Available online at www.sciencedirect.com

ScienceDirect

Current Opinion in Behavioral Sciences

Learning Structures: Predictive Representations, Replay, and Generalization

Ida Momennejad

Memory and planning rely on learning the structure of relationships among experiences. Compact representations of these structures guide flexible behavior in humans and animals. A century after 'latent learning' experiments summarized by Tolman, the larger puzzle of cognitive maps remains elusive: how does the brain learn and generalize relational structures? This review focuses on a reinforcement learning (RL) approach to learning compact representations of the structure of states. We review evidence showing that capturing structures as predictive representations updated via replay offers a neurally plausible account of human behavior and the neural representations of predictive cognitive maps. We highlight multi-scale successor representations, prioritized replay, and policy-dependence. These advances call for new directions in studying the entanglement of learning and memory with prediction and planning.

Address
Columbia University, United States

Current Opinion in Behavioral Sciences 2020, **32**:155–166

This review comes from a themed issue on **Understanding memory: Which level of analysis?**

Edited by **Morgan Barense** and **Hugo J Spiers**

For a complete overview see the Issue and the Editorial

Available online 5th May 2020

https://doi.org/10.1016/j.cobeha.2020.02.017

Introduction

As we navigate the world we learn and update the relational structure of experienced events. The idea that relational representations are used as internal maps for navigation and planning was popularized by Tolman, as 'cognitive maps' [1]. Early theories of cognitive maps localized them in the hippocampus of various species and proposed that these maps were spatial, allocentric, and Euclidean in nature [2]. However, these theories did not provide a learning mechanism that could explain how cognitive maps represent the structure of the environment. In the decades that followed, a wealth of neural findings provided pieces to this larger puzzle. Place cells signalled the current location of an animal, entorhinal grid cells tiled spaces into grids, and the environment was further summarized in representations that each served a purpose: boundary vector cells, head direction cells, reward cells, object vector cells, etc [3]. It was also shown that simulated experience during offline replay can update cognitive maps [4], e.g., piecing together current experience with memory of past experiences to make inferences about unseen links or integrating structural knowledge and rewards to update action policies [5••,6••]. Some of these findings contradict predictions of earlier cognitive map theories. For instance, inconsistent with merely spatial structures, place and grid fields capture non-spatial state spaces as well [7•,8,9]. Counter to a merely Euclidean representation, it has been shown that place field representations are path-dependent [10] and skew asymmetrically toward goal locations [11], and sequential trajectories to goal locations twist around obstacles [10,12••,13,14]. Furthermore, entorhinal grid fields are shown to capture principle components or basis sets of state spaces [15••,16], and these grid fields can get over-represented or warped near the locations of goals and rewards [17,18,89] (Section 3).

The disagreement of some findings with earlier neural theory poses a larger puzzle: how does the brain learn and update cognitive maps and how do they represent and generalize structures? Solving the cognitive map puzzle calls for a theoretical and computational framework that can capture decades of cumulative evidence, is biologically plausible, and makes testable behavioral and neural predictions. Various theoretical proposals have attempted to solve the puzzle with computational frameworks such as manifolds and neural networks [19,20], topological models [21,22], relational learning [23], and reinforcement learning [15••]. While these frameworks are not mutually exclusive, this review focuses on representation learning using a reinforcement learning (RL) framework to solve the cognitive map puzzle by learning structures of the state spaces (spatial or non-spatial).

Representation learning in RL refers to learning the structure of a state space. The states can be spatial locations, experimental stimuli, associated memory items, or task states [24]. The RL framework can be used to learn mappings between observed perceptual features and states [25], compact representations of relational and associative structures of the state space [15••,23,26••], as well as abstractions enabling transfer between tasks and environments [27,28]. As discussed later, while in classic model-based RL learning the structure of states implies learning the probabilities of transition between adjacent states [29], other RL approaches can learn compact representations of

the multi-step and multi-scale structure of the environment [5••,15••]. This review is focused on learning compact representations of the structure of states that are multi-step, multi-scale, and path-dependent using successor representations and replay in the RL framework.

In what follows, we briefly review how representation learning and replay help acquire multi-scale predictive cognitive maps (Section 2), how predictive representations are learned and the role of policy-dependence in learning representations (Section 3), how the content and prioritization of memory replay updates these representations (Section 4), and emerging computational directions in generalization and transfer (Section 5). In short, learning predictive representations of structures using policy-dependent SR and replay accounts for neural evidence (path-dependent and non-spatial) that disagrees with earlier (euclidean and spatial) cognitive map theory. This calls for further work on learning compact and generalized representations of structures and updating our notions of cognitive maps.

Learning predictive representations of structures

The reinforcement learning problem is typically that of an agent flexibly learning paths to reward and avoiding death in a dynamic environment [30]. Many RL problems require planning over long time scales, latent learning (used by Tolman to support the idea of a cognitive map), and inference in complex environments with sparse rewards. Planning in such problems requires remembering the relational structure of the environment. While RL is famous for reward-based learning, it offers a framework for learning representations in the absence of rewards. There are different notions of learning and memory for structures. You can remember the specific structure of your childhood home, or the abstract structure of rooms in any house. Here we review evidence that the RL framework can capture both notions and their neural implementations.

Why use the RL framework to study how biological agents learn structures? There are at least two reasons for using RL for structure learning. First, RL is biologically plausible and offers testable hypotheses about the neural implementation of structure learning and their correspondence to behavior. This is an advantage over Bayesian cognitive models that have been more commonly used to study structure learning and causal inference, but cannot offer an adequate account of neural implementation [31]. Second, while model-free RL is more broadly known, RL representation learning principles can acquire compact and predictive representations of structures, even in the absence of rewards. Moreover, the eigendecomposition of predictive representations supports a powerful abstraction of memory for structures: basis sets for compositional representations. This review highlights how representation learning using the RL framework can account for learning structures, generalization, and transfer, as well as their neural implementation.

