As Expected? An Analysis of Distributional Reinforcement Learning

Clare Lyle [1] Marc G. Bellemare [2]

Abstract

Distributional reinforcement learning, in which an agent predicts distributions of returns instead of their expected values, has seen empirical success in several Atari 2600 games, outperforming both the human baseline and previously state-of-the-art algorithms. It remains unclear precisely what drives this improvement in performance over traditional reinforcement learning approaches. In this paper, we take initial steps towards answering this question by determining under what conditions the distributional perspective leads to behaviour different from what one would see in the expected case, and conversely when they are equivalent. We supplement the theoretical findings presented in this paper with empirical results in tabular settings.

1. Introduction and Related work

The reinforcement learning problem consists of an agent interacting with an environment so as to maximize cumulative reward. We formalize the notion of an environment with a Markov Decision Process (MDP) defined as the tuple $(\mathcal{X}, A, R, P, \gamma)$, where $\mathcal{X}$ denotes the state space, A the set of possible actions, $R : \mathcal{X} \times A \to \mathbb{R}$ the reward function, P the transition probability kernel, and $\gamma \in [0, 1]$ the discount factor. We denote by π a policy in the MDP, i.e. $\pi(a|x)$ is the agent's probability of choosing action a in state x. The policy is evaluated on each state-action pair as the discounted sum of expected future rewards after visiting the state-action pair and choosing actions according to π.

$$Q^\pi(x, a) \equiv \mathbb{E}_{\pi,P}\left[\sum_{t=0}^{\infty} \gamma^t R(x_t, a_t) \middle| x_0 = x, a_0 = a\right] \quad (1)$$

One can also express this value as the fixed point of the Bellman operator T^π, defined as

$$T^\pi Q(x, a) = R(x, a) + \gamma \mathbb{E}_{P,\pi}[Q(x', a')] \quad (2)$$

Instead of considering only the expected return from a state-action pair, one can construct an analogous distributional Bellman operator for random variables, as shown by Bellemare et al. (2017a):

$$T^\pi Z(x, a) \overset{D}{=} R(x, a) + \gamma \sum_{x', a'} \pi(a'|x') P(x'|x, a) Z^\pi(x', a'). \quad (3)$$

We highlight that this is equality in distribution, and not an equality of random variables. To make this more explicit, we note that equation (3) is really an equality of probability measures (Morimura & Kashima, 2010). The concept of a distance between two probability measures naturally plays an integral role in distributional reinforcement learning. We will place particular emphasis on the Cramér metric (Székely, 2003) in the results that follow. Many motivations for using this metric are outlined in (Bellemare et al., 2017b).

Definition 1 (Cramér Metric)**.** *Let p, q be two probability distributions with CDFs F_p, F_q. The Cramér metric ℓ_2 between p and q is defined as follows:*

$$\ell_2(p, q) = \sqrt{\int_{\mathbb{R}} (F_p(x) - F_q(x))^2 dx}$$

We will overload notation and write equivalently $\ell_2(p, q) \equiv \ell_2(F_p, F_q)$ or, when X and Y are random variables with laws p and q $\ell_2(p, q) \equiv \ell_2(X, Y)$.

The Cramér metric can also be used to define a projection onto the set of distributions of some fixed support (Rowland et al., 2018). This notion will be of use to us when we wish to approximate observed distributions.

Definition 2 (Cramér Projection)**.** *Let $\mathbf{z} = z_1 \le \cdots \le z_K \in \mathbb{R}$. For a Dirac measure δ_y, the Cramér projection $\Pi_C(\delta_y)$ onto the support $\mathbf{z}$ is defined by:*

$$\Pi_C(\delta_y) = \begin{cases} \delta_{z_1} & \text{if } y \le z_1 \\ \frac{z_{i+1} - y}{z_{i+1} - z_i} \delta_{z_i} + \frac{y - z_i}{z_{i+1} - z_i} \delta_{z_{i+1}} & \text{if } z_i < y \le z_{i+1} \\ \delta_{z_K} & \text{if } y > z_K \end{cases}$$

[1] School of Computer Science, McGill University, Montreal, Canada [2] Google Brain, Montreal, Canada. Correspondence to: Clare Lyle <clare.lyle@mail.mcgill.ca>.

Accepted at the FAIM workshop "Prediction and Generative Modeling in Reinforcement Learning", Stockholm, Sweden, 2018.

