How Do You Act? An Empirical Study to Understand Behavior of Deep Reinforcement Learning Agents

Richard Meyes, Moritz Schneider, and Tobias Meisen

Chair of Technologies and Management of the Digital Transformation,
University of Wuppertal, Rainer-Gruenter-Strae 21, 42119 Wuppertal, Germany
{meyes, m.schneider-hk, meisen}@uni-wuppertal.de

Abstract. The demand for more transparency of decision-making processes of deep reinforcement learning agents is greater than ever, due to their increased use in safety critical and ethically challenging domains such as autonomous driving. In this empirical study, we address this lack of transparency following an idea that is inspired by research in the field of neuroscience. We characterize the learned representations of an agent's policy network through its activation space and perform partial network ablations to compare the representations of the healthy and the intentionally damaged networks. We show that the healthy agent's behavior is characterized by a distinct correlation pattern between the network's layer activation and the performed actions during an episode and that network ablations, which cause a strong change of this pattern, lead to the agent failing its trained control task. Furthermore, the learned representation of the healthy agent is characterized by a distinct pattern in its activation space reflecting its different behavioral stages during an episode, which again, when distorted by network ablations, leads to the agent failing its trained control task. Concludingly, we argue in favor of a new perspective on artificial neural networks as objects of empirical investigations, just as biological neural systems in neuroscientific studies, paving the way towards a new standard of scientific falsifiability with respect to research on transparency and interpretability of artificial neural networks.

Keywords: Transparency, Interpretability, Explainability, Deep Reinforcement Learning, Neuroscience

1 Introduction

Recent research on general-purpose artificial intelligence (AI) has seen some major breakthroughs in the past few years spurred by the advances of deep reinforcement learning (DRL) algorithms utilized in environments with sparse rewards and complete information [1,2] or in complex multi-agent environments with incomplete information [3,4,5]. However, the research path leading up to today's pinnacle of these applications is marked by a crisis of reproducibility

and required intense manual trial-and-error efforts such as finding a good network initialization and subsequent hyper-parameter tuning, which can make all the difference between a working and a failing solution [6]. What complicates the problem even more is that many working solutions are interspersed with unwanted behavioral artifacts that manifest in the learned policy of agents, if the environment allows for such manifestation, e.g. in the domain of learning locomotion [7]. Such artifacts are commonly caused by incentivizing an agent to solely maximize a possibly richly shaped reward without any constraints on its policy. The usual approach of training agents to maximize their cumulative reward and quantitatively evaluating them solely based on this reward or any other performance metric, such as the ELO rating in chess, raises a key question: **How can we trust an agent, if we do not understand how its behavior emerges from its internal processes and the complex interplay of its individual functional components?**

In this paper, we aim to contribute towards answering this question following a research paradigm from the field of neuroscience based on empirical studies of large and complex neural systems. Such systems have been the objects of investigation for decades starting with the influential work of Hubel and Wiesel in the 1950s [8], aiming to make them transparent and interpretable with respect to how their inner processes contribute to abstract concepts like consciousness and decision-making. Specifically, we investigate the behavior of DRL agents in three different classic control environments based on the learned representations of their policy networks, aiming to find a link between these representations and different behavioral stages during the execution of the trained policy. We characterize the actor's learned representations based on its layer activation during the execution of the policy and use network ablations (cf. section 3.2) to intentionally damage agents, evoking malfunctioning behavior to compare the representations of the fully intact and damaged networks to each other.

First, we investigate the impact of network ablations with different sizes in different layers on the agent's capability to solve its trained control task and show that the the agent exhibits a task specific robustness to these ablations depending on the size and location of the ablations. We further investigate how the activations of single units contribute to solving the control task, uncovering specific correlation patterns between these activations and the executed actions during an episode. Finally, we investigate patterns in the temporal evolvement of the actor's layer activation and find that the healthy agent's learned representation contains distinct activation states that can be directly linked to the different behavioral stages of the policy that successfully solves the control task, ultimately providing a link between the agent's behavior and its internal processes.

2 Related Work

Most of the recent work on transparency, interpretability and explainability of AI comes from the field of computer vision (CV), where the main focus is com-

monly placed on investigations of convolutional neural networks (CNNs) and the importance of specific input variables for a network's output [9,10,11,12,13]. Similar efforts are made in the field of natural language processing (NLP), where recurrent neural networks (RNNs) are investigated for their representations of linguistic properties, contextual understanding or sentiment [14,15,16,17]. Typically, learned network representations are characterized via embedding methods like t-SNE [18] or UMAP [19] visualizing the high dimensional activation-space of neural networks to identify the role of specific network components in solving a given task [20,21,22,23,24]. To this end, network ablations were used to study the impact of single units on a network's performance [25], aiming to decide which units can be pruned without affecting a network's discriminative power [26,27,28]. Subsequently, network ablations revealed that a single unit's importance can be characterized by the magnitude of its weights [29] and the extent to which the distribution of its incoming weights changes during training [30,31]. Additionally, it was shown that units, which are easily interpretable, are not necessarily more important than units with a less accessible interpretability [32]. Recently, controversial insights on methods how to evaluate the similarity of learned network representations have been reported and demonstrate the early stage of current knowledge and thus, the importance and the need for more research on the topic [33,34]. In general, the extensive efforts of recent research aimed to map the classification result of a supervised trained network to humanly interpretable explanations. We aim to extend these efforts towards the DRL domain, where despite some work on understanding Deep Q-Networks and interpreting their learned policies in environments with a discrete action space [35], to the best of our knowledge, work on facilitating transparency and interpretability of learned representations by means of network ablations has not been conducted yet. However, in view of the fact that robust DRL and its application in real world scenarios is still a matter of current research [6,36], we argue that a better interpretability of DRL agents is of utmost importance.

