DEINFORCEMENT LEARNING

Chaytan Inman
University of Washington
Seattle
chaytan@uw.edu

Varun Ananth
University of Washington
Seattle
vananth3@uw.edu

Marlene Grieskamp
University of Washington
Seattle
markamp@uw.edu

Janna Hong
University of Washington
Seattle
jannahg@uw.edu

ABSTRACT

An important challenge of human decision-making is determining via trial and error which options maximize reward and minimize punishment. In computer science, this problem is known as reinforcement learning (RL), and particular RL paradigms such as the advantage actor-critic (A2C) have been the subject of extensive research (Niv, 2009). Current RL algorithms are insufficient representations of the brain, despite the fact that this biological analogy has historically advanced the field of computer science (Tassa et. al, 2018). When mimicking dopamine pathways, RL often disregards one of the most potent biological signals: pain. The absence of a reward signal, also known as a negative signal, is frequently interpreted as being equivalent to punishment (Schultz et. al, 1997). However, the biological mechanisms that interpret, transmit, and permit pain in the body contradict this assertion. We argue that people avoid unfavorable situations more rapidly if they learn through pain as opposed to through a lack of reward. Therefore, we propose that adding pain into current RL models will not only allow algorithms to converge more quickly, but also cause behavior to become more safe, sophisticated, and generalizable. This work examines the historical connections between RL and neuroscience, synthesizes neuroscientific understandings of pain, and proposes refinements to current biologically inspired techniques for incorporating pain into RL algorithms.

***K*eywords** Machine Learning · Reinforcement Learning · Neuroscience · Cognitive Science · Psychology

Introduction

The history of reinforcement learning (RL) is indelibly wound with the mathematical foundations of neuroscience. Neuroscience has much to offer computer science, given that the brain is one of the most flexible, adaptive, efficient, and intelligent learning paradigms. This long-standing relationship has yielded significant benefits, but more multidisciplinary effort is required to reach the full potential of applications to computer science as neuroscience advances.

Richard Sutton, who was trained in both psychology and computer science, created the ’Temporal Difference Learning’ (TD Learning) algorithm in the 1980s to explain the neuronal response to violated expectations. This marked the beginning of the intertwined history of computer science and neuroscience. The discrepancy between the predicted and actual reward is the definition of prediction error. TD Learning asserts that learning occurs through prediction error. The strong similarity between dopamine firing rates and the reward prediction error (RPE) signal was then described by Wolfram Schultz in 1997.

Schultz demonstrated that when monkeys get an unanticipated reward, their brains respond with an influx of action potentials, often known as a phasic firing pattern. Nonetheless, the phasic pattern disappears when the incentive is anticipated. These results demonstrated that TD error (the difference between the predicted and actual firing rates) was stored in the spiking patterns of dopaminergic neurons.

At the same time, developments in reinforcement learning initiated the use of TD error to update the weights of neurons in artificial neural networks. In this approach, reinforcement learning has created unique optimization algorithms that mimic human-like learning. Popular algorithms such as actor-critic reinforcement learning continue to rely on TD error. Given the significance of programmatic algorithms that replicate ancient biological brain algorithms, it is imperative that new RL paradigms continue to emerge. To date, the brain is more generalizable than any known learning algorithm; we have much to learn from studying it. Thus, we aim to improve the performance of reinforcement learning algorithms by upgrading models that learn from both pain and reward, with our implementation based on a detailed evaluation of the various neuroscience studies on pain pathways.

Roadmap

We begin with a review of dopaminergic learning and reinforcement learning in the brain. We then pull concepts from psychology (operant conditioning) to differentiate between various types of reinforcement and punishment, as well as consider which ones are relevant to our ultimate theoretical implementation into RL. The pain route is then compared to the previously outlined dopaminergic learning pathway. In conclusion, we synthesize these ideas to present a high-level RL framework based on the MaxPain algorithm while incorporating a biologically-inspired pain signal, distinct from a negative reward signal.

Dopaminergic Learning

Dopamine is a neurotransmitter with a reputation for its reward and pleasure qualities in popular culture. Studies indicate that the same dopamine are also implicated in pain, which may involve the remodeling of the reward circuitry. Approximately 90% of dopamine producing neurons are in two areas of the midbrain nuclei called the *substantia nigra pars compacta* (SNc) and the *mesolimbic ventral tegmental area* (VTA) (Arias-Carrión et. al, 2014). These neurons project to the *nucleus accumbens* (NAc), which is the reward-related dopamine site. Blocking the dopamine pathway to NAc deprives the rewarding effects.

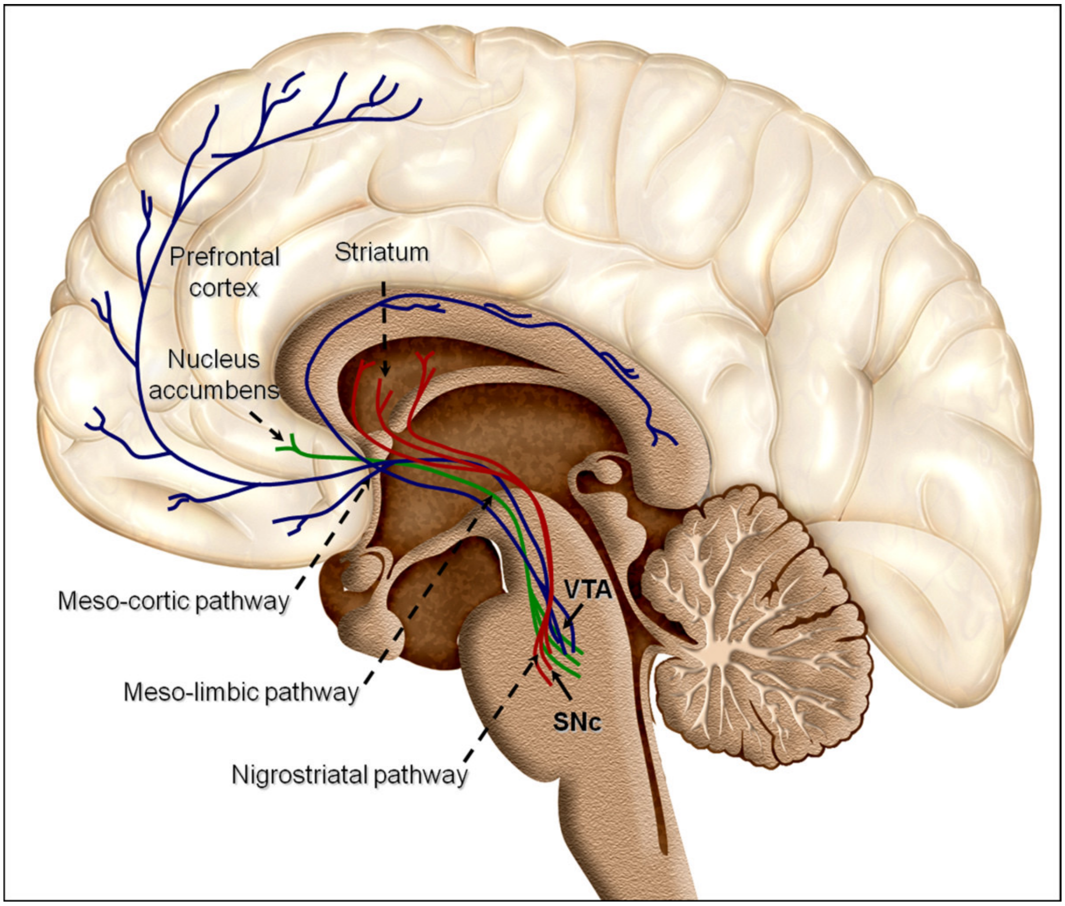


Figure 1: The mesolimbic pathway shows the dopaminergic neurons made in the ventral tegmental area projecting to the nucleus accumbens. (Arias-Carrión et. al, 2014)

