On Predictive Representations for Efficient Temporal Credit Assignment

Anthony GX-Chen
Center for Data Science, New York University
Mila / McGill University
anthony.gx.chen@nyu.edu

Veronica Chelu
Mila / McGill University

Blake Richards
Mila / McGill University

Joelle Pineau
Mila / McGill University
Meta AI Research

Abstract

We introduce a novel bootstrapping target for efficient value learning: the η-return mixture. This target combines value-predictive knowledge (used by temporal difference methods) with state-predictive knowledge in the form of successor representations (SR). A parameter η capturing how much to rely on each. We illustrate that incorporating predictive knowledge through our $\eta\gamma$-discounted SR model makes more efficient use of sampled experience, compared to either extreme: bootstrapping entirely on the value function estimate, or bootstrapping on the product of separately estimated SR and instantaneous rewards. We empirically show this approach leads to faster policy evaluation and better control performance, for tabular and nonlinear function approximations, indicating scalability and generality. Finally, our model is potentially relevant as an algorithmic level model for the hippocampus: it encode predictive maps in the form of SR, and use it for rapid learning.

Keywords: Reinforcement Learning, Prediction, Successor Representation, Hippocampus, Efficient Credit Assignment, Learning

Acknowledgements

AGXC was supported by the NSERC CGS-M, FRQNT, and UNIQUE excellence scholarships. This work was supported by NSERC (Discovery Grant: RGPIN-2020-05105; Discovery Accelerator Supplement: RGPAS-2020-00031) and CIFAR (Canada AI Chair; Learning in Machine and Brains Fellowship) grants to BAR. This research was enabled in part by computational resources provided by Calcul Quebec (www.calculquebec.ca) and Compute Canada (www.computecanada.ca). We thank the anonymous reviewers for their valuable feedback. We thank our colleagues at Mila for the insightful discussions that have made this project better.

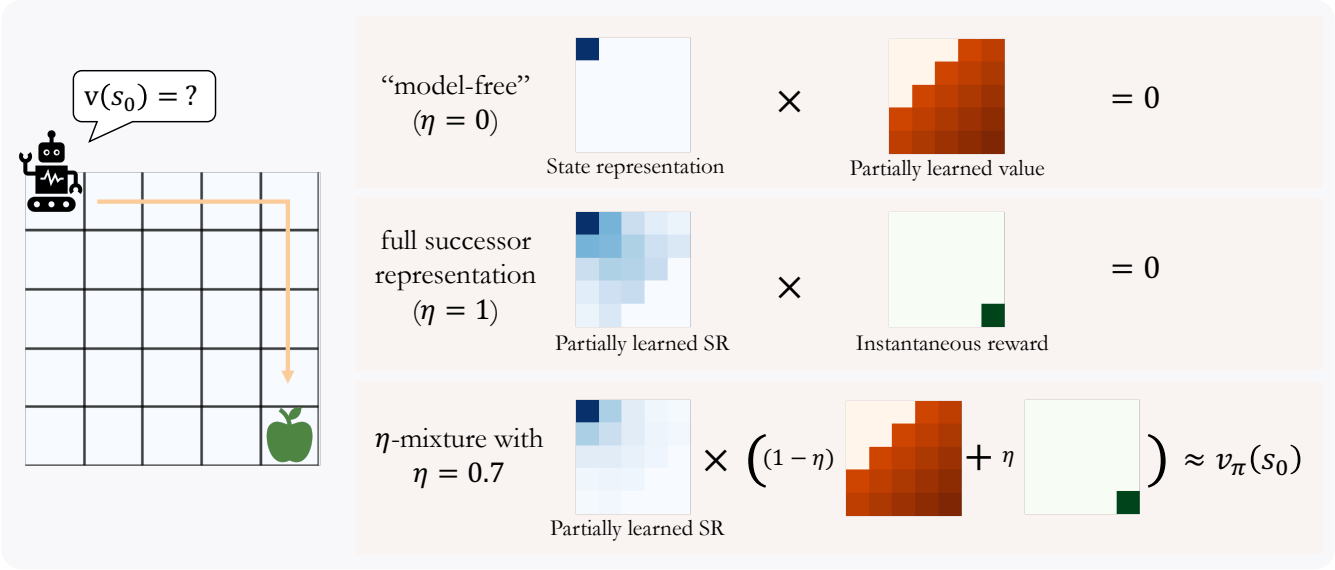

Figure 1: **Intuition for efficient value learning using an η-*mixture*.** Suppose the agent wishes to learn about the value of the starting state, $\mathbf{v}(s_0)$. A "model-free" value estimate only considers the immediate state and predicts future rewards (via value function). The full successor representation (SR) predicts future states but only considers the immediate reward. Using an η-mixture with intermediate η, one predicts *both* future rewards and future states, and combine them in a complementary way to more quickly arrive at the correct value estimate.

