Learning Temporal Distances: Contrastive Successor Features Can Provide a Metric Structure for Decision-Making

Vivek Myers [1] **Chongyi Zheng** [2] **Anca Dragan** [1] **Sergey Levine** [1] **Benjamin Eysenbach** [2]

Abstract

Temporal distances lie at the heart of many algorithms for planning, control, and reinforcement learning that involve reaching goals, allowing one to estimate the transit time between two states. However, prior attempts to define such temporal distances in stochastic settings have been stymied by an important limitation: these prior approaches do not satisfy the triangle inequality. This is not merely a definitional concern, but translates to an inability to generalize and find shortest paths. In this paper, we build on prior work in contrastive learning and quasimetrics to show how successor features learned by contrastive learning (after a change of variables) form a temporal distance that does satisfy the triangle inequality, even in stochastic settings. Importantly, this temporal distance is computationally efficient to estimate, even in high-dimensional and stochastic settings. Experiments in controlled settings and benchmark suites demonstrate that an RL algorithm based on these new temporal distances exhibits combinatorial generalization (i.e., "stitching") and can sometimes learn more quickly than prior methods, including those based on quasimetrics.

1 Introduction

Graph search is one of the most important ideas in CS, being introduced in almost every introductory CS class. However, classes often overlook a key assumption: that transitions be deterministic. With deterministic transitions, shortest-path lengths obey the triangle inequality. This property, encoded into dynamic programming algorithms, allows one to search over an exponential number of paths and find the shortest in polynomial time. This property also allows for generalization, finding new paths unseen in the data. However, in graphs (or, more generally, Markov processes) with stochastic transitions, it is unclear how to define the distance between two states such that this distance obeys the triangle inequality.

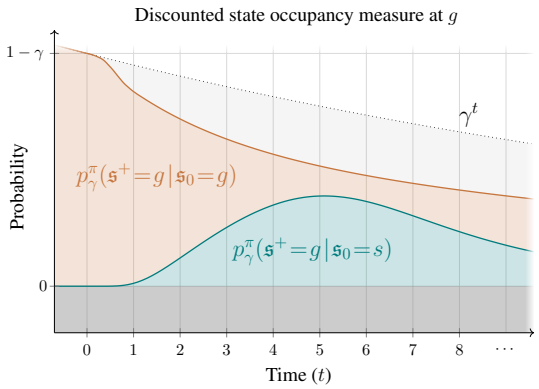
A reasonable solution for goal-reaching is to learn *temporal distances*, which reflect some notion of transit time between states (Venkattaramanujam et al., 2019; Savinov et al., 2018; Durugkar et al., 2021; Ma et al., 2023b; Hartikainen et al., 2019). However, simply defining distances as hitting times breaks down in stochastic settings, as shown in prior work (Akella et al., 2023). Stochastic settings are ubiquitous in real-world problems: from autonomous vehicles navigating around drunk bar-goers, to healthcare systems rife with unobservable features. Indeed, many advances in ML over the last decade have been predicated on probabilistic models (e.g., diffusion models, VAEs), so it seems rather anachronistic that an important control primitive (the notion of distances) is not well defined in a probabilistic sense.

The key challenge is that the prior notions of temporal distance break down in stochastic settings. Nonetheless, the triangle inequality holds great appeal as a strong inductive bias for learning temporal distances: the distance between two states should be less than the length of a path that goes through a particular waypoint state. Indeed, prior work has aimed to exploit this notion by learning "temporal distance metrics" that can broadly generalize from less data.

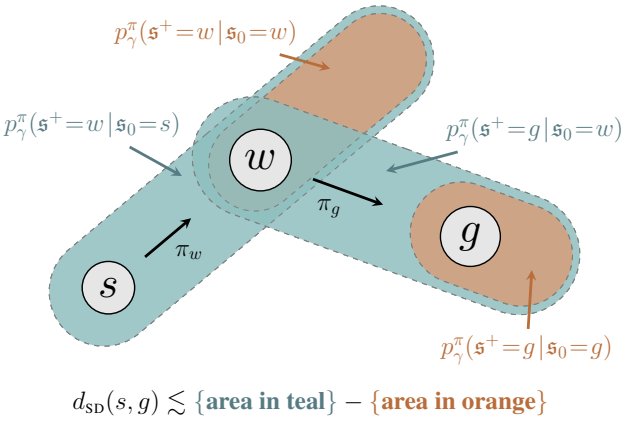
The starting point for our work is to think about distances probabilistically. Because the dynamics may be stochastic, the number of steps it takes to traverse between two states is not a definite quantity, but rather a random variable. To estimate the (long-term) probabilities of transiting between two states, we will build on prior temporal contrastive learning (van den Oord et al., 2019; Eysenbach et al., 2022), a popular and stable class of time series representation learning methods. Intuitively, these methods learn representations from time series data so that observations that occur nearby in time are given similar representations. Importantly, contrastive methods based on NCE and infoNCE have a probabilistic interpretation, making them ripe for application to stochastic environments. Like prior work (Eysenbach et al., 2022; 2023), we account for the arrow of time (Popper, 1956) by using asymmetric repre-

[1]University of California, Berkeley [2]Princeton University. Correspondence to: Vivek Myers <vmyers@berkeley.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.



(a) Starting at state s, we visualize the (discouted) probability of reaching state g after exactly t steps (teal). The sum of these probabilities (■ area) is the probability of reaching state g at some point in the future. Our method defines the distance between states s and g as the difference in these shaded areas (■ area - ■ area).



(b) Our proposed distance obeys the triangle inequality. Starting at state s, we look at the distribution over future states (■ area) and subtract off those states that the policy would reach starting from w (■ area). Our distance is defined as the difference in these areas, $d_{\text{SD}}(s,w) \triangleq$ ■ − ■.

Figure 1. An overview of our theoretical distance construction as well as the concrete implementation with metric distillation.

sentations, allowing the learned representations reflect the fact that (say) climbing up a mountain is more difficult from sliding back down. The representations learned by these temporal contrastive learning methods do not themselves satisfy the triangle inequality. However, we prove that a simple change of variables results in representations that do satisfy the triangle inequality. Intuitively, this change of variables corresponds to subtracting off the "distance" between a state and itself. Note that because the representations are asymmetric (see above), this extra "distance" is not zero, but rather corresponds to the likelihood of returning to the current state at some point in the future.

The main contribution of this paper is to propose a notion of temporal distance that provably satisfies the triangle inequality, even in stochastic settings. Our constructed temporal distance is easy to learn – simply take the features from (temporal) contrastive learning and perform a change of variables – no additional training required! After introducing and analyzing our proposed temporal distance, we demonstrate an application of our temporal distance to goal-conditioned reinforcement learning, using the distance function as a value function. We use a carefully controlled synthetic benchmark to test properties such as combinatorial generalization, temporal generalization, and finding shortest paths; our results here show that the proposed distance has appealing properties that prior methods lack. We also show that the RL method based on our distances can scale to 111-dimensional locomotion tasks, where it is competitive with prior methods on a parameter-adjusted basis.

2 Related Work

Our work builds on prior work in learning temporal distances and contrastive representation learning.

2.1 Learning distances

Within any Markov decision process (MDP), there is an intuitive notion of "distance" between states as the difficulty of transitioning between them. There are many seemingly reasonable definitions for distance a priori: likelihood of reaching the goal at a particular time, expected time to reach the goal, likelihood of ever reaching the goal, etc. (under some policy). The key mathematical structure for a distance to be useful for reaching goals is that it must satisfy the triangle inequality $d(a,c) \leq d(a,b) + d(b,c)$: being able to go from $a \to b$ and from $b \to c$ means going from $a \to c$ can be no harder than both of the aforementioned steps. Such a distance is called a *metric* over the state space if it is symmetric and more generally a *quasimetric* (Paluszyński & Stempak, 1931).

While prior work on bisimulations (Hansen-Estruch et al., 2022; Ferns et al., 2011) use a reward function to construct such a distance, our aim will be to define a notion of distance that does not require a reward function.

For the correct choice of distance, learning a goal-conditioned value function will correspond to selecting a distance metric that best enables goal reaching. Such a distance can then be learned with an architecture that directly enforces metric properties, e.g., Euclidean distance, metric residual network (MRN), interval quasimetric estimator (IQE), etc. (Wang & Isola, 2022a;b; Liu et al., 2023). Since the space of value (quasi)metrics imposes a strong induction

bias over value functions, using the right metric architecture can enable better combinatorial and temporal generalization *without* requiring additional samples (Wang et al., 2023).

In deterministic MDPs, these notions of distance all coincide with distance $d(s, g)$ being proportional to the (minimum) amount of time needed to reach the goal g when starting in state s. Approaches like Quasimetric RL (Wang et al., 2023; Liu et al., 2023) learn this notion of distance, allowing optimal goal reaching in deterministic MDPs. In general MDPs, alternative notions of distance are required (Akella et al., 2023; N'Guyen et al., 2013; Ma et al., 2023b; N'Guyen et al., 2013; Lan et al., 2021; Hejna et al., 2023). Existing approaches are often limited by assumptions such as symmetry or fail to satisfy metric properties. Our contribution is to construct a general formulation for a quasimetric over MDPs that can be easily learned from discounted state occupancy measures.

2.2 Contrastive Representations

Contrastive learning has seen widespread adoption for learning to represent time series (van den Oord et al., 2019; Mikolov et al., 2013). These representations can be trained to approximate mutual information without requiring labels or reconstruction (Gutmann & Hyvärinen, 2010; Mazoure et al., 2020; Wu et al., 2021; van den Oord et al., 2019; Gutmann & Hyvarinen, 2012), and are useful for learning self-supervised representations across broad application areas (Radford et al., 2021; Sermanet et al., 2017; Qian et al., 2021; Chen et al., 2020; 2021b; Saunshi et al., 2019; Wang & Isola, 2020; Saunshi et al., 2019).

Within RL, contrastive learning can be used for goal-conditioned control as successor features (Barreto et al., 2017; Eysenbach et al., 2022; 2023). Approaches that use contrastive representations for control are typically limited in combinatorial and temporal generalization since they do not bootstrap value functions (Zheng et al., 2023). Unlike past approaches that use contrastive learning for decision-making, we show that these generalization capabilities *can* be obtained from contrastive successor features by imposing an additional metric structure.

2.3 Goal-conditioned reinforcement learning (GCRL)

Goal-reaching presents an attractive formulation for learning useful behaviors in unsupervised RL settings (Laird et al., 1987; Kaelbling, 1993). Recent advances in deep reinforcement learning have renewed interest in this problem as many real-world offline and online RL problems lack clear reward signals (Andrychowicz et al., 2017; Eysenbach et al., 2022; Park et al., 2023a; Yang et al., 2023; Ghosh et al., 2019). GCRL methods can learn goal-conditioned policies (Yang et al., 2022; Ghosh et al., 2021), value functions (Eysenbach et al., 2021; Ghosh et al., 2023), and/or representations that enable goal-reaching (Eysenbach et al., 2022; Zheng et al., 2023; Ma et al., 2023b). Approaches that recover goal-conditioned policies can also enable additional capabilities like planning (Fang et al., 2023; Chane-Sane et al., 2021), skill discovery (Mendonca et al., 2021; Park et al., 2023b) and interface with other forms of task specification like language (Ma et al., 2023a; Myers et al., 2023; Shah et al., 2023; Black et al., 2023; Touati & Ollivier, 2021).

