ON BONUS-BASED EXPLORATION METHODS IN THE ARCADE LEARNING ENVIRONMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Research on exploration in reinforcement learning, as applied to Atari 2600 game-playing, has emphasized tackling difficult exploration problems such as MONTEZUMA'S REVENGE (Bellemare et al., 2016). Recently, bonus-based exploration methods, which explore by augmenting the environment reward, have reached above-human average performance on such domains. In this paper we reassess popular bonus-based exploration methods within a common evaluation framework. We combine Rainbow (Hessel et al., 2018) with different exploration bonuses and evaluate its performance on MONTEZUMA'S REVENGE, Bellemare et al.'s set of hard of exploration games with sparse rewards, and the whole Atari 2600 suite. We find that while exploration bonuses lead to higher score on MONTEZUMA'S REVENGE they do not provide meaningful gains over the simpler ϵ-greedy scheme. In fact, we find that methods that perform best on that game often underperform ϵ-greedy on easy exploration Atari 2600 games. We find that our conclusions remain valid even when hyperparameters are tuned for these easy-exploration games. Finally, we find that none of the methods surveyed benefit from additional training samples (1 billion frames, versus Rainbow's 200 million) on Bellemare et al.'s hard exploration games. Our results suggest that recent gains in MONTEZUMA'S REVENGE may be better attributed to architecture change, rather than better exploration schemes; and that the real pace of progress in exploration research for Atari 2600 games may have been obfuscated by good results on a single domain.

1 INTRODUCTION

In reinforcement learning, the exploration-exploitation trade-off describes an agent's need to balance maximizing its cumulative rewards and improving its knowledge of the environment. While many practitioners still rely on simple exploration strategies such as the ϵ-greedy scheme, in recent years a rich body of work has emerged for efficient exploration in deep reinforcement learning. One of the most successful approaches to exploration in deep reinforcement learning is to provide an exploration bonus based on the relative novelty of the state. This bonus may be computed, for example, from approximate counts (Bellemare et al., 2016; Ostrovski et al., 2017; Tang et al., 2017; Machado et al., 2018a), from the prediction error of a dynamics model (ICM, Pathak et al., 2017) or by measuring the discrepancy to a random network (RND, Burda et al., 2019).

Bellemare et al. (2016) argued for the importance of the Atari 2600 game MONTEZUMA'S REVENGE as a challenging domain for exploration. MONTEZUMA'S REVENGE offers a sparse reinforcement signal and relatively open domain that distinguish it from other Atari 2600 games supported by the Arcade Learning Environment (ALE; Bellemare et al., 2013). As a consequence, recent exploration research (bonus-based or not) has aspired to improve performance on this particular game, ranging from a much deeper exploration (6600 points; Bellemare et al., 2016), to completing the first level (14000 points; Pohlen et al., 2018; Burda et al., 2019), to super-human performance (400,000 points; Ecoffet et al., 2019).

Yet the literature on exploration in reinforcement learning still lacks a systematic comparison between existing methods, despite recent entreaties for better practices to yield reproducible research (Henderson et al., 2018; Machado et al., 2018b). One of the original tenets of the ALE is that agent evaluation should take on the entire suite of 60 available Atari 2600 games. Even within the context of bonus-based exploration, MONTEZUMA'S REVENGE is but one of seven hard exploration, sparse

rewards games (Bellemare et al., 2016). Comparisons have been made between agents trained under different regimes: using different learning algorithms and varying numbers of training frames, with or without reset, and with and without controller noise (e.g. *sticky actions* Machado et al., 2018b). As a result, it is often unclear if the claimed performance improvements are due to the exploration method or other architectural choices, and whether these improvements carry over to other domains. The main conclusion of our research is that an over-emphasis on one Atari 2600 game, combined with different training conditions, may have obfuscated the real pace of progress in exploration research.

To come to this conclusion, we revisit popular bonus-based exploration methods (pseudo-counts, PixelCNN-counts, RND, and ICM) in the context of a common evaluation framework. We apply the Rainbow (Hessel et al., 2018) agent in turn to MONTEZUMA'S REVENGE, Bellemare et al.'s seven hardest Atari 2600 games for exploration, and the full suite from the ALE.

Source of exploration bonus. We find that all agents perform better than an ϵ-greedy baseline, confirming that exploration bonuses do provide meaningful gains. However, we also find that more recent exploration bonuses do not, by themselves, lead to higher performance than the older pseudo-counts method. Across the remainder of hard exploration games, we find that all bonuses lead to performance that is comparable, and in one case worse, than ϵ-greedy.

Performance across full Atari 2600 suite. One may expect a good exploration algorithm to handle the exploration/exploitation trade-off efficiently: exploring in difficult games, without losing too much in games where exploration is unnecessary. We find that performance on MONTEZUMA'S REVENGE is in fact *anticorrelated* with performance across the larger suite. Of the methods surveyed, the only one to demonstrate better performance across the ALE is the non-bonus-based Noisy Networks (Fortunato et al., 2018), which provide as an additional point of comparison. Noisy Networks perform worse on MONTEZUMA'S REVENGE.

Hyperparameter tuning procedure. The standard practice in exploration research has been to tune hyperparameters on MONTEZUMA'S REVENGE then evaluate on other Atari 2600 games. By virtue of this game's particular characteristics, this approach to hyperparameter tuning may unnecessarily increase the exploratory behaviour of exploration strategies towards a more exploratory behavior, and explain our observation that these strategies perform poorly in easier games. However, we find that tuning hyperparameters on the original ALE training set does not improve performance across Atari 2600 games beyond that of ϵ-greedy .

Amount of training data. By design, exploration methods should be sample-efficient. However, some of the most impressive gains in exploration on MONTEZUMA'S REVENGE have made use of significantly more data (1.97 billion Atari 2600 frames for RND, versus 100 million frames for the pseudo-count method). We find that, within our common evaluation framework, additional data does not play an important role on exploration performance.

Altogether, our results suggests that more research is needed to make bonus-based exploration robust and reliable, and serve as a reminder of the pitfalls of developing and evaluating methods primarily on a single domain.

1.1 RELATED WORK

Closest to our work, Burda et al. (2018) benchmark various exploration bonuses based on prediction error (Schmidhuber, 1991; Pathak et al., 2017) within a set of simulated environment including many Atari 2600 games. Their study differs from ours as their setting ignores the environment reward and instead learns exclusively from the intrinsic reward signal. Outside of the ALE, Osband et al. (2019) recently provide a collection of experiments that investigate core capabilities of exploration methods.

2 EXPLORATION METHODS

We focus on bonus-based methods, that is, methods that promote exploration through a reward signal. An agent is trained with the reward $r = r^{\text{ext}} + \beta \cdot r^{\text{int}}$ where r^{ext} is the extrinsic reward provided by the environment, r^{int} the intrinsic reward computed by agent, and $\beta > 0$ is a scaling parameter. We now summarize the methods we evaluate to compute the intrinsic reward r^{int}.

2.1 PSEUDO-COUNTS

Pseudo-counts (Bellemare et al., 2016; Ostrovski et al., 2017) were proposed as way to estimate counts in high dimension states spaces using a density model. The agent is then encouraged to visit states with a low visit count. Let ρ be a density model over the state space $\mathcal{S}$, we write $\rho_t(s)$ the density assigned to a state s after training on a sequence of states $s_1, ..., s_t$. We write $\rho'_t(s)$ the density assigned to s if ρ were to be trained on s an additional time. We require ρ to be learning positive (i.e $\rho'_t(s) \geq \rho_t(s) \, \forall s_1, ..., s_t, s \in \mathcal{S}$), the *pseudo-count* $\hat{N}$ is then defined such that updating ρ on a state s leads to a one unit increase of its pseudo-count

$$\rho_t(s) = \frac{\hat{N}(s)}{\hat{n}}, \quad \rho_t^{'}(s) = \frac{\hat{N}(s)+1}{\hat{n}+1}, \tag{1}$$

where $\hat{n}$, the pseudo-count total is a normalization constant. This formulation of pseudo-counts match empirical counts when the density model corresponds to the empirical distribution. Equation 1 can be rewritten as

$$\hat{N}(s) = \frac{\rho_t(s)(1-\rho_t^{'}(s))}{\rho_t^{'}(s)-\rho_t(s)}. \tag{2}$$

The intrinsic reward is then given by

$$r^{\text{int}}(s_t) \coloneqq (\hat{N}_t(s_t))^{-1/2}. \tag{3}$$

CTS (Bellemare et al., 2014) and PixelCNN (Van den Oord et al., 2016) have been both used as density models. We will disambiguate these agent by the name of their density model.

