Understanding When Dynamics-Invariant Data Augmentations Benefit Model-free Reinforcement Learning Updates

Nicholas E. Corrado, Josiah P. Hanna (University of Wisconsin – Madison)
ncorrado@wisc.edu

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

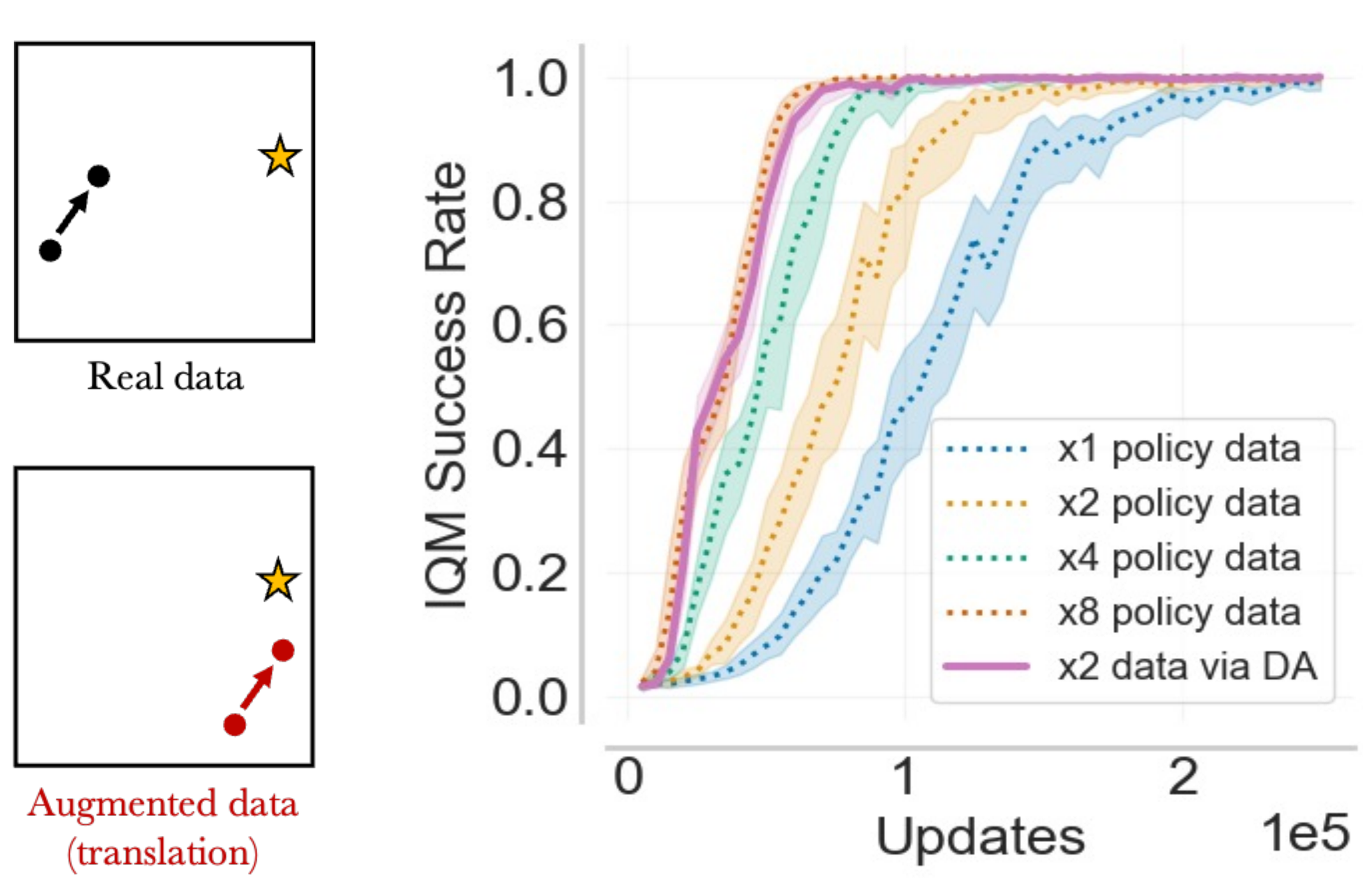
We study how different aspects of data augmentation affect the data efficiency of RL and provide practical guidelines on how to most effectively apply data augmentation

Background & Motivation

Data augmentation (DA) is a technique in which RL agents generate additional synthetic experience by transforming real experience collected through environment interaction.