3 Study Design

3.1 Experimental Setup

In this empirical study, we trained a DRL agent in three different classic control environments, namely the cart-pole swing-up (CPSU) environment [37], the pendulum swing-up (PSU) environment [38] and the cart-pole balance (CPB) environment [37] (cf. Figure 1). Although each environment poses an individual challenge, they share the partial objectives of controlling a cart on a rail or balancing a pendulum/pole in an upright position, providing some degree of comparability of the observed agent's behavior across tasks. We refrain from a more detailed explanation of the intricacies of these environments regarding their state space, action space and reward functions at this point, as they are well-known benchmark environments for DRL research and have been extensively explained elsewhere [37,38].

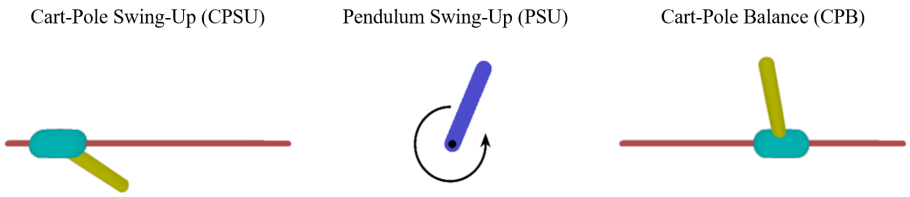


Fig. 1: Three exemplary rendered images of the respective control environments.

As the object of investigation, we trained an actor-critic agent in the three described environments with the deep deterministic policy gradient algorithm as outlined in [39]. Both, the actor and the critic network consist of two hidden layers with 400 units in the first layer and 300 units in the second layer with both layers using ReLU activation and layer normalization [40]. The critic is supplied with the actor's chosen actions, which is superimposed by an OrnsteinUhlenbeck noise process [41], only in the second hidden layer. Each agent was trained for 800,000 time steps and optimized via Adam [42] with all other hyper-parameters being the same as in [39]. All computations were performed on a single machine containing two Intel Xeon Platinum 8168 processors with a total number of 48 physical cores and 8 NVIDIA Tesla V100 32G GPUs.

3.2 Characterization of Learned Representations

We characterize the actor's learned representations based on its layer activation during policy execution. We use network ablations to intentionally damage the actor, evoking malfunctioning agent behavior to compare the representations of the fully intact and damaged networks. To this end, we record the activation of each single unit within the fully intact actor and its predicted actions for each time step of an episode in addition to the cumulative episodic reward to establish a baseline recording. Additionally, we record the same data for each individual ablation case to compare it to the baseline recording.

Network Ablations. We perform partial network ablations in a single layer with varying proportions of ablated units by manually clamping their activations to zero, effectively preventing any flow of information through the ablated units. We select the amount of ablated units in a range from 5% to 90% in steps of 5% until 30% and then in steps of 10% until 90%. In addition, we deviate from this pattern once by ablating 33.33% of units within a layer. Thereby, the ablated units are selected in a sliding window manner that is shifted across the layer, similar to sliding a kernel over an image in a CNN while the window position is frozen during an episode. Note that the total number of ablations with the same proportion varies because they depend on the size of the layer, the size of the window and the stride of the window. For instance, in a layer with 300 units and a chosen window size of 5% with a stride of 10 units, 15 units are ablated

at once resulting in 29 different network ablations in total. For all ablations, we chose a constant stride value of 10 units to gather sufficient activation recordings for statistical analysis while at the same time keeping the computational efforts manageable.

Extraction of Activation Patterns. To determine how single units contribute to the control task, we calculate the Pearson correlation coefficient of its set of activations $A_{i,j} = \{a_t | t \in [0, T]\}$ and the outputs of the actor network $U = \{u_t | t \in [0, T]\}$, for each time step within an episode, where t denotes the time step within the episode, T denotes the total number of time steps per episode, i denotes the i-th layer and j the j-th unit within that layer.

Furthermore, to characterize the learned representations within a layer of the actor, we store the activations of each single unit in that specific layer for each time step of an episode in a matrix $M^{T \times N}$, where T denotes the number of time steps per episode and N denotes the number of units per layer. We visualize the evolvement of the actor's activation during an episode using an open source Python implementation of UMAP [19] to embed the stored activations into a two-dimensional space, i.e. $M \in \mathbb{R}^{T \times 2}$. Thus, each point in the embedded space represents the activation of a specific layer of the actor network for a single time step of an episode. We chose the default parameters for the UMAP embeddings after an initial attempt for finding better values for the number of nearest neighbours or the minimum distance between data points yielded no significant visual improvement of the embeddings.