2.2 INTRINSIC CURIOSITY MODULE

Intrinsic Curiosity Module (ICM, Pathak et al., 2017) promotes exploration via curiosity. Pathak et al. formulates curiosity as the error in the agent's ability to predict the consequence of its own actions in a learned feature space. ICM includes three submodules: a learned embedding, a forward and an inverse model. At the each timestep the module receives a transition (s_t, a_t, s_{t+1}) – where s_t and a_t are the current state and action and s_{t+1} is the next state. States s_t and s_{t+1} are encoded into the features $\phi(s_t)$ and $\phi(s_{t+1})$ then passed to the inverse model which is trained to predict a_t. The embedding is updated at the same time, encouraging it to only model environment features that are influenced by the agent's action. The forward model has to predict the next state embedding $\phi(s_{t+1})$ using $\phi(s_t)$ and a_t. The intrinsic reward is then given by the error of the forward model in the embedding space between, $\phi(s_{t+1})$, and the predicted estimate $\hat{\phi}(s_{t+1})$:

$$r^{\text{int}}(s_t) = \|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\|_2^2. \tag{4}$$

2.3 RANDOM NETWORK DISTILLATION

Random Network Distillation (RND, Burda et al., 2019) derives a bonus from the prediction error of a random network. The intuition being that the prediction error will be low on states that are similar to those previously visited and high on newly visited states. A neural network $\hat{f}$ with parameters θ is trained to predict the output of a fixed, randomly initialized neural network f, the intrinsic reward is given by

$$r^{\text{int}}(s_t) = \|\hat{f}(s_t;\theta) - f(s_t)\|_2^2 \tag{5}$$

where θ represents the parameters of the network $\hat{f}$.

2.4 NOISY NETWORKS

We also evaluate Noisy Networks (NoisyNets; Fortunato et al., 2018), which is part of the original Rainbow implementation. NoisyNets does not explore using a bonus. Instead NoisyNets add noise in parameter space and replace the standard fully-connected layers, $y = Ws + b$, by a noisy version that combines a deterministic and a noisy stream:

$$y = (W + W_{noisy} \odot \epsilon^W)s + (b + b_{noisy} \odot \epsilon^b), \tag{6}$$

where ϵ^W and ϵ^b are random variables and $\odot$ denotes elementwise multiplication.

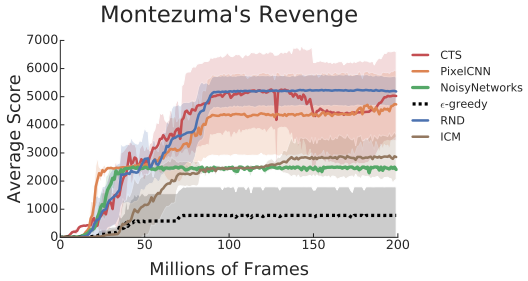


Figure 1: A comparison of different exploration methods on MONTEZUMA'S REVENGE.

3 EMPIRICAL RESULTS

In this section we present our experimental study of bonus-based exploration methods.

3.1 EXPERIMENTAL PROTOCOL

We evaluate the three bonus-based methods introduced in Section 2 as well as NoisyNets and ϵ-greedy exploration. To do so, we keep our training protocol fixed throughout our experiments. All methods are trained with a common agent architecture, the Rainbow implementation provided by the Dopamine framework (Castro et al., 2018). It includes Rainbow's three most important component: n-step updates (Mnih et al., 2016), prioritized experience replay (Schaul et al., 2015) and distributional reinforcement learning (Bellemare et al., 2017). Rainbow was designed combining several improvements to value-based agents that were developed independently. Pairing Rainbow with recent exploration bonuses should lead to further benefits. We also keep the original hyperparameters of the learning algorithm fixed to avoid introducing bias in favor of a specific bonus.

Unless specified our agents are trained for 200 million frames. Following Machado et al. (2018b) recommendations we run the ALE in the stochastic setting using sticky actions ($\varsigma = 0.25$) for all agents during training and evaluation. We also do not use the mixed Monte-Carlo return (Bellemare et al., 2016; Ostrovski et al., 2017) or other algorithmic improvements that are combined with bonus-based methods (e.g. Burda et al., 2019).

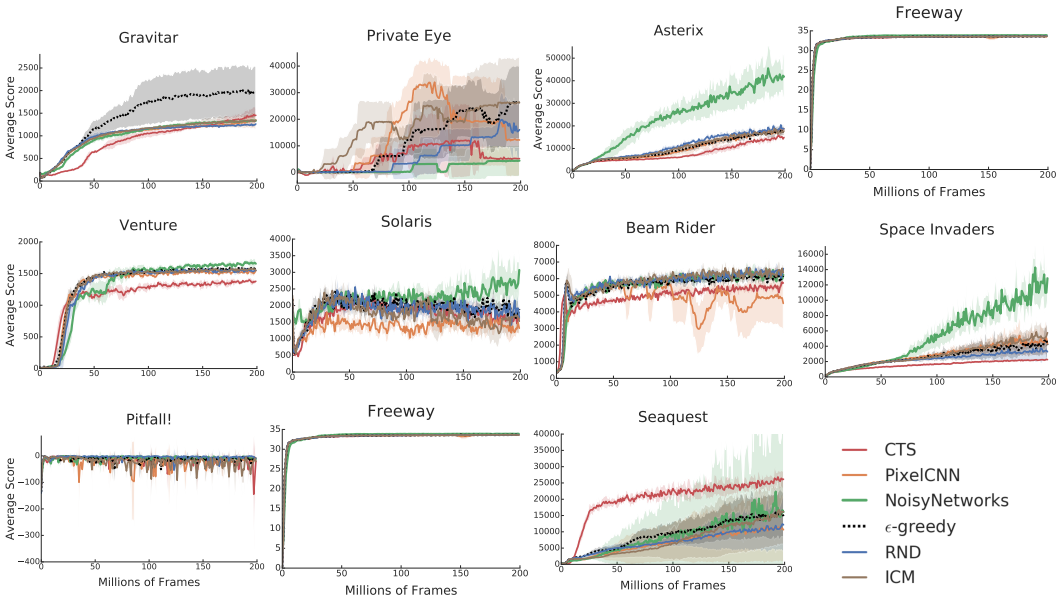
3.2 COMMON EVALUATION FRAMEWORK

Our first set of results compare exploration methods introduced in Section 2 on MONTEZUMA'S REVENGE. We follow the standard procedure to use this game to tune the hyperparameters of each exploration bonus. Details regarding implementation and hyperparameter tuning may be found in Appendix A. Figure 1 shows training curves (averaged over 5 random seeds) for Rainbow augmented with the different exploration bonuses.

As anticipated, ϵ-greedy exploration performs poorly here and struggles in such a hard exploration game. On the other hand, exploration bonuses have a huge impact and all eventually beat the baselines ϵ-greedy and NoisyNets. Only ICM is unable to surpass the baselines by a large margin. RND, CTS and PixelCNN all average close to 5000 points. Interestingly, we observe that a more recent bonus like RND does not lead to higher performance over the older pseudo-count method. Of note, the performance we report at 200 millions frames improves on the performance reported in the original paper for each method. As expected, Rainbow leads to increased performance over older DQN variants previously used in the literature.

3.3 EXPLORATION METHODS EVALUATION

It is standard in the exploration community to, first tune hyperparameters on MONTEZUMA'S REVENGE, and then, evaluate these hyperparameters on the remaining hard exploration games with sparse rewards. This is not in line with the original ALE guidelines which recommend to evaluate on the entire suite of Atari 2600 games. In this section we provide empirical evidence that, failing to follow the ALE guidelines may interfere with the conclusions of the evaluation.



(a) Hard exploration games.

(b) Atari training set.

Figure 2: Evaluation of different bonus-based exploration methods on several Atari games, curves are averaged over 5 runs, shaded area denotes variance.

3.3.1 HARD EXPLORATION GAMES

We now turn our attention to the set of games categorized as hard exploration games with sparse rewards in Bellemare et al.'s taxonomy (the taxonomy is available in Appendix B). It includes ALE's most difficult games for exploration, this is where good exploration strategies should shine and provide the biggest improvements. These games are: GRAVITAR, PRIVATE EYE, VENTURE, SOLARIS, PITFALL! and FREEWAY.

