Distributional Reinforcement Learning via Moment Matching

Thanh Nguyen-Tang, Sunil Gupta, Svetha Venkatesh

Applied Artificial Intelligence Institute (A^{2}I^2), Deakin University, Australia
{thanhnt, sunil.gupta, svetha.venkatesh}@deakin.edu.au

Abstract

We consider the problem of learning a set of probability distributions from the empirical Bellman dynamics in distributional reinforcement learning (RL), a class of state-of-the-art methods that estimate the distribution, as opposed to only the expectation, of the total return. We formulate a method that learns a finite set of statistics from each return distribution via neural networks, as in the distributional RL literature. Existing distributional RL methods however constrain the learned statistics to *predefined* functional forms of the return distribution which is both restrictive in representation and difficult in maintaining the predefined statistics. Instead, we learn *unrestricted* statistics, i.e., deterministic (pseudo-)samples, of the return distribution by leveraging a technique from hypothesis testing known as maximum mean discrepancy (MMD), which leads to a simpler objective amenable to backpropagation. Our method can be interpreted as implicitly matching all orders of moments between a return distribution and its Bellman target. We establish sufficient conditions for the contraction of the distributional Bellman operator and provide finite-sample analysis for the deterministic samples in distribution approximation. Experiments on the suite of Atari games show that our method outperforms the distributional RL baselines and sets a new record in the Atari games for non-distributed agents.

Introduction

A fundamental aspect in reinforcement learning (RL) is the value of an action in a state which is formulated as the expected value of the *return*, i.e., the expected value of the discounted sum of rewards when the agent follows a policy starting in that state and executes that action (Sutton, Barto et al. 1998). Learning this expected action-value via Bellman's equation (Bellman 1957) is central to value-based RL such as temporal-difference (TD) learning (Sutton 1988), SARSA (Rummery and Niranjan 1994), and Q-learning (Watkins and Dayan 1992). Recently, however, approaches known as *distributional* RL that aim at learning the distribution of the return have shown to be highly effective in practice (Morimura et al. 2010b,a; Bellemare, Dabney, and Munos 2017; Dabney et al. 2018b,a; Yang et al. 2019).

Despite many algorithmic variants with impressive practical performance (Bellemare, Dabney, and Munos 2017;

Dabney et al. 2018b,a; Yang et al. 2019), they all share the same characteristic that they explicitly learn a set of statistics of *predefined* functional forms to approximate a return distribution. Using predefined statistics can limit the learning due to the statistic constraints it imposes and the difficulty to maintain such predefined statistics. In this paper, we propose to address these limitations by instead learning a set of *unrestricted* statistics, i.e., deterministic (pseudo-)samples, of a return distribution that can be evolved into any functional form. We observe that the deterministic samples can be deterministically learned to simulate a return distribution by utilizing an idea from statistical hypothesis testing known as maximum mean discrepancy (MMD). This novel perspective requires a careful design of algorithm and a further understanding of distributional RL associated with MMD.

Leveraging this perspective, we are able to provide a novel algorithm to eschew the predefined statistic limitations in distributional RL and give theoretical understanding of distributional RL within this perspective. Our approach is also conceptually amenable for natural extension along the lines of recent modelling improvements to distributional RL brought by Implicit Quantile Networks (IQN) (Dabney et al. 2018a), and Fully parameterized Quantile Function (FQF) (Yang et al. 2019). Specifically, our key contributions are

1. We provide a novel approach to distributional RL using deterministic samples via MMD that addresses the limitations in the existing distributional RL;

2. We provide theoretical understanding of distributional RL within our framework, specifically the contraction property of the distributional Bellman operator and the non-asymptotic convergence of the approximate distribution from deterministic samples;

3. We demonstrate the practical effectiveness of our framework in both tabular RL and large-scale experiments where our method outperforms the standard distributional RL methods and even establishes a new record in the Atari games for non-distributed agents.

Outline. After carefully reviewing relevant background and related works, and discussing their limitations, we present our novel algorithmic approach to address these issues followed by theoretical analysis. We then present the experiments to confirm the effectiveness of our approach empirically and conclude our work.

Background and Related Works

Expected RL

In a standard RL setting, an agent interacts with an environment via a Markov Decision Process $(\mathcal{S}, \mathcal{A}, R, P, \gamma)$ (Puterman 2014) where $\mathcal{S}$ and $\mathcal{A}$ denote state and action spaces, resp., $\mathcal{R}$ the reward measure, $P(\cdot|s,a)$ the transition kernel measure, and $\gamma \in [0,1)$ a discount factor. A policy $\pi(\cdot|s)$ maps a state to a distribution over the action space.

Given a policy π, the discounted sum of future rewards following policy π is the random variable

$$Z^\pi(s,a) = \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t), \tag{1}$$

where $s_0 = s, a_0 = a, s_t \sim P(\cdot|s_{t-1}, a_{t-1}), a_t \sim \pi(\cdot|s_t)$, and $R(s_t, a_t) \sim \mathcal{R}(\cdot|s_t, a_t)$. The goal in expected RL is to find an optimal policy π^* that maximizes the action-value function $Q^\pi(s,a) := \mathbb{E}[Z^\pi(s,a)]$. A common approach is to find the unique fixed point $Q^* = Q^{\pi^*}$ of the Bellman optimality operator (Bellman 1957) $T : \mathbb{R}^{\mathcal{S} \times \mathcal{A}} \to \mathbb{R}^{\mathcal{S} \times \mathcal{A}}$ defined by

$$TQ(s,a) := \mathbb{E}[R(s,a)] + \gamma \mathbb{E}_P[\max_{a'} Q(s',a')], \forall(s,a).$$

A standard approach to this end is Q-learning (Watkins and Dayan 1992) which maintains an estimate Q_θ of the optimal action-value function Q^* and iteratively improves the estimation via the Bellman backup

$$Q_\theta(s,a) \leftarrow \mathbb{E}[R(s,a)] + \gamma \mathbb{E}_P[\max_{a'} Q_\theta(s',a')].$$

Deep Q-Network (DQN) (Mnih et al. 2015) achieves human-level performance on the Atari benchmark by leveraging a convolutional neural network to represent Q_θ while using a replay buffer and a target network to update Q_θ.