2. Problem Formulation

Given the discrepancy in performance between the distributional C-51 algorithm introduced by Bellemare et al. (2017a) and expectation-based algorithms, it is clear that the distributional perspective results in different behaviour in the deep learning setting, but less clear whether this holds in simpler cases such as tabular MDPs. Before approaching the question of why distributional RL behaves better than expected RL, one can ask a simpler question: when is distributional RL *different* from expected RL?

To answer this question, we propose the following thought experiment. Suppose we have a distributional RL algorithm A_D and a counterpart A_E which considers only the expected returns of state-action pairs, with initial value (distribution) functions Q_0 and Z_0 such that $Q_0(x, a) = \mathbb{E}[Z_0(x, a)]$ for all $x, a \in \mathcal{X} \times \mathcal{A}$. We will denote this equality of expectations as $Z_0 \overset{\mathbb{E}}{=} Q_0$. We suppose that these two algorithms then experience the same trajectory through an MDP and undergo their respective update procedures at each time step, producing a sequence of value and return distribution functions $Z_1, \ldots, Z_n$ and $Q_1, \ldots, Q_n$ respectively. We are interested in the following question: will these two sequences both agree on what the best policy is at each step of the trajectory? Specifically, do they still agree on the expected return of each state-action pair at every time step? This question can be expressed as the following conjecture on the equivalence of the value and return distribution functions at each timestep.

$$Z_0 \overset{\mathbb{E}}{=} Q_0 \implies Z_t \overset{\mathbb{E}}{=} Q_t \; \forall t$$

This setting is inspired by the notion of a coupling from the probability theory literature, and we will at times refer to this thought experiment as the coupling experiment.

2.1. Settings

The claims that follow will concern updates in the following setting: let $M = (\mathcal{X}, A, R, P, \gamma)$ be an MDP with P known, Z_0, Q_0 be functions mapping state-action pairs to random variables and scalars respectively and $\mathbb{E}(Z_0(x, a)) = Q_0(x, a) \; \forall (x, a) \in \mathcal{X} \times A$. We set Z_t and Q_t to be given by application of the distributional Bellman operator (Equation 3) and Bellman operator (Equation 2) respectively. We will call this the *operator setting*.

To produce the *trajectory setting*, we require that the distributional algorithm and the expected algorithm take the same trajectory (x_t, a_t, r_{t+1}) through M. The MDP dynamics are not assumed to be known. The *approximate trajectory setting* will refer to the trajectory setting in which sampled distributions are first projected onto some fixed support via the Cramér projection.

3. Theoretical Results

Table 1 provides a summary of the theoretical results presented in this paper.

3.1. Tabular environments

We first consider the problem in the setting of tabular MDPs, where the predicted return distribution for each state-action pair can be stored separately in memory. There are three classes of updates to be considered in this case. We will first consider updates with the Bellman operator (respectively the distributional Bellman operator). We then consider mixture updates with sampled transitions, where the current estimate is updated via a convex combination of the current estimate and a sample from the MDP. Finally, we consider the behaviour of distributional algorithms which update their estimates using a sampled gradient with respect to some loss function.

Proposition 1. *Assume we are in the operator setting. We fix some policy π and let (Z_t), (Q_t) be sequences obtained by application of the distributional and expectation-based Bellman operators respectively. That is*

$$Z_{t+1}(x, a) \overset{D}{\equiv} T^\pi Z_t(x, a)) \text{ and } Q_{t+1}(x, a) \equiv T^\pi Q_t(x, a)$$

Then we claim that $\mathbb{E}[Z_t(x, a)] = Q_t(x, a) \forall t \in \mathbb{N}, x \in \mathcal{X}, a \in A$.

See the appendix for the proof of Lemma 1 and those that follow. As it is generally not computationally possible to store arbitrary density functions, we consider the class of categorical distributions defined by some fixed, finite support $z = \{z_1, \ldots, z_K\}$, as one can use such distributions to approximate a target distribution. Lemma 2 reveals that, provided that our approximating support is sufficiently large (so that the projection operator doesn't clip the return distribution and alter its expected value), we don't see any difference in the predicted values of state-action pairs when we use approximate distributions.