4 Results

4.1 Impact of Ablations on the Agent's Capability

To establish a baseline evaluation, we train the healthy agent to achieve near state-of-the-art results in all three environments, i.e. a maximum total episodic reward of 886.4 for the CPSU task, -275.87 for the PSU task and 1000 for the CPB task. For reasons of performance comparability across the three environments, the absolute return is normalized so that the minimum return value in each environment is 0 and the respective baseline return value is 1.

Figure 2 shows the normalized return for the baseline in comparison to all 29 network ablations in the first and second layer with a window size of 30% (120 units) for the three control tasks. For both swing-up tasks, most ablations in the first layer have a negative impact on the agent's capability to solve the tasks. Interestingly, there are some ablations that have little to no impact or even a positive impact, thus increasing the return. In case of the CPB task, ablating 30% of the units in the first layer does not affect the agent's capability to solve the task at all. Contrary to the first layer, all ablations in the second layer have a strong negative impact for the CPSU task and the CPB task (except for two cases), however, only a few ablations have a comparably negative impact for the PSU task, where many ablations have little to no impact or even a positive impact. The negligible impact of ablations suggests that either the capacity of

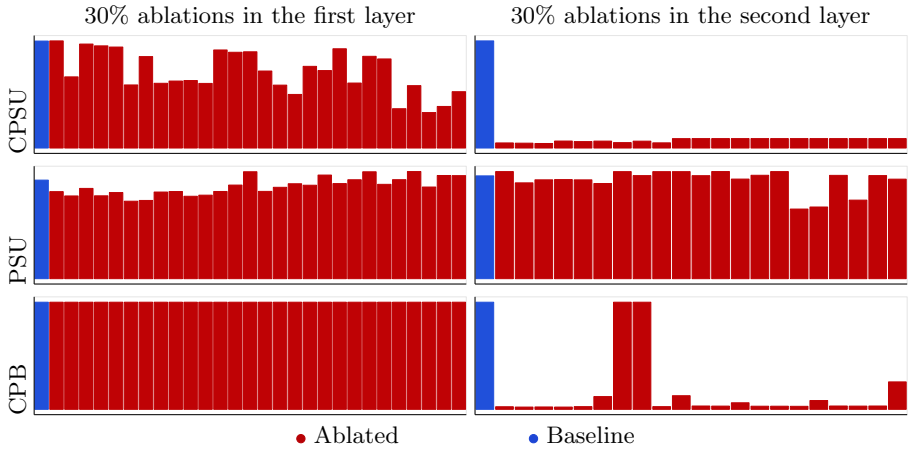


Fig. 2: Comparison of the normalized returns achieved as a result of ablations of 30% of the units (red bars) in to its respective baselines (blue bars).

the network has not been exploited to its fullest extent so that some units do not contribute to solving the task and could be pruned or that the information represented by the ablated units is redundantly represented by other units making the agent robust against network ablations. The positive impact of ablations suggests that some units may play competing roles in the learned representation and that resolving this competition by targeted ablations improves the agent's capability to solve a task. Both observations are consistent with previously reported findings on the impact of ablations in supervised trained neural networks on image recognition tasks [30,31].

Figure 3 shows the distributions of the normalized returns resulting from the different network ablations in the first layer and second layer for the three control tasks.

On average, the return decreases proportionally to the amount of ablated units. Comparing the impacts in the first layer across the three tasks shows a similar trend for the CPSU and the PSU task, i.e. a slow but steady decrease of the achieved return with increasing sizes of ablations but a much more robust behavior for the CPB task, where ablations of up to 50% generally do not affect the agent's capability to solve the task. Further, comparing the impacts in the second layer shows a similar trend for the CPSU and the CPB task, i.e. a strong and sudden decrease in the achieved return for small ablation sizes, but a much more robust behavior for the PSU task, where ablations of up to 33.33% only marginally affect the agent's capability to solve the task. Interestingly, connecting the similarity of the ablation impacts with the similarity of the different tasks suggests that the first layer holds a representation of how to swing up the pole/pendulum while the second layer holds a representation of how to control the moving cart. More precisely, ablations in the first layer impact the agent in both tasks, in which a pole has to be swung up, while the representation for