These GCRL techniques typically require bootstrapping with a learned value function, which can be costly and unstable, or struggle with long-horizon combinatorial and temporal generalization (Ghugare et al., 2024). Our approach avoids both of these shortcomings by learning a distance metric that can implicitly combine behaviors without bootstrapping or making any assumptions about the environment dynamics.

3 General distances for goal-reaching

In this section, we introduce a novel distance metric for goal-reaching in controlled Markov processes. We show that this distance is a quasimetric, i.e., a metric that relaxes the assumption of symmetry. In the subsequent section (4), we show that this distance construction can enable additional generalization capabilities through a choice of model parameterization for temporal contrastive learning.

3.1 Preliminaries

We consider a discrete *controlled Markov process* M consisting of states $s \in \mathcal{S}$, actions $a \in \mathcal{A}$, dynamics $P(s' \mid s, a)$, initial state distribution $p_0(\mathfrak{s}_0 = s_0)$, and a discount $\gamma \in (0, 1)$.

By augmenting M with the reward for any fixed goal $g \in \mathcal{S}$, which we define as

$$r_g(s) = (1 - \gamma)\, \delta_g(s),$$

where $\delta_g(s) = \begin{cases} 1 \text{ if } s{=}g \\ 0 \text{ otherwise} \end{cases}$ is the Kronecker delta, we can extend M to a goal-dependent Markov decision process M_g. Denote by Π the (compact) set of stationary of policies $\pi(a \mid s)$ on M. We also define $\Pi_{\text{NM}} \supset \Pi$ to be the set of non-Markovian policies $\pi(a_t \mid s_0 \dots s_t)$. We can then derive the optimal goal-conditioned value function,

$$V_g^*(s) = \max_{\pi \in \Pi} p_\gamma^\pi(\mathfrak{s}^+{=}g \mid \mathfrak{s}_0{=}s), \tag{1}$$

where the *discounted state occupancy measure* p_γ^π is defined as the discounted distribution over future states $\mathfrak{s}^+$,

$$p_\gamma^\pi(\mathfrak{s}^+{=}s' | \mathfrak{s}_0{=}s) = (1-\gamma) \sum_{k=0}^{\infty} \gamma^k p^\pi(\mathfrak{s}_k{=}s' | \mathfrak{s}_0{=}s)$$
$$\text{where } p^\pi(\mathfrak{s}_{t+1}{=}s' | \mathfrak{s}_t = s){=} \sum_{a \in \mathcal{A}} \pi(a|s) P(s'|s,a). \tag{2}$$

i.e., the distribution of $\mathfrak{s}^+ \triangleq \mathfrak{s}_K$ for $K \sim \text{Geom}(1-\gamma)$.

Here, $\mathfrak{s}_t$ denotes the state at time t as a random variable, and $\mathfrak{s}^+$ denotes the state at a geometrically distributed time in the future. When needed, under a policy π, we will additionally use the notation $\mathfrak{a}_t$ and $\mathfrak{a}^+$ to denote actions as random variables, defined analogously to $\mathfrak{s}_t$ and $\mathfrak{s}^+$.

Since (1) is the optimal value function corresponding to the reward r_g, there will always be a stationary optimal goal-reaching policy $\pi^g \in \Pi$ that attains the max in (1).

We can additionally view the setting of an *uncontrolled Markov process* (i.e., a Markov chain) as a special case of controlled Markov processes where there is a single action $\mathcal{A} = \{a\}$ with a fixed policy $\Pi = \{\pi\}$.

To reason about the effects of actions, we can also consider the natural generalization of the *successor state-action distribution*, which is the distribution over future states and actions s', a' given that action a is taken in state s under π:

$$p^\pi_\gamma(\mathfrak{s}^+ = g, \mathfrak{a}^+ = a' \,|\, \mathfrak{s}_0 = s, \mathfrak{a}_0 = a) = (1-\gamma)\delta_{s,a}(g,a') + (1-\gamma)\gamma\Big[\sum_{k=0}^{\infty}\sum_{s'\in\mathcal{S}} \gamma^k p^\pi\big(\mathfrak{s}_k = g \mid \mathfrak{s}_0 = s'\big)\pi(a' \mid g)P(s' \mid s, a)\Big]. \quad (3)$$

Finally, we recall the definition of a quasimetric space:

Definition 3.1. *A quasimetric on S is a function $d : S \times S \to \mathbb{R}$ satisfying the following for any $x, y, z \in S$.*

Positivity: $d(x, y) \geq 0$

Identity: $d(x, y) = 0 \iff x = y$

Triangle inequality: $d(x, z) \leq d(x, y) + d(y, z)$

3.2 Our Proposed Temporal Distance

With these definitions in place, we can now define the proposed temporal distance. We will start by describing a "strawman" approach, and then proceed with the full method.

Motivated by prior work on successor representations (Dayan, 1993) and self-predictive representations (Schwarzer et al., 2020; Ni et al., 2024), a candidate temporal distance is to directly use the critic function from temporal contrastive learning. When positive examples are sampled from the discounted state occupancy measure, this critic has the following form:

$$-d(s,g) = \log\left(\frac{p^\pi_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s)}{p(g)}\right). \quad \text{(not a quasimetric)}$$

However, a distance defined in this way does not satisfy the identity property of the quasimetric; namely, the distance between a state and itself can be non-zero. Our solution is to subtract off the "extra distance" between a state and itself, $\tilde{d}(s,g) = d(s,g) - d(g,g)$. Doing this results in the proposed temporal distance that we propose in this paper.

We now proceed with our main definition, which is a temporal distance that obeys the triangle inequality (and is a quasimetric) even in stochastic settings. We provide two definitions, one for controlled Markov processes and one for (uncontrolled) Markov processes:

Definition 3.2. *We define the **successor distance** for a controlled Markov processes by:*

$$d_{\text{SD}}(s,g) \triangleq \min_{\pi\in\Pi} \log\left(\frac{p^\pi_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g)}{p^\pi_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s)}\right), \quad (4)$$

As a special case for an uncontrolled Markov process, we can define:

$$d_{\text{SD}}(s,g) \triangleq \log\left(\frac{p_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g)}{p_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s)}\right). \quad (5)$$

To use these distances for control in model-free settings, we can extend this notion to include actions, yielding a distance over $\mathcal{S} \times \mathcal{A}$.

Definition 3.3. *We define the **successor distance with actions** for a controlled Markov process by:*

$$d_{\text{SD}}\big((s,a),(g,a')\big) \triangleq \min_{\pi\in\Pi}\left(\log \frac{p^\pi_\gamma(\mathfrak{s}^+ = g, \mathfrak{a}^+ = a' \,|\, \mathfrak{s}_0 = g, \mathfrak{a}_0 = a')}{p^\pi_\gamma(\mathfrak{s}^+ = g, \mathfrak{a}^+ = a' \,|\, \mathfrak{s}_0 = s, \mathfrak{a}_0 = a)}\right). \quad (6)$$

We make two brief lemmas about this definition; the proofs can be found in Appendix C Within $\mathcal{S} \times \mathcal{A}$, we can also say:

Lemma 3.1. *$d_{\text{SD}}\big((s,a),(s',a')\big)$ is independent of a' when $s \neq s'$.*

In light of this independence, we denote $d_{\text{SD}}\big(s,a,s'\big) \triangleq d_{\text{SD}}\big((s,a),(s',a')\big)$ where applicable. Selecting actions that minimizes this distance corresponds to policy improvement:

Lemma 3.2. *Selecting actions to minimize the successor distance is equivalent to selecting actions to maximize the (scaled and shifted) Q-function:*

$$-d_{\text{SD}}(s,a,g) = \tfrac{1}{p_g(g)}Q(s,a,g) + c_\psi(g) \implies \arg\max_a d_{\text{SD}}(s,a,g) = \arg\max_a Q(s,a,g).$$

Geometric interpretation. Before proceeding to prove that this distance construction obeys the triangle inequality and the other quasimetric properties (Section 3.3), we provide intuition for this distance. We visualize this distance construction in Figure 1 **(b)**. The distribution over states visited starting at s ($p^\pi_\gamma(\mathfrak{s}^+ = w \,|\, \mathfrak{s}_0 = s)$) is shown as the teal region; while states visited starting at w ($p^\pi_\gamma(\mathfrak{s}^+ = w \,|\, \mathfrak{s}_0 = w)$)

is shown as the orange region. Our proposed distance metric is the difference in the areas of these regions (■ − ■). The theoretical results in the next section prove that this difference is always non-negative. Zooming out to look at the s, w, and g together, we see that these set differences obey the triangle inequality – the area between s and g is smaller than the areas between s and w and between w and g. Concrete examples to build intuition for these definitions and results are presented in Appendix H.

Hitting times as a special case. To provide additional intuition into our construction, we consider a special case; the subsequent section shows that the proposed distance is a valid quasimetric in much broader settings. In this special case, consider a controlled Markov process where the agent can remain at a state indefinitely. This assumption means that the $p^\pi_\gamma(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}g) = 1$, so the proposed distance metric can be simplified to $d(s,g) = -\log p^\pi_\gamma(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s)$. This assumption also means that the hitting time of g from s has a deterministic value, which we will call $H(s,g)$. Thus, we can write the discounted state occupancy measure as $p^\pi_\gamma(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s) = \gamma^{H(s,g)}$, so the proposed distance metric is equivalent to the hitting time: $d(s,g) = H(s,g)$. Importantly, and unlike prior work, our proposed distance continues to be a quasimetric outside of this special case, as we prove in the following section.

3.3 Theoretical results

Before proving this distance is a quasimetric over $\mathcal{S}$, we provide a helper lemma relating the difficulty of reaching a goal through a waypoint to the difficulty without the waypoint. The key insight we use here is that the notion of a hitting time can be generalized to represent distances in terms of discounted state occupancies.

Lemma 3.3. *For any $s, w, g \in \mathcal{S}$, $\pi \in \Pi$,*

$$\max_{\pi' \in \Pi}\left[\frac{p^{\pi'}_\gamma(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}w)p^\pi_\gamma(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}s)}{p^\pi_\gamma(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}w)}\right] \le \max_{\pi' \in \Pi} p^{\pi'}_\gamma(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s).$$

The proof is in Appendix D. This lemma is the key to proving our main result:

Theorem 3.4. *d_{SD} is a quasimetric over $\mathcal{S}$, satisfying the triangle inequality and other properties from Definition 3.1.*

The proof is in Appendix E. Compared with prior work (Wang et al., 2023), our result extends to stochastic settings; we will empirically compare to this and other prior methods in Section 5.

To make this result applicable to settings with unknown dynamics or without actions, we note the following corollaries

Corollary 3.4.1. *d_{SD} is a quasimetric over $\mathcal{S} \times \mathcal{A}$.*

Corollary 3.4.2. *d_{SD} is a quasimetric over an uncontrolled Markov process as in Eq. (5).*

See Appendix E for discussion of these results.

4 Using our Temporal Distance for RL

In this section we describe an application of our proposed temporal distance to goal-conditioned reinforcement learning. The main challenge in doing this will be (1) estimating the successor distance defined in Eq. (4), and (2) doing so with an architecture that respects the quasimetric properties. Once learned, we will use the successor distance as a value function for training a policy.