Proposition 2. *Consider again the operator setting. Fix some support $\{z_1, \ldots, z_k\}$ such that for all policies π, the distribution of returns from π on M is bounded in $[z_1, z_k]$. Observe that the Cramér projection Π_C induces the operator $T^\pi_C = \Pi_C T^\pi$. Fix some Z_0, Q_0 as before, with Z_0 supported on the interval $[z_1, z_k]$ but now let*

$$Z_{t+1} \equiv T^\pi_C Z_t \;\text{ and } Q_{t+1} \equiv T^\pi Q_t$$

Then we claim that the sequence Z_t satisfies $\mathbb{E}[Z_t(x, a)] = Q_t(x, a)$.

This result justifies the use of categorical distributional RL in the tabular setting, and so we now consider the trajectory setting, where we no longer know the transition probability matrix and instead must sample trajectories through the MDP to update our predictions.

Table 1. Summary of main theoretical results

Value Function	Distribution Representation	Update Rule	Same?
Tabular	All	Bellman	✓
Tabular	All	Mixture	✓
Tabular	Approximate (CDF)	Cramér gradient	✓
Tabular	Approximate (PMF)	Cramér gradient	X
Linear	Approximate (CDF)	Semigradient TD(0) (Cramér gradient)	✓
Linear	Approximate	General gradient	X
Nonlinear	Approximate	General gradient	X

Proposition 3. *Consider the trajectory setting, and let the law of Z_t be denoted P_{Z_t} and the law of the sampled target P'_t, corresponding to the law of the random variable $Z' = r_{t+1} + \gamma Z_t(x_{t+1}, a_{t+1})$. Suppose that at timestep t, Z_t is updated according to the rule*

$$P_{Z_{t+1}}(x_t, a_t)(z_i) \equiv (1-\alpha)P_{Z_t}(x_t, a_t)(z_i) + \alpha(P'_t)(z_i)$$

as defined in Rowland et al. (2018). Let Q_t be defined as

$$Q_{t+1}(x_t, a_t) \equiv (1-\alpha)Q_t(x_t, a_t) + \alpha(r_{t+1} + \gamma Q_t(x_{t+1}, a_{t+1}))$$

Then $\mathbb{E}[Z_{t+1}(x, a)] = Q_{t+1}(x, a)$.

Unlike in expected RL, where one predicts scalars and so has an obvious way of measuring the distance between prediction and target, there exist many different metrics on probability distributions. This then motivates the use of gradient updates even in the tabular setting, where the predicted distribution is updated to move in the direction of steepest descent in terms of its distance from the target according to some metric.

Proposition 4. *We place ourselves in the approximate trajectory setting, and let Z_t be a sequence of return distribution functions which have finite and equally spaced support of distance c. We let F_t be the sequence of CDFs corresponding to each Z_t at time t. Let Q_t be the sequence defined by*

$$\begin{aligned}Q_{t+1}(x_t, a_t) = (1-\alpha)Q_t(x_t, a_t) +& \\ &\alpha(r_{t+1} + \gamma Q_t(x_{t+1}, a_{t+1}))\end{aligned}$$

and

$$\begin{aligned}F_{t+1}(x_t, a_t) =& F_t(x_t, a_t) + \\ &\alpha' \nabla_F \ell_2^2(Z_t(x_t, a_t), r_{t+1} + \gamma Z_t(x_{t+1}, a_{t+1}))\end{aligned}$$

Then if $\alpha' = \frac{\alpha}{2c}$, we have that for all t, $Q_t(x, a) = \mathbb{E}[Z_t(x, a)]$.

Lemma 4 is significant as it shows that it is possible to perform gradient descent in the distributional setting and obtain the same behaviour as in the expected setting.

We emphasize that the settings in which this equivalence holds for gradient descent procedures are limited. Updating the distribution using the gradient of the Cramér metric with respect to the PMF, for example, does not result in the updated distribution being equal in expectation to the value that expected RL would predict. Further, the results of the previous proposition no longer hold if the atoms of a distribution are not equally spaced, even when we update according to the gradient of the Cramér metric with respect to the CDF of the distribution.

We provide an illustrative example for the first claim. Suppose we have a support $\mathbf{z} = (0, 1, 2)$ and two CDFs: $F' = (\frac{1}{2}, \frac{1}{2}, 1)$ and $F = (\frac{1}{3}, \frac{2}{3}, 1)$. Taking the gradient of the Cramér metric between the two distributions with respect to the PMF of the first gives $\nabla_p \ell_2^2(F, F') = (0, -0.083, 0)$. Now, when we consider $P + \alpha\nabla\ell_2^2(P, P') = (\frac{1}{2}, -0.083\alpha, \frac{1}{2})$ we can immediately observe that this is not a probability distribution. Further, it has expected value $1 - 0.083\alpha$. The expectations of P and P' are both 1, so a Cramér gradient update w.r.t. the CDFs would give a new distribution with expectation 1 as well.