Additional Notations

Let $\mathcal{X} \subseteq \mathbb{R}^d$ be an open set. Let $\mathcal{P}(\mathcal{X})$ be the set of Borel probability measures on $\mathcal{X}$. Let $\mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$ be the Cartesian product of $\mathcal{P}(\mathcal{X})$ indexed by $\mathcal{S} \times \mathcal{A}$. For any $\alpha \geq 0$, let $\mathcal{P}_\alpha(\mathcal{X}) := \{p \in \mathcal{P}(\mathcal{X}) : \int_{\mathcal{X}} \|x\|^\alpha p(dx) < \infty\}$. When $d = 1$, let $m_n(p) := \int_{\mathcal{X}} x^n p(dx)$ be the n-th order moment of a distribution $p \in \mathcal{P}(\mathcal{X})$, and let

$$\mathcal{P}_*(\mathcal{X}) = \left\{ p \in \mathcal{P}(\mathcal{X}) : \limsup_{n \to \infty} \frac{|m_n(p)|^{1/n}}{n} = 0 \right\}.$$

Note that if $\mathcal{X}$ is a bounded domain in $\mathbb{R}$, then $\mathcal{P}_*(\mathcal{X}) = \mathcal{P}(\mathcal{X})$. Denote by δ_z the Dirac measure, i.e., the point mass, at z. Denote by Δ_n the n-dimensional simplex.

Distributional RL

Instead of estimating only the expectation Q^π of Z^π, distributional RL methods (Bellemare, Dabney, and Munos 2017; Dabney et al. 2018b,a; Rowland et al. 2018; Yang et al. 2019) explicitly estimate the return distribution $\mu^\pi = \mathrm{law}(Z^\pi)$ as an auxiliary task. Empirically, this auxiliary

task has been shown to significantly improve the performance in the Atari benchmark. Theoretically, in the policy evaluation setting, the distributional version of the Bellman operator is a contraction in the p-Wasserstein metric (Bellemare, Dabney, and Munos 2017) and Crámer distance (Rowland et al. 2018) (but not in total variation distance (Chung and Sobel 1987), Kullback-Leibler divergence and Komogorov-Smirnov distance (Bellemare, Dabney, and Munos 2017)). The contraction implies the uniqueness of the fixed point of the distributional Bellman operator. In control settings with tabular function approximations, distributional RL has a well-behaved asymptotic convergence in Crámer distance when the return distributions are parameterized by categorical distributions (Rowland et al. 2018). Bellemare et al. (2019) establish the asymptotic convergence of distributional RL in policy evaluation in linear function approximations. Lyle, Bellemare, and Castro (2019) examine behavioural differences between distributional RL and expected RL, aligning the success of the former with non-linear function approximations.

Categorical Distributional RL (CDRL) CDRL (Bellemare, Dabney, and Munos 2017) approximates a distribution η by a categorical distribution $\hat{\eta} = \sum_{i=1}^N \theta_i \delta_{z_i}$ where $z_1 \leq z_2 \leq ... \leq z_N$ is a set of fixed supports and $\{\theta_i\}_{i=1}^N$ are learnable probabilities. The learnable probabilities $\{\theta_i\}_{i=1}^N$ are found in such way that $\hat{\eta}$ is a projection of η onto $\{\sum_{i=1}^N p_i \delta_{z_i} : \{p_i\}_{i=1}^N \in \Delta_N\}$ w.r.t. the Crámer distance (Rowland et al. 2018). In practice, C51 (Bellemare, Dabney, and Munos 2017), an instance of CDRL with $N = 51$, has shown to perform favorably in Atari games.

Quantile Regression Distributional RL (QRDRL) QRDRL (Dabney et al. 2018b) approximates a distribution η by a mixture of Diracs $\hat{\eta} = \frac{1}{N} \sum_{i=1}^N \delta_{\theta_i}$ where $\{\theta_i\}_{i=1}^N$ are learnable in such a way that $\hat{\eta}$ is a projection of η on $\{\frac{1}{N} \sum_{i=1}^N \delta_{z_i} : \{z_i\}_{i=1}^N \in \mathbb{R}^N\}$ w.r.t. to the 1-Wasserstein distance. Consequently, $\theta_i = F_\eta^{-1}(\frac{2i-1}{2N})$ where F_η^{-1} is the inverse cumulative distribution function of η. Since the quantile values $\{F_\eta^{-1}(\frac{2i-1}{2N})\}$ at the fixed quantiles $\{\frac{2i-1}{2N}\}$ is a minimizer of an asymmetric quantile loss from quantile regression literature (thus the name QRDRL) and the quantile loss is compatible with stochastic gradient descent (SGD), the quantile loss is used for QRDRL in practice. QR-DQN-1 (Dabney et al. 2018b), an instance of QRDRL with Huber loss, performs favorably empirically in Atari games.

Other Distributional Methods Some recent distributional RL methods have made modelling improvements to QRDRL. Two typical improvements are from Implicit Quantile Networks (IQN) (Dabney et al. 2018a), and Fully parameterized Quantile Function (FQF) (Yang et al. 2019). IQN uses implicit models to represent the quantile values $\{\theta_i\}$ in QRDRL, i.e., instead of being represented by fixed network outputs, $\{\theta_i\}$ are the outputs of a differentiable function (e.g., neural networks) on the samples from a base sampling distribution (e.g., uniform). FQF further improves IQN by optimizing the locations of the base samples for IQN, instead of using random base samples as in IQN, i.e.,

both quantiles and quantile values are learnable in FQF.

Predefined Statistic Principle

Formally, a statistic is any functional $\zeta : \mathcal{P}(\mathcal{X}) \to \mathbb{R}$ that maps a distribution $p \in \mathcal{P}(\mathcal{X})$ to a scalar $\zeta(p)$, e.g., the expectation $\zeta(p) = \int_{\mathcal{X}} x p(dx)$ is a common statistic in RL. Here, we formally refer to a *predefined statistic* as the one whose functional form is specified before the statistic is learned. In contrast, an *unrestricted statistic* does not subscribe to any specific functional form (e.g., the median of a distribution η is a predefined statistic as its functional form is predefined via $F_\eta^{-1}(\frac{1}{2})$ while any empirical sample $z \sim \eta$ can be considered an unrestricted statistic of η).

