역학 감지형 임베딩을 위한 상위행동 공간에서의 등축 투영 정규화

조태현, 이도혁, 이정우

서울대학교

{talium, dohyeok}@cml.snu.ac.kr, junglee@snu.ac.kr

Isometric regularization for high-level actions on dynamic-aware embeddings

Taehyun Cho, Dohyeok Lee, Jungwoo Lee

Seoul National University

요 약

본 논문은 강화학습에서의 자기지도 표현학습(self-supervised representation learning)에서의 상태 혹은 행동 수열을 임베딩하여 샘플 효율성을 높이는 것을 목표로 한다. 좋은 표현력을 가지는 것은 에이전트가 적은 샘플로부터 주변의 학습을 쉽게 일반화시키는 것에 도움을 주곤 한다. 우리는 지금까지의 임베딩과는 달리 초기 데이터와 행동-상태 공간의 내재적 구조로의 왜곡이 최소화되는 접근에 대해 논의해볼 것이다. 이에 따라, 상위행동 공간에서의 기하학적 구조를 보존하도록, 인코더와 디코더에 여러 종류의 등축 투영 정규화(isometric regularization)를 적용해볼 것이며, Mujoco 환경에서 기존 알고리즘과 성능을 비교할 것이다.

Ⅰ. 서 론

본 논문에서는 심층강화학습(Deep Reinforcement Learning)에서 넓은 표현력을 지녔지만, 낮은 샘플 효율성을 극복하기 위해 환경과의 상호작용으로 환경의 역학(dynamics)를 효율적으로 익히는 방법 중 하나인 DynE (Dynamic-aware Embedding) 모델을 다루고 있다. 가공되지 않은 데이터 혹은 픽셀과 달리, 환경은 잠재된 저차원 공간에서 기술될 수 있으며, 이는 높은 샘플 효율성, 일반성, 그리고 강건성을 확보할 수 있는 매개체가 된다. 최근까지의 논문은 상태-행동 쌍에 대해 다음 행동으로 임베딩되는 방법에 집중하였다면, 본 논문은 샘플 효율성을 더 높이기 위한 등축 투영 정규화 기법을 적용하는 것을 살펴볼 것이다.[1]

Ⅱ. 본론

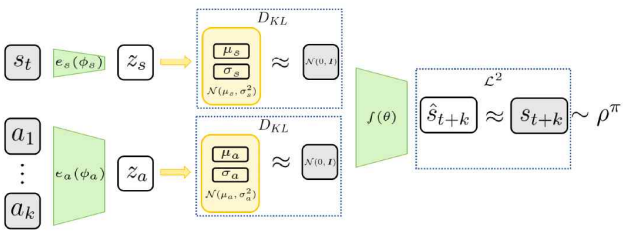

그림 1 DynE 구조 도식화

본 논문에서는 MDP (S,A,r,P,γ)에서 다루며, $k-$스텝 행동 수열에 대한 전치함수를 $s_{t+k}\sim P(\cdot|s_t,a_{1:k})$로 표기할 것이다. 또한, 완만한(smooth) 매핑 $f:M\to N,\ x\in R^m\mapsto y\in R^n$에 대해, x에서의 자코비안(Jacobian) 행렬을 $J(x)=\left(\frac{\partial f^i}{\partial x^j}(x)\right)\in R^{n\times m}$으로 표기한다.

[4]에 따라 저차원 상태 공간 $z_s\in Z_s$와 행동 공간 $z_a\in Z_a$로 매핑시키는 인코더를 각각 e_s와 e_a로 표기하며, 각각은 매개변수 ϕ_s와 ϕ_a로 나타낸다. 행동 정책(behavior policy) $\pi(s'|s,a_{1:k})$를 통해 얻게 된 주변확률분포 (Marginal distribution)은 ρ^π로 표기한다. 이때, DynE의 목적함수는 다음과 같이 나타낼 수 있다.

$$L(\phi_s,\phi_a,\theta)=E_{s,a_{1:k},s'\sim\rho^\pi}[-\log P(s'|z_s,z_a;\theta)$$
$$+\alpha D_{KL}(e_s(s;\phi_s)\|N(0,I))+\beta D_{KL}(e_a(a_{1:k};\phi_a)\|N(0,I))$$