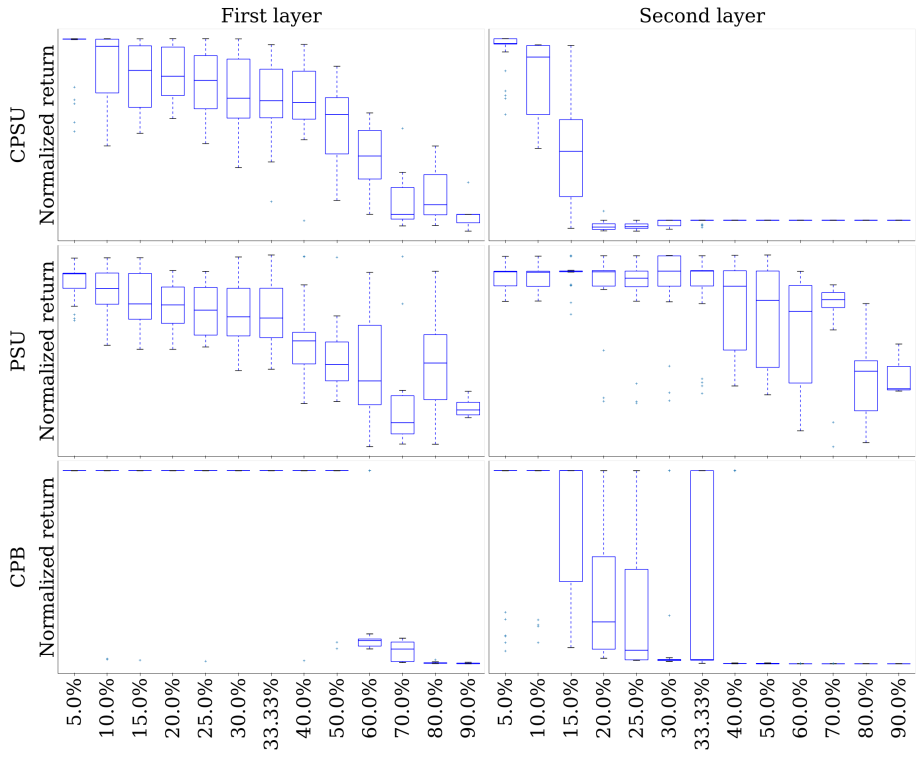


Fig. 3: Distributions of the normalized returns for all ablations performed in the first layer (left side) and second layer (right side).

the task, which merely requires balancing the pole, is very robust against ablations in this layer. Analogously, ablations in the second layer strongly impact the agent in both tasks, in which a cart has to be controlled, while the representation for the task without a cart is fairly robust against ablations in this layer. These results suggest that interlinked learning objectives to solve the task such as controlling the cart, swinging up the pendulum and subsequently balancing it, are represented in different locations of the network. These observations are consistent with previously reported findings on the localized representations of specific classes in supervised trained neural networks on image classification tasks [43,44,45].

4.2 Impact of Ablations on Single Unit Activity (SUA)

Following the observations described above, we wonder what role the precise interplay of SUA plays with respect to the agent's executed policy. More specifically, we ask whether the contribution of SUA to the executed actions during an episode shows a distinct pattern for the healthy agent and to what extent this

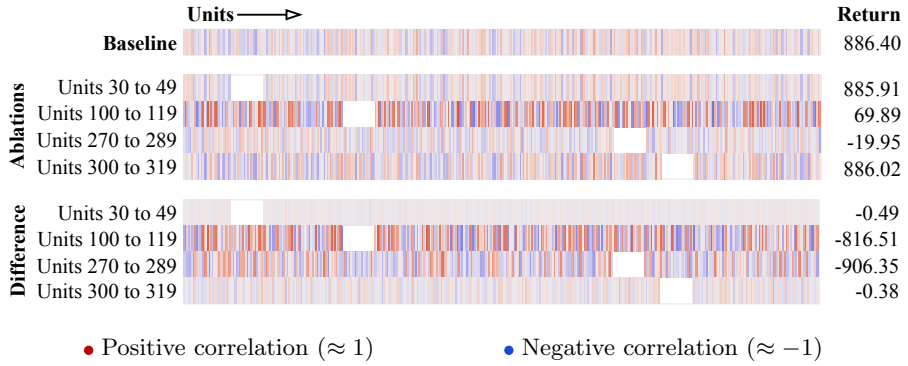


Fig. 4: Correlation pattern of the activations of all 400 units in the first layer during the CPSU task for the healthy agent (baseline) and four exemplary ablations, as well as the change of these patterns compared to the baseline (bottom four rows).

pattern is distorted in case of ablations with a negative impact on the achieved return. To this end, we characterize this pattern via the set of Pearson correlation coefficients calculated for the activations of single units within a layer and the outputs of the actor network for each time step within an episode (cf. 3.2).

Figure 4 shows this pattern for the baseline and four exemplary ablations of 5% of units in the first layer activated in the CPSU task. Each row contains 400 entries corresponding to the 400 units in the first layer. Each entry contains the correlation value and shows how the unit's activation correlates with the actor's chosen action. The empty spaces in the rows show the ablated units, for which no correlation coefficient is calculated. The top row shows the baseline correlation pattern in comparison to the following four rows, which show the correlation patterns corresponding to the four exemplary ablations. The bottom four rows show to what extent the patterns resulting from the ablations change compared to the baseline pattern, specified by the difference between the baseline pattern and the ablation patterns. The ablations of units 100 to 119 and 270 to 289, resulting in the agent's failure to solve the task, show a general increase in correlation between the SUA and the chosen actions and the strongest difference of the pattern compared to the baseline. A high correlation value indicates a unit's exclusive contribution to a specific control direction, i.e. whenever the cart is moved to either side, specific units are selectively active and contribute to the control in a specific direction. However, such distinct contributions of single units do not seem to resemble a robust representation as we find that patterns with less distinct correlations between single unit activations and the chosen actions generally lead to higher returns. This observation shows some similarity with previously reported findings about the importance of single units in supervised trained networks for image classification tasks. Specifically, networks that memorize well instead of generalizing are more reliant on units that show a high selectivity in their activation for specific classes, indicating that units which se-