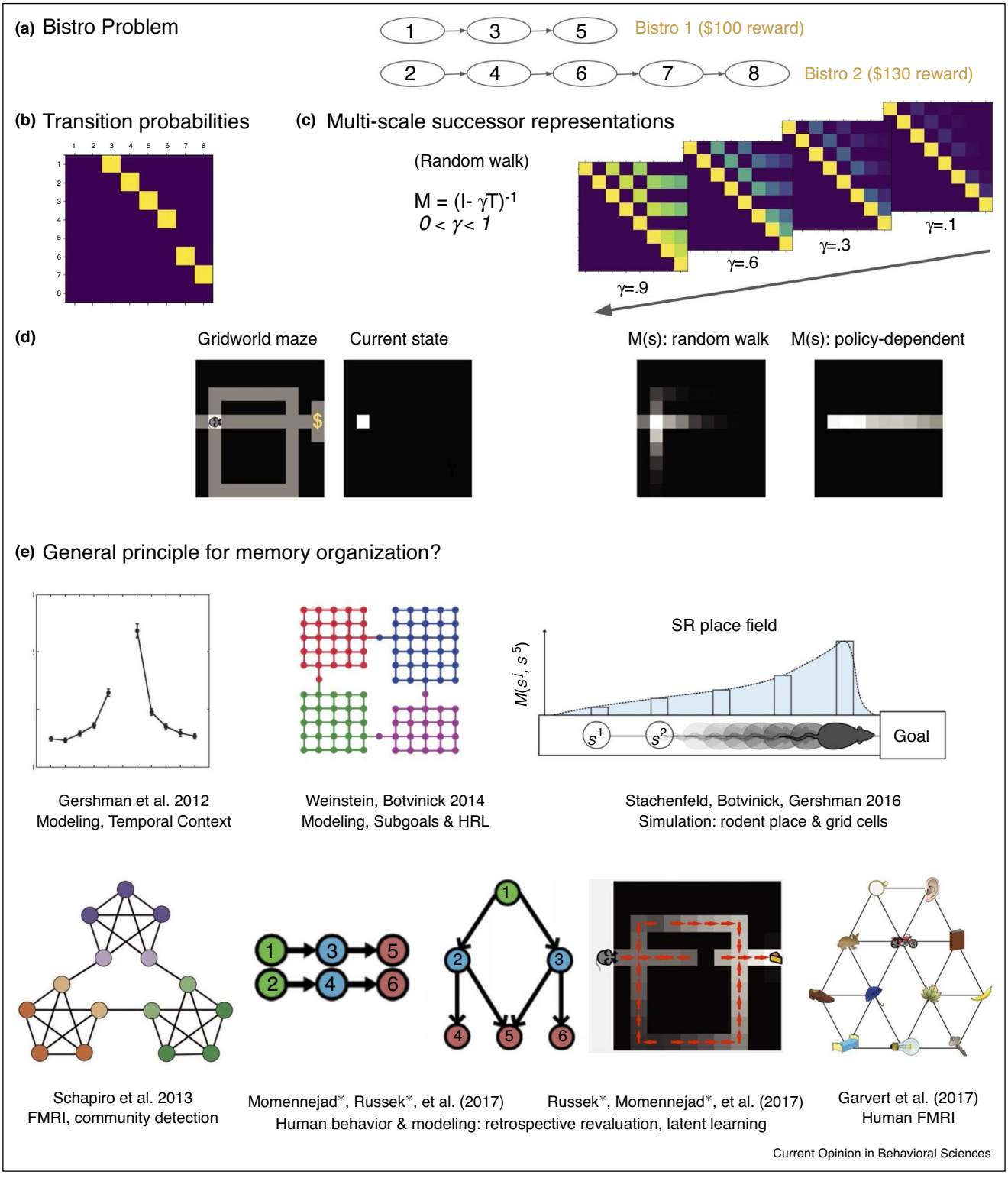
Let us briefly review two RL agents prominent in cognitive neuroscience: model-free and model-based RL, and why they do not offer satisfactory solutions to the cognitive map puzzle. A model-free agent (MF) takes actions, observes the running average of outcomes, and stores (caches) the discounted expected value of actions. This cached value can be learned via temporal difference (TD) learning and prediction errors. MF agents act fast and are computationally cheap, but have no memory of structures or the relationship among states; they only store the expected value of actions for every given state (see Equation 1). Model-based agents (MB), on the other hand, store a relational representation of the environment in terms of the probabilities of 1-step transitions between states. This model is later used to compute action policies that maximize expected value. Classic MB agents learn a representation of relationships between states that are one step apart. This is formalized as a one-step Transition matrix denoted as T. When faced with a decision, MB iteratively unfolds consecutive states taking mental actions, one step at a time, in order to simulate different trajectories and compute and compare their expected value. Thus, MB agents can flexibly compute expected value (long-term cumulative reward) for each possible sequence, but do so at a high computational cost incurred by iterations [29].

In short, MF agents are fast and cheap but have no knowledge of structures and are hence inflexible. MB agents are flexible but using their representation of structures requires iterations that are computationally expensive – even intractable for realistically large decision trees [32••]. Therefore, neither can sufficiently capture flexible and efficient use of cognitive maps in mammals. An alternative RL approach offers a computationally efficient intermediate solution between MB and MF: the flexibility of MB behavior without the intractable costs. This is the successor representation (SR) [26••,32••,33••]. SR enables representation learning and uses abstractions to acquire relationships between a state and all its successor states, multiple steps away (Figure 1C). Note that this representation captures non-adjacent dependencies, unlike model-based RL's one-step probability matrix (Figure 1B). Importantly, combining SR with learning from replay enables the agent to precompute multi-step and multi-scale dependencies offline, increasing planning efficiency when faced with a decision later on [26••,32••].

What is the successor representation and how is it learned?

If an environment has n states, think of SR as a *nXn* matrix of state-state relations. SR captures relational structures by learning expected future visitations among states. For instance, consider the 2nd row of SR matrices in Figure 1C: the expected future visitations from state 2 to all other states are captured within each predictive

Figure 1



Successor representation and empirical evidence. (A) Illustration of the bistro problem [5••], an RL problem where there are two bistros and the further bistro is more rewarding. The agent needs to choose between state 1 and 2. The environment is represented as a graph of states and associated rewards. (B) The corresponding 1-step transition matrix (T) of the Bistro problem. Computing value using T requires iteration. (C) Multi-scale successor representations of the Bistro environment across a number of scales. Note that these are policy-independent or random-walk SRs. Value can be computed by a linear product of an SR matrix with a vector of rewards. Since no iteration is required, this is computationally

horizon. This predictive horizon or scale of SR, i.e., the furthest successor state that is 'visible' from every starting state, depends on a discount parameter ($0 < \gamma < 1$).

SR can be gradually learned via temporal difference (TD) learning of the counts of visits among states and their successor states within a given horizon (Equation 2) [26••]. For an intuitive understanding, recall that TD learning can be used for model-free learning of cached value using reward prediction errors (Equation 1). SR can be learned using successor prediction errors (Equation 2). Note that while MF learning caches discounted expected future value (running average of rewards), SR learning caches discounted expected future visitations from one state to another state (mean count visits). We address SR learning in more detail in Section 3.

In short, the successor representation is a count-based compact representation: it does not store transition probabilities (which are < = 1, Figure 1B) like MB agents, but the mean discounted counts of future visitations (can be >1, Figure 1C and D). Consider again the SR in Figure 1C. The 4th **row** of the learned SR matrix (M) stores the mean discounted number of expected future visitations to any successor state of 4, in the **columns** of M (4). Therefore, when an agent is in state 4, the successor representation of state 4 is not only one state (i.e., not merely a matrix cell), but the entire *4th row* of the SR matrix. This is why the structure of SR is inherently predictive: it offers predictive representations that can solve inference and planning, which offers predictions about the structure of underlying neural representations.

Empirical evidence for the successor representation

A series of empirical and computational studies have tested and found support for the proposal that relational maps are organized in memory according to the principles of the successor representation [15••,26••,32••,37•]. Human behavioral evidence comes from two studies specifically designed to compare the predictions of MF, MB, SR, and hybrid models to human performance on varieties of retrospective revaluation (reward revaluation, transition revaluation, policy revaluation). Human behavior was best captured by a model that combined SR and replay, SR-Dyna. Sr-Dyna learns successor representations via both direct and replayed experience [32••] (see Section 4). A normative study [26••] showed that the SR-Dyna class of algorithms outperform other normative models. It has also been shown that SR helps discover subgoals for planning in hierarchical reinforcement learning (HRL) problems [36]. SR-Dyna could be applied to the problem of building models that discover hierarchical structures more efficiently via offline replay.

Capturing structures and spectral clustering

The successor representation has theoretically rich properties. In linear algebra, dimensionality reduction, and graph theory, the SR matrix is mathematically equivalent or related to concepts such as the matrix dissolvant, fundamental matrix, communicability distance, and the inverse of the graph Laplacian. The graph Laplacian is computed by subtracting a graph's adjacency matrix, where there is a 1 for an edge between nodes i and j, from its degree matrix, a di (see Equation 3, the graph Laplacian is shown the be captured by I-T). In graph theory, the eigenvectors of the graph Laplacian are widely used across the sciences for spectral clustering, graph diffusion, and community detection [38]. Spectral clustering detects communities of connected nodes based on their edges. Such community detection can serve abstraction and detection of efficient paths between any two nodes. Spectral clustering is possible thanks to the eigenvalues (spectrum) of the graph Laplacian. Intuitively, eigenvectors of the Laplacian form a compressed representation of the structure of graph, or the relational space. Matrix compression can be applied to achieve compact and abstract representations by setting eigenvector components with small eigenvalues (less influence) to zero, hence ignoring them. If we consider the state space a graph of connected states, the SR approximates the inverse of the graph laplacian (Equation 3). Mathematically, the eigenvectors of SR tile the state space into grid-like representations. The linear combination of these grids can help solve problems involving the partitioning and chunking of experience, subgoal discovery, graph diffusion, and communicability in graphs and complex networks [39]. These properties lend well to understanding hippocampal cognitive maps, discussed next.