1 Introduction

The problem of efficient temporal credit assignment—associating distance rewards with the states and actions that caused them—remains a fundamental challenge for reinforcement learning (RL) algorithms. Similarly, how the brain does efficient temporal credit assignment remains poorly understood. A recent neuroscience theory proposes that the brain encodes representations about future states in a predictive cognitive map known as the *successor representations* (SR) [Dayan, 1993, Stachenfeld et al., 2014, 2017]. This is a powerful framework explaining a range of previous physiological observations in the brain region *hippocampus* [Stachenfeld et al., 2017], and have been extended to explain human behaviour [Momennejad et al., 2017, Gershman, 2018, Momennejad, 2020]. We ask: given that the hippocampus encode SR and have been long been associated with rapid learning [McClelland et al., 1995], could SR be used to improve the fundamental RL question of efficient temporal credit assignment?

We answer the above in the affirmative, by introducing a novel algorithm which use the SR for efficient propagation of value and reward information. Unlike the "traditional" factorization of value into the SR and instantaneous rewards, we show that one can factorize a more general cumulative quantity—the averaged expected multi-step return—which naturally results in a complementary combination of state predictive (SR), value predictive, and instantaneous reward information. This leads to a novel backup target, the η**-return mixture**, for efficient credit assignment.

Our contribution is as follows,

- We introduce the *η-return mixture*, a simple yet novel way of constructing a backup target for value learning, using an $\eta\gamma$-discounted SR to efficiently combine state-predictive, value, and instantaneous rewards information.
- We describe a new value learning algorithm using the η-return mixture as the learning target.
- We provide empirical results showing more efficient use of experience with the η-return mixture as the learning target, in both prediction and control, for tabular and nonlinear (neural network based) approximation.
- For neuroscience, we expand the space of algorithmic models of how the SR might be used, which is the first to jointly explain how the SR can be directly used in rapid learning.

2 Reinforcement Learning Preliminaries

We denote random variables with uppercase (e.g., S) and the obtained values with lowercase letters (e.g., $S = s$). Multi-dimensional functions or vectors are bolded (e.g., $\mathbf{v}$).

A discounted Markov Decision Process (MDP) [Puterman, 1994] is defined as the tuple $(\mathcal{S}, \mathcal{A}, P, r)$, with state space $\mathcal{S}$, action space $\mathcal{A}$, reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and transition probability function $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathcal{P}(\mathcal{S})$ (with $\mathcal{P}(\mathcal{S})$ the set of probability distributions on $\mathcal{S}$, and $P(s'|s, a)$ the probability of transitioning to state s' by choosing action a at state s). A policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ maps states to distributions over actions; $\pi(a|s)$ denotes the probability of choosing action a in state s. Let S_t, A_t, R_t denote the random variables of state, action and reward at time t, respectively.

2.1 Value Learning

In RL, the goal is to estimate and maximize *value*. We use $\mathbb{E}_\pi[\cdot]$ to denote the expectation taken under a policy π and the MDP dynamics,

$$\mathbf{v}_\pi(s_t) = \mathbb{E}_\pi[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ...|S_t = s_t]\,. \tag{1}$$

We define a tabular *value function*, $\mathbf{v} \in \mathbb{R}^{|\mathcal{S}|}$, a $|\mathcal{S}|$-dimensional vector, and $\mathbf{v}(s)$ to denote the entry of this vector which contain the value estimate for state s. For notation convenience we also use a "one-hot" indicator variable, $\mathbf{1}_{S_t} = [0, ..., 1, ..., 0]^\top \in \mathbb{R}^{|\mathcal{S}|}$, a random variable for the state encountered at time t. We can equivalently write $\mathbf{v}(S_t) = \mathbf{1}_{S_t}^\top \mathbf{v}$.