While prior work has demonstrated that incorporating augmented data directly into model-free RL updates can improve data efficiency, we lack a clear understanding of when and why augmented data improves data efficiency.

Our goal: understand which aspects of DA improve data efficiency and provide guidelines on how to most effectively apply DA.



In a toy navigation task, doubling the agent's data via DA is just as good as learning from x8 as much real data collected by the agent!

Aspects of Data Augmentation

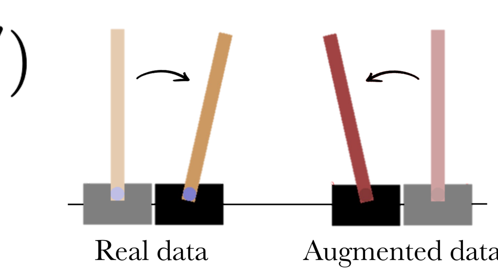
We focus on sparse-reward RL tasks with dynamics-invariant augmentations – augmentations which generate realistic data that respect the tasks dynamics and reward structure:

$$(\boldsymbol{s}, \boldsymbol{a}, r, \boldsymbol{s}') \longrightarrow \boxed{f} \longrightarrow (\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{a}}, \tilde{r}, \tilde{\boldsymbol{s}}')$$

Real data — Data augmentation function — Augmented data

$$p(\tilde{\boldsymbol{s}}'|\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{a}}) > 0 \qquad \tilde{r} = r(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{a}})$$

Augmented data is realistic



CartPole dynamics are invariant under translation and reflection

We study how three aspects of data augmentation affect data efficiency:

1. Increasing state-action coverage
2. Additional reward signal

} Improves exploration

3. Decreasing the replay ratio of augmented data (# of updates per augmented sample generated)

} Diversifies the augmented data used in each update

Data Augmentation Framework

1 Data Collection

π

Collect $(\boldsymbol{s}, \boldsymbol{a}, r, \boldsymbol{s}')$ with π, and add it to the observed replay buffer

2 Data Augmentation

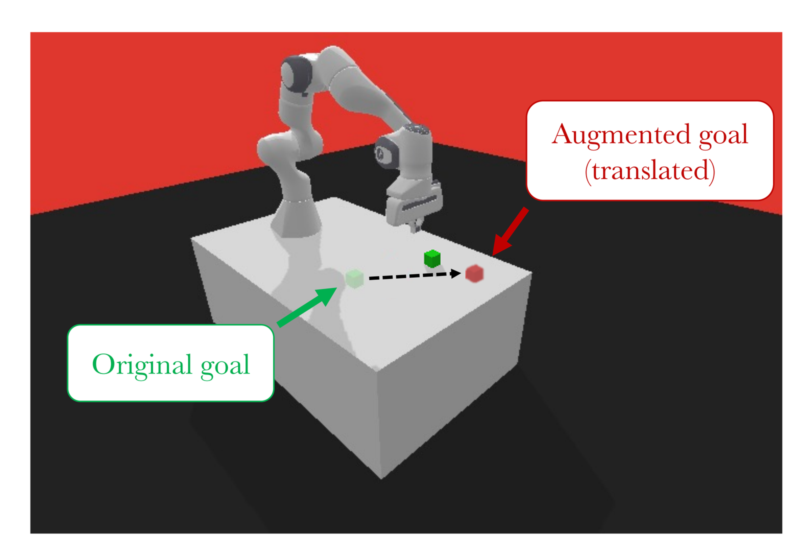
$(\boldsymbol{s}, \boldsymbol{a}, r, \boldsymbol{s}') \longrightarrow \boxed{f} \longrightarrow (\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{a}}, \tilde{r}, \tilde{\boldsymbol{s}}')$