To introduce our methods, Section 4.1 will first discuss how contrastive learning *almost* estimates the successor distance. We will then introduce two variants of our method, Contrastive Metric Distillation (CMD). The first method (CMD 1-step, Section 4.2) will acquire the successor distance by applying contrastive learning with an energy function that is the difference of two other functions. The second method (CMD 2-step, Section 4.3) will acquire the successor distance by taking the features from contrastive learning and distilling those features into a quasimetric architecture. In both cases, we then use the learned successor distance to train a goal-conditioned policy.

We emphasize that the key contribution here is the mathematical construct of *what* constitutes a temporal distance, not that we use a certain architecture to represent this temporal distance. Practically, we will use the Metric Residual Network (MRN) architecture (Liu et al., 2023) in our implementation. Pseudocode for the full algorithms (both one-step and two-step) is provided in Algorithms 1 and 2. We highlight the differences between the two methods in orange for clarity.

Algorithm 1: 1-step Contrastive Metric Distillation (CMD-1)

input: batch size B, number of iterations T
initialize potential ψ, quasimetric ϕ, and policy μ parameters
define $f_\theta(s,a,g) \triangleq c_\psi(g) - d_\phi(s,a,g)$
for $t = 1 \dots T$ **do**
  sample $\{(s_i, a_i) \sim p_s\}_{i=1}^B$
  sample $\{(g_i, a'_i) \sim p^\pi_\gamma(\mathfrak{s}^+{=}g_i\,|\,\mathfrak{s}_0{=}s_i, a_i)\}_{i=1}^B$
  $\phi \leftarrow \phi - \alpha\nabla_\phi\left[\mathcal{L}^{\text{c}}_{\phi,\psi}\left(\{s_i,a_i\},\{g_i\}\right)\right]$ (Eqs. 8,12)
  $\psi \leftarrow \psi - \alpha\nabla_\psi\left[\mathcal{L}^{\text{c}}_{\phi,\psi}\left(\{s_i,a_i\},\{g_i\}\right)\right]$ (Eqs. 8,12)
  $\mu \leftarrow \mu - \alpha\nabla_\mu\left[\mathcal{L}^{\pi}_{\mu}(\{s_i,a_i\},\{g_i,a'_i\})\right]$ (Eq. 19)
end for
output π_μ

4.1 Building block: contrastive learning

Both of our proposed methods will use contrastive learning as a core primitive, so we start by discussing how we use

Algorithm 2: 2-step Contrastive Metric Distillation (CMD-2)

input: batch size B, number of iterations T
initialize representations ϕ, ψ, and policy parameters μ
initialize quasimetric $\hat{\theta}$, margin λ
define $f_\theta(s,a,g) \triangleq \phi(s,a)^T\psi(g)$
for $t = 1 \dots T$ **do**
  sample $\{(s_i, a_i) \sim p_s\}_{i=1}^B$
  sample $\{(g_i, a_i') \sim p_\gamma^\pi(\mathfrak{s}^+ = g_i \,|\, \mathfrak{s}_0 = s_i, a_i)\}_{i=1}^B$
  $\phi \leftarrow \phi - \alpha\nabla_\phi\big[\mathcal{L}^{\mathsf{c}}_{\phi,\psi}\left(\{s_i,a_i\},\{g_i\}\right)\big]$ (Eqs. 8,16)
  $\psi \leftarrow \psi - \alpha\nabla_\psi\big[\mathcal{L}^{\mathsf{c}}_{\phi,\psi}\left(\{s_i,a_i\},\{g_i\}\right)\big]$ (Eqs. 8,16)
  $\mu \leftarrow \mu - \alpha\nabla_\mu\big[\mathcal{L}^{\pi}_{\mu}(\{s_i,a_i\},\{g_i,a_i'\})\big]$ (Eq. 19)
  $\hat{\theta} \leftarrow \theta - \alpha\nabla_{\hat{\theta}}\big[\mathcal{L}^{d}_{\hat{\theta},\phi,\psi}\big(\{s_i,a_i\},\{g_i,a_i'\}\big)\big]$ (Eq. 17)
  $\lambda \leftarrow \lambda + \alpha\big(\mathcal{C}_{\hat{\theta}}(\{s_i,a_i\},\{g_i,a_i'\}) - \varepsilon^2\big)$ (Eq. 17)
end for
output π_μ

contrastive learning to learn an energy function $f_\theta(s,a,g)$, and the relationship between that energy function and the desired successor distance.

Following prior work (Eysenbach et al., 2022), we will apply contrastive learning to learn an energy function $f_\theta(s,a,g)$ that assigns high scores to (s,a,g) triplets from the same trajectory, and low scores to triplets where the goal g is unlikely to be visited at some point after the state-action (s,a) pair. Let $p_{sa}(s,a)$ be a marginal distribution over state-action pairs, and let $p_g(g) = \sum_{s\in\mathcal{S}} p_s(s) p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s)$ be the corresponding marginal distribution over future states. Contrastive learning learns the energy function by sampling pairs of state-action (s,a) and goals g from the joint distribution $s_i, a_i, g_i \sim p_\gamma^\pi(\mathfrak{s}^+ = g_i \,|\, \mathfrak{s}_0 = s_i, a_i) p_{sa}(s_i, a_i)$. We will use the symmetrized infoNCE loss function (without re-substitution) (van den Oord et al., 2019; Sohn, 2016), which provides the following objective:

$$\min_\theta \mathbb{E}_{\{s_i,a_i,g_i\}_{i=1}^B} \mathcal{L}^{\mathsf{c}}_\theta\big(\{s_i,a_i\},\{g_i\}\big). \tag{7}$$

given the forward and backward classification losses:

$$\mathcal{L}^{\mathsf{c}}_\theta = \mathcal{L}^{\text{fwd}}_\theta + \mathcal{L}^{\text{bwd}}_\theta \tag{8}$$

$$\mathcal{L}^{\text{fwd}}_\theta\big(\{s_i,a_i\},\{g_i\}\big) = \sum_{i=1}^B \log\left(\frac{e^{f_\theta(s_i,a_i,g_i)}}{\sum_{j=1}^B e^{f_\theta(s_i,a_i,g_j)}}\right)$$

$$\mathcal{L}^{\text{bwd}}_\theta\big(\{s_i,a_i\},\{g_i\}\big) = \sum_{i=1}^B \log\left(\frac{e^{f_\theta(s_i,a_i,g_i)}}{\sum_{j=1}^B e^{f_\theta(s_j,a_j,g_i)}}\right).$$

We highlight the indices i and j for clarity. As the batch size B becomes large, the optimal critic parameters θ^* then satisfy (Ma & Collins, 2018; Poole et al., 2019)

$$f_{\theta^*}(s,a,g) = \log\left(\frac{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s, a)}{C \cdot p_g(g)}\right), \tag{9}$$

where C is a free parameter. Finally, note that we can represent the successor distance (4) as the *difference* of this optimal critic evaluated on two different inputs:

$$f_{\theta^*}(g,a,g) - f_{\theta^*}(s,a,g) = \frac{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g, a)}{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s, a)}. \tag{10}$$

The next two section present practical methods for representing this difference, either via (1) a special parametrization of this critic (Section 4.2) or (2) distillation (Section 4.3).

4.2 One-step distillation (CMD 1-step):

In this section, we describe how to *directly* learn the successor distance using an architecture that is guaranteed to satisfy the triangle inequality and other quasimetric properties.

The key idea is to apply the contrastive learning discussed in the prior section to a particular parametrization of the energy function, so that the difference in Eq. (10) is represented as a single quasimetric network. We start by noting that the function learned by contrastive learning (Eq. 9) can be decomposed into the successor distance plus an additional function that depends only on the future state g:

$$f_{\theta^*}(s,a,g) = \log\left(\frac{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s, a)}{C \cdot p_g(g)}\right) \tag{11}$$

$$= \underbrace{\log\left(\frac{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s, a)}{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g)}\right)}_{-d_\phi(s,a,g)} - \underbrace{\log\left(\frac{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g)}{C \cdot p_g(g)}\right)}_{-c_\psi(g)}.$$

Thus, we will apply the contrastive objective from Eq. 8 to an energy function $f_{\theta=(\phi,\psi)}(s,a,g)$ parametrized as the difference of a quasimetric network $d_\phi(s,a,g)$ an another learned function $c_\psi : \mathbb{R} \to \mathbb{R}$:

$$f_{\phi,\psi}(s,a,g) = c_\psi(g) - d_\phi(s,a,g). \tag{12}$$

The term $c_\psi(g)$ is important for allowing $f_\theta(s,a,g)$ to represent positive numbers, as $-d_\phi(s,a,g)$ is non-positive because it is a quasimetric network. With this parametrization, we can use Eq. (10) to obtain the successor distance as

$$f_{\theta^*}(g,a,g) - f_{\theta^*}(s,a,g)$$

$$= \cancelto{0}{-d_\phi(g,a,g)} + \cancel{c_\psi(g)} + d_\phi(s,a,g) - \cancel{c_\psi(g)}. \tag{13}$$

After contrastive learning, we will discard $c_\psi(g)$ and use $d_\phi(s,a,g)$ as our successor distance. We conclude by providing the formal result that this approach recovers the successor distance:

Lemma 4.1. *For $s \neq g$, the unique solution to the the loss function in Eq. (8) with the parametrization in Eq. (12) is*

$$d_{\phi^*}(s,a,g) = \log \frac{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s, \mathfrak{a}_0 = a)}{p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g)}. \tag{14}$$

See Appendix F for the proof.

One appealing aspect of this approach is that it only involves one learning step. The next section provides an alternative approach that proceeds in two steps.

4.3 Two-step distillation (CMD 2-step)

In this section we present an alternative approach to estimating the successor distance with a quasimetric network. While the first approach (CMD 1-step) is appealing because of its simplicity, this approach may be appealing in settings where pre-trained contrastive features are already available, but users want to boost performance by capitalizing on the inductive biases of quasimetric networks.

The key idea behind our approach is that that optimal critic from contrastive learning (Eq. 9) can be used to estimate the successor distance by performing a change of variables:

$$\begin{aligned} &f_\theta(g,a,g) - f_\theta(s,a,g) \\ &\quad = \log \frac{p^\pi_\gamma(\mathbf{s}^+=g \mid \mathbf{s}_0=g,a)}{Cp_g(g)} - \log \frac{p^\pi_\gamma(\mathbf{s}^+=g \mid \mathbf{s}_0=s,a)}{Cp_g(g)} \\ &\quad = \log \frac{p^\pi_\gamma(\mathbf{s}^+=g \mid \mathbf{s}_0=g,a)}{p^\pi_\gamma(\mathbf{s}^+=g \mid \mathbf{s}_0=s,a)}. \end{aligned} \tag{15}$$

This final expression is the successor distance; Appendix I will discuss why the action a in the numerator can be ignored. Because the successor distance obeys the triangle inequality (and the other quasimetric properties), we will distill this difference into a quasimetric network. We will call this method CMD 2-Step.

4.3.1 Distilling to a quasimetric architecture

The representations in Eq. 15 already form a quasimetric on $S \times A$, and could directly be used for action selection. However, because we know that these representations satisfy the triangle inequality, distilling them into a network that is architecturally-constrained to obey the triangle inequality serves as a very strong prior: a way of potentially combating overfitting and improving generalization. To do this, we distill the bound into a distance d_ϕ parameterized by an MRN quasimetric (Liu et al., 2023).

