Beyond Marginals: Capturing Correlated Returns through Joint Distributional Reinforcement Learning

Ege C. Kaya, Mahsa Ghasemi, Abolfazl Hashemi
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47906
kayae@purdue.edu, mahsa@purdue.edu, abolfazl@purdue.edu

Abstract

Distributional reinforcement learning (DRL) has emerged in recent years as a powerful paradigm that aims to learn the full distributions of returns starting from different state-action pairs under a policy, rather than only their expected values. The existing DRL algorithms learn the return distribution independently for each action at a state. However, we establish that in many environments, the returns for different actions at the same state are statistically dependent due to shared transition and reward structure, and that learning only per-action marginals discards potentially exploitable information. We formalize a joint MDP view that lifts and MDP into a POMDP whose hidden states encode coupled potential outcomes across actions, and we derive joint distributional Bellman equations together with a joint iterative policy evaluation (JIPE) scheme with convergence guarantees. On the algorithmic side, we introduce a deep learning method that represents joint returns with homoscedastic Gaussian mixture models and trains them by matching a multivariate TD target. Empirically, we validate the proposed framework on two custom MDPs with known correlation structure (a bandit with shared randomness in rewards, and a windy gridworld environment), and illustrate the learned joint structure in the classic control task CartPole and the Arcade Learning Environment game Pong. Together, these results demonstrate that modeling cross-action return dependence yields accurate joint moments and informative joint distributions that can support safer, more sample-efficient control.

1 Introduction

Classic RL. Reinforcement learning (RL) has long been utilized as a powerful framework for sequential decision-making problems where the interaction of the agent and the environment follows a Markov decision process (MDP). An MDP $\mathcal{M} = (\mathcal{S}, \varsigma_0, \mathcal{A}, R, P, \gamma)$ is a quintuple where $\mathcal{S}$ is a finite set designating the space of states, $\varsigma_0 \in \Delta(\mathcal{S})$ is the initial distribution of states, $\mathcal{A}$ is the set of actions that the agent may take, $R : \mathcal{S} \times \mathcal{A} \to \Delta(\mathbb{R})$ is a stochastic, real-valued reward function, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition kernel, and $0 < \gamma < 1$ is the discount factor [1]. In conventional RL, the learning objective is to maximize the expected utility, called return, which captures the agent's cumulative reward throughout its interaction. A policy π for MDP $\mathcal{M}$ may be thought of as a decision rule $\pi : \mathcal{S} \to \mathcal{A}$.[1]

To evaluate the merit of a given policy π, one may consider the expected sum of discounted rewards over the time horizon obtained by the policy, denoted $\mathbb{E}[Z^\pi(s,a)] := \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \mid s_0 =$

[1] A celebrated result in RL states that in the discounted and infinite-horizon setting, one can find an optimal policy π^* that is *stationary* and *deterministic* [2], and we will direct our attention to this case.

2nd Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET 2025).

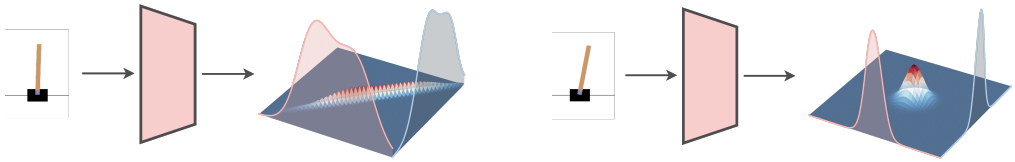

Figure 1: Joint distributions of returns learned by our method in the CartPole environment. On the left: When the pole is perfectly balanced, the returns of both actions are perfectly correlated and the joint distribution is basically a ridge. On the right: When the pole starts to lose balance to one side, the joint distribution becomes less degenerate. The degenerate ridge structure, in the case of a bivariate Gaussian distribution, is observed when the correlation coefficient of the two marginal random variables approaches 1 (or -1). This indicates that the two marginals are extremely correlated (or extremely negatively correlated, respectively). We anticipate this to be the case when the pole is already perfectly balanced and stable, as the system is in near-complete symmetry and thus pushing the cart to either side should have nearly the same exact returns. The curves on the edge of the plot show the two marginal distributions, which would have been learned by a conventional DRL method.

$s, a_0 = a]$, also known as the Q-function or the state-action value function. We remark that $Z^\pi(s, a)$ is the random variable which we will refer to as the *state-action returns*.

The objective in RL, then, is to find an *optimal policy* π^* with an optimal state-action value function, i.e., to find $\pi^* \in \operatorname{argmax}_{\pi \in \Pi} Q^\pi(s, a)$, for all $(s, a) \in \mathcal{S} \times \mathcal{A}$, where Π denotes the set of all possible policies for $\mathcal{M}$. Famously, an optimal state-action value function Q^* satisfies the *Bellman optimality equation* [3]

$$Q^*(s, a) = \mathbb{E}[R(s, a)] + \gamma \mathbb{E}[\max_{a' \in \mathcal{A}} Q^*(s', a')], \tag{1}$$

a premise which many RL algorithms have been built upon [4, 5, 6, 7, 8, 9, 10, 11, 12, 13].

Distributional RL. More recently, a new paradigm called distributional reinforcement learning (DRL) [14] has emerged, partly based on the argument that reducing the return to its average value can obscure important aspects of uncertainty, variability, and risk, often leading to suboptimal exploration or brittle policies in highly stochastic settings. DRL augments the RL paradigm by modeling the full probability distribution over returns rather than only its expectation, capturing higher-order moments and tail behavior. This richer characterization aims to enable agents to reason about variability and risk, with the end goal of improving the agent's performance, alongside secondary objectives such as sample efficiency and policy robustness.

Every DRL method up to now has been built around the same central tenet of trying to estimate marginal return distributions for each action independently. Although different methods estimate different characterizations for these marginal distributions, at the end of the day, the entity they propose to model and estimate is some characterization of the θ-parameterized marginal state-action return $Z_\theta(s, a)$ for a given state s and for each action $a \in \mathcal{A}$. While their differences in approach naturally bring about changes in the modeling and estimation processes due to the change in the parametrization of the distributions, the fact that all methods are still only interested in estimating a single, independent marginal distribution for each of the $|\mathcal{A}|$ actions remains unchanged.

1.1 Novelty

In this paper, we argue for the rationality, or at least the theoretical interest, in pursuing a more unifying approach: It is only natural, we think, to be curious about the *joint distribution of these state-action returns* (see Figure 1).

