Probabilistic Successor Representations with Kalman Temporal Differences

Jesse P. Geerts[1, 2], **Kimberly L. Stachenfeld**[3], **Neil Burgess**[2]
jesse.geerts.14@ucl.ac.uk, stachenfeld@google.com, n.burgess@ucl.ac.uk
[1]Sainsbury Wellcome Centre, [2] Institute of Cognitive Neuroscience, University College London; [3]DeepMind
London, UK

Abstract

The effectiveness of Reinforcement Learning (RL) depends on an animal's ability to assign credit for rewards to the appropriate preceding stimuli. One aspect of understanding the neural underpinnings of this process involves understanding what sorts of stimulus representations support generalisation. The Successor Representation (SR), which enforces generalisation over states that predict similar outcomes, has become an increasingly popular model in this space of inquiries. Another dimension of credit assignment involves understanding how animals handle uncertainty about learned associations, using probabilistic methods such as Kalman Temporal Differences (KTD). Combining these approaches, we propose using KTD to estimate a distribution over the SR. KTD-SR captures uncertainty about the estimated SR as well as covariances between different long-term predictions. We show that because of this, KTD-SR exhibits partial transition revaluation as humans do in this experiment without additional replay, unlike the standard TD-SR algorithm. We conclude by discussing future applications of the KTD-SR as a model of the interaction between predictive and probabilistic animal reasoning.

Keywords: Reinforcement Learning; Successor Representation; Kalman Filter; Transition Revaluation

Introduction

An impressive signature of animal behavior is the capacity to flexibly learn relationships between the environment and reward. One approach to understanding this behavior involves investigating how the brain represents different stimuli such that credit for reward is generalised appropriately. Predictive representations, like the Successor Representation (SR) (Dayan, 1993), generalise over stimuli that predict similar futures and can provide a useful balance between efficiency and flexibility (Gershman, 2018; Russek, Momennejad, Botvinick, & Gershman, 2017). SR learning is faster to adapt to change than model-free (MF) learning, particularly changes in reward location, and supports more efficient state evaluation than model-based (MB) algorithms, which use time-consuming forward simulations to evaluate state. Since this efficiency depends on caching long-term expected state occupancies, however, the SR is worse than MB at handling changes in the environment's transition structure. In neuroscience and psychology, the SR offers a compelling explanation for a range of behavioural and neural findings (Momennejad et al., 2017; Stachenfeld, Botvinick, & Gershman, 2017; Gardner, Schoenbaum, & Gershman, 2018; Garvert, Dolan, & Behrens, 2017).

While the SR offers a solution to some of the shortcomings of model-free learning, existing methods for estimating the SR, such as temporal difference (TD) learning, do not take into account uncertainty. Here, we attempt to rectify this by drawing on the Kalman TD (KTD) method for value learning (Geist & Pietquin, 2010), which explains a range of animal conditioning phenomena that standard TD cannot explain (Gershman, 2015). KTD-SR gives the agent an estimate of its uncertainty in the SR as well as the covariance between different entries of the SR. We show how this augments the SRs capacity to support revaluation following changes in transition structure.

Results

The successor representation

We define an RL environment to be a Markov Decision Process consisting of *states* s the agent can occupy, *transition probabilities* $T_\pi(s'|s)$ of moving from state s to states s' given the agent's policy $\pi(a|s)$ over actions a, and the reward available at each state, for which $R(s)$ denotes the expectation. An RL agent is tasked with finding a policy that maximises its expected discounted total future reward, or *value*:

$$V(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \,|s_0 = s \right] \tag{1}$$

where t indexes timestep and γ, where $0 \leq \gamma < 1$, is a discount factor that down-weights distal rewards.

