A Reinforcement Learning Model for Fear Reconsolidation and Exinction During Dreaming

Wenjie Li (wenjieli@nyu.edu)
Courant Institute of Mathematical Sciences, New York University

Monika Dagar (md4676@nyu.edu)
Courant Institute of Mathematical Sciences, New York University

Ziheng Chen (zc2068@nyu.edu)
Finance and Risk Engineering Department, New York University

Abstract

We present a simulation of fear reconsolidation and extinction in dream using reinforcement learning in the Pac-Man framework. Our computational model is based on the hypothesis that mismatches of memory units in dream facilitate regulations of fear memories. We use experience replay in reinforcement learning to model the mechanism of hippocampal replay during sleep. Our experiments show that fear regulation in dream is possible, and the brain could smoothly switch between fear reconsolidation and extinction in respond to environmental changes via a dynamically updated mismatch probability.

Keywords: dream, reinforcement learning, experience replay, hippocampal replay, fear extinction, emotion regulations

Introduction

Dreaming is important for learning and emotion regulations. Researchers have found strong reactivation of communications between the hippocampus and the basolateral amygdala in rodents during sleep, in correspondence with contextual emotional memory consolidation. Nightmares make up a disproportional amount of dreams in one's life. These dreams usually involve people's everyday concerns, stresses, or traumas. They are particularly frequent and prevalent in the psychiatric population (Lin, 1992). PTSD patients have more replicative flashback-like nightmares together with strong amygdala activities, indicating activation of fear memories in dreams. The functions and mechanisms of nightmares are under scrutiny. Many studies suggest the utility in fear emotion regulation, in particular with fear extinction learning (Tibor Bosse & Treur, 2013). During sleep, fear memories acquired during the day are replayed, and under some boundary conditions, the brain may weaken or strengthen the connection between fear stimulus and response. The mechanism of how this process is administered is under debate (Van Izquierdo & Myskiw, 2016).

We propose a reinforcement learning model that simulates fear reconsolidation and extinction learning in dreams. We use the classical Pac-Man game as our basic framework, and show how Pac-Man's behaviors change after "dreaming." We model the dreaming process with experience replay in reinforcement learning, where the replay episodes are sampled from Pac-Man's interactions with ghosts. Our experiments show that replaying of fear memory help an agent regulate fear reaction to better suit the change in environment, which is shown to be beneficial for Pac-Man's performances. Our code is available in the GitHub repository. [1]

The remainder of this paper is structured as follows. In Section 2, an overview is given about the psychology basis behind fear reconsolidation and extinction in nightmares. Section 3 presents the implementation details of the Pac-Man game, with dream related features. Section 4 shows our experiments and results. Lastly, Section 5 discusses some results of the experiments that validate the model.

Background

Fear extinction and reconsolidation

Fear extinction is the decline in conditioned fear responses (CR) when there is a reduction in the predictive value of the conditioned stimulus (CS) as to the occurrence of the unconditioned stimulus (US). Losing fear is not the erasure of fear memory. In fact, it is commonly believed as an inhibitory learning process that disassociate the stimulus with the fear-related consequences (Myers & Davis, 2007). The most famous behavioral experiment that demonstrates fear extinction is Pavlov's dog. When the animals realize a reinforcement is no longer administered, a learned response (fear in this case) fades away.

Fear reconsolidation, on the contrary, is a process that strengthens the conditioned fear response and the stimulus. Unlike fear extinction which can be initiated anytime after the triggered learning event, fear reconsolidation can only be formed in a brief time window after the latest training session. Fear extinction and reconsolidation are commonly viewed as two sides of the same coin. Their occurrence utilize the same brain structure. It is believed that when a boundary condition is triggered, the brain can decide which course of actions to follow – whether to strengthen or weaken a learned response. However, it is unknown what the boundary condition is. Some propose the presence and removal of the CS is what triggers reconsolidation and extinction respectively, but it is inapplicable in cases where the boundary between the onset and offset of CS are vague (Van Izquierdo & Myskiw, 2016). We will present a simplified model that allows the agent to switch between fear extinction and reconsolidation. It will be discussed later on in this report.

[1] https://github.com/wliwenjieli/dreamer

Unlikely Combinations Nightmares often involve events that are bizarre and incompatible with wake-time experiences. Sometimes they are in the form of mismatching combinations of events or subjects that are unlikely to happen together in everyday life. The occurrence of mismatches usually happens with major shift in emotions in dreams. To the psychiatric population mismatches are more prevalent with nightmares, as opposed to mundane dreams that are similar to everyday activities. Some researchers find the aspect of unlikely combination to be the key to fear extinction in nightmares, though the details of how it happens are not well studied (Levin & Nielsen, 2007). In our project, we propose a computational model that specifies how mismatching enables fear extinction as well as reconsolidation. Specifically, our hypothesis is that mismatching introduces a new non-aversive outcome to a fear-eliciting cue that the agent has been previously conditioned to, and thereby forming a new conditioned connection between the stimulus and a non-fear response in place of the original fear response.