We argue that there is a nontrivial set of MDPs where, given a state s, there is dependence to be discovered between the returns of different actions. We present the following two motivating examples, after which a formal explanation follows:

Example 1 (Dependence due to rewards)**.** *Consider an MDP with bounded $\mathcal{S} \subset \mathbb{R}^2$, $\mathcal{A} = \{1, 2\}$, and with stochastic reward defined as $R(s, a) := x_a$ for $a \in \mathcal{A}$, where $x \sim \mathcal{N}(s, \Sigma)$, and Σ is a nondiagonal positive definite matrix. It is self-evident, in this case, that at any state s, the rewards $R(s, 1)$ and $R(s, 2)$ will be dependent random variables, and their covariance will be $\Sigma_{1,2}$. Because the return $Z^\pi(s, a)$ of any policy π for any state-action pair is a weighted sum of such dependent*

rewards, the returns will also have a nontrivial joint distribution, where by nontrivial we mean a joint distribution which is not simply the product of its marginal distributions.

Example 2 (Dependence due to transition dynamics)**.** *Consider an MDP with $\mathcal{S} = \mathbb{R}$ and $\mathcal{A} = \{-1, 1\}$. Let X be a Bernoulli random variable. Let the next state be determined in terms of a state-action-dependent measurable function of X as*

$$S' = f(s, a, X) = \begin{cases} s + a - 1, & \text{if } X = 1 \\ s + a, & \text{otherwise,} \end{cases} \tag{2}$$

and $P(\cdot \mid s, a) := \mathrm{Law}(S')$. The stochasticity of the transition dynamics of the environment is dependent on the random variable X. Clearly, the next state random variables $S'_1 = f(s, -1, X)$ and $S'_2 = f(s, 1, X)$ will be dependent random variables.

In this example, X might be thought of as modeling the presence of an environmental factor such as wind, which pushes the agent towards the left. Other examples of such factors may include a market fluctuation, or a system-wide latency spike, factors which simultaneously affect the results of all possible actions an agent could take at that moment.

The fact that these examples are specifically constructed to have dependencies should not give the impression that they do not arise in regular RL problems. The dependence of action returns is highly intrinsic even in the presence of a deterministic reward function, a key feature in practical applications of RL, particularly those involving function approximation.

1.2 Related Work

An early formulation of the DRL paradigm was introduced by [15, 16], who developed parametric and nonparametric estimators for return distributions via the distributional Bellman equations, preceding deep learning methods. Following the success of DQN [4], several DRL approaches emerged. [17] proposes a taxonomy based on return distribution parameterization and choice of optimization metric, which we adopt. C51 [14] and its extension Rainbow [18] model each state-action return distribution as a 51-bin categorical distribution, trained using KL divergence. In contrast, QR-DQN [19], IQN [17], and FQF [20] model the quantile function with increasing flexibility and train using Huber quantile regression. MoG-DQN [21], most related to our work, uses Gaussian mixture models (GMMs) and minimizes the Jensen-Tsallis divergence. Similarly, [22] employ GMMs but optimize the Cramér-2 distance, which we adopt in our experiments. Lastly, DRL methods addressing multivariate rewards relate to our approach. Bellman GAN [23] leverages GANs to model multivariate return distributions, while [24] introduces MD3QN to jointly model returns from multiple reward sources and their correlations.

2 Joint Distributional RL: A Principled Framework

2.1 Principled Modeling of Correlations via Joint DRL

Having established the existence of actions' interdependency, in this section we develop a principled framework to facilitate studying this phenomenon. We start by stating two assumptions which are common in RL literature [25, 26].

Assumption 1. *We assume that the action space $\mathcal{A}$ is finite and $|\mathcal{A}| = N$. Since there exists a bijection between $\mathcal{A}$ and $[N]$, in the rest of the work, we will directly think $\mathcal{A} = [N]$ for ease of notation, so each action will be referred to by an integer $1 \leq n \leq N$.*

Assumption 2. *For all (s, a), it holds that $r_{\min} \leq R(s, a) \leq r_{\max}$ almost surely for some $r_{\min}, r_{\max} \in \mathbb{R}$.*

Now, in light of the motivating examples of the previous section, we formalize our analysis of the phenomena at play with the following definition.

Definition 1 (Joint MDP)**.** *Let $\mathcal{M}$ be an MDP $(\mathcal{S}, \varsigma_0, \mathcal{A}, R, P, \gamma)$. For any $s \in \mathcal{S}$, let $C_P(s)$ be some coupling on $\mathcal{S}^N$ with marginals $\{P(\cdot \mid s, i)\}_{i=1}^N$ and $C_R(s)$ some coupling on $\mathbb{R}^N$ with marginals $\{R(s, i)\}_{i=1}^N$. Consider the partially observable MDP (POMDP) $\mathcal{J} = (\mathcal{X}, \varsigma'_0, \mathcal{A}, P', R', \Omega, O, \gamma)$, which we will refer to as the joint MDP behind $\mathcal{M}$, where*

1. $\mathcal{X} := \mathcal{S}^N \times \mathbb{R}^N$. *We write a typical element* $x \in \mathcal{X}$ *as* $x = (\mathbf{s}, \mathbf{r})$, *where* $\mathbf{s} = (s_1, \ldots, s_N)$ *and* $\mathbf{r} = (r_1, \ldots, r_N)$.

2. $\varsigma_0'(x) := \begin{cases} \varsigma_0(s) \times \delta(\mathbf{0}), & \text{if } s_i = s \text{ for all } i \in [N] \\ 0, & \text{otherwise}, \end{cases}$

   where $\mathbf{0}$ *indicates a vector of* N *zeros. In other words, the initial distribution over* $\mathcal{X}$ *only assigns nonzero probability to configurations which initialize all* N *states in* $\mathbf{s}$ *at the same state, with all "initial rewards" being* 0.

3. $P'(\cdot \mid x, a) := C_P(s_a) \times C_R(s_a)$, *the product measure of the transition and reward couplings.*

4. $R'(x, a) := r_a$, *deterministic.*

5. $\Omega := \mathcal{S} \times \mathbb{R}$.

6. $O(o \mid x, a) := \delta_{(s_a, r_a)}(o)$.

At each decision time t within the POMDP $\mathcal{J}$, the hidden state is a pair of vectors $x_t = (\mathbf{s}_t, \mathbf{r}_t)$, with $\mathbf{s}_t = (s_{t,1}, \ldots, s_{t,N}) \in \mathcal{S}^N$ and $\mathbf{r}_t = (r_{t,1}, \ldots, s_{r,N}) \in \mathbb{R}^N$. The ith entries $s_{t,i}$ and $r_{t,i}$ denote the next state and reward that would be obtained if action i were to be taken from the current base state. After the agent selects a_t, the environment reveals (s_{t,a_t}, r_{t,a_t}) and the base state updates to $s_{t+1} = s_{t,a_t}$, and a fresh pair $(\mathbf{s}_{t+1}, \mathbf{r}_{t+1})$ is drawn at s_{t+1} according to the specified couplings. For initialization, a state $s \sim \varsigma_0$ is sampled and $\mathbf{s}_0 = (s, \ldots, s)$ is set. $\mathbf{r}_0 = \mathbf{0}$ is set as a placeholder, since the reward at the initial state is a reward obtained before any actions have been played, and hence has no meaning and is unused.