We evaluate agents whose hyperparameters were tuned on MONTEZUMA'S REVENGE on this set of games. Training curves averaged over 5 runs are shown in Figure 2a. We find that performance of each method on MONTEZUMA'S REVENGE does not correlate with performance on other hard exploration domains. All methods seem to behave similarly contrary to our previous observations on MONTEZUMA'S REVENGE. In particular, there is no visible difference between ϵ-greedy and more sophisticated exploration bonuses. ϵ-greedy exploration is on par with every other method and even outperforms them by a significant margin on GRAVITAR. These games were initially classified to be hard exploration problems because a DQN agent using ϵ-greedy exploration was unable to achieve a high scoring policy; it is no longer the case with stronger learning agents available today.

3.3.2 FULL ATARI SUITE

The set of hard exploration games was chosen to highlight the benefits of exploration methods, and, as a consequence, performance on this set may not be representative of an algorithm capabilities on the whole Atari suite. Indeed, exploration bonuses can negatively impact performance by skewing the reward landscape. To demonstrate how the choice of particular evaluation set can lead to different conclusions we also evaluate our agents on the original Atari training set which includes the games ASTERIX, FREEWAY, BEAM RIDER, SPACE INVADERS and SEAQUEST. Except for FREEWAY all these games are easy exploration problems (Bellemare et al., 2016).

Figure 2b shows training curves for these games. Pseudo-counts with a CTS model appears to struggle when the environment is an easy exploration problem and end up performing worse on every game except SEAQUEST. RND and ICM are able to consistently match ϵ-greedy exploration but never beat it. It appears that bonus-based methods have limited benefits in the context of easy

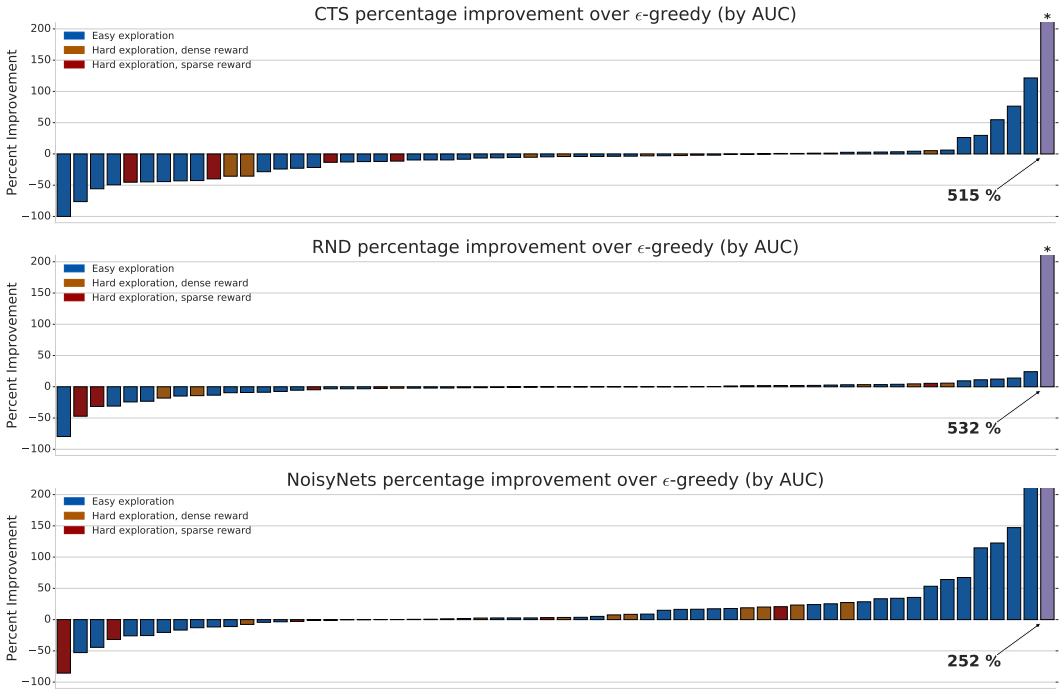


Figure 3: Improvements (in percentage of AUC) of Rainbow with various exploration methods over Rainbow with ϵ-greedy exploration in 60 Atari games. The game MONTEZUMA'S REVENGE is represented in purple.

exploration problems. Finally, despite its limited performance on MONTEZUMA'S REVENGE, we found that NoisyNets gave the most consistent improvements.

Overall, while the Atari training set and the set of the hard exploration games both show that bonus-based method only provide marginal improvements they lead us to different conclusions regarding the best performing exploration scheme. It appears that to fully quantify the behavior of exploration methods one cannot avoid evaluating on the whole Atari suite. We do so and add the remaining Atari games to our study. See Figure 3 for a high level comparison for CTS, RND and NoisyNets (PixelCNN and ICM are available in Figure 12 in the Appendix). Altogether, it appears that the benefit of exploration bonuses is mainly allocated towards MONTEZUMA'S REVENGE. None of them, except NoisyNets seems to improve over ϵ-greedy by significant margin. Bonus-based methods may lead to increased performance on a few games but seem to deteriorate performance by a roughly equal amount on other games. We could have hoped that these methods focus their attention on hard exploration games at the expense of easier ones, meaning they trade exploitation for exploration. Unfortunately it does not seem to be the case as they do not exhibit a preference for hard exploration games. We may wonder if these methods are overfitting on MONTEZUMA'S REVENGE.

3.4 HYPERPARAMETER TUNING PROCEDURE

Previous experiments have so far depicted a somewhat dismal picture of bonus-based methods, in particular their penchant to overfit on MONTEZUMA'S REVENGE. Though it is unfortunate to see they do not generalize to the full Atari gamut, one may wonder if their tendency to overfit to MONTEZUMA'S REVENGE specifically is caused by the hyperparameter tuning procedure or their inherent design. The experiments in this section aim at addressing this issue. We ran a new hyperparameter sweep on CTS and RND using a new training set. We chose these algorithms as we found them to be particularly sensitive to their hyperparameters. The new training set includes the games PONG, ASTERIX, SEAQUEST, Q*BERT and BREAKOUT. This set as been previously used by others as a training set to tune reinforcement learning agents (Bellemare et al., 2017; Dabney et al., 2018).

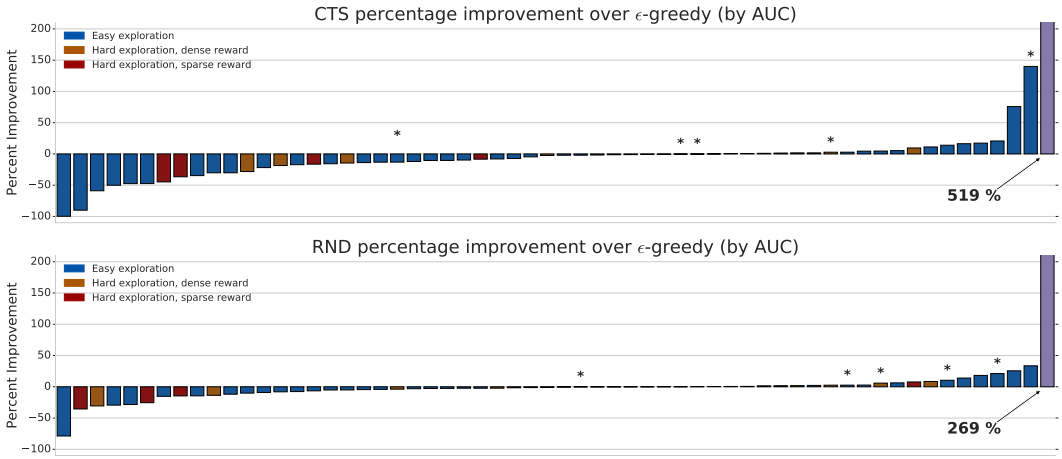


Figure 4: Improvements (in percentage of AUC) of Rainbow with CTS and RND over Rainbow with ϵ-greedy exploration in 60 Atari games when hyperparameters have been tuned on SEAQUEST, QBERT, PONG, BREAKOUT and ASTERIX. The game MONTEZUMA'S REVENGE is represented in purple. Games in the training set have stars on top of their bar.