Neural implementation of the successor representation

How is the SR implemented in the brain? What is the correspondence between learned SR and neural

(Figure 1 Legend Continued) more efficient than MB value computation. The interesting property of the bistro problem is that an SR with a scale of .1 does not capture the relationship between states that are connected but furthest apart, i.e., states 2 and 8. When combined with the reward vector, a single SR matrix with a small predictive horizon may fail to recognize state 2 as the optimal starting state leading to state 8, with the highest reward. (D) A maze similar to those used in rodent experiments is depicted as a grid-world, where every location corresponds to a square [26••]. The RL problem is to navigate to a reward location, marked with a $ sign on the right side. Assume that an agent is in the location depicted in the second matrix from the left. Two possible successor representations for this state are depicted to the right: Random-walk SR and policy-dependent SR. Random-walk SR represents the structure of the maze regardless of the current goal. Policy-dependent SR over-represents the states on the trajectory to the current goal. In both cases, the SR for state *s, M(s),* activates the representation of successor states within a given predictive horizon of the current state. (E) Computational support and empirical evidence from rodents and humans behavior, fMRI, and electrophysiology. The successor representation may offer a general principle for the organization of memory representations. (Figures in E are reprinted from cited papers [15••,26••,32••,34,35,36,37•]. Note that these are schematic summaries and readers should refer to the original papers for detail.)

representation? Recall that the *s*th **column** of SR represents the *predecessors* of state *s*, i.e., what predicts this current state, its past. Now when the agent is in state *s*, the *s*th **row** of SR is the cached representation of the discounted *successors* of state *s*: what state *s* predicts, its future. This idea makes predictions about the neural similarity relationships among different states.

Let us return to hippocampal cognitive maps, and consider place fields. A place field is activated when the agent is in the field's preferred location or is approaching it within a given horizon. A recent study compared rodent hippocampal and entorhinal electrophysiology results with SR, and shows that SR's columns simulate place fields. Moreover, they showed that SR's eigenvectors look like grid field representations [15••]. Recall that eigenvectors of a matrix are linearly uncorrelated principal axes of the state space, or a smaller state space within which all information in the matrix can be captured as the linear combination of these axes. It has thus been suggested that entorhinal grid cells encode a low-dimensional basis set that extracts the multi-scale structure of predictive representations for hierarchical planning and subgoal processing [16]. Section 5 notes further advances of these topics in generalization and transfer.

There are two other properties of place fields that any sufficient computational account needs to capture. The first is the observation that place fields fire asymmetrically and are skewed towards goal locations [11,15••]. This asymmetry would be expected of predictive representations as well. Since SR-Dyna learns visitation-counts during experience and prioritizes replay of trajectories toward goals, it will learn higher expected future visitations for locations that are expected to be visited more often (Figure 1D, rightmost). This property can capture the asymmetry of place fields, and is related to policy-dependent representations discussed in Section 3. The second property that any account of place fields needs to capture is that they are on average larger along the hippocampal long axis towards the anterior (or ventral) hippocampus. Multi-scale successor representations (MSR), learned simultaneously with different discount parameters and corresponding to different predictive horizons (Figure 1C) can capture this property [5••]. Furthermore, it has been shown that the derivative of the multi-scale ensemble of SR matrices can be used to reconstruct an entire trajectory of future states and account for path-dependent distance to goal representations in the medial temporal lobe [5••].

Can SR's predictions about the structures of neural similarities be tested using functional magnetic resonance imaging (fMRI)? A number of studies have compared hypothesis similarity structures with the similarity among fMRI patterns associated with different states. Consistent with predictions, SR could account for fMRI pattern similarity in statistical learning [35] and learning non-spatial relational concepts [37•]. A recent fMRI study has shown that during planned hierarchical navigation, predictive horizons of small to medium lengths are represented along the long axis of the hippocampus (with longer horizons in anterior hippocampus of humans, corresponding to ventral hippocampus in rodents) and the prefrontal cortex (PFC) hierarchy [40,41]. The largest horizons were represented in gradually more anterior regions of the rostral and orbital prefrontal cortex, corresponding to Brodmann areas 10 and 11 [40,41](Figure 2). As noted in Sections 3 and 5, prefrontal representations are not merely representations of larger scales. Similar regions have been reported in the representation of prospective tasks in human fMRI [86,87]. Thus, prefrontal and entorhinal representations may compute and represent generalization and basis sets of structures, task sets, and schema [3,23,42].

These and other empirical evidence have been taken to suggest the successor representation as a principle for memory organization and temporal context (Figure 1E) [34]. However, there are a variety of algorithms for learning the structure of state-spaces using the SR, and they generate different behavioral and neural predictions depending on at least three factors. The first is whether SR acquired during goal-directed experience is different from SR computed analytically or with random walks (Section 3). This determines the policy-dependence of the learned representations. The second regards whether and how memory replay updates SR offline, stitching together distal experiences to infer and update unseen relations (Section 4).

Learning structures and policy-dependence

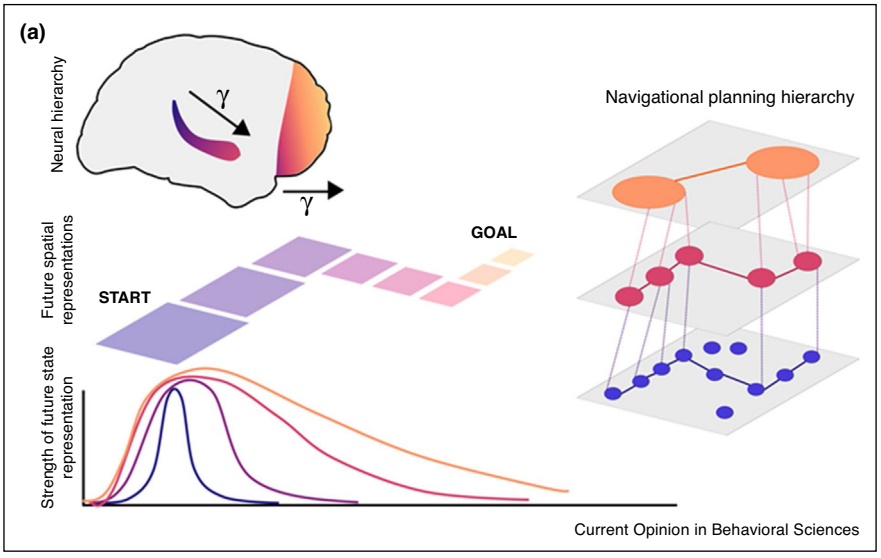
How does an agent learn the graph structure of the state space while navigating it? A temporal difference (TD) learning approach, which is typically used to learn and update state-action values, can be applied to the learning and updating of state-state structures irrespective of rewards. Before showing how, let us recall how TD learning can update cached value in model-free RL in Equation 1:

$$V(s) = V(s) + \alpha(R_{observed} + \gamma V(s_{new}) - V(s))$$

$$Model-free\ RL: TD\ gradually\ updates\ cached\ value\ for\ state\ s.$$

Equation 1 shows how the value of state *s* is gradually learned using a simple value-based TD update [30]: a learning rate is applied to the observed reward it leads to plus the prediction error (i.e., the difference between the discounted value of the next state and the present state). TD learning can also be used to gradually update expected counts of visits from a state to its successors. Equation 2 shows how the SR can be gradually learned