The value function can be decomposed into a product of the reward function R and the SR matrix M (Dayan, 1993):

$$V(s) = \sum_{s'} M\left(s, s'\right) R\left(s'\right) \tag{2}$$

M is defined such that each entry $M(s,s')$ gives the expected discounted future number of times the agent will visit s' from starting state s, under the current policy (Dayan, 1993):

$$M(s,s') = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \, \mathbb{I}(s_t = s') | s_0 = s \right] \tag{3}$$

where $\mathbb{I}(s_t = s') = 1$ if $s_t = s'$ and 0 otherwise. Each row $M(s,:)$ in this matrix constitutes the SR for some state s, thus representing each state as a vector over future "successor states." Factorising value into an SR term and a reward term permits greater flexibility because if one term changes, it can be relearned while the other remains intact (Dayan, 1993; Gershman, 2018).

We first consider the SR in a tabular setting with deterministic transitions and a fixed, deterministic policy. This means that there is only one possible state s_{t+1} following any predecessor state s. In this setting, the SR matrix rows of two temporally adjacent states s_t, s_{t+1} can be recursively related as follows:

$$M(s_t,:) = \boldsymbol{\phi}(s)^T + \gamma M(s_{t+1},:), \tag{4}$$

where $\boldsymbol{\phi}(s)$ is the feature vector (of length n, the number of features) observed by the agent in state s. In this article, we consider problems with discrete state spaces, for which the feature vector $\boldsymbol{\phi}(s)$ is a one-hot vector with an entry for every state and a 1 only in the s^{th} position. Equation 4 is analogous to the Bellman equation for value widely used in RL (Sutton & Barto, 1998), with the vector-valued $M(s_t,:)$ in lieu of scalar $V(s_t)$.

We can express the estimated current one hot state vector (based on the SR) as the difference between two successive SRs:

$$\begin{aligned}\hat{\boldsymbol{\phi}}(s_t) &= \hat{M}(s_t,:)^T - \gamma\hat{M}(s_{t+1},:)^T \\ &= \hat{M}^T\boldsymbol{\phi}(s_t) - \gamma\hat{M}^T\boldsymbol{\phi}(s_{t+1}) \\ &= \hat{M}^T\boldsymbol{h}_t \end{aligned} \tag{5}$$

where we have defined $\mathbf{h}_t = \boldsymbol{\phi}(s_t) - \gamma\boldsymbol{\phi}(s_{t+1})$: the discounted temporal difference between state features. The (vector valued) successor prediction error, used to update the SR in TD methods, is then given by $\boldsymbol{\delta}_t = \boldsymbol{\phi}(s_t) - \hat{\boldsymbol{\phi}}(s_t)$.

Learning a probabilistic SR using a Kalman Filter

The algorithm described above produces a point estimate of the SR. While useful for approximating expected value, it is not capable of expressing certainty in these estimates. In order to derive a probabilistic interpretation of the SR, we assume that the agent has an internal generative model of how sensory data are generated from the SR parameters that can be learned with KTD (Geist & Pietquin, 2010; Gershman, 2015). This model consists of a *prior distribution* on the (hidden) parameters, $p(\boldsymbol{m}_0)$ – where $\mathbf{m}_t = vec(M_t^T)$ is the SR reshaped into a vector – an *evolution process* on the parameters, $p(\boldsymbol{m}_t|\boldsymbol{m}_{t-1})$, and a distribution of observed (one-hot) feature vectors given the current parameters and observations $p(\boldsymbol{\phi}_t|\boldsymbol{m}_t,\boldsymbol{h}_t)$. As with earlier work on KTD, we assume a Gaussian model: $\boldsymbol{m}_0 \sim \mathcal{N}\left(\mathbf{0}, C_{0|0}\right)$, $\boldsymbol{m}_t \sim \mathcal{N}(\boldsymbol{m}_{t-1}, C_{v_t})$ and $\boldsymbol{\phi}_t \sim \mathcal{N}\left(\hat{\boldsymbol{\phi}}_t, C_{n_t}\right)$, where $C_{0|0}$ is the prior covariance between SR matrix entries, C_{v_t} is the process covariance, describing how the evolution of different parameters covaries, and C_{n_t} is the observation covariance, describing covariance in the observations. $C_{0|0}$, C_{v_t} and C_{n_t} are set by the practitioner (see Table 1).