Hippocampal replay in reinforcement learning

It is recorded in multiple in situ experiments that in dreams, multiple regions of brains are reactivated that mimic daytime experience. Many cognitive neuroscience models declare hippocampal replays during sleep to be a crucial step for memory consolidation and learning. In replays, a mental map of the environment is simulated to guide future decisions by conducting both on-line and off-line explorations. This prospect of hippocampal replay draws a parallel with reinforcement learning, where policy-making is learnt through explorations. In particular, it shares a similar structure with experience replay, which is a process often incorporated with reinforcement learning to speed up learning by retraining an agent with data sampled from the original dataset. Different reinforcement learning replays may model different kinds of hippocampal replays. In particular, the model-free family of reinforcement learning algorithm enables optimization in decision making through trial-and-error, which is similar to on-line hippocampal replay. An example would be a mouse trying to find the food reward at the end of a maze. When it faces an intersection, the mouse explores the left path first and return to explore the right path if the first try was unsuccessful. The model-based approach on the other hand conducts off-line learning where the benefit of a decision is calculated before execution by knowledge of the model. In the same maze scenario, the mouse would mentally calculate the results of going each direction with its knowledge of the model and make a decision (Romain Cazé & Girard, 2018).

In most real-life scenario as well as in the Pac-Man game, the environment is too complex to be modeled, and therefore a model-free algorithm is often more realistic with human decision-making. As a result, we choose approximate Q-learning, a model-free approach, as the backbone of our reinforcement learning algorithm.

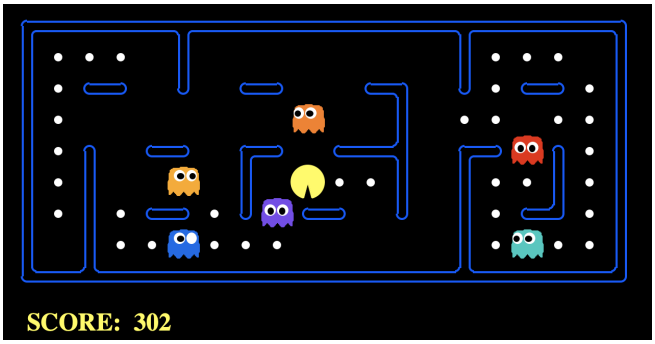

Figure 1: Gridworld

Stimulation with the Pac-Man Game

We use the Pac-Man framework developed by the Berkeley AI Lab [2]. This open-sourced project contains the basic set-up of a Pac-Man game, providing a perfect scaffolding for testing AI models. In our set-up, we create a reinforcement learning agent using approximate Q-learning and add an experience replay feature to model the dreaming process.

Basic set-up

Gridworld Gridworld is the environment where agents operate in. It is a rectangular maze with walls and food dots. Gridworld is a discrete world with each location represented by a state tuple $s \in \{(x,y) : x = 0,\ldots,width, y = 0,\ldots,height\}$. Figure 1 shows the Gridworld with one Pac-Man agent and six ghost agents.

Agent There are two kinds of competing agents: Pac-Man and ghosts. The agents can make action $a \in \{east, south, west, north\}$. A move is possible when the resulting new state s' of the state-action pair (s,a) is not a wall. Ghosts move randomly in the Gridworld and when they collide with Pac-Man, the latter will be attacked and lose points with probability $P_w(attack)$ where $w \in \{war, peace\}$ is the mode of the environment. A *Score* variable measures Pac-Man's performance in the game. For Pac-Man, the goal of the game is to maximize *Score*. Being attacked by a ghost makes Pac-Man loses 200 points, eating a food dot allows Pac-Man to gain 10 points, eating all the dots give Pac-Man another 200 points and the game finishes. In addition, Pac-Man loses 10 point for every time lapse it stays in an unfinished game.

Pac-Man knows the location of all the food dots. All agents know the distance between themselves and the others. All distance in the game is manhattan distance given by:

$$d = |i_1 - i_2| + |j_1 - j_2|$$

where (i_1, j_2) and (i_1, j_2) are the states two agents/food dots locate at.