In the early 1950s, scientists discovered the significance of the VTA region by observing the effects of electrical stimulation to certain regions of the brain in rats. When electrical stimulation in the VTA was followed by a certain task such as lever pressing, rats repeatedly executed that action (Olds & Milner, 1954). The rats ended up pressing on the lever 2000 times per hour when they learned that this specific behavior reliably leads to an electrical stimulation. In this case, the action of lever pulling is followed by the electrical stimulation, thus the stimulus is the “reward”. Olds and Milner elaborate on the concept of reward as follows: “In its reinforcing capacity, a stimulus increases, decreases, or leaves unchanged the frequency of preceding responses, and accordingly it is called a reward, a punishment, or a neutral stimulus” (Olds & Milner, 1954, p. 419). Recent experiments in humans undergoing deep brain stimulation (DBS) for Parkinson’s show similar results. After participants learned that a certain task was followed by electrical stimulation of the SNc, an area with abundant dopaminergic neurons, they repeatedly performed that task, soon even without the stimulus (Perelman School of Medicine at the University of Pennsylvania, 2014).

Dopaminergic neurons (DA) can fire in two distinct patterns in response to varying stimuli: phasic and tonic activity. Phasic activity refers to a burst of action potentials firing in a short period of time, with a rate of up to 20Hz. In contrast, tonic activity indicates a steady firing rate of around 5 Hz. Tonic activity recorded in monkeys implies congruence between actual and expected reward, while phasic signal indicates a component of surprise and mediates a prediction error during learning (Schultz et. al, 1997). As a result, dopaminergic reactions diminish as learning and estimation of rewards progress. Likewise, the phasic firing activity decreases following the delivery of the reward. DA might indicate the difference between the anticipated reward and the actual reward. Though the complete neurological effects of dopamine, such as tonic firing pattern effects, have not yet been integrated into reinforcement learning models, contemporary improvements to reinforcement learning are based on the understanding of the phasic dopamine response (Beeler et. al. 2010). By comprehending the phasic and tonic expressions of dopaminergic neurons, we can inform the future direction of action selection in models for temporal difference reinforcement learning.

Reinforcement Learning

Reinforcement Learning (RL) is a branch of machine learning involved with choosing optimal policies, state value estimates, or both to optimize an agent’s selection of the most rewarding action in a given environment. RL has been used by companies such as DeepMind to train humanoid and non-humanoid physical models to walk, run, jump, and play games (Tassa et. al, 2018). It tends to perform best in scenarios where there are complex states and decisions are plentiful, not dissimilar to our own environment. The advantage actor-critic (A2C) RL model employs TD learning to adjust its reward predictions through time, analogous to how humans learn through classical and operant conditioning (Niv, 2009). **We have chosen to study A2C because it contains the most straightforward implementation of advantage.** Before exploring the relevance of the "advantage" variable and its linkages to reward prediction error (RPE) in the brain, it is crucial to differentiate between the actor and critic components.

The actor is (often) a neural network which learns policy $\boldsymbol{\pi}$ parameterized by $\boldsymbol{\theta}$, that is a function of state **s**. In other words, a policy determines which action should be taken in a given state and this choice is influenced by the internal values of $\boldsymbol{\theta}$ (in this case the neural network weights). Policies can be tuned to maximize reward (as is the case with A2C), or achieve a goal parameterized as a function of that reward. Actors in A2C are stochastic by nature, meaning they output a probability distribution for taking an action in the action space based on the current state (Geron, 2019). This has high level parallels to how we as humans interact with our environment. We consider the “state” of the world around us and then “act” to maximize some reward tied to a goal. For example, when you decide to take action "A" to walk your dog, one might think of this as applying your internal policy ($\boldsymbol{\pi}$) to the state **s**, wherein your dog is barking at the door wanting to go for a walk. You have, in this example, tuned your internal weights $\boldsymbol{\theta}$ so that your policy outputs the action "A" of dog walking when given the state “s” of your dog barking. In many cases, the actor is a deep neural network with input dimensions being equivalent to the dimensionality of the state, and the output dimensionality being equivalent to that of the action space. Actions selected by the actor are taken in the environment, physical or simulated, and they will receive a scalar reward for taking that action. The actor’s weights are updated with the critic’s weights along the “advantage” variable, which will be discussed in a later section.

The critic network in A2C is more abstract. Its purpose is to approximate the function $V(s)$. This function returns the overall value of being in a state **s**, given a policy $\boldsymbol{\pi}$. It is equivalent to the expected return of starting in state “**s**” and following policy $\boldsymbol{\pi}$ thereafter (return in terms of discounted rewards) (Mnih et. al, 2016).

Building upon the dog walking example; assume two separate states s_{leash} and s_{no_leash} In these states, your dog is barking and you have a leash or don’t have a leash respectively. Say you don’t have it, and decide to walk the dog anyways (following policy $\boldsymbol{\pi}$ where you walk the dog if it is barking). It runs away from you and now you have no dog, which is a terrible situation to be in. If you had the leash, you would walk your dog without incident and your situation is overall better. The “value” of the state s_{leash} is greater than the “value” of the state s_{no_leash} because after following policy $\boldsymbol{\pi}$ in both of those states one led to ruin and the other was just fine.

$$V^{\pi}(s) = E_{\pi}\{R_t|s_t = s\} = E_{\pi}\{\sum_{k=0}^{\infty} \gamma r(s_{t+k+1}, a_{t+k+1})|s_t = s\}$$

In most cases, the "perfect" V function shown above is incomputable. Such is the case when we do not have access to the rewards for any given state, which we do not when we do not have a model of the environment. $V(s)$ is, however, commonly estimated by another deep neural network: in our case the critic. The input of the critic is the state and the output is a scalar estimate of the value function. The critic is important because the value function it estimates is used in the calculation of the crucial “advantage” variable which is then used to update the weights of both itself and the actor. The actor, as discussed before, is the driving force behind the model’s decision making. A better performing actor means a better performing network.

The following sections explore the relevance of extending the synergy between reinforcement learning and dopaminergic learning beyond reward-based learning.

