Monte Carlo and Temporal Difference prediction

Hendrik Baier

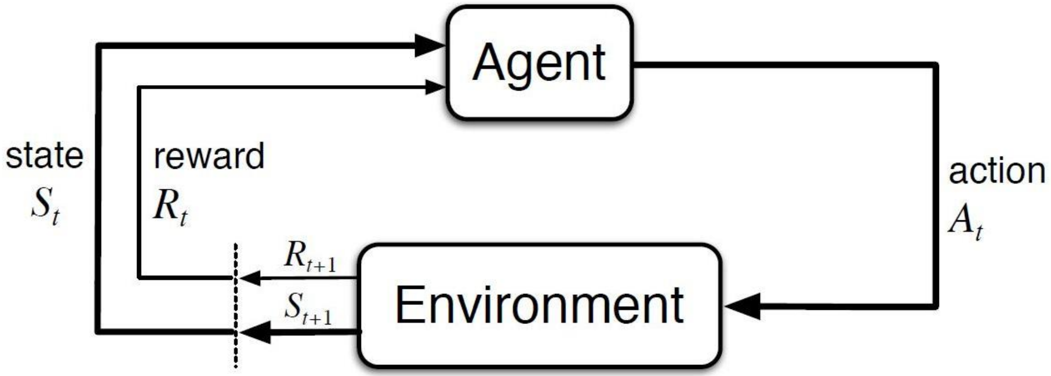
Centrum Wiskunde & Informatica

Motivation

Centrum Wiskunde & Informatica

Refresher: RL and MDPs

- reinforcement learning: learning from trial and error
- useful model for RL: *Markov Decision Process*



CWI
Centrum Wiskunde & Informatica

Refresher: MDPs

- MDP: set of states S, set of actions A, *dynamics*

$$p(s', r \mid s, a) \doteq \Pr\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\}$$

- Dynamics unknown: we have to learn from *trajectories*

$$S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, \ldots$$

- assuming *episodic* tasks here (all trajectories end)

Refresher: MDPs

- What are we trying to achieve?
- (discounted) sum of rewards is called *return*

$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots$$

- behavior of agent is described by a *policy*

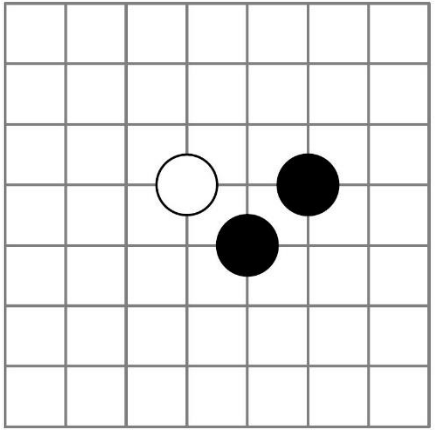
$\pi(a|s)$ (stochastic) or $\pi(s)$ (deterministic)

Centrum Wiskunde & Informatica

Refresher: value functions

- *State value function*: expected return when starting in s and following π from there

$$v_\pi(s) \doteq \mathbb{E}_\pi[G_t \mid S_t = s]$$



Prediction/evaluation vs. control

- *Control:* Solving the full RL problem of finding the *optimal* policy π_* and its value function v_*
- *Prediction:* Solving the important subproblem first of finding the value function of any *given* policy

$$\pi \rightarrow v_\pi$$

Two basic prediction methods

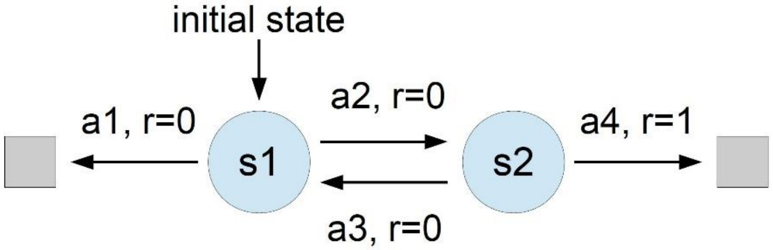
- *Monte Carlo (MC)* prediction: estimating values based on sampled returns of whole episodes
- *Temporal Difference (TD)* prediction: estimating values based on estimated values of next states (bootstrapping)
- Both are *model-free*: dynamics not given or learned
- (And they will turn out to be special cases of the same algorithm in the end!)