This representation is observationally equivalent to the original MDP $\mathcal{M}$: For any (s, a), the revealed pair (s_a, r_a) has exactly the same distribution as (S', R) under the kernels $P(\cdot \mid s, a)$ and $R(s, a)$ of $\mathcal{M}$. The only change is that the POMDP's hidden state preserves the joint and potentially dependent *counterfactual* outcomes across actions at each step. This makes it possible to state and learn joint statistics and to write joint Bellman relations, without altering the agent's observed interaction process.

In practice, how do we attempt to model this joint MDP? The following two definitions formalize the vector-valued random variable of joint returns whose distribution we aim to estimate.

Definition 2. *Let* $Z^\pi(s, a)$ *denote the state-action return of policy* π *at* (s, a). *Then, the* N*-variate joint return of policy* π *at* s *is defined as*

$$Z^\pi(s) = [Z^\pi(s, 1), \ldots, Z^\pi(s, N)]^T. \tag{3}$$

Definition 3. *Let* $\eta^\pi(s, a) = \mathrm{Law}(Z^\pi(s, a))$. *Then, the joint return distribution of policy* π *at* $s \in \mathcal{S}$ *is a coupling of* $\{\eta^\pi(s, i)\}_{i=1}^N$. *Additionally, we use*

$$\eta^\pi(s; i, j) = \int_{z_{\bar{\mathbf{a}}}} \eta^\pi(s) \mathrm{d}z_{\bar{\mathbf{a}}} \tag{4}$$

to denote the bivariate marginal distribution of $\eta^\pi(s)$ *over the* i^{th} *and* j^{th} *dimensions. The notation* $\mathrm{d}z_{\bar{\mathbf{a}}}$ *denotes that the integral is over dimensions* $\mathcal{A} \backslash \{i, j\}$.

We now refer to the following simple example for intuition about an additional problem we face in trying to estimate these joint distributions through samples.

Example 3. *Suppose a simple scenario where we want to estimate a simple bivariate Gaussian distribution but we are limited to only observing samples from its marginals. We observe samples* $x_1, \ldots, x_N$ *for the first marginal dimension, and samples* $y_1, \ldots, y_N$ *for the second marginal dimension, each an element of* $\mathbb{R}$. *We can then use sample statistics to naively estimate* $\hat{\mu}_1 = \frac{1}{N} \sum_{i=1}^N x_i$, $\hat{\mu}_2 = \frac{1}{N} \sum_{i=1}^N y_i$, $\hat{\sigma}_1^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \hat{\mu}_1)^2$, $\hat{\sigma}_2^2 = \frac{1}{N-1} \sum_{i=1}^N (y_i - \hat{\mu}_2)^2$. *We can then attempt to estimate the joint, bivariate Gaussian distribution as* $\mathcal{N}(\hat{\mu}, \hat{\Sigma})$, *where*

$$\hat{\boldsymbol{\mu}} = \begin{bmatrix} \hat{\mu}_1 \\ \hat{\mu}_2 \end{bmatrix}, \qquad \hat{\Sigma} = \begin{bmatrix} \hat{\sigma}_1^2 & \rho\, \hat{\sigma}_1 \hat{\sigma}_2 \\ \rho\, \hat{\sigma}_1 \hat{\sigma}_2 & \hat{\sigma}_2^2 \end{bmatrix}.$$

It is at this point that we must stop and ask a question: If we suspect a nondiagonal covariance matrix, how does one estimate the value for ρ without ever observing a sample from the joint distribution? Although it is possible to make educated estimates for every marginal statistical, it is impossible to have an informed estimate of ρ without having at any point been exposed to joint samples from the bivariate distribution.

In this work, we will be interested in learning the joint distribution $\eta^\pi(s)$ associated with a reference policy π, or certain statistical functionals that could aid us in inferring it. Conventional RL is concerned with learning the mean functional $\mu^\pi(s) \in \mathbb{R}^{\mathcal{A}}$, i.e., the state-action value function Q. As we are interested in inferring the correlations, a natural functional to consider in addition to $\mu^\pi(s)$ is the $N \times N$ covariance matrix derived from $\eta^\pi(s)$ which we denote by $\Sigma^\pi(s)$.

However, as illustrated by Example 3, since any off-diagonal element $\Sigma^\pi(s)_{a,a'}$ of $\Sigma^\pi(s)$ relates knowledge about the joint returns of two actions at state s, the customary transition structure of $\tau := (s, a, r, s', a')$ will no longer suffice to estimate these elements. Indeed, if we hope to learn a meaningful joint distribution of the returns of multiple actions at a state, we must change the structure of our saved and sampled experience replays to be $\tau^2 := (s, a_1, a_2, r_1, r_2, s'_1, s'_2, a'_1, a'_2)$, where a_1 and a_2 are two distinct actions that can potentially be played at state s, r_1 and r_2 the ensuing rewards, s'_1 and s'_2 the respective next states, and a'_1 and a'_2 the actions chosen by π in the next states. Much like how, in the conventional DRL setting, we would expect the observation of the transition τ to lend us guidance in updating our estimate of $\eta^\pi(s, a)$, we would now expect to leverage the observation of τ^2 to update our estimates of $\mu^\pi(s)_{a_1}$, $\mu^\pi(s)_{a_2}$, the diagonal covariance elements $\Sigma^\pi(s)_{a_1,a_1}$, $\Sigma^\pi(s)_{a_2,a_2}$ and finally the off-diagonal covariance element $\Sigma^\pi(s)_{a_1,a_2}$. Obviously, this would result in updating our estimate of the bivariate marginal distribution $\eta^\pi(s; a_1, a_2)$.

In the formalism of Definition 1, to get access to such samples as τ^2, we must modify our observation space and kernel to allow us a look into the joint structure. Specifically, letting $\Omega := \mathcal{S}^2 \times \mathbb{R}^2$ and $O(o \mid x, a_1) := \delta_{((s_{a_1}, s_{a_2}),(r_{a_1}, r_{a_2}))}(o)$ suffices. Namely, at any decision time of the joint MDP, the observation model lets us peek into the next state and reward of the played action a_1 (dictated by a policy π), and the next state and reward of one additional, counterfactual action $a_2 \neq a_1$ that could have been played instead.

Can we obtain such joint samples from an MDP in practice, and if yes, how? Many applications of RL are increasingly relying on digital twin technologies, enabling a near-perfect simulation of the reality. Consider, for instance, a robotics task. It is not implausible to assume that we would have access to a perfect simulation of the system, in which, at a state, we can take an action, observe its consequences, rewind the simulation to the previous state and then take another action to observe its consequences. In programming terms, this process can be thought of as saving a state of the environment and the random number generator before playing an action, and then restoring these states to play another action.

We now introduce the joint Bellman equations.