Dopaminergic Influences on Actor-Critic Systems

Temporal Difference learning is the framework upon which actor-critic updates are based, and TD error calculations in RL are done with the use of the advantage variable; the advantage can be broken down to its constituent parts below:

$$A = r(s_t, a_t) + \gamma V(s_{t+1}) - V(s_t)$$

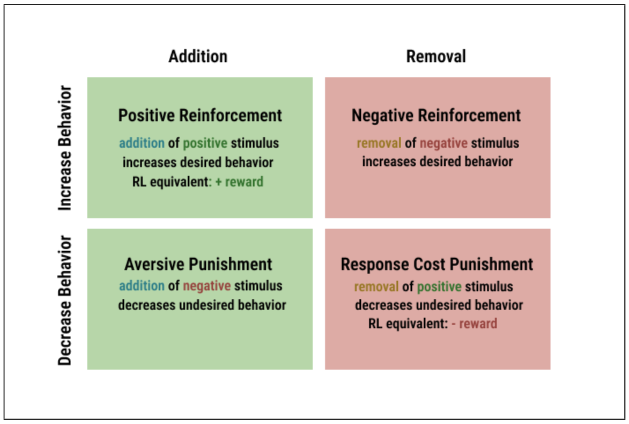
Assuming the agent just took an action and has moved to a new state: $r(s_t, a_t)$ is the reward given for taking that certain action in that certain state. Reward is often given by the environment after an action is taken and not model-intrinsic. The $\gamma V(s_{t+1})$ term represents the estimated future discounted rewards starting at the new state the agent has just entered. Finally, the $V(s_t)$ term is the estimated value function for the state the agent was just in before acting. Combining the first two terms, $r(s_t, a_t)$ + $\gamma V(s_{t+1})$, we get the reward the agent received for taking an action in its previous state plus the discounted predicted value of the agent’s current state. We call this the TD Target. Recall that the critic network is being trained to provide precise estimates of $V(s)$. To train the network with backpropagation, we need some ground truth to understand how well the model predicted the V function for this timestep. The TD Target contains the ground truth in the form of $r(s_t, a_t)$. $V(s_t)$ is subtracted from the TD target, and the difference between these two is the TD Error. The term "error" refers to the discrepancy that exists between the rewards that were actually gained and those that were estimated to be obtained. Note that the $\gamma V(s_{t+1})$ term is partially nullified by the tail end of the $V(s_t)$ term, since this value function contains discounted value predictions for future states as well (Watabe-Uchida et. al, 2017). The worse the prediction of $V(s_t)$ by the critic, the further from zero the advantage is, and vice versa. This is why (A^2) is the critic loss in an A2C network: squaring the advantage allows us to treat negative and positive TD Error equivalently while preserving the differentiability of the function and advantage is an appropriate measure of how well the network is doing at approximating $V(s_t)$. The actor’s loss is also scaled by the advantage, but the calculation of loss for a stochastic model is beyond the scope of this paper.

Since TD learning lays the groundwork for calculating advantage in actor-critic systems, it stands to reason that A2C weight updates are linked to dopaminergic learning in the brain. In the case of RPE, as more stimuli are experienced by the organism and dopaminergic learning occurs (see Dopaminergic Learning section), the RPE approaches zero. This means the organism has learned how to correctly predict the reward given its state (Schultz et. al, 1997). Just as RPE lowers while an organism learns to predict reward, the advantage variable in an actor-critic system lowers as the critic learns to estimate $V(s)$.

If no reward is present but reward is predicted, then dopamine activity is heavily depressed which causes updates to the organism’s “value estimation function”. However, an important distinction needs to be drawn between the lack of a reward and a punishment. In RL, negative rewards stemming from a state-action pair are seen as “punishment” for a model. Initially, this seems accurate. Value updates propagate through a network, telling the agent that this state is not as valuable as it initially estimated. However, if we consider on a high level the effect of pain, we see that there is a large difference in how organisms react to pain versus negative reward. If a child puts their hand on a hot plate, they are unlikely to do it again, effective immediately. An A2C agent would perhaps place its hand on the hot plate thousands of iterations under the guise of “exploration”. The result: slow model convergence and/or unsafe actions, experienced by almost all modern reinforcement learning algorithms (Ghiassian et. al, 2020). Learning for these models can be made more efficient by recognizing these fundamental differences in learning mechanisms.

Operant Conditioning

In psychology, operant conditioning explores how humans learn and displays the cognitive and behavioral differences in learning from positive stimuli, negative stimuli, and lack thereof (Grison & Gazzaniga, 2019). While operant conditioning is criticized as an overly simplistic view of human learning, it serves to differentiate between negative reward and pain. In operant conditioning, *response cost punishment* describes the removal of a positive stimulus as opposed to *aversive punishment*, or the addition of a negative stimulus.



	Addition	Removal
Increase Behavior	**Positive Reinforcement** addition of positive stimulus increases desired behavior RL equivalent: + reward	**Negative Reinforcement** removal of negative stimulus increases desired behavior
Decrease Behavior	**Aversive Punishment** addition of negative stimulus decreases undesired behavior	**Response Cost Punishment** removal of positive stimulus decreases undesired behavior RL equivalent: - reward

Figure 2: A grid showing different operant conditioning punishment/reward delineations and what they mean, along with their RL “equivalent”

Numerous operant conditioning experiments demonstrate the differences between aversive punishment and positive reinforcement (Gershman, 2015; Kubanek et. al, 2015; Steel, 2016). These findings imply that there is a behavioral, and consequently, a neurological difference in aversive punishment and reward, and also between response cost punishment and aversive punishment. Consider a child with an obsessive desire to climb trees. Their parents may warn them about the dangers of falling and perhaps take away their video game privileges as a consequence for climbing (response cost punishment). However, it is likely that the stubborn child will continue climbing until they fall and break their leg. The painful broken leg (an aversive punishment) is a much faster and stronger conditioning response than losing gaming privileges. In other scenarios, such as doing homework and getting a problem wrong, aversive punishment is far less effective than response cost, as it may deter the child from attempting the homework in the first place. It is evident from these psychological principles that both types of punishment are required for efficient learning.

As our current concepts of pain in traditional reinforcement learning are based solely on reward, we only observe the equivalent response cost punishment and positive reinforcement without any use of aversive punishment and negative reinforcement. Next sections explore the neuronal differences in these phenomena (reward and pain) in greater depth.

Neurological Pain Pathways

Neurological Pain Pathways in the MaxPain Model

In this section, we examine the neurological evidence of dissociable processes in the prediction of punishment in action systems. In light of these findings, we examine a recent technique termed "MaxPain" that uses an RL framework to strike an equilibrium between punishment and reward prediction (See Discussion: Existing Literature and MaxPain). Several studies corroborate the MaxPain algorithm’s central tenet that positive and negative reinforcement have distinct but complementary effects on learning and ultimately converge in the brain’s striatum. Recent research indicates that it is capable of distinguishing between pain/punishment and penalty omission learning rates when utilizing TD-learning models to represent behavior during an avoidance learning task (Elfwing et al., 2017). In addition, there is mounting evidence that the decision-making processes of animals incorporate separate reward and punishment systems, calling into question the fundamental validity of this approach. These results not only give a theoretical foundation for understanding punishment in the brain in both health and sickness, but they also underscore the necessity for independent punishment prediction in RL. While it is true that pain can activate regions of the brain involved in our reward circuitry, additional variables impact pain perception and should be researched to improve the current simplified model (Schmidt et al., 2002).

Basic Circuitry of Pain

The mesolimbic reward circuitry, including VTA-to-NAc dopaminergic projections, modulates pain. Dopaminergic neurons produced in the ventral tegmental region project to the nucleus accumbens through the mesolimbic route (Russo et al., 2013). In rodents, prolonged pain triggers dopamine release in the NAc and painful events can rapidly excite the dopaminergic neurons in the VTA (Schmidt et al., 2002). Furthermore, the spinal cord delivers afferent nociceptive pain signals to the brain. The thalamus communicates with the main somatosensory cortex (S1, S2), the anterior cingulate cortex (ACC), and the insula. Similarly, the basal ganglia gets signals from the amygdala (Bushnell et al., 2013). A visual representation of the distinct differences in the reward and pain pathways are summarized in Figure 3 below.