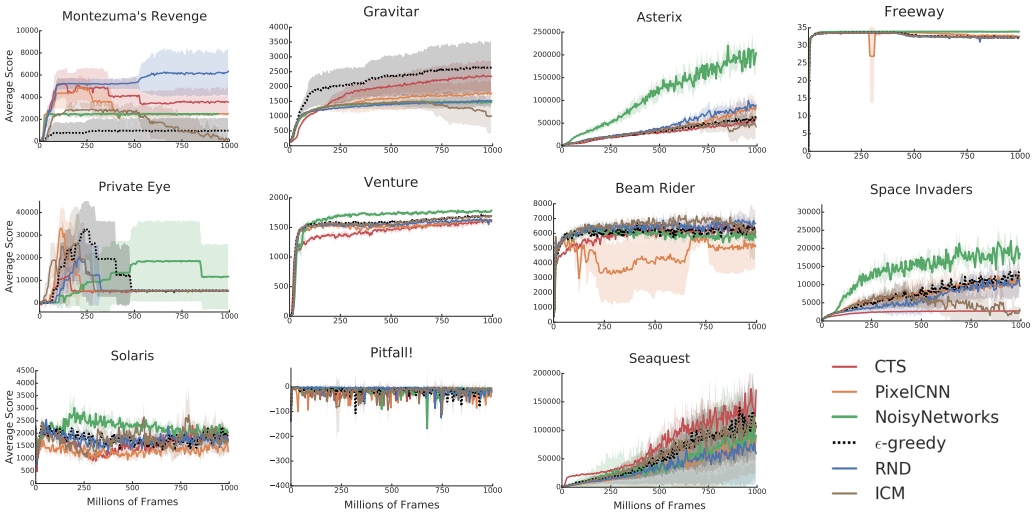
Results are depicted in Figure 4. Both algorithm still perform much better than ϵ-greedy exploration on MONTEZUMA'S REVENGE. As it is to be expected, performance also increased for games within the training set. A notable example is CTS on SEAQUEST which now improves over the ϵ-greedy baseline by 140% instead of 120% previously. Nevertheless, the conclusions from Section 3.3.2 remain valid. Bonus-based exploration strategies still provide only limited value except for a few games. In addition, neither algorithm seems to achieve better results on easy exploration problems outside of the training set.

3.5 AMOUNT OF TRAINING DATA

Current results so far showed that exploration bonuses provide little benefits in the context of exploration in the ALE. Nonetheless, our experiments have been limited to 200 millions environment frames, it is possible that with additional data exploration bonuses would perform better. This would be in line with an emerging trend of training agents an order of magnitude longer in order to produce a high-scoring policy, irrespective of the sample cost (Espeholt et al., 2018; Burda et al., 2019; Kapturowski et al., 2019). In this section we provide experiments that contradict this hypothesis. We reuse agents trained in Section 4.2.2 and lengthen the amount training data they process to 1 billion environment frames. We use Atari 2600 games from the set hard exploration with sparse rewards and the Atari training set. See Figure 5 for training curves. On easier exploration problems all exploration strategies see their score gracefully scale with additional data. In hard exploration games, none of the exploration strategies seem to benefit from receiving more data. Score on most games seem to plateau and may even decrease. This is particularly apparent in MONTEZUMA'S REVENGE where only RND actually benefits from longer training. After a while, agents are unable to make further progress and their bonus may collapse. When that happens, bonus-based methods cannot even rely on ϵ-greedy exploration to explore and may therefore see their performance decrease. This behavior may be attributed to our evaluation setting, tuning hyperparameters to perform best after one billion training frames will likely improve results. It is however unsatisfactory to see that exploration methods do not scale effortlessly as they receive more data. In practice, recent exploration bonuses have required a particular attention to handle large amount of training data (e.g. Burda et al., 2019).

4 CONCLUSION

Recently, many exploration methods have been introduced with confounding factors within their evaluation – longer training duration, different model architecture and other ad-hoc techniques. This practice obscures the true value of these methods and makes it difficult to prioritize more promising directions of investigation. Therefore, following a growing trend in the reinforcement learning



(a) Hard exploration games.

(b) Atari training set.

Figure 5: Evaluation of bonus-based methods when training is extended to one billion frames

community, we advocate for better practices on empirical evaluation for exploration to fairly assess the contribution of newly proposed methods. In a standardized training environment and context, we found that current bonus-based methods are unable to surpass ϵ-greedy exploration. As a whole this suggest progress in bonus-based exploration may have been driven by confounding factors rather than improvement in the bonuses. This shows that more work is still needed to address the exploration problem in complex environments.

We may wonder why do we not see a positive impact of bonus-based exploration. One possible explanation is our use of the relatively sophisticated Rainbow learning algorithm. The benefits that others have observed using exploration bonuses might be made redundant through other mechanisms already in Rainbow, such as a prioritized replay buffer. An important transition may only be observed with low frequency but it can be sampled at a much higher rate by the replay buffer. As a result, the agent can learn from it effectively without the need for encouragement from the exploration bonus to visit that transition more frequently. Though the merits of bonus-based methods have been displayed with less efficient learning agents, these benefits did not carry on to improved learning agents. This is disappointing given that the simple NoisyNets baseline showed efficient exploration can still achieve noticeable gains on the ALE. As of now, the exploration and credit assignment communities have mostly been operating independently. Our work suggests that they may have to start working hand in hand to design new agents to make reinforcement learning truly sample efficient.

REFERENCES

Marc Bellemare, Joel Veness, and Erik Talvitie. Skip context tree switching. In *International Conference on Machine Learning*, pp. 1458–1466, 2014.

Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.

Marc G. Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pp. 1471–1479, 2016.

Marc G. Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 449–458. JMLR. org, 2017.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. *arXiv preprint arXiv:1808.04355*, 2018.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *Proceedings of the International Conference on Learning Representations*, 2019.

Pablo Samuel Castro, Subhodeep Moitra, Carles Gelada, Saurabh Kumar, and Marc G. Bellemare. Dopamine: A research framework for deep reinforcement learning. *arXiv preprint arXiv:1812.06110*, 2018.

Will Dabney, Mark Rowland, Marc G. Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. Go-explore: a new approach for hard-exploration problems. *arXiv*, 2019.

Lasse Espeholt, Hubert Soyer, Rémi Munos, Karen Simonyan, Volodymyr Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *ICML*, 2018.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, et al. Noisy networks for exploration. In *Proceedings of the International Conference on Learning Representations*, 2018.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Steven Kapturowski, Georg Ostrovski, John Quan, Remi Munos, and Will Dabney. Recurrent experience replay in distributed reinforcement learning. In *Proceedings of the International Conference on Learning Representations*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Marlos C. Machado, Marc G. Bellemare, and Michael Bowling. Count-Based Exploration with the Successor Representation. *CoRR*, abs/1807.11622, 2018a.

Marlos C. Machado, Marc G. Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018b.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, pp. 1928–1937, 2016.

Ian Osband, Yotam Doron, Matteo Hessel, John Aslanides, Eren Sezener, Andre Saraiva, Katrina McKinney, Tor Lattimore, Csaba Szepezvari, Satinder Singh, et al. Behaviour suite for reinforcement learning. *arXiv preprint arXiv:1908.03568*, 2019.

Georg Ostrovski, Marc G. Bellemare, Aäron van den Oord, and Rémi Munos. Count-based exploration with Neural Density Models. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2721–2730. PMLR, 2017.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the International Conference on Machine Learning*, 2017.

Tobias Pohlen, Bilal Piot, Todd Hester, Mohammad Gheshlaghi Azar, Dan Horgan, David Budden, Gabriel Barth-Maron, Hado van Hasselt, John Quan, Mel Veerk, Matteo Hessel, Rmi Munos, and Olivier Pietquin. Observe and look further: Achieving consistent performance on atari. *arXiv*, 2018.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

Jürgen Schmidhuber. Curious model-building control systems. In *Proc. international joint conference on neural networks*, pp. 1458–1463, 1991.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # Exploration: A Study of Count-Based Exploration for Deep Reinforcement Learning. In *Advances in Neural Information Processing Systems*, pp. 2750–2759, 2017.

Aaron Van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, 2016.

A HYPERPARAMETER TUNING

All bonus based methods are tuned with respect to their final performance on MONTEZUMA'S REVENGE after training on 200 million frames averaged over five runs. When different hyper parameter settings led to comparable final performance we chose the one that achieve the performance the fastest.

A.1 RAINBOW AND ATARI PREPROCESSING

We used the standard Atari preprocessing from Mnih et al. (2015). Following Machado et al. (2018b) recommendations we enable sticky actions and deactivated the termination on life loss heuristic. The remaining hyperparameters were chosen to match Hessel et al. (2018) implementation.

Hyperparameter	Value
Discount factor γ	0.99
Min history to start learning	80K frames
Target network update period	32K frames
Adam learning rate	6.25×10^{-5}
Adam ϵ	1.5×10^{-4}
Multi-step returns n	3
Distributional atoms	51
Distributional min/max values	[-10, 10]

Every method except NoisyNets is trained with ϵ-greedy following the scheduled used in Rainbow with ϵ decaying from 1 to 0.01 over 1M frames.

A.2 HYPERPARAMETER TUNING ON MONTEZUMA'S REVENGE

A.2.1 NOISYNETS

We did not tune NoisyNets. We kept the original hyperparameter $\sigma_0 = 0.5$ as in Fortunato et al. (2018) and Hessel et al. (2018).