Though CDRL and QRDRL are two different variants of distributional RL methodology, they share a unifying characteristic that they both explicitly learn a finite set of predefined statistics, i.e., statistics of *predefined* functional forms (Rowland et al. 2019). We refer to this as predefined statistic principle. This is clear for QRDRL as the statistics to be learned about a distribution η are $\{\zeta_1, ..., \zeta_N\}$ where

$$\zeta_i(\eta) := F_\eta^{-1}(\frac{2i-1}{N}), \forall i \in \{1, ..., N\}.$$

It is a bit more subtle for CDRL. It can be shown in (Rowland et al. 2019) that CDRL is equivalent to learning the statistics $\{\zeta_1, ..., \zeta_{N-1}\}$ where

$$\zeta_i(\eta) := \mathbb{E}_{Z \sim \eta} \left[1_{\{Z < z_i\}} + 1_{\{z_i \le Z < z_{i+1}\}} \frac{z_{i+1} - Z}{z_{i+1} - z_i} \right], \forall i.$$

Learning predefined statistics as in CDRL and QRDRL however can suffer two limitations in (i) statistic representation and (ii) difficulty in maintaining the predefined statistics. Regarding (i), given the same fixed budget of N statistics to approximate a return distribution η, CDRL restrictively associates the statistic budget to N fixed supports $\{z_i\}_{i=1}^N$ while QRDRL constrains the budget to N quantile values at specific quantiles. Instead, the statistic budget should be freely learned into any form as long as it could simulate the target distribution η sensibly. Regarding (ii), the fixed supports in CDRL require a highly involved projection step to be able to use KL divergence as the Bellman backup changes the distribution supports; QRDRL requires that the statistics must satisfy the constraints for valid quantile values at specific quantiles, e.g., the statistics to be learned are order statistics. In fact, QR-DQN (Dabney et al. 2018b), a typical instance of QRDRL, implicitly maintains the order statistics via an asymmetric quantile loss but still does not guarantee the monotonicity of the obtained quantile estimates. A further notice regarding (ii) recognized in (Rowland et al. 2019) is that since in practice we do not observe the environment dynamic but only samples of it, a naive update to learn the predefined statistics using such samples can collapse the approximate distribution due to the different natures of samples and statistics (in fact, (Rowland et al. 2019) proposes imputation strategies to overcome this problem). Instead, the statistics to be learned should be free of all such difficulties to reduce the learning burden.

One might say that IQN/FQF (discussed in the previous subsection) can help QRDRL overcome these limitations. While the modeling improvements in IQN/FQF are practically effective, IQN/FQF however still embrace the predefined statistic principle above as they built upon QR-DRL with an improved modelling capacity. In this work we propose an alternative approach to distributional RL that directly eschews the predefined statistic principle used in the prior distributional RL methods, i.e., the finite set of statistics in our approach can be evolved into any functional form and thus also reduces the need to maintain any statistic constraints. If we informally view improvements to CDRL/QRDRL into two dimensions: either modelling dimension or statistic dimension, IQN/FQF lie in the modelling dimension while our work belongs to the statistic dimension. We notice that this does not necessarily mean one approach is better than the other, but rather two orthogonal approaches where the modelling improvements in IQN/FQF can naturally apply to our work to further improve the modeling capacity. We leave these modeling extensions to the future work and focus the present work only on unrestricted statistics with the simplest modelling choice as possible.

Distributional RL via Moment Matching

Maximum Mean Discrepancy

Let $\mathcal{F}$ be a reproducing kernel Hilbert space (RKHS) associated with a continuous kernel $k(\cdot, \cdot)$ on $\mathcal{X}$. Consider $p, q \in \mathcal{P}(\mathcal{X})$, and let Z and W be two random variables with distributions p and q, respectively. The maximum mean discrepancy (MMD) (Gretton et al. 2012) between p and q is defined as

$$\text{MMD}(p, q; \mathcal{F}) := \sup_{f \in \mathcal{F}: \|f\|_{\mathcal{F}} \le 1} (\mathbb{E}[f(Z)] - \mathbb{E}[f(W)])$$

$$= \|\psi_p - \psi_q\|_{\mathcal{F}}$$

$$= \left(\mathbb{E}[k(Z, Z')] + \mathbb{E}[k(W, W')] - 2\mathbb{E}[k(Z, W)] \right)^{1/2}$$

where $\psi_p := \int_{\mathcal{X}} k(x, \cdot) p(dx)$ is the Bochner integral, i.e., the mean embedding of p into $\mathcal{F}$ (Smola et al. 2007), and Z' (resp. W') is a random variable with distribution p (resp. q) and is independent of Z (resp. W). In sequel, we interchangeably refer to MMD by $\text{MMD}(p, q; \mathcal{F}), \text{MMD}(p, q; k)$, or $\text{MMD}(p, q)$ if the context is clear.

Empirical Approximation Given empirical samples $\{z_i\}_{i=1}^N \sim p$ and $\{w_i\}_{i=1}^M \sim q$, MMD admits a simple empirical estimate as

$$\text{MMD}_b^2(\{z_i\}, \{w_i\}; k) = \frac{1}{N^2} \sum_{i,j} k(z_i, z_j)$$

$$+ \frac{1}{M^2} \sum_{i,j} k(w_i, w_j) - \frac{2}{NM} \sum_{i,j} k(z_i, w_j).$$

Though there is also a simple unbiased estimate of MMD, the *biased estimate* MMD_b has smaller variance in practice and thus is adopted in our work.

Problem Setting

Consider $d = 1$. For any policy π, let $\mu^\pi = \text{law}(Z^\pi)$ be the law (distribution) of the return r.v. Z^π as defined in Eq. (1).