CMD 2-Step works by applying contrastive learning (Eq. 8). Following prior work (Eysenbach et al., 2022), we will parametrize the energy function as the inner product between learned representations: $f_{\phi,\psi}(s,a,g) = \phi(s,a)^T\psi(g)$. The critic parameters are thus $\theta = (\phi,\psi)$. We then distill the quasimetric architecture using Eq. (15) as a constraint. We enforce the constraint with a Lagrange multiplier λ to ensure that the margin $\mathcal{C}_{\hat\theta}(\{s_i,a_i\},\{g_i,a'_i\})$ for Eq. (15) satisfies $\mathcal{C}_{\hat\theta}(\{s_i,a_i\},\{g_i,a'_i\}) \le \varepsilon^2$ on pairs of states and future goals sampled from the data:

$$\begin{aligned} &\mathcal{C}_{\hat\theta}(\{s_i,a_i\},\{g_i,a'_i\}) \\ &\triangleq \sum_{i,j=1}^{B} \max\big(0, d_{\hat\theta}\big((s_i,a_i),(g_j,a'_j)\big) - f_{\phi,\psi}(s_i,a_i,g_i)\big)^2 \end{aligned}$$

$$\text{where } f_{\phi,\psi}(s,a,g) \triangleq \big(\phi(g,a) - \phi(s,a)\big)^T\psi(g). \tag{16}$$

When distilling a distance d_{SD}, subject to the constraint above, we want to be maximally conservative in determining which goals we can reach. We assume Eq. (15) as a prior, and use dual descent to perform a constrained minimization of the objective

$$\begin{aligned} &\mathcal{L}^d_{\hat\theta,\phi,\psi}\big(\{s_i,a_i\},\{g_i,a'_i\}\big) \triangleq \\ &\quad \sum_{i,j=1}^{B} \max\big(0, f_{\phi,\psi}(s_i,a_i,g_j) - d_{\hat\theta}(s_i,a_i,g_j)\big)^2, \end{aligned} \tag{17}$$

yielding an overall optimization

$$\begin{aligned} \min_{\hat\theta} \max_{\lambda\ge 0} \sum_{\{s_i,a_i,g_i,a'_i\}_{i=1}^B} \Big[&\mathcal{L}^d_{\hat\theta,\phi,\psi}\big(\{s_i,a_i\},\{g_i,a'_i\}\big) \\ &+ \lambda\big(\mathcal{C}_{\hat\theta}\big(\{s_i,a_i\},\{g_i,a'_i\}\big) - \varepsilon^2\big)\Big]. \end{aligned} \tag{18}$$

4.4 Policy extraction

Once we extract distance d_{SD}, we learn a goal-conditioned policy π_μ to select actions that minimize the distance successor between states and random goals (Schaul et al., 2015):

$$\min_\mu \mathbb{E}_{p_s(s)\ p_g(g,a')\pi_\mu(\hat a|s,g)}\big[\mathcal{L}^\pi_\mu\big(\{s_i,\hat a_i\},\{g_i,a'_i\}\big)\big] \tag{19}$$

To prevent the policy from sampling out-of-distribution actions for offline RL (Fujimoto & Gu, 2021; Kumar et al., 2020; 2019), we adopt another goal-conditioned behavioral cloning regularization from Zheng et al. (2023) or use advantage weighted regression (Nair et al., 2021).

With the behavior cloning regularization, the policy extraction loss becomes:

$$\begin{aligned} \mathcal{L}^\pi_\mu\big(\{s_i,a_i\},\{g_i,a'_i\}\big) = \sum_{i,j=1}^{B} &\mathbb{E}_{\hat a\sim\pi_\mu(\hat a|s_i,g_j)} \\ &\big[d_\phi\big((s_i,\hat a),(g_j,a'_j)\big) + \log\pi_\mu(a_i \mid s_i,g_i)\big]. \end{aligned} \tag{20}$$

5 Experiments

Our experiments study a synthetic 2D navigation task to see whether our proposed temporal distance can learn meaningful distances of pairs of states unseen together during training (i.e., *combinatorial generalization*). We also study the efficacy of extracting policies from this learned distance function, both in this 2D navigation setting and in a 111-dim robotic locomotion problem from the AntMaze benchmark suite. As discussed below, for the latter experiment our comparison will be restricted to small neural network sizes. Code for our experiments is linked in Appendix A and additional implementation details are provided in Appendix G.

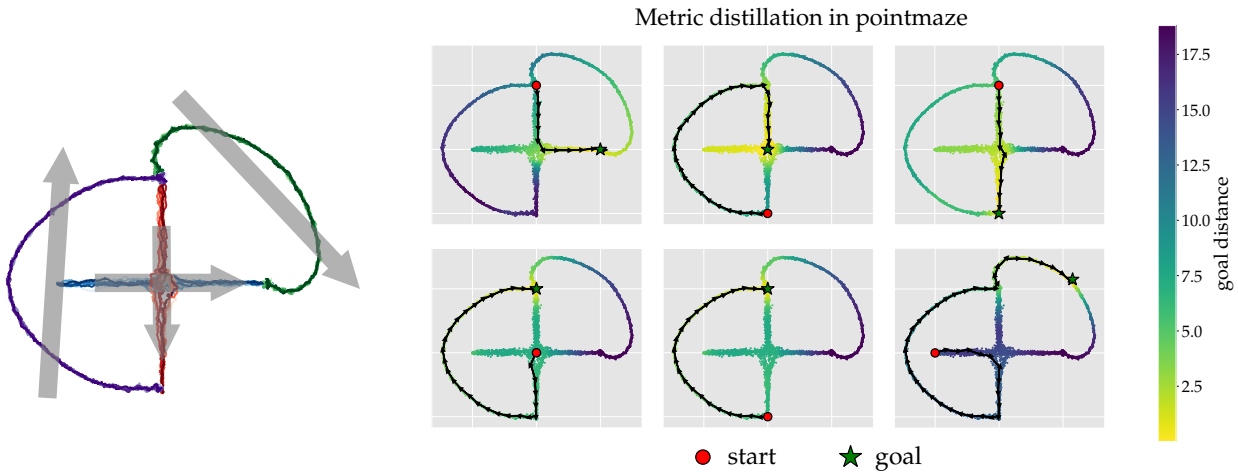


Figure 2. (Left) We collect four types of trajectories on this 2D navigation task. The large gray arrows depict the direction of motion. Note that navigating between certain states requires piecing together trajectories of different colors *(Right)* Our proposed temporal distance correctly pieces together trajectories, allowing an RL agent to successfully navigate between pairs of states that never occur on the same trajectory. This combinatorial generalization (Ghugare et al., 2024) or "stitching" (Fu et al., 2020) property is typically associated with bootstrapping with temporal difference learning, which our temporal distances do not require.

Table 1. **Offline RL benchmarks**: We use the AntMaze suite (Fu et al., 2020) of goal-conditioned RL tasks to compare our method to prior methods, measuring the success rate and standard error across multiple seeds.

	CMD 1-step (Ours)	**CMD 2-step (Ours)**	QRL	CRL (CPC)	GCBC	IQL[1]
umaze	90.3 ± 4.2	$\mathbf{97.0 \pm 0.4}$	76.8 ± 2.3	79.8 ± 1.6	65.4 ± 87.5	87.5
umaze-diverse	$\mathbf{90.3 \pm 4.6}$	$\mathbf{90.5 \pm 1.4}$	80.1 ± 1.3	77.6 ± 2.8	60.9 ± 62.2	62.2
medium-play	$\mathbf{78.0 \pm 4.0}$	72.3 ± 2.6	$\mathbf{76.5 \pm 2.1}$	72.6 ± 2.9	58.1 ± 71.2	71.2
medium-diverse	$\mathbf{83.0 \pm 3.1}$	71.8 ± 1.0	73.4 ± 1.9	71.5 ± 1.3	67.3 ± 70.0	70.0
large-play	$\mathbf{68.0 \pm 2.1}$	59.2 ± 1.8	52.9 ± 2.8	48.6 ± 4.4	32.4 ± 39.6	39.6
large-diverse	$\mathbf{74.5 \pm 2.3}$	63.6 ± 1.9	51.5 ± 3.8	54.1 ± 5.5	36.9 ± 47.5	47.5

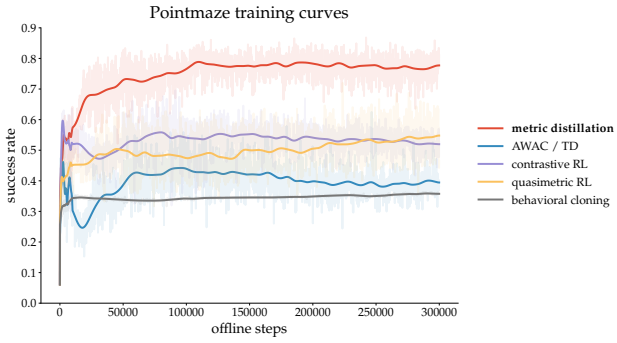


Figure 3. Metric distillation enables more efficient training and long-horizon compositional generalization.

5.1 Controlled experiments on synthetic data

We first present results in a simple 2D navigation environment to illustrate how our approach can recombine pieces of data to navigate between pairs of states unseen together during training (i.e., combinatorial generalization).

We start by collecting four types of trajectories, identified in Fig. 2 (left). We will be primarily interested in what distances our method assigns to pairs of states that occur on different types of trajectories. Our hypothesis is that, by virtue of the triangle inequality, our method will correctly reason about *global distances*, despite only being trained on *locally* on individual trajectories. Note that the collected data is directed, so we will also be test whether our learned distance obeys the arrow of time.

Visualizing the paths. Using these data, we learn the contrastive representations and distill them into a quasimetric architecture, as described in Section 4. In the subfigures in Fig. 2 (right), we visualize these distances using the colormap, with the goal set to the state identified with the ★. This figure also visualizes paths created using the learned distances. Starting at the state identified as ●, greedily select a next state within an L2 ball that has minimal temporal distance to the goal. We repeat this process until arriving at the goal. These planned paths demonstrate that the learned temporal distances perform combinatorial generalization;

each of the subfigures in Fig. 2 show examples of inferred paths that require correctly assigning distances to pairs of states that were unseen together during training. Note, too, that these paths follow the arrow of time: the small arrows depicting the paths go in the same direction that the data was collected (large gray arrows in the left subplot).

Control performance. We next study whether these learned distances can be used for control, using the same synthetic dataset as above. We will compare with four baselines. **AWAC/TD** learns distances using Q-learning with a reward that is -1 at every transition until the goal is reached (Lin et al., 2019; Kaelbling, 1993); at least in deterministic settings, these distances should correspond to hitting times. Quasimetric RL (Wang et al., 2023) is an extension of this baseline that uses a quasimetric architecture to represent these distances. Contrastive RL (Eysenbach et al., 2022) estimates distances directly using the contrastive features (the same as used for our method), but without the metric distillation step. For all these methods as well as our method, a policy is learned using advantage-weighted maximum likelihood (Neumann & Peters, 2008; Peters & Schaal, 2007). We also compare with a behavioral cloning baseline, which predicts the action that was most likely to occur in the dataset conditioned on state and goal.

