

G. Tesauro, Temporal difference learning and TD-Gammon

Joel Hoffman
CS 541

October 19, 2006

Your mission

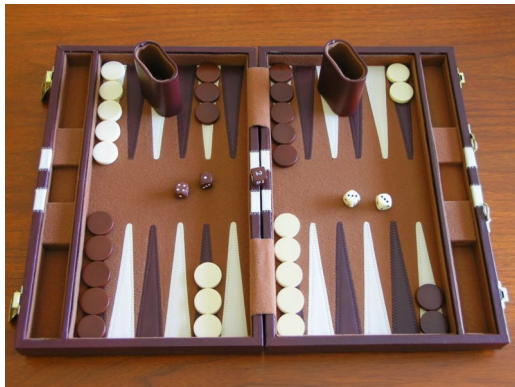
Goal:

Learn to achieve reward through optimal sequence of actions

The Enemy:

Temporal credit assignment

The Plan:



Play a lot of backgammon

Reinforcement Learning

- ▶ Unsupervised agent
- ▶ takes actions in environment
- ▶ FEEDBACK: consequences of actions alter the model
 - ▶ applied backwards in time at a decreasing, tunable rate

Temporal Credit Assignment Problem

- ▶ multiple actions taken to achieve goal
- ▶ which were responsible for success?
- ▶ what *is* (partial) success?

Random Evaluation Function?!?!

- ▶ Error signal at each step
- ▶ ... from the network itself
- ▶ ... even on untrained networks

Random Evaluation Function?!?!

- ▶ Error signal at each step
- ▶ ... from the network itself
- ▶ ... even on untrained networks
- ▶ Final unambiguous reward signal: Win or loss
- ▶ Tilts the randomness a little toward accurate learning

Random Evaluation Function?!?!

- ▶ Error signal at each step
- ▶ ... from the network itself
- ▶ ... even on untrained networks
- ▶ Final unambiguous reward signal: Win or loss
- ▶ Tilts the randomness a little toward accurate learning
 - ▶ (in several thousand games)
 - ▶ Initially took thousands of random moves just to finish a game

TD-Gammon vs. Neurogammon

TD-Gammon's model

At first:

- ▶ Only inputs were board positions
- ▶ 40-80 hidden units
- ▶ Equalled performance of Neurogammon after 200,000 self-played games

Then:

- ▶ Added human-identified features as additional inputs
 - ▶ Became invincible (nearly)

TD(λ) function

For each output unit Y :

$$w_{t+1} - w_t = \alpha(Y_{t+1} - Y_t) \sum_{k=1}^t \lambda^{t-k} \nabla_w Y_k$$

t Model state at the end of the last step

$t + 1$ Model state at the beginning of the next step

w Vector of neural network connection weights

α “learning rate” – exploration speed of the problem space

λ Feedback rate $\in (0, 1)$ – weighted error applied to past choices

$Y_{t+1} - Y_t$ Error signal at the current state

Y_k History of Y 's value from the first step (random) to last step

∇_w Gradient of network weights - Direction of steepest ascent

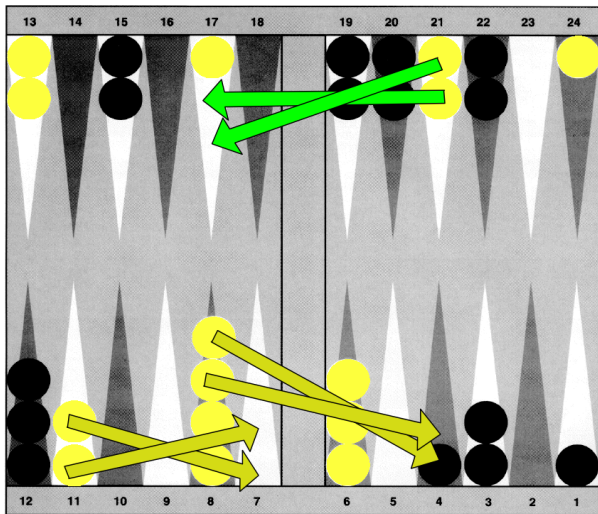


Figure 3. A complex situation where TD-Gammon's positional judgment is apparently superior to traditional expert thinking. White is to play 4-4. The obvious human play is 8-4*, 8-4, 11-7, 11-7. (The asterisk denotes that an opponent checker has been hit.) However, TD-Gammon's choice is the surprising 8-4*, 8-4, 21-17, 21-17! TD-Gammon's analysis of the two plays is given in Table 3.

Advantages of unsupervised TD learning

That is, advantages in backgammon specifically

- ▶ Can train continuously
- ▶ Not subject to human biases
- ▶ Has its own biases (explore too small a part of the state space)
 - ▶ Occurred in checkers and Go
 - ▶ Dice roll helps eliminate this
- ▶ Dice roll also smooths out the evaluation function
- ▶ Easy concepts are linear wrt the variables
 - ▶ (hidden variables don't help)