



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

Interaction
[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

Tools
[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)

Print/export
[Create a book](#)
[Download as PDF](#)
[Printable version](#)

Languages
[Add links](#)

[Create account](#) [Log in](#)

Article [Talk](#)

[Read](#) [Edit](#) [View history](#)

Temporal difference learning

From Wikipedia, the free encyclopedia

Temporal difference (TD) learning is a prediction-based [machine learning](#) method. It has primarily been used for the [reinforcement learning](#) problem, and is said to be "a combination of [Monte Carlo](#) ideas and [dynamic programming](#) (DP) ideas."^[1] TD resembles a [Monte Carlo method](#) because it learns by [sampling](#) the environment according to some *policy*, and is related to [dynamic programming](#) techniques as it approximates its current estimate based on previously learned estimates (a process known as [bootstrapping](#)). The TD learning algorithm is related to the temporal difference model of animal learning.^[2]

As a prediction method, TD learning takes into account the fact that subsequent predictions are often correlated in some sense. In standard supervised predictive learning, one learns only from actually observed values: A prediction is made, and when the observation is available, the prediction is adjusted to better match the observation. As elucidated by Richard Sutton, the core idea of TD learning is that we adjust predictions to match other, more accurate, predictions about the future.^[3] This procedure is a form of bootstrapping, as illustrated with the following example:

Suppose you wish to predict the weather for Saturday, and you have some model that predicts Saturday's weather, given the weather of each day in the week. In the standard case, you would wait until Saturday and then adjust all your models. However, when it is, for example, Friday, you should have a pretty good idea of what the weather would be on Saturday - and thus be able to change, say, Monday's model before Saturday arrives.^[3]

Mathematically speaking, both in a standard and a TD approach, we would try to optimize some cost function, related to the error in our predictions of the expectation of some random variable, $E[z]$. However, while in the standard approach we in some sense assume $E[z] = z$ (the actual observed value), in the TD approach we use a model. For the particular case of reinforcement learning, which is the major application of TD methods, z is the total return and $E[z]$ is given by the [Bellman equation](#) of the return.

Contents [\[hide\]](#)

- [1 Mathematical formulation](#)
- [2 TD-Lambda](#)
- [3 TD algorithm in neuroscience](#)
- [4 See also](#)
- [5 Notes](#)
- [6 Bibliography](#)
- [7 External links](#)

Mathematical formulation [\[edit\]](#)

Let r_t be the reinforcement on time step t . Let \bar{V}_t be the correct prediction that is equal to the discounted sum of all future reinforcement. The discounting is done by powers of factor of γ such that reinforcement at distant time step is less important.

$$\bar{V}_t = \sum_{i=0}^{\infty} \gamma^i r_{t+i}$$

where $0 \leq \gamma < 1$. This formula can be expanded

$$\bar{V}_t = r_t + \sum_{i=1}^{\infty} \gamma^i r_{t+i}$$

by changing the index of i to start from 0.

$$\bar{V}_t = r_t + \sum_{i=0}^{\infty} \gamma^{i+1} r_{t+i+1}$$

$$\bar{V}_t = r_t + \gamma \sum_{i=0}^{\infty} \gamma^i r_{t+i+1}$$

$$\bar{V}_t = r_t + \gamma \bar{V}_{t+1}$$

Thus, the reinforcement is the difference between the ideal prediction and the current prediction.

$$r_t = \bar{V}_t - \gamma \bar{V}_{t+1}$$

TD-Lambda [\[edit\]](#)

TD-Lambda is a learning algorithm invented by [Richard S. Sutton](#) based on earlier work on temporal difference learning by [Arthur Samuel](#).^[1] This algorithm was famously applied by [Gerald Tesauro](#) to create [TD-Gammon](#), a program that learned to play the game of [backgammon](#) at the level of expert human players.^[4]

The lambda (λ) parameter refers to the trace decay parameter, with $0 \leq \lambda \leq 1$. Higher settings lead to longer lasting traces; that is, a larger proportion of credit from a reward can be given to more distant states and actions when λ is higher, with $\lambda = 1$ producing parallel learning to Monte Carlo RL algorithms.