Definition 4. *Let $Z^\pi(s)$ be the N-variate joint return of policy π. Consider a sample transition $(S = s, A_1 = a_1, R_1, S'_1, A'_1, A_2 = a_2, R_2, S'_2, A'_2)$. Then, the 2nd-order N-variate joint Bellman equations are given as*

$$\begin{aligned} Z^\pi(s, a_1) &\overset{D}{=} R_1 + \gamma Z^\pi(S'_1, A'_1), \\ (Z^\pi(s, a_1))^2 &\overset{D}{=} (R_1 + \gamma Z^\pi(S'_1, A'_1))^2, \\ Z^\pi(s, a_1) \cdot Z^\pi(s, a_2) &\overset{D}{=} (R_1 + \gamma Z^\pi(S'_1, A'_1)) \cdot (R_2 + \gamma Z^\pi(S'_2, A'_2)). \end{aligned} \tag{5}$$

Proposition 1. *Consider a sample transition $(S = s, A_1 = a_1, R_1, S'_1, A'_1, A_2 = a_2, R_2, S'_2, A'_2)$. Then, the 2nd-order N-variate joint Bellman distributional equations are given as follows*

$$\begin{aligned} \mathbb{E}[Z^\pi(s, a_1)] &= \mathbb{E}[R_1 + \gamma Z^\pi(S'_1, A'_1) \mid S = s, A_1 = a_1], \\ \mathbb{E}[(Z^\pi(s, a_1))^2] &= \mathbb{E}[(R_1 + \gamma Z^\pi(S'_1, A'_1))^2 \mid S = s, A_1 = a_1], \\ \mathbb{E}[Z^\pi(s, a_1) \cdot Z^\pi(s, a_2)] &= \\ &\mathbb{E}[(R_1 + \gamma\ Z^\pi(S'_1, A'_1)) \cdot (R_2 + \gamma Z^\pi(S'_2, A'_2)) \mid S = s, A_1 = a_1, A_2 = a_2], \end{aligned} \tag{6}$$

where $\mathbb{E}[\cdot]$ denotes the expectation with respect to the joint distribution over all random variables involved.

Evidently, these equations provide us with consistency conditions that the first and second moments of the return $Z^\pi(s,a)$ must satisfy, in distribution and in expectation.

2.2 Joint Iterative Policy Evaluation (JIPE)

We can compactly represent the 2nd-order N-variate joint Bellman equations by defining a suitable operator. For each (s,a), let $M_{(s,a)} \in \mathbb{R}^{N+1}$ be a vector that concatenates $\mathbb{E}[Z^\pi(s,a)]$ and $\mathbb{E}[(Z^\pi(s,a))^2]$ as its first and second coordinates and $\mathbb{E}[Z^\pi(s,a) \cdot Z^\pi(s,\tilde{a})]$, where $\tilde{a} \in \mathcal{A}$, $\tilde{a} \neq a$, as its last $N-1$ coordinates. With this notation, let us define, for all (s,a),

$$\begin{aligned} M_\mu(s,a) &:= M_{(s,a),1}, \quad M_\mu \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \\ M_\sigma(s,a) &:= M_{(s,a),2}, \quad M_\sigma \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \\ M_c(s,a) &:= \begin{bmatrix} M_{(s,a),3} & \cdots & M_{(s,a),N+1} \end{bmatrix}^T, \quad M_c \in \mathbb{R}^{(N-1)\times\mathcal{S}\times\mathcal{A}} \\ M &:= \begin{bmatrix} M_\mu^T & M_\sigma^T & M_c^T \end{bmatrix}^T, \quad M \in \mathbb{R}^{(N+1)\times\mathcal{S}\times\mathcal{A}}. \end{aligned} \tag{7}$$

M describes the collection of the first moment (mean) and the second moments of the N-variate joint return, collected by M_μ and M_σ, M_c, respectively. Furthermore, we can represent the 2nd-order N-variate joint Bellman equations by the following 2nd-order N-variate joint Bellman operator

$$\mathcal{T}^\pi_{2,N} : \mathbb{R}^{(N+1)\times\mathcal{S}\times\mathcal{A}} \to \mathbb{R}^{(N+1)\times\mathcal{S}\times\mathcal{A}}, \qquad M = \mathcal{T}^\pi_{2,N} M. \tag{8}$$

Based on this notation, we propose the following dynamic programming approach, referred to as the *N-variate joint iterative policy evaluation (JIPE) scheme*, which repeatedly applies the 2nd-order N-variate joint Bellman operator $\mathcal{T}^\pi_{2,N}$

$$M^{k+1} = \mathcal{T}^\pi_{2,N} M^k, \quad M^0 \in \mathbb{R}^{(N+1)\times\mathcal{S}\times\mathcal{A}}. \tag{9}$$

Theorem 1 (proved in Appendix A) states the convergence of the scheme in (9). Note that, for the simplicity of notation, the theorem is stated in terms of learning the uncentered matrix of second moments, $\bar{\Sigma}^\pi(s)$, from which the covariance can be derived easily as $\Sigma^\pi(s) = \bar{\Sigma}^\pi(s) - \mu^\pi(s)\mu^\pi(s)^T$.

Theorem 1 (Convergence of JIPE)**.** *Suppose Assumptions 1 and 2 hold. Consider the JIPE scheme in* (9)*. For any $s \in \mathcal{S}$, let $\mu^k(s)$ and $\bar{\Sigma}^k(s)$ denote the mean and the second moment matrix recovered from M^k. Then,*

$$\|\mu^k(s) - \mu^\pi(s)\|_\infty = \mathcal{O}\left(\gamma^k\right), \quad \|\bar{\Sigma}^k(s) - \bar{\Sigma}^\pi(s)\|_\infty = \mathcal{O}\left(\frac{\gamma^k}{1-\gamma}\right).$$

3 Learning Optimal Joint Distributions via Neural Networks

We are now ready to present an algorithmic approach to learning the joint distribution, leveraging the deep learning paradigm. We propose to model the state-action return as a Gaussian mixture model with K components (K-GMM), whose parameters are estimated by a neural network with weights θ. We note that [21] and [22] have previously suggested such an approach, however, as usual, these works only consider the estimation of the marginal returns and not of the joint. For reasons such as reduced computational complexity and feasibility, we choose to only deal with homoscedastic K-GMMs, i.e., for a given $s \in \mathcal{S}$, $\Sigma^\theta_i(s) = \Sigma^\theta(s)$ for all $i \in [K]$.

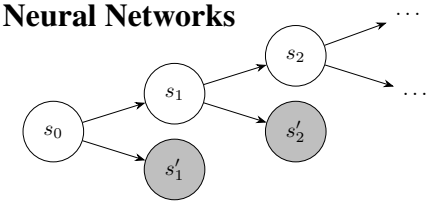

Figure 2: Demonstrating the state transitions stored as a tree. Starting at state s_0, we store two possible next states s_1 and s'_1 reached by playing two actions. Afterwards, s_1 is considered to be the *base state*, that is, the next two states to be stored s_2 and s'_2 are only possible next states reachable from s_1. The next states reachable from s'_1 are not considered. This prevents the number of stored states from increasing exponentially.

