
Revisiting the Fuzzy Tiling Activation and How to Set its Hyperparameters

Muhammad Gohar Javed

Department of Electrical and Computer Engineering
University of Alberta
javed4@ualberta.ca

Tian Xiang Du

Department of Computer Science
University of Alberta
tdu@ualberta.ca

Amir Bahmani

Department of Computer Science
University of Alberta
bahmani1@ualberta.ca

Vlad Tkachuk

Department of Computer Science
University of Alberta
vtkachuk@ualberta.ca

Abstract

Sparsity has been shown to improve model performance on decision making problems with non-stationary data, such as online supervised learning and reinforcement learning (RL). Sparsity is when a large number of weights in a neural network are approximately zero. The fuzzy tiling activation (FTA) has been proposed to enforce sparsity by design, and has been shown to outperform other activations, such as ReLU and Tanh, that do not enforce sparsity. However, a difficulty of using the FTA is that it is sensitive to a new *tiling bound* hyperparameter, which currently requires a search to be set effectively. In this work we do two things. First, we reproduce experiments comparing FTA to ReLU using deep Q-learning (DQN) on the LunarLander RL environment, showing that indeed, for this environment FTA is no better than ReLU if the tiling bound is not set appropriately. Second, we empirically test if a simple technique of normalizing the activation values passed into the FTA cell can remove the need for setting the tiling bound.

1 Introduction

Basically rephrase the abstract in a bit more detail...

Briefly explain how FTA works by partitioning activation into bins and how you have to pick number of bins and tiling bound. Then explain the intuitive issue that arises if the tiling bound is too large (i.e. all the activation values are in the same bin, since none of them are large enough to even enter the second bin). This explanation will help the reader understand why we are investigating the questions listed below and what the logic is behind them.

In this work we aim to do two things:

1. We aim to reproduce the FTA vs ReLU using DQN [Mnih et al., 2013] in LunarLander experiment presented in [Pan et al., 2019].
2. We aim to test if the simple technique of normalizing the input values (using tanh) to FTA can remove the need for tuning the tiling bound parameter in FTA

Experiments related to the first point are shown in section 4.1.

To answer the question in the second point we normalize the values passed into FTA by passing them into a tanh first. This method was discussed in Pan et al. [2019], but not tested. Pan et al. [2019]

mentioned that using this method might suffer from vanishing gradients, which we hope to confirm from this experiment. We hypothesis that normalizing the values should solve the need to set tiling bounds using search, since normalizing will ensure that then values passed into FTA are in $[-1, 1]$, thus the tiling bound can always be set to $[-1, 1]$. Experiments related to the second point are shown in section 4.2.

2 Background

Some background about sparsity and how it is useful in RL. Mention FTA paper here of course and maybe find any new papers that use FTA as well?

3 Preliminaries

In this section we introduce the Fuzzy Tiling Activation (FTA) and formally define the episodic RL setting.

3.1 Fuzzy Tiling Activation

First we introduce the simpler *tiling activation* (TA) and then extend it to the FTA. A TA is a function $\phi : \mathbb{R} \rightarrow \mathbb{R}^k$, which expects inputs $z \in [-u, u]$ and maps them to one-hot vectors (standard basis) in \mathbb{R}^k . For example, if $u = 10$ and $k = 4$, then any $z \in [-10, -5]$ would be mapped to $(1, 0, 0, 0) \in \mathbb{R}^k$, any $z \in [-5, 0]$ would be mapped to $(0, 1, 0, 0)$, any $z \in [0, 5]$ would be mapped to $(0, 0, 1, 0)$, and any $z \in [5, 10]$ would be mapped to $(0, 0, 0, 1)$. Pan et al. [2019] show that the TA can implemented efficiently as follows. Assume you want evenly spaced bins of size $\delta > 0$, and $k = 2u/\delta$, where WLOG u was chosen such that it is divisible by δ . Define the tiling vector

$$\mathbf{c} = (-u, -u + \delta, l + 2\delta, \dots, u - 2\delta, u - \delta) \in \mathbb{R}^k$$

Then the TA can be defined as

$$\phi(z) = \mathbb{1} - I_+(\max(\mathbf{c} - z\mathbb{1}, 0) + \max((z - \delta)\mathbb{1} - \mathbf{c}, 0))$$

where $\mathbb{1} \in \mathbb{R}^k$, and $I_+(\cdot)$ is an indicator function which returns 1 if the input is positive and zero otherwise, and is applied element wise to vectors.