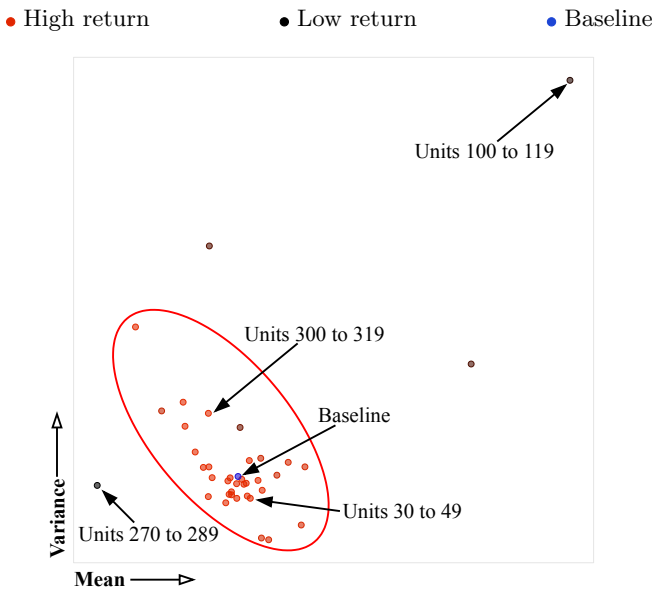
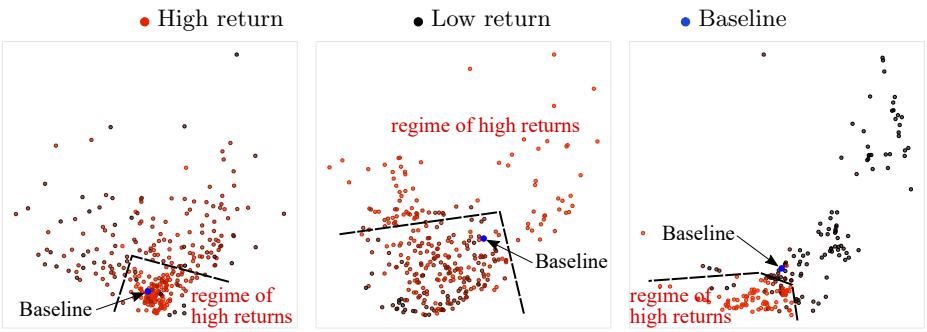


Fig. 5: Scatter plot of the mean and the variance of the correlation patterns for the baseline and all 29 ablations of the size of 5% and their corresponding returns in the CPSU task.

lectively get activated for specific classes do not contribute as much to a robust and generalized representation as units with a less selective activation [32].

In order to further solidify that notion, we compared the mean and the variance of the correlation patterns of all ablations with the mean and the variance of the baseline pattern, hypothesizing that high values for the mean and the variance, corresponding to strong and distinct correlations, result in a low return. Figure 5 shows a scatter plot of the mean and the variance of the correlation patterns for the baseline and all 29 ablations of the size of 5% and their corresponding returns. Confirming the hypothesis, ablations of units resulting in large values for the mean and the variance, e.g. units 100 to 119 (marked in the top right corner of the scatter plot) lead to low returns. Almost all other ablations with mean and variance values close to the baseline (points within the red ellipsis) do not result in task failures but achieve returns comparable to the baseline. Interestingly, the ablation of the units 270 to 289, which results in small values for the mean and the variance, also leads to a low return, suggesting that our hypothesis can be extended towards small values for the mean and the variance, corresponding to no clear contribution for most of the single units to the control task.

To further test the validity of the hypothesis across different sizes of ablations and across the three tasks, Figure 6 summarizes the effects of all ablations (5% to 90%) on the return and the dependency on the characteristics of the



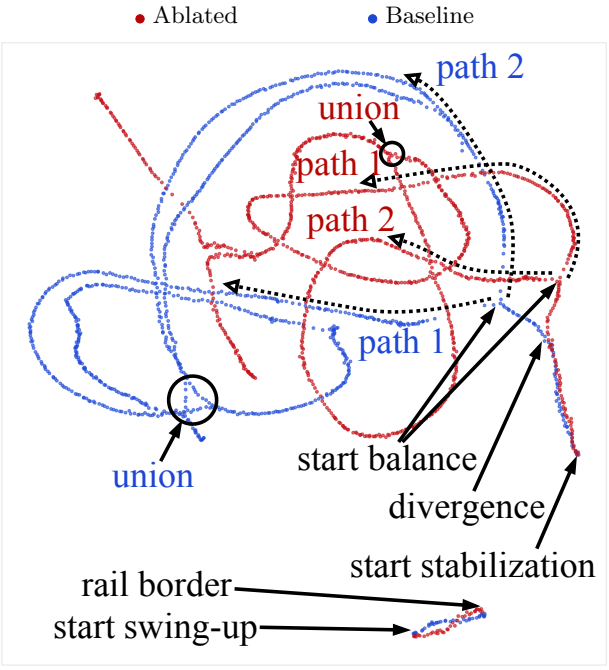
(a) First layer, CPSU (b) First layer, PSU (c) Second layer, CPB

Fig. 6: Scatterplot showing the mean (x-axis) and variance (y-axis) of the correlation coefficients for all ablations of the specified layer.