Firstly, we would like to remind the reader of the discussion in Section 2 of how our experience replay transitions must have the form $\tau^2 = (s, a_1, a_2, r_1, r_2, s'_1, s'_2)$ for us to have any hope of learning the correlation structure of the joint returns. (See Figure 2 for a way to gather trajectories τ^2 without the number of states exponentially increasing.) We now propose the following

Table 1: SRB joint moments, calculated analytically and estimated by JIPE.

	True	JIPE	$\|\Delta\|_\infty$
$\boldsymbol{\mu}^T$	$\begin{bmatrix}1.8 & 2.0\end{bmatrix}$	$\begin{bmatrix}1.8 & 2.0\end{bmatrix}$	1.849e−12
Corr	$\begin{bmatrix}1.000 & 0.942\\ 0.942 & 1.000\end{bmatrix}$	$\begin{bmatrix}1.000 & 0.942\\ 0.942 & 1.000\end{bmatrix}$	4.441e−16

Table 2: WGW joint moments in the starting cell for actions `RIGHT`, `LEFT`, `UP`, `DOWN`.

	True	JIPE	$\|\Delta\|_\infty$
$\boldsymbol{\mu}^T$	$\begin{bmatrix}0.771 & 0.732 & 0.792 & 0.732\end{bmatrix}$	$\begin{bmatrix}0.771 & 0.732 & 0.792 & 0.732\end{bmatrix}$	2.004e−6
Corr	$\begin{bmatrix}1.000 & 0.833 & 0.866 & 0.833\\ 0.833 & 1.000 & 0.866 & 1.000\\ 0.866 & 0.866 & 1.000 & 0.866\\ 0.833 & 1.000 & 0.866 & 1.000\end{bmatrix}$	$\begin{bmatrix}1.000 & 0.833 & 0.866 & 0.833\\ 0.833 & 1.000 & 0.866 & 1.000\\ 0.866 & 0.866 & 1.000 & 0.866\\ 0.833 & 1.000 & 0.866 & 1.000\end{bmatrix}$	1.612e−4

distributional variant of the standard Q-learning algorithm, utilizing transitions of form τ^2 for the learning of joint state-action returns: At each update step, for a sampled experience replay transition τ^2, we calculate the distributional temporal difference error between the current state's bivariate marginal return distribution where all the actions $a \in \mathcal{A} \setminus \{a_1, a_2\}$ have been marginalized out, i.e., $\eta_\theta(s; a_1, a_2)$, and a TD target distribution $\eta^*_\omega(s'_1, s'_2)$, which will be the distribution of a random variable we denote by $\mathbf{r} + \gamma Z^*_\omega(s'_1, s'_2)$. In other words, we take our temporal difference error to be $\mathcal{L}\left(\eta_\theta(s; a_1, a_2),\ \eta^*_\omega(\, s'_1, s'_2)\right)$, which then gets used to update the neural network weights θ through backpropagation and stochastic gradient descent-based methods.[2] Any statistical distance may in theory be used for $\mathcal{L}$.

The nature of $\eta^*_\omega(s'_1, s'_2) = \mathrm{Law}(\mathbf{r}+\gamma Z^*_\omega(s'_1, s'_2))$ must now be specified. This distribution resembles the familiar TD target of both the conventional distributional and non-distributional RL settings, but due to its multivariate nature, some clarifications must be made. In truth, $\eta^*_\omega(s'_1, s'_2)$ is a coupling: It is a bivariate joint distribution whose univariate marginal distributions are the TD target distributions for $\{\eta_\theta(s, a_i)\}^2_{i=1}$, i.e., $\mathrm{Law}(r_i + \gamma Z_\omega(s'_i, a^*_i))$, where $a^*_i \in \mathrm{argmax}_{a' \in \mathcal{A}} \mathbb{E}[Z_\omega(s; a')]$. Going back to GMM terminology, the ith univariate marginal dimension of $\eta^*_\omega(s'_1, s'_2)$ is a K-GMM with mixing coefficients $\rho_\omega(s'_i)$ and means $r_i + \gamma\mu_{\omega,k}(s'_i, a^*_i)$.

We have now specified the mixing coefficients and the means of the TD target K-GMM $\eta^*_\omega(s'_1, s'_2)$, and only the covariance remains to be specified. The covariance requires more special consideration. Let us now refer to that covariance matrix as $\Sigma_\omega(s'_1, s'_2)$. Given our previous logic for how we construct the coupling target distribution, our covariance matrix must now satisfy

$$\Sigma_{\omega,i,j}(s'_1, s'_2) = \mathrm{cov}(r_i + \gamma Z_\omega(s'_i, a^*_i), r_j + \gamma Z_\omega(s'_j; a^*_j)) \tag{10}$$

as a sample-based estimate of the true covariance $\mathrm{cov}(R(s, a_i)+\gamma Z(S'_i, a^*_i), R(s, a_j)+\gamma Z(S'_j, a^*_j))$.

We remark that with the provision that the TD target distribution $\eta^*_\omega(s'_1, s'_2)$ must be a coupling of the TD target distributions of the two univariate marginal distributions $\eta_\theta(s, a_1)$ and $\eta_\theta(s, a_2)$, it must, at its most general form, be the distribution of a K^2-GMM. Letting $(k_1, k_2) \in \{1, \ldots, K\}^2$ index the K^2 components of the target mixture, and referring back to the homoscedasticity assumption

[2]We remind that the target random variable is calculated with a separate set of parameters ω (as opposed to θ), the parameters of the so-called *target network* [4].

mentioned in the beginning of the section, we finally have

$$\eta^*_\omega(s'_1, s'_2) = \sum_{k_1=1}^{K} \sum_{k_2=1}^{K} \left(\rho_{\omega,k_1}(s'_1) \cdot \rho_{\omega,k_2}(s'_2)\right) \mathcal{N}\left(\begin{bmatrix} r_1 + \gamma\mu_{\omega,k_1}(s'_1, a^*_1) \\ r_2 + \gamma\mu_{\omega,k_2}(s'_2, a^*_2) \end{bmatrix}, \Sigma_\omega(s'_1, s'_2)\right). \quad (11)$$

In practice, this implies that we are perpetually fitting a K-GMM to a K^2-GMM, at each update step. This is not to be seen as a nuisance, however. In fact, this might be seen as favorable if one envisions the process as distilling the most prominent features of the K^2-GMM down to a K-GMM, keeping the model size reasonably bounded at all times. Notably, in the case of 1-GMMs, both $\eta_\theta(s; a_1, a_2)$ and $\eta^*_\omega(s'_1, s'_2)$ have the same number of components.

4 Experimental Results

We now concretize the methodology presented in Sections 2 and 3 further with experimental results.

4.1 Joint Iterative Policy Evaluation (JIPE)

We report two minimal, fully-specified MDPs that manifest correlated state-action returns.