Algorithm 1: Generic MMDRL update

Require: Number of particles N, kernel k, discount factor $\gamma \in [0, 1]$

Input: Sample transition (s, a, r, s')

1 **if** *Policy evaluation* **then**
2 $\quad\mid\quad a^* \sim \pi(\cdot|s')$
3 **end**
4 **else if** *Control setting* **then**
5 $\quad\mid\quad a^* \leftarrow \arg\max_{a' \in \mathcal{A}} \frac{1}{N} \sum_{i=1}^{N} Z_\theta(s', a')_i$
6 **end**
7 $\hat{T}Z_i \leftarrow r + \gamma Z_{\theta^-}(s', a^*)_i, \forall 1 \leq i \leq N$

Output: $\mathrm{MMD}_b^2\left(\{Z_\theta(s, a)_i\}_{i=1}^{N}, \{\hat{T}Z_i\}_{i=1}^{N}; k\right)$

The distributional Bellman operator $\mathcal{T}^\pi$ (Bellemare, Dabney, and Munos 2017) specifies the relation of different return distributions across state-action pairs along the Bellman dynamic; that is, for any $\mu \in \mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$, and any $(s, a) \in \mathcal{S} \times \mathcal{A}$,

$$\mathcal{T}^\pi \mu(s, a) :=$$
$$\int_{\mathcal{S}} \int_{\mathcal{A}} \int_{\mathcal{X}} (f_{\gamma, r})_\# \mu(s', a') \mathcal{R}(dr|s, a) \pi(da'|s') P(ds'|s, a),$$

where $f_{\gamma, r}(z) := r + \gamma z, \forall z$ and $(f_{\gamma, r})_\# \mu(s', a')$ is the pushforward measure of $\mu(s', a')$ by $f_{\gamma, r}$. Note that μ^π is the fixed point of $\mathcal{T}^\pi$, i.e., $\mathcal{T}^\pi \mu^\pi = \mu^\pi$. We are interested in the problem of learning $\mu^\pi \in \mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$ via the distributional Bellman operator $\mathcal{T}^\pi$.

Algorithmic Approach

In practical setting, we must approximate the return distribution via a finite set of statistics as the space of Borel probability measures is infinite-dimensional. Let $Z_\theta(s, a) := \{Z_\theta(s, a)_i\}_{i=1}^{N}$ be a set of parameterized statistics of $\mu^\pi(s, a)$ where θ represents the parameters of the model, e.g., neural networks. Instead of restricting $Z_\theta(s, a)$ to predefined statistic functionals, we model unrestricted statistics, i.e., deterministic samples where each $Z_\theta(s, a)_i$ can be evolved into any form of statistics and we use the Dirac mixture $\hat{\mu}_\theta(s, a) = \frac{1}{N} \sum_{i=1}^{N} \delta_{Z_\theta(s, a)_i}$ to approximate $\mu^\pi(s, a)$. We refer to the deterministic samples $Z_\theta(s, a)$ as particles, and our goal is reduced into learning the particles $Z_\theta(s, a)$ to approximate $\mu^\pi(s, a)$. To this end, the particles $Z_\theta(s, a)$ is deterministically evolved to minimize the MMD distance between the approximate distribution and its distributional Bellman target. Algorithm 1 below presents the generic update in our approach, namely MMDRL.

Intuition The MMDRL reduces into the standard TD or Q-learning when $N = 1$. For $N > 1$, the objective MMD_b when used with SGD and Gaussian kernels $k(x, y) = \exp(-|x - y|^2/h)$ contributes in two ways: (i) The term $\frac{1}{N^2} \sum_{i,j} k(Z_\theta(s, a)_i, Z_\theta(s, a)_j)$ serves as a repulsive force that pushes the particles $\{Z_\theta(s, a)_i\}$ away from each other, preventing them from collapsing into a single mode, with force proportional to $\frac{2}{h} e^{-(Z_\theta(s, a)_i - Z_\theta(s, a)_j)^2/h} |Z_\theta(s, a)_i -$

$Z_\theta(s, a)_j|$; (ii) the term $-\frac{2}{N^2} \sum_{i,j} k(Z_\theta(s, a)_i, \hat{T}Z_j)$ acts as an attractive force which pulls the particles $\{Z_\theta(s, a)_i\}$ closer to their target particles $\{\hat{T}Z_i\}$. This can also be intuitively viewed as a two-sample counterpart to Stein point variational inference (Liu and Wang 2016; Chen et al. 2018).

Particle Representation We can easily extend MMDRL to DQN-like architecture to create a novel deep RL, namely MMDQN (Algorithm 3 in Appendix B). In this work, we explicitly represent the particles $\{Z_\theta(s, a)_i\}_i^N$ in MMDQN via fixed N network outputs as in QR-DQN (Dabney et al. 2018b) for simplicity (details in Appendix B). We emphasize that modeling improvements from IQN (Dabney et al. 2018a) and FQF (Yang et al. 2019) can be naturally applied to MMDQN: we can implicitly generate $\{Z_\theta(s, a)_i\}_i^N$ via applying a neural network function to N samples of a base sampling distribution (e.g., normal or uniform distribution) as in IQN, or we can use the proposal network in FQF to learn the weights of each Dirac components in MMDQN instead of using equal weights $1/N$.

Theoretical Analysis

Here we provide theoretical understanding of MMDRL. Before that, we define the notion of supremum MMD, a MMD counterpart to the supremum Wasserstein in (Bellemare, Dabney, and Munos 2017), to work on $\mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$.

Definition 1. Supremum MMD is a functional $\mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}} \times \mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}} \to \mathbb{R}$ defined by

$$\mathrm{MMD}_\infty(\mu, \nu; k) := \sup_{(s, a) \in \mathcal{S} \times \mathcal{A}} \mathrm{MMD}(\mu(s, a), \nu(s, a); k)$$

for any $\mu, \nu \in \mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$.

We are concerned with the following questions:

1. **Metric property**: When does MMD_∞ induce a metric on $\mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$?

2. **Contraction property**: When is $\mathcal{T}^\pi$ a contraction in MMD_∞?

3. **Convergence property**: How fast do the particles returned by minimizing MMD approach the target distribution it approximates?

The metric property in the first question ensures that MMD_∞ is a meaningful test to distinguish two return distributions on $\mathcal{P}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$. The contraction property in the second question guarantees that following from Banach's fixed point theorem (Banach 1922), $\mathcal{T}^\pi$ has a unique fixed point which is μ^π. In addition, starting with an arbitrary point $\mu_0 \in \mathcal{P}(\mathcal{X})^{\mathcal{X} \times \mathcal{A}}$, $\mathcal{T}^\pi \circ \mathcal{T}^\pi \circ ... \circ \mathcal{T}^\pi \mu_0$ converges at an exponential rate to μ^π in MMD_∞. We provide sufficient conditions to answer the first two questions and derive the convergence rate of the optimal particles in approximating a target distribution for the third question. A short answer is that the first two properties highly depend on the underlying kernel k, and the particles returned by minimizing MMD enjoy a rate $O(1/\sqrt{n})$ regardless of the dimension d of the underlying space $\mathcal{X}$.