이는 VAE(Variantional Autoencoder)의 구조와 유사하며, 첫 항은 상태, 행동에 대한 잠재변수를 통해 다음 상태를 예측을 위한 식이며, 두 번째, 세 번째 항은 가공되지 않은 상태, 행동을 압축하는 과정을 나타낸 식이다. 실험환경에서는 $P(s'|z_s,z_a;\theta)$를 평균이 $f(z_s,z_a;\theta)$인 정규분포로 설계하며, 각각의 인코더 또한 정규분포 $N(\mu_s,\sigma_s^2),N(\mu_a,\sigma_a^2)$로 치환시킬 것이다. 또한 행동 정책 $\pi(s'|s,a_{1:k})$는 균등 분포로 세팅하였다.

학습이 끝난 인코더의 매개변수를 $\overline{\phi_a}$라 할 때, 디코더 $d_a(\psi_a)$는 아래와 같은 목적함수로부터 학습이 진행된다.

$$L(\psi_a)=E_{z_a\sim N(0,I)}\left[\|e_a(d_a(z_a;\psi_a);\overline{\phi_a})-z_a\|_2^2+\eta\|d_a(z_a;\psi_a)\|_2^2\right]$$

기존의 autoencoder 구조와 달리, 복원을 위한 목적함수는 z_a에서 이루어지고 있다. 여러 해가 나오는 것을 방지하기 위하여, norm이 최소화되도록 정규화를 부여하였으며, η는 10^{-4}으로 설정하였다. 토대가 되는 알고리즘은 TD3[1]와 DPG[2]를 사용하였으며, 상위행동에 의한 정책 μ^{DynE}과 재정의된 벨만 방정식 Q^{DynE}은 아래와 같다.

$$Q^{DynE}(e_s(s_t),z_{a,t},i)=\sum_{j=0}^{k-i-1}\gamma^j r_{t+j}$$
$$+\gamma^{k-i}Q^{DynE}\left(e_s(s_{t+k-i}),\mu^{DynE}(e_s(s_{t+k-i})),i=0\right)$$

이로부터 보상함수 J_π의 그래디언트(gradient)는 다음과 같다.

$$\nabla_w J_\pi(\mu_w^{DynE}) \approx E_{s\sim\rho^\pi}\left[\nabla_w \mu_w^{DynE}(e_s(s)), \nabla_{z_a} Q^{DynE}(e_s(s),z,0)|_{z=\mu_w^{DynE}(e_s(s))}\right]$$

우리는 아래의 세 등축 투영 정규화 방법론을 앞선 목적함수에 추가적으로 부여하여 학습하였다.

$$L_{is}(f(x),H,G) = \sum_i (\lambda^i_{J^T HJG^{-1}} - 1)^2$$

$$L_{is-\log}(f(x),H,G) = \sum_i \log^2(\lambda^i_{J^T HJG^{-1}})$$

$$L_{is-harmonic}(f(x),H,G) = Tr(J^T HJG^{-1})$$

$\lambda^i_{J^T HJG^{-1}}$은 $J^T HJG^{-1}$의 $i-$번째 eigenvalue를 나타내며, 각각은 eigenvalue를 모두 1로 만드는 것으로 identity matrix가 되도록 유도한다. 이 때, H, G는 모두 identity matrix I로 설정하였다.

우리는 [3]에서 실험한 환경인 ReacherVertical-v2와 ReacherTurn-v2 에서 알고리즘 성능을 확인하였다. 그림 2에서의 ReacherVertical-v2 환경은 주황색 effector가 빨간색 공에 해당하는 지점에 도달하는 것이 목표인 작업으로 학습 소요시간이 2-3시간 정도인 비교적 단순한 환경이다. 반면, 그림 3에서의 ReacherTurn-v2 환경은 파란색 축에 의해 고정된 강체를 움직이는 것으로 이전보다 복잡한 목표를 요구하며, 학습 소요시간이 10시간 이상이 요구되었다.