Shared-Randomness Bandit. A one-state, two-action MDP with per-step reward vector $R_t \sim \mathcal{N}(\mu_r, \Sigma_r)$, in the spirit of Example 1. The shared Gaussian draw at each step induces dependence between the two actions' rewards. We set $\mu_r = [0.0 \quad 0.2]^T$. The variance of the first action is 0.8, the second action is 1.0 and the covariance of the two actions is 0.6. The discount factor is 0.9. The evaluated policy plays action 2 for all time steps.

Windy Gridworld. A 3x3 gridworld environment with a leftward gust of wind, the presence of which is dictated by a Bernoulli random variable with parameter $p = 0.35$. The discount factor is 0.95. The wind perturbs the transition dynamics irrespective of the chosen action. At any state, two different actions experience the same gust, leading to dependent successive states and hence dependent returns, akin to Example 2. The evaluated policy is presented in Figure 3.

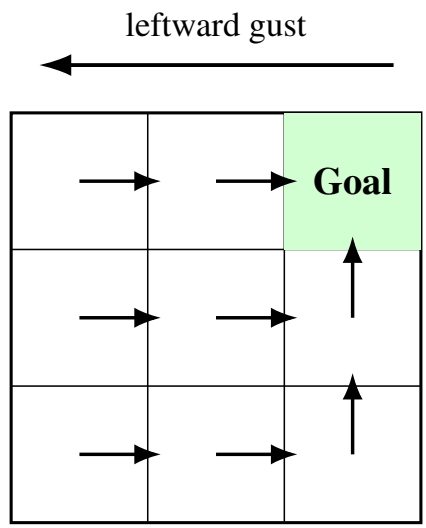

Figure 3: Deterministic policy evaluated in the windy gridworld environment with leftward gust. The gust is shared each time step between all actions.

For each setup we evaluate a fixed policy using our N-variate joint iterative policy evaluation operator from Section 2.2 to compute the means, the uncentered second moments and the cross moments to recover the covariance matrix. For the bandit environment, we derive closed-form ground truth values for these quantities and observe machine-precision agreement in terms of maximum absolute distance. For the gridworld environment, we compute the ground truth values by Monte Carlo estimation. Our proposed policy evaluation scheme matches in the order of 10^{-3} for means and 10^{-4} for covariances in terms of maximum absolute distance. The results of these experiments are presented in Tables 1 and 2. These experiments directly validate that the iterative policy evaluation scheme recovers both the means and covariances implied by the coupled dynamics and rewards.

4.2 Optimal Control with Deep Learning

Firstly, we present the learned joint distributions of near-optimal state-action returns for two states from the classic control problem of CartPole, obtained after 50K training frames. Because this is an environment with $N = 2$, we are able to plot the full joint distribution of state-action returns. Figure 1 shows plots of these distributions for the given frames.

To showcase our method's ability to learn covariance matrices, we present Figure 4, which shows three correlation matrices coefficients (covariance matrices normalized by the standard deviations) belonging to an estimate of an optimal return distribution of the game Pong after 7.5M training frames.

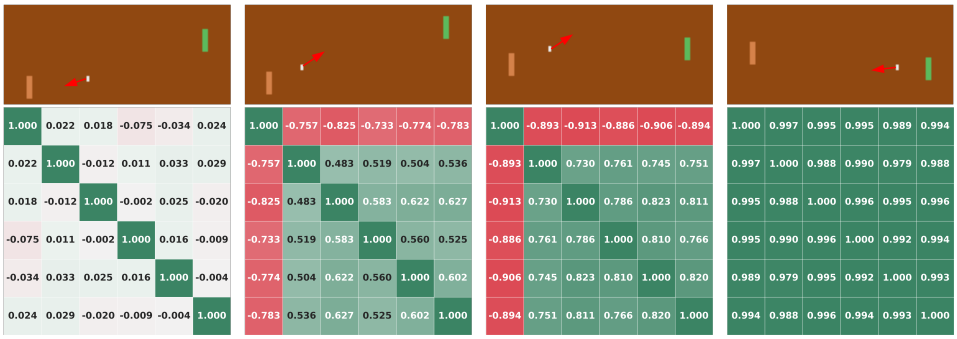

Figure 4: Three examples of covariance matrices of a near-optimal return at the shown states of the game Pong. The arrows are added by the authors to give the reader context as to where the ball is headed and are not part of the game.

On the left is what we dub a *noncritical state*. The game has just initialized, the ball is heading towards the opponent, and there is no urgency to take any action as the agent has not observed how the ball will be heading towards them. The corresponding correlation matrix shows that the returns of actions are almost completely uncorrelated. In the middle is a *critical state*. The ball has almost reached the agent but the agent is not yet in position to return it. The correlation matrix shows clear correlations and inverse correlations between the returns of actions, as taking some actions at this point will lead to conceding a score. On the right is a *post-critical state*. By this point, the agent has taken the correct actions and has full belief that they have returned the ball with a perfectly placed shot. They already know that they have scored, and any actions taken while they wait for the ball to pass the opponent's boundary have no effect on the outcome of the episode. All actions after this point are perfectly correlated, because they will lead to the same score.

All of these examples showcase the possible uses of learning the joint distribution of returns, in terms of deriving possibly safer, more explainable and interpretable policies in sequential decision-making problems.

5 Conclusion

We argued that action-wise return dependencies are intrinsic in many MDPs and developed a principled way to capture them by learning joint (rather than marginal) return distributions. Concretely, we cast the problem as a POMDP whose hidden states store coupled potential outcomes across actions, derived joint Bellman equations and a JIPE operator with convergence guarantees to the mean and second moments, and proposed a practical deep learning method that fits homoscedastic GMMs to estimate optimal joint return distributions. Empirical results on synthetic environments with known correlations and on standard control benchmarks showed that the approach recoverse accurate moments and brings into light interpretable cross-action structure.

We argued that the conventional single-action transitions of form $\tau = (s, a, r, s')$ are insufficient to identify off-diagonal moments, motivating joint obvservations of form $\tau^2 = (s, a_1, a_2, r_1, r_2, s'_1, s'_2)$ that reveal the consequences of one played action and one counterfactual at the same state. In practical settings with high-fidelity simulators or digital twins, these joint samples are feasible.

We envision that the limitations of the current work and directions of future research include but are not limited to scaling beyond second moments, exploring richer parametrizations of joint distributions such as normalizing flows or copulas, estimating joint distributions as couplings of existing acclaimed DRL methods (e.g., C51), and extending the methodology to continuous action spaces. On the control side, leveraging information on cross-action dependencies for risk-aware planning and safer and more sample-efficient exploration strategies is a promising direction.

References

[1] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. USA: John Wiley & Sons, Inc., 1st ed., 1994.

[2] A. Agarwal, N. Jiang, S. M. Kakade, and W. Sun, "Reinforcement learning: Theory and algorithms," *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, vol. 32, p. 96, 2019.

[3] R. Bellman, *Dynamic Programming*. Dover Publications, 1957.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[5] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Machine Learning*, vol. 3, no. 1, pp. 9–44, 1988.

[6] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, 1992.

