0.05$, and therefore reproducibility was confirmed.

IV. Discussion

As seen in the previous section, the original article is generally reproducible, and concepts are clearly mirrored between the original and reproduction.

The biggest difference in results was the consistently lower general error values across the board: this was likely due to bit-level accuracy. When the original article was written in 1988, the experiments were likely implemented using a 16- or 32-bit system, in contrast to our more accurate 64-bit system. Due to our improved ability to maintain higher precision digits than in the past, our inherent error in numeric storage was lower. In addition, modern-day compilers are slightly more efficient than the optimizers of the late '80s.

The other difference of note was that of Figure 4's $\lambda=1$ line. It's worth noting that as λ increases, it likely will be overfitting to the training data and performing more poorly on test data. One can see this in action in the concave shape of Figure 5: too low of a λ value means that learning isn't happening properly, but too high of a λ value means that it is overfitting. The large-scale differences in the highest level of overfitting, $\lambda=1$ for example, therefore, were likely differences in compiler optimization.

Variance between the randomly generated datasets likely accounted for the remaining slight differences between the graphs.

The largest pitfall in implementation was that of lacking numeric details in the article. No seed for generation of the training dataset was given, which caused a high degree of potential variability in the training dataset, and hence, a high degree of potential difference between the original and replication articles. In addition, the number of states was not explicitly defined, so 7 was assumed, due to it being used in Figure 2. It's quite possible that the number of states used in Sutton's experiment was actually different, which would cause a great degree of variability. While Sutton correctly noted that any sufficiently small alpha value would converge for Experiment 1, he did not define "small", which required some trial-and-error to find; in practice, any number $\leq$ approximately 0.05 seemed to work. Sutton defined convergence as "no longer producing significant changes", which could mean just about any small number. We picked $1 \cdot 10^{-6}$ -- with a convergence number this low, the difference between small values was more or less negligible. It is possible that the original article could have used a lower convergence number, which would have caused slightly different results.