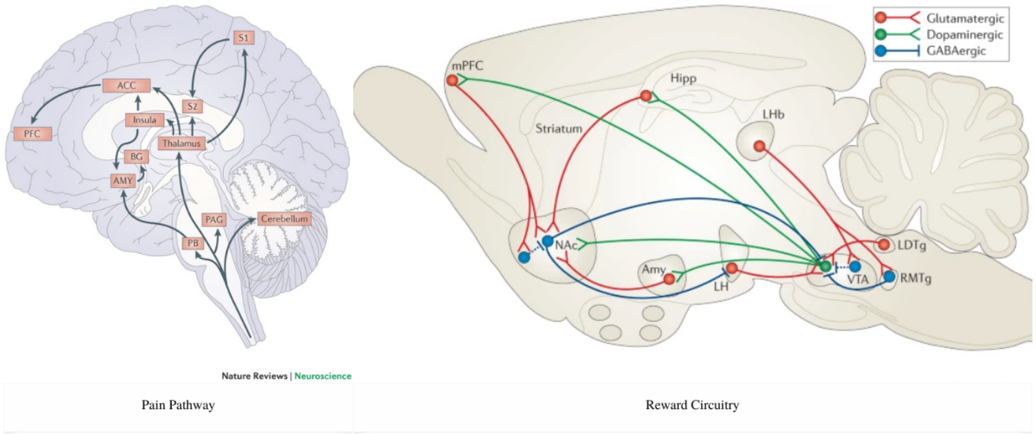


Figure 3: A contrast of the neural pathways controlling painful and rewarding sensations respectively. Acute pain begins at nociceptors – specific receptors of somatic neurons that detect noxious stimuli apart from other stimuli. One of the first modules of the pain pathway in the brain is the thalamus; typical reward circuitry does not involve the thalamus.
(Left) The spinal cord sends afferent nociceptive pain information to the brain. The pathway covers several brain regions; the primary somatosensory cortex (S1, S2), anterior cingulate cortex (ACC), and insula receives information from the thalamus. Similarly, the basal ganglia receives information from the amygdala (Bushnell et al., 2013).
(Right) The mesolimbic pathway shows dopaminergic neurons made in the ventral tegmental area projecting to the nucleus accumbens (Russo et al., 2013).

To examine the neurological basis of RL in the human brain, researchers rely on functional magnetic resonance imaging (fMRI), which permits the non-invasive monitoring of neural activity correlations. Temporal difference models reflect neurophysiological data like fMRI scans, and Pavlovian reinforcement learning works in a way that is comparable to reinforcement learning (Niv, 2009). Investigations of the neural circuit employing fMRI on avoidance revealed that action learning may be properly represented using basic temporal difference action-learning models (Sutton, 1988), with consistently identifiable prediction errors in dorsal striatal areas (Kim, 2006). Experiment results revealed an increase in activity in the medial orbitofrontal cortex, a region previously associated with the storing of the value of sensory pleasures, after individuals averted a negative outcome and were rewarded. Furthermore, as demonstrated in rats, dopamine treatments increase reward responses but not avoidance responses, indicating that the two ostensibly contradictory actions have distinct neural substrates (Fernando et al., 2013). Also, shock elicited much larger responses in the striatum than no-shock, indicating that learning may happen through punishment (Eldar et al., 2016). Overall, prediction errors that converge to the striatum are "reward-signed" in those who learn mostly from omission events and "punishment-signed" (aversive) in those who learn largely from punishment. This provides conclusive evidence that many action-value signals, including those for reward and punishment, converge on the striatum to govern behavior. Convergence of avoidance and reward acquisition values has been explored utilizing separate mixed reward-punishment schemes, and the outcome was independent. Studies from O'Doherty and colleagues (2004) revealed that fMRI correlates of prediction error signals may be distinguished in the dorsal and ventral striatum based on whether active decision behavior is required to obtain a reward related to Pavlovian conditioning. During the active choice task, the reward prediction error was observed in both the ventral and dorsolateral striatums, but during the passive prediction-learning task, it was only observed in the ventral striatum. These results supported a previously proposed Actor/Critic architecture in the basal ganglia. The ventral striatum, according to this hypothesis, comprises a prediction-learning Critic, whereas the dorsal striatum has a policy-learning Actor (Joel et al., 2002). These results support the Max Pain model.

Thus, the interpretation of pain in MaxPain is reliable, but due to the subjective nature of pain and the complexity of the field, we believe there are many alternative ways to accurately model high level concepts of pain in RL. Individual differences in pain processing add another layer of complexity to the pain pathway, which extends beyond the convergence of reward and punishment signals on the striatum. In spite of many behavioral parallels between not receiving a reward and being given a punishment, these two events appear to be unique in terms of prediction learning, and the substrate for unpleasant prediction learning is still unknown. It is important to note, however, that these studies do not equate the mechanism of learning from painful stimuli to that of reward stimuli. Although reward is clearly intertwined in pain/pleasure, physiological response, learning rate, observed behavior, and our own experience draws a clear distinction between the two. Furthermore, signaling or prediction mistakes for negative outcomes do not always appear to involve dopaminergic neurons (Mirenowicz & Schultz, 1996), despite the fact that they indicate negative prediction errors due to the absence of appetitive events (Bayer et al., 2007).

Using a unique brain mapping method, Kohoutová and colleagues located regions of the brain that show either high or low inter-individual variability in relation to pain. In addition to the anterior midcingulate cortex, the dorsolateral prefrontal cortex, and the cerebellum, twenty-one other brain areas have been shown to have a role in pain prediction as well. And contrary to common assumption, electropharmacogram analysis of brain recordings reveals that punishment prediction errors have been recorded in several brain areas, including the insula cortex, co-occurring with and with opposite sign to reward (Pessiglione et. a., 2006). Given the preexisting psychological data described in the Operant Conditioning section of this research, the variety of the areas engaged in pain processing indicates even greater variability than the negation of reward.

Moreover, pain is highly subjective, making direct measurement difficult; instead, we must rely on self-report and, to some extent, behavior to make sense of it. Variations in cerebral activity caused by the same painful stimuli corroborate self-reported pain differences and are predicted by brain morphology (Coghill, 2003). High levels of individual variability were found in the ventromedial prefrontal cortex, whereas lower amounts were seen in the posterior midcingulate cortex, implying that these regions' contributions to pain vary greatly among people. Analyzing the brain regions collectively as opposed to independently (i.e. multivariate analysis) yielded the same results. Individual variance was highest in the ventrolateral, vermis, and ventromedial prefrontal cortex. Individually, the posterior midcingulate cortex, the supplementary motor area, and the sensorimotor cortex were the most stable regions. Intriguingly, these findings were confirmed by tests performed with a completely new set of data. Collectively, these findings show that the relationship between brain regions and pain perception at the level of the individual is more complex than it is often portrayed at the level of the group. As a result, the fact that pain is experienced differently by different people demonstrates how subjective data may be used to refine algorithms.

Both animals and humans are able to solve in online, generalized, and sample efficient manners despite the fact that real-time neural computation is severely limited; this suggests neural mechanisms can be a source for new theoretical approaches, such as modifications to improve computational efficiency and mechanisms for interacting with constant and noisy sensory experience.