The goal of value learning is to update the value function to approximate the true value, $\mathbf{v}(s) \approx \mathbf{v}_\pi(s), \forall s \in \mathcal{S}$. In general, this is done by constructing a learning target, G_t, to update the value function towards,

$$\mathbf{v}(S_t) = \mathbf{v}(S_t) + \alpha[G_t - \mathbf{v}(S_t)]\,, \tag{2}$$

with step-size $\alpha \in (0, 1]$. A variety of learning targets can be constructed, such as the one-step return used in TD(0),

$$G^{(1)} \equiv R_{t+1} + \gamma \mathbf{1}_{S_{t+1}}^\top \mathbf{v}\,, \tag{3}$$

which when combined with equation 2 give rise to the canonical TD update, $\mathbf{v}(S_t) = \mathbf{v}(S_t) + \alpha[R_{t+1} + \gamma \mathbf{v}(S_{t+1}) - \mathbf{v}(S_t)]$. Other kinds of returns are also available. Suppose we have a model of transition dynamics which gives us the expected states of the next step given the current state, $\mathbf{P}^{(k)}(\mathbf{1}_{s_t}) = \mathbb{E}_\pi[\mathbf{1}_{S_{t+k}}|S_t = s_t]$, we can "roll-out" this model to constructed multi-step expected returns, for example,

$$G^{(2)} \equiv R_{t+1} + \gamma \mathbf{P}^{(1)}(\mathbf{1}_{S_t})^\top \mathbf{r} + \gamma^2 \mathbf{P}^{(2)}(\mathbf{1}_{S_t})^\top \mathbf{v}\,, \qquad \text{2-step expected return.} \tag{4}$$

$$G^{(3)} \equiv R_{t+1} + \gamma \mathbf{P}^{(1)}(\mathbf{1}_{S_t})^\top \mathbf{r} + \gamma^2 \mathbf{P}^{(2)}(\mathbf{1}_{S_t})^\top \mathbf{r} + \gamma^3 \mathbf{P}^{(3)}(\mathbf{1}_{S_t})^\top \mathbf{v}\,. \qquad \text{3-step expected return.} \tag{5}$$

2.2 Successor Representation (SR)

The successor representation [Dayan, 1993] encodes each state as a temporally-dependant, predictive representations about states an agent will encounter in the future. Denote $\boldsymbol{\psi}(s_t) \in \mathbb{R}^{|\mathcal{S}|}$ as the SR for state s_t,

$$\boldsymbol{\psi}_\pi(s_t) \equiv \mathbb{E}_\pi[\textstyle\sum_{n=0}^{\infty} \mathbf{1}_{S_{t+n}}|S_t = s_t]\,. \tag{6}$$

The SR can be used to estimate the value, by linearly combining with the instantaneous reward function, $\mathbf{r}_\pi$,

$$\mathbf{v}_\pi(s_t) = \boldsymbol{\psi}_\pi(s_t)^\top \mathbf{r}_\pi\,, \qquad \text{where } \mathbf{r}_\pi \in \mathbb{R}^{|\mathcal{S}|}, \mathbf{r}_\pi(s_t) = \mathbb{E}_\pi[R_{t+1}|S_t = s_t]\,. \tag{7}$$

The SR value estimate can also be thought of as an expected "infinite-step" return,

$$G^{(\infty)} \equiv R_{t+1} + \gamma \textstyle\sum_{n=1}^{\infty} \gamma^{n-1} \mathbf{P}^{(n)}(\mathbf{1}_{S_{t+1}})^\top \mathbf{r} = R_{t+1} + \gamma \boldsymbol{\psi}(S_{t+1})^\top \mathbf{r}\,. \tag{8}$$

In neuroscience, Stachenfeld et al. [2017] theorizes that the hippocampus encodes the SR. Specifically, the population activity of the hippocampal place cells at time t encode the SR of state S_t: $\boldsymbol{\psi}_\pi(S_t)$.

3 The η-return mixture

We derive a novel target for value learning, starting from an exponential averaging over all multi-step expected returns,

$$G_t^\eta = (1 - \eta)\, \mathbb{E}_\pi\Big[\sum_{n=1}^{\infty} \eta^{n-1} \underbrace{\big[\textstyle\sum_{k=0}^{n-1} \gamma^k \mathbf{r}(S_{t+k}) + \gamma^n \mathbf{v}(S_{t+n})\big]}_{\text{n-step expected return}} \Big]\,, \tag{9}$$

$$= R_{t+1} + \gamma\, \mathbb{E}_\pi\big[\sum_{n=1}^{\infty} (\eta\gamma)^{n-1} [(1 - \eta)\, \mathbf{v}(S_{t+n}) + \eta\, \mathbf{r}(S_{t+n})] \big]\,, \tag{10}$$

$$= R_{t+1} + \gamma\, \mathbb{E}_\pi[\textstyle\sum_{n=1}^{\infty} (\eta\gamma)^{n-1} \mathbf{P}^{(n)}(\mathbf{1}_{S_{t+1}})]^\top [(1 - \eta)\, \mathbf{v} + \eta\, \mathbf{r}]\,, \tag{11}$$

$$= R_{t+1} + \gamma\, \boldsymbol{\psi}^\eta(S_{t+1})^\top [(1 - \eta)\, \mathbf{v} + \eta\, \mathbf{r}]\,, \qquad \textbf{[the } \eta\textbf{-return mixture]} \tag{12}$$

where $\eta \in [0, 1]$ is the averaging factor, and $\boldsymbol{\psi}^{\eta}(S_{t+1})^{\top}$ is an $\eta\gamma$-discounted SR.[1]