Metric Property

Proposition 1. *Let $\tilde{\mathcal{P}}(\mathcal{S}) \subseteq \mathcal{P}(\mathcal{S})$ be some (Borel) subset of the space of the Borel probability measures. If MMD is a metric on $\tilde{\mathcal{P}}(\mathcal{X})$, then MMD_∞ is also a metric on $\tilde{\mathcal{P}}(\mathcal{X})^{\mathcal{S} \times \mathcal{A}}$.*

Proof. See Appendix A.1. □

Theorem 1 below provides *sufficient conditions* for MMD to induce a metric on $\tilde{\mathcal{P}}(\mathcal{X})$.

Theorem 1. *We have*

1. *If the underlying kernel k is **characteristic** (i.e., the induced Bochner integral ψ_p is injective), e.g., Gaussian kernels, then MMD is a metric on $\mathcal{P}(\mathcal{X})$ (Fukumizu et al. 2007; Gretton et al. 2012).*

2. *Define **unrectified** kernels $k_\alpha(x, y) := -\|x - y\|^\alpha, \forall \alpha \in \mathbb{R}, \forall x, y \in \mathcal{X}$. Then $MMD(\cdot, \cdot; k_\alpha)$ is a metric on $\mathcal{P}_\alpha(\mathcal{X})$ for all $\alpha \in (0, 2)$ but not a metric for $\alpha = 2$ (Székely 2003).*

3. *MMD associated with the so-called **exp-prod** kernel $k(x, y) = \exp(\frac{xy}{\sigma^2})$ for any $\sigma > 0$ is a metric on $\mathcal{P}_*(\mathcal{X})$.*

Proof. See Appendix A.2. □

Contraction Property We analyze the contraction of $\mathcal{T}^\pi$ for several important classes of kernels. One such class is shift invariant and scale sensitive kernels. A kernel $k(\cdot, \cdot)$ is said to be *shift invariant* if $k(x + c, y + c) = k(x, y), \forall x, y, c \in \mathcal{X}$; it is said to be *scale sensitive* with order $\alpha > 0$ if $k(cx, cy) = |c|^\alpha k(x, y), \forall x, y \in \mathcal{X}$ and $c \in \mathbb{R}$. For example, the unrectified kernel k_α considered in Theorem 1 is both shift invariant and scale sensitive with order α while Gaussian kernels are only shift invariant.

Theorem 2. *We have*

1. *If the underlying kernel is $k = \sum_{i \in I} c_i k_i$ where each component kernel k_i is both shift invariant and scale sensitive with order $\alpha_i > 0$, $c_i \geq 0$, and I is a (possibly infinite) index set, then $\mathcal{T}^\pi$ is a $\gamma^{\alpha_*/2}$-contraction in MMD_∞ where $\alpha_* := \min_{i \in I} \alpha_i$.*

2. *$\mathcal{T}^\pi$ is **not** a contraction in MMD_∞ associated with either Gaussian kernels or exp-prod kernels $k(x, y) = \exp(\frac{xy}{\sigma^2})$.*

Proof. See Appendix A.3. □

Practical Consideration Theorem 2 provides a negative result for the commonly used Gaussian kernel. In practice, however, we found that Gaussian kernels can promote to match the moments between two distributions and have better empirical performance as compared to the other kernels analyzed in this section. In fact, MMD associated with Gaussian kernels $k(x, y) = \exp(-(x - y)^2/(2\sigma^2))$ can be decomposed into

$$\text{MMD}^2(\mu, \nu; k) = \sum_{n=0}^{\infty} \frac{1}{\sigma^{2n} n!} \left(\tilde{m}_n(\mu) - \tilde{m}_n(\nu) \right)^2$$

where $\tilde{m}_n(\mu) = \mathbb{E}_{x \sim \mu} \left[e^{-x^2/(2\sigma^2)} x^n \right]$, and similarly for $\tilde{m}_n(\nu)$. This indicates that MMD associated with Gaussian

kernels approximately performs moment matching (scaled with a factor $e^{-x^2/(2\sigma^2)}$ for each moment term).

Convergence Rate of Distribution Approximation We justify the goodness of the particles obtained via minimizing MMD in terms of approximating a target distribution.

Theorem 3. *Let $\mathcal{X} \subseteq \mathbb{R}^d$ and $P \in \mathcal{P}(\mathcal{X})$. For any $n \in \mathbb{N}$, let $\{x_i\}_{i=1}^n \subset \mathcal{X}$ be a set of n deterministic points such that $\{x_i\}_{i=1}^n \in \arg\inf_{\{\tilde{x}_i\}_{i=1}^n} MMD(\frac{1}{n} \sum_{i=1}^n \delta_{\tilde{x}_i}, P; \mathcal{F})$. Then, $P_n := \frac{1}{n} \sum_{i=1}^n \delta_{x_i}$ converges to P at a rate of $O(1/\sqrt{n})$ in the sense that for any function h in the unit ball of $\mathcal{F}$, we have*

$$\left| \int_{\mathcal{X}} h(x) dP_n(x) - \int_{\mathcal{X}} h(x) dP(x) \right| = O(1/\sqrt{n}).$$

Remark. MMD enjoys a convergence rate of $O(n^{-1/2})$ [1] regardless of the underlying dimension d while 1-Wasserstein distance has a convergence rate of $O(n^{-1/d})$ (if $d > 2$) (Fournier and Guillin 2015), which is slower for large d.

Proof. We first present two relevant results below (whose proofs are deferred to Appendix A.4) from which the theorem can follow.

Proposition 2. *Let $(X_i)_{i=1}^n$ be n i.i.d. samples of some distribution P. We have*

$$MMD \left(\frac{1}{n} \sum_{i=1}^n \delta_{X_i}, P; \mathcal{F} \right) = O_p(1/\sqrt{n}),$$

where O_p denotes big-O in probability.

