

---

# Evaluating Counterfactual Data Augmentation in Reinforcement Learning

---

Sajan Kumar  
Purdue University

Shilpa Noushad  
Purdue University

Pratyush Uppuluri  
Purdue University

## Abstract

We present a verified, open-source reimplementation and extension of CTRL, a causal reinforcement learning method using counterfactual data augmentation. Through a validation matrix across diverse datasets (CartPole, LunarLander, MuJoCo, D4RL), we show that counterfactual augmentation is conditionally useful rather than uniformly superior, with reliability depending on generator fidelity, data regime and evaluation protocol. By comparing against a non-causal world model, we identify a critical “coverage-versus-bias” tradeoff where excessive augmentation amplifies transition inaccuracies. Finally, we fill a significant gap in the community by providing a verified, ground-up open-source implementation of the CTRL architecture to facilitate further research in causal RL.

## 1 Introduction

Reinforcement Learning (RL) has shown remarkable success in simulated settings, but its real-world adoption in domains such as healthcare, robotics, and industrial control remains limited by costly data collection and unsafe exploration. Improving **sample efficiency and exploration** has thus become a central goal in practical RL, motivating approaches that learn effectively from small, fixed datasets while preserving theoretical convergence guarantees.

The CTRL framework (*Sample-Efficient Reinforcement Learning via Counterfactual Based Data Augmentation*, NeurIPS 2020) introduces a **Structural Causal Model (SCM)** to represent environment dynamics as

$$S_{t+1} = f(S_t, A_t, U_{t+1}), \quad (1)$$

where  $U_{t+1}$  denotes exogenous noise. By inferring this latent variable for each observed transition and

reusing it to compute **counterfactual next states**  $S'_{t+1} = f(S_t, a', U_{t+1})$  for alternate actions, CTRL generates additional, causally consistent experiences without new interactions—enabling **sample-efficient “imaginative exploration”** that broadens data coverage by asking “*what if the agent had taken action  $a'$  instead of  $a$ ?*” The paper shows that if  $f$  is monotone in  $U$ , counterfactual outcomes are identifiable (Theorem 1 in [3]) and that Q-learning trained on this augmented dataset converges to the optimal value function  $Q^*$  (Theorem 2 in [3]). Together, these results make CTRL particularly compelling—uniting causal validity and reinforcement learning optimality within a single, data-efficient framework.

We implement the underlying causal mechanism via a Bidirectional Conditional GAN (BiCoGAN)[2] as in Figure 1a to enable counterfactual generation. Leveraging this framework, we conduct a ground-up reproduction on *CartPole-SD* and extend our analysis to *LunarLander-v3*, MuJoCo, and D4RL to identify specific regimes where causal augmentation succeeds or fails.

**Our contributions are as follows:**

- **Ground-Up Reproduction:** A from-scratch implementation of the CTRL pipeline (SCM data generation, BiCoGAN training, and offline D3QN+CQL).
- **Controlled Ablation Matrix:** Targeted CartPole studies over CF mixing ratio, evaluation noise, dataset size, and generator quality to isolate when CF helps and when it harms.
- **Baseline Differentiation:** Implementation of a non-causal **Base-S** world model to isolate the specific advantages of causal identifiability over probabilistic augmentation.
- **Mechanistic Analysis:** Identification of a **coverage-versus-bias tradeoff**, documenting how excessive augmentation amplifies model error in sensitive regions, leading to non-monotonic performance.
- **Cross-Domain Diagnostics:** Extension to LunarLander, MuJoCo, and D4RL-style settings to

test external validity and to identify task regimes where CF transfer is unstable.

Our results support a scoped claim: causal counterfactual augmentation is conditionally beneficial, but outcomes are sensitive to data regime, generator quality, and evaluation protocol.

### Related Work:

This paper is positioned as a *reproduction-and-analysis study*, not a new algorithmic proposal. We build on the SCM formulation of CTRL [3] and the underlying causal framework [4]. We also relate our findings to broader data-augmentation baselines such as CoDA [5] and RAD [7], and to model-based world-model perspectives [6].