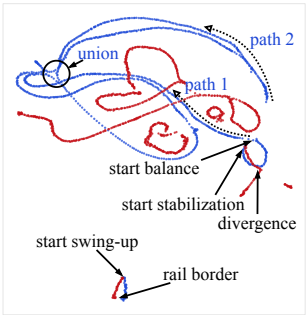
correlation patterns. Analogously to figure 5, the x- and y-axis show the mean and the variance of the correlation patterns. For the CPSU task, the highest return is generally achieved for patterns with a low variance as ablations leading to larger variances show a decreased return. This suggests that the CPSU task requires single units to be generically involved in the control task and not to specialize too strongly on specific controls. On the contrary for the PSU task, higher returns are generally achieved for patterns with a high mean and high variance, suggesting a further refinement of our hypothesis with respect to task specific characteristics. Interestingly, ablations that increase both values beyond the baseline lead to even higher returns while patterns with low values lead to low returns. This suggests that the ability to swing-up the pendulum requires the units to contribute to the control in a very specific rather than generic way. Consistently, a very clear picture emerges for the CPB task, where no swing-up is required and only patterns with low values for mean and variance result in high returns, verifying our initial hypothesis. In combination with the CPSU task, this suggests that the ability to control the moving cart requires a generic involvement of single units in the control task rather than specific roles.

4.3 Impact of Ablations on Layer Activation

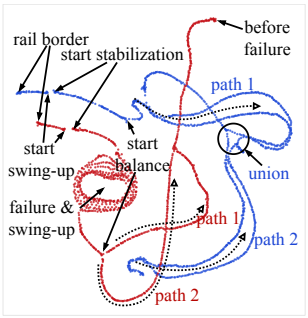
Although the correlation patterns provide some insights on how the agent acts, they do not capture the temporal evolvement of the learned representations and do not answer questions with respect to such evolvements, e.g. at what point during the episode does the agent fail? When does it diverge from the baseline behavior and in what way? Does the agent go through different behavioral stages during an episode and can these stages be linked to specific patterns in the the learned representation? In order to answer these questions, we characterize the learned representations by embedding the layer activations recorded during an episode (cf. 3.2) and compare the representations of the baseline to the representations resulting from the ablations.



(a) Ablation of units 20 to 39 (5%) in the first layer.



(b) Units 110-149 (10%), layer 1.



(c) Units 260-289 (10%), layer 2.

Fig. 7: Comparison of the temporal evolvement of layer activations between the baseline and three exemplary ablation cases for the CPSU task.

Figure 7 shows this comparison for three exemplary ablation cases for the CPSU task. Each scatter plot contains 1000 blue and 1000 red points corresponding to the layer activation for each time step during an episode for the baseline and the ablation case, respectively. Note that even though the baselines in (a) and (b) show the exact same values, they are embedded slightly differently as the embeddings were calculated separately for all cases. The three cases correspond to ablations, which had no effect on the agent's capability to solve the task (Figure 7a) or which lead to only half the return of the baseline (Figure 7b and c). Figure 7a shows the evolvement of the layer activation during an episode for the healthy and the damaged agent and how the different behavioral stages of the episode are linked to different sections of this evolvement. Both, the healthy and the damaged agent, start with moving the cart to the side, accelerating the pendulum to swing it up. After the initial swing-up (upon reaching the rail border), the agent is required to compensate for the excess momentum of the pole via corresponding cart movement to stabilize its upright position. This change in behavior results in a jump in the activation space from the initial activation

path that corresponds to the initial swing-up behavior to another path that corresponds to the stabilization behavior. The difference in activations is likely due to the movement of the cart into the opposite direction upon reaching the rail border. Following the successful stabilization, the agent is required to balance the pole by rapidly switching directions of the cart to maintain an upright pole position. Interestingly, this behavior is represented in the activation space by two paths, along which the layer activation progresses as the agent acts throughout the episode. The layer activation repeatedly switches between these two paths suggesting that the network constantly changes between two distinct activation states corresponding to the balancing act of the pole. At some point during the episode, these two paths merge together (union) as the balancing act leads to an almost static position of the cart and the pole. However, from a mechanical perspective, this constitutes an unstable equilibrium point for the pole, where small perturbations of the pole's angular position result in its downfall triggering a renewed balancing act that is resembled by a renewed separation of the merged paths. This observations suggests that the convergence of the actor's activation towards a single final activation state is not sufficient to solve the task. Rather, a stable and continuous transition between two distinct activation states is necessary to sufficiently represent the balancing act. This observation seems somewhat surprising considering the weak correlations of SUA to the actor's chosen actions throughout an episode (cf. 4.2). Although the SUA does not correlate strongly with the network's executed actions, their combined activations lead to two distinct activation states of the network, each of them corresponding to the movement of the cart in either one of the two possible directions during the balancing act. This suggests that single units do not contribute individually to the control task, but rather as part of a larger conglomerate of units that constitute the two different activation states.