We have discussed the basics of dopaminergic learning and how it relates to the A2C model from RL literature through the concept of TD-learning. We then reviewed the difference between response cost punishment and aversive punishment in the operant conditioning subsection. The former is equivalent to negative reward but is imprecisely considered punishment in the current RL paradigm. Aversive punishment more accurately portrays pain in human learning. In the Neurological Pain Pathways section, we concluded our justification of separable and unique pain and reward pathways. This is the basis for the justification of our proposed architectures. Now, we move to the discussion where we analyze existing literature and how we can improve upon the paradigm through the incorporation of the empirical and neuronal differences of pain and reward.

Discussion

Typical RL algorithms do not incorporate learning from pain. Furthermore, learning from an action resulting in negative reward, or response cost, mirrors neither observed human behavior in aversive punishment nor the neural circuitry involved in processing pain. This is in contrast to the rough parallels of reward based operant conditioning to reinforcement learning via reward prediction error. Therefore, we review several approaches to incorporate pain with reinforcement learning, finally proposing alternatives and expansions to form a landscape of pain in RL which we call deinforcement learning.

Before beginning, it should be noted that in environments where reward is uniformly distributed across all possible outcomes, such as binary right/wrong object classification, there is no purpose to learning a function to approximate pain. The purpose of pain is to learn to avoid certain states much more vehemently and faster than learning from a lack of a reward in the same situation. If misclassifications are always weighted uniformly, there is no distinction between pain and a lack of a reward.

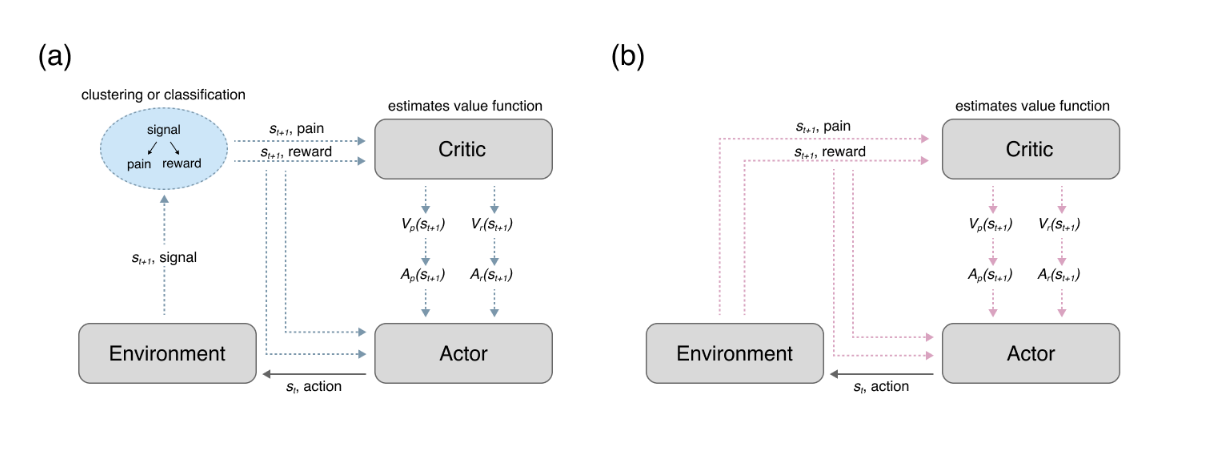


Figure 4: Possible Implementations of Pain in A2C
(a) Implementation option where the environment outputs a single feedback signal (as in traditional A2C) and this signal is split into pain and reward of various magnitudes. (b) Example where pain and reward are separate signals from the environment established through rewards-shaping.

Existing Literature and MaxPain

Existing research in the complement to our argument—understanding pain through RL—is thoroughly explored in a paper titled “Pain: A Precision Signal for Reinforcement Learning and Control”, which outlines the “underlying computational architecture of the pain system” (Seymour, 2019). They formalize the concept of pain in a high-level computational model that has a basis in RL. However, it does not apply these parallels as a pain signal in RL. More concretely, another paper titled “Parallel Reward and Punishment Control in Humans and Robots: Safe Reinforcement Learning Using the MaxPain Algorithm” focuses on using pain signals in RL to avoid physical damage. The relative “safety” of this algorithm makes it useful for robotics applications, where physical damage can occur in undesirable states and therefore the agent (robot) must learn to avoid what are perceived as “painful” states (Elfwing & Seymour, 2017). Yet another paper expands on the work of Elfwing and Seymour by implementing a similar architecture with multiple Q values, and learning entirely separate value estimations for the two (Lin, 2019). They dubbed this “split-Q learning”. Both of these papers show that considering pain in some form as a control signal results in faster convergence and more complex behavior. We argue that the MaxPain implementation can be improved upon to increase its efficacy as well as its accuracy in modeling pain.

The MaxPain paper implements pain in RL by splitting the reward scalar based on its sign. Any positive rewards remain positive. Negative rewards are inverted to be positive and are now considered “pain”. These separate signals are then evaluated by distinct networks. To estimate these values with respect to the state of the agent, the MaxPain architecture considers two distinct state-action-pair-value estimation functions (Q-functions). One is for estimating the accumulated discounted reward for taking action “a” in state “s” and thereafter following policy π. The other estimates pain in the same way. These two resulting estimations are linearly combined into one objective through the equation shown below:

$$Q_w(s,a) = wQ_r(s,a) - (1-w)Q_p(s,a)$$

Here, w is the weighting factor between 0 and 1. From this point onward, the goal of the MaxPain algorithm is to maximize that combination of reward minus pain through following a policy. It is implicit that finding this policy allows the agent to “solve” the problem in the environment (i.e balancing the pole in cartpole, finding the exit to a maze, etc.). The MaxPain model saw “significantly safer exploration, as well as effective learning and near-optimal long-term performance” (Elfwing & Seymour, 2017). The average learning curves presented for a “dangerous grid world search” task indicate faster convergence to a solution as well. Next, based on a neuroscientific foundation, we compare these approaches to existing RL methodologies.

Representing Pain

There appears to be two possible ways to represent pain in the context of reinforcement learning. One must distinguish between painful and non-painful states. This can either be the job of the environment, or the agent. In the former proposition, painful states may be a completely separate input, labeled as painful or not by the environment itself. For example, in the context of the game chess, the environment could send painful signals when pieces are lost, and reward when pieces are captured. This type of approach is seen in the current practice of reward shaping, but lacking a concept of pain (though the reward could have a negative sign). If we take this approach, then there is now a scalar describing the painfulness of a state and we need to augment the state to contain this new knowledge. This will allow us to describe to another network the painfulness of a state. One possibility is to use the positional encoding technique used by Vaswani et al. (2017) in "Attention is All You Need".

But this is not how humans perceive pain as the universe does not define pain for us. Fundamentally, it is the latter approach, the one that passes a raw state to the agent and allows the agent to interpret what is reward and what is pain, that is biologically inspired. We can see this by studying the body's path of pain: nociception.