[7] R. S. Sutton, "Dyna, an integrated architecture for learning, planning, and reacting," *SIGART Bull.*, vol. 2, p. 160–163, July 1991.

[8] G. A. Rummery and M. Niranjan, "On-line Q-learning using connectionist systems," CUED/F-INFENG/TR 166, Cambridge University Engineering Department, September 1994.

[9] S. J. Bradtke and A. G. Barto, "Linear least-squares algorithms for temporal difference learning," *Machine Learning*, vol. 22, no. 1, pp. 33–57, 1996.

[10] M. G. Lagoudakis and R. Parr, "Least-squares policy iteration," *J. Mach. Learn. Res.*, vol. 4, p. 1107–1149, Dec. 2003.

[11] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," in *Advances in Neural Information Processing Systems* (S. Solla, T. Leen, and K. Müller, eds.), vol. 12, MIT Press, 1999.

[12] H. v. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, p. 2094–2100, AAAI Press, 2016.

[13] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas, "Dueling network architectures for deep reinforcement learning," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, p. 1995–2003, JMLR.org, 2016.

[14] M. G. Bellemare, W. Dabney, and R. Munos, "A distributional perspective on reinforcement learning," in *Proceedings of the 34th International Conference on Machine Learning* (D. Precup and Y. W. Teh, eds.), vol. 70 of *Proceedings of Machine Learning Research*, pp. 449–458, PMLR, 06–11 Aug 2017.

[15] T. Morimura, M. Sugiyama, H. Kashima, H. Hachiya, and T. Tanaka, "Nonparametric return distribution approximation for reinforcement learning," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, (Madison, WI, USA), p. 799–806, Omnipress, 2010.

[16] T. Morimura, M. Sugiyama, H. Kashima, H. Hachiya, and T. Tanaka, "Parametric return density estimation for reinforcement learning," in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, UAI'10, (Arlington, Virginia, USA), p. 368–375, AUAI Press, 2010.

[17] W. Dabney, G. Ostrovski, D. Silver, and R. Munos, "Implicit quantile networks for distributional reinforcement learning," in *Proceedings of the 35th International Conference on Machine Learning* (J. Dy and A. Krause, eds.), vol. 80 of *Proceedings of Machine Learning Research*, pp. 1096–1105, PMLR, 10–15 Jul 2018.

[18] M. Hessel, J. Modayil, H. van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver, "Rainbow: combining improvements in deep reinforcement learning," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.

[19] W. Dabney, M. Rowland, M. G. Bellemare, and R. Munos, "Distributional reinforcement learning with quantile regression," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.

[20] D. Yang, L. Zhao, Z. Lin, T. Qin, J. Bian, and T.-Y. Liu, "Fully parameterized quantile function for distributional reinforcement learning," *Advances in neural information processing systems*, vol. 32, 2019.

[21] Y. Choi, K. Lee, and S. Oh, "Distributional deep reinforcement learning with a mixture of gaussians," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 9791–9797, 2019.

[22] R. Zhang, "Cramer type distances for learning gaussian mixture models by gradient descent," 2023.

[23] D. Freirich, T. Shimkin, R. Meir, and A. Tamar, "Distributional multivariate policy evaluation and exploration with the Bellman GAN," in *Proceedings of the 36th International Conference on Machine Learning* (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of *Proceedings of Machine Learning Research*, pp. 1983–1992, PMLR, 09–15 Jun 2019.

[24] P. Zhang, X. Chen, L. Zhao, W. Xiong, T. Qin, and T.-Y. Liu, "Distributional reinforcement learning for multi-dimensional reward functions," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, (Red Hook, NY, USA), Curran Associates Inc., 2021.

[25] M. G. Bellemare, W. Dabney, and M. Rowland, *Distributional Reinforcement Learning*. The MIT Press, 05 2023.

[26] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018.

A Proof of Theorem 1

We first state a simple lemma which is used to derive the convergence results.

Lemma 1. *Consider two non-negative sequences a_k and b_k. Assume $a_k \leq \gamma^k a_0$ and $b_{k+1} \leq a_0 B\gamma^k + \gamma^2 b_k$ for some $B > 0$ and $\gamma \in [0, 1)$. Then, $b_k \leq \gamma^{2k} b_0 + \frac{a_0 B\gamma^k}{1-\gamma}$.*

Proof. We proceed by unrolling the recurrence

$$\begin{aligned} b_{k+1} &\leq \gamma^2 b_k + a_0 B\gamma^k, \\ &\leq \gamma^2(\gamma^2 b_{k-1} + a_0 B\gamma^k) + a_0 B\gamma^k \\ &= \gamma^4 b_{k-1} + a_0 B\gamma^{k+1} + a_0 B\gamma^k. \end{aligned} \tag{12}$$

Thus, by induction we have

$$b_{k+1} \leq \gamma^{2(k+1)} b_0 + a_0 B \sum_{j=0}^{k} \gamma^{2(k-j)}\gamma^{j+1} = \gamma^{2(k+1)} b_0 + a_0 B \sum_{j=0}^{k} \gamma^{2(k-j)}\gamma^{j}. \tag{13}$$

Using a change of variable $i = k - j$ we can calculate the second geometric sum:

$$\sum_{j=0}^{k} \gamma^{2(k-j)}\gamma^{j} = \sum_{i=0}^{k} \gamma^{2i}\gamma^{k-i} = \gamma^k \sum_{i=0}^{k} \gamma^{i} = \gamma^k \cdot \frac{1-\gamma^{k+1}}{1-\gamma}. \tag{14}$$

Using this in the previous equation furnishes the proof. □

We now state the proof of the main result. We re-state the theorem for convenience.

Theorem 1 (Convergence of N-variate joint iterative policy evaluation)**.** *Suppose Assumptions 1 and 2 hold. Consider the N-variate joint iterative policy evaluation scheme in* (9)*. Let $\mu^k(s)$ and $\bar{\Sigma}^k(s)$ denote the mean and the uncentered matrix of second moments recovered from M^k. Then,*

$$\|\mu^k(s) - \mu^\pi(s)\|_\infty = \mathcal{O}\left(\gamma^k\right), \quad \|\bar{\Sigma}^k(s) - \bar{\Sigma}^\pi(s)\|_\infty = \mathcal{O}\left(\frac{\gamma^k}{1-\gamma}\right). \tag{15}$$

Proof. We adopt and strengthen an argument from Chapter 8 in [25]. We will first define the following semi-norms

$$\begin{aligned} \|M\|_{\infty,\mu} &= \sup_{(s,a)} |M_\mu(s,a)| \\ \|M\|_{\infty,\sigma} &= \sup_{(s,a)} |M_\sigma(s,a)| \\ \|M\|_{\infty,c} &= \sup_{(s,a,j)} |M_c(s,a)_j| \end{aligned} \tag{16}$$