A.2.2 PSEUDO-COUNTS

We followed Bellemare et al.'s preprocessing, inputs are 42×42 greyscale images, with pixel values quantized to 8 bins.

CTS: We tuned the scaling factor β. We ran a sweep for $\beta \in \{0.5, 0.1, 0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001\}$ and found that $\beta = 0.0005$ worked best.

PixelCNN: We tuned the scaling factor and the prediction gain decay constant c. We ran a sweep with the following values: $\beta \in \{5.0, 1.0, 0.5, 0.1, 0.05\}$, $c \in \{5.0, 1.0, 0.5, 0.1, 0.05\}$ and found $\beta = 0.1$ and $c = 1.0$ to work best.

A.3 ICM

We tuned the scaling factor and the scalar α that weighs the inverse model loss against the forward model. We ran a sweep with $\alpha = \{0.4, 0.2, 0.1, 0.05, 0.01, 0.005\}$ and $\beta = \{2.0, 1.0, 0.5, 0.1, 0.05, 0.01, 0.005, 0.001, 0.0005\}$. We found $\alpha = 0.005$ and $\beta = 0.005$ to work best.

A.4 RND

Following Burda et al. (2019) we did not clip the intrinsic reward while the extrinsic reward was clipped. We tuned the reward scaling factor β and learning rate used in Adam (Kingma & Ba, 2014) by the RND optimizer. We ran a sweep with $\beta = \{0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001, 0.0005, 0.00001, 0.000005\}$ and lr $= \{0.005, 0.001, 0.0005, 0.0002, 0.0001, 0.00005\}$. We found that $\beta = 0.00005$ and lr $= 0.0001$ worked best.

A.5 HYPERPARAMETER TUNING ON SECOND TRAINING SET (SECTION 3.4)

A.5.1 PSEUDO-COUNTS CTS

We ran a sweep for $\beta \in \{0.1, 0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001\}$ and found that $\beta = 0.0001$ worked best.

A.6 RND

We did not tune the learning rate and kept lr $= 0.0001$. We ran a sweep for $\beta = \{[0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001, 0.00005\}$ and found that $\beta = 0.00005$ worked best.

B TAXONOMY OF EXPLORATION ON THE ALE

Bellemare et al.'s taxonomy of exploration propose a classification of Atari 2600 games in terms of difficulty of exploration. It is provided in Table 1.

Table 1: A classification of Atari 2600 games based on their exploration difficulty.

Easy Exploration			Hard exploration	
Human Optimal		Score Exploit	Dense Reward	Sparse Reward
ASSAULT	ASTERIX	BEAM RIDER	ALIEN	FREEWAY
ASTEROIDS	ATLANTIS	KANGAROO	AMIDAR	GRAVITAR
BATTLE ZONE	BERZERK	KRULL	BANK HEIST	MONTEZUMA'S REVENGE
BOWLING	BOXING	KUNG-FU MASTER	FROSTBITE	PITFALL!
BREAKOUT	CENTIPEDE	ROAD RUNNER	H.E.R.O	PRIVATE EYE
CHOPPER CMD	CRAZY CLIMBER	SEAQUEST	MS. PAC-MAN	SOLARIS
DEFENDER	DEMON ATTACK	UP N DOWN	Q*BERT	VENTURE
DOUBLE DUNK	ENDURO	TUTANKHAM	SURROUND	
FISHING DERBY	GOPHER		WIZARD OF WOR	
ICE HOCKEY	JAMES BOND		ZAXXON	
NAME THIS GAME	PHOENIX			
PONG	RIVER RAID			
ROBOTANK	SKIING			
SPACE INVADERS	STARGUNNER			

C ADDITIONAL FIGURES

The variance of the return on MONTEZUMA'S REVENGE is high because the reward is a step function, for clarity we also provide all the training curves in Figure 6

We also provide training curves presented in the main paper in larger format.