Nociception defines a distinction between cells that can receive painful input and those that do not. When you touch something, a signal is propagated along mechanoreceptors. If you touch something hard enough, pointy enough, or hot enough, the signal propagates along pain specific fibers (e.g. A-delta fibers, C-fibers) to signal an acute pain to the brain (Yam et al., 2018). As previously discussed, this pathway is disparate from that of an unpainful signal. The mechanisms that perceive pain and other stimuli fundamentally represent the state differently, before interpretation in the brain. How would this look in reinforcement learning? This may take the form of a clustering algorithm whose clusters represent painful vs rewarding stimuli and various interpolations of those classes. After clustering into discrete signals, the pain and reward signals could be processed and interpreted with different learning mechanisms as they are in the brain. It may take the form of a classification neural network (or support vector machines among other algorithms), whose logit probabilities can be interpreted as dimensions along various sensory stimuli such as touch, pain, temperature etc. This leaves the state open to interpretation by multiple perception pathways; you can not only feel pain when pricked by a needle, you can also feel pressure. There are many other possibilities for representing this distinction between pain and other sensory information at the initial reception at the sensory level.

Interpreting and Learning from Pain

Next, how does one interpret pain within the RL equivalent of a brain? In our working example, the A2C method (see Reinforcement Learning section) uses the critic to evaluate how valued a state is with respect only to estimated future reward. Operant conditioning shows that reward and pain pathways trigger learning at different rates and to varying effect. Therefore, it is necessary to have different representations of $V(s)$ with respect to pain, and with respect to reward. This allows learning to be modulated according to the painfulness of experiences. One way to achieve this is to first modify the state with information from the pain classification processing mentioned above. Then this information is passed to the critic, whose weights should learn a representation of $V(s)$ which estimates and takes into account both the estimation of future pain and reward, then outputs two values of $V(s)$ with respect to both pain and reward (Figure 5). This is similar to the framework by Elfwing and Seymour (2017), but instead learns the weighting factor w of combining $Q(s,a)$ outputs. It also relies on an overparameterized critic network which converges to two nodes at some intermediate layer rather than using two separate critics.

Alternatively, the modified state information is passed to the critic, which must learn to accurately represent the state as estimations of $V(s)_{pain}$ and $V(s)_{reward}$, output these two different values, and pass these values to the actor. The actor can then learn weights to combine pain and reward into a customary single $V(s)$ rather than the two layer output in Figure 5. This shifts the burden of estimating $V_{pain}(s)$ and $V_{reward}(s)$ to the actor, as in Figure 6. Similarly, this learns the weighting factor w to combine the different estimated values of state rather than linearly combining them as originally tested by Elfwing and Seymour (2017).

Finally, another possible implementation is to train separate critic algorithms after binary classification of pain or nonpainful stimuli. This can be likened to ensemble approaches. This approach is the least biologically faithful, since states are not best represented by such a binary classification. The complicated dopaminergic ties to pain were explored by the Neurological Pain Pathways section, and demonstrate that discretizing the pain and reward pathways by using entirely different critics is also not entirely biologically accurate. However, despite its shortcomings, any incorporation of a pain estimate is likely more accurate than none. Moreover, this approach, taken by Elfwing and Seymour (2017), as the basis of the MaxPain architecture, showed faster convergence and safer behavior.

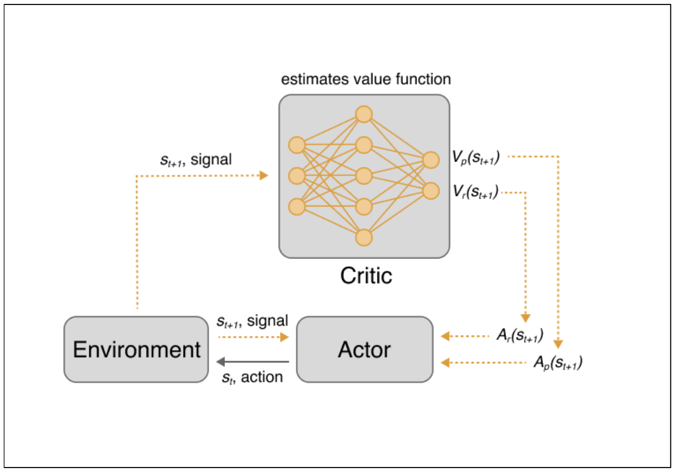


Figure 5: Example Critic with $V_{pain}(s)$ and $V_{reward}(s)$ Outputs
Fig. 5 extends Fig. 4 with an additional example implementation of pain in A2C, wherein the output layer of the critic has two nodes in order to explicitly represent the approximated $V(s)$ with respect to both pain and reward. The network's overall size could have any number of layers and nodes, but in an explicit representation of $V_{pain}(s)$, the output has at least two nodes in order to represent these values.

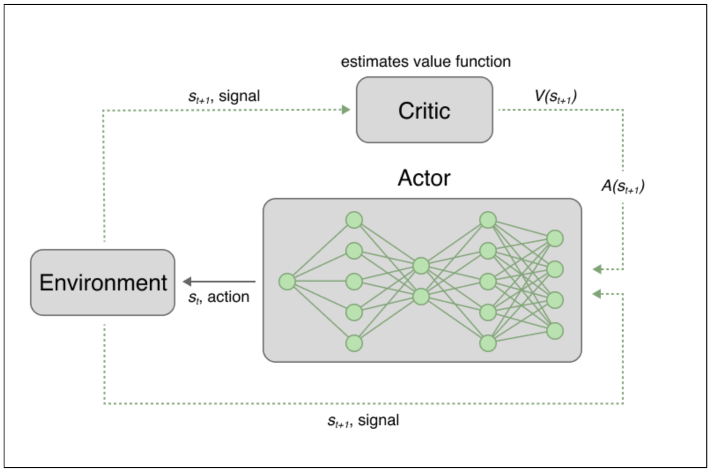


Figure 6: Example Actor with $V(s)_{pain}$ and $V(s)_{reward}$ in Third Layer
Fig. 6 depicts an example wherein the critic only outputs one value as in the typical A2C paradigm (contrary to Fig. 5). Thus, the burden of representing $V(s)_{pain}$ is shifted onto the actor, which at some point should have a two node layer as shown. There is no exact recommendation for in which layer this compression should occur, but in this example it is depicted in the third layer.

As shown, reward and motivation to escape is caused by painful stimuli. To mirror this type of aversion, one could use concepts like memory buffers, and algorithms may learn to associate the end of painful states with higher values, increasing the expected reward term as in TD learning.

Valuing and Weighting Pain

In order to delve deeper into the MaxPain implementation of pain valuation in reinforcement learning. we will continue to refer to Seymour's (2019) investigation of pain. Seymour states that "it is clear that pain is constructed not only from nociceptive input, but also from a set of cortical and subcortical components that compute the effective magnitude of pain as a control signal" (Seymour, 2019, p. 1036). As discussed, MaxPain takes the latter into consideration through a fixed w (weighting) variable. The closer to 1 this w variable is, the less that pain is considered in the final Q_w result and vice versa. There is no change to how pain is weighted in the agent's "mental model" as it learns or based on state context. The hyperparameter w is set before training.

The figures in the MaxPain paper illustrate how modifying this weighting factor affects how the agent performs in the “dangerous grid world” task. In this task, crashing into the wall is considered “painful” and the frequency of that occurrence should be reduced. The authors tested various levels of pain aversion by setting w to 0.1, 0.5, and 0.9. The figures in the paper present the tradeoff; models that are heavily pain-aversive will be more “careful”, usually at the cost of solution efficiency (Elfwing & Seymour, 2017). In this case, fixing w and explicitly modeling different values made the results of the MaxPain algorithm more interpretable and clear.