Policy-Making

In our reinforcement learning model, emotions are simplistically made binary: negative emotion associated with events

[2] http://ai.berkeley.edu/project_overview.html

to be avoided and positive emotion related to rewards. More specifically, the expected value of an action made by an agent given the game state is determined by four features: fear of immediate danger – the number of ghosts one step away, fear of potential danger – the distance to the closest ghost, immediate reward – there is a food dot at the location the agent occupies, and potential reward – distance to the closest food dot. We use approximate Q-learning to maintain and update policy-making. Q-value $Q(s,a)$ is a function of a state-action pair that describes the expected reward for the rest of the game when an agent take an action a from state s

$$Q(s,a) = w_1 f_1(s,a) + \ldots + w_4 f_4(s,a) + w_5 b$$

where f_i for $i = 1,2,3$, and 4 are the features variables described above and b is a bias term. The goal of training is to find appropriate weights w_i that optimize Pac-Man's behaviors to maximize its score. The following pseudocoe describes the optimization of the weights (Russell, 2010). Note that α is the learning rate and γ is a discount factor.

Initialize $Q(s,a)$ arbitrarily
for *each episode* **do**
  Initialize s
  for *each step in an episode* **do**
    Choose a at s using policy derived from the current Q
    Take action a and observe r, s'
    $Q(s,a) \leftarrow$
     $Q(s,a) + \alpha[r + \gamma \max_{a'} Q(a',a') - Q(s,a)]$
    $s \leftarrow s'$
  end
end

Memory Replay

Fear Memory Units Sampling We use a replay buffer to store Pac-Man's fear memory. Every time Pac-Man is attacked by a ghost, three tuples (s,a,r,s') recorded from two steps before and up to the present attack are pushed into the replay buffer. We will call the collection of these three tuples an attack event. The fear memory buffer store all attack events from the entire training process.

The purpose of choosing exactly three tuples to make up a single event is for maintain the relative continuity of event episodes and facilitate planning in advance. An event is made no longer than three movements in order to model the fact that complete episodic memories do not typically appear in dreams. Rather, events are replayed as if fear memories were reduced to basic units (Levin & Nielsen, 2007).

Mismatching Replay The replay session trains Pac-Man with events randomly sampled from the replay buffer. Before training, the reward variable r in the last slice of an attack event – the slice at the exact moment that the attack happen – is modified to create the mismatch effect. Specifically, $r = -200$ with probability $P_{mismatch}(attack)$ and $r = 0$ otherwise, signifying Pac-Man is attacked by a ghost at a collision

```
Pac-Man grows up in peace for 200 episodes...
A ghost attacks Pac-Man with probability 0.2.
Average Score: 831
Pac-Man goes to war for 5 episodes...
A ghost attacks Pac-Man with probability 1.
Average Score: -203
Pac-Man dreams at war camp...
A ghost attacks Pac-Man with probability 0.6.
Pac-Man wakes up and go to war for 5 episodes...
A ghost attacks Pac-Man with probability 1.
Average Score: -2
```

Figure 2: Performance in a fear consolidation scenario

```
Pac-Man at warzone for 200 episodes...
A ghost attacks Pac-Man with probability 1.
Average Score: -76
Pac-Man lives in a peaceful world for 5 episodes...
A ghost attacks Pac-Man with probability 0.2.
Average Score: 281
Pac-Man dreams about war time...
A ghost attacks Pac-Man with probability 0.6.
Pac-Man wakes up in a peaceful world for 5 episodes...
A ghost attacks Pac-Man with probability 0.2.
Average Score: 466
```

Figure 3: Performance in a fear extinction scenario

or not, respectively. We ignore the effects of other rewards and penalties at this event because they are trivial comparing to the points loss from an attack. Essentially, we regenerate outcomes for a series of previously conditioned stimuli (colliding with ghosts) to achieve mismatch. The mismatch probability variable is dynamically sampled from the attack rate of ghosts in the latest 10 episodes of game before the time replay takes place.

Our hypothesis is that the mismatch probability determines whether fear reconsolidation or extinction take place. We believe the proportion of weights the reinforcement learning algorithm assigns to features related to avoidance of ghosts represents an intrinsic estimation of the probability of a ghost attack given its presence at the moment. If before dreaming, this probability is larger than $P_{mismatch}(attack)$, then Pac-Man learns to be less scared of ghosts in the dream, resulting in fear extinction, and vice versa. This agrees with the hypothesis that fear reconsolidation and extinction are both adjustments of fear reactions that can be interchangeable according to the environments.

