EXPLORATION USING DISTRIBUTIONAL RL AND UCB

Anonymous authors
Paper under double-blind review

ABSTRACT

We establish the relation between Distributional RL and the Upper Confidence Bound (UCB) approach to exploration. In this paper we show that the density of the Q function estimated by Distributional RL can be successfully used for the estimation of UCB. This approach does not require counting and, therefore, generalizes well to the Deep RL. We also point to the asymmetry of the empirical densities estimated by the Distributional RL algorithms like QR-DQN. This observation leads to the reexamination of the variance's performance in the UCB type approach to exploration. We introduce truncated variance as an alternative estimator of the UCB and a novel algorithm based on it. We empirically show that newly introduced algorithm achieves better performance in multi-armed bandits setting. Finally, we extend this approach to high-dimensional setting and test it on the Atari 2600 games. New approach achieves better performance compared to QR-DQN in 26 of games, 13 ties out of 49 games.

1 INTRODUCTION

Exploration is a long standing problem in Reinforcement Learning (RL). It's been the main focus of the multi-armed bandits literature. Here the algorithms are easier to design and analyze. However, these solutions are quite unfeasible for high dimensional Deep RL setting, where the complication comes from the presence of the function approximator.

The multi-armed bandit can be represented by a slot machine with several arms. Each arms' expected reward is unknown to the gambler. Her/his goal is to maximize cumulative reward by pulling bandit's arms. If the true expected rewards are known, then the best strategy is to pull the arm with the highest value. However, gambler only observes stochastic reward after the arm is pulled. One possible solution described by Sutton et al. (1998) is to initialize values of arms' estimated means optimistically and then improve the estimates by pulling the same arm again. Arm with a lower true mean will get its estimate decreased over time. Eventually, the best arm will be discovered. The drawback is that the set of the arms has to be enumerated and every arm has to be pulled infinitely many times. In the RL setting an arm corresponds to a state-action pair, which implies that both assumptions are too strong for the Deep RL.

Another line of reasoning is Upper Confidence Bound (UCB) type algorithms, e.g. UCB-1, introduce by Kocsis & Szepesvári (2006). The essence of the approach is nicely summarized by Audibert et al. (2009): 'optimism in the face of uncertainty principle'. The idea is statistically intuitive: pull the arm which has the highest upper confidence bound, hoping for a better mean. Estimation of the arm's UCB is performed via Hoeffdings Inequality[1] which is entirely based on counting the number of times the arm was pulled. UCB extends to the tree search case in the form of UCT developed by Kocsis & Szepesvári (2006). Although this idea was successfully applied to the problem when perfect model is accessible, i.e. AlphaGo by Silver et al. (2016), it does not generalize in a straightforward fashion to the general Deep RL setting without perfect model. The main obstacle is the requirement of counting of the state-action pairs. Another popular variation is UCB-V introduced by Audibert et al. (2009). It estimates UCB via the empirical variance, which again involves counting. Therefore, the requirement of counting prevents UCB ideas from successful generalization to the high dimensional setting of Deep RL.

[1] Proved by Hoeffding (1963)

The generalization of exploration ideas from multi-armed bandits to Deep RL is challenging. Therefore, one the most popular exploration approaches in Deep RL is the annealed epsilon greedy approach popularized by Mnih et al. (2015). However, epsilon greedy approach is not very efficient, especially in Deep RL. It does not take into account the underlying structure of the environment. Therefore, researchers have been looking for other more efficient ways of exploration in Deep RL setting. For example the idea of parametric noise was explored by Fortunato et al. (2017). Posterior sampling for reinforcement learning (Osband et al. (2013)) in Deep RL setting was developed by Osband et al. (2016). Uncertainty Bellman Equation proposed by O'Donoghue et al. (2017), generalizes Bellman equation to the uncertainty measure. The closest UCB type approach was developed by Chen et al. (2017). In order to avoid counting authors estimate UCB based on the empirical distribution of the Q function produced by Bootstrapped DQN (Osband et al. (2016)). The approach reduces to estimating an ensemble of randomly initialized Q functions. According to the averaged human normalized learning curve the performance improvement was insignificant. Currently, there is a much better approach to estimating empirical distributions of Q function, i.e. distributional RL (Bellemare et al. (2017), Dabney et al. (2017)). The results in the distributional RL are both theoretically sound and achieve state of the art performance in Deep RL environments, like Atari 2600. However, we should note that Distributional RL does not use the whole distribution, but only the mean.