Generate augmentations of $(\boldsymbol{s}, \boldsymbol{a}, r, \boldsymbol{s}')$ and add now to the augmented replay buffer

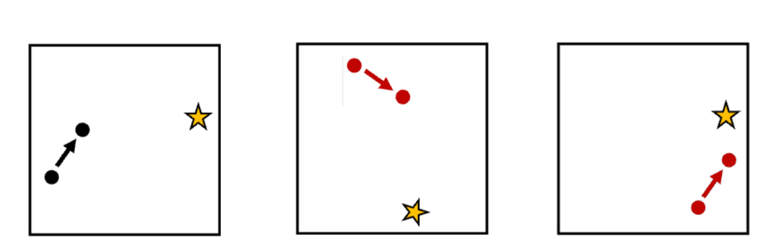
3 Policy Improvement

$\pi \longrightarrow \pi'$

Update π using a fixed split of data sampled from both replay buffers

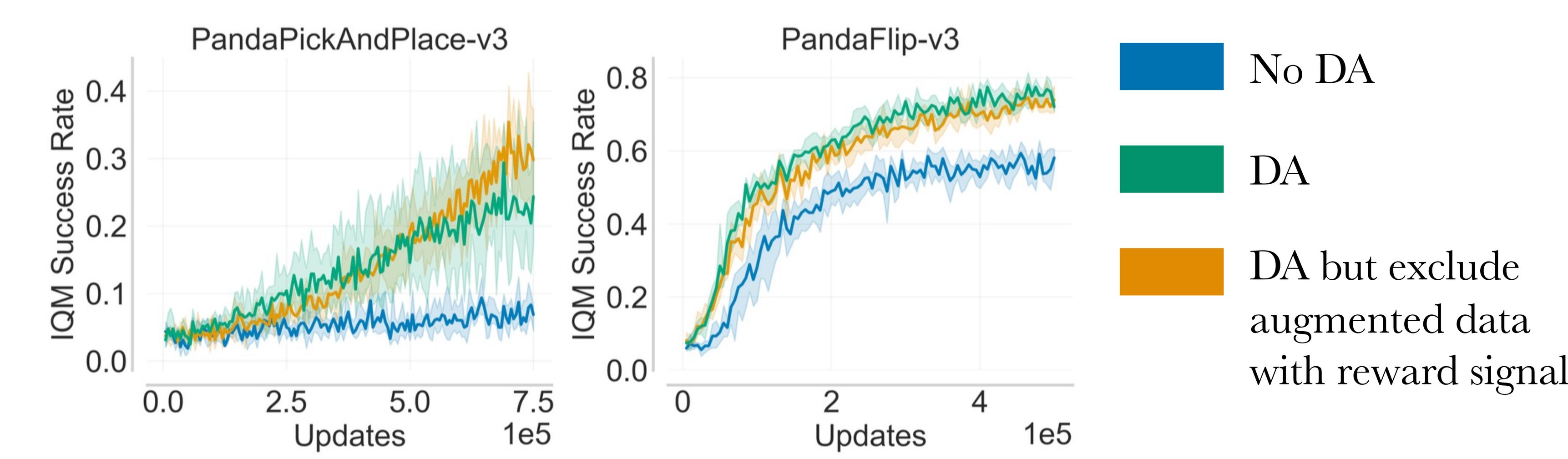


TranslateGoal: Translate the agent's goal to a new goal sampled uniformly at random from the task's goal space.