3.2. Linear Function Approximators

In the linear approximation setting, we assume that it is no longer feasible to represent the MDP as a collection of state-action pairs in memory. Instead, we get that each state x corresponds to a feature vector ϕ_x (generally of dimension less than the number of states), and we wish to find a linear function given by a weight vector θ such that $\theta^T \phi_x \approx v(\phi_x)$, where v is the true value function on the feature vectors. In the distributional setting where the approximating distribution has finite support $z = \{z_1, \ldots, z_t\}$, θ becomes a matrix W, such that $W\phi_x[i] = p(z_i)$ or $F(z_i)$, i.e. each row of the matrix W corresponds to each support point of the distribution. We note that in this setting it is impossible for W to be such that Wv is a proper PMF or CDF for all vectors v, as if $W\phi_x$ is a proper CDF then $W(-\phi_x)$ will be negative for each z_i, and so will not correspond to a proper CDF, instead producing a signed measure where some points in the support are assigned negative mass. Importantly, we can still define an analogue of the Cramér metric in the signed distribution case and perform updates in a similar manner as with probability distributions, as shown by (Anonymous, 2018). In spite of this drawback, the linear setting does show that it is possible to have distributional algorithms agree with expected algorithms in the function approximation setting and so we include its analysis.

We will consider the case where $W\phi[i] = F(z_i)$.

Proposition 5. *Consider the trajectory setting, with W_t the weight matrix which computes the CDF of the predicted return distribution (of categorical support $\mathbf{z} = z_1, \ldots, z_k$ with $z_i - z_{i-1} = 1$) function at time t and θ_t the vector computing Q_t. Define the sequence*

$$W_{t+1} \equiv W_t + \alpha \nabla \ell_2^2(W\phi_{x_t}, F_t)$$

where F_t has the same support as $W\phi_{x_t}$, i.e. Z_t is updated via the ℓ_2 semigradient TD(0) updates. Let

$$\theta_{t+1} = \theta_t + \alpha \nabla(\theta^T \phi_{x_t} - v_t)^2$$

where $\mathbb{E}[P_t] = v_t$. Then $\mathbb{E}[Z_t(\phi_x)] = Q_t(\phi_x)$ for all t and x.

3.3. Non-linear Function Approximators

We define a non-linear return distribution function as a function parameterized by weights w which takes as input a feature vector x and outputs a categorical distribution $p = (p(z_1), \ldots, p(z_t))$. This function, however, need not be linear in either the weights or the features. For example, if $W = a$ and our feature vectors are in $\mathbb{R}^1$, then the function $f(a;x) = (\frac{1}{1+e^{-ax}}, 1 - \frac{1}{1+e^{-ax}})$ would be a non-linear return distribution function.

Proposition 6. *Let $\psi_W : \mathbb{R}^n \to \mathbb{R}^k$, $\psi_\theta : \mathbb{R}^n \to \mathbb{R}$ be non-linear function approximators for $Z(x,a)$ and $Q(x,a)$ parameterized by W and θ respectively. Suppose that $\psi_\theta(x,a) = \mathbb{E}[\psi_W(x,a)] \forall x, a \in \mathcal{X} \times \mathcal{A}$. Suppose that we observe action a from state x and sampled return distribution Z_t. Then if*

$$W' = W + \alpha_t \nabla_W(\ell_2^2(\psi_W(x,a), Z_t))$$

and

$$\theta' = \theta + \alpha_t(\nabla_\theta(\psi_\theta(x,a) - R_t)^2)$$

then there exist function approximators and update data such that $\mathbb{E}[\psi_{W'}(x,a)] \neq \psi_{\theta'}(x,a)$.

Proof. See counterexample in appendix. □

4. Experimental Results

Though theoretical results indicate that performing gradient updates with respect to the distribution's CDF should produce different predicted distributions from gradient updates with respect to its PMF, it is not immediately clear how these differences affect performance. To explore this, we considered a 12x12 gridworld environment and ran two distributional versions of Q-learning, one which performed gradient updates with respect to the CDF and one which performed updates with respect to the PMF of the predicted distribution, alongside traditional Q-learning. We found that, as predicted by Lemma 4, when given the same random seed the CDF update method had identical performance to traditional Q-learning as shown by Figure 2. The PMF update method, though not significantly worse in performance, did exhibit *different* performance, performing better on some random seeds and worse on others than expected Q-learning.