Experiments

Procedure

We construct a peaceful environment and a war environment where a fear-eliciting cue is weakly and strongly presented, respectively. This is achieved by modifying the probability of attack when a ghost and Pac-Man collide. For our experiment specifically, we set $P_{peace}(attack) = 0.2$ in the peaceful envi-

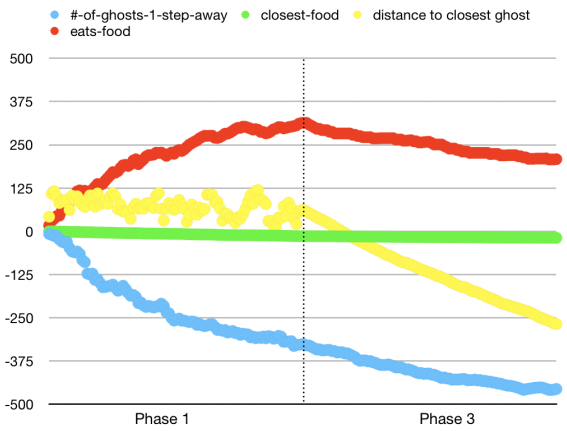

Figure 4: Change of feature weights during the training of a scenario of fear reconsolidation. Note: phase 1 the first training session that familiarizes Pac-Man with an environment and phase 3 is the dreaming process. Phases 2 and 4 are testing stages where change of feature weights do not apply. Same for below.

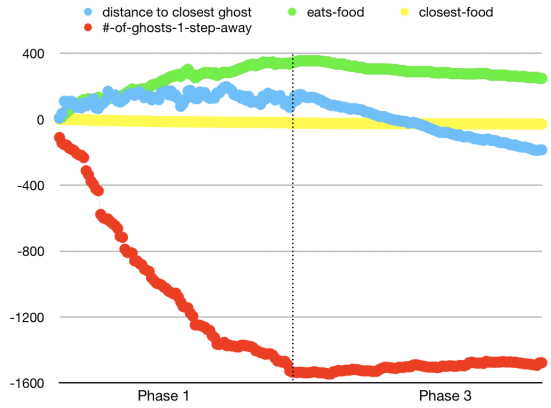

Figure 5: Change of feature weights during the training of a scenario of fear extinction.

ronment and $P_{war}(attack) = 1$ at the war environment. All other set-ups are the same for the two environments.

For fear reconsolidation simulation, we first train Pac-Man in a peaceful environment for 200 episodes. Then we test Pac-Man's performance in a war environment for 5 episodes in which Pac-Man is expected to perform poorly because it has not learnt to avoid ghosts adequately. We will then train Pac-Man with events sampled from the fear memory replay buffer with mismatch probability sampled from the probability of attack in the last 10 episodes. Finally, Pac-Man is tested in the war environment again for 5 episodes. We expect Pac-Man to develop more avoidanance behaviors towards ghosts together with an improvement in *Score* after the replay.

Similarly, for fear extinction simulation, we train Pac-man in a war environment for 200 episodes. Then we test Pac-Man's performance in a peaceful environment for 5 episodes in which Pac-Man is expected to perform suboptimally because of its inaccurate estimation of harm (irrational fear) from the ghosts. We train Pac-Man with events sampled from the fear memory replay buffer with mismatch probability sampled from the probability of attack in the last 10 episodes. Finally, we test Pac-Man in a peaceful environment again for 5 episodes. We expect the algorithm to weigh food-related reward features more and show improvement in Pac-Man's overall performance.

Results

For the fear reconsolidation experiment (see Figure 2), we see a high performance with an average score of 831 in a peaceful environment where Pac-Man's collision with a ghost only results in 0.2 probability of attack. However, the lack of avoidance behaviors towards ghosts results in poor performance when Pac-Man is in a war environment where ghosts attack Pac-Man more aggressively. The mismatch probability at replay sampled the chance of attack in the previous 10 games, 5 of which are in war environments and the rest 5 in peace environments, resulting in $P_{mismatch}(attack) = 0.6$. Replaying attack events with this probability results in an improvement in performance from an average score of -203 to -2 in the war environment. The plot for feature weights (see Figure 5) shows an increased preference to avoid immediate danger (#-of-ghosts-1-step-away). The motivation of eating food (eats-food) decreases. The weights corresponding to the distance to the closest food stays neutral throughout the entire experiment. Surprisingly, the weights regarding distance to the closest ghost changes sign during replay sessions.