Figure 2



Multiscale successor representations. Successor representations with different predictive horizons from start to goal location are depicted with different colors. Predictive horizons are represented in cold-warm color shades (blue-orange) along the hippocampal long axis as well as the prefrontal cortex (PFC). The horizon of spatial representations are depicted with squares of different sizes: when the agent is in the start state, nearby successor states are represented in blue shades and further successors are in warmer shades. Their size of the spatial tiles capture the discount applied to states further away. Corresponding shades are used in the graph depiction of abstraction across the scales of representation. Such a multi-scale predictive structure in the agent's representations of the environment readily enables hierarchical planning as follows. Planning at larger horizons uses the large scale PFC representations, while fine-grained plans are hierarchically unfolded backward down to the smallest horizon of place fields in posterior hippocampus. Note that prefrontal and entorhinal representations are not merely representations of larger scales, but may compute and represent generalized structures, such as grid fields and schema (see Sections 3 and 5). Reprinted from Brunec, Momennejad (2019)[40].

using a simple *count-based* TD update. Each time the agent starts from a starting state s, and arrives to a new state s_{new}, the count of expected visitations to the new state and its successors are updated. As such, gradually a "successor representation" is learned: a predictive matrix of how often we expect to visit s_{new} and its successors if we are currently in s (Figure 1). TD updates require a discount parameter ($0 < \gamma < 1$) that determines the scale or predictive horizon of the SR, and a learning rate ($0 < \alpha < 1$) applied to the successor prediction error (the difference between the expected successors of s and the discounted successors of the new state), as follows [26••,43••].

$$M(s) = M(s) + \alpha\Big(onehot(s_{new}) + \gamma M(s_{new}) - M(s)\Big)$$

$$Representation\ learning: TD\ gradually\ updates\ the\ success\ or\ representation\ for\ state\ s.$$

Here M denotes the successor representation matrix (or predictive representation) at scale. Here $onehot(s_{new})$ is a vector of all zeros with a 1 for successor state s_{new}. When the agent starts from s and visits state s_{new}, $onehot(s_{new})$ adds one visit to the count of visits to s_{new} in the s'th row of M (learning rates apply). The subtraction of the expected successors of s (the current state) from the discounted successors of state s_{new} (discounted s_{new}th row of M minus the sth row of M) is the "successor prediction error". It is used for temporal difference learning of SR. Gradually, the SR matrix is updated via TD learning. Notably, due to more visitation counts to states that fall on the reward policy, successor states along the trajectory to rewards and bottlenecks gain higher SRs (mean expected discounted visits). Hence, this manner of learning leads to a policy-dependent SR.

As mentioned earlier, a key difference in successor representation learning methods is whether the learning is policy-dependent or policy-independent. Policy-independent learning of a graph structure (e.g., during navigation, see also [91]) is like learning by taking random walks on a graph in all directions (Figure 1). The agent has no reason to prefer a policy that favors one direction or trajectory more than another, and hence the structure of the graph as a whole is learned. If goals or rewards were present, a random walk would emerge while exploring and learning in an open field where rewards are uniformly distributed. Given enough experience, policy-independent or random-policy SR can be

learned directly from the transition matrix as in Equation 3 (Figure 1).

$$M = (I - \gamma T)^{-1} \quad (3)$$

Random-walk policy SR can be computed using the transition matrix

Thus, SR can be either gradually learned or directly computed from an MB learner's transition matrix, T. Note that when rewards are not uniformly distributed across the state space, when there are obstacles, or when the agent visits some states more than others, the successor representation that is gradually learned via TD using Equation 2 no longer converges to the one computed from T in Equation 3. In other words, when the agent's policy is to visit rewarded locations and bottleneck states (such as doors) more often than others, the SR can no longer be computed as if the agent were taking a random walk.

Policy-dependence and path-dependence

An important property of policy-dependent SR is that, consistent with path-dependent representations reflected in behavior and the hippocampus [10], its estimate of distance to goal would be non-Euclidean. This bears other predictions about vectorial and distance to goal representations [44]. Namely, policy-dependent goal-vectors, unlike previous suggestions, would not point to the goal location through a wall that obstructs it, but twist around barriers. This is consistent with recent findings in vectorial goal-representations in the hippocampus of flying bats [12••]. It may also be related to the behavioral finding that circumnavigation led to underestimating travel times and expanding spatial distance [45]. Furthermore, recall that SR can capture place fields and SR's eigenvectors capture grid fields that tile the state space [15••]. These eigenvectors can compress representations, encoding a low-dimensional basis set of predictive representations of the state space [16]. Notably, in some cases the eigenvectors of policy-independent and policy-dependent SR will likely be different, making the prediction that the grid fields of policy-dependent and random-policy SR may be different. Policy-dependent SR values increase as an agent approaches bottlenecks and goal locations. As such, the eigenvectors of the policy-dependent SR may lead to grid fields that over-represent often-visited states, tiling "more space" near highly visited locations [15]. This prediction is consistent with recent evidence showing grid fields are attracted to reward locations [17,18]. Future studies and reviews are required to test and compare these predictions to neural and behavioral findings.

As we will see in the next section, prioritized replay as well contributes to policy-dependence. Even when the number of visitations to all locations are empirically controlled [32••], replay prioritization can simulate goal-related states (e.g., reward location, subgoals and doors, locations on the path to reward) more often than others. Preferential replay of goal-related states leads to higher number of virtual visitations to those states during replay cycles. Since the agent is learning and updating SR during offline replay, a policy-dependent SR map emerges [26••], over-representing goal-directed states even though the agent's experience is controlled.

Learning structures via replay and prioritization

Often learning the relational structure of the environment requires combining current experience with our memory of past experiences. This can be mediated by memory replay [4,46]. Rodent electrophysiology shows that while animals rest, or eat, there are brief high-frequency network oscillations in the hippocampus known as ripples [47], accompanied by neural activity patterns that sweep trajectories of place cells forwards and backwards at compressed time-scale [48,49]. These replayed trajectories are not merely memories of recent experience: they are task-related prospective trajectories to remembered goals [50] that are replayed backward when they encounter prediction errors [49]. They help capture the structure of the environment and support learning, prediction, inference, and planning [21,51–53].

In reinforcement learning, the proposal of planning as learning from memory replay has been captured in a family of models called Dyna [30,43••], in which an RL agent is trained during both direct experience as well as simulated experience during offline replay. Replay here is simulation of experience by replaying past episodes stored in memory, or reconstructing experience from a matrix of transitions. Table 1 summarizes four replay-based learning algorithms, including Dyna-Q [43••], Dyna-Q+ [30], prioritized sweeping [6••,54,55], and SR-Dyna [26••,32••]. In Dyna-Q, a model-free Q learner agent learns cached values of actions both during experience, and offline via replayed trajectories from a model-based transition matrix. Dyna-Q + gives a bonus to sampling states that have not been recently explored. This enables shortcut discovery, a caveat in Dyna-Q as it greedily maximizes value, missing the opportunity to discover new shortcuts. In prioritized Dyna, replay priority is given to experiences where prediction errors occurred, as well as their predecessors and successors. In SR-Dyna, SR's predictive representations are learned both online during direct experience and offline via memory replay. A more recent machine learning study offers a similar approach to SR-Dyna, using successor representations and simulated experience to learn structures in partially observable environments [92].