Alternatively, one way to modify the weighting factor is discussed in “Modular Deep Reinforcement Learning from Reward and Punishment for Robot Navigation” (Wang et al., 2021). The authors of the paper proposed a Boltzmann distribution-based selection mechanism for finding weighting factors that are applied to separate reward and pain optimization policies to find a joint policy. The weighting factor is dependent on a state-evaluation function $V(s)$ in an interesting way. There is a temperature variable τ_w that determines how evenly mixed the w^+ and w^- variables are. The way temperature and $V(s)$ affect the weighting factors are shown below:

$$\text{if } \tau_w \to \infty,\, w^+(s) = w^-(s) = 0.5$$

$$\text{if } \tau_w \to 0,\, w^+(s) = 1 \text{ and } w^-(s) = 0 \text{ when } V^+(s) \geq V^-(s)$$

$$w^+(s) = 0 \text{ and } w^-(s) = 1 \text{ when } V^+(s) \leq V^-(s)$$

The latter, where τ_w is zero, is called hard-max weighting. After experimentation, the authors concluded that “Deep MaxPain with hard-max weighting achieved the best overall performance” compared to fixed weights and standard DQN because it “utilized real-time assessments for weighting two sub-policies” (Wang et al., 2021, p. 125). However, pain and reward in the real world are not binaries to choose from when considering a policy to follow. There is always an influence of both future reward and pain when choosing actions in a state. As environments grow more complex, so should the considerations that affect the weighting of the policy. We propose that a separate predictor network be used to provide the weighting variable at each timestep. Making the weighting variable dynamic and learned across timesteps may have several advantages. Firstly, it more accurately models the aforementioned “set of cortical and subcortical components that compute the effective magnitude of pain” (Seymour, 2019, p. 1036), since neurons are not fixed hyperparameters but instead dynamic and context-dependent mechanisms of learning. Moreover, it allows for more complex, higher-level behavior where an agent has to decide if the “pain is worth the gain”, depending on the context of the state.

As for the inputs of the weighting variable network, it could consider either the current state, a state memory buffer, time spent in the current training episode, time spent until the episode terminates, the level of “damage” the agent has already sustained, or any further possibilities and combinations of relevant information. The output would be a normalized scalar between 0 and 1 used to linearly combine Q_r and Q_p in the case of MaxPain, or two separate sub-policies in the case of Deep MaxPain. Regardless of this specific proposed implementation, allowing the weighting variable to be dynamic and context-dependent will lead to more balanced and adaptable behavior in a MaxPain agent.

Designing or Discovering Painful Stimuli

Finally, in returning to general mechanisms of deinforcement learning, there remains the question of what is painful. Secondarily, how do we construct a state provided by the environment that might allow us to learn what is painful? This may be the most difficult component of deinforcement learning. For humans, this is initially partially encoded by genetics. We learn what is painful through evolutionary genetic iterations. Each iteration we approximate reward such that fitness increases. Similarly pain can approximate behaviors to avoid such that fitness increases. Along with genetics, what is painful changes throughout a lifetime; it is learned through experiences as well as internally modulated through complex top-down modulatory pathways, beyond the scope of the neuroscience described in this paper.

The most literal machine learning analog to the genetically encoded aspect of pain might be an actor which can reproduce or spawn new networks with its learned weights. Here we suggest applying genetic algorithms to the context of deinforcement learning agents, providing a method to allow networks to learn what should be considered painful. In such an implementation, the probability of reproduction in this context correlates with the problem that the network is trying to solve. The network also needs a reward heuristic, such as time alive or reproductive success. It learns as described above, making estimations of how painful or rewarding the environment might be. If the pain estimate is not correct, this negatively affects the network’s predictions, making it less likely to reproduce. If an inaccurate pain estimation led to a very low performing model, the agent may be deactivated, or effectively killed. Thus, a successful, fit network should learn to define pain in a similar manner to humans – that which should be avoided for the sake of reproducing the network. These reproductive odds defined within the genetic algorithm provide a separate signal to learn from besides the immediate-term reward signal and potential pain signal.

Concretely, if the network should learn to make a stick figure walk like in OpenAI's MuJoCo framework, then reproductive success may be set as a function of time spent walking versus energy expended. Networks that avoid fatal falls or expend less energy in their movement should have a higher probability of reproducing or replicating their weights in new networks. Here, an initial sensory layer as discussed in the "Representing Pain" section would group together similar states and outcomes based on their features, modulate the given state of the environment to hold this painful information, then finally pass it to a critic. The critic estimates the value of the state with respect to potential pain and reward. Alternatively, two critics could be used similar to the MaxPain architecture. Then, if the agent were near a box it might trip over, the current sensed pain may be 0, but the critic may weigh future states as very painful and $V_{pain}(s)$ very low. Its estimation $V_{pain}(s)$ would be tuned as it attempts to walk, and further tuned as it replicates its weights in other agents based on its walking success.

One may not find a need for literal analogs such as genetic algorithms. In the above case, one implicitly defines pain merely by defining what success is. In the MuJoCo example we did this – pain was implicitly that which must be avoided to achieve success of walking, for example, tripping. However, a model can receive a single signal (like reward in the current paradigm) and learn pain aversive behavior. Crucially this must, at some point in processing, interpret the signal with respect to pain and pleasure, which together update the model's policy in dissimilar ways, just as the proposed algorithms in Figure 4 describe, or the algorithms discussed in "Existing Literature and MaxPain".

Those familiar with RL may now be wondering: initial clustering algorithms, modified critics, and learned hyperparameters – are these really a significant change from standard RL practices? The key difference is not only in these simple, fundamental algorithmic changes but in coupling them with environmental changes conducive to pain aversive learning. As initially described, without an environment where correctness or incorrectness is non-uniformly distributed, pain is not a useful concept. This means that to test the incorporation of pain processing with something as simple as an MNIST classifier, one would need to quantify how close each classification was to the correct classification, and build that into the reward or ground truth. Ultimately, there are many possible scenarios where pain would increase convergence or safety. Anywhere where particular states must be avoided more than a typical "failure" is a good application of deinforcement learning.

Conclusion

We have seen that reinforcement learning and neuroscience are intricately intertwined, beginning with their overlapping uses of reward prediction error and TD learning. Continuing to draw inspiration from the brain and body to enhance modern RL algorithms is a fruitful frontier. Many areas of how humans learn from pain are yet to be investigated; the role of emotional pain and trauma was not examined in this paper. However, the growing neuroscientific body of knowledge on pain allows us to examine the phenomena as a model for novel RL algorithms. Thus, we conclude new RL models which learn pain aversive behavior are necessary to propel the field towards more realistic, safe, efficient learning paradigms.

Acknowledgments

This work was created by the Interactive Intelligence team, a research group at the University of Washington fusing neuroscience and computer science to make machines learn like humans. We would like to thank the I2 advisor, Dr. Eric Chudler, for his support and guidance.

Thank you to everyone on the team, and everyone on and off it who helped make this paper and team possible.

Glossary

Aversive Punishment – addition of negative stimuli

Response Cost – removal of positive stimuli

Positive Reinforcement – addition of positive stimuli

Reward (RL) – Scalar value provided to a reinforcement learning agent at each timestep, usually crafted in such a way that maximizing discounted reward through time lets the RL agent accomplish a goal

Deep Neural Network – A multilayer artificial neural network (ANN), often fully connected across adjacent layers

Weights (RL) – In relation to machine learning, parameters within a neural network that transform input data within the ANN's layers. Adjusting/tuning weights results in a performance change to the network.

