Complex Skill Acquisition Through Simple Skill Imitation Learning

Pranay Pasula
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
pasula@berkeley.edu

Abstract

Humans often think of complex tasks as combinations of simpler subtasks in order to learn those complex tasks more efficiently. For example, a backflip could be considered a combination of four subskills: jumping, tucking knees, rolling backwards, and thrusting arms downwards. Motivated by this line of reasoning, we propose a new algorithm that trains neural network policies on simple, easy-to-learn skills in order to cultivate latent spaces that accelerate imitation learning of complex, hard-to-learn skills. We focus on the case in which the complex task comprises a *concurrent* (and possibly *sequential*) combination of the simpler subtasks, and therefore our algorithm can be seen as a novel approach to *concurrent hierarchical imitation learning*. We evaluate our approach on difficult tasks in a high-dimensional environment and find that it consistently outperforms a state-of-the-art baseline in training speed and overall performance.

1 Introduction

Humans have the ability to reason about complex tasks as combinations of simpler, interpretable subtasks. There are many hierarchical reinforcement learning approaches designed to handle tasks comprised of sequential subtasks [1, 2], but what if a task is made up of *concurrent* subtasks? For example, someone who wants to learn to do a backflip may consider it to be combination of sequential *and* concurrent subtasks: jumping, tucking knees, rolling backwards, and thrusting arms downwards. Little focus has been given to designing algorithms that decompose complex tasks into distinct concurrent subtasks. Even less effort has been put into finding decompositions that are made up of independent yet interpretable concurrent subtasks, even though analogous approaches have been effective on many challenging artificial intelligence problems [3, 4].

We propose a new generative model for encoding and generating arbitrarily complex trajectories. We augment the VAE objective used in [5] in order to induce latent space structure that captures the relationship between a behavior and the subskills that comprise this behavior in a disentangled and interpretable way. We evaluate both the original and modified objectives on a moderately complex imitation learning problems, in which agents are trained to perform behaviors after being trained on subskills that qualitatively comprise those behaviors.

2 Embedding and reconstructing trajectories

We use a conditional variational autoencoder (CVAE) [6, 7] to learn a semantically-meaningful low-dimensional embedding space that can (1) help an agent learn new behaviors more quickly, (2) be sampled from to generate behaviors, (3) and shed light on high-level factors of variation (e.g. subskills) that comprise complex behaviors.

1st NeurIPS workshop on Interpretable Inductive Biases and Physically Structured Learning (2020), virtual.

Illustrated by Figure 1, our CVAE has a bi-directional LSTM (BiLSTM) [8, 9] state-sequence encoder $q_\phi(z|s_{1:T})$, an attention module [10, 11] that maps the BiLSTM output to values that parametrize the distribution from which the latent (i.e. trajectory) embedding z is sampled, a conditional WaveNet [12] state decoder $\mathcal{P}_\psi(s_{t+1}|s_t, z)$, which serves as a *dynamics model*, and a multi-layer perceptron (MLP) action decoder $\pi_\theta(a_t|s_t, z)$, which serves as a *policy*. The bidirectional LSTM network captures sequential information over the states of the trajectories, and the conditional WaveNet to handle multi-modal dynamics. We can train this CVAE by minimizing the