Lemma 1. *Let $(a_n)_{n \in \mathbb{N}} \subset \mathbb{R}$ and $(X_n)_{n \in \mathbb{N}} \subset \mathbb{R}$ be sequences of deterministic variables and of random variables, respectively, such that for all n, $|a_n| \leq |X_n|$ almost surely (a.s.). Then, if $X_n = O_p(f(n))$ for some function $f(n) > 0$, we have $a_n = O(f(n))$.*

It follows from the Cauchy-Schwartz inequality in $\mathcal{F}$ that for any function h in the unit ball of $\mathcal{F}$, we have

$$\left| \int_{\mathcal{X}} h(x) dP_n(x) - \int_{\mathcal{X}} h(x) dP(x) \right|$$
$$\leq \|h\|_{\mathcal{F}} \times \left\| \int_{\mathcal{X}} k(x, \cdot) dP_n(x) - \int_{\mathcal{X}} k(x, \cdot) dP(x) \right\|_{\mathcal{F}}$$
$$\leq \text{MMD} \left(\frac{1}{n} \sum_{i=1}^n \delta_{x_i}, P; \mathcal{F} \right). \quad (2)$$

Now by letting $X_n = \text{MMD}(\frac{1}{n} \sum_{i=1}^n \delta_{\tilde{x}_i}, P; \mathcal{F})$, $a_n = \text{MMD}(\frac{1}{n} \sum_{i=1}^n \delta_{x_i}, P; \mathcal{F})$, and $f(n) = 1/\sqrt{n}$, and noting that $a_n \leq X_n, \forall n$, Proposition 2, Lemma 1 and Eq. (2) immediately imply Theorem 3. □

---

[1] In fact, the convergence rate can be improved to $O(1/n)$ using kernel herding technique (Chen, Welling, and Smola 2010).
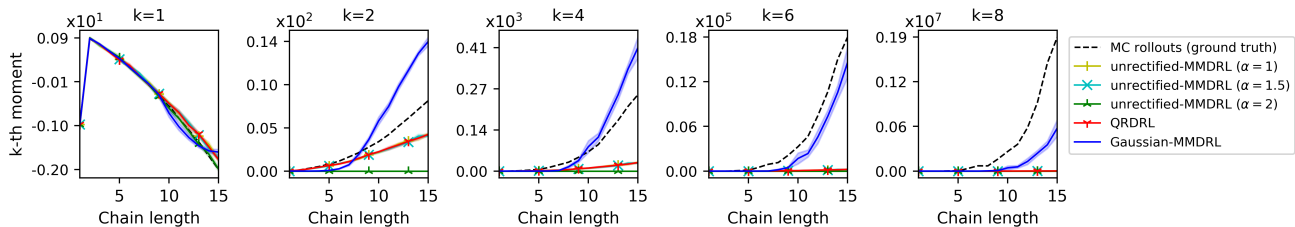
Figure 1: Performance of different methods in approximating the optimal policy's return distribution at the initial state in the chain environment of various chain lengths $K = \{1, 2, ..., 15\}$. The distribution approximation is evaluated in terms of how well a method can approximate the k-th central moment (except $k = 1$ means the expectation) of the target distribution. 95% C.I. with 30 seeds. A variant (Gaussian-MMDRL) of our proposed MMDRL matches the MC rollouts (representing ground truth) much better than QRDRL.

Experimental Results

We first present results with a tabular version of MMDRL to illustrate its behaviour in distribution approximation task. We then combine the MMDRL update to the DQN-style architecture to create a novel deep RL algorithm namely MMDQN, and evaluate it on the Atari-57 games. We give full details of the architectures and hyperparameters used in the experiments in the Appendix B. [2]

Tabular Policy Evaluation

We empirically evaluate that MMDRL with Gaussian kernels (*Gaussian-MMDRL*) can approximately learn the moments of a policy's return distribution as compared to the MMDRL with unrectified kernels (*unrectified-MMDRL*) and the baseline QRDRL.

We use a variant of the classic chain environment (Rowland et al. 2019) . The chain environment of length K is a chain of K states $s_0, ..., s_{K-1}$ where s_0 is the initial state and s_{K-1} is the terminal state (see Figure 2). In each state, there are only two possible actions: (i) forward, which moves the agent one step to the right with probability 0.9 and to s_0 with probability 0.1, or (ii) backward, which transitions the agent to s_0 with probability 0.9 and one step to the right with probability 0.1. The agent receives reward -1 when transitioning to the initial state s_0, reward 1 when reaching the terminal state s_{K-1}, and 0 otherwise. The discount factor is $\gamma = 0.9$. We estimate μ_0^* the return distribution at the initial state of the optimal policy π^* which selects forward action in every state. The longer the chain length K, the more stochastic the optimal policy's return distribution at s_0. We use $10,000$ Monte Carlo rollouts under policy π^* to compute the central moments of μ_0^* as ground truth values. Each method uses only $N = 30$ samples to approximate the target distribution μ_0^* (more algorithm details are presented in Algorithm 2 in Appendix B).

The result is presented in Figure 1. While all the methods approximate the expectation of the target distribution well, their approximation qualities differentiate greatly when it comes to higher order moments. Gaussian-MMDRL, though with only $N = 30$ particles, can approximate higher order

moments more reasonably in this example whereas the rest highly suffer from underestimation. We also experimented with the kernel considered in Theorem 1.3 in this tabular experiment and the Atari game experiment (next part) but found that it is highly inferior to the other kernel choices (even though it has an exact moment matching form as compared to Gaussian kernels, see Appendix A.2) thus we did not include it (we speculate that the shift invariance of Gaussian kernels seems effective when interacting with transition samples from the Bellman dynamics).
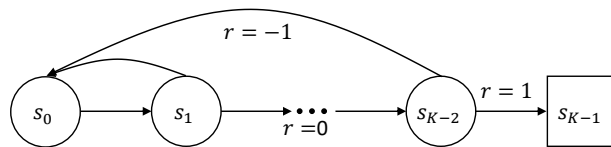

Figure 2: An illustration of a variant of the classic chain MDP with the chain length K.

Atari Games

To demonstrate the effectiveness of MMDRL at scale, we combine the MMDRL in Algorithm 1 with DQN-like architecture to obtain a deep RL agent namely MMDQN (Algorithm 3 in Appendix B). Specifically in this work, we used the same architecture as QR-DQN (Dabney et al. 2018b) for simplicity but more advanced modeling improvements from IQN (Dabney et al. 2018a) and FQF (Yang et al. 2019) can naturally be used in combination to our framework.