We measure performance by evaluating the success rate of each these approaches at reaching randomly sampled goals. In Fig. 3, we plot this success rate over the course of training. Note that this experiment is done in the offline setting, so the X axis corresponds to the number of gradient steps. We observe that our temporal distance can successfully navigate to approximately 80% of goals, while the best prior method has a success rate of around 50%. Because our method starts with the same contrastive features as the contrastive RL baseline, the better performance of ours highlights the importance of the quasimetric architecture (i.e., of imposing the triangle inequality as an inductive bias). While both our method and quasimetric RL use a quasimetric architecture to represent a distance, we aim to represent the proposed distance metric from Section 4 while quasimetric RL aims to represent a hitting time; the better performance of our method highlights the need to use a temporal distance that is well defined in stochastic settings such as this.

5.2 Scaling to higher-dimensional tasks

To study whether our temporal distance learning approach is applicable to higher-dimensional tasks, we apply it to a 111-dimensional robotic control task (AntMaze (Fu et al., 2020)). In this problem setting we additionally condition the temporal distance on the action and use the learned distance as a value function for selecting actions.

We compare our approach to three competitive baselines. **GCBC** is a conditional imitation learning method that learns a goal-conditioned policy directly, without a value function or distance function (Ding et al., 2023; Lynch & Sermanet, 2021; Chen et al., 2021a; Ghosh et al., 2021). Both our method and Contrastive RL (**CRL**) (Eysenbach et al., 2022) learn representations in the same way (Sec. 4); the difference is that our method additionally distills these representations into a quasimetric architecture. Thus, comparing our method to CRL tests the importance of the triangle inequality as an inductive bias. We consider two variants of CRL using either rank-based NCE (van den Oord et al., 2019; Zheng et al., 2023) or binary-NCE (Gutmann & Hyvärinen, 2010), namely CRL (CPC) and CRL (NCE). Finally, Quasimetric RL (**QRL**) (Wang & Isola, 2022b) represents a different type of temporal distance with the same quasimetric architecture as our method; it is unclear whether the temporal distance from QRL obeys the triangle inequality in stochastic settings. Thus, comparing our method to QRL tests the importance of using a temporal distance that is well defined in stochastic settings. Prior work (Zheng et al., 2023) has shown that these baselines are more competitive than other recent alternatives, including IQL (Kostrikov et al., 2021) with HER (Andrychowicz et al., 2017) and decision transformer (Chen et al., 2021a).

6 Conclusion

The main contribution of this paper is a mathematical definition of temporal distance: one that obeys the triangle inequality, is meaningful in stochastic settings, and can be effectively estimated using modern deep learning techniques. Our results build upon prior work on quasimetric networks by showing how those architectures networks can be used to estimate temporal distances, including in stochastic settings. Our empirical results show that our learned distances stitch together data, allowing RL agents to navigate between states even when there does not exist a complete path between them in the training data. Taken together, these results suggest that some elements of dynamic programming methods might be realized by simple supervised learning methods combined with appropriate architectures.

Limitations. While we show that the method works effectively even on continuous settings, our theoretical results require that the MDP have discrete states. Our proposed distance may also be infinite in non-ergodic settings.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

[1]IQL results are taken from Kostrikov et al. (2021) which does not report standard errors.

Acknowledgements

We would like to acknowledge funding provided by ONR N00014-21-1-2838, AFOSR FA9550-22-1-0273, as well as NSF 2310757. This work was also supported by Princeton Research Computing, a consortium of groups led by the Princeton Institute for Computational Science and Engineering (PICSciE) and Office of Information Technology's Research Computing.

References

Akella, R. T., Eysenbach, B., Schneider, J., and Salakhutdinov, R. Distributional Distance Classifiers for Goal-Conditioned Reinforcement Learning. *arXiv*, 2023.

Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., McGrew, B., Tobin, J., Abbeel, O. P., and Zaremba, W. Hindsight Experience Replay. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H. P., and Silver, D. Successor Features for Transfer in Reinforcement Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Black, K., Nakamoto, M., Atreya, P., Walke, H., Finn, C., Kumar, A., and Levine, S. Zero-Shot Robotic Manipulation with Pretrained Image-Editing Diffusion Models, October 2023.

Chane-Sane, E., Schmid, C., and Laptev, I. Goal-Conditioned Reinforcement Learning with Imagined Subgoals, July 2021.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. Decision Transformer: Reinforcement Learning via Sequence Modeling. In *Advances in Neural Information Processing Systems*, volume 34, pp. 15084–15097. Curran Associates, Inc., 2021a.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1597–1607. PMLR, November 2020.

Chen, T., Luo, C., and Li, L. Intriguing Properties of Contrastive Losses. In *Advances in Neural Information Processing Systems*, volume 34, pp. 11834–11845. Curran Associates, Inc., 2021b.

Dayan, P. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

Ding, W., Lin, H., Li, B., and Zhao, D. Generalizing Goal-Conditioned Reinforcement Learning with Variational Causal Reasoning, May 2023.

Durugkar, I., Tec, M., Niekum, S., and Stone, P. Adversarial intrinsic motivation for reinforcement learning. *Advances in Neural Information Processing Systems*, 34: 8622–8636, 2021.

Eysenbach, B., Salakhutdinov, R., and Levine, S. C-Learning: Learning to Achieve Goals via Recursive Classification. In *9th International Conference on Learning Representations, ICLR 2021*, 2021.

Eysenbach, B., Zhang, T., Levine, S., and Salakhutdinov, R. R. Contrastive Learning as Goal-Conditioned Reinforcement Learning. *Advances in Neural Information Processing Systems*, 35:35603–35620, December 2022.

Eysenbach, B., Myers, V., Levine, S., and Salakhutdinov, R. Contrastive Representations Make Planning Easy. In *NeurIPS 2023 Workshop on Generalization in Planning*, December 2023.

Fang, K., Yin, P., Nair, A., Walke, H. R., Yan, G., and Levine, S. Generalization with Lossy Affordances: Leveraging Broad Offline Data for Learning Visuomotor Tasks. In *Proceedings of The 6th Conference on Robot Learning*, pp. 106–117. PMLR, March 2023.

Ferns, N., Panangaden, P., and Precup, D. Bisimulation metrics for continuous markov decision processes. *SIAM Journal on Computing*, 40(6):1662–1714, 2011.

Fu, J., Kumar, A., Nachum, O., Tucker, G., and Levine, S. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Fujimoto, S. and Gu, S. S. A minimalist approach to offline reinforcement learning. *Advances in neural information processing systems*, 34:20132–20145, 2021.

Ghosh, D., Gupta, A., and Levine, S. Learning Actionable Representations with Goal-Conditioned Policies, January 2019.

Ghosh, D., Gupta, A., Reddy, A., Fu, J., Devin, C. M., Eysenbach, B., and Levine, S. Learning to Reach Goals via Iterated Supervised Learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Ghosh, D., Bhateja, C., and Levine, S. Reinforcement Learning from Passive Data via Latent Intentions, April 2023.

Ghugare, R., Geist, M., Berseth, G., and Eysenbach, B. Closing the Gap between TD Learning and Supervised Learning – A Generalisation Point of View, January 2024.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, March 2010.

Gutmann, M. U. and Hyvarinen, A. Noise-Contrastive Estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics. *Journal of Machine Learning Research*, 2012.

Hansen-Estruch, P., Zhang, A., Nair, A., Yin, P., and Levine, S. Bisimulation makes analogies in goal-conditioned reinforcement learning. In *International Conference on Machine Learning*, pp. 8407–8426. PMLR, 2022.

Hartikainen, K., Geng, X., Haarnoja, T., and Levine, S. Dynamical distance learning for semi-supervised and unsupervised skill discovery. *arXiv preprint arXiv:1907.08225*, 2019.

Hejna, J., Gao, J., and Sadigh, D. Distance Weighted Supervised Learning for Offline Interaction Data. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 12882–12906. PMLR, July 2023.

Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

Kaelbling, L. P. Learning to Achieve Goals. *IJCAI*, 1993.

Kostrikov, I., Nair, A., and Levine, S. Offline Reinforcement Learning with Implicit Q-Learning, October 2021.

Kumar, A., Fu, J., Soh, M., Tucker, G., and Levine, S. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

Kumar, A., Zhou, A., Tucker, G., and Levine, S. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 1179–1191, 2020.

Laird, J. E., Newell, A., and Rosenbloom, P. S. SOAR: An architecture for general intelligence. *Artificial Intelligence*, 33(1):1–64, September 1987. ISSN 0004-3702. doi: 10.1016/0004-3702(87)90050-6.

Lan, C. L., Bellemare, M. G., and Castro, P. S. Metrics and Continuity in Reinforcement Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(9): 8261–8269, May 2021. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v35i9.17005.

Lin, X., Baweja, H. S., and Held, D. Reinforcement learning without ground-truth state. *arXiv preprint arXiv:1905.07866*, 2019.

Liu, B., Feng, Y., Liu, Q., and Stone, P. Metric Residual Network for Sample Efficient Goal-Conditioned Reinforcement Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(7):8799–8806, June 2023. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v37i7.26058.

Lynch, C. and Sermanet, P. Language Conditioned Imitation Learning over Unstructured Data. In *Robotics: Science and Systems*. arXiv, July 2021.

Ma, Y. J., Kumar, V., Zhang, A., Bastani, O., and Jayaraman, D. LIV: Language-Image Representations and Rewards for Robotic Control. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 23301–23320. PMLR, July 2023a.

Ma, Y. J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., and Zhang, A. VIP: Towards Universal Visual Reward and Representation via Value-Implicit Pre-Training. In *International Conference on Learning Representations*. arXiv, 2023b. doi: 10.48550/arXiv.2210.00030.

Ma, Z. and Collins, M. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. *arXiv preprint arXiv:1809.01812*, 2018.

Mazoure, B., Tachet des Combes, R., Doan, T. L., Bachman, P., and Hjelm, R. D. Deep Reinforcement and InfoMax Learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 3686–3698. Curran Associates, Inc., 2020.

Mendonca, R., Rybkin, O., Daniilidis, K., Hafner, D., and Pathak, D. Discovering and Achieving Goals via World Models. In *Advances in Neural Information Processing Systems*, volume 34, pp. 24379–24391. Curran Associates, Inc., 2021.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

Myers, V., He, A. W., Fang, K., Walke, H. R., Hansen-Estruch, P., Cheng, C.-A., Jalobeanu, M., Kolobov, A., Dragan, A., and Levine, S. Goal Representations for Instruction Following: A Semi-Supervised Language Interface to Control. In *Proceedings of The 7th Conference on Robot Learning*, pp. 3894–3908. PMLR, December 2023.

Nair, A., Gupta, A., Dalal, M., and Levine, S. AWAC: Accelerating Online Reinforcement Learning with Offline Datasets, April 2021.

Neumann, G. and Peters, J. Fitted q-iteration by advantage weighted regression. *Advances in neural information processing systems*, 21, 2008.

N'Guyen, S., Moulin-Frier, C., and Droulez, J. Decision Making under Uncertainty: A Quasimetric Approach. *PLoS ONE*, 8(12):e83411, December 2013. doi: 10.1371/journal.pone.0083411.

Ni, T., Eysenbach, B., Seyedsalehi, E., Ma, M., Gehring, C., Mahajan, A., and Bacon, P.-L. Bridging state and history representations: Understanding self-predictive RL. *arXiv preprint arXiv:2401.08898*, 2024.

Paluszyński, M. and Stempak, K. On quasi-metric and metric spaces. *Proceedings of the American Mathematical Society*, 137(12):4307–4312, 1931. ISSN 0002-9939, 1088-6826. doi: 10.1090/S0002-9939-09-10058-8.