Another important characteristic of Distributional RL is that both C51 (Bellemare et al. (2017)) and Quantile Regression DQN (QR-DQN) (Dabney et al. (2017)) are non parametric in the sense that the estimated distribution is not assumed to belong to any specific parametric family. Hence, it is not assumed to be symmetric or even unimodal [2]. We argue that in the case of asymmetric distributions, variance might become less sensitive in estimating UCB. This problem seems to be overseen by the existing literature. However, this issue might become more important in a more general setting, when symmetric assumption is not simply relaxed but is a very rare case. We empirically show in the Section 4 that symmetry is in fact rare in Distributional RL.

In this paper we build upon generic UCB idea. We generalize it to the asymmetric distributions and high-dimensional setting. In order to extend UCB approach to asymmetric distributions, we introduce truncated variability measure and show empirically that it achieves higher performance than variance in bandits setting. Extension of this measure to rich visual environments provided by Atari 2600 platform is based on recent advances in Distributional RL.

2 BACKGROUND

2.1 UCB

As it was mentioned in the introduction UCB is based on the statistical relation between the number of observations of a random variable and the tightness of the mean estimates based on these observations. More formally the connection is provided the inequality proved by Hoeffding (1963):

Theorem 1 (Hoeffdings Inequality) *Let $X_1, \dots, X_t$ be independent random variables bounded by $[0; 1]$. Let $X = \frac{1}{t} \sum_{i=1}^{t} X_i$, $= \mathbb{E}[X]$. Then for $u \geq 0$:*

$$Pr(X \quad + u) \quad e^{-2tu^2} \tag{1}$$

Therefore, Theorem 1 quantifies the relation between upper confidence bound for X and the number of realizations of X. On the other hand if the estimate of the probability density function (PDF) is available, then UCB can be estimated directly:

$$X \quad + c \frac{p}{\hat{}^2} \tag{2}$$

where $\hat{}^2$ is the variance computer from $\hat{P}(X)$ and c is a constant reflecting the confidence level. Now the question is how to estimate the empirical PDF. Bayes-UCB introduced by Kaufmann et al. (2012) uses restricted family of distributions which allows for the closed form Bayesian update. On the other hand it is possible to model $P[X]$ in a more expressive way using Neural Networks as it is done in the Distributional RL. We lose closed form solutions, but gain more realistic estimates of the underlying distribution. In this paper we explore Quantile Regression approach towards Distributional RL, which we introduce next.

[2] See analysis by Bellemare et al. (2017).

2.2 DISTRIBUTIONAL RL

The core idea behind QR-DQN is the Quantile Regression introduced by Koenker & Bassett Jr (1978). Let us first describe QR in the supervised machine learning setting. Given $\{(y_i, x_i)\}_i$, τ-th linear quantile regression loss is defined as:

$$L(\theta) = \sum_i \rho_\tau(y_i - \theta x_i) \tag{3}$$

where

$$\rho_\tau(u) = u(\tau - \mathbb{I}_{u<0}) = |u|\mathbb{I}_{u>0}\tau + (1-\tau)|u|\mathbb{I}_{u<0} \tag{4}$$

is the weighted sum of residuals. Weights are proportional to the counts of the residual signs and order of the estimated quantile. For higher quantiles positive residuals get higher weight and vice versa. If $\tau = 0.5$, then the estimate of the median for y_i is $Q_{0.5}(y_i|x_i) = x_i\hat{\theta}$, with $\hat{\theta} = \arg\min L(\theta)$.

