Machine Learning 32, 241–243 (1998)
© 1998 Kluwer Academic Publishers. Manufactured in The Netherlands.

Comments on "Co-Evolution in the Successful Learning of Backgammon Strategy"

GERALD TESAURO tesauro@watson.ibm.com
IBM T.J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598

Editor: Thomas Dietterich and Leslie Pack Kaelbling

Abstract. The results obtained by Pollack and Blair substantially underperform my 1992 TD Learning results. This is shown by directly benchmarking the 1992 TD nets against Pubeval. A plausible hypothesis for this underperformance is that, unlike TD learning, the hillclimbing algorithm fails to capture nonlinear structure inherent in the problem, and despite the presence of hidden units, only obtains a linear approximation to the optimal policy for backgammon. Two lines of evidence supporting this hypothesis are discussed, the first coming from the structure of the Pubeval benchmark program, and the second coming from experiments replicating the Pollack and Blair results.

Keywords: co-evolution, backgammon, temporal difference learning

In "Co-evolution in the Successful Learning of Backgammon Strategy," Pollack and Blair (1998) present a novel and intriguing self-teaching approach to learning a policy for backgammon. In constast to TD learning, which aims to learn a value function estimating expected outcome of a given board position, their approach directly optimizes policy strength. The policy is represented in terms of a discriminant value function that is unrelated to expected outcome and is only used to discriminate between competing legal moves. Random mutations of the policy are generated and evaluated by running a short head-to-head test against the base policy. If the mutant wins a sufficient number of games, the weights are adjusted by a small amount in the direction of the mutation. That this approach works at all can be viewed as a surprising result. One might reasonably have expected beforehand, as Pollack and Blair point out, that as the strength of the net improves, the probability of a random mutation of several thousand parameters improving the performance might quickly become vanishingly small. Furthermore, this approach is able to work despite the absence of a population of learners (there is only one "champion" at any given time), a "crossover" operation to combine successful members of the population (there is only a mutation operation), or a low-dimensional "genome" specifying the network structure (mutation operates directly on the "phenotype" of network weights). All of these factors indicate that this work constitutes a clear contribution to the field of evolutionary neural networks, and is worthy of further study in its own right.

However, the authors' introduction and conclusions omit any mention of the work's intrinsic interest, and instead focus solely on implications for why TD-Gammon worked. The hillclimbing networks typically obtain a benchmark performance of about 40% against

Table 1. Fraction of wins in 10 K benchmark trials of the neural nets of (Tesauro, 1992) in various testing procedures. "Gam" benchmarks against Gammontool, letting Gammontool race for both sides. "Pub-1" benchmarks against Pubeval, letting Pubeval race for both sides. "Pub-2" is a straight benchmark against Pubeval, i.e., the TD net plays the entire game including the race. "Pub-2" is an exact match to the testing procedure of Pollack and Blair.

Hidden units	Gam	Pub-1	Pub-2
10	0.564	0.527	0.507
20	0.619	0.571	0.557
40	0.655	0.611	0.602

Pubeval, and on one singular occasion scored 45%. They claim that this represents "similar levels of skills" to the results of (Tesauro, 1992), and therefore one can dismiss the expected outcome value-function approach as being "not essential" to the success of TD-Gammon. On this point the authors miss the mark. This is easily shown by re-benchmarking the 1992 TD nets against Pubeval; results are shown in Table 1.

For comparison with the Pollack and Blair results of 40–45% against Pubeval, the most appropriate figure is the score of nearly 56% of the Pub-2 benchmark of the 20 hidden unit net. (This net has an identical architecture to that of Pollack and Blair, except that it lacks a race/contact input feature, and it recognizes both wins and gammons, and thus makes plays that are suboptimal in a win-only benchmarking.) Readers familiar with backgammon will recognize that the comparison is not close: the TD net is significantly better. To put these figures in perspective, a 1% differential in the Pubeval benchmark would translate into about 35 rating points in the ratings system used in human tournaments and on FIBS, the internet backgammon server. Thus, the difference between 56% and 40–45% should translate into a difference of approximately 400–550 rating points, a very significant difference indeed, given that the difference between an average human player ($\sim$1500 rating) and a world-class player ($\sim 1900+$ rating) is about the same magnitude.

My current working hypothesis is that the weakness of hillclimbing relative to the TD results is due to hillclimbing's failure to extract nonlinear structure inherent in the problem domain, despite the presence of hidden units in its network architecture. One indication of this comes from the nature of Pubeval itself. Pubeval consists of two evaluation functions, each of which is a linear function of the raw board state. One of these evaluators is general-purpose and is used for nearly all move decisions, and the other is specialized to endgame "race" positions when the forces have broken contact. Hence it should be possible in principle for a multilayer perceptron net, which can represent nonlinear functions of the same raw board inputs, to beat Pubeval, provided that the learning algorithm can discover the nonlinear solution. TD learning can clearly do this, as indicated in Table 1. However, hillclimbing's failure to even equal Pubeval strongly suggests that it fails to capture any nonlinear structure.

A second supporting factor emerged in my own experiments replicating the Pollack and Blair results. These experiments exactly duplicated their network architecture and all of the implementation details in Section 2, except that uniform random noise was used instead of Gaussian random noise. My results were similar to theirs—the performance saturated in similar training times, and the best score I was able to achieve was 41% against Pubeval. Monitoring the average weight magnitude during training revealed that when the performance saturated, the weight magnitudes were still at very small values. It is well-known in this case that the MLP acts as a linear function approximator, since all of the sigmoidal units are operating in the linear regime.

In summary, there is a massive performance discrepancy between the hill-climbing results and the results of (Tesauro, 1992). Strong conclusions about why TD-Gammon worked are therefore unwarranted. The tantalizing preliminary indications regarding hillclimbing are worthy of further exploration, to determine if it really is incapable of learning nonlinear functions, and if so, for what reason. The required training times to extract the nonlinearities may be infeasible, or it may be that a network containing several thousand weights is too large to evolve by random mutations, or there may be some deeper theoretical principle at work.

References

Pollack, J.B., & Blair, A.D. (1998). Co-evolution in the successful learning of backgammon strategy. *Machine Learning*, 32.

Tesauro, G. (1992). Practical issues in temporal difference learning. *Machine Learning*, *8*, 257–277.