Park, S., Ghosh, D., Eysenbach, B., and Levine, S. HIQL: Offline Goal-Conditioned RL with Latent States as Actions, October 2023a.

Park, S., Rybkin, O., and Levine, S. METRA: Scalable Unsupervised RL with Metric-Aware Abstraction, October 2023b.

Peters, J. and Schaal, S. Reinforcement learning by reward-weighted regression for operational space control. In *Proceedings of the 24th international conference on Machine learning*, pp. 745–750, 2007.

Poole, B., Ozair, S., van den Oord, A., Alemi, A. A., and Tucker, G. On Variational Bounds of Mutual Information, May 2019.

Popper, K. R. The Arrow of Time. *Nature*, 177(4507): 538–538, March 1956. ISSN 1476-4687. doi: 10.1038/177538a0.

Qian, R., Meng, T., Gong, B., Yang, M.-H., Wang, H., Belongie, S., and Cui, Y. Spatiotemporal Contrastive Video Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6964–6974, 2021.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning Transferable Visual Models From Natural Language Supervision, February 2021.

Saunshi, N., Plevrakis, O., Arora, S., Khodak, M., and Khandeparkar, H. A Theoretical Analysis of Contrastive Unsupervised Representation Learning. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 5628–5637. PMLR, May 2019.

Savinov, N., Dosovitskiy, A., and Koltun, V. Semi-parametric topological memory for navigation. *arXiv preprint arXiv:1803.00653*, 2018.

Schaul, T., Horgan, D., Gregor, K., and Silver, D. Universal value function approximators. In *International conference on machine learning*, pp. 1312–1320. PMLR, 2015.

Schwarzer, M., Anand, A., Goel, R., Hjelm, R. D., Courville, A., and Bachman, P. Data-efficient reinforcement learning with self-predictive representations. *arXiv preprint arXiv:2007.05929*, 2020.

Sermanet, P., Lynch, C., Hsu, J., and Levine, S. Time-Contrastive Networks: Self-Supervised Learning from Multi-view Observation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 486–487, July 2017. doi: 10.1109/CVPRW.2017.69.

Shah, R., Martín-Martín, R., and Zhu, Y. MUTEX: Learning Unified Policies from Multimodal Task Specifications, September 2023.

Sohn, K. Improved Deep Metric Learning with Multi-class N-pair Loss Objective. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Touati, A. and Ollivier, Y. Learning One Representation to Optimize All Rewards. In *Advances in Neural Information Processing Systems*, volume 34, pp. 13–23. Curran Associates, Inc., 2021.

van den Oord, A., Li, Y., and Vinyals, O. Representation Learning with Contrastive Predictive Coding, January 2019.

Venkattaramanujam, S., Crawford, E., Doan, T., and Precup, D. Self-supervised learning of distance functions for goal-conditioned reinforcement learning. *arXiv preprint arXiv:1907.02998*, 2019.

Wang, T. and Isola, P. Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9929–9939. PMLR, November 2020.

Wang, T. and Isola, P. Improved Representation of Asymmetrical Distances with Interval Quasimetric Embeddings, November 2022a.

Wang, T. and Isola, P. On the Learning and Learnability of Quasimetrics. In *International Conference on Learning Representations*. ICLR, 2022b.

Wang, T., Torralba, A., Isola, P., and Zhang, A. Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 36411–36430. PMLR, July 2023.

Wu, C., Wu, F., and Huang, Y. Rethinking InfoNCE: How Many Negative Samples Do You Need?, May 2021.

Yang, R., Lu, Y., Li, W., Sun, H., Fang, M., Du, Y., Li, X., Han, L., and Zhang, C. Rethinking Goal-Conditioned Supervised Learning and Its Connection to Offline RL. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Yang, R., Yong, L., Ma, X., Hu, H., Zhang, C., and Zhang, T. What is Essential for Unseen Goal Generalization of Offline Goal-conditioned RL? In *Proceedings of the 40th International Conference on Machine Learning*, pp. 39543–39571. PMLR, July 2023.

Zheng, C., Salakhutdinov, R., and Eysenbach, B. Contrastive Difference Predictive Coding, October 2023.

A Code

An implementation of the evaluated methods is available at https://github.com/vivekmyers/contrastive_metrics.

B Hitting Times

In this section, we show several lemmas relating the discounted state occupancy measure (defined in Eqs. (2) and (3)) to the hitting times of states and goals. We start by defining a notion of hitting time:

Definition B.1. *For $\pi \in \Pi$ and $s, g \in \mathcal{S}$, define the random variable $H_s^\pi(g)$ by*

$$H_s^\pi(g) = \min\{t \geq 0 : E_t\} \tag{21}$$
$$\text{where } E_t \text{ is the event that } \mathfrak{s}_t = g \text{ given } \mathfrak{s}_0 = s.$$

In other words, $H_s^\pi(g)$ is the smallest t such that $\mathfrak{s}_t = g$ starting in $\mathfrak{s}_0 = s$, i.e., the hitting time of g.

Now, we can relate the discounted state occupancy measure to the hitting time of a goal.

Lemma B.1. *For $H_s^\pi(g)$ defined as (21),*

$$p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s) = \mathbb{E}\left[\gamma^{H_s^\pi(g)}\right] p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g).$$

Proof. Let $p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = s, H_s^\pi(g) = h)$ be the probability of reaching goal g at time step t when starting at state s given hitting time $H_s^\pi(g) = h$. By the definition of $H_s^\pi(g)$, we have

$$p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = s, H_s^\pi(g) = h) = \begin{cases} 0 & t < h \\ p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_h = g) & t \geq h \end{cases}. \tag{22}$$

Thus,

$$\begin{aligned}
p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s) &= (1-\gamma) \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = s) \\
&= (1-\gamma) \sum_{t=0}^\infty \sum_{h=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g, H_s^\pi(g) = h \mid \mathfrak{s}_0 = s) \\
&= \sum_{h=0}^\infty p(H_s^\pi(g) = h) \Big((1-\gamma) \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = s, H_s^\pi(g) = h) \Big) \\
&= \sum_{h=0}^\infty p(H_s^\pi(g) = h) \Big((1-\gamma) \sum_{t=h}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_h = g) \Big) && \text{(Plug in Eq. (22))} \\
&= \sum_{h=0}^\infty \gamma^h p(H_s^\pi(g) = h) \Big((1-\gamma) \sum_{t=h}^\infty \gamma^{t-h} p^\pi(\mathfrak{s}_{t-h} = g \mid \mathfrak{s}_0 = g) \Big) && \text{(Stationary property of MDP)} \\
&= \sum_{h=0}^\infty \gamma^h p(H_s^\pi(g) = h) \Big((1-\gamma) \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = g) \Big) && \text{(Change of variables)} \\
&= \mathbb{E}\left[\gamma^{H_s^\pi(g)}\right] p_\gamma^\pi(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g),
\end{aligned}$$

as desired. □

We can generalize this result to account for actions as well.

Definition B.2. *For $\pi \in \Pi$, $s, g \in \mathcal{S}$, and $a, a' \in \mathcal{A}$, we define the following additional hitting time random variables*

$$H_{s,a}^\pi(g, a') = \min\{t \geq 0 : E_t\} \tag{23}$$
$$\text{where } E_t \text{ is the event that } \mathfrak{s}_t = g, \mathfrak{a}_t = a' \text{ given } \mathfrak{s}_0 = s, \mathfrak{a}_0 = a$$
$$H_{s,a}^\pi(g) = \min\{t \geq 0 : E_t\} \tag{24}$$
$$\text{where } E_t \text{ is the event that } \mathfrak{s}_t = g \text{ given } \mathfrak{s}_0 = s, \mathfrak{a}_0 = a.$$

We now show an analogous result for the discounted state-action occupancy measure.

Lemma B.2. *For $H^\pi_{s,a}(g,a)$ defined as (23) and $s \neq g$,*

$$p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' | \mathfrak{s}_0{=}s, \mathfrak{a}_0{=}a) = \mathbb{E}\left[\gamma^{H^\pi_{s,a}(g,a')}\right] p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' | \mathfrak{s}_0{=}g, \mathfrak{a}_0{=}a').$$

Proof. Let $p^\pi(\mathfrak{s}_t = g, \mathfrak{a}_t = a' \mid \mathfrak{s}_0 = s, \mathfrak{a}_0 = a, H^\pi_{s,a}(g,a') = h)$ be the probability of reaching goal g at time step t then taking action a', when starting at state s given the hitting time $H^\pi_{s,a}(g) = h$ and π takes action a' at time h. By the definition of $H^\pi_{s,a}(g)$, we have

$$p^\pi(\mathfrak{s}_t = g, \mathfrak{a}_t = a' \mid \mathfrak{s}_0 = s, \mathfrak{a}_0 = a, H^\pi_{s,a}(g,a') = h) = \begin{cases} 0 & t < h \\ 1 & t = h \\ \pi(a' \mid g) p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_h = g, \mathfrak{a}_h = a') & t > h. \end{cases} \tag{25}$$

Thus,

$$\begin{aligned}
&p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' | \mathfrak{s}_0{=}s, \mathfrak{a}_0{=}a) \\
&\quad = (1-\gamma) \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g, \mathfrak{a}_t = a' \mid \mathfrak{s}_0 = s, \mathfrak{a}_0 = a) \\
&\quad = (1-\gamma) \sum_{t=0}^\infty \sum_{h=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g, \mathfrak{a}_t = a', H^\pi_{s,a}(g,a') = h \mid \mathfrak{s}_0 = s, \mathfrak{a}_0 = a) \\
&\quad = \sum_{h=0}^\infty p\big(H^\pi_{s,a}(g,a') = h\big) \Big((1-\gamma) \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g, \mathfrak{a}_t = a' \mid \mathfrak{s}_0 = s, \mathfrak{a}_0 = a, H^\pi_{s,a}(g,a') = h) \Big) \\
&\quad = \sum_{h=0}^\infty p\big(H^\pi_{s,a}(g,a') = h\big) (1-\gamma) \Big(1 + \sum_{t=h+1}^\infty \gamma^t \pi(a' \mid g) p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_h = g, \mathfrak{a}_h = a') \Big) && \text{(Plug in Eq. (25))} \\
&\quad = \sum_{h=0}^\infty \gamma^h p\big(H^\pi_{s,a}(g,a') = h\big) (1-\gamma) \Big(1 + \sum_{t=h+1}^\infty \gamma^t \pi(a' \mid g) p^\pi(\mathfrak{s}_{t-h} = g \mid \mathfrak{s}_h = g, \mathfrak{a}_h = a') \Big) \\
&\qquad && \text{(Stationary property of MDP)} \\
&\quad = \sum_{h=0}^\infty \gamma^h p\big(H^\pi_{s,a}(g,a') = h\big) (1-\gamma) \Big(1 + \sum_{t=1}^\infty \gamma^t \pi(a' \mid g) p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = g, \mathfrak{a}_0 = a') \Big) && \text{(Change of variables)} \\
&\quad = \sum_{h=0}^\infty \gamma^h p\big(H^\pi_{s,a}(g,a') = h\big) (1-\gamma) \Big(1 + \sum_{t=1}^\infty \gamma^t \pi(a' \mid g) p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_1 = s') P(s' \mid s, a) \Big) \\
&\quad = \sum_{h=0}^\infty \gamma^h p\big(H^\pi_{s,a}(g,a') = h\big) (1-\gamma) \Big(1 + \gamma \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = s') \pi(a' \mid g) P(s' \mid s, a) \Big) && \text{(Change of variables)} \\
&\quad = \sum_{h=0}^\infty \gamma^h p\big(H^\pi_{s,a}(g,a') = h\big) (1-\gamma) \Big(\delta_{g,a'}(g,a') + \gamma \sum_{t=0}^\infty \gamma^t p^\pi(\mathfrak{s}_t = g \mid \mathfrak{s}_0 = s') \pi(a' \mid g) P(s' \mid s, a) \Big) \\
&\quad = \sum_{h=0}^\infty \gamma^h p\big(H^\pi_{s,a}(g,a') = h\big) p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' | \mathfrak{s}_0{=}g, \mathfrak{a}_0{=}a') \\
&\quad = \mathbb{E}\left[\gamma^{H^\pi_{s,a}(g,a')}\right] p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' | \mathfrak{s}_0{=}g, \mathfrak{a}_0{=}a'),
\end{aligned}$$

□

Remark B.3. *The hitting times $H^\pi_s(g)$ and $H^\pi_{s,a}(g)$ are independent of the distribution $\pi(\cdot \mid g)$.*

Remark B.4. *We can write*

$$H^\pi_{s,a}(g,a') = H^\pi_{s,a}(g) + \mathbb{E}_{\pi(\hat{a}|g)}\left[H^\pi_{g,\hat{a}}(g,a')\right].$$

These remarks follow from the definitions in Eqs. (23) and (24) and the conditional independence of the states before g is reached and the action taken at g.