Evaluation Protocol We evaluated our algorithm on 55 [3] Atari 2600 games (Bellemare et al. 2013) following the standard training and evaluation procedures (Mnih et al. 2015; van Hasselt, Guez, and Silver 2016) (the full details are in appendix B). We computed human normalized scores for each agent per game. From the human normalized scores for an agent across all games, we extracted three statistics for the agent's performance: the median, the mean and the number of games where the agent's performance is above the human expert's performance.

---

[2]Our official code is available at https://github.com/thanhnguyentang/mmdrl.

[3]We failed to include Defender and Surround games using OpenAI and Dopamine framework.

(a) Different kernel bandwidths (at fixed $N = 200$).

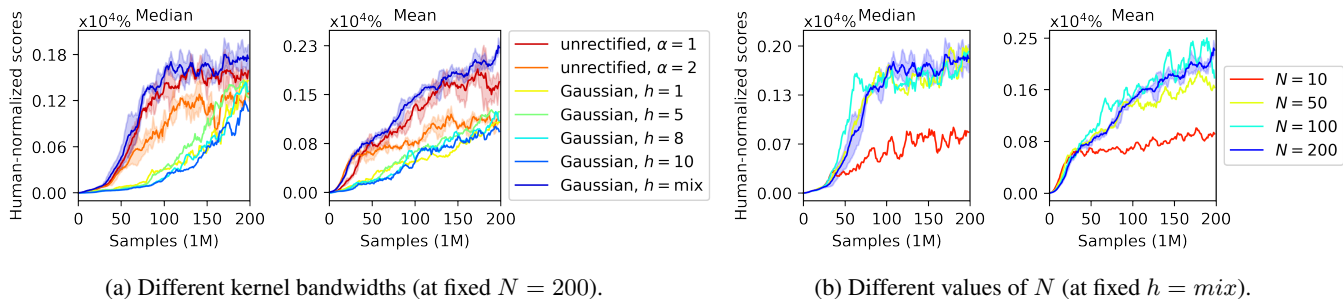(b) Different values of N (at fixed $h = mix$).

Figure 3: The sensitivity of (the human-normalized scores of) MMDQN in the 6 tuning games with respect to: (a) the kernel choices (Gaussian kernels with different bandwidths h and unrectified kernels), and (b) the number of particles N. Here $h = mix$ indicates the mixture of bandwidth values in $\{1, 2, ..., 10\}$. All curves are smoothed over 5 consecutive iterations. 95% C.I. for the $h = mix$, $N = 200$, and unrectified kernel curves (3 seeds) and 1 seed for the other curves.
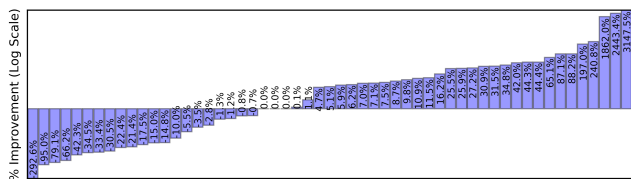

Figure 4: Percentage improvement per-game of MMDQN over QR-DQN-1.

	Mean	Median	>Human	>DQN
DQN	221%	79%	24	0
PRIOR.	580%	124%	39	48
C51	701%	178%	40	50
QR-DQN-1	902%	193%	41	54
RAINBOW	1213%	227%	42	52
IQN	1112%	218%	39	54
FQF	1426%	272%	44	54
MMDQN	**1969**%	213%	41	**55**

Table 1: Mean and median of *best* human-normalized scores across 55 Atari 2600 games. The results for MMDQN are averaged over 3 seeds and the reference results are from (Yang et al. 2019).

Baselines We categorize the baselines into two groups. The first group contains *comparable* methods: DQN, PRIOR., C51, and QR-DQN-1, where DQN (Mnih et al. 2015) and PRIOR. (prioritized experience replay (Schaul et al. 2016)) are classic baselines. The second group includes *reference* methods: RAINBOW (Hessel et al. 2018), IQN, and FQF, which contain algorithmic/modeling improvements orthogonal to MMDQN: RAINBOW combines C51 with prioritized replay and n-step update while IQN and FQF contain modeling improvements as described in the related work section. Since in this work we used the same architecture as QR-DQN and C51 for MMDQN, we directly compare MMDQN with the first group while including the second group for reference.

Hyperparameter Setting For fair comparison with QR-DQN, we used the same hyperparameters: $N = 200$, Adam optimizer (Kingma and Ba 2015) with $lr = 0.00005, \epsilon_{ADAM} = 0.01/32$. We used ϵ-greedy policy with ϵ being decayed at the same rate as in DQN but to a lower value $\epsilon = 0.01$ as commonly used by the distributional RL methods. We used a target network to compute the distributional Bellman target as with DQN. Our implementation is based on OpenAI Gym (Brockman et al. 2016) and the Dopamine framework (Castro et al. 2018).

Kernel Selection We used Gaussian kernels $k_h(x, y) = \exp\left(-(x - y)^2/h\right)$ where $h > 0$. The kernel bandwidth h is crucial to the statistical quality of MMD: overestimated bandwidth results in a flat kernel while underestimated one makes the decision boundary highly irregular. We utilize the kernel mixture trick in (Li, Swersky, and Zemel 2015) which is a mixture of K kernels covering a range of bandwidths $k(x, y) = \sum_{i=1}^{K} k_{h_i}(x, y)$. The Gaussian kernel with a bandwidth mixture yields much a better performance than that with individual bandwidth and unrectified kernels in 6 tuning games: Breakout, Assault, Asterix, MsPacman, Qbert, and BeamRider (see Figure 3 (a)). Figure 3 (b) shows the sensitivity of MMDQN in terms of the number of particles N in the 6 tuning games where too small N adversely affects the performance.