Centrum Wiskunde & Informatica

Assignment 1

Centrum Wiskunde & Informatica

1. What are the equivalents of prediction and control in the single-state case, the bandit problem?
2. Why could a video game company be interested in both prediction and control?
3. For this MDP:



   a) Solve the control problem (find π_*)
   b) Solve the prediction problem for the uniformly random policy by hand (find v_π)
   c) Why do we need a different approach for most interesting MDPs?

Monte Carlo prediction

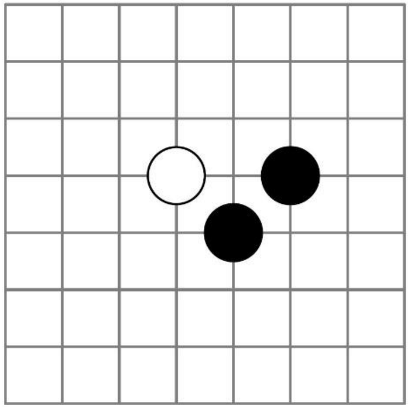
Centrum Wiskunde & Informatica

Monte Carlo prediction

- MC methods learn directly from episodes of experience
- MC uses simplest possible idea: *value = average return*
  - episode 1 counting from s: return 10
  - episode 2 counting from s: return 12
  - current best estimate of value of s (under given policy): 11

Monte Carlo prediction

state s:



- game results starting from s: loss, loss, win, loss, win...
  - returns: 0,0,1,0,1...
- current estimate of $v_\pi(s)$: 0.4

Monte Carlo prediction

- goal: learn v_π from episodes of experience:

  $$S_0, A_0, R_1, S_1, A_1, R_2, \ldots, S \sim \pi$$

- Value function is defined as expected return:

  $$v_\pi(s) \doteq \mathbb{E}_\pi[G_t \mid S_t = s]$$

- MC prediction/evaluation uses *empirical mean* return to estimate expected return
- Works because of the law of large numbers

CWI Centrum Wiskunde & Informatica

Monte Carlo prediction

- Input: a policy π to be evaluated
- Initialize:
  - $V(s)$ arbitrarily for all $s \in S$
  - $Sum(s) \leftarrow$ zero for all $s \in S$; $N(s) \leftarrow$ zero for all $s \in S$
- For each episode

  For each visited state S_t with following return G_t

  $$N(S_t) \leftarrow N(S_t) + 1$$

  $$Sum(S_t) \leftarrow Sum(S_t) + G_t$$

  $$V(S_t) \leftarrow Sum(S_t) / N(S_t)$$

(*first-visit* and *every-visit* updates both work: **$V(s) \rightarrow v_\pi$ as $N(s) \rightarrow \infty$**)

Incremental average

For each visited state S_t with following return G_t

$N(S_t) \leftarrow N(S_t) + 1$

$Sum(S_t) \leftarrow Sum(S_t) + G_t$

$V(S_t) \leftarrow Sum(S_t) / N(S_t)$

$$m_n = \frac{1}{n}\sum_{i=1}^{n} a_i \qquad m_n = m_{n-1} + \frac{a_n - m_{n-1}}{n}$$

Incremental average

For each visited state S_t with following return G_t

$$N(S_t) \leftarrow N(S_t) + 1$$

$$V(S_t) \leftarrow V(S_t) + \frac{1}{N(St)} (G_t - V(S_t))$$

Constant step-size

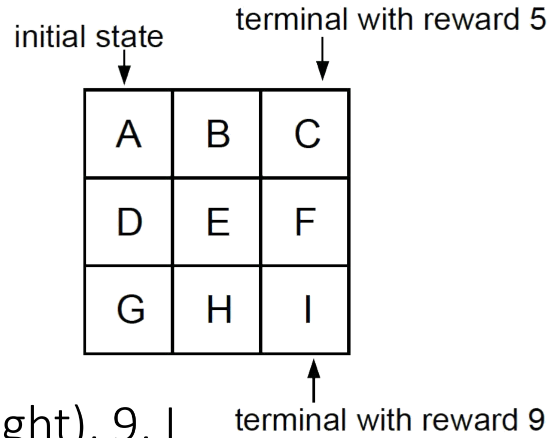
For each visited state S_t with following return G_t

$$N(S_t) \leftarrow N(S_t) + 1$$

$$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t))$$

Assignment 2

1. For this gridworld MDP:



and these two observed episodes:

A, (down), 0, D, (right), 0, E, (down), 0, H, (right), 9, I

A, (right), 0, B, (down), 0, E, (right), 0, F, (up), 5, C

What are the resulting MC value estimates V(s) for all states? Assume α = 1/N(s)

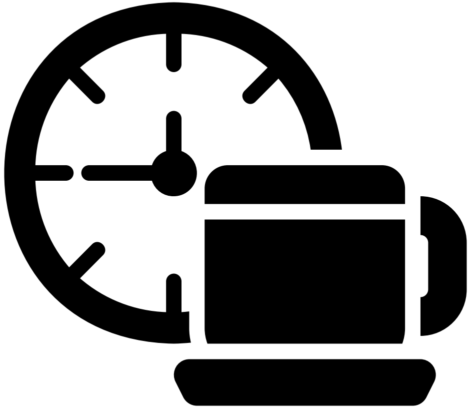
2. What's the effect of a constant step-size on updates (earlier vs. later episodes)? Why could that be helpful for control later?

3. What are some limitations of Monte Carlo prediction? What does it require of the input data?

Bonus: Prove the formula for the incremental average

Centrum Wiskunde & Informatica

15 minutes break!



Temporal difference prediction

Centrum Wiskunde & Informatica

Temporal difference prediction

- TD methods also learn directly from episodes of experience
- TD methods are also model-free: no knowledge of the MDP's transition or reward structure is needed
- TD can learn online
- TD uses *bootstrapping*: improving an estimate based on another estimate

CWI
Centrum Wiskunde & Informatica

Temporal difference prediction

- goal: learn v_π *online* from experience under π
- You don't have to die in traffic to learn about safe driving!
- Monte Carlo prediction:
  - Update value estimate $V(S_t)$ towards *actual* return G_t

    $$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t))$$
- TD prediction in its simplest form: TD(0)
  - Update value estimate $V(S_t)$ towards *estimated* return $R_{t+1} + \gamma V(S_{t+1})$

    $$V(S_t) \leftarrow V(S_t) + \alpha\boxed{(\underbrace{\boxed{R_{t+1} + \gamma V(S_{t+1})}}_{\text{TD target}} - V(S_t))}$$