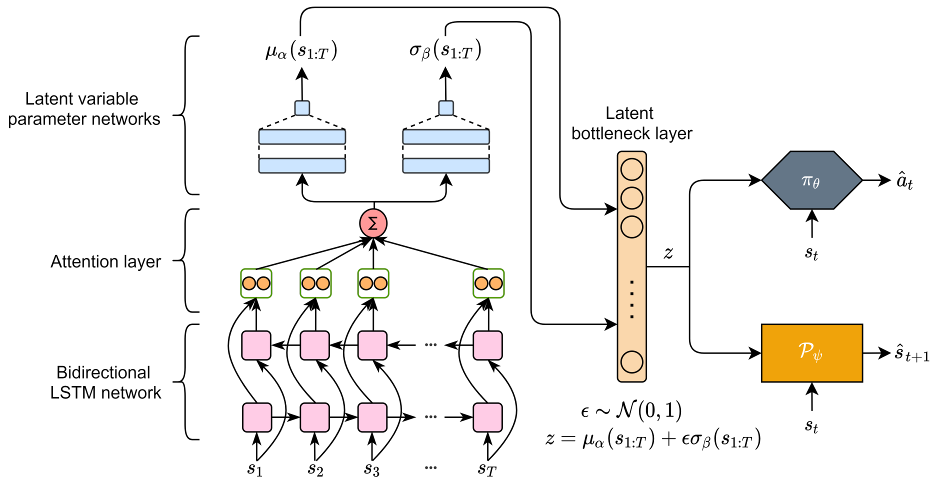

Figure 1: The conditional VAE we use to encode and generate trajectories. The *bidirectional LSTM network*, *attention layer*, and *latent variable parameter networks* comprise the *encoder* $q_\phi(z|s_{1:T})$ (or $q_\phi(z|\tau)$). To generate trajectories we sample $\epsilon \sim \mathcal{N}(0, 1)$ and compute latent vector z. Then we condition *policy* π_θ and *dynamics model* $\mathcal{P}_\psi$ on z and s_t for each timestep $t = 1, 2, \ldots, T-1$ to output $\tilde{\tau} = (\hat{s_1}, \hat{a_1}, \hat{s_2}, \hat{a_2}, \ldots, \hat{s_T}, \hat{a_T})$.

following objective

$$\mathcal{L}(\theta, \phi, \psi; \tau^i) = -\mathbb{E}_{z \sim q_\phi(z|s^i_{1:T_i})} \left[\sum_{t=1}^{T_i} \log \pi_\theta(a^i_t|s^i_t, z) + \log \mathcal{P}_\psi(s^i_{t+1}|s^i_t, z) \right] + D_{KL}\big(q_\phi(z|s^i_{1:T_i}) \parallel p(z)\big). \quad (1)$$

In Section 3 we will modify this objective in order to encourage the latent space to capture semantically meaningful relationships between complex behaviors and their subskills.

3 Shaping the latent (i.e. trajectory embedding) space

Some skills can be seen as approximate combinations of certain subskills. Training a VAE to embed and reconstruct demonstrations of these skills and subskills using (1) would generally result in an embedding space with no clear relationship between skill and subskill embedding, especially if the dimensionality of the latent space is large or the number of demonstrated behaviors is small.

Motivated by semantically meaningful latent representations found in other work [13], we aim to induce a latent space structure so that a behavior embedding is the sum of the subskill embeddings. Concretely, if z_A is a backflip embedding and z_a, z_b, z_c, z_d are embeddings corresponding to jumping, tucking knees, rolling backwards, and thrusting arms downwards, we want $z_A = z_a + z_b + z_c + z_d$. An example of such latent space restructuring is shown in Figure 2. However, VAEs model probability distributions, so enforcing equality between one instance of a behavior and one instance of its subskills is insufficient. Instead, we want the random variables (RVs) representing the embeddings of the subskills to relate to the RV representing the embedding of the behavior comprised of those subskills. Another way to do this is to relate the subskill embedding RVs with the RV representing the trajectory generated by decoder networks $\mathcal{P}_\psi$ and π_θ when conditioned on an embedding of the corresponding complex behavior, as shown in 1.