C Proofs of Lemmas 3.1 and 3.2

Lemma 3.1. *$d_{\text{SD}}\big((s,a),(s',a')\big)$ is independent of a' when $s \neq s'$.*

Proof. Suppose $s \neq g$. We have from Eq. (6) that

$$\begin{aligned}
&d_{\text{SD}}\big((s,a),(g,a')\big) \\
&\quad= \min_{\pi\in\Pi}\left[\log \frac{p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' \,|\, \mathfrak{s}_0{=}g, \mathfrak{a}_0{=}a')}{p^\pi_\gamma(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a' \,|\, \mathfrak{s}_0{=}s, \mathfrak{a}_0{=}a)}\right] \\
&\quad= -\max_{\pi\in\Pi}\Big[\log \mathbb{E}\big[\gamma^{H^\pi_{s,a}(g,a')}\big]\Big] && \text{(Lemma B.2)} \\
&\quad= -\max_{\pi\in\Pi} \log \mathbb{E}\big[\gamma^{H^\pi_{s,a}(g)+\mathbb{E}_{\pi(\hat{a}\,|\,g)}[H^\pi_{g,\hat{a}}(g,a')]}\big]. && \text{(Remark B.4)}
\end{aligned}$$

Now, from Remark B.3, the first term $H^\pi_{s,a}(g)$ is independent of $\pi(\cdot \mid g)$. Meanwhile, the second term $\mathbb{E}_{\pi(\hat{a}|g)}[H^\pi_{g,\hat{a}}(g,a')]$ is minimized when $\pi(\hat{a} \mid g) = \delta_{a'}(\hat{a})$, i.e., when the action taken at g is a'. Thus, at the maximum $\pi(\cdot \mid g) = \delta_{a'}(\cdot)$; continuing, we see

$$\begin{aligned}
&d_{\text{SD}}\big((s,a),(g,a')\big) \\
&\quad= -\max_{\pi\in\Pi} \log \mathbb{E}\big[\gamma^{H^\pi_{s,a}(g)+\mathbb{E}_{\pi(\hat{a}\,|\,g)}[H^\pi_{g,\hat{a}}(g,a')]}\big] \\
&\quad= -\max_{\pi\in\Pi} \log \mathbb{E}\big[\gamma^{H^\pi_{s,a}(g)}\big].
\end{aligned}$$

From this last expression we see that $d_{\text{SD}}\big((s,a),(g,a')\big)$ is independent of the action at the goal a', as desired. □

Lemma 3.2. *Selecting actions to minimize the successor distance is equivalent to selecting actions to maximize the (scaled and shifted) Q-function:*

$$\begin{aligned}
&-d_{\text{SD}}(s,a,g) = \tfrac{1}{p_g(g)} Q(s,a,g) + c_\psi(g) \\
&\implies \arg\max_a d_{\text{SD}}(s,a,g) = \arg\max_a Q(s,a,g).
\end{aligned}$$

Proof. As noted in prior work Eysenbach et al. (2022, Lemma 4.1), the optimal critic (Eq. 9) is equivalent to a scaled Q function:

$$e^{f_{\theta^*}(s,a,g)} = \frac{1}{C \cdot p_g(g)} \underbrace{p^\pi_\gamma(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}s, a)}_{Q(s,a,g)}.$$

Eq. (11) then tells us that the successor distance differs from $f_{\theta^*}(s,a,g)$ by a term that depends only on g, so taking the argmin of the successor distance is the same as taking the argmax of this scaled Q function. □

D Proof of Lemma 3.3

Now, we will prove Lemma 3.3.

Lemma 3.3. *For any $s, w, g \in \mathcal{S}$, $\pi \in \Pi$,*

$$\begin{aligned}
\max_{\pi'\in\Pi}\left[\frac{p^{\pi'}_\gamma(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}w) p^\pi_\gamma(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}s)}{p^\pi_\gamma(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}w)}\right] \\
\leq \max_{\pi'\in\Pi} p^{\pi'}_\gamma(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}s).
\end{aligned}$$

Proof. Define $\tilde{\pi} \in \Pi_{\text{NM}}$ to be the non-Markovian policy that starts executing π' and switches to π after reaching w:

$$\tilde{\pi}(a_t \mid s_t) = \begin{cases} \pi(a_t \mid s_t) & w \in \{s_0, s_1, \ldots, s_t\} \\ \pi'(a_t \mid s_t) & \text{otherwise.} \end{cases}$$

We take $\pi' \in \Pi$ to be an arbitrary policy. Let E_1 be the event where the hitting time of waypoint w is less than the hitting time of goal g starting from state s, i.e., $E_1 = \{H_s^{\tilde{\pi}}(w) < H_s^{\tilde{\pi}}(g)\}$. Complementary, let E_2 be the event where the hitting time of waypoint w is greater than or equal to the hitting time of goal g starting from state s, i.e., $E_2 = \{H_s^{\tilde{\pi}}(w) \geq H_s^{\tilde{\pi}}(g)\}$. We note that E_1 and E_2 are mutually exclusive.

We start by rewriting $p_\gamma^{\tilde{\pi}}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s)$:

$$\begin{aligned} p_\gamma^{\tilde{\pi}}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s) &= \sum_{h=0}^{\infty} p(H_s^{\tilde{\pi}}(w)=h)\left((1-\gamma)\sum_{t=0}^{\infty}\gamma^t p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h)\right) \\ &= \sum_{h=0}^{\infty} p(H_s^{\pi'}(w)=h)\left((1-\gamma)\sum_{t=0}^{\infty}\gamma^t p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h)\right). \end{aligned} \tag{26}$$

Now, $p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h)$ can be written as

$$p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h) = \begin{cases} 0 & t<h, \text{under } E_1 \\ p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h, H_s^{\tilde{\pi}}(g)\leq h) & t<h, \text{under } E_2 \\ p^{\pi}(\mathfrak{s}_t=g \mid \mathfrak{s}_h=w) & t\geq h \end{cases}. \tag{27}$$

Dropping the first h terms (which are all non-negative), we get

$$\sum_{t=0}^{\infty}\gamma^t p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h) \geq \sum_{t=h}^{\infty}\gamma^t p^{\tilde{\pi}}(\mathfrak{s}_t=g \mid \mathfrak{s}_0=s, H_s^{\tilde{\pi}}(w)=h) = \sum_{t=h}^{\infty}\gamma^t p^{\pi}(\mathfrak{s}_t=g \mid \mathfrak{s}_h=w)$$

Plugging this inequality into Eq. (26), we have

$$\begin{aligned} p_\gamma^{\tilde{\pi}}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s) &\geq \sum_{h=0}^{\infty} p(H_s^{\pi'}(w)=h)\left((1-\gamma)\sum_{t=h}^{\infty}\gamma^t p^{\pi}(\mathfrak{s}_t=g \mid \mathfrak{s}_h=w)\right) \\ &= \sum_{h=0}^{\infty}\gamma^h p(H_s^{\pi'}(w)=h)\left((1-\gamma)\sum_{t=h}^{\infty}\gamma^{t-h} p^{\pi}(\mathfrak{s}_{t-h}=g \mid \mathfrak{s}_0=w)\right) && \text{(Stationary property of MDP)} \\ &= \sum_{h=0}^{\infty}\gamma^h p(H_s^{\pi'}(w)=h)\left((1-\gamma)\sum_{t=0}^{\infty}\gamma^{t} p^{\pi}(\mathfrak{s}_{t}=g \mid \mathfrak{s}_0=w)\right) && \text{(Change of variables)} \\ &= \mathbb{E}\left[\gamma^{H_s^{\pi'}(w)}\right] p_\gamma^{\pi}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}w). \end{aligned}$$

Applying Lemma B.1 to the last step, we see

$$\begin{aligned} p_\gamma^{\tilde{\pi}}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s) &\geq \mathbb{E}\left[\gamma^{H_s^{\pi}(w)}\right] p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}w) \\ &= \frac{p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}w) p_\gamma^{\pi}(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}s)}{p_\gamma^{\pi}(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}w)}. \end{aligned}$$

Since there is a stationary Markovian optimal policy π^* for r_g in M, we know from Lemma 3.2 that

$$p_\gamma^{\tilde{\pi}}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s) \leq \max_{\pi'\in\Pi} p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s),$$

so we have

$$\max_{\pi'\in\Pi} p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s) \geq \frac{p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}w) p_\gamma^{\pi}(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}s)}{p_\gamma^{\pi}(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}w)}.$$

Since π' on the RHS was arbitrary, we conclude

$$\max_{\pi'\in\Pi}\left[\frac{p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}w) p_\gamma^{\pi}(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}s)}{p_\gamma^{\pi}(\mathfrak{s}^+{=}w\,|\,\mathfrak{s}_0{=}w)}\right] \leq \max_{\pi'\in\Pi} p_\gamma^{\pi'}(\mathfrak{s}^+{=}g\,|\,\mathfrak{s}_0{=}s).$$

□

Lemma D.1. *For any $s, w, g \in \mathcal{S}$, $a_s, a_w, a_g \in \mathcal{A}$, and $\pi \in \Pi$, we have*

$$\max_{\pi' \in \Pi}\left[\frac{p_\gamma^{\pi'}(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a_g \,|\, \mathfrak{s}_0{=}w, \mathfrak{a}_0{=}a_w) p_\gamma^{\pi'}(\mathfrak{s}^+{=}w, \mathfrak{a}^+{=}a_w \,|\, \mathfrak{s}_0{=}s, \mathfrak{a}_0{=}a_s)}{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w, \mathfrak{a}^+{=}a_w \,|\, \mathfrak{s}_0{=}w, \mathfrak{a}_0{=}a_w)}\right] \leq \max_{\pi' \in \Pi}\left[p_\gamma^{\pi'}(\mathfrak{s}^+{=}g, \mathfrak{a}^+{=}a_g \,|\, \mathfrak{s}_0{=}s, \mathfrak{a}_0{=}a_s)\right].$$

The proof follows from the same argument as in Lemma 3.3 but applying Lemma B.2 instead of Lemma B.1.