Special Cases We observe that when $\eta = 0$, we recover the usual TD(0) target (equation 3),

$$G_t^{\eta=0} = R_{t+1} + \gamma\, \mathbb{E}_{\pi}[\textstyle\sum_{n=1}^{\infty}(0)^{n-1}\mathbf{P}^{(n)}(\mathbf{1}_{S_{t+1}})]^{\top}[1\,\mathbf{v} + 0\,\mathbf{r}] = R_{t+1} + \gamma\mathbf{v}(S_{t+1}) = G_t^{(1)}\,, \tag{13}$$

which uses only the value function $\mathbf{v}$ to estimate the value of S_{t+1}. This is sometimes called "model-free" learning.

When $\eta = 1$, we recover the one-step target with the SR value estimate (equation 8),

$$G_t^{\eta=1} = R_{t+1} + \gamma\, \mathbb{E}_{\pi}[\textstyle\sum_{n=1}^{\infty}\gamma^{n-1}\mathbf{P}^{(n)}(\mathbf{1}_{S_{t+1}})]^{\top}[0\,\mathbf{v} + 1\,\mathbf{r}] = R_{t+1} + \gamma\,\boldsymbol{\psi}(S_{t+1})^{\top}\mathbf{r} = G_t^{(\infty)}\,, \tag{14}$$

which uses an expected "infinite model"—the SR—to estimate the value of S_{t+1}.

Estimating the η-return mixture There are three components in the η-return mixture: the value function $\mathbf{v}$, the $\eta\gamma$-discounted SR $\boldsymbol{\psi}$, and the instantaneous reward function $\mathbf{r}$. All three components can be learned jointly using single-step experiences of the form (S_t, R_{t+1}, S_{t+1}). Further, while we have intentionally introduced concepts in the tabular setting for easier intuitions, equation 12 is readily extendable to the function approximation setting using successor features [Barreto et al., 2017]. For a more comprehensive formulation in the approximation setting with algorithms for linear and nonlinear function approximation, see GX-Chen et al. [2022].

4 Results

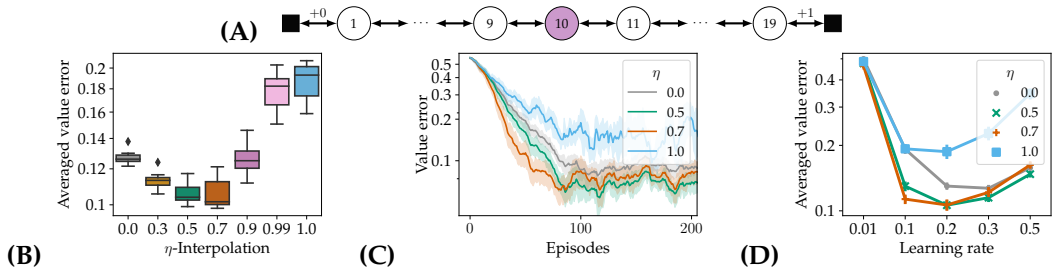

Figure 2: **Policy evaluation in 19-state tabular random chain. (A)** The agent starts in the center and transitions left/right randomly until either end is reached. Reward is 0 on all transitions, except the on the right-side termination, which yields a reward of $+1$. **(B) Parameter study for η:** The y-axis shows the root mean squared error (RMSE) (minimized over learning rates for each η) averaged over first 400 episodes. **(C) Learning dynamics:** The y-axis shows the RMSE for four illustrative η values. **(D) Parameter study for the learning rate** The y-axis shows the RMSE for four illustrative η values, across different learning rates. Results averages over first 400 episodes. Error bars and shaded areas denote 95 confidence intervals (some too small to see), with 10 independent seeds.

In both policy evaluation with tabular features (figure 2), and in nonlinear control with neural network policies trained end-to-end (figure 3), we observe "U"-shaped performance curves as we interpolate across η's. This suggests that an *intermediate* value of η results in the most optimal learning, out-performing the purely "model-free" ($\eta = 0$) and the fully factorized successor representation value estimates ($\eta = 1$).