Suppose τ_V is a behavior comprised of M subskills $\{\tau_{(1)}, \tau_{(2)}, \ldots, \tau_{(M)}\}$. Let $\tilde{\tau}_v = (\hat{s}_1, \hat{a}_1, \hat{s}_2, \hat{a}_2, \ldots, \hat{s}_T, \hat{a}_T)_v$ represent the trajectory generated from an embedding corresponding to τ_v. Define $V = z_1 + z_2 + \cdots + z_M$, where $z_i \sim q_\phi(z|s_{(i),\, 1:T_{(i)}})$. To train the encoder $q_\phi(z|s_{1:T})$,

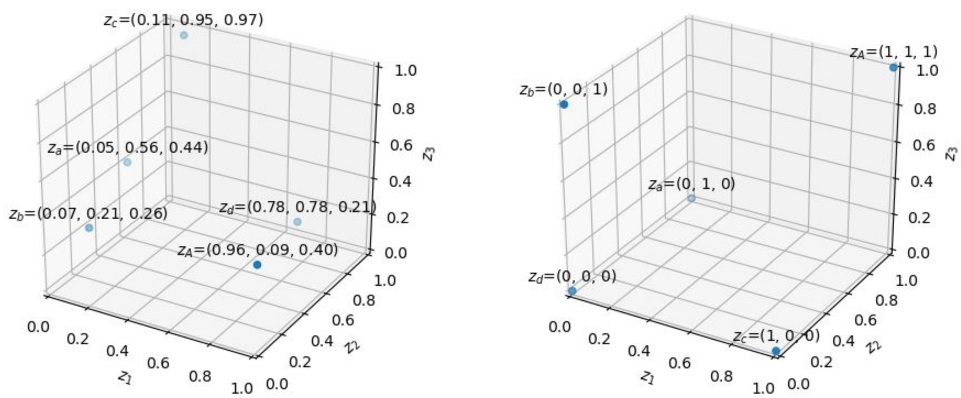

Figure 2: An example of latent space restructuring. *Left:* original latent space. *Right:* hypothetical latent space induced by our approach (created intentionally for illustrative purposes).

state decoder $\mathcal{P}_\psi(\hat{s}_{t+1}|s_{t-1}, z)$, and action decoder $\pi_\theta(\hat{a}_t|s_t, z)$ simultaneously, we aim to maximize the mutual information between V and $\tilde{\tau}_v$, which can be expressed as

$$I(V;\tilde{\tau}_v) = H(V) - H(V|\tilde{\tau}_v) \tag{2}$$

$$= -\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \mathbb{E}_{V\sim p(V|\tilde{\tau}_v)}\big[\log p(V|\tilde{\tau}_v)\big]. \tag{3}$$

If the latent variable prior distribution $p(z_i)$ is Gaussian for all $i = 1, 2, ..., M$, the entropy $H(V)$ is easy to compute, with an analytical solution under minor assumptions. We describe how to evaluate $H(V)$ in Appendix B. Going forward we omit the subscript in $\tilde{\tau}_v$ and hats in $\mathcal{P}(\hat{s}_{t+1}|\cdot)$ and $\pi(\hat{a}_t|\cdot)$.

3.1 Lower bounding mutual information through variational inference

We can't compute (3) directly because we don't have access to the true posterior distribution $p(V|\tilde{\tau})$. Therefore, in an approach similar to that of [3], we instead introduce a distribution $Q(V|\tilde{\tau})$ as a variational approximation to $p(V|\tilde{\tau})$ to get $L_I(\tilde{\tau}, Q)$, a variational lower bound of $I(V;\tilde{\tau})$,

$$\begin{aligned} L_I(\tilde{\tau}, Q) &= \mathbb{E}_{V\sim p(V), \tau\sim\tilde{\tau}|V}\big[\log Q(V|\tau)\big] + H(V) \\ &= \mathbb{E}_{\tau\sim\tilde{\tau}}\big[\mathbb{E}_{V\sim p(V|\tau)}\big[\log Q(V|\tau)\big]\big] + H(V) \\ &\leq I(V;\tilde{\tau}) \end{aligned}$$

But, unlike in [3], Q is *not* distribution approximated by q_ϕ in our CVAE. Furthermore, even though embedding variables $z_1, z_2, ..., z_M$ are independent, they are *not* conditionally independent given $\tilde{\tau}$. Therefore we *cannot* simply replace $Q(V|\tilde{\tau})$ with $\sum_{i=1}^{M} q(z_i|\tilde{\tau})$ and instead may need to use variational inference again to find $Q(V|\tilde{\tau})$, which requires training an additional VAE. Fortunately by learning a reasonably good approximation to $p(z|\tilde{\tau})$, we can avoid this additional expense.