An issue with the TA is that it has zero derivative almost everywhere. In order to solve this issue Pan et al. [2019] proposed modifying the TA for a fuzzy tiling activation (FTA). The FTA can be implemented as follows. Define a fuzzy version of the indicator function as

$$I_{\eta,+} = I_+(\eta - x)x + I_+(x - \eta)$$

where $\eta \geq 0$. Then the FTA can be defined as

$$\phi_\eta(z) = \mathbb{1} - I_{\eta,+}(\max(\mathbf{c} - z\mathbb{1}) + \max((z - \delta)\mathbb{1} - \mathbf{c}, 0)).$$

where $I_{\eta,+}$ is applied element wise to vectors. The reason for the use of fuzzy in the name of FTA can be understood from the definition of $I_{\eta,+}$. The term $I_+(\eta - x)$ is 1 when $x < \eta$, while when $x > \eta$, $I_+(x - \eta)$ is 1. Thus, when $x < \eta$ we have that $I_{\eta,+}$ evaluates to x , giving smoother (fuzzy) indicator function. Notice, that this means for $x < \eta$ the derivative is non-zero, allowing us to use backpropagation to train out networks. Also, when $\eta = 0$ the original indicator function I_+ can be recovered.

3.2 Reinforcement Learning

The reinforcement learning (RL) setting can be formulated as an markov decision process (MDP). Formally an MDP is characterized by the tuple $(\mathcal{S}, \mathcal{A}, \mathbb{P}, R, \gamma)$, where \mathcal{S} is that state space, \mathcal{A} is the action space, \mathbb{P} is the transition probability kernel, $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is the reward function, and $\gamma \in [0, 1]$ is the discount factor. A policy $\pi : \mathcal{S} \rightarrow \mathcal{M}_1$ is a map from states to distributions over actions, where \mathcal{M}_1 is the set of all probability measures over actions \mathcal{A} . Then, an agent (policy) interacts with an MDP as follows. At each time step $t \in 1, 2 \dots$ the agent observes a state $s_t \in \mathcal{S}$. The agent takes an action according to its policy $a_t \sim \pi(s_t)$. The agent then transits to its next state according to $s_{t+1} \sim \mathbb{P}(s, a)$, and receives a reward $R(s_t, a_t, s_{t+1})$.

An action value function under policy π is defined as

$$Q_{\pi}(s, a) = \mathbf{E}[G_t | S_t = s, A_t = a; A_{t+1:\infty} \sim \pi]$$

where $G_t = \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})$ is the return. The goal of the agent is maximize its expected reward from each state in \mathcal{S} .

4 Experiments

Explain the environment we test on. How the LunarLander env works (i.e. state, action space, and reward structure) Explain how DQN works since we use it as our algorithm.

4.1 Reproducibility Experiment

Explain the network structure and how we change only final layer with FTA and ReLU. Then explain what hyper-parameter sweeps we did and add the plot showing them (should discuss patterns we saw here for FTA hyper-params and how they confirm those discussed in [Pan et al., 2019]). Show the plot of FTA vs ReLU using the same hyper-params as in [Pan et al., 2019] and discuss how we see basically same results as them hopefully.

4.2 Normalizing Experiment

Explain how we plan to normalize the activation value going into FTA (i.e. $z = \tanh(Xw)$). I think we can leave this as TODO for final report.

5 Discussion

Discuss how the reproducibility experiment basically matches the results in [Pan et al., 2019] paper Discuss what we find from normalizing activations (i.e. if it worked or not, and if not any hypothesis why might have as to why)

References

- V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.
- Y. Pan, K. Banman, and M. White. Fuzzy tiling activations: A simple approach to learning sparse representations online. *arXiv preprint arXiv:1911.08068*, 2019.