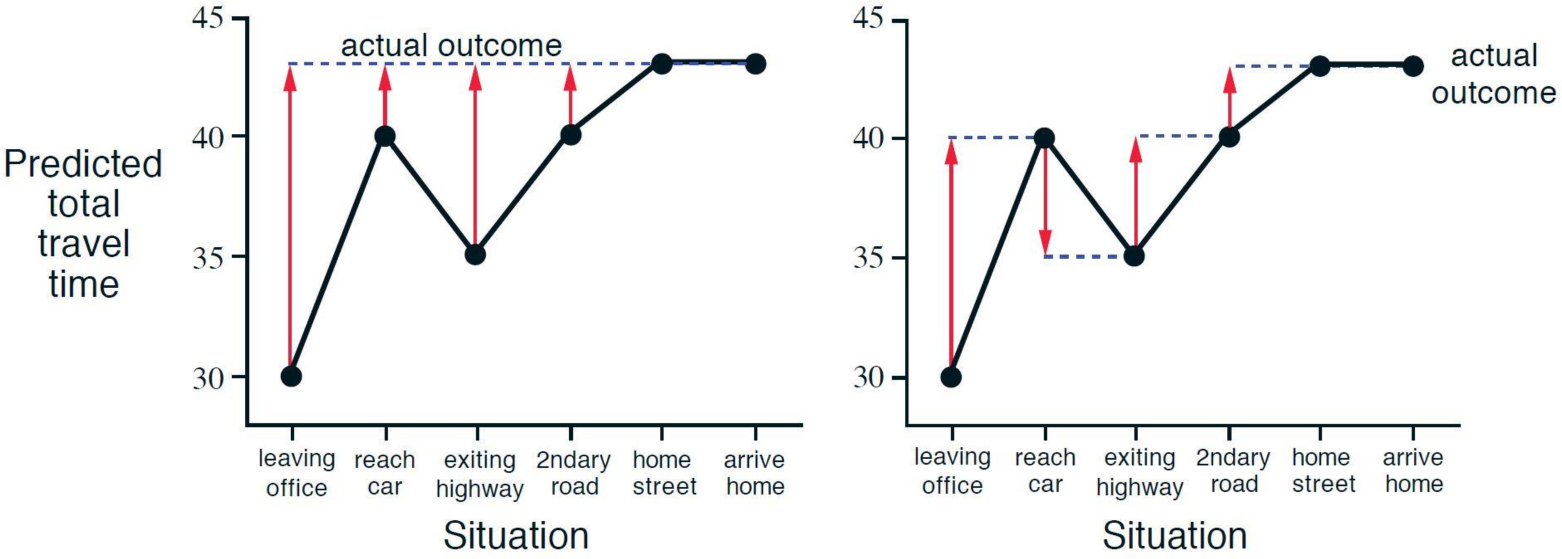
TD target TD error

Centrum Wiskunde & Informatica

MC vs. TD example

State	*Elapsed Time (minutes)*	*Predicted Time to Go*	*Predicted Total Time*
leaving office, friday at 6	0	30	30
reach car, raining	5	35	40
exiting highway	20	15	35
2ndary road, behind truck	30	10	40
entering home street	40	3	43
arrive home	43	0	43

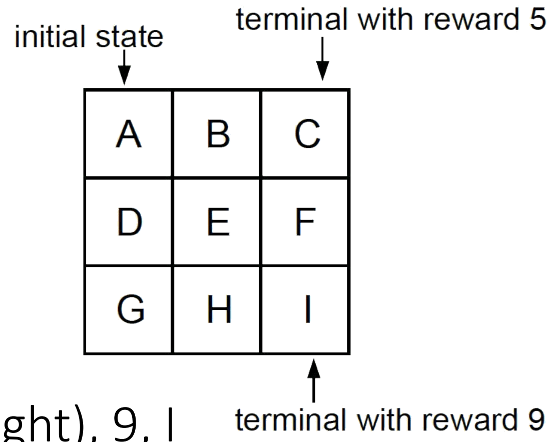
MC vs. TD example



Assignment 3

Centrum Wiskunde & Informatica

1. For this gridworld MDP:



and these two observed episodes:

A, (down), 0, D, (right), 0, E, (down), 0, H, (right), 9, I

A, (right), 0, B, (down), 0, E, (right), 0, F, (up), 5, C

What are the resulting TD value estimates V(s) for all states? Assume α=0.5, γ=1

2. What could be some advantages of TD learning over MC learning?

3. Back to the car driving example: Imagine you've learned good estimates for traffic times. Now you're moving to a different house, but you're still entering the highway at the same place. Can you explain why TD updates are going to be better than MC, at least initially?

CWI Centrum Wiskunde & Informatica

Comparing MC and TD

Bias/variance trade-off

- The return G_t is an *unbiased* estimate of $v_\pi(S_t)$

$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots$$

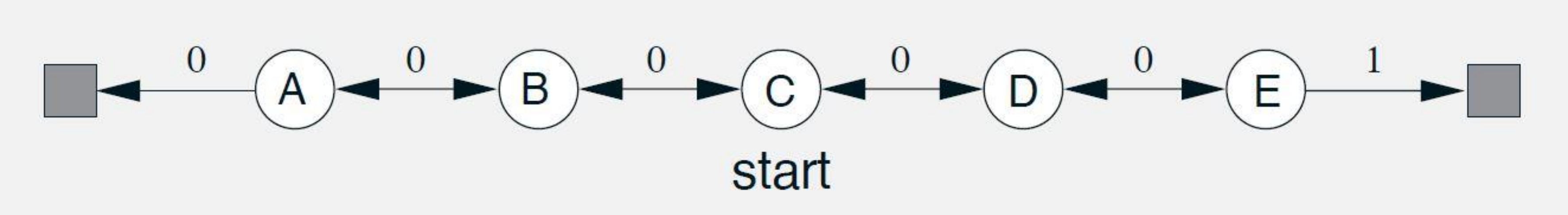
- The "true TD target" $R_{t+1} + \gamma v_\pi(S_{t+1})$ is also unbiased
- The TD target $R_{t+1} + \gamma V(S_{t+1})$ is a *biased* estimate of $v_\pi(S_t)$
- However, the TD target has a much lower *variance* than G_t:
  - Return depends on *many* random actions, transitions, rewards
  - TD target depends on *one* random action, transition, reward