Empirically it has been shown that SR-Dyna outperforms MF, MB, MF-MB, Dyna-Q(+), prioritized Dyna, and

Table 1

Replay-based (Dyna) algorithms in reinforcement learning.

Algorithm	Learning and prioritization	Advantage	Caveat
Dyna-Q [43••]	MF Q-learner learns and acts during experience, is trained by offline replay of random states using MB	Fast, but flexible	New shortcuts
Dyna-Q+ [30]	Dyna-Q, but not random: bonus for exploring states that have not been visited for a while	Finds new shortcuts	Needs many samples, HRL
Prioritized sweeping [6••,54,55]	Events with unsigned prediction error (PE) during experience put in a priority queue. Later, states on trajectories leading to and from the PE-tagged state are given priority for replay.	HRL solutions	Large decision trees
SR-Dyna [26••,32••]	learns successor representations through experience and updates SR via offline episodic replay, prioritizes recent experience	Human-like retrospective revaluation	Discrete states, future: feature based

pure SR (without replay) in capturing human retrospective revaluation behavior [26••,32••]. One human fMRI study in particular focused on how prioritizing replay by prediction errors affected planning behavior. Consistent with any prioritized Dyna model, larger unsigned prediction errors were followed by more offline replay, and offline replay of predecessors of states tagged with PE was correlated with future revaluation behavior [6••]. The correlation between fMRI evidence of replay and revaluation behavior was more pronounced in the condition with higher reward variance, or uncertainty. In the real world, rewards and transition structures often vary as well: your favorite food truck moves elsewhere, and the subway maps rewire more often than desired. Future studies are required to better understand the role of uncertainty and volatility of rewards and transition structures on offline replay and prioritization.

Notably, the direction of replay sequences as well as their optimal prioritization can be computed using the successor representation itself. In Figure 1C, the SR matrix's 5*th* column represents its predecessors or states that predict it (its past, states 1 and 3), and the 2nd row represents the successors of state 2 or the states that 2 predicts (its future, states 4, 6, 7, and 8). Note that depending on the scale of the SR matrix, the horizons of past and future within which predecessors and successors are represented differ. A recent model of prioritized forward and reverse replay leverages these properties and consults SR to compare the value of taking different mental actions during planning in order to determine what replay content maximizes planning outcomes [56].

Offline replay is also a key memory process contributing to generalization and consolidation [57,58]. Human and animal neural evidence supports a role for wake and sleep memory replay in generalization and community detection [59,60], inference [52,61] especially of unseen relations among states, problem solving, and memory consolidation [62,63]. Deep reinforcement learning algorithms also benefit from prioritized replay in generalization, discovery, and adversarial self-learning [64–66]. Different replay algorithms differ with regards to how memories are stored and the model from which they generate simulated experience. This includes how far into the past memories are replayed (e.g., an exponential decay may prioritize replaying more recent memories, or replay may be limited to the past 2000 episodes), and whether replayed memories are in sequential or random order. Future experiments with hierarchical structures, uncertainty, and sparse rewards are required to test different replay and sampling prioritization methods against human and animal performance. Furthermore, future modeling work is required to capture the relationship between hippocampal and cortical replay, sharp wave ripples and neural oscillations, and behavioral outcomes.

Abstraction, generalization, and transfer

Humans learn structures at multiple levels of abstraction. We can remember the specific structure of our childhood home, as well as the generalized structure of a house or an airport. We also use abstract structures to learn and discern new environments faster. Upon looking at an outlandish indoor scene in a sci fi movie, we can typically guess if it is a kitchen or an airport. Such higher levels of structural abstraction are sometimes discussed as schema, and have been studied in the medial prefrontal cortex [67]. Learning structures with the RL framework can capture both memory of specific structures and memory for generalized structures as well as their neural implementations. Here we briefly review a number of ways in which compression of predictive representations can benefit abstraction and transfer.

Successor feature abstraction and transfer

Most RL approaches assume learning from discrete states. However, we experience the world as a continuous series of dynamic features rather than discrete states. Different tasks and goals depend on different dynamic features of the same environment. An important challenge for RL algorithms concerns decomposing complex task structures to learn relevant feature weights and generalizing predictive feature representations to improve the learning of unseen tasks [68]. Successor feature learning offers a solution. If different tasks can

be done in the same environment (e.g., at home), decoupling the dynamics of the environment (e.g. doors open to rooms, obstacles block passage) from specific goals and rewards (e.g., food in the kitchen) enables the algorithm to consider not one but a set of policies during policy improvement [28]. Other than transfer across tasks with different reward functions [27], successor features have been shown to support behavior shown by classic model-based and model-free reinforcement learning models as well [69]. A recent model shows that hippocampal place cell firing can be captured using successor features. Place cells are learned from a basis set of known neurobiological features (i.e., boundary vector cells and grid cells) that offer low-dimensional representations of successor features [70].

Options

While successor feature learning enables generalization and transfer over features, the most challenging RL problems regard hierarchically structured environments in which rewards are sparse. A benchmark example is Atari's Montezuma's revenge, which most deep learning models fail to solve [71•]. A series of solutions come from the options framework in hierarchical reinforcement learning (HRL). Through options, agents can abstract policies or temporally extended series of primitive actions (e.g., move one square left) that simplify hierarchical goal discovery (e.g., "go forward until wall", "climb ladder up", "find the key", "use key to open door") [72••,73,74]. Recent evidence shows the use of options in human behavior in hierarchical tasks and transfer [75]. However, discovering a useful set of options is a challenging problem. One way to identify useful options and subgoals is via states that are frequently visited [72••]. This is why count-based representation learning, such as deep successor representation learning and successor feature learning, have been shown to support option discovery [28,76••,77]. As discussed earlier, the eigen-decomposition of count-based predictive representations can give us the basis sets for the entire state space. Similarly, the eigen-decomposition of options leads to "eigen-options", or compact abstracted options that factorize temporarily extended policies. Eigen-options simplify planning in complex environments into more manageable components. They allow transfer of learned structures as well: solutions to different tasks can be captured as the linear combination of eigen-options [76••,78,79••]. Mathematically, this powerful property of working with eigen-options is closely related to hierarchical decomposition and discovery of sub-task structure usingnon-negative matrix factorization [90].

Intrinsic motivation

Learning in the real world often takes place without a specific goal and with sparse rewards that are not immediately reinforced. Such learning requires intrinsically motivated information seeking and structure discovery [80]. Intrinsic motivation has been an active topic in machine learning and RL, discussed in terms of intrinsic reward, intrinsic value, drive, and curiosity-driven learning [81•,82•]. Intuitively, intrinsic motivation can be thought of in terms of any learning or inference approach that decomposes the environment into task-independent components, which can be later combined to estimate value once a new task or goal is introduced. It has been shown that Laplacian eigenmaps can function as intrinsic motivation. This is welcome news for a model that uses the SR. because Laplacian eigenmaps have been shown to be the equivalent of the eigenvectors of SR and linear slow feature analysis [15••,76••,83]. In the context of value function estimation, Laplacian eigenmaps are also knowm as proto-value function. When computing the proto-value function, an environment is decomposed into basis functions of successor features and options, the linear combinations of which could compute any given task's value function and hence serve as intrinsic motivation [79••,84]. It has been shown that approximating proto-value function using eigen-options (mentioned above) serves as intrinsic motivation to solve problems such as Montezuma's revenge [85]. These emerging findings open up exciting new directions for studying the role of structure discovery in planning in uncertain environments [93]. Such an endeavor can lead to a novel biologically plausible understanding of the link between learning and memory processes and intrinsic motivation.