To situate this work in the post-2020 literature, we explicitly acknowledge direct successors to counterfactual augmentation (MoCoDA, ACAMDA, CA-IAC, RoCoDA) [12, 13, 14, 15], survey/framework references [16, 17, 18], and foundational causal RL/representation works [22, 19, 20, 21].

## 2 Methodology

We faithfully reproduce the theoretical framework of [3], implementing the architecture from scratch due to the absence of an official repository. To mitigate bias and ensure alignment with the original theory, we conducted multiple independent experiments to validate our implementation choices against the authors’ described behavior.

In our study, we implement Algorithm 1 from [3] and replicate the core CTRL components: environment dynamics, SCM assumptions, and counterfactual augmentation. We first reproduce the CartPole-SD setting and then extend evaluation beyond the original scope to LunarLander-v3, MuJoCo, and D4RL-style offline checks to test how behavior changes under more complex domains and broader validation protocols.

### 2.1 Stochastic CartPole-SD Environment and Evaluation Protocols

We follow the SD setting from [3] and explicitly document the environment details used in our implementation. The underlying CartPole dynamics remain unchanged; we use the same 11-action parameterization, additive action noise, and noisy observations. The agent selects among 11 discrete actions  $a \in \{0, \dots, 10\}$ , which are mapped to a continuous control value  $a_{\text{cont}} = a/10$  and further perturbed by Gaussian execution noise  $\tilde{a} = a_{\text{cont}} + \epsilon$ , with  $\epsilon \sim \mathcal{N}(0, 0.05^2)$ . The resulting applied force is  $F = (2\tilde{a} - 1)F_{\text{max}}$  with  $F_{\text{max}} = 10$ . Termination is evaluated on the *clean*

next state prior to adding any stochastic noise, i.e., whenever  $|x'_{\text{clean}}| > 2.4$  or  $|\theta'_{\text{clean}}| > 12^\circ$ . After termination is checked, the environment returns a noisy observation  $s' = s'_{\text{clean}} + 0.05\eta$ , with  $\eta \sim \mathcal{N}(0, I_4)$ , subsequently clipped to the usual CartPole bounds  $x' \in [-4.8, 4.8]$  and  $\theta' \in [-0.418, 0.418]$ . Dataset generation samples random discrete actions, applies noisy actions  $\tilde{a}$ , and records  $(s, a, \tilde{a}, r, s')$  until termination.

**Clean vs. noisy evaluation.** To avoid ambiguity in the results section:

- **Clean** refers to evaluation on standard **CartPole-v1** (no SD observation noise in environment state transitions).
- **CTRL/noisy** refers to evaluation on our SD environment with built-in observation noise ( $\sigma_s = 0.05$ ) and 11-action mapping; additional action-evaluation noise is set to zero unless stated.

### 2.2 Offline D3QN + CQL Training with Real and Counterfactual Data

After training BiCoGAN (Algorithm 2), we construct the offline RL dataset by combining real SD transitions with counterfactual (CF) transitions generated via the learned  $(G, E)$  pair. For each real transition  $(s_t, a_t, r_t, s_{t+1})$ , we first infer the transition-specific latent exogenous variable using the encoder and then *reuse that inferred latent* for alternative actions:

$$\hat{u}_t = E(s_{t+1}), \quad \hat{s}' = G(s_t, a_{\text{cf}}, \hat{u}_t), \quad a_{\text{cf}} \neq a_t.$$

Each CF transition inherits reward  $r_t$  and uses a done flag based on the clean termination thresholds applied to  $\hat{s}'$ .

We evaluate two regimes:

- **Real-only**: the offline dataset consists solely of the clean SD transitions.
- **Real + CF ( $k$  per real)**: each real transition is augmented with  $k$  counterfactual transitions generated from BiCoGAN.

All states and next states are standardized using the real dataset mean and standard deviation. We train an offline Dueling D3QN agent as in Algorithm 3 with a Conservative Q-Learning (CQL) penalty, following the update structure of [3]. The value network is trained using Double DQN targets and soft target updates. All experiments, result plots, and evaluation logs used in this section can be fully reproduced using our publicly released notebook available at **Counterfactual-RL Experiments**.