E Proof of Theorem 3.4

Theorem 3.4. *d_{SD} is a quasimetric over $\mathcal{S}$, satisfying the triangle inequality and other properties from Definition 3.1.*

Proof. We check the conditions of Definition 3.1:

Positivity: Applying Lemma B.1, we see

$$\begin{aligned} d_{\text{SD}}(s, g) &= \min_{\pi \in \Pi} \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) - \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}s) \\ &= \min_{\pi \in \Pi} \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) - \log \mathbb{E}\left[\gamma^{H_s^\pi(g)}\right] p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) \\ &\geq \min_{\pi \in \Pi} \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) - \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) \\ &= 0. \end{aligned}$$

Identity: We see $d_{\text{SD}}(s, g) = 0$ precisely iff $p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) = p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}s)$ for some $\pi \in \Pi$. This holds when $s = g$. For $s \neq g$, we have $p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}s) \leq \gamma p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g)$. Since $p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) \geq 1 - \gamma$ by construction, $d_{\text{SD}}(s, g) \neq 0$.

Triangle inequality: We see:

$$\begin{aligned} d_{\text{SD}}(s, g) &= \min_{\pi \in \Pi} \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) - \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}s) \\ &\leq \min_{\pi \in \Pi} \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) - \log\left(\max_{\pi' \in \Pi}\left[\frac{p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}w) p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}s)}{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}w)}\right]\right) && \text{(Lemma 3.3)} \\ &= \min_{\pi \in \Pi} \log p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g) - \max_{\pi' \in \Pi} \log\left(\frac{p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}w) p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}s)}{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}w)}\right) \\ &= \left(\min_{\pi \in \Pi} \log \frac{p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g)}{p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}w)}\right) - \left(\max_{\pi' \in \Pi} \log \frac{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}s)}{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}w)}\right) \\ &= \left(\min_{\pi \in \Pi} \log \frac{p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}g)}{p_\gamma^\pi(\mathfrak{s}^+{=}g \,|\, \mathfrak{s}_0{=}w)}\right) + \left(\min_{\pi' \in \Pi} \log \frac{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}w)}{p_\gamma^{\pi'}(\mathfrak{s}^+{=}w \,|\, \mathfrak{s}_0{=}s)}\right) \\ &= d_{\text{SD}}(w, g) + d_{\text{SD}}(s, w) && (28) \end{aligned}$$

as desired.

□

Consider the following didactic example for why we might want to extend the successor distance to the state-action space $\mathcal{S} \times \mathcal{A}$ (Eq. 4).

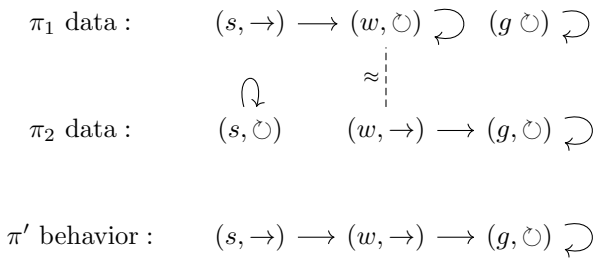


Figure 4. A simple illustration of a metric over $\mathcal{S} \times \mathcal{A}$. To stitch the behavior $s \to w$ from π_1 and $w \to g$ from π_2 to the behavior $s \to g$ that is possible under some policy π', we enforce an additional constraint that distances to $(w, \rightarrow)$ are the same as distances to $(w, \circlearrowleft)$.

Corollary 3.4.1. *d_{SD} is a quasimetric over $\mathcal{S} \times \mathcal{A}$.*

This statement follows from the same argument as in Theorem 3.4 but applying Lemma D.1 instead of Lemma 3.3 to the triangle inequality.

Corollary 3.4.2. *d_{SD} is a quasimetric over an uncontrolled Markov process as in Eq. (5).*

This statement follows from Theorem 3.4 by taking $\mathcal{A} = \{a\}$ so $\Pi = \{\pi\}$.

F Analysis of CMD-1

Lemma 4.1. *For $s \neq g$, the unique solution to the the loss function in Eq. (8) with the parametrization in Eq. (12) is*

$$d_{\phi^*}(s, a, g) = \log \frac{p^\pi_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = s, \mathfrak{a}_0 = a)}{p^\pi_\gamma(\mathfrak{s}^+ = g \,|\, \mathfrak{s}_0 = g)}. \tag{14}$$

Proof. Eq. (10) together with Eq. (13) tell us that, if $f(s, a, g)$ satisfies Eq. (9), then the learned $d_\phi(s, a, g)$ is the successor distance. What remains is to show that the parametrization in Eq. (12) is sufficient to represent Eq. (9): Eq. (11) tells us that it is sufficient for (1) we use a universal quasimetric network for $d_\phi(s, a, g)$ (Liu et al., 2023; Wang & Isola, 2022a), and (2) use a universal network $c_\psi(g)$ (e.g., sufficient layers in a neural network (Hornik et al., 1989)). □

G Implementation Details

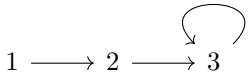
We implement CMD, CRL (CPC / NCE), and GCBC using JAX building upon the official codebase of contrastive RL (Eysenbach et al., 2022). For the QRL baseline, we use the implementation provided by the author (Wang & Isola, 2022b). Whenever possible, we used the same hyperparameters as contrastive RL (Eysenbach et al., 2022) and match the number of parameters in the model for different algorithms. We used 4 layers of 512 units of MLP as our neural network architectures and set batch size to 256. We find that using a smaller learning rate $5 \cdot 10^{-6}$ for the contrastive network is useful for improving performance. In light of Lemma 3.1, when learning the d_{SD} critic in Eqs. (14) and (15), we use a dummy action a' sampled from the marginal distribution over geometrically-discounted future actions.

We compared approaches in the offline settings across the best performance from 500k steps of training, consistent with past work (Zheng et al., 2023; Eysenbach et al., 2022). All approaches were tested with similar model sizes and runtime, and used tuned hyperparameters. Our code at `https://github.com/mnm-anonymous/qmd` features the precise configurations for the experiments.

H Worked Examples

We present a few examples of how the successor distance defined in Eq. (5) yields a valid quasimetric.

H.1 Example 1: 3-state Markov Process.

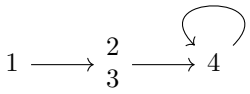


Eq. (2) Assume that the initial state is "1", a discount factor of γ, and that state "3" is absorbing. We assume that the discounted state occupancy measure states at $t = 0$, so that it includes the current time step.

$$
\begin{aligned}
p(3 \mid 3) &= 1 \\
p(2 \mid 2) &= 1 - \gamma \\
p(2 \mid 1) &= \gamma(1 - \gamma) \\
p(3 \mid 2) &= \gamma \\
p(3 \mid a) &= \gamma^2 \\
d(1, 3) &= \log p(3 \mid 3) - \log p(3 \mid 1) = \log 1 - \log \gamma^2 = 0 + 2 \log \tfrac{1}{\gamma} \\
d(1, 2) &= \log p(2 \mid 2) - \log p(2 \mid 1) = \log(1 - \gamma) - \log \gamma(1 - \gamma) = \log \tfrac{1}{\gamma} \\
d(2, 3) &= \log p(3 \mid 3) - \log p(3 \mid 2) = \log 1 - \log \gamma = \log \tfrac{1}{\gamma} \\
d(1, 2) + d(2, 3) &= 2 \log \tfrac{1}{\gamma} \geq d(1, 3) = 2 \log \tfrac{1}{\gamma}. \checkmark
\end{aligned}
$$

In this example, note that the triangle inequality is tight. This is because there is a single state that we are guaranteed to visit between states "1" and "3."

H.2 Example 2: 4-state Markov Process.



From state "1", states "2" and "3" each occur with probability 0.5.

$$
\begin{aligned}
p(4 \mid 4) &= 1 \\
p(2 \mid 2) &= 1 - \gamma \\
p(2 \mid 1) &= \tfrac{1}{2}(1 - \gamma)\gamma \\
p(4 \mid 1) &= \gamma^2 \\
p(4 \mid 2) &= \gamma \\
d(1, 2) &= \log p(2 \mid 2) - \log p(2 \mid 1) \\
&= \log(1 - \gamma) - \log \tfrac{1}{2}(1 - \gamma)\gamma = \log \tfrac{1}{\gamma} + \log 2 \\
d(2, 4) &= \log p(4 \mid 4) - \log p(4 \mid 2) \\
&= \log 1 - \log \gamma = \log \tfrac{1}{\gamma} \\
d(1, 4) &= \log p(4 \mid 4) - \log p(1 \mid 4) \\
&= \log 1 - \log \gamma^2 = 2 \log \tfrac{1}{\gamma} \\
d(1, 2) + d(2, 4) &= 2 \log \tfrac{1}{\gamma} + \log 2 \geq d(1, 4) = 2 \log \tfrac{1}{\gamma}. \checkmark
\end{aligned}
$$

In this example, the triangle inequality is loose. This is because we have uncertainty over which states we will visit between "1" and "4." One way to resolve this uncertainty is to aggregate states "2" and "3" together; if we did this, we'd be back at example 1, where the triangle inequality is tight.

I Action-Invariance

Let's assume that data are collected with a Markovian policy, so $p(s', a' \mid s, a) = \beta(a' \mid s')p(s' \mid s, a)$. Then CRL will learn

$$e^{f(s,a,s',a')} = \frac{p(s', a' \mid s, a)}{p(s', a')} \tag{29}$$

$$= \frac{\cancel{\beta(a' \mid s')}p(s' \mid s, a)}{\cancel{\beta(a' \mid s')}p(s')}. \tag{30}$$

Thus, if data are collected with a Markovian policy, then the optimal critic will not depend on the future actions. Note that this remains true for any parametrization of the critic (including MRN) that can represent the optimal critic.

However, the assumption on a Markovian data collection policy can be violated in a few ways:

1. In the online setting, data are collected from policies at different iterations. In this setting, conditioning on a previous state and action can give you a better prediction of a' (violating the Markov assumption) because it can allow you to infer which policy you're using.
2. In goal-conditioned settings, the data collection policy is conditioned on the goal. Conditioning on a previous state and action can leak information about the desired goal.

One way of fixing this is to apply CRL to a different data distribution. Let $p(s', a' \mid s, a)$ be given, and let $\beta(a)$ be some distribution over actions (in practice, we might use the marginal distribution over actions in the dataset). Define

$$\tilde{p}(s', a' \mid s, a) \triangleq p(s' \mid s, a)\beta(a'), \qquad \tilde{p}(s', a') \triangleq p(s')\beta(a'). \tag{31}$$

In practice, this corresponds to augmenting the CRL training examples $(s, a, s', a') \to (s, a, s', \tilde{a}')$ by resampling the future actions. Now, consider applying CRL to this new distribution:

$$e^{f(s,a,s',a')} = \frac{\tilde{p}(s', a' \mid s, a)}{\tilde{p}(s', a')} \tag{32}$$

$$= \frac{\cancel{\beta(a' \mid s')}\tilde{p}(s' \mid s, a)}{\cancel{\beta(a' \mid s')}\tilde{p}(s')}. \tag{33}$$

Thus, if we apply CRL to data augmented in this way, we're guaranteed to learn a critic function $f(s, a, s', a')$ that is invariant to a'.