The purpose of the Kalman Filter is to infer a posterior distribution over that hidden state $\boldsymbol{m}_t$ given the observations $\boldsymbol{\phi}$:

$$p(\boldsymbol{m}_t|\boldsymbol{\phi}_{1:t}) \propto p(\boldsymbol{\phi}_{1:t}|\boldsymbol{m}_t)p(\boldsymbol{m}_t) \tag{6}$$

Under the Gaussian model described above, this posterior distribution is Gaussian with mean $\boldsymbol{m}_t$ and covariance C_t parameters which will be estimated by the Kalman Filter. To set up the filter, we specify an *evolution equation* describing how the hidden parameters (the SR) evolve over time and an *observation equation* describing how observation relates to our hidden parameters. These two equations comprise the *state-space formulation* for KTD SR:

$$\begin{cases}\mathbf{m}_t = \mathbf{m}_{t-1} + \boldsymbol{v}_t & \text{(evolution equation)} \\ \boldsymbol{\phi}(s_t) = (\mathbf{h}_t \otimes I)^T\mathbf{m}_t + \boldsymbol{n}_t & \text{(observation equation)}\end{cases} \tag{7}$$

where $\boldsymbol{v}_t$ is the *process noise* and $\boldsymbol{n}_t$ the *observation noise*, $\otimes$ denotes the Kronecker product and I the identity matrix. We will start from the assumption that the process noise is white, meaning that $\mathbb{E}[\boldsymbol{m}_t] = \boldsymbol{m}_{t-1}$, i.e. the expected mean SR on time t equals the estimated SR on time $t-1$.

The Kalman Filter keeps track of the mean $\boldsymbol{m}_t$ and covariance C_t of the posterior (6). At each timestep, the parameters of the posterior are updated using the Kalman Filter equations:

$$\hat{\mathbf{m}}_{t|t} = \hat{\mathbf{m}}_{t|t-1} + K_t(\boldsymbol{\phi}_t - \hat{\boldsymbol{\phi}}_t) \tag{8}$$

$$C_{t|t} = C_{t|t-1} - K_t C_{\boldsymbol{\phi}_t} K_t^T \tag{9}$$

$$K_t = C_{\mathbf{m}\boldsymbol{\phi}_t} C_{\boldsymbol{\phi}_t}^{-1} \tag{10}$$

where $C_{\mathbf{m}\boldsymbol{\phi}_t}$ is the covariance between the parameters and the prediction error, and $C_{\boldsymbol{\phi}_t}$ is the covariance of the prediction error. The notation $C_{t|t} = \mathbb{E}\left[C_t|\boldsymbol{\phi}_1...\boldsymbol{\phi}_t\right]$ means that the estimate of the parameter covariance is conditioned on all observations until time t (see Geist & Pietquin, 2010). Importantly, and in contrast to standard TD updates for the SR (Dayan, 1993), the Kalman gain K_t is stimulus specific (it is a matrix of number of SR entries by number of features) and dependent on the ratio between covariance in the parameters and covariance in the observations, allowing for a principled weighting of prior knowledge and incoming data. See Algorithm 1 for a full description of the method, including how these quantities are computed.

In summary, we have introduced a method of handling uncertainty over SR estimates. This allows for an efficient combination of prior knowledge and incoming information when updating the SR estimates. Furthermore, it allows us to estimate dependencies between different entries in the SR that inform SR updates. This permits non-local updates which, in the case of KTD for value estimation, have proven to better explain animal behaviour than the strictly local updates of vanilla TD (Gershman, 2015). We explore a possible role for non-local updates in the following section.