### 2.3 Experiments on LunarLander-v3

For our second environment, we use the standard **LunarLander-v3** from Gymnasium, with default

physics (gravity, wind, and turbulence parameters), and no modifications to the underlying dynamics.

**BiCoGAN SCM for LunarLander.** We reuse the same Structural Causal Model (SCM) framework developed for CartPole-SD: a bidirectional conditional GAN that learns forward dynamics  $G(s_t, a_t, u_t)$  and an encoder  $E(s_{t+1})$  for recovering latent exogenous structure. The architecture, loss functions, and training schedule are identical to the CartPole version; only the state and action dimensions differ.

**Base-S World Model.** To compare SCM-based counterfactual generation with a simpler probabilistic world model, we implement *Base-S*, a Gaussian next-state-and-reward model following the formulation in the CTRL paper. Base-S conditions on the current state and a one-hot action vector and produces

$$(\mu_{s'}, \mu_r, \log \sigma_{s'}^2, \log \sigma_r^2),$$

the means and log-variances of a diagonal Gaussian distribution over next state and reward. Samples are drawn via reparameterization, and the model is trained using a Gaussian negative log-likelihood loss:

$$\mathcal{L}_{\text{NLL}} = \mathbb{E} [\mathcal{N}(s_{t+1}; \mu_{s'}, \sigma_{s'}^2) + \mathcal{N}(r_t; \mu_r, \sigma_r^2)].$$

**Training Base-S.** We train Base-S for 20 epochs using Adam with learning rate  $10^{-3}$ , weight-normalized MLP layers, and minibatch NLL over both next state and reward. This follows the training setup stated in the CTRL paper and provides a baseline generative model for producing counterfactual transitions. The trained Base-S model is then used to generate synthetic rollouts for offline RL, enabling a direct comparison with the BiCoGAN SCM approach.

### 3 Results

**Main framing.** We report results in two tiers: (i) controlled CartPole-SD offline ablations to study mechanism-level behavior, and (ii) cross-domain validation on LunarLander, MuJoCo, and D4RL-style offline evaluation for external validity. Throughout this section, *clean* means evaluation on standard CartPole-v1; *CTRL/noisy* means evaluation on the SD environment with observation noise and 11-action control. Unless otherwise noted, values are mean returns; exploratory settings with smaller seed counts are explicitly marked in tables.

#### 3.1 CartPole Offline Matrix (Controlled Analysis)

On clean CartPole, CF augmentation improves over real-only D3QN in several settings (notably  $f =$

Table 1: CartPole-SD offline evaluation (mean return  $\pm$  std over seeds; main settings use  $n = 30$ , exploratory edge fractions use  $n = 10$ ).

Agent	CF frac	Clean	CTRL (noisy)
Rainbow	-	$37.20 \pm 44.37 (n = 30)$	N/A
D3QN (real only)	0	$33.14 \pm 10.23 (n = 30)$	$17.99 \pm 1.49 (n = 30)$
D3QN + CF	0.02	$37.77 \pm 21.21 (n = 10)$	$17.15 \pm 1.47 (n = 10)$
D3QN + CF	0.05	$34.25 \pm 10.87 (n = 30)$	$17.82 \pm 1.46 (n = 30)$
D3QN + CF	0.10	$35.60 \pm 20.73 (n = 30)$	$17.73 \pm 1.37 (n = 30)$
D3QN + CF	0.25	$33.87 \pm 12.45 (n = 30)$	$18.37 \pm 1.60 (n = 30)$
D3QN + CF	0.50	$44.66 \pm 23.30 (n = 10)$	$18.47 \pm 1.53 (n = 10)$

0.10 and exploratory  $f = 0.50$ ), but gains are non-monotonic in CF fraction and have wide dispersion in some regimes. On CTRL/noisy evaluation, all D3QN variants remain tightly clustered around  $\sim 18$ , indicating limited robustness separation.