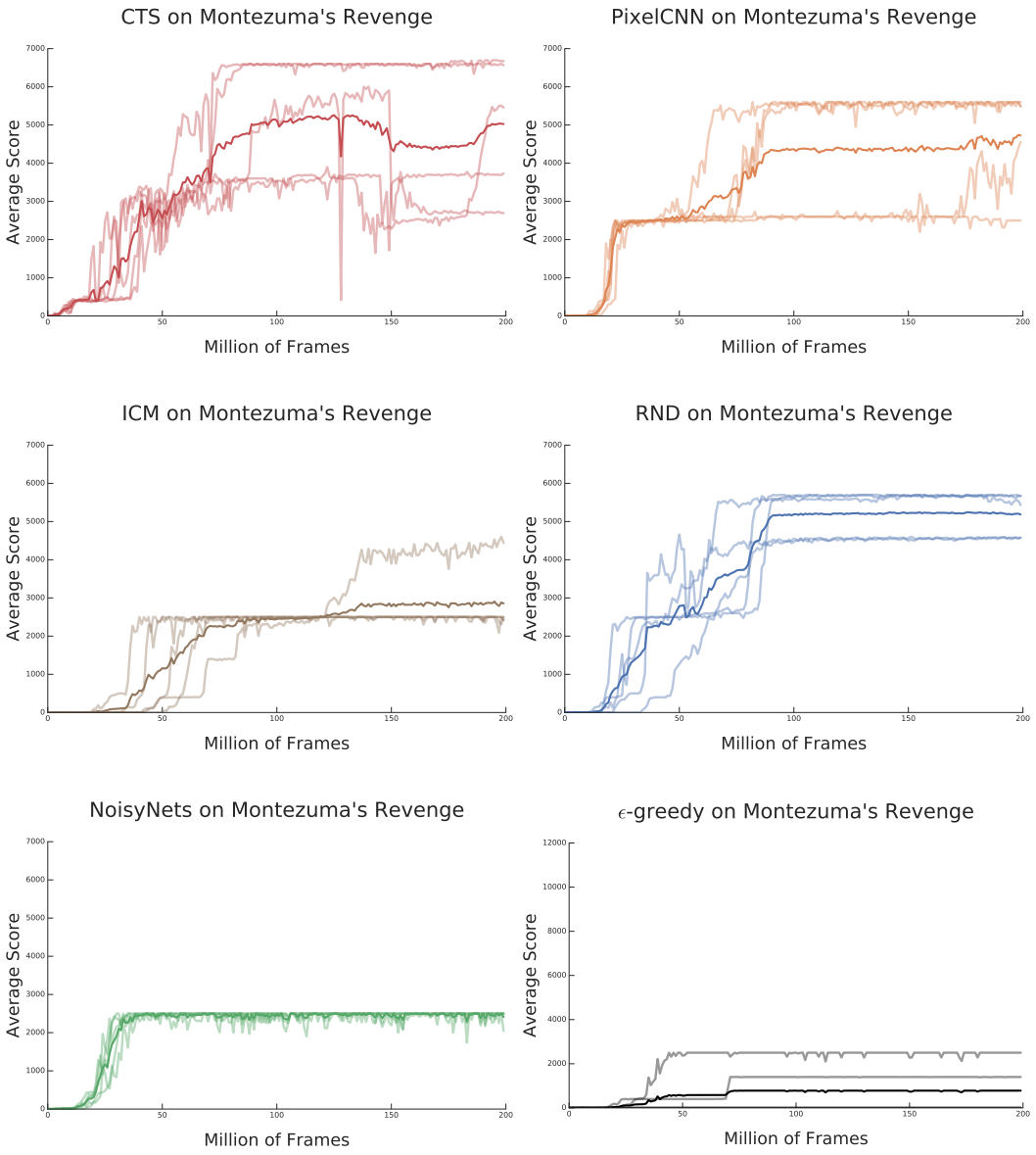


Figure 6: Training curves on MONTEZUMA'S REVENGE

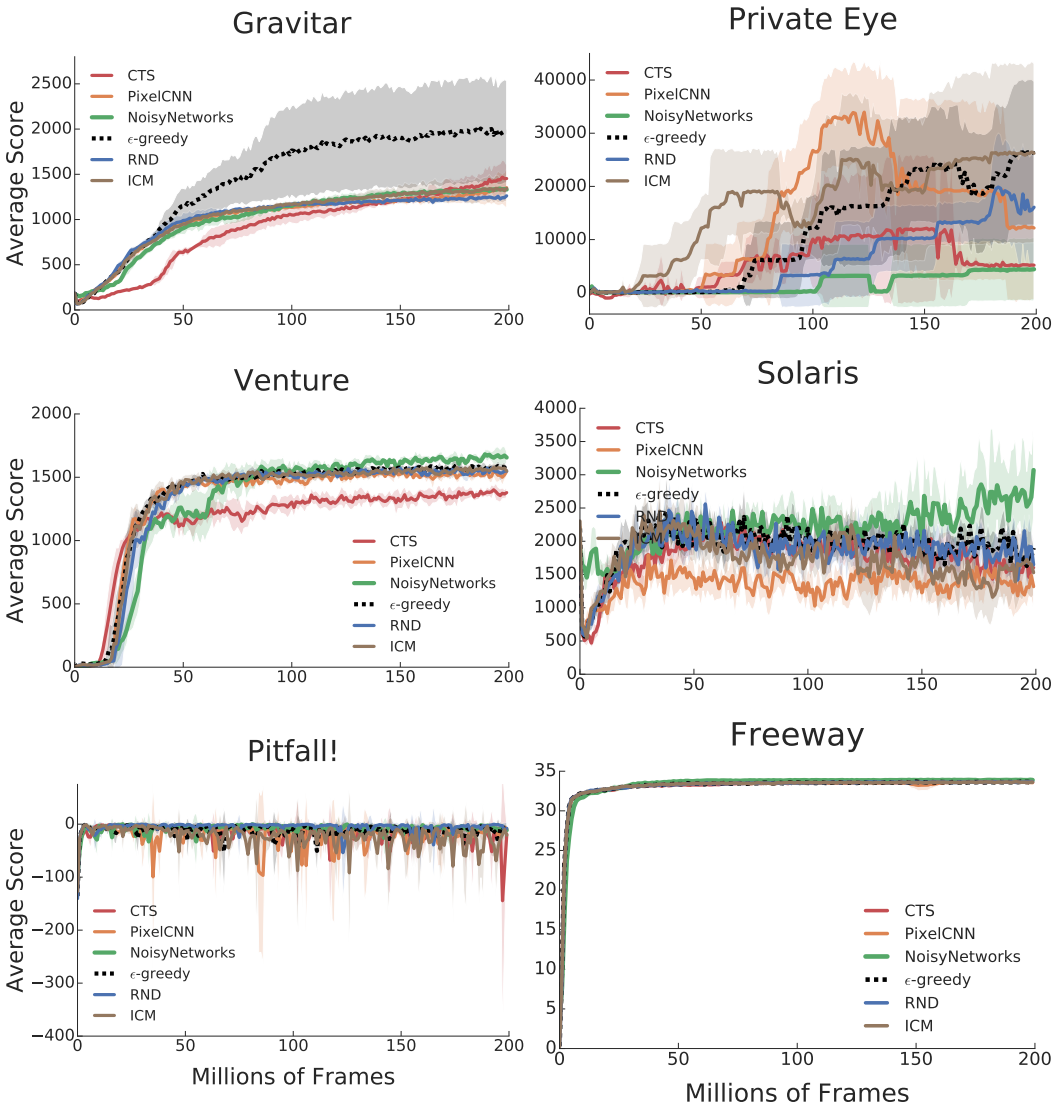


Figure 7: Evaluation of different bonus-based exploration methods on the set of hard exploration games with sparse rewards. Curves are average over 5 runs and shaded area denotes variance.

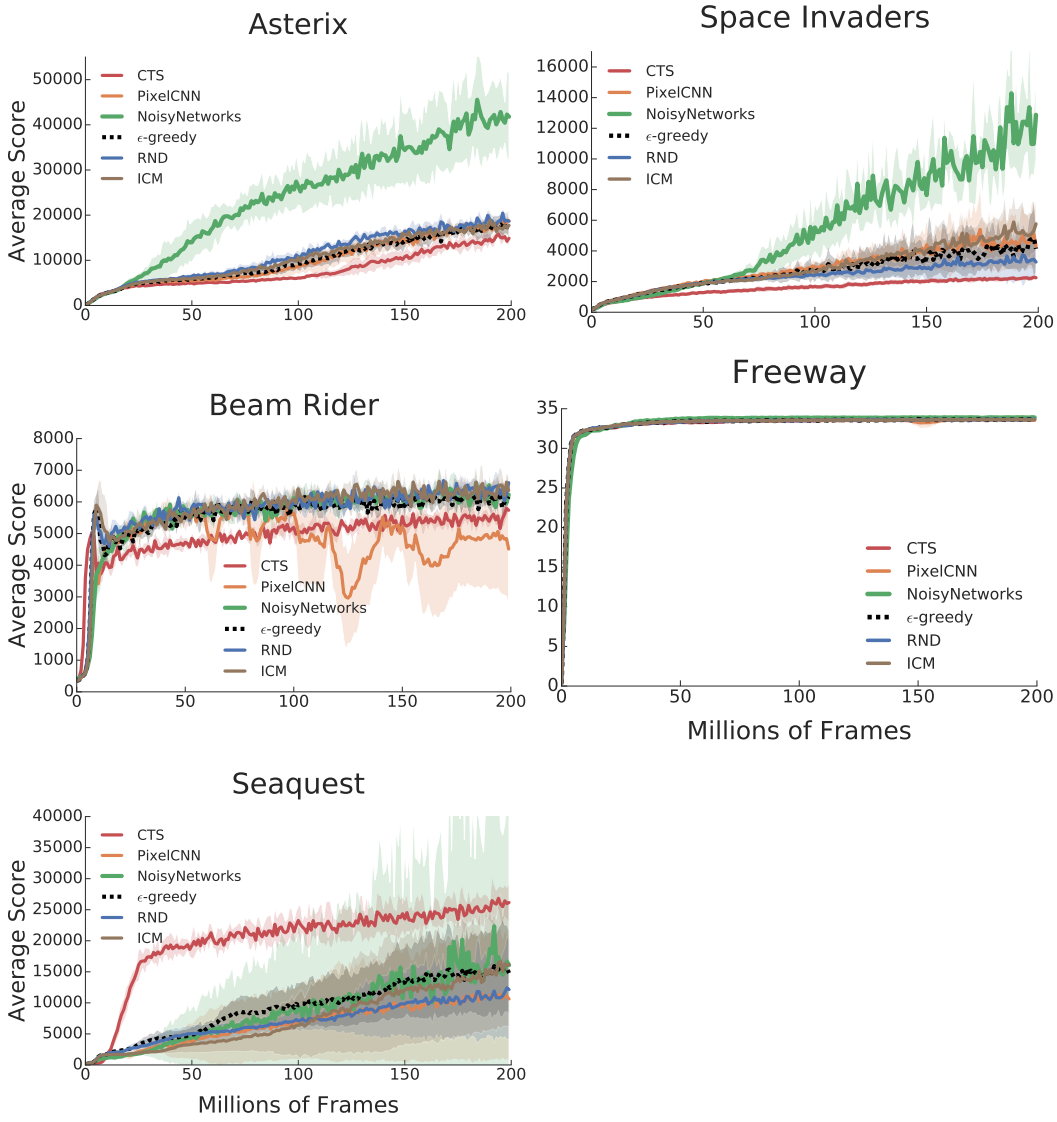


Figure 8: Evaluation of different bonus-based exploration methods on the Atari training set. Curves are average over 5 runs and shaded area denotes variance.