3.2 Lower bounding mutual information without variational inference

We derive a simpler lower bound to $I(V;\tilde{\tau})$ that allows us to circumvent the time and memory costs associated with training a VAE to model $Q(V|\tilde{\tau})$. We show the main result (4) here, and provide our derivation of this result in Appendix A.

$$I(V;\tilde{\tau}) \gtrapprox -\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M}\log q_\phi(z_{n,i}|\tilde{\tau}) \tag{4}$$

By maximizing this lower bound, we (approximately) maximize $I(V;\tilde{\tau})$.

3.3 Regularization with variational approximation

To encourage a semantically meaningful relationship between a behavior embedding and this behavior's subskill embeddings, we regularize the objective in (1) with $L_I(\tilde{\tau}, Q_\alpha)$ to get

$$\begin{aligned} \mathcal{L}(\theta, \phi, \psi; \tau^i) = -\mathbb{E}_{z\sim q_\phi(z|s^i_{1:T_i})}&\left[\sum_{t=1}^{T_i}\log \pi_\theta(a^i_t|s^i_t, z) + \log \mathcal{P}_\psi(s^i_{t+1}|s^i_t, z)\right] \\ &+ D_{KL}\big(q_\phi(z|s^i_{1:T_i}) \parallel p(z)\big) + \lambda L_I(\tilde{\tau}, Q_\alpha), \end{aligned} \tag{5}$$

where $\lambda > 0$ is a hyperparameter that controls the trade-off between original objective and degree of latent space shaping.

3.4 Regularization without variational approximation

If we want to avoid performing potentially expensive variational inference, we can use (7), the result we derived earlier, in place of $L_I(\tilde{\tau}, Q)$,

$$\mathcal{L}(\theta, \phi, \psi; \tau^i) = -\mathbb{E}_{z \sim q_\phi(z|s^i_{1:T_i})}\left[\sum_{t=1}^{T_i} \log \pi_\theta(a^i_t|s^i_t, z) + \log \mathcal{P}_\psi(s^i_{t+1}|s^i_t, z)\right]$$
$$+ D_{KL}\big(q_\phi(z|s^i_{1:T_i}) \parallel p(z)\big) + \lambda\Bigg(-\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M} \log q_\phi(z_{n,i}|\tilde{\tau})\Bigg). \quad (6)$$

As shown in Appendix B, the inner expectation in (6) can be evaluated analytically if the latent variables $\{z_i\}_{i=1}^M$ are independent and normally distributed—the standard case with VAEs.

4 Experiments and results

We evaluate our approach on a 197-dimensional state and 34-dimensional action space humanoid simulated in Bullet [14]. We use policies that were pre-trained by [15] to perform *kick*, *spin*, and *jump*, as subskills that qualitatively comprise the behavior *spin kick*. We also take a similar approach for the behavior *backflip*. For each objective function of interest, we train a set of five VAEs on the subskills. We evaluate the training process of each set of VAEs by considering the similarity between the generated trajectories and the pre-trained *spin kick* and *backflip* policy demonstrations. Results of the mean squared error (MSE) between the generated and demonstration states averaged over 5 different random seeds are shown in Figure 3. We find that our approaches attain better