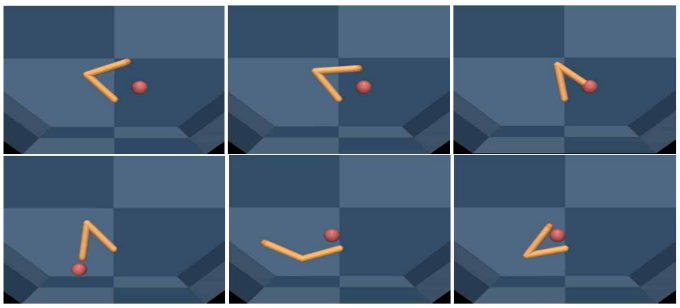

그림 2 ReacherVertical-v2 환경

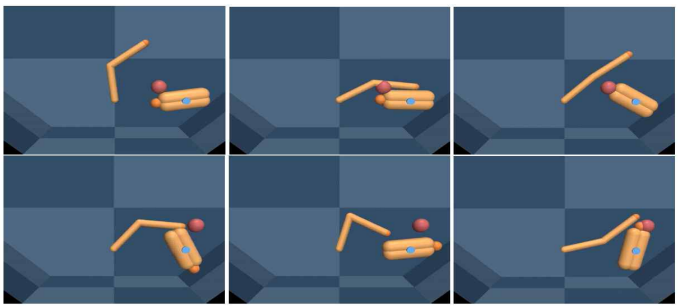

그림 3 ReacherTurn-v2 환경

우리는 TD3를 기반으로 한 DynE-TD3에 정규화를 적용하는 것으로 성능을 비교하였고, 각각을 iso-TD3, iso-log-TD3, iso-harmonic-TD3로 명명하였다. 각 성능을 나타내는 곡선은 3개의 실험에 대한 결과이다. 그림 4에서 볼 수 있듯이, DynE-TD3에 비해 iso-TD3, iso-log-TD3는 눈에 띄는 성능 향상을 보였으며, iso-harmonic-TD3는 큰 변화를 보이지 않았다.

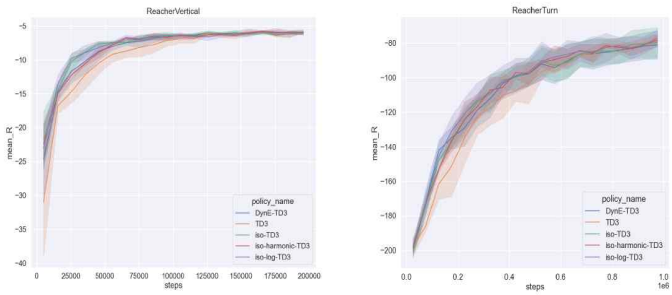

그림 4 DynE과 여러 등축 정규화 기법을 적용한 방법의 성능 비교.

Ⅲ. 결론

본 논문에서는 DynE 알고리즘에 잠재공간으로의 임베딩 왜곡을 최소화하는 여러 종류의 등축 투영 정규화를 실험하였다. iso 및 iso-log와 같은 유효한 방법론이 있는 반면, iso-harmonic의 경우에는 큰 변화를 주지 못하였는데 이에 대한 원인을 찾는 것이 후속 연구가 될 것이다.

ACKNOWLEDGMENT

This work was supported by National Research Foundation of Korea (NRF, 2021R1A2C2014504(40%)), Institute of Information & communications Technology Planning & Evaluation (IITP, 2021-0-00106(30%), 2021-0-02068(30%)) grant funded by the Ministry of Science and ICT (MSIT), the Agency For Defense Development by the Korean Government(UD190031RD), INMAC, and BK21-plus.

참 고 문 헌

[1] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In International conference on machine learning, pages 1587-1596, PMLR, 2018

[2] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In International conference on machine learning, pages 387-395, PMLR, 2014.

[3] Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Hosh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dm_control: Software and tasks for continuous control. Software Impacts, 6:100022, nov 2020. doi: 10.1016/j.simpa.2020.1000022. URL https://doi.org/10.1016%2Fj.simpa.2020.100022.

[4] William Whiteney, Rajat Agarwal, Kyunghyun Cho, and Abhinav Gupta. Dynamics-aware embeddings. arXiv preprint arXiv:1908.09357, 2019