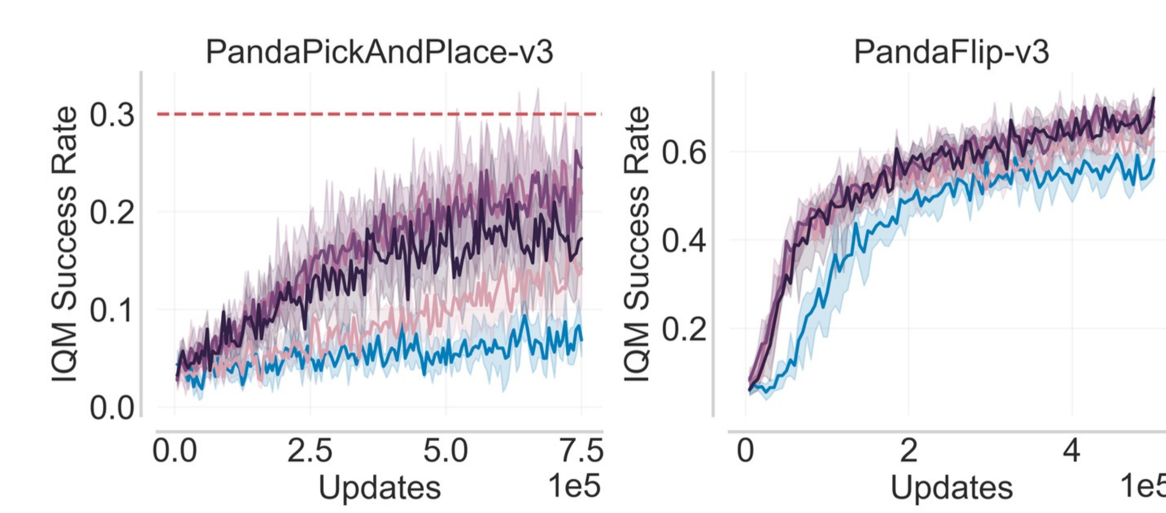


Original data — Rotated data — Translated data

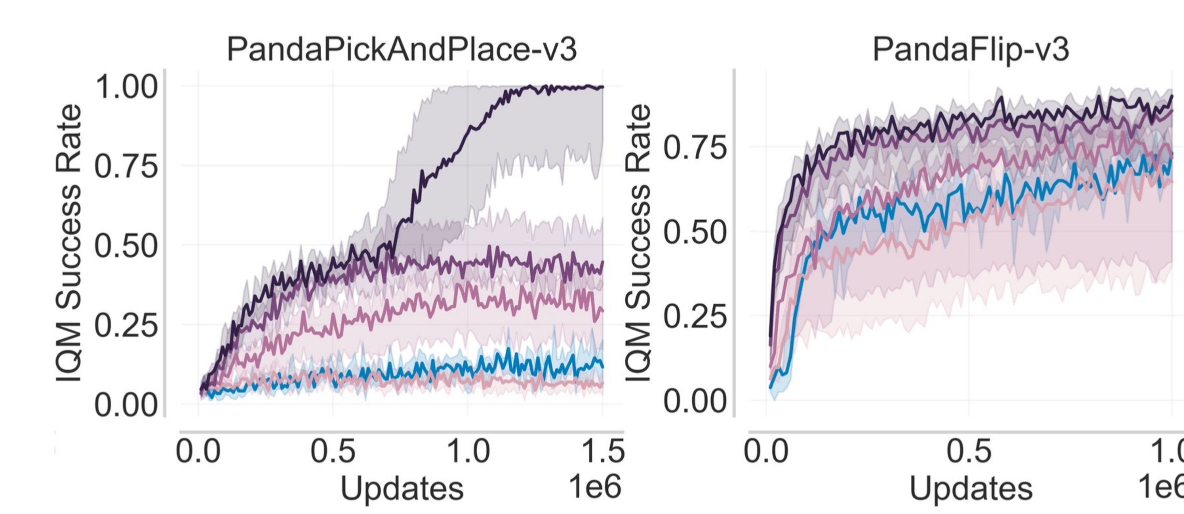
Empirical Highlights



DA often performs just as well when it increases state-action coverage without generating any additional reward signal.



Increasing the probability p of generating augmented data with reward signal often yields minimal gains.



Decreasing the replay ratio β of augmented data can dramatically improve data efficiency.

Practical Guidelines

1. When designing new augmentations, focus on increasing coverage rather than generating additional reward signal.
2. Decrease the replay ratio of augmented data.