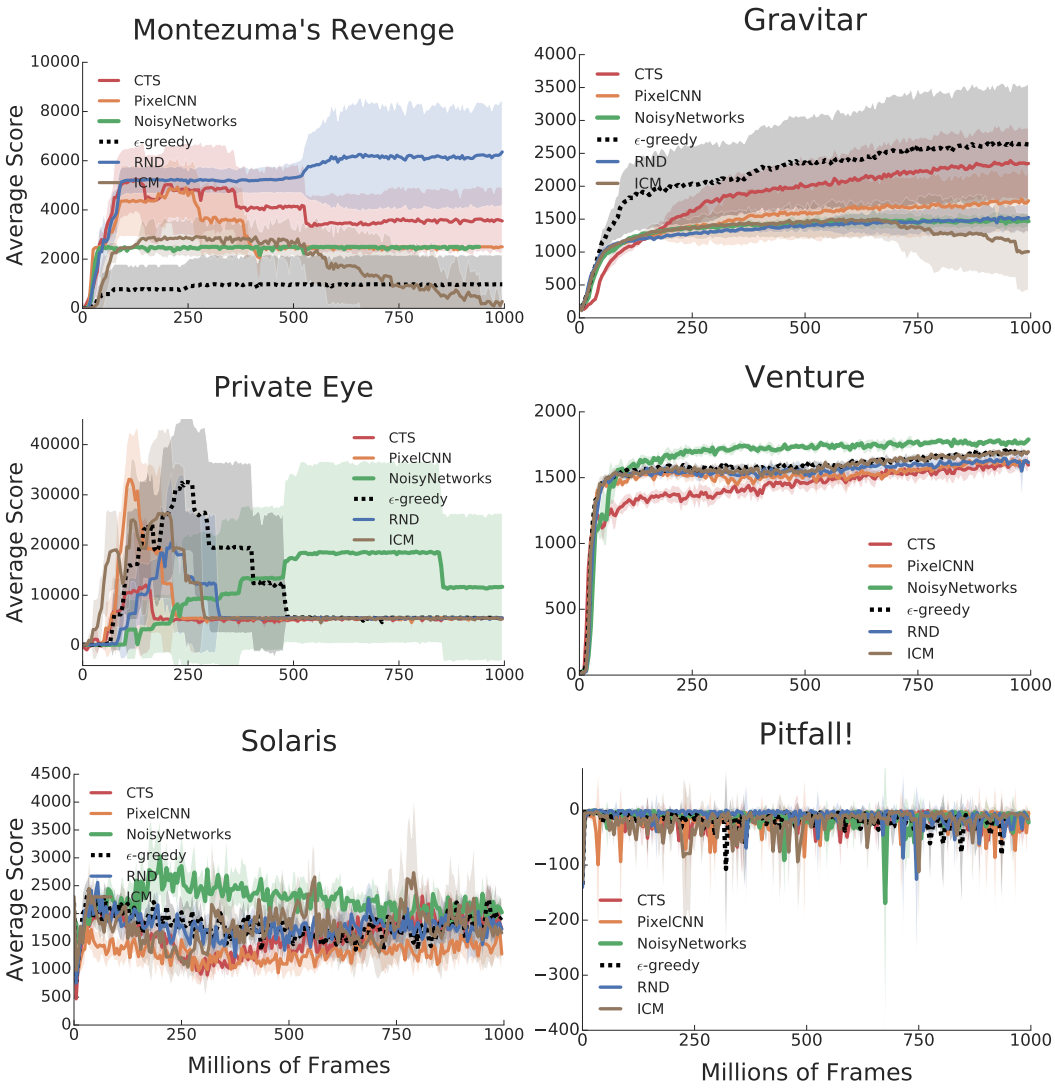


Figure 9: Evaluation of different bonus-based exploration methods on the set of hard exploration games with sparse rewards. Exploration methods are trained for one billion frames. Curves are average over 5 runs and shaded area represents variance.

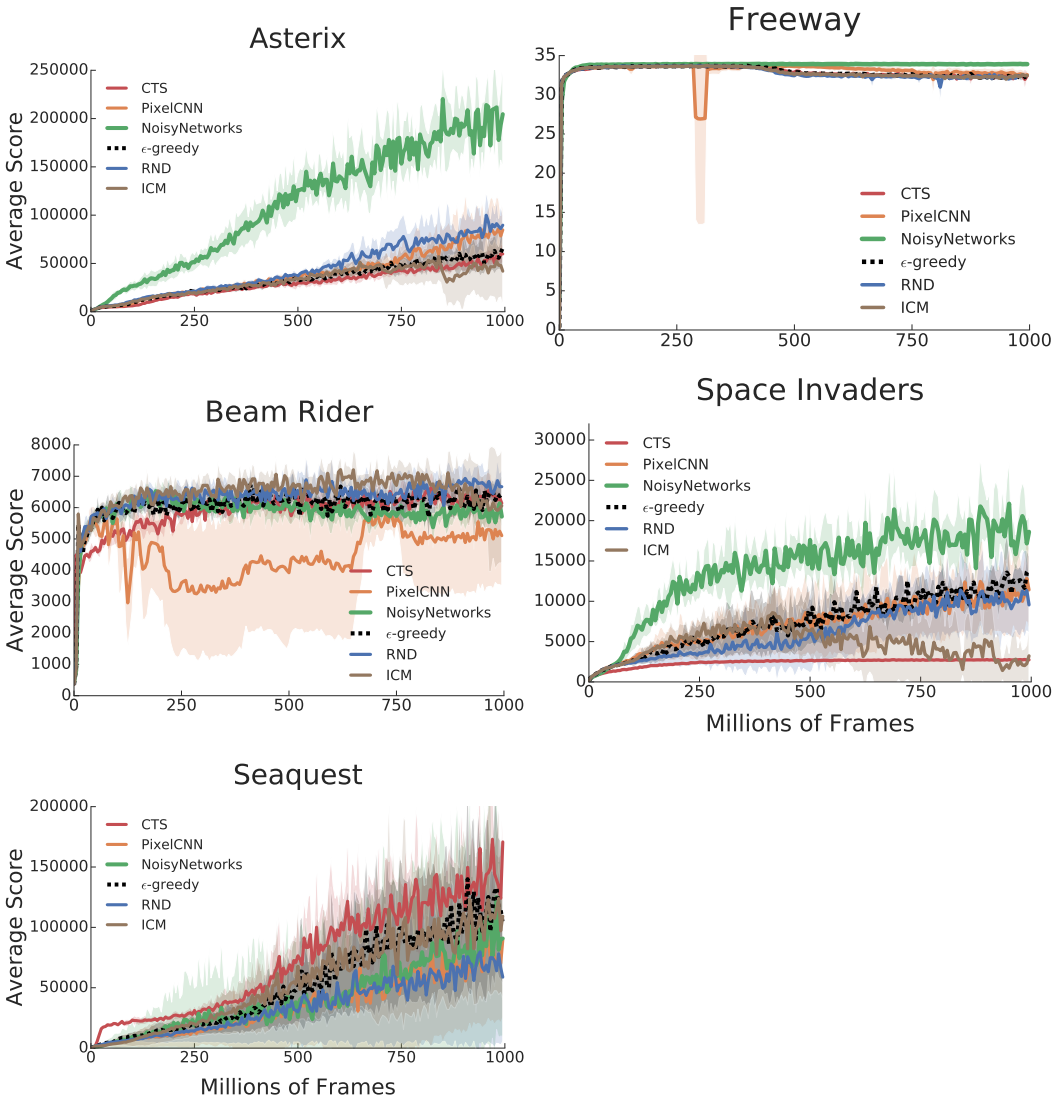


Figure 10: Evaluation of different bonus-based exploration methods on the Atari training set. Exploration methods are trained for one billion frames. Curves are average over 5 runs and shaded area represents variance.