References

Arias-Carrión, O., Caraza-Santiago, X., Salgado-Licona, S. et al. (2014). Orquestic regulation of neurotransmitters on reward seeking behavior. *Int Arch Med*, 7, 29. https://doi.org/10.1186/1755-7682-7-29

Bayer, H. M., Lau, B., & Glimcher, P. W. (2007). Statistics of midbrain dopamine neuron spike trains in the awake primate. *J Neurophysiol*, 98(3), 1428–1439.

Beeler, J. A., Daw, N., Frazier, C. R., & Zhuang, X. (2010). Tonic dopamine modulates exploitation of reward learning. *Frontiers in behavioral neuroscience*, 4, 170. https://doi.org/10.3389/fnbeh.2010.00170

Bushnell, Mary & Čeko, Marta & Low, Lucie. (2013). Cognitive and emotional control of pain and its disruption in chronic pain. *Nature reviews*. Neuroscience, 14. https://doi.org/10.1038/nrn3516.

DosSantos MF, Moura BS and DaSilva AF. (2017) Reward Circuitry Plasticity in Pain Perception and Modulation. *Frontiers in Pharmacology*, http://dx.doi.org/10.3389/fphar.2017.00790

Edlar, E, Hauser T. U., Dayan P., and Dolan R. J. (2016) "Striatal structure and function predict individual biases in learning to avoid pain," *Proceedings of the National Academy of Sciences*, vol. 113, no. 17, pp. 4812–4817

Elfwing, S., Seymour, B. (2017). Parallel reward and punishment control in humans and robots: Safe reinforcement learning using the MaxPain algorithm. *2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, 140-147, https://doi.org/10.1109/DEVLRN.2017.8329799.

Fernando A., Urcelay G., Mar, A., Dickinson, A. (2013) and Robbins, T., Comparison of the conditioned reinforcing properties of a safety signal and appetitive stimulus: effects of d-amphetamine and anxiolytics, *Psychopharmacology*, 227(2) 195–208

Geron, A. (2019). Hands-on machine learning with scikit-learn, keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems (2nd ed.). O'Reilly Media.

Gershman, S.J. (2015). Do learning rates adapt to the distribution of rewards?. *Psychon Bull Rev*, 22, 1320–1327. https://doi.org/10.3758/s13423-014-0790-3

Ghiassian, S., Patterson, A., Garg, S., Gupta, D., White, A., & White, M. (2020). Gradient Temporal-Difference Learning with Regularized Corrections. *ICML*, https://doi.org/10.48550/arXiv.2007.00611

Grison, S., & Gazzaniga, M. S. (2019). Psychology in Your Life (3rd ed.). W.W. Norton.

Hauser, T. U., Eldar, E., and Dolan R. J. (2016) "Neural mechanisms of harm-avoidance learning: A model for obsessive-compulsive disorder?" *JAMA psychiatry*, 73,(11) 1196–1197

Joel, D., Niv, Y., & Ruppin, E. (2002). Actor-critic models of the basal ganglia: New anatomical and computational perspectives. *Neural Networks*, 15, 535-547.

Kim H., Shimojo S., and O'Doherty J. P. (2006) Is avoiding an aversive outcome rewarding? neural substrates of avoidance learning in the human brain, *PLoS Biol*, 4(8) 233

Kubanek, J., Snyder, L. H., & Abrams, R. A. (2015). Reward and punishment act as distinct factors in guiding behavior. *Cognition*, 139, 154–167. https://doi.org/10.1016/j.cognition.2015.03.005

Lin, B., Cecchi, G., Bouneffouf, D., Reinen, J., & Rish, I. (2019). A Story of Two Streams: Reinforcement Learning Models from Human Behavior and Neuropsychiatry.

Mirenowicz, J., & Schultz, W. (1996). Preferential activation of midbrain dopamine neurons by appetitive rather than aversive stimuli. *Nature*, 379, 449-451.

Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D. &amp; Kavukcuoglu, K. (2016). Asynchronous Methods for Deep Reinforcement Learning. *Proceedings of The 33rd Interna-*

tional Conference on Machine Learning, in Proceedings of Machine Learning Research, 48, 1928-1937 https://proceedings.mlr.press/v48/mniha16.html

Niv, Y. (2009) “Reinforcement learning in the brain,” *Journal of Mathematical Psychology*, 53(3) 139–154

O’Doherty, J. P., Dayan, P., Friston, K., Critchley, H. and R. J. Dolan, (2003). “Temporal difference models and reward-related learning in the human brain,” *Neuron*, 38(2) 329–337

Perelman School of Medicine at the University of Pennsylvania. (2014). Human learning altered by electrical stimulation of dopamine neurons. ScienceDaily. ScienceDaily, www.sciencedaily.com/releases/2014/05/140513175006.htm>

Pessiglione M., Seymour B., Flandin G., Dolan R. J. and Frith C. D. (2006) “Dopamine-Dependent Prediction Errors Underpin Reward-Seeking Behaviour in Humans,” *Nature*, 442(7106) 1042-1045 doi:10.1038/nature05051

Russo, S., Nestler, E. (2013). The brain reward circuitry in mood disorders. Nat Rev Neurosci, 14, 609–625. https://doi.org/10.1038/nrn3381 Seymour, B. (2019). Pain: A Precision Signal for Reinforcement Learning and Control. *Neuron*, 101(6), 1029-1041. https://doi.org/10.1016/j.neuron.2019.01.055

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science (New York, N.Y.)*, 275(5306), 1593–1599. https://doi.org/10.1126/science.275.5306.1593

Steel, A. (2016). The impact of reward and punishment on skill learning depends on task demands. *Nature*, https://www.nature.com/articles/srep36056

Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D. de L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., Lillicrap, T., & Riedmiller, M. (2018). DeepMind Control Suite (Version 1). *arXiv*, https://doi.org/10.48550/ARXIV.1801.00690

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 5998–6008.

Wang J, Elfwing S, Uchibe E. (2021). Modular deep reinforcement learning from reward and punishment for robot navigation. *Neural Netw*, 135(115-126). https://doi.org/10.1016/j.neunet.2020.12.001

Watabe-Uchida, M., Eshel, N., & Uchida, N. (2017). Neural Circuitry of Reward Prediction Error. *Annual review of neuroscience*, 40, 373–394. https://doi.org/10.1146/annurev-neuro-072116-031109

Wood P. B. (2006). Mesolimbic dopaminergic mechanisms and pain control. *Pain*, 120(3), 230–234. https://doi.org/10.1016/j.pain.2005.12.014

Yam, M. F., Loh, Y. C., Tan, C. S., Khadijah Adam, S., Abdul Manan, N., & Basir, R. (2018). General Pathways of Pain Sensation and the Major Neurotransmitters Involved in Pain Regulation. *International journal of molecular sciences*, 19(8), 2164. https://doi.org/10.3390/ijms19082164

Yin, H. H., Knowlton, B. J., & Balleine, B. W. (2004). Lesions of the dosolateral striatum pre- serve outcome expectancy but disrupt habit formation in instrumental learning. *European Journal of Neuroscience*, 19, 181-189.