Partial Transition Revaluation Simulations

A key prediction of standard TD-SR learning is that "reward revaluation" should be supported while "transition revaluation" should not. Momennejad et al. (2017) tested this in humans. In the first phase of their experiment, participants learned two different sequences of states terminating in different reward amounts: 2→4→6→\$1 and 1→3→5→\$10 (see Figure 1B). In the next stage, half of the participants were exposed to the transition revaluation condition, observing novel 4→5→\$10

Algorithm 1: Kalman TD Successor Representation

```
Initialization: priors $\mathbf{m}_{0|0}$ and $C_{0|0}$ ;
for $t \leftarrow 1, 2, \ldots$ do
    Observe transition $(s_t, s_{t+1})$ ;
    Prediction step;
    $\hat{\mathbf{m}}_{t|t-1} = \hat{\mathbf{m}}_{t-1|t-1}$ ;
    $C_{t|t-1} = C_{t-1|t-1} + C_{v_t}$ ;
    Compute statistics of interest ;
    $\hat{\boldsymbol{\phi}}(s_t) = (\mathbf{h}_t \otimes I)^T \hat{\mathbf{m}}_t$ ;
    $C_{\mathbf{m}\phi_t} = C_{t|t-1}(\mathbf{h}_t \otimes I)$ ;
    $C_{\phi_t} = (\mathbf{h}_t \otimes I)^T C_{t|t-1}(\mathbf{h}_t \otimes I) + C_{n_t}$ ;
    Correction step ;
    $K_t = C_{\mathbf{m}\phi_t} C_{\phi_t}^{-1}$ ;
    $\hat{\mathbf{m}}_{t|t} = \hat{\mathbf{m}}_{t|t-1} + K_t(\boldsymbol{\phi}_t - \hat{\boldsymbol{\phi}}_t)$ ;
    $C_{t|t} = C_{t|t-1} - K_t C_{\phi_t} K_t^T$
end
```

Table 1: Parameter values

Name	Symbol	Value
Discount factor	γ	0.9
Process covariance	C_{v_t}	$(1 \times 10^{-3})I$
Observation covariance	C_{n_t}	I
Prior covariance	$C_{0\|0}$	$0.1I$
Prior SR	$\boldsymbol{m}_{0\|0}$	$vec(I)$
Rescorla Wagner learning rate	α_r	0.1
Number of trials per phase	N	50

and 3→6→$1 transitions. The other half experienced "reward revaluation" in the form of novel reward amounts 6→$10 and 5→$1 (Figure 1A). Importantly, the novel experiences start from intermediate states such that transitions from 1 or 2 are not seen following phase 1. While participants were significantly better at reward revaluation than transition revaluation, they were capable of some transition revaluation as well (Figure 1C). Accordingly, the authors proposed a hybrid SR model: an SR-TD agent that is also endowed with capacity for replaying experienced transitions (Figure 1F). This permits updating of the SR vectors of states 1 and 2 through simulated experience.

Here, we simulate this experiment and find that the probabilistic KTD-SR accounts for partial transition revaluation even without replay (Figure 1D). KTD-SR correctly learns the SR matrix after phase 1 (Figure 1E) as well as an estimate of the covariance between all entries in the SR matrix, $C_{t|t}$. Unlike TD-SR, KTD-SR uses the covariance matrix to estimate the Kalman gain and uses that to update the whole matrix. This means that after seeing $3 \rightarrow 6$, it updates not just $M(3,:)$ but also $M(1,6)$ because these entries have historically covaried (same for $M(4,:)$ and $M(2,5)$) (Figure 1F). To estimate direct reward $\hat{r}$, the agent uses a Rescorla-Wagner rule (Rescorla & Wagner, 1972). Model parameters are listed in Table 1 and experimental parameters are kept the same as in (Momennejad et al., 2017).