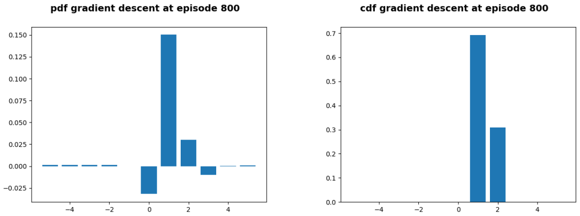

Figure 1. Distributions learned by PMF and CDF update methods

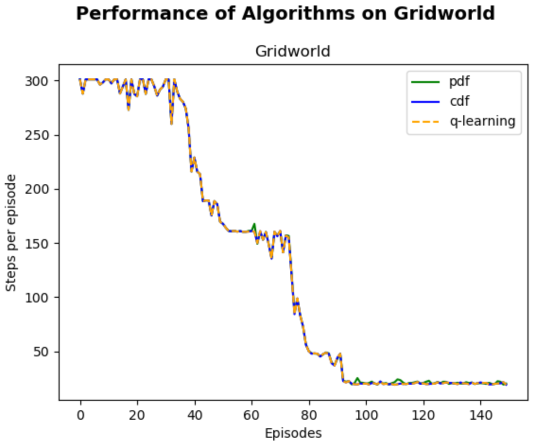

Figure 2. Performance on a large-room gridworld

To better see how the PMF updates were changing the predicted distribution, we also considered a simple 3-state chain MDP with goal states to the left and right of the start state, for which the agent received a reward of 1 for taking the left (respectively right) action from that state and 0 otherwise, and examined the distribution predicted for the left action on the leftmost state. We observed that the PMF update method immediately introduced negative probabilities on atoms, and continued to do so for the 800 episodes in which we ran the algorithm. Visualizations of these can be seen in Figure 1.

5. Discussion and Future Work

The work covered in this report examined the conditions under which distributional reinforcement learning algorithms will exhibit behaviour that cannot be replicated by an expected algorithm, and the conditions under which they can. In general, the more complex the setting the less likely it is that distributional and expected RL algorithms will behave in the same way, although we note the importance of both the choice of metric on distributions that we wish to minimize and also the parameterization of the distribution in determining whether such an equivalence exists.

Although focused primarily on the Cramér metric and on categorical distributions in our analysis, it should be noted that minimizing the KL divergence between prediction and target of a softmax distribution Bellemare et al. (2017a) still produced state-of-the-art results, so the examples studied in depth are not the only empirically viable ones. A comparison of the performance and theoretical properties of algorithms which minimize different divergences and use different representations of distributions may shed some light on this matter.

Finally, our analysis focused only on when the distributional perspective results in different behaviour, and didn't shed light on how this divergent behaviour could result in improved performance over traditional reinforcement learning algorithms. Further investigation of not just *where* expected and distributional algorithms differ, but *how* they differ are therefore of great interest.

References

Anonymous. Distributional reinforcement learning with linear function approximation. 2018.

Bellemare, Marc G., Dabney, Will, and Munos, Rémi. A distributional perspective on reinforcement learning. In Precup, Doina and Teh, Yee Whye (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 449–458, International Convention Centre, Sydney, Australia, 06–11 Aug 2017a. PMLR. URL http://proceedings.mlr.press/v70/bellemare17a.html.

Bellemare, Marc G., Danihelka, Ivo, Dabney, Will, Mohamed, Shakir, Lakshminarayanan, Balaji, Hoyer, Stephan, and Munos, Rémi. The cramer distance as a solution to biased wasserstein gradients. *CoRR*, abs/1705.10743, 2017b. URL http://arxiv.org/abs/1705.10743.

Morimura, Tetsuro, Hachiya Hirotaka Sugiyama Masashi Tanaka Toshiyuki and Kashima, Hisashi. Parametric return density estimation for reinforcement learning. *Uncertainty in Artificial Intelligence*, 2010.