TD algorithm in neuroscience [\[edit\]](#)

The TD [algorithm](#) has also received attention in the field of [neuroscience](#). Researchers discovered that the firing rate of [dopamine neurons](#) in the [ventral tegmental area](#) (VTA) and [substantia nigra](#) (SNc) appear to mimic the error function in the algorithm.^[2] The error function reports back the difference between the estimated reward at any given state or time step and the actual reward received. The larger the error function, the larger the difference between the expected and actual reward. When this is paired with a stimulus that accurately reflects a future reward, the error can be used to associate the stimulus with the future [reward](#).

[Dopamine](#) cells appear to behave in a similar manner. In one experiment measurements of dopamine cells were made while training a monkey to associate a stimulus with the reward of juice.^[5] Initially the dopamine cells increased firing rates when the monkey received juice, indicating a difference in expected and actual rewards. Over time this increase in firing back propagated to the earliest reliable stimulus for the reward. Once the monkey was fully trained, there was no increase in firing rate upon presentation of the predicted reward. Continually, the firing rate for the dopamine cells decreased below normal activation when the expected reward was not produced. This mimics closely how the error function in TD is used for [reinforcement learning](#).

The relationship between the model and potential neurological function has produced research attempting to use TD to explain many aspects of behavioral research.^[6] It has also been used to study conditions such as [schizophrenia](#) or the consequences of pharmacological manipulations of dopamine on learning.^[7]

See also [\[edit\]](#)

- [Reinforcement learning](#)
- [Q-learning](#)
- [SARSA](#)
- [Rescorla-Wagner model](#)
- [PVLV](#)

Notes [\[edit\]](#)







- ^a ^b Richard Sutton and Andrew Barto (1998). *Reinforcement Learning*^[?]. MIT Press. ISBN 0-585-02445-6.
- ^a ^b Schultz, W, Dayan, P & Montague, PR. (1997). "A neural substrate of prediction and reward". *Science* **275** (5306): 1593–1599. doi:10.1126/science.275.5306.1593^[?]. PMID 9054347^[?].
- ^a ^b Richard Sutton (1988). "Learning to predict by the methods of temporal differences". *Machine Learning* **3** (1): 9–44. doi:10.1007/BF00115009^[?]. (A revised version is available on [Richard Sutton's publication page](#)^[?])
- ^a Tesauro, Gerald (March 1995). "Temporal Difference Learning and TD-Gammon"^[?]. *Communications of the ACM* **38** (3). Retrieved 2010-02-08.
- ^a Schultz, W. (1998). "Predictive reward signal of dopamine neurons". *J Neurophysiology* **80** (1): 1–27.
- ^a Dayan, P. (2001). "Motivated reinforcement learning"^[?] ^[PDF]. *Advances in Neural Information Processing Systems* (MIT Press) **14**: 11–18.
- ^a Smith, A., Li, M., Becker, S. and Kapur, S. (2006). "Dopamine, prediction error, and associative learning: a model-based account". *Network: Computation in Neural Systems* **17** (1): 61–84. doi:10.1080/09548980500361624^[?]. PMID 16613795^[?].

Bibliography [\[edit\]](#)

- Sutton, R.S., Barto A.G. (1990). "Time Derivative Models of Pavlovian Reinforcement"^[?] ^[PDF]. *Learning and Computational Neuroscience: Foundations of Adaptive Networks*: 497–537.
- Gerald Tesauro (March 1995). "Temporal Difference Learning and TD-Gammon"^[?]. *Communications of the ACM* **38** (3).

- Imran Ghory. [Reinforcement Learning in Board Games](#) .
- S. P. Meyn, 2007. [Control Techniques for Complex Networks](#) , Cambridge University Press, 2007. See final chapter, and appendix with abridged [Meyn & Tweedie](#) .

External links [\[edit\]](#)

- Scholarpedia [Temporal difference Learning](#) .
- TD-Gammon .
- TD-Networks Research Group .
- [Connect Four TDGravity Applet](#)  (+ mobile phone version) - self-learned using TD-Leaf method (combination of TD-Lambda with shallow tree search)
- [Self Learning Meta-Tic-Tac-Toe](#)  Example web app showing how temporal difference learning can be used to learn state evaluation constants for a minimax AI playing a simple board game.
- [Reinforcement Learning Problem](#) , document explaining how temporal difference learning can be used to speed up Q-learning

Categories: [Computational neuroscience](#) | [Machine learning algorithms](#)

This page was last modified on 24 July 2015, at 06:38.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