Dabney et al. (2017) introduced QR in a more general setting of RL with function approximator. The design of the neural network is the same as in the original DQN introduced by Mnih et al. (2015) except for the last linear layer, which outputs N quantiles $\{\theta_i\}_i$ instead of the a single estimate of Q. For a given transition (x, a, r, x') and a discount factor γ the Bellman update is:

$$\forall j \in 1, \dots, N \quad T\theta_j = r + \gamma\theta_j(x', a^*) \tag{5}$$

Note the similarity between loss in 3 and Algorithm 1. The difference is in the type of the function approximator: linear in the former and neural network in the later. This work is closely related to that of Bellemare et al. (2017) with a major contribution in the way the empirical distribution is estimated. Bellemare et al. (2017) use Boltzmann distribution in conjunction with a projection step. QR-DQN seems to be a more elegant solution, since it does not involve the projection step and and there is no need for explicitly bounding the support.

It is worth emphasizing that function approximator in this case is crucial for generalization of UCB approach to DeepRL, since it eliminates the need for state-action pair counting.

Algorithm 1 Quantile Regression Q-learning

Input: $w, w', (x, a, r, x'), \gamma \in [0, 1)$. network weights, sampled transition, discount factor

1: $Q(x', a') = \sum_j q_j \theta_j(x', a', w')$
2: $a^* = \arg\max_{a'} Q(x, a')$
3: $T\theta_j = r + \gamma\theta_j(x', a^*, w')$
4: $L(w) = \sum_i \frac{1}{N} \sum_j [\rho_{\hat{\tau}_i}(T\theta_j - \theta_i(x, a, w))]$
5: $w' = \arg\min_w L(w)$

Output: w' . Updated weights of θ

3 ALGORITHM

QR approach allows for a very elegant estimation of distributions in the RL setting. We apply this idea to the multi-armed bandits. This environment is more tractable and easier to explore. Since the distributions are not assumed to have any regularities, like symmetry, we conjecture that variance might not the best UCB measure. Therefore, we explore truncated variability measure. We, then, generalize this idea to the Deep RL setting.

3.1 QUCB

We propose to estimate empirical distribution of returns for each arm by the means of the QR. The basic idea is to estimate mean and variance of the return for each arm based on the empirical distribution provided by QR and pick the arm according to the highest mean plus standard deviation. We call this algorithm QUCB.

In the setting of multi-armed bandits denote quantiles by $\{\tau_i\}_i$ and observed reward for by R. Then for a single arm the set of estimates of quantiles is the solution to:

$$\arg\min_{\{\theta_i\}_i} \sum_i \rho_{\tau_i}(\theta_i - R) \tag{6}$$

As opposed to the supervised example there are no features. Algorithm 2 outlines the QUCB. Note the definitions of $\theta_{t;k}$ and $Var(\theta_{t;k})$:

$$\theta_{t;k} = \frac{1}{N}\sum_{i=1}^{N} \theta_{t;k;i} \tag{7}$$

$$Var(\theta_{t;k}) = \frac{1}{N}\sum_{i=1}^{N} (\theta_{t;k} - \theta_{t;k;i})^2 \tag{8}$$

Note the presence of the multiplier c_t in the Algorithm 2. To ensure the optimality in the limit $t \to \infty$, $\{c_t\}_t$ have to be chosen so that $\lim_{t\to\infty} c_t = 0$. In case the number of quantiles is big compared to the sample size, it might help to warm-up the initial estimates by performing a few steps of pure exploration ($c = 1$).

Hence, the algorithm estimates empirical quantiles. QR approach allows to estimate any quantile. In fact it allows to estimate multiple quantiles in one update step producing empirical density. More importantly, having empirical distribution at hand opens up the way for new possible approaches to computing upper confidence bound, exploration bonuses or some other means of ordering empirical distributions when choosing action/arm. One such approach is developed in the next section.

Algorithm 2 QUCB

Input: $\{\theta_{0;k;i}\}_{k;i}; N; \gamma; \{c_t\}_t$. Initial values, number of quantiles, learning rate, schedule

1: **for** t in [0, number of runs] **do**
2: **if** t < number of burn-in steps **then**
3: $I_t \sim Uniform(K)$. Pick an arm randomly
4: **else**
5: $I_t = \arg\max_k \{\theta_{t;k} + c_t \sqrt{Var(\theta_{t;k})}\}$
6: **end if**
7: draw reward R_t
8: $\theta_{t+1;I_t;i} = \theta_{t;I_t;i} + \gamma \sum_i \rho_{\tau_i}(\theta_{t;I_t;i} - R_t)$. Update quantiles of the corresponding arm
9: **end for**