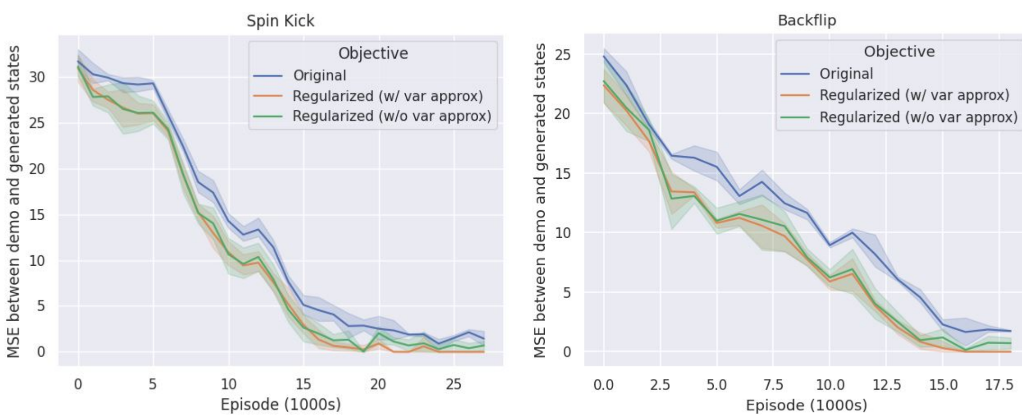

Figure 3: MSE (lower is better) between demonstration states and generated states on the DeepMimic *spin kick* and *backflip* tasks averaged over 5 different random seeds. ***Regularized* denotes *our* approaches (5), (6), and *Original* denotes the *state-of-the-art* baseline (1).**

overall performance and train faster than the baseline algorithm. This suggests that we can bootstrap the learning of difficult tasks by training agents on simpler, related subtasks while inclining their representations toward certain hierarchical structures.

5 Discussion and future work

We explored the idea of inducing certain latent structure through the maximization of mutual information between generated behaviors and embeddings of the subskills that qualitatively comprise those behaviors, which, to the best of our knowledge, has not yet been investigated. Though our algorithm outperformed the state-of-the-art baseline, there is much room for future work. A larger number of behaviors, such as those put forth by [16], could be trained at once, both to constrain the latent space and to enrich the pool of subskills from which to train on and inspect relationships between. The non-variational mutual information approximation could be compared to the variational one in order to quantify accuracy. Interpolations within the convex hull of subskill embeddings could be used to fine-tune known behaviors or generate completely new behaviors.

References

[1] Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

[2] George Konidaris and Andrew G Barto. Building portable options: Skill transfer in reinforcement learning. In *IJCAI*, volume 7, 2007.

[3] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[4] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[5] Ziyu Wang, Josh S Merel, Scott E Reed, Nando de Freitas, Gregory Wayne, and Nicolas Heess. Robust imitation of diverse behaviors. In *Advances in Neural Information Processing Systems*, pages 5320–5329, 2017.

[6] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015.

[7] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9 (8):1735–1780, 1997.

[9] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[11] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, 2016.

[12] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[14] Erwin Coumans. Bullet, 2015. URL `http://bulletphysics.org`.

[15] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018.

[16] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning*, pages 1094–1100, 2020.

A Derivation of mutual information lower bound without variational information

We derive a simpler lower bound to $I(V;\tilde{\tau})$ that allows us to circumvent the time and memory costs associated with training a VAE to model $Q(V|\tilde{\tau})$.

Theorem 1. *Let V be the sum of embeddings $z_1, z_2, \ldots, z_M$, and let p and q_ϕ respectively be true and estimated distributions of z conditioned on $\tilde{\tau}$. Assuming that p and q are well-defined for all $\tilde{\tau}$, if $p(z|\tilde{\tau}) \approx q_\phi(z|\tilde{\tau})$, we have*

$$I(V;\tilde{\tau}) \ \gtrapprox\ -\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M}\log q_\phi(z_{n,i}|\tilde{\tau}), \tag{7}$$

in which the approximate inequality approaches inequality as the number of samples N approaches infinity.