5 Discussion

We introduce the η-return mixture, a generalization of the one-step "model-free" return, and the expected "infinite-step" return using the SR value estimate. The parameter η controls learning SRs at different temporal horizons, smoothly interpolating between a purely "model-free" estimate ($\eta = 0$) and an estimate constructed by estimating future states ($\eta = 1$). Crucially, with an intermediate $0 < \eta < 1$, we combine the value function, instantaneous reward function, and state prediction in the SR in a complementary way. We empirically show this leads to the most efficient learning.

References

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993.

[1] See GX-Chen et al. [2022] for a more detailed derivation.

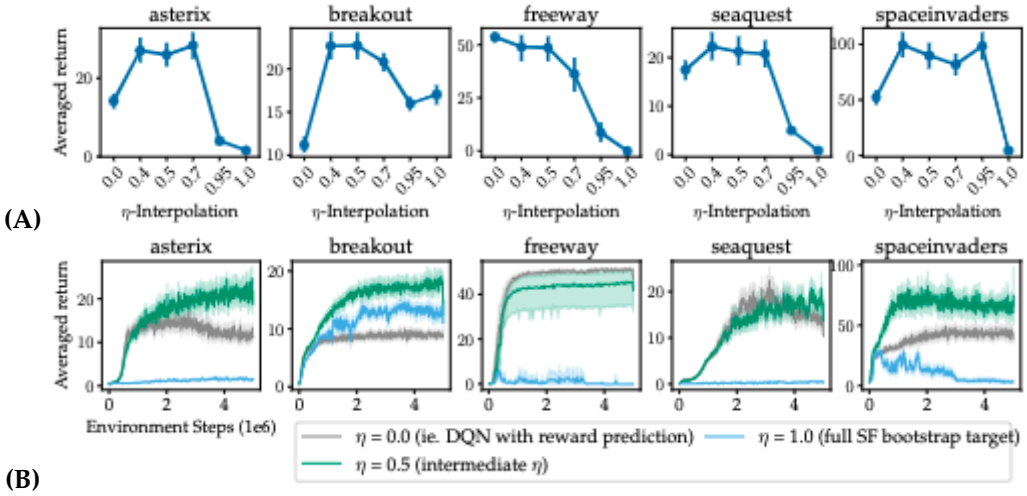

Figure 3: **Performance for value-based control in the Mini-Atari [Young and Tian, 2019] environment** using a DQN [Mnih et al., 2015] with the η-return mixture. **(A)** Parameter study for different values of η. The y-axis shows the average performance over 10k timesteps and 10 seeds using an ϵ-greedy policy with $\epsilon = 0.05$, after stopping training after $5e6$ learning steps. **(B)** Learning curves for 3 illustrative η values over the course of training. The y-axis displays the average return over 10 independent seed. Shaded area and error bars depicts 95 confidence interval.

Kimberly L Stachenfeld, Matthew Botvinick, and Samuel J Gershman. Design principles of the hippocampal cognitive map. *Advances in neural information processing systems*, 27:2528–2536, 2014.

Kimberly L Stachenfeld, Matthew M Botvinick, and Samuel J Gershman. The hippocampus as a predictive map. *Nature neuroscience*, 20(11):1643–1653, 2017.

Ida Momennejad, Evan M Russek, Jin H Cheong, Matthew M Botvinick, Nathaniel Douglass Daw, and Samuel J Gershman. The successor representation in human reinforcement learning. *Nature human behaviour*, 1(9):680–692, 2017.

Samuel J Gershman. The successor representation: its computational logic and neural substrates. *Journal of Neuroscience*, 38(33):7193–7200, 2018.

Ida Momennejad. Learning structures: Predictive representations, replay, and generalization. *Current Opinion in Behavioral Sciences*, 32:155–166, 2020.

James L McClelland, Bruce L McNaughton, and Randall C O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review*, 102(3):419, 1995.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., USA, 1st edition, 1994. ISBN 0471619779.

Anthony GX-Chen, Veronica Chelu, Blake A Richards, and Joelle Pineau. A generalized bootstrap target for value-learning, efficiently combining value and feature predictions. *arXiv preprint arXiv:2201.01836*, 2022.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In *Advances in neural information processing systems*, pages 4055–4065, 2017.

Kenny Young and Tian Tian. Minatar: An atari-inspired testbed for thorough and reproducible reinforcement learning experiments. *arXiv preprint arXiv:1903.03176*, 2019.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 2015.