Figure 7b shows an ablation case, for which the agent fails to balance the pole continuously after the initial swing-up and drops it after a short period of holding it in the upright position, reattempting the swing-up and balancing act. The layer activation diverges slightly from the baseline right from the start of the swing-up and further diverges completely after a short period of the stabilization phase. Consequently, due to this divergence, the layer activation of the damaged agent does not show the emergence of two distinct paths connected to the balancing act as the agent never succeeds in stabilizing the pole compensating its excess momentum after the initial swing-up. Interestingly, the existence of two distinct activation states is not exclusive to the actor's first layer but also apparent in its second layer. Figure 7c shows an ablation case in layer two, in which the failure of the agent is caused by a drop of the pole after the initial swing-up and a short period of balancing, causing the pole to rotate at high speed until the end of the episode. The blue points resemble a similar pattern of the second layer's activation compared to the first layer including the divergence of the activation along two distinct paths, the attempt to merge these paths and the renewed separation. The failure of the agent, i.e. the continuous rotation of the pole at high speed, is visible in the activation space by the circularly ar-

ranged red points, from which the agent is not able to recover back onto the stabilization path and the both connected paths corresponding to the balancing act.

5 Conclusions & Future Work

In this paper, we conducted an empirical study to understand how a DRL agent acts based on characterizing the learned representations of its policy network. We shed some light on the role of single units for the control task and found that despite the absence of a strong correlation between their activations and the actor's chosen actions throughout an episode, agents, that solve their tasks successfully, show task specific patterns of weakly correlated SUA that get distorted by network ablations leading to low returns. The importance of these patterns for a successful solution of the control task suggests that the careful interplay between single units with respect to the executed policy is essential rather than their sole and isolated behavior. However, we have only scratched the surface of how such patterns of joint activations can be characterized. In our future work, we plan to systematically investigate the role of functional neuron populations and their involvement in solving a given control task. Specifically, we plan to investigate the activation of sub-populations of neurons aiming to uncover if there is a link between their activations and the emergent agent behavior.

We further investigated the temporal evolvement of the actor's layer activations during an episode and showed that, in case of the CPSU task, the consecutive steps executed during the episode to solve the task are precisely represented by the policy network and mapped onto its layer activations. We fruther showed that this mapping is essential for solving the task as its distortion as a result of network ablations leads to low returns and failed attempts to solve the task. The arrangement of the consecutive points in the embedded activation space revealed that the agent runs along specific paths in its activation space and that diverging from this path is fatal for its task performance. The most striking observation of these paths is given by the fact that the actor's layer activations can be very different for very similar states. We naively expected that the layer activation would converge to a single specific activation vector just as the consecutive states to be processed by the network become more and more similar to each other as the pole is balanced. However, we found that this is not the case, suggesting that the learned representations may contain some information that is encoded in the temporal dimension on which the states are ordered, i.e. that the same state evokes a different activation of the network depending on when it is presented to the network. In our future work, we plan to investigate how these distinct activation patterns evolve during training, aiming to answer the question, whether the different behaviors are learned hierarchically, i.e. in a specific order, or whether they emerge collectively.

Considering that our study was limited to a single agent solving three distinct control tasks, the universality of our results is strongly limited and their implications for other networks and tasks is not clear. We plan to address this

issue by transferring our study design to a larger number of different networks and control tasks aiming to establish a scientific standard for the falsifiability of empirical studies conducted in the field of artificial neural networks. Ultimately, we aim to pave the way towards a new perspective of neuroscience inspired empirical studies on artificial neural networks to exploit them as a test bed for neuroscientific research. Uncovering parallels between the structure and organization of represented knowledge in artificial and biological systems opens up measures and possibilities for initial large scale studies in artificial systems before transferring them to biological systems. Specifically, this addresses the issue of reproducibility, which, despite modern experimental methods, is one of the most critical issues in modern neuroscience, stemming from the large differences between brains and the commonly small sample sizes in neuroscientific studies.

References

1. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.
2. D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, *et al.*, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," *arXiv preprint arXiv:1712.01815*, 2017.
3. OpenAI, "Openai five," *https://blog.openai.com/openai-five/*, 2018.
4. B. Baker, I. Kanitscheider, T. Markov, Y. Wu, G. Powell, B. McGrew, and I. Mordatch, "Emergent tool use from multi-agent autocurricula," *arXiv preprint arXiv:1909.07528*, 2019.
5. M. Jaderberg, W. M. Czarnecki, I. Dunning, L. Marris, G. Lever, A. G. Castaneda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, *et al.*, "Human-level performance in 3d multiplayer games with population-based reinforcement learning," *Science*, vol. 364, no. 6443, pp. 859–865, 2019.
6. A. Irpan, "Deep reinforcement learning doesn't work yet." `https://www.alexirpan.com/2018/02/14/rl-hard.html`, 2018.
7. I. Popov, N. Heess, T. Lillicrap, R. Hafner, G. Barth-Maron, M. Vecerik, T. Lampe, Y. Tassa, T. Erez, and M. Riedmiller, "Data-efficient deep reinforcement learning for dexterous manipulation," *arXiv preprint arXiv:1704.03073*, 2017.
8. D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurones in the cat's striate cortex," *The Journal of physiology*, vol. 148, no. 3, pp. 574–591, 1959.
9. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 506–519, ACM, 2017.
10. R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3429–3437, 2017.
11. K. Faust, Q. Xie, D. Han, K. Goyle, Z. Volynskaya, U. Djuric, and P. Diamandis, "Visualizing histopathologic deep learning classification and anomaly detection using nonlinear feature space dimensionality reduction," *BMC bioinformatics*, vol. 19, no. 1, p. 173, 2018.