With the fear extinction scenario (see Figure 3), we first see a poor performance by Pac-Man in the war environments due to overwhelming ghost attacks, resulting in an average score of -76 in the most recent training sessions. When Pac-Man is then tested in a peace environment, its performance improves due to reduced ghost attacks. However, we see that this performance is not optimal as we see an improvement in performance after the replay process which trains Pac-Man with old collision data with $P_{mismatch}(attack) = 0.6$. After the replay, Pac-Man scores 466 in a peaceful environment with

185 points in improvement. In the weights plot (see Figure 3), we notice a slight increase regarding the features #-of-ghosts-1-step-away, implying that Pac-Man does not react as aversively towards immediate danger, though caution is still presented. We also notice similar neutral weights associated with the distance to the closest food, a slight decrease in interest of eating food, as well as an interesting change of signs of the weights with distance to closest ghost.

Discussion

Our framework shows a simplified model of fear reconsolidation and extinction in dreams through mismatching fear memory replay. The experimental results show drastic changes in motivation in immediate danger or rewards in both fear reconsolidation and extinction scenarios. This result validates that fear regulation through mismatching experience replays is possible. Pac-Man's responses to long-term reward and danger stay relatively neutral during training, especially with long-term reward, indicating a lack of consideration in decision making under the game context. Both long-term features stay stable during phase 1 training. Interestingly, the weights corresponding to the feature that describes Pac-Man's distance to the closest ghost drastically decrease and change signs during replay, implying that Pac-Man "prefers" to stay close to ghosts. We suspect this is because in all sampled scenarios in the replay buffer Pac-Man is either in a collision with a ghost or about to have a collision with a ghost. In other words, its distance to the closest ghost is rather small comparing to the average scenarios. We are unsure how this aspect deflates a fear of ghosts in a long-term perspective specifically, but we believe they are related. More experiments are required for validation and for understanding its potential psychological implications.

In addition, our experiments show that fear reconsolidation and extinction are two sides of the same coin. In both the fear consolidation and fear extinction scenarios of our experiments, the mismatching probabilities are the same – $P_{mismatch}(attack) = 0.6$ as it sampled from the most recent 10 games. With the fear reconsolidation scenario, these 10 games consist of 5 games in the peaceful environment where Pac-Man is used to and 5 games in the new war environment. With the fear extinction scenario, they are 5 games with the war environment which Pac-Man is used to as well as 5 games with the new peaceful environment. Namely, the proportions of war and peaceful environments in the short-term memories are the same for the two scenarios. However, the replays in the two scenarios produce different effects: in the fear reconsolidation scenario, Pac-Man develops more fear towards ghost because the mismatch attack probability is higher relative to before the replay and vice versa. The dynamically updating mismatch probability allows a smooth transition between fear reconsolidation and extinction in our model. This discovery may shed light on how the brain decides which course to take in fear regulations in response to the environment.

Limitations and Future Work

Our model is simplified to capture the dynamic relations between fear reconsolidation and extinction via replays in dream. A lot of other factors in the dreaming process are not considered and should be looked at more comprehensively in order to understand emotion regulations in the brain.

For future work, more experiments can be done with our Pac-Man framework with replays. For example, we may create a replay buffer that samples from ordinary memories instead of only fear memory. In particular, with the Gridworld environment, the consolidation of spatial memory in dreams can be studied. We are also interested in seeing how multiple replay session during training may affect Pac-Man's performances. Lastly, besides using forward replay, we could experiment with backward replay or random replay, where tuples in an event of the replay buffer are played backward starting from the attack state, or random in time, respectively, in reference to reverse replay in hippocampal place cells (Foster & Wilson, 2006).

Acknowledgments

We thank the Berkeley AI Pac-Man project for providing the open sourced framework for the Pac-Man game.

References

Foster, D., & Wilson, M. (2006). Reverse replay of behavioural sequences in hippocampal place cells during the awake state. *Nature*, *440*, 680–683.

Levin, R., & Nielsen, T. A. (2007). Disturbed dreaming, posttraumatic stress disorder, and affect distress: A review and neurocognitive model. *Psychological Bulletin*, *133*, 482–528.

Lin, L. (1992). Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, *8*, 293–321.

Myers, K., & Davis, M. (2007, 03). Mechanisms of fear extinction. *Molecular psychiatry*, *12*, 120-50. doi: 10.1038/sj.mp.4001939

Romain Cazé, L. A., Mehdi Khamassi, & Girard, B. (2018). Hippocampal replays under the scrutiny of reinforcement learning models. *Journal of Neurophysiology*, *6*, 2877–2896.

Russell, S. (2010). Artificial intelligence : a modern approach. *Upper Saddle River, N.J. :Prentice Hall*.

Tibor Bosse, J. d. M., Charlotte Gerritsen, & Treur, J. (2013). Learning emotion regulation strategies: a cognitive agent model..

Van Izquierdo, C. R. G. F., & Myskiw, J. C. (2016). Fear memory. *Journal of Neurophysiology*, *96*, 695–750.