MC vs. TD

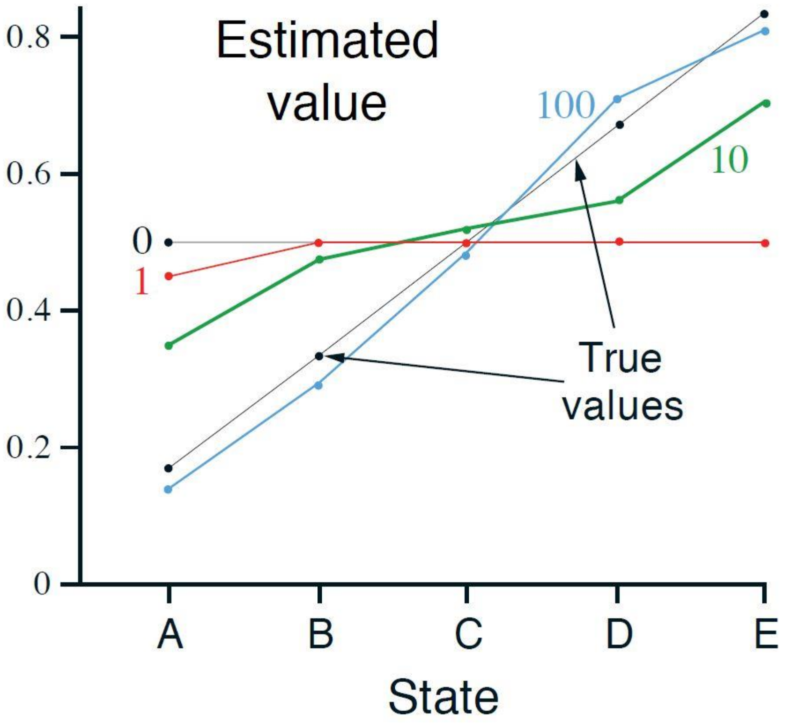
- MC has high variance, zero bias
  - converges (even with function approximation) to v_π
  - not too sensitive to initial values (except for including them in average)
  - very simple to understand and use out of the box
- TD has much lower variance, but some bias
  - often much more efficient than MC
  - TD(0) converges to v_π, but not always with function approximation
  - more sensitive to initial values (bootstrapping from them)

CWI Centrum Wiskunde & Informatica

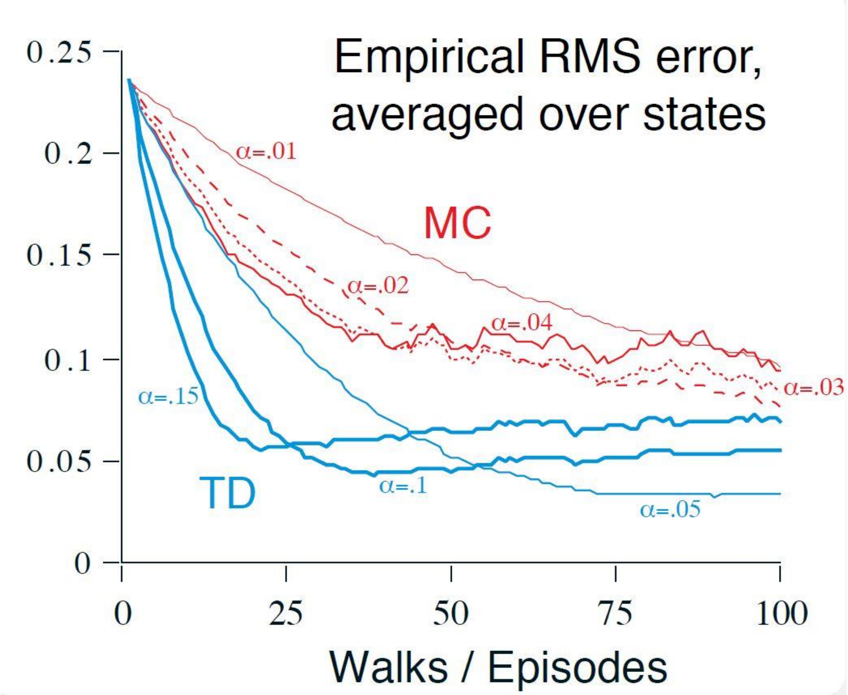
Random walk example



Random walk example



Random walk example



CWI Centrum Wiskunde & Informatica

Batch updating

- In the limit of experience → ∞, MC and TD have $V(S) \rightarrow v_\pi(S)$
- But what if we have a limited amount of experience?
- We typically present this experience over and over, computing only 1 average update from it each time → *batch updating* until convergence
- With batch updating, TD(0) converges to a specific answer, and constant-α MC also converges to a specific answer – but to a different one! This can give us some intuition for their differences in behavior/performance also in the non-batch case.