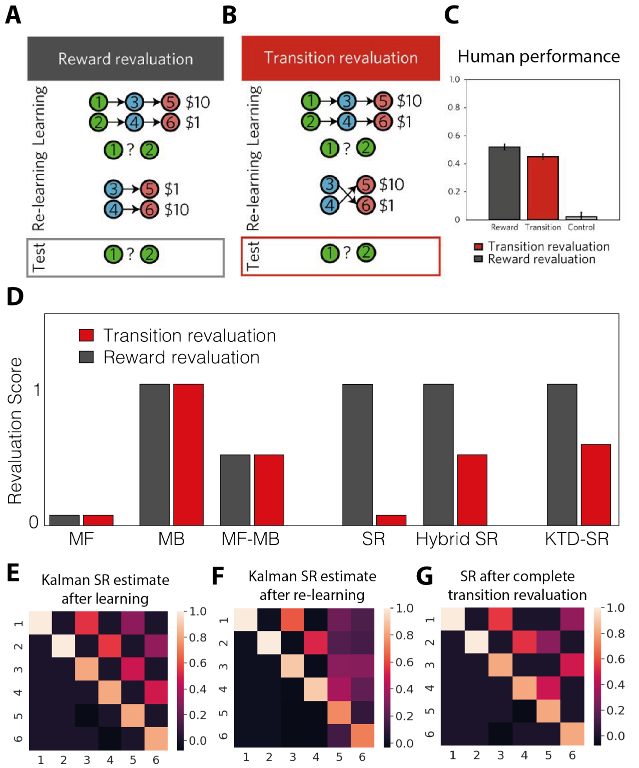

Figure 1: KTD SR performance on a transition and reward revaluation experiment. (A-B) Task structure for (A) reward revaluation and (B) transition revaluation experiments. (C) Human performance on transition and reward revaluation tasks. (D) Model predictions for classic model-free, model-based or a hybrid of model free and model-based algorithms, TD-SR, hybrid SR and KTD-SR. (E-G) The SR matrix estimated by KTD after (E) learning (phase 1), (F) re-learning (phase 2) and (G) after a hypothetical complete transition revaluation. Panels A–C reprinted with permission from Momennejad et al. (2017).

Discussion

The SR constitutes a middle ground between model-based and model-free RL algorithms by separating reward representations from cached long-run state predictions. Here we learn a probabilistic SR model using KTD that supports principled handling of uncertainty about state predictions and interdependencies between these predictions. We exploit this feature to show that, unlike standard TD-SR, KTD-SR can perform partial transition revaluation. In later work, we plan to test our model on other tasks that could benefit from KTD-SR in a similar way, such as policy revaluation (a well-known weak spot of TD-SR; Barreto, Munos, Schaul, & Silver, 2016).

We note the relative strengths and weaknesses of KTD-SR when compared to a hybrid-MB-SR approach. Replay requires a buffer to store experienced episodes and a sufficient number of replays that information is propagated throughout the SR model. While KTD-SR can incorporate information about long-range in a single update, it must learn and store a large $n^2 \times n^2$ matrix (although dimensionality reduction can reduce this burden; Fisher, 1998). There is compelling evidence in favor of both replay (Carr, Jadhav, & Frank, 2011; Ólafsdóttir, Bush, & Barry, 2018) and probabilistic representations (Ma, Beck, Latham, & Pouget, 2006) driving behavior. Future work will consider how the relative tradeoffs of these approaches constrain hypotheses.

Probabilistic models provide a number of advantages for RL in terms of optimal credit assignment (Kruschke, 2008), uncertainty-minimising exploration (Dearden, Friedman, & Russell, 1998), arbitration between competing models (Daw, Niv, & Dayan, 2005). Distributional RL-trained neural network agents achieve state of the art performance (Bellemare & Dabney, 2017). Furthermore, a range of animal learning findings suggest that animals are capable of probabilistic reasoning (Gershman, 2015; Kruschke, 2008; Courville, Daw, & Touretzky, 2006). Future work will involve exploring these advantages in the context of SR learning (Gardner et al., 2018).

We make several assumptions in order to make this model tractable. The Gaussian assumption is clearly violated in the case of one-hot state vectors (i.e. neither ϕ nor M should have negative entries). However, the model is sufficiently expressive that a good approximation can still be found, and a "successor feature" model could be applied over arbitrary features for which the Gaussian assumption might hold. The random walk process noise is useful for capturing slow changes in the environment, but might be ill-suited for step changes or sub-optimal when the dynamics are predictable. While we assume deterministic transitions and linear function approximation in this work, it is straightforward to extent the model to stochastic transitions and nonlinear function approximation with a "coloured noise" approach (Geist & Pietquin, 2010).