Rowland, Mark, Bellemare, Marc, Dabney, Will, Munos, Remi, and Teh, Yee Whye. An analysis of categorical distributional reinforcement learning. In Storkey, Amos and Perez-Cruz, Fernando (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 29–37, Playa Blanca, Lanzarote, Canary Islands, 09–11 Apr 2018. PMLR. URL http://proceedings.mlr.press/v84/rowland18a.html.

Székely, GJ. E-statistics: The energy of statistical samples. *Bowling Green State University, Department of Mathematics and Statistics Technical Report*, 3(05):1–18, 2003.

A. Proofs of Main Results

Proposition 1. *Assume we are in the operator setting. We fix some policy π and let (Z_t), (Q_t) be sequences obtained by application of the distributional and expectation-based Bellman operators respectively. That is*

$$Z_{t+1}(x,a) \overset{D}{\equiv} T^\pi Z_t(x,a)) \text{ and } Q_{t+1}(x,a) \equiv T^\pi Q_t(x,a)$$

Then we claim that $\mathbb{E}[Z_t(x,a)] = Q_t(x,a) \forall t \in \mathbb{N}, x \in \mathcal{X}, a \in A$.

Proof. By induction. By construction this is the case for Z_0, Q_0. Suppose it holds for timestep t. Then for timestep $t+1$, we have:

$$\begin{aligned}
\mathbb{E}[Z_{t+1}(x,a)] &= \mathbb{E}[R(x,a) + \gamma \sum_{x',a'} P(x'|x,a)\pi(a'|x')Z_t(x',a')] \\
&= \mathbb{E}[R(x,a)] + \gamma \sum_{x',a'} P(x'|x,a)\pi(a'|x')\mathbb{E}[Z_t(x',a')] \\
&= \mathbb{E}[R(x,a) + \gamma \sum_{x',a'} P(x'|x,a)\pi(a'|x')Q_t(x,a)] \\
&= Q_{t+1}(x,a) \qquad \square
\end{aligned}$$

Proposition 2. *Consider again the operator setting. Fix some support $\{z_1, \ldots, z_k\}$ such that for all policies π, the distribution of returns from π on M is bounded in $[z_1, z_k]$. Observe that the Cramér projection Π_C induces the operator $T_C^\pi = \Pi_C T^\pi$. Fix some Z_0, Q_0 as before, with Z_0 supported on the interval $[z_1, z_k]$ but now let*

$$Z_{t+1} \equiv T_C^\pi Z_t \text{ and } Q_{t+1} \equiv T^\pi Q_t$$

Then we claim that the sequence Z_t satisfies $\mathbb{E}[Z_t(x,a)] = Q_t(x,a)$.

Proof. The fact that the Cramér projection preserves expectation so long as it does not truncate the projected distribution was shown by (Rowland et al., 2018). Then since $\mathbb{E}[T^\pi Z_t(x,a)] = T^\pi Q_t(x,a)$ we have that $\mathbb{E}[T^\pi_C Z_t(x,a)] = \mathbb{E}[T^\pi Z_t(x,a)] = T^\pi Q_t(x,a)$ which proves the proposition. □

Proof. From Lemma 1 we know that $\mathbb{E}T^\pi Z_t = T^\pi Q_t$. We also note that $T^\pi Z_t$ is a finite linear combination of dirac distributions, so T^π_C is well-defined. Finally, we note as shown in Rowland et al. (2018) $\mathbb{E}[\Pi_C Z] = \mathbb{E}[Z]$ whenever the support of Z lies within the support we are projecting the distribution onto. So we leverage the results of the previous lemmas to obtain:

$$\begin{aligned}\mathbb{E}[Z_{t+1}] &= \mathbb{E}[\Pi_c T^\pi Z_t] \\ &= \mathbb{E}[T^\pi Z_t] = Q_{t+1}\end{aligned}$$ □

Proposition 3. *Consider the trajectory setting, and let the law of Z_t be denoted P_{Z_t} and the law of the sampled target P'_t, corresponding to the law of the random variable $Z' = r_{t+1} + \gamma Z_t(x_{t+1}, a_{t+1})$. Suppose that at timestep t, Z_t is updated according to the rule*

$$P_{Z_{t+1}}(x_t, a_t)(z_i) \equiv (1-\alpha)P_{Z_t}(x_t,a_t)(z_i) + \alpha(P'_t)(z_i)$$

as defined in Rowland et al. (2018). Let Q_t be defined as

$$Q_{t+1}(x_t,a_t) \equiv (1-\alpha)Q_t(x_t,a_t) + \alpha(r_{t+1} + \gamma Q_t(x_{t+1}, a_{t+1}))$$

Then $\mathbb{E}[Z_{t+1}(x,a)] = Q_{t+1}(x,a)$.