Next, we demonstrate that the second-order N-variate joint Bellman operator $\mathcal{T}^\pi_{2,N}$ is a contraction with respect to $\|\cdot\|_{\infty,\mu}$ with constant γ. To see this, we remark that by the definition of M_μ, M, and $\|\cdot\|_{\infty,\mu}$ we have that

$$(\mathcal{T}^\pi_{2,N} M)_\mu = \mathcal{T}^\pi M_\mu, \tag{17}$$

where $\mathcal{T}^\pi \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ is the usual Bellman operator. Furthermore, note that $\|M\|_{\infty,\mu} = \|M_\mu\|_\infty$. Thus,

$$\begin{aligned} \|\mathcal{T}^\pi_{2,N} M - \mathcal{T}^\pi_{2,N} M'\|_{\infty,\mu} &= \|(\mathcal{T}^\pi_{2,N} M)_\mu - (\mathcal{T}^\pi_{2,N} M')_\mu\|_\infty \\ &= \|\mathcal{T}^\pi M_\mu - \mathcal{T}^\pi M'_\mu\|_\infty \\ &\leq \gamma\|M_\mu - M'_\mu\|_\infty \\ &= \gamma\|M - M'\|_{\infty,\mu} \end{aligned} \tag{18}$$

where we used the γ-contraction of $\mathcal{T}^\pi$ with respect to $\|\cdot\|_\infty$. Now recall, by linear convergence of the regular Bellman update $M^{k+1}_\mu = \mathcal{T}^\pi M^k_\mu$, we have

$$\|M^k - M^\pi\|_{\infty,\mu} = \|M^k_\mu - M^\pi_\mu\|_\infty \leq \gamma^k \|M^0_\mu - M^\pi_\mu\|_\infty = \gamma^k \|M^0 - M^\pi\|_{\infty,\mu}. \tag{19}$$

This result establishes that the iterative policy evaluation scheme in (9) which repeatedly applies the second order N-variate joint Bellman operator $\mathcal{T}^{\pi}_{2,N}$ converges linearly to the mean of the N-variate joint return distribution $\eta^{\pi}(s)$.

To prove the rest of the statement, recall that for any (s, a), by Assumption 2, $|\mathbb{E}[R(s, a)]| \leq \max\{|r_{\min}|, |r_{\max}|\} \leq B$ and $|\mathbb{E}[R(s, a)^2]| \leq \max\{|r_{\min}|^2, |r_{\max}|^2\} \leq B$ for some $B > 0$.

Furthermore, by the definition of M_{μ}, M_{σ}, M, $\|\cdot\|_{\infty,\mu}$, and $\|\cdot\|_{\infty,\mu}$, for all (s, a), we have

$$\begin{aligned}
|(\mathcal{T}^{\pi}_{2,N}M)(s,a)_2 - (\mathcal{T}^{\pi}_{2,N}M')(s,a)_2| &\leq 2B\gamma \left| \sum_{(s',a')\in\mathcal{S}\times\mathcal{A}} P(s' \mid s,a)\pi(a' \mid s')(M - M')(s',a')_1 \right| \\
&\quad + \gamma^2 \left| \sum_{(s',a')\in\mathcal{S}\times\mathcal{A}} P(s' \mid s,a)\pi(a' \mid s')(M - M')(s',a')_2 \right| \\
&\leq 2B\gamma \|M_{\mu} - M'_{\mu}\|_{\infty} + \gamma^2 \|M_{\sigma} - M'_{\sigma}\|_{\infty} \\
&= 2B\gamma \|M - M'\|_{\infty,\mu} + \gamma^2 \|M - M'\|_{\infty,\sigma}
\end{aligned} \tag{20}$$

Hence,

$$\|\mathcal{T}^{\pi}_{2,N}M - \mathcal{T}^{\pi}_{2,N}M'\|_{\infty,\sigma} \leq 2B\gamma \|M - M'\|_{\infty,\mu} + \gamma^2 \|M - M'\|_{\infty,\sigma}. \tag{21}$$

Similarly, we can establish a recursive inequality for the cross covariance M_c. In particular, for all $(s, a, j) \in \mathcal{S} \times \mathcal{A} \times \{3, \ldots, N+1\}$,

$$\begin{aligned}
|(\mathcal{T}^{\pi}_{2,N}M)(s,a)_j - (\mathcal{T}^{\pi}_{2,N}M')(s,a)_j| &\leq B\gamma \left| \sum_{(s'_1,a'_1)\in\mathcal{S}\times\mathcal{A}} P(s'_1 \mid s,a)\pi(a'_1 \mid s'_1)(M - M')(s'_1,a'_1)_1 \right| \\
&\quad + B\gamma \left| \sum_{(s'_2,a'_2)\in\mathcal{S}\times\mathcal{A}} P(s'_2 \mid s,a_j)\pi(a'_2 \mid s'_2)(M - M')(s'_2,a'_2)_1 \right| \\
&\quad + \gamma^2 \left| \sum_{(s',a')\in\mathcal{S}\times\mathcal{A}} P(s' \mid s,a)\pi(a' \mid s')(M - M')(s',a')_j \right| \\
&\leq 2B\gamma \|M_{\mu} - M'_{\mu}\|_{\infty} + \gamma^2 \|M_c - M'_c\|_{\infty} \\
&= 2B\gamma \|M - M'\|_{\infty,\mu} + \gamma^2 \|M - M'\|_{\infty,c}
\end{aligned} \tag{22}$$

where a_j denotes the action used to calculate the cross covariance term for (s, a) which is stored in $M_c(s, a)_j$, and we use the definition of the joint MDP, notably the fact that $P'(\cdot \mid x, a) := C_P(s_a) \times C_R(s_a)$, to bound the term in the bound (that is, the next state transition is dictated by a, not a_j). Hence,

$$\|\mathcal{T}^{\pi}_{2,N}M - \mathcal{T}^{\pi}_{2,N}M'\|_{\infty,c} \leq 2B\gamma \|M - M'\|_{\infty,\mu} + \gamma^2 \|M - M'\|_{\infty,c}. \tag{23}$$

Thus, by invoking Lemma 1, one can readily establish

$$\begin{aligned}
\|M^k - M^{\pi}\|_{\infty,\sigma} &\leq \gamma^{2k} \|M^0 - M^{\pi}\|_{\infty,\sigma} + \frac{2\|M^0 - M^{\pi}\|_{\infty,\mu} B\gamma^k}{1-\gamma} \\
\|M^k - M^{\pi}\|_{\infty,c} &\leq \gamma^{2k} \|M^0 - M^{\pi}\|_{\infty,c} + \frac{2\|M^0 - M^{\pi}\|_{\infty,\mu} B\gamma^k}{1-\gamma}
\end{aligned} \tag{24}$$

These results establish that the iterative policy evaluation scheme in (9) which repeatedly applies the 2nd order N-variate joint Bellman operator $\mathcal{T}^{\pi}_{2,N}$ converges linearly to the second moment (shifted covariance) of the N-variate joint return distribution $\eta^{\pi}(s)$. □