#### Key takeaways from the CartPole matrix.

- CF augmentation can improve clean-return performance, but the effect is not monotonic in CF fraction.
- Noisy evaluation remains tightly clustered across D3QN variants, indicating limited robustness gains in this setting.
- Rainbow shows high variance in this offline protocol, so the main causal comparison is between D3QN real-only and D3QN+CF variants.
- Exploratory edge fractions ( $f = 0.02$ ,  $f = 0.50$ ) use fewer seeds and should be interpreted as trend indicators, not final estimates.

#### 3.2 Ablations: When Does CF Help?

Dataset-size ablation is shown in Figure 1b. We define **small** as 100 episodes (1669 transitions), **medium** as 250 episodes (4149 transitions), and **large** as 500 episodes (8543 transitions). CF is regime-dependent: it hurts in small/medium data but helps in large data (clean means: small 35.20 vs real 44.83; medium 31.37 vs real 35.15; large 42.38 vs real 35.07).

Figure 1c summarizes CF-quality and BiCoGAN-quality effects. Raw CF gives the highest clean score in this matrix (55.80), while aggressive filtering/subsampling reduces gains. Generator quality is also monotonic in this study (weak 31.01, medium 34.52, strong 41.68 clean mean), confirming that downstream RL performance is sensitive to SCM quality.

#### 3.3 Cross-Domain Validation

Cross-domain results are mixed: CF improves LunarLander mean and 2/4 MuJoCo tasks (Ant, HalfCheetah), but degrades 2/4 MuJoCo tasks (Hopper, Walker2d). D4RL numbers are reported as budget-limited baseline checks (one CQL configuration per dataset at 100k steps, raw returns) and are not yet

sufficient for a strong causal-augmentation claim. See Table 3 and Figure 1d in Appendix.

### Task-level interpretation and statistical caveat.

The MuJoCo split is consistent with task sensitivity: Ant and HalfCheetah are more tolerant to local model error, while Hopper and Walker2d are balance-heavy and can fail under small transition inaccuracies. We therefore treat these cross-domain outcomes as diagnostic rather than definitive, because current budgets use limited seeds on several tasks ( $n = 2$  or  $n = 3$ ), and D4RL reporting is currently a low-budget sanity pass rather than a normalized-score benchmark.

### 3.4 Focused Diagnostic: Bellman-Selected CF on LunarLander-v3

To isolate one failure mode, we evaluate a Bellman-score selector that keeps only the highest-scoring counterfactual per transition. This diagnostic uses a different protocol from the cross-domain matrix and is reported separately.

Table 2: Final evaluation returns (20 episodes) using Bellman-selected counterfactuals.

Agent	CF Source	Mean Return $\pm$ Std
Real-only	None	-145.95 $\pm$ 55.89
Hybrid CTRL (best-CF)	Base-S	-326.99 $\pm$ 116.05
Hybrid CTRL (best-CF)	BiCoGAN	-262.48 $\pm$ 86.85

Even with Bellman-score selection (Table 2), counterfactual augmentation did not improve *LunarLander-v3* in our runs. Both Base-S and BiCoGAN variants remained high-variance and underperformed the Real-only baseline. Lunar lander experiments can be observed and run with code at [Counterfactual-RL LunarLander Experiments](#).

### 3.5 Interpretation and Next Methodological Steps

**Interpretation (hypothesis, not proof).** Our non-monotonic CartPole gains are consistent with a coverage-versus-bias tradeoff: limited CF can improve support, while excessive CF can amplify model error in synthetic regions. This interpretation is consistent with the high variance observed in LunarLander and with a possible Bellman-selection feedback loop where biased high-value CF samples are repeatedly over-selected [3, 8].

**Proposed methodological changes (not yet evaluated).** We propose three concrete upgrades for the next revision cycle. First, replace unconstrained GAN generators with monotonic or identifiable SCM parameterizations [24, 25]. Second, replace fixed CF mixing with uncertainty-aware acceptance and discrepancy-

aware augmentation [13, 26, 27]. Third, benchmark stronger offline RL baselines (for example, IQL and support-constrained variants) and report scaling behavior