Proof. For clarity in the following derivation, let $V_p = \sum_{i=p}^{M} z_i$. Then we have

$$\begin{aligned}
H(V|\tilde{\tau}) &= H(V_1|\tilde{\tau})\\
&= H(z_1 + z_2 + \cdots + z_M|\tilde{\tau})\\
&= H(z_1|\tilde{\tau}) + H(z_1 + z_2 + \cdots + z_M|z_1,\tilde{\tau}) - H(z_1|z_1 + z_2 + \cdots + z_M,\tilde{\tau})\\
&= H(z_1|\tilde{\tau}) + H(z_2 + z_3 + \cdots + z_M|z_1,\tilde{\tau}) - H(z_1|z_1 + z_2 + \cdots + z_M,\tilde{\tau})\\
&\leq H(z_1|\tilde{\tau}) + H(z_2 + z_3 + \cdots + z_M|\tilde{\tau}) - H(z_1|z_1 + z_2 + \cdots + z_M,\tilde{\tau})\\
&= H(z_1|\tilde{\tau}) + H(V_2|\tilde{\tau}) - H(z_1|V_1,\tilde{\tau})
\end{aligned}$$

By rolling out $H(V_p|\tilde{\tau})$ recursively for $p = 1, 2, 3, ..., M-1$, we get

$$\begin{aligned}
H(V|\tilde{\tau}) &\leq \sum_{i=1}^{M}\left[H(z_i|\tilde{\tau}) - H(z_i|V_i,\tilde{\tau})\right]\\
&\leq \sum_{i=1}^{M} H(z_i|\tilde{\tau})\\
&= \sum_{i=1}^{M} -\mathbb{E}_{z_i\sim p(z_i|\tilde{\tau})}\left[\log p(z_i|\tilde{\tau})\right]\\
&\approx \sum_{i=1}^{M} -\mathbb{E}_{z_i\sim q_\phi(z_i|\tilde{\tau})}\left[\log q_\phi(z_i|\tilde{\tau})\right]
\end{aligned}$$

since $p(z|\tilde{\tau}) \approx q_\phi(z|\tilde{\tau})$. Plugging this result into (2) allows us to lower bound $I(V;\tilde{\tau})$ as follows,

$$\begin{aligned}
I(V;\tilde{\tau}) &\geq -\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \sum_{i=1}^{M}\mathbb{E}_{z_i\sim p(z_i|\tilde{\tau})}\left[\log p(z_i|\tilde{\tau})\right]\\
&\approx -\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \sum_{i=1}^{M}\mathbb{E}_{z_i\sim q_\phi(z_i|\tilde{\tau})}\left[\log q_\phi(z_i|\tilde{\tau})\right],
\end{aligned}$$

and we can obtain an unbiased estimate of the second term by sampling $z_i \sim q_\phi(z_i|\tilde{\tau})$ to get

$$I(V;\tilde{\tau}) \ \gtrapprox\ -\mathbb{E}_{V\sim p(V)}\big[\log p(V)\big] + \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M}\log q_\phi(z_{n,i}|\tilde{\tau}),$$

in which $x \gtrapprox y$ denotes that x is approximately greater than or equal to y. □

By maximizing the lower bound in (7), we (approximately) maximize $I(V;\tilde{\tau})$.

B Entropy evaluation for sums of subskill embeddings

Computing the entropy for an arbitrary distribution may be difficult, but by setting X to be a Gaussian RV—the standard choice for VAE encoders—$H(X)$ has the simple, closed-form expression

$$H(X) = \frac{1}{2}(1 + \ln(2\pi\sigma_X^2)),$$

where σ_X is the standard deviation of X. We choose $q_\phi(z|s_{1:T})$ to parametrize a Gaussian distribution and assume that state sequences from different subskills are sufficiently unrelated so that they can be considered statistically independent. This is generally a safe assumption because even minor differences in subskills will tend to place trajectories corresponding to different skills in very different locations within the trajectory space. It follows that V is the sum of Gaussian RVs and has the simple form

$$V \sim \mathcal{N}(\mu_{z_a} + \mu_{z_b} + \cdots + \mu_{z_M},\ \sigma_{z_a}^2 + \sigma_{z_b}^2 + \cdots + \sigma_{z_M}^2),$$

and the entropy of V is

$$H(V) = \frac{1}{2}(1 + \ln(2\pi(\sigma_{z_a}^2 + \sigma_{z_b}^2 + \cdots + \sigma_{z_M}^2))). \tag{8}$$