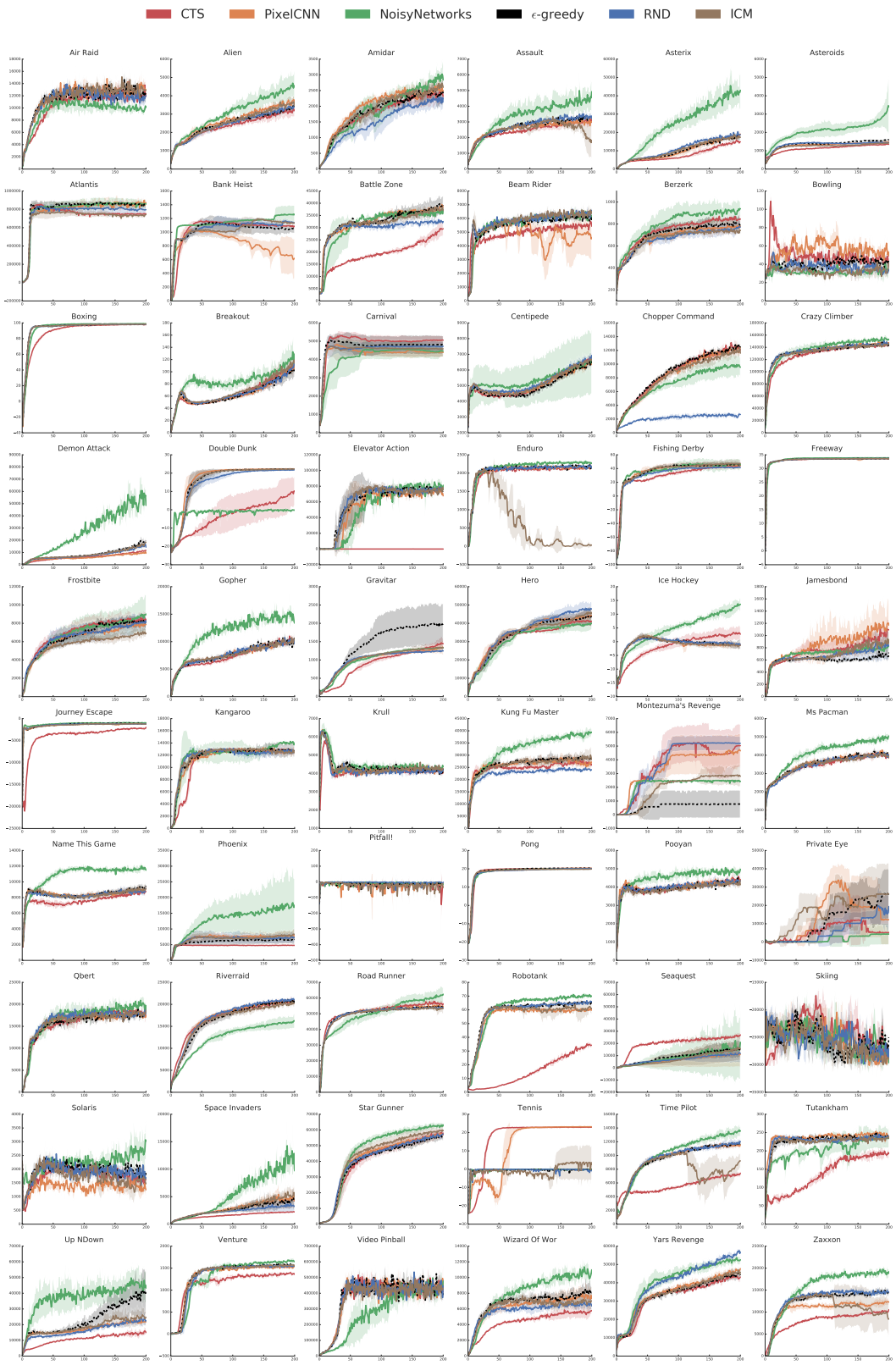


Figure 11: Training curves of Rainbow with ϵ-greedy, CTS, PixelCNN, NoisyNets, ICM and RND.

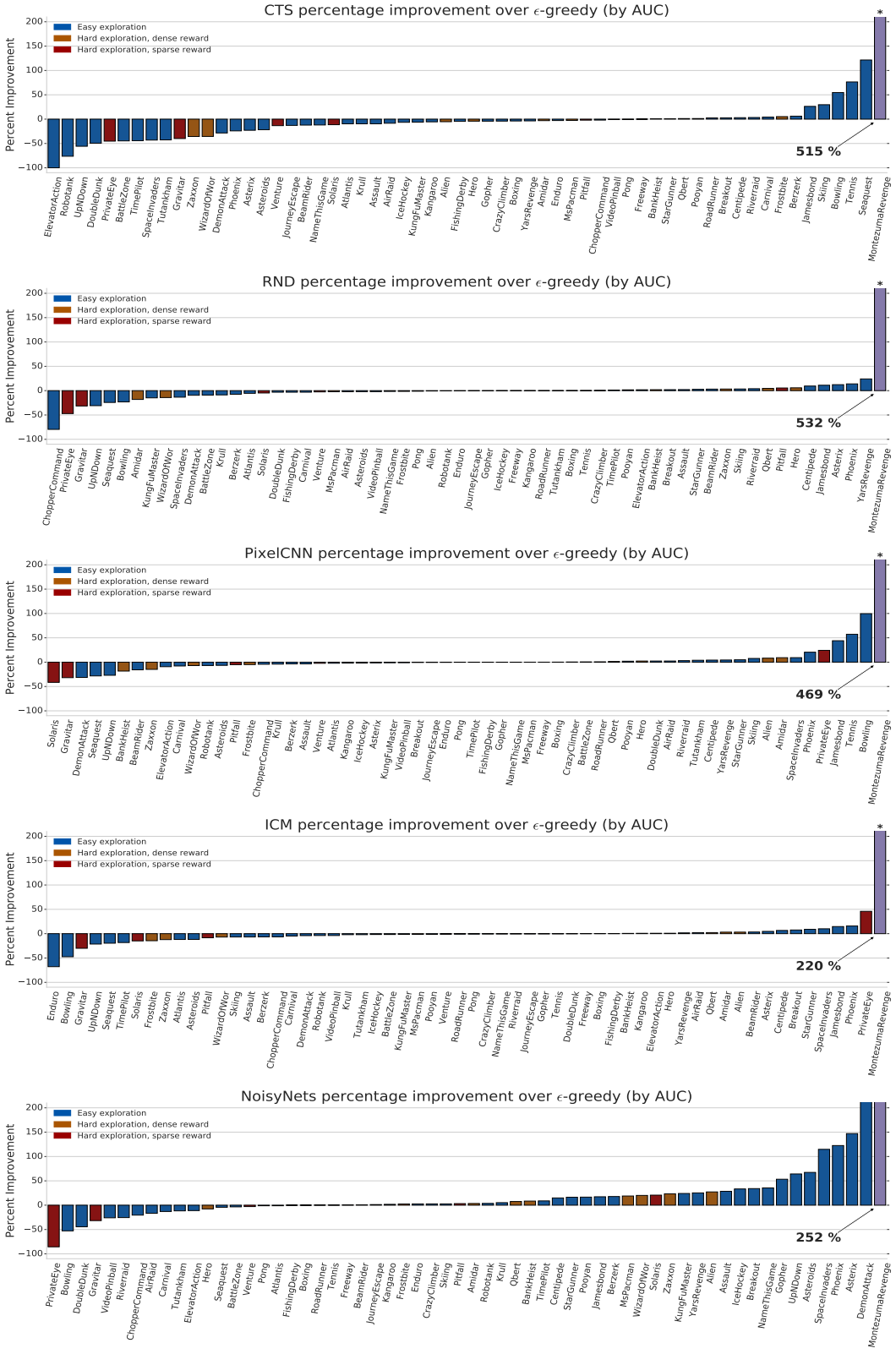


Figure 12: Improvements (in percentage of AUC) of Rainbow with various exploration methods over Rainbow with ϵ-greedy exploration in 60 Atari games when hyperparameters are tuned on MONTEZUMA'S REVENGE. The game MONTEZUMA'S REVENGE is represented in purple.

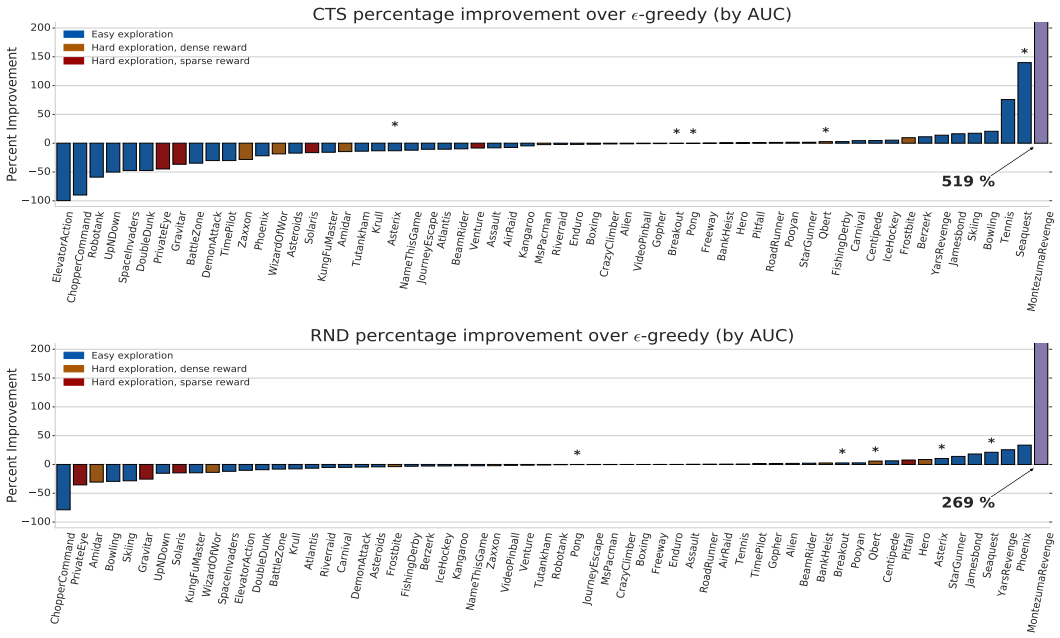


Figure 13: Improvements (in percentage of AUC) of Rainbow with CTS and RND over Rainbow with ϵ-greedy exploration in 60 Atari games when hyperparameters have been tuned on SEAQUEST, QBERT, PONG, BREAKOUT and ASTERIX. The game MONTEZUMA'S REVENGE is represented in purple. Games in the training set have stars on top of their bar.