3.2 Assymetry, truncated measures, QUCB+

In the case of non parametric approaches or parametric approaches that are not exclusively restricted to symmetric distributions the symmetry is not guaranteed. In fact, it might be a very rare case as it can be seen from Figure 1. Note that the game of Pong from Atari 2600 is the simplest one. In the end of training presented in Figure 1, agent achieves almost the perfect score. Hence the distributions in the end of training correspond to the near optimal policy and these distributions are not symmetric. That is, the asymmetry of empirical distributions is regular case, but not an exception. Hence, the question: is the variance a 'good' measure of the upper confidence bound for asymmetric distributions in the setting of multi-armed bandits and RL?

For the sake of the argument consider a simple decomposition of the variance into the two truncated variability measures: lower and upper variance[3]:

$$Var(X) = \frac{1}{N}\sum_{i=1}^{N}(X - X_i)^2 = \frac{1}{2N}\sum_{i=1}^{\frac{1}{N}}(X - X_i)^2 + \frac{1}{2N}\sum_{i=\frac{1}{2N}}^{N}(X - X_i)^2 = \sigma_l^2 + \sigma_u^2 \tag{9}$$

It is clear that in the case of a symmetric probability density function (PDF) the lower and upper variances are equivalent. However, in the case of asymmetric distribution the equality $\sigma_l^2 = \sigma_u^2$ does

[3] In case N is odd, partition indexes into lower and upper sets as $\{1, \ldots, \lfloor \frac{N}{2} \rfloor\}$, $\{\lceil \frac{N}{2} \rceil, \ldots, N\}$.

Figure 1: Pong. (left) Empirical distributions of the Q function for a single action obtained from QR-DQN-1 during training for 20 millions of frames. y-axis is the training step number. (right) Standard deviation of the empirical distribution of the Q function for a single action.

not always hold[4]. ^2_l contains the information about the lower tail variability whereas ^2_u about upper tail variability.

In case of the UCB type approach to the exploration problem upper tail variability seem to be more relevant than lower tail one, especially if the estimated PDF is asymmetric. Intuitively speaking, ^2_u is an optimistic measure of variability. ^2_u is biased towards rare 'positive' rewards. However, the availability of empirical PDF makes it possible to truncate the variance in many different ways. ^2_u might be a good candidate, although one potential draw back is the robustness of the estimator of the mean. In order to mitigate that, we propose the following truncated measure of the variability based on the median rather than the mean[5]

$$^2_+ = \frac{1}{2N} \sum_{i=\frac{N}{2}}^{N} (\quad _i)^2 \tag{10}$$

where $_i$'s are $\frac{i}{N}$-th quantiles. As opposed to ^2_u, $^2_+$ captures some 'negative' variability but still being optimistic.

We propose QUCB+, the algorithm based on $^2_+$ which is a slight modi cation of QUCB. Instead of the $Var(_{t;k})$ we propose to use $^2_+$. We hypothesize that $^2_+$ might be a more robust upper tail variability measure. We support our hypothesis by empirical results in multi-armed bandits setting and Atari 2600, presented in Section 4.

3.3 DQN-QUCB+

The ideas presented in the previous section generalize in a straightforward fashion to the tabular RL and most importantly to the Deep RL setting. As it was mentioned QR-DQN's output is the quantile distribution with equispaced quantiles $\{\frac{i}{N}\}_{i=1}^{N}$. The update step does not change. Action selection step incorporates bias in the form of $^2_+$ from Equation 10. Algorithm 3 outlines DQN-QUCB+.

In the presence of the function approximator the variance encoded in $_i$ is largely effected by the variation in the parameters. Therefore, the variance produced by QR-DQN has at least two sources: intrinsic variation coming from the underlying MDP and parametric variation. The important question is the dynamics of the parametric uncertainty during training. As it can be seen from Figure 1 the variance drops signi cantly meaning that the parametric uncertainty goes down as the network approaches optimal solution. Hence, if the model tends to learn then the parametric component in the variance decreases.

[4] Consider discrete empirical distribution with support $\{-1; 0; 2\}$ and probability atoms $\{\frac{1}{3}; \frac{1}{3}; \frac{1}{3}\}$.

[5] For details see Huber (2011), Hampel et al. (2011)

Algorithm 3 DQN-QUCB+

Input: $w; w^{-}; (x, a, r, x'); \gamma \in [0, 1)$. network weights, sampled transition, discount factor

1: $Q(x', a') = \sum_j q_{ij}(x', a', w^{-})$

2: $a^{*} = \arg\max_{a'}(Q(x, a') + c_t \sqrt{\frac{2}{+}})$

3: $T_j = r + \gamma \beta(x', a^{*}; w^{-})$

4: $L(w) = \sum_i \frac{1}{N} \sum_j [\hat{}_i(T_j - {}_i(x, a; w))]$

5: $w' = \arg\min_w L(w)$

Output: w' . Updated weights of ()

Figure 2: Comparison of performance of QUCB and QUCB+ in multi-armed bandits with 10 arms. Lines represent averages over 2000 runs, bands are standard errors. (left) The rewards are normally distributed with unit variance. (right) Rewards are drawn from (right- and left- skewed) lognormal distribution with unit variance.

4 EXPERIMENTS

4.1 MULTI-ARMED BANDITS

Following Sutton et al. (1998) we applied QUCB and QUCB+ to the multi-armed bandits test bed. In order to study the effect of asymmetric distributions we set up two con gurations of the test bed: with normally and asymmetrically distributed rewards. Both con gurations consist of 10 arms. In both con gurations true means of arms $\{_i\}_{k=1}^{K}$ are drawn from the normal distribution with $= 1$, $= 1$.

In the rst con guration. During each step the the reward for k-th arm is drawn from $N(_k; 1)$. As it can be seen from 2 there is no statistically signi cant difference between QUCB and QUCB+. It is expected, since the the rewards are normally distributed, hence, symmetric around the mean. In addition median and mean of the normal distribution coincide. Therefore, estimate of ^{2} is close to that of $\frac{2}{+}$. In the second con guration the reward for the best arm is drawn from the lognormal distribution centered at $_k$ and variance 1. And the rewards for other arms are drawn from the re ected about $_k$ lognormal distribution with variance one. Hence true variances of all arms are the same, however in the presence of slight asymmetry QUCB+ performs better, see Figure 2.

4.2 ATARI 2600

As it was claimed earlier in the paper the QUCB generalizes to the Deep RL setting in the straightforward fashion. The architecture of the network is that of QR-DQN. Dabney et al. (2017) experimented with two losses: the original QR loss and Huber loss with $= 1$. Both architectures proved to be stable. For our experiments we chose only one loss: the Huber loss with $= 1$[6], due to high

[6] QR-DQN with $= 1$ is denoted as QR-DQN-1 in the work by Dabney et al. (2017). We use QR-DQN and QR-DQN-1 interchangeably.

Figure 3: Atari 2600 Venture game. Online training curves for QUCB with vanishing schedule and QUCB with constant schedule. Curves are averaged over 3 runs. Shaded area represents corresponding standard errors.

computational costs of experiments. Another reason for picking the Huber loss is its smoothness compared to L_1 loss of QR. Smoothness is better suited for gradient descent methods. Overall, we followed closely Dabney et al. (2017) in setting the hyper parameters, except for the learning rate of the Adam optimizer which we set to $= 0.0001$.

The most significant distinction is the way the exploration is performed in DQN-QUCB. As opposed to QR-DQN there is no epsilon greedy exploration schedule in DQN-QUCB. Exploration is performed via the $\frac{2}{+}$ term only.

An important hyper parameter which is introduced by DQN-QUCB is the schedule, i.e. the sequence of multipliers for $\frac{2}{+}$, $\{c_t\}_t$. The choice depends on the specific problem. In case of the stationary environment the UCB term involving $\frac{2}{+}$ should eventually vanish, i.e. $c_t \to 1$. In the non stationary environment the agent might always need to explore, so that c_t is eventually a constant[7]

In our experiments we used the following schedule:

$$c_t = 50 \sqrt{\frac{\log t}{t}} \tag{11}$$

The motivation behind the schedule is to gradually vanish the exploration term as it is done in QR-DQN. This makes performance comparison more adequate. During experiments we observed that DQN-QUCB is sensitive to the schedule to some extent, see for example Figure 3. We conjecture that tuning the schedule might yield better performance across games.

We evaluated DQN-QUCB on the set of 49 Atari games initially proposed by Mnih et al. (2015). Algorithms were evaluated on 40 million frames[8] 3 runs per game. The summary of the results is

[7] Here by is eventually a constant we mean $\exists T \forall t \quad T \; c_t = c$

[8] Equivalently, 10 million agent steps.

Figure 4: Cumulative rewards performance comparison of DQN-QUCB+ and QR-DQN-1. The bars represent relative gain/loss of DQN-QUCB+ over QR-DQN-1.

presented in Figure 4. DQN-QUCB achieves better performance (gain of 3%) or with respect to cumulative reward measure in 26 games.

We argue that cumulative reward is a suitable performance measure for our experiments, since none of the learning curves exhibit plummeting behaviour[9]. A more detailed discussion of this point is presented in Machado et al. (2017).

5 CONCLUSIONS

Recent advancements in RL, namely Distributional RL, not only established new theoretically sound principles but also achieved state-of-the-art performance in challenging high dimensional environments like Atari 2600. The by-product of the Distributional RL is the empirical PDF for the Q function which is not directly used except for the mean computation. UCB on the other hand is a very attractive exploration algorithm in the multi-armed bandits setting, which does not generalize in a straightforward fashion to Deep RL.

In this paper we established the connection between the UCB idea and Distributional RL. We also pointed to the asymmetry of the PDFs estimated by Distributional RL, which is not a rare exception but rather the only case. We introduced truncated variability measure as an alternative to the variance and empirically showed that it can be successfully applied to multi-armed bandits and rich visual environments like Atari 2600. It is highly likely that DQN-QUCB+ might be improved through schedule tuning. DQN-QUCB+ might be combined with other advancements in Deep RL, e.g. Rainbow by Hessel et al. (2017), to yield better results.

[9] We present leaning curves for all 49 games in the Appendix.

REFERENCES

Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. Theoretical Computer Science, 410(19):1876–1902, 2009.

Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. arXiv preprint arXiv:1707.06887, 2017.

Richard Y Chen, Szymon Sidor, Pieter Abbeel, and John Schulman. Ucb exploration via q-ensembles. arXiv preprint arXiv:1706.01502, 2017.

Will Dabney, Mark Rowland, Marc G Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. arXiv preprint arXiv:1710.10044, 2017.

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. arXiv preprint arXiv:1706.10295, 2017.

Frank R Hampel, Elvezio M Ronchetti, Peter J Rousseeuw, and Werner A Stahel. Robust statistics: the approach based on in uence functions, volume 196. John Wiley & Sons, 2011.

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. arXiv preprint arXiv:1710.02298, 2017.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. Journal of the American statistical association, 58(301):13–30, 1963.

Peter J Huber. Robust statistics. In International Encyclopedia of Statistical Science, pp. 1248–1251. Springer, 2011.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On bayesian upper con dence bounds for bandit problems. In Arti cial Intelligence and Statistics, pp. 592–600, 2012.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In European conference on machine learning, pp. 282–293. Springer, 2006.

Roger Koenker and Gilbert Bassett Jr. Regression quantiles. Econometrica: journal of the Econometric Society, pp. 33–50, 1978.

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. arXiv preprint arXiv:1709.06009, 2017.

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.

Brendan O'Donoghue, Ian Osband, Remi Munos, and Volodymyr Mnih. The uncertainty bellman equation and exploration. arXiv preprint arXiv:1709.05380, 2017.

Ian Osband, Daniel Russo, and Benjamin Van Roy. (more) ef cient reinforcement learning via posterior sampling. In Advances in Neural Information Processing Systems, pp. 3003–3011, 2013.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In Advances in neural information processing systems, pp. 4026–4034, 2016.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. nature, 529(7587):484, 2016.

Richard S Sutton, Andrew G Barto, et al. Reinforcement learning: An introduction. MIT press, 1998.