Acknowledgments

This work is funded by the Gatsby Foundation and the Wellcome Trust. We thank Samuel Gershman, Talfan Evans, Eszter Vértes, Steven Hansen and Matthew Botvinick for helpful comments and suggestions.

References

Barreto, A., Munos, R., Schaul, T., & Silver, D. (2016). Successor Features for Transfer in Reinforcement Learning. *arXiv*, 1–13.

Bellemare, M. G., & Dabney, W. (2017). A Distributional Perspective on Reinforcement Learning.

Carr, M. F., Jadhav, S. P., & Frank, L. M. (2011). Hippocampal replay in the awake state: A potential substrate for memory consolidation and retrieval. *Nature Neuroscience*, *14*(2), 147–153.

Courville, A. C., Daw, N. D., & Touretzky, D. S. (2006). Bayesian Theories of Conditioning in a Changing World. *Trends in Cognitive Sciences*, *10*(7), 294–300.

Daw, N. D., Niv, Y., & Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience*, *8*(12), 1704–1711.

Dayan, P. (1993). Improving Generalisation for Temporal Difference Learning: The Successor Representation. *Neural Computation*, *5*(4), 613–624.

Dearden, R., Friedman, N., & Russell, S. (1998). Bayesian Q-learning. *AAAI/IAAI*.

Fisher, M. (1998). *Development of a simplified Kalman filter*. European Centre for Medium-Range Weather Forecasts.

Gardner, M. P., Schoenbaum, G., & Gershman, S. J. (2018). Rethinking Dopamine as Generalized Prediction Error. *Proceedings of the Royal Society B: Biological Sciences*, *285*(1891).

Garvert, M. M., Dolan, R. J., & Behrens, T. E. (2017). A Map of Abstract Relational Knowledge in the Human Hippocampal-Entorhinal Cortex. *eLife*, *6*, 1–20.

Geist, M., & Pietquin, O. (2010). Kalman Temporal Differences. *Journal of Artificial Intelligence Research*, *39*, 483–532.

Gershman, S. J. (2015). A Unifying Probabilistic View of Associative Learning. *PLoS Computational Biology*, *11*(11), 1–21.

Gershman, S. J. (2018). The Successor Representation: Its Computational Logic and Neural Substrates. *The Journal of Neuroscience*, *38*(33), 7193–7200.

Kruschke, J. K. (2008). Bayesian Approaches to Associative Learning: From Passive to Active Learning. *Learning and Behavior*, *36*(3), 210–226.

Ma, W. J., Beck, J. M., Latham, P. E., & Pouget, A. (2006). Bayesian Inference With Probabilistic Population Codes. *Nature Neuroscience*, *9*(11), 1432–1438.

Momennejad, I., Russek, E. M., Cheong, J. H., Botvinick, M. M., Daw, N. D., & Gershman, S. J. (2017). The Successor Representation in Human Reinforcement Learning. *Nature Human Behaviour*, *1*(9), 680–692.

Ólafsdóttir, H. F., Bush, D., & Barry, C. (2018). The Role of Hippocampal Replay in Memory and Planning. *Current Biology*, *28*(1), R37-R50.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical conditioning II: Current research and theory*, *2*, 64–99.

Russek, E. M., Momennejad, I., Botvinick, M. M., & Gershman, S. J. (2017). Predictive Representations Can Link Model-Based Reinforcement Learning to Model-Free Mechanisms. *PLoS Computational Biology*, 1–42.

Stachenfeld, K. L., Botvinick, M. M., & Gershman, S. J. (2017). The Hippocampus as a Predictive Map. *Nature Neuroscience*, *20*(11), 1643–1653.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement Learning: an Introduction*. MIT Press.