While RL approaches to structure learning successfully capture behavioral and neural evidence for learning relational and statistical associations, one open question is how they can account for causal inference. Another open question is: what kinds of structures or graphs can a given learning algorithm learn and what are its limits of structure learning? How do these limits compare to the limits of human structure learning? A possible angle to approach these questions from is using the successor feature and eigen-option framework in RL.

Computational psychiatry

An important application of models of structure representation is in computational psychiatry. Many psychiatric disorders can emerge from learning maladaptive models of the environment's structures, or having beliefs that lead to using otherwise useful models maldaptively. For instance, in the SR-Dynra framework prioritizing the replay of memories with negative prediction error can lead to a predictive internal map that over-represents trajectories to negative outcome and under-represents trajectories to reward. Using such a maladaptive map for behavior can simulate anxiety, avoidance, and freezing behavior. On the other hand, an agent may have an accurate model of the world but a maladaptive pessimistic belief about the accuracy of these models. A recent computational study has shown that such an agent's behavior simulates avoidance behavior akin to anxiety

[88]. Future modeling and empirical work is required to understand the role of structure learning in pathological behavior.

Conclusion

Flexible memory and planning behavior rely on learning compact representations of environmental structures. This review focuses on using reinforcement learning for learning the structure of state spaces – irrespective of rewards. As an agent navigates an environment, a count-based representation learning approach can gradually learn compact representation of relationships in the graph of states (by learning how often a state leads to its successor states). This learned predictive representation of successor states can be updated via prioritized offline replay of past experience, enabling inference of unseen relationships among states. Factorization of this predictive representation offers abstract properties of the relational structure such as grids, bottlenecks, subgoals, etc. We summarized evidence that predictive representation learning and replay capture cumulative neural and behavioral evidence. We noted emerging topics in generalization and transfer. This rich body of studies calls for further computational work on structure learning, matched experimental work, and a computational update to earlier cognitive map theory.

Conflict of interest statement

Nothing declared.

Acknowledgement

The author gratefully acknowledges Salman Qasim for helpful comments on the manuscript and Lynn Nadel and Kim Stachenfeld for helpful discussions. This work was supported by NIMH Grant R01-MH104606 granted to Joshua Jacobs.

References and recommended reading

Papers of particular interest, published within the period of review, have been highlighted as:

- • of special interest
- •• of outstanding interest

1. Tolman EC: **Cognitive maps in rats and men**. *Psychol Rev* 1948, **55**:189-208.

2. O'Keefe J, Nadel L: *The Hippocampus as a Cognitive Map*. Oxford: Clarendon Press; 1978.

3. Behrens TEJ, Muller TH, Whittington JCR, Mark S, Baram AB, Stachenfeld KL, Kurth-Nelson Z: **What is a cognitive map? Organizing knowledge for flexible behavior**. *Neuron* 2018, **100**:490-509.

4. Foster DJ: **Replay comes of age**. *Annu Rev Neurosci* 2017, **40**:581-602.

5. •• Momennejad I, Howard MW: **Predicting the future with multi-scale successor representations**. *bioRxiv* 2018
A mathematical framework connecting multi-scale representations of past memories to multi-scale predictive representations of future trajectories. The authors show the equivalence of Laplace transform and multiscale successor representations, and show that the inverse of a Lapalce transform (roughly the derivative of the multi-scale SR ensemble) can reconstruct future trajectory and estimate path-dependent distance to goal locations.

6. •• Momennejad I, Otto AR, Daw ND, Norman KA: **Offline replay supports planning in human reinforcement learning**. *Elife* 2018, **7**:e32548
This human fMRI study tested the role of prioritized sweeping (prioritization by prediction errors) on planning behavior. They found that larger unsigned prediction errors were followed by more offline replay, and offline replay of predecessors of states tagged with PE was correlated with future revaluation behavior. The hippocampus, prefrontal cortex, and category-selective cortical regions were engaged during offline replay, and the correlation between fMRI evidence for replay and revaluation behavior was more pronounced in the condition with higher reward uncertainty.

7. • Aronov D, Nevers R, Tank DW: **Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit**. *Nature* 2017, **543**:719
The authors showed that place and grid cells that fire in response to space, show similar firing in response to non-spatial state-spaces.

8. Constantinescu AO, O'Reilly JX, Behrens TEJ: **Organizing conceptual knowledge in humans with a gridlike code**. *Science* 2016, **352**:1464-1468.

9. Schafer M, Schiller D: **Navigating social space**. *Neuron* 2018, **100**:476-489.

10. Howard LR, Javadi AH, Yu Y, Mill RD, Morrison LC, Knight R, Loftus MM, Staskute L, Spiers HJ: **The hippocampus and entorhinal cortex encode the path and Euclidean distances to goals during navigation**. *Curr Biol* 2014, **24**:1331-1340.

11. Mehta MR, Quirk MC, Wilson MA: **Experience-dependent asymmetric shape of hippocampal receptive fields**. *Neuron* 2000, **25**:707-715.

12. •• Sarel A, Finkelstein A, Las L, Ulanovsky N: **Vectorial representation of spatial goals in the hippocampus of bats**. *Science* 2017, **355**:176-180
The authors showed distance-to-goal firing in hippocampal neurons of the bat. Notably, they showed that the vectorial representations towards the goal location respect the introduction of an obstacle along the way. This finding is in-line with policy-dependent SR, but challenges earlier notions of hippocampal cognitive maps as Euclidean.

13. Gauthier JL, Tank DW: **A dedicated population for reward coding in the hippocampus**. *Neuron* 2018, **99**:179-193.e7.

14. Bellmund JLS, Deuker L, Doeller CF: **Mapping sequence structure in the human lateral entorhinal cortex**. *Elife* 2019, **8**: e45333.

15. •• Stachenfeld KL, Botvinick MM, Gershman SJ: **The hippocampus as a predictive map**. *Nat Neurosci* 2017, **20**:1643-1653
This elegant computational study shows that the successor representation can serve as an organizing principle for place and grid fields in the medial temporal lobe. The authors showed that place cell firing patterns can be simulated by a column of a successor representation, while grid cell firing patterns are simulated by an eigendecomposition of the successor representations. They compare simulations to a number of rodent electrophysiology studies of spatial navigation and an fRMI study of statistical learning.

16. Stachenfeld K: **Learning neural representations that support efficient reinforcement learning**. *Doctoral Dissertation*. Princeton University; 2018.

17. Boccara CN, Nardin M, Stella F, O'Neill J, Csicsvari J: **The entorhinal cognitive map is attracted to goals**. *Science* 2019, **363**:1443-1447.

18. Butler WN, Hardcastle K, Giocomo LM: **Remembered reward locations restructure entorhinal spatial maps**. *Science* 2019, **363**:1447-1452.

19. Chaudhuri R, Gerçek B, Pandey B, Peyrache A, Fiete I: **The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep**. *Nat Neurosci* 2019, **22**:1512-1520.

20. Low RJ, Lewallen S, Aronov D, Nevers R, Tank DW: **Probing variability in a cognitive map using manifold inference from neural dynamics**. *bioRxiv* 2018.

21. Wu X, Foster DJ: **Hippocampal replay captures the unique topological structure of a novel environment**. *J Neurosci* 2014, **34**:6459-6469.

22. Babichev A, Cheng S, Dabaghian YA: **Topological schemas of cognitive maps and spatial learning**. *Front Comput Neurosci* 2016, **10**.

23. Whittington JCR, Muller TH, Mark S, Chen G, Barry C, Burgess N, Behrens TEJ: **The Tolman-Eichenbaum machine: Unifying space and relational memory through generalisation in the hippocampal formation**. *bioRxiv* 2019 http://dx.doi.org/10.1101/770495.