Centrum Wiskunde & Informatica

Mini assignment

Estimating state values

We are faced with an unknown MDP and observe the following 8 episodes:

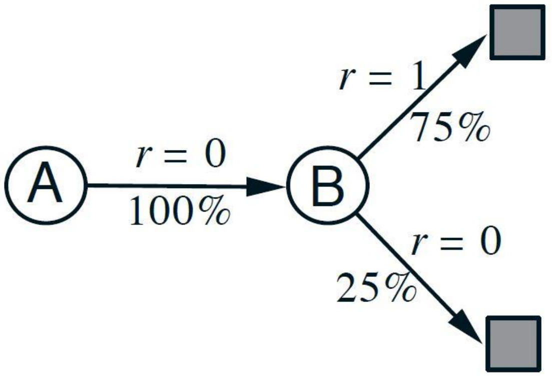
A, 0, B, 0	B, 1
B, 1	B, 1
B, 1	B, 1
B, 1	B, 0

Given this batch of data, what are your predictions/estimates for the values of A and B? (Just use your intuition, no specific algorithm)

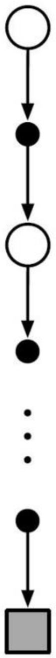
Centrum Wiskunde & Informatica

Batch updating

- Batch Monte Carlo finds the values that minimize the mean-squared error on the data. Here, that means V(A)=0.
- Batch TD(0) finds the values that would be correct for the maximum-likelihood model of the MDP. Here, that is V(A)=3/4, and the model is:
- In a sense, TD implicitl ov property that relates state values to es not. That’s why TD is often more effici rant to violated Markov property (e.g. es!)



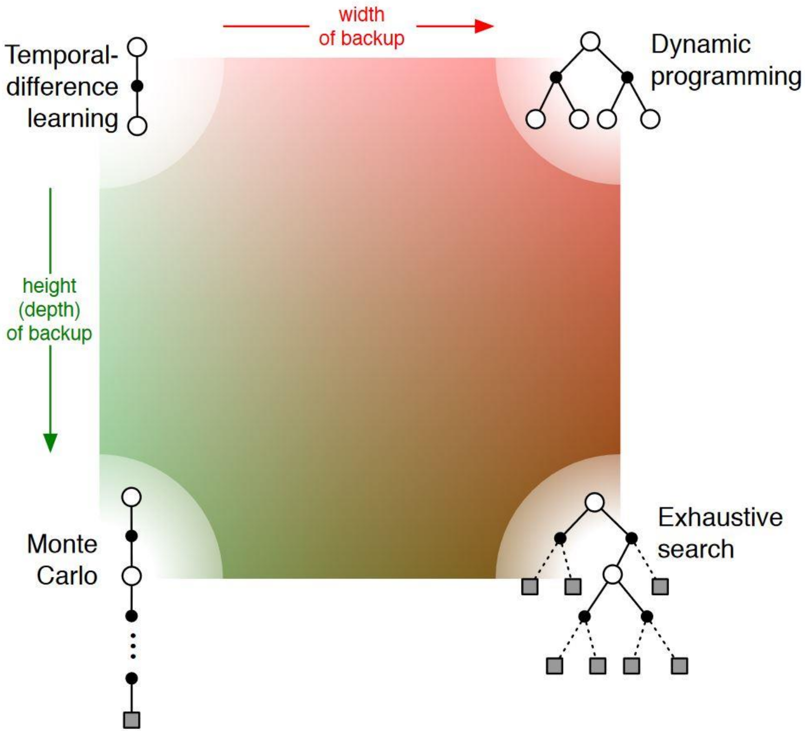
Monte Carlo backup



Temporal difference backup

Centrum Wiskunde & Informatica

Unified view of reinforcement learning



Centrum Wiskunde & Informatica

Need more explanation?

Need more explanation?

- On YouTube: „RL Course by David Silver - Lecture 4: Model-Free Prediction”
  - with some bonus content at the end
- In Sutton & Barto: Chapter 5 (introduction & 5.1 – 5.2) and Chapter 6 (introduction & 6.1 – 6.3)
  - with more examples and in-depth explanations