12. J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Transactions on Evolutionary Computation*, 2019.
13. R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," 2019.
14. A. Karpathy, J. Johnson, and L. Fei-Fei, "Visualizing and understanding recurrent networks," *arXiv preprint arXiv:1506.02078*, 2015.
15. A. Radford, R. Jozefowicz, and I. Sutskever, "Learning to generate reviews and discovering sentiment," *arXiv preprint arXiv:1704.01444*, 2017.
16. A. Bau, Y. Belinkov, H. Sajjad, N. Durrani, F. Dalvi, and J. Glass, "Identifying and controlling important neurons in neural machine translation," *arXiv preprint arXiv:1811.01157*, 2018.
17. A. Madsen, "Visualizing memorization in rnns," *Distill*, 2019. https://distill.pub/2019/memorization-in-rnns.
18. L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
19. L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.
20. M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, and S. Liu, "Towards better analysis of deep convolutional neural networks," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 91–100, 2016.
21. P. E. Rauber, S. G. Fadel, A. X. Falcao, and A. C. Telea, "Visualizing the hidden activity of artificial neural networks," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 101–110, 2016.
22. Z. Elloumi, L. Besacier, O. Galibert, and B. Lecouteux, "Analyzing learned representations of a deep asr performance prediction model," *arXiv preprint arXiv:1808.08573*, 2018.
23. D. V., "Convnet playground," *https://convnetplayground.fastforwardlabs.com*, 2019.
24. S. Carter, Z. Armstrong, L. Schubert, I. Johnson, and C. Olah, "Activation atlas," *Distill*, vol. 4, no. 3, p. e15, 2019.
25. F. Dalvi, A. Nortonsmith, A. Bau, Y. Belinkov, H. Sajjad, N. Durrani, and J. Glass, "Neurox: A toolkit for analyzing individual neurons in neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 9851–9852, 2019.
26. P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," *arXiv preprint arXiv:1611.06440*, 2016.
27. H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient convnets," *arXiv preprint arXiv:1608.08710*, 2016.
28. N. Cheney, M. Schrimpf, and G. Kreiman, "On the robustness of convolutional neural networks to internal architecture and weight perturbations," *arXiv preprint arXiv:1703.08245*, 2017.
29. F. Dalvi, N. Durrani, H. Sajjad, Y. Belinkov, D. A. Bau, and J. Glass, "What is one grain of sand in the desert? analyzing individual neurons in deep nlp models," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2019.
30. R. Meyes, M. Lu, C. W. de Puiseau, and T. Meisen, "Ablation studies in artificial neural networks," 2019.
31. R. Meyes, M. Lu, C. W. de Puiseau, and T. Meisen, "Ablation studies to uncover structure of learned representations in artificial neural networks," *Int'l Conf. Artificial Intelligence 2019*, 2019.

32. A. S. Morcos, D. G. Barrett, N. C. Rabinowitz, and M. Botvinick, "On the importance of single directions for generalization," *arXiv preprint arXiv:1803.06959*, 2018.
33. A. Morcos, M. Raghu, and S. Bengio, "Insights on representational similarity in neural networks with canonical correlation," in *Advances in Neural Information Processing Systems*, pp. 5727–5736, 2018.
34. S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," *arXiv preprint arXiv:1905.00414*, 2019.
35. T. Zahavy, N. Ben-Zrihem, and S. Mannor, "Graying the black box: Understanding dqns," in *International Conference on Machine Learning*, pp. 1899–1908, 2016.
36. G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," *arXiv preprint arXiv:1904.12901*, 2019.
37. OpenAI, "Openai roboschool," 2017.
38. G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016.
39. T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.
40. J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.
41. G. E. Uhlenbeck and L. S. Ornstein, "On the theory of the brownian motion," *Physical review*, vol. 36, no. 5, p. 823, 1930.
42. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
43. A. Veit, M. J. Wilber, and S. Belongie, "Residual networks behave like ensembles of relatively shallow networks," in *Advances in neural information processing systems*, pp. 550–558, 2016.
44. C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, and A. Mordvintsev, "The building blocks of interpretability," *Distill*, 2018. https://distill.pub/2018/building-blocks.
45. I. Rafegas, M. Vanrell, L. A. Alexandre, and G. Arias, "Understanding trained cnns by indexing neuron selectivity," *Pattern Recognition Letters*, 2019.