24. Collins AGE: **The cost of structure learning**. *J Cogn Neurosci* 2017, **29**:1646-1655.

25. Radulescu A, Niv Y, Ballard I: **Holistic reinforcement learning: The role of structure and attention**. *Trends Cogn Sci* 2019, **23**:278-292.

26. •• Russek EM, Momennejad I, Botvinick MM, Gershman SJ, Daw ND: **Predictive representations can link model-based reinforcement learning to model-free mechanisms**. *PLoS Comput Biol* 2017, **13**:e1005768
In this computational study the authors show that a family of algorithms that learn successor representations can display a subset of model-based RL behavior, while requiring less decision-time than dynamic programming required by MB learners. SR-Dyna is introduced, combining successor representation learning and learning from replay. SR-Dyna outperforms other models in capturing human behavior and outperforms varieties of Dyna architecture in solving specific RL problems especially given insufficient sampling.

27. Lehnert L, Tellex S, Littman ML: *Advantages and Limitations of using Successor Features for Transfer in Reinforcement Learning*. 2017.

28. Barreto A, Borsa D, Quan J, Schaul T, Silver D, Hessel M, Mankowitz D, Zidek A, Munos R: **Transfer in deep reinforcement learning using successor features and generalised policy improvement**. *International Conference on Machine Learning* 2018:501-510.

29. Daw ND, Dayan P: **The algorithmic anatomy of model-based evaluation**. *Philos Trans R Soc Lond B Biol Sci* 2014, **369**.

30. Sutton RS, Barto AG: *Reinforcement Learning: An Introduction*. MIT Press; 2018.

31. Tenenbaum JB, Kemp C, Griffiths TL, Goodman ND: **How to grow a mind: statistics, structure, and abstraction**. *Science* 2011, **331**:1279-1285.

32. •• Momennejad I, Russek EM, Cheong JH, Botvinick MM, Daw ND, Gershman SJ: **The successor representation in human reinforcement learning**. *Nat Hum Behav* 2017, **1**:680-692
The authors designed human behavioral experiments with varieties of revaluation to test the predictions of the successor representation against model-free, model-based, and hybrid models. They showed that an algorithm combining successor representation learning and offline replay best captures human performance.

33. •• Dayan P: **Improving generalization for temporal difference learning: The successor representation**. *Neural Comput* 1993, **5**:613-624
The original proposal of successor representations as a method for generalization in temporal difference learning.

34. Gershman SJ, Moore CD, Todd MT, Norman KA, Sederberg PB: **The successor representation and temporal context**. *Neural Comput* 2012, **24**:1553-1568.

35. Schapiro AC, Rogers TT, Cordova NI, Turk-Browne NB, Botvinick MM: **Neural representations of events arise from temporal community structure**. *Nat Neurosci* 2013, **16**:486-492.

36. Botvinick M, Weinstein A: **Model-based hierarchical reinforcement learning and human action control**. *Philos Trans R Soc Lond B Biol Sci* 2014, **369**.

37. • Garvert MM, Dolan RJ, Behrens TE: **A map of abstract relational knowledge in the human hippocampal-entorhinal cortex**. *Elife* 2017, **6** 04 27
The authors show that the successor representation can capture fMRI-based pattern similarity in an association learning task.

38. Girvan M, Newman MEJ: **Community structure in social and biological networks**. *Proc Natl Acad Sci U S A* 2002, **99**:7821-7826.

39. Grindrod P, Parsons MC, Higham DJ, Estrada E: **Communicability across evolving networks**. *Phys Rev E* 2011, **83**:046120.

40. Brunec IK, Momennejad I: **Predictive representations in hippocampal and prefrontal hierarchies**. *bioRxiv* 2019.

41. Brunec IK, Bellana B, Ozubko JD, Man V, Robin J, Liu Z-X, Grady C, Rosenbaum RS, Winocur G, Barense MD *et al.*: **Multiple scales of representation along the hippocampal anteroposterior axis in humans**. *Curr Biol* 2018, **28**:2129-2135.e6.

42. Epstein RA, Patai EZ, Julian JB, Spiers HJ: **The cognitive map in humans: spatial navigation and beyond**. *Nat Neurosci* 2017, **20**:1504-1513.

43. •• Sutton RS: **Dyna, an integrated architecture for learning, planning, and reacting**. *SIGART Bull* 1991, **2**:160-163
The original proposal of the Dyna architecture in reinforcement learning. The agent has both model-free and model-based components, learned and updated during direct experience. In addition, offline simulated experience generated by a model-based learner helps update cached value in a model-free learner offline. Hence, the Dyna agent's value-based model-free policy can be inferred and updated in the absence of direct experience.

44. Banino A, Barry CJ, Benigno U, Blundell C, Lillicrap T, Mirowski P, Pritzel A, Chadwick M, Hassabis D, Hadsell R *et al.*: **Vector-based navigation using grid-like representations in artificial agents**. *Nature* 2018 http://dx.doi.org/10.1038/s41586-018-0102-6.

45. Brunec IK, Javadi A-H, Zisch FEL, Spiers HJ: **Contracted time and expanded space: The impact of circumnavigation on judgements of space and time**. *Cognition* 2017, **166(9)**:425-432.

46. Pfeiffer BE: **The content of hippocampal "replay"**. *Hippocampus* 2017 http://dx.doi.org/10.1002/hipo.22824.

47. Buzsáki G: **Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning**. *Hippocampus* 2015, **25**:1073-1188.

48. Diba K, Buzsáki G: **Forward and reverse hippocampal place-cell sequences during ripples**. *Nat Neurosci* 2007, **10**:1241-1242.

49. Ambrose RE, Pfeiffer BE, Foster DJ: **Reverse replay of hippocampal place cells is uniquely modulated by changing reward**. *Neuron* 2016, **91**:1124-1136.

50. Pfeiffer BE, Foster DJ: **Hippocampal place-cell sequences depict future paths to remembered goals**. *Nature* 2013, **497**:74-79.

51. Johnson A, Redish AD: **Hippocampal replay contributes to within session learning in a temporal difference reinforcement learning model**. *Neural Netw* 2005, **18**:1163-1171.

52. Ólafsdóttir HF, Barry C, Saleem AB, Hassabis D, Spiers HJ: **Hippocampal place cells construct reward related sequences through unexplored space**. *Elife* 2015, **4**.

53. Cazé R, Khamassi M, Aubin L, Girard B: **Hippocampal replays under the scrutiny of reinforcement learning models**. *J Neurophysiol* 2018, **120**:2877-2896.

54. Peng J, Williams RJ: **Efficient learning and planning within the dyna framework**. *Adapt Behav* 1993, **1**:437-454.

55. Moore AW, Atkeson CG: **Prioritized sweeping: Reinforcement learning with less data and less time**. *Mach Learn* 1993, **13**:103-130.

56. Mattar MG, Daw ND: **Prioritized memory access explains planning and hippocampal replay**. *Nat Neurosci* 2018, **21**:1609-1617 http://dx.doi.org/10.1101/225664.

57. Atherton LA, Dupret D, Mellor JR: **Memory trace replay: The shaping of memory consolidation by neuromodulation**. *Trends Neurosci* 2015, **38**:560-570.

58. Tambini A, Davachi L: **Awake reactivation of prior experiences consolidates memories and biases cognition**. *Trends Cogn Sci* 2019, **23**:876-890.

59. Schapiro AC, Turk-Browne NB, Norman KA, Botvinick MM: **Statistical learning of temporal community structure in the hippocampus**. *Hippocampus* 2016, **26(1)**:3-8.

60. Schapiro AC, McDevitt EA, Rogers TT, Mednick SC, Norman KA: **Human hippocampal replay during rest prioritizes weakly learned information and predicts memory performance**. *Nat Commun* 2018, **9**.