The main empirical result is provided in Table 1 where we compute the mean and median of *best* human normalized scores across 55 Atari games in the 30 no-op evaluation setting (the full raw scores for each game are provided in Appendix C). The table shows that MMDQN significantly outperforms the comparable methods in the first group (DQN, PRIOR., C51 and QR-DQN-1) in all metrics though it shares the same network architecture with C51 and QR-DQN-1. Although we did not include any orthogonal algorithmic/modelling improvements from the reference methods to MMDQN, MMDQN still performs comparably with these methods and even achieve a state-of-the-art mean human-normalized score. In Figure 4 we also provide the percentage improvement per-game of MMDQN over QR-DQN-1 where MMDQN offers significant gains over QR-DQN-1 in a large array of games.

Acknowledgements

References

Banach, S. 1922. Sur Les Operations Dans Les Ensembles Abstraits et Leur Application Aux Equations Integrales. *Fundamenta Mathematicae 3* 133–181.

Bellemare, M. G.; Dabney, W.; and Munos, R. 2017. A Distributional Perspective on Reinforcement Learning. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, 449–458. PMLR.

Bellemare, M. G.; Le Roux, N.; Castro, P. S.; and Moitra, S. 2019. Distributional reinforcement learning with linear function approximation. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2203–2211. PMLR.

Bellemare, M. G.; Naddaf, Y.; Veness, J.; and Bowling, M. 2013. The Arcade Learning Environment: An Evaluation Platform for General Agents. *Journal of Artificial Intelligence Research* 47: 253–279. ISSN 1076-9757. doi: 10.1613/jair.3912.

Bellman, R. 1957. *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press, 1 edition.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *arXiv preprint arXiv:1606.01540* .

Castro, P. S.; Moitra, S.; Gelada, C.; Kumar, S.; and Bellemare, M. G. 2018. Dopamine: A Research Framework for Deep Reinforcement Learning. *CoRR* abs/1812.06110.

Chen, W. Y.; Mackey, L. W.; Gorham, J.; Briol, F.; and Oates, C. J. 2018. Stein Points. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, 843–852. PMLR.

Chen, Y.; Welling, M.; and Smola, A. J. 2010. Super-Samples from Kernel Herding. In *UAI*, 109–116. AUAI Press.

Chung, K.; and Sobel, M. J. 1987. Discounted MDP's: distribution functions and exponential utility maximization. *Siam Journal on Control and Optimization* 25: 49–62.

Dabney, W.; Ostrovski, G.; Silver, D.; and Munos, R. 2018a. Implicit Quantile Networks for Distributional Reinforcement Learning. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, 1104–1113. PMLR.

Dabney, W.; Rowland, M.; Bellemare, M. G.; and Munos, R. 2018b. Distributional Reinforcement Learning With Quantile Regression. In *AAAI*, 2892–2901. AAAI Press.

Fournier, N.; and Guillin, A. 2015. On the rate of convergence in Wasserstein distance of the empirical measure. *Probability Theory and Related Fields* 162(3-4): 707–738.

Fukumizu, K.; Gretton, A.; Sun, X.; and Schölkopf, B. 2007. Kernel Measures of Conditional Dependence. In *NIPS*, 489–496. Curran Associates, Inc.

Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A. J. 2012. A Kernel Two-Sample Test. *J. Mach. Learn. Res.* 13: 723–773.

Hessel, M.; Modayil, J.; van Hasselt, H.; Schaul, T.; Ostrovski, G.; Dabney, W.; Horgan, D.; Piot, B.; Azar, M. G.; and Silver, D. 2018. Rainbow: Combining Improvements in Deep Reinforcement Learning. In *AAAI*, 3215–3222. AAAI Press.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *ICLR (Poster)*.

Li, Y.; Swersky, K.; and Zemel, R. S. 2015. Generative Moment Matching Networks. In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, 1718–1727. JMLR.org.

Liu, Q.; and Wang, D. 2016. Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm. In *NIPS*, 2370–2378.

Lyle, C.; Bellemare, M. G.; and Castro, P. S. 2019. A Comparative Analysis of Expected and Distributional Reinforcement Learning. In *AAAI*, 4504–4511. AAAI Press.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540): 529–533.

Morimura, T.; Sugiyama, M.; Kashima, H.; Hachiya, H.; and Tanaka, T. 2010a. Nonparametric Return Distribution Approximation for Reinforcement Learning. In *ICML*, 799–806. Omnipress.

Morimura, T.; Sugiyama, M.; Kashima, H.; Hachiya, H.; and Tanaka, T. 2010b. Parametric Return Density Estimation for Reinforcement Learning. In *UAI*, 368–375. AUAI Press.

Puterman, M. L. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Rowland, M.; Bellemare, M. G.; Dabney, W.; Munos, R.; and Teh, Y. W. 2018. An Analysis of Categorical Distributional Reinforcement Learning. In *AISTATS*, volume 84 of *Proceedings of Machine Learning Research*, 29–37. PMLR.

Rowland, M.; Dadashi, R.; Kumar, S.; Munos, R.; Bellemare, M. G.; and Dabney, W. 2019. Statistics and Samples in Distributional Reinforcement Learning. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, 5528–5536. PMLR.

Rummery, G. A.; and Niranjan, M. 1994. *On-line Q-learning using connectionist systems*, volume 37. University of Cambridge, Department of Engineering Cambridge, UK.

Schaul, T.; Quan, J.; Antonoglou, I.; and Silver, D. 2016. Prioritized Experience Replay. In *ICLR (Poster)*.

Smola, A. J.; Gretton, A.; Song, L.; and Schölkopf, B. 2007. A Hilbert Space Embedding for Distributions. In *ALT*, volume 4754 of *Lecture Notes in Computer Science*, 13–31. Springer.

Sutton, R. S. 1988. Learning to Predict by the Methods of Temporal Differences. *Mach. Learn.* 3: 9–44.

Sutton, R. S.; Barto, A. G.; et al. 1998. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge.

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

van Hasselt, H.; Guez, A.; and Silver, D. 2016. Deep Reinforcement Learning with Double Q-Learning. In *AAAI*, 2094–2100. AAAI Press.

Watkins, C. J. C. H.; and Dayan, P. 1992. Q-learning. In *Machine Learning*, 279–292.

Yang, D.; Zhao, L.; Lin, Z.; Qin, T.; Bian, J.; and Liu, T.-Y. 2019. Fully Parameterized Quantile Function for Distributional Reinforcement Learning. In *Advances in Neural Information Processing Systems*, 6190–6199.