Proof. We proceed again by induction. We let $Z_t(x,a)$ be a random variable with law $\eta_t(x,a)$ for each x, a. By assumption, $\mathbb{E}[Z_0(x,a)] = Q_0(x,a)$ for all x, a. Now, for the induction step:

$$\begin{aligned}\mathbb{E}(Z_{t+1}(x_t,a_t)) &= \sum_{i=1}^k P_{Z_{t+1}}(z_i) \\ &= \sum_{i=1}^k (1-\alpha)P_{Z_t}(z_i) + \alpha P'_t(z_i) \\ &= (1-\alpha)\sum_{i=1}^k P_{Z_t}(z_i) + \alpha\sum_{i=1}^k P'_t(z_i) \\ &= (1-\alpha)\mathbb{E}[Z_t(x_t,a_t)] + \alpha\mathbb{E}[r_{t+1} + \gamma Z_t(x_{t+1},a_{t+1})] \\ &= (1-\alpha)Q_t(x_t,a_t) + \alpha[r_t + \gamma Q_t(x_{t+1},a_{t+1})] \\ &= Q_{t+1}(x_t,a_t)\end{aligned}$$ □

Proposition 4. *We place ourselves in the approximate trajectory setting, and let Z_t be a sequence of return distribution functions which have finite and equally spaced support of distance c. We let F_t be the sequence of CDFs corresponding to each Z_t at time t. Let Q_t be the sequence defined by*

$$Q_{t+1}(x_t,a_t) = (1-\alpha)Q_t(x_t,a_t) + \alpha(r_{t+1} + \gamma Q_t(x_{t+1},a_{t+1}))$$

and

$$\begin{aligned}F_{t+1}(x_t,a_t) &= F_t(x_t,a_t) + \\ &\alpha'\nabla_F \ell_2^2(Z_t(x_t,a_t), r_{t+1} + \gamma Z_t(x_{t+1},a_{t+1}))\end{aligned}$$

Then if $\alpha' = \frac{\alpha}{2c}$, we have that for all t, $Q_t(x,a) = \mathbb{E}[Z_t(x,a)]$.

Proof. We first note that $\nabla_{F_p}\ell_2^2(P,Q) = 2c(F_q - F_p)$. Thus the gradient update in this case is simply a mixture update, and the result follows from Proposition 3. □

Proposition 5. *Consider the trajectory setting, with W_t the weight matrix which computes the CDF of the predicted return distribution (of categorical support $\mathbf{z} = z_1, \ldots, z_k$ with $z_i - z_{i-1} = 1$) function at time t and θ_t the vector computing Q_t. Define the sequence*

$$W_{t+1} \equiv W_t + \alpha\nabla\ell_2^2(W\phi_{x_t}, F_t)$$

where F_t has the same support as $W\phi_{x_t}$, i.e. Z_t is updated via the ℓ_2 semigradient TD(0) updates. Let

$$\theta_{t+1} = \theta_t + \alpha\nabla(\theta^T\phi_{x_t} - v_t)^2$$

where $\mathbb{E}[P_t] = v_t$. Then $\mathbb{E}[Z_t(\phi_x)] = Q_t(\phi_x)$ for all t and x.

Proof.

$$\begin{aligned}\mathbb{E}\nabla_W\ell_2^2(W\phi(x),P)(\phi(x')) &= z^T C^{-1}((W\phi(x) - F)\phi(x)^T)\phi(x') \\ &= (\phi(x)^T\phi(x'))z^T C^{-1}(W\phi(x) - F)\end{aligned}$$

Observing that by assumption $z^T C^{-1} F(x) = v(x)$ and $z^T C^{-1} W\phi(x) = v_\theta(x) = \theta^T\phi(x)$ we get:

$$\begin{aligned}&= \phi(x)^T\phi(x')(\theta^T\phi(x) - v) \\ &= (\phi(x)^T(\theta^T\phi(x) - v))\phi(x') \\ &= \nabla_\theta(v_\theta(x) - v)^2\phi(x')\end{aligned}$$ □

Proposition 6. *Let $\psi_W : \mathbb{R}^n \to \mathbb{R}^k$, $\psi_\theta : \mathbb{R}^n \to \mathbb{R}$ be non-linear function approximators for $Z(x,a)$ and $Q(x,a)$ parameterized by W and θ respectively. Suppose that $\psi_\theta(x,a) = \mathbb{E}[\psi_W(x,a)] \forall x, a \in \mathcal{X} \times \mathcal{A}$. Suppose that we observe action a from state x and sampled return distribution Z_t. Then if*

$$W' = W + \alpha_t\nabla_W(\ell_2^2(\psi_W(x,a), Z_t))$$

and

$$\theta' = \theta + \alpha_t(\nabla_\theta(\psi_\theta(x,a) - R_t)^2)$$

then there exist function approximators and update data such that $\mathbb{E}[\psi_{W'}(x,a)] \neq \psi_{\theta'}(x,a)$.

We prove the proposition by showing that a gradient update with respect to a distribution with the same expected value as the initial estimate can result in the updated distribution having a different expected value than the first one. Then since in the expectation-based case our gradient is zero when the expected values are equal, we would have that the updated expectation-based approximator would have the same value as that of the old distribution, and so won't be equal to the new predicted distribution.

Concretely, let $z = (-1, 0, 1)$. Set $\psi_W(x) = [\sigma(w_1 x), \sigma(w_2 x), 1]$ corresponding to $F(-1), F(0), F(1)$, with $W_0 = [-\ln(2), -\ln(1/2)/2]$. Suppose $\psi_\theta(x) = z^T C^{-1} \psi_W(x)$ for all x. Then suppose we sample a state with feature vector $(1, 2)$ target distribution $F = [0, 1, 1]$. Then θ remains the same but the expected value of ψ changes when we perform a gradient update. We first calculate the gradient of ψ_W:

$$\begin{aligned}\frac{\partial}{\partial W_1}(1/2)\ell_2^2(\psi_W(x), F) &= (F(-1) - F_\psi(-1))\frac{\partial}{\partial W_1}(F_\psi(-1)) \\ &= (0 - \frac{1}{3})\sigma(\ln(2))(1 - \sigma(\ln(2)))(-x_1) \\ \frac{\partial}{\partial W_2}(1/2)\ell_2^2(\psi_W(x), F) &= (F(0) - F_\psi(0))\frac{\partial}{\partial W_1}(F_\psi(0)) \\ &= (1 - \frac{2}{3})\sigma(-\ln(2))(1 - \sigma(-\ln(2)))(-x_2)\end{aligned}$$

Plugging in the relevant x values gives the precise directional derivatives:

$$\begin{aligned}\partial\ell_2^2/\partial w_1 &= (0 - 0.33)(1 - 0.33)(0.33)(-1) = \frac{2}{27} \\ \partial\ell_2^2/\partial w_2 &= (1 - 0.67)(1 - 0.67)(0.67)(-2) = \frac{-4}{27}\end{aligned}$$

The negation of these gives a descent direction. Let $\alpha = 1, W' = W - \alpha\nabla_\theta \ell_2^2(F_\psi, F)$. Then we claim that the expected value of the CDF $\psi'(1,2) \equiv \psi_{W'}(1,2)$ denoted $F_{\psi'(1,2)}$ is different from the expectation of $F_{\psi(1,2)}$. To see this, consider:

$$\begin{aligned}\mathbb{E}[F_{\psi'(1,2)}] &= (-1)p_{\psi'(1,2)}(z_1) + (1)p_{\psi'(1,2)}(z_3) \\ &= -F_{\psi'}(z_1) + (1 - F_{\psi'}(z_2)) \\ &= \frac{-1}{1 + e^{(\ln(2) + 2/27)(1)}} + 1 - \frac{1}{1 + e^{(\ln(\frac{1}{2})/2 - 4/27)(2)}} \\ &\approx -0.05 \neq 0 = \mathbb{E}[F_\psi]\end{aligned}$$

Thus, if we update our weights w.r.t. $\ell_2^2(F(1,2), F_\psi(1,2))$ we get a different expected value from the updated distribution at the same point $(1,2)$ than if we update our weights w.r.t. $|\mathbb{E}F - \mathbb{E}F_\psi|$. However, F_ψ and F have the same expected values, so the weights wouldn't be updated in the expectation-based algorithm, and so the expected value of the state wouldn't change.