61. Liu Y, Dolan RJ, Kurth-Nelson Z, Behrens TEJ: **Human replay spontaneously reorganizes experience**. *Cell* 2019, **178**:640-652.e14.

62. Genzel L, Robertson EM: **To replay, perchance to consolidate**. *PLoS Biol* 2015, **13**:e1002285.

63. Lewis PA, Knoblich G, Poe G: **How memory replay in sleep boosts creative problem-solving**. *Trends Cogn Sci* 2018, **22**:491-503.

64. Schaul T, Quan J, Antonoglou I, Silver D: **Prioritized experience replay**. *arXiv [csLG]* 2015.

65. Horgan D, Quan J, Budden D, Barth-Maron G, Hessel M, van Hasselt H, Silver D: **Distributed prioritized experience replay**. *arXiv [csLG]* 2018.

66. Shin H, Lee JK, Kim J, Kim J: **Continual learning with deep generative replay**. *arXiv [csAI]* 2017.

67. Gilboa A, Marlatte H: **Neurobiology of schemas and schema-mediated memory**. *Trends Cogn Sci* 2017, **21**:618-631.

68. Borsa D, Barreto A, Quan J, Mankowitz DJ, van Hasselt H, Munos R, Silver D, Schaul T: **Universal successor features approximators**. *arXiv* 2018.

69. Lehnert L, Littman ML: *Successor Features Support Model-based and Model-free Reinforcement Learning*. 2019.

70. de Cothi W, Barry C: **Neurobiological successor features for spatial navigation**. *bioRxiv* 2019 http://dx.doi.org/10.1101/789412.

71. • Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller M, Fidjeland AK, Ostrovski G *et al.*: **Human-level control through deep reinforcement learning**. *Nature* 2015, **518**:529-533

The initial deep reinforcement learning proposal, applied to the Atari suite. Though impressive in its human level performance on games with short-term feedback, this algorithm performed very poorly on games with hierarchical structures and sparse rewards. Many solutions to these problems in recent years are closely related to successor representation learning.

72. •• Stolle M, Precup D: **Learning options in reinforcement learning**. In *Abstraction, Reformulation, and Approximation.* Edited by Koenig S, Holte RC. Berlin Heidelberg: Springer; 2002:212-223

The authors extend the options framework proposed by Precup 2000 to account for how to identify good options in an environment. They assume the agent will perform different tasks in the same environment, and hence set subgoals as states that are frequently visited (similar to the count-based property of the successor representations discussed in this review).

73. Bacon P-L, Harb J, Precup D: **The option-critic architecture**. *arXiv:160905140 [cs]* 2016.

74. Mankowitz DJ, Mann TA, Bacon P-L, Precup D, Mannor S: **Learning robust options**. *arXiv:180203236 [cs, stat]* 2018.

75. Xia L, Collins AGE: **Temporal and state abstractions for efficient learning, transfer and composition in humans**. *bioRxiv* 2020 http://dx.doi.org/10.1101/2020.02.20.958587.

76. •• Machado MC, Rosenbaum C, Guo X, Liu M, Tesauro G, Campbell M: **Eigenoption discovery through the deep successor representation**. *arXiv:171011089 [cs]* 2017

Options enable hierarchically decomposing a task into subtasks. The authors propose eigenoptions: options obtained from successor representations that encode diffusive information flow in the environment. They test the algorithm for subgoal discovery in Montezuma's revenge, when handcrafted features are not available. They assume equivalence between proto-value functions and the successor representation.

77. Harutyunyan A, Vrancx P, Hamel P, Nowe A, Precup D: **Per-decision option discounting**. *International Conference on Machine Learning* 2019:2644-2652.

78. Machado MC, Bellemare MG, Bowling M: **Count-based exploration with the successor representation**. *arXiv:180711622 [cs, stat]* 2019.

79. •• Machado MC, Bellemare MG, Bowling M: **A Laplacian framework for option discovery in reinforcement learning**. *Proceedings of the 34th International Conference on Machine Learning - Volume 70* 2017:2295-2304. JMLR.org

The authors solve representation learning and option discovery in benchmark Atari environments introducing eigenpurposes: intrinsic reward functions derived from the learned representations. The options discovered from eigenpurposes act at different time scales (useful for exploration), traverse the principal directions of the state space, and are discovered without reliance external rewards: hence they are useful for multiple tasks.

80. Barto AG, Simsek Ö: *Intrinsic Motivation For Reinforcement Learning Systems*. 2005.

81. • Burda Y, Edwards H, Pathak D, Storkey A, Darrell T, Efros AA: **Large-scale study of curiosity-driven learning**. *ICLR*. 2019

This paper reports that purely curiosity-driven learning without any extrinsic reward performs well across 54 benchmark environments (including the Atari stuie). They observed a strong alignment between intrinsic curiosity objective and extrinsic rewards of many game environments.

82. • Pathak D, Agrawal P, Efros AA, Darrell T: **Curiosity-driven exploration by self-supervised prediction**. *ICML*. 2017

The authors propose curiosity as an intrinsic reward signal to enable exploration and structure learning in the absence of external rewards. They defined curiosity in terms of the prediction errors about consequences of actions in a visual feature space, including high-dimensional continuous state spaces. The model performed well on exploration in VizDoom and Super Mario Bros, given sparse to no extrinsic reward, and showed generalization to unseen scenarios.

83. Sprekeler H: **On the relation of slow feature analysis and Laplacian eigenmaps**. *Neural Comput* 2011, **23**:3287-3302.

84. Ramesh R, Tomar M, Ravindran B: **Successor options: An option discovery framework for reinforcement learning**. *arXiv:190505731 [cs, stat]* 2019.

85. Xing L: **Learning and exploiting multiple subgoals for fast exploration in hierarchical reinforcement learning**. *arXiv:190505180 [cs, stat]* 2019.

86. Momennejad I, Haynes J-D: **Human anterior prefrontal cortex encodes the 'what' and' when' of future intentions**. *Neuroimage* 2012, **61(1)**:139-148.

87. Momennejad I, Haynes J-D: **Encoding of prospective tasks in the human prefrontal cortex under varying task load**. *J Neurosci* 2013, **33(44)**:17342-17349.

88. Zorowitz S, Momennejad I, Daw N: **Anxiety, avoidance, and sequential evaluation**. *Computational Psychiatry*. In press.

89. Bellmund JLS, de Cothi W, Ruiter TA, Nau M, Barry C, Doeller CF: **Deforming the metric of cognitive maps distorts memory**. *Nat Hum Behav* 2020, **4**:177-188.

90. Saxe AM: **Hierarchical subtask discovery with non-negative matrix factorization**. In *International Conference on Learning Representations.* Edited by Bengio Y, LeCun Y.*Vancouver, Canada: 2018*.

91. Lynn CW, Bassett DS: **Graph learning: How humans infer and represent networks**. *arXiv [physics.soc-Ph]* 2019.

92. Vértes E, Sahani M: **A neurally plausible model learns successor representations in partially observable environments**. In *Advances in Neural Information Processing Systems 32.* Edited by Wallach H, Larochelle H, Beygelzimer A, Alché-Buc F, Fox E, Garnett R. Curran Associates, Inc.; 2019:13714-13724.

93. Janz D, Hron J, Mazur P, Hofmann K, Hernández-Lobato JM, Tschiatschek S: **Successor uncertainties: Exploration and uncertainty in temporal difference learning**. In *Advances in Neural Information Processing Systems 32.* Edited by Wallach H, Larochelle H, Beygelzimer A, Alché-Buc F, Fox E, Garnett R. Curran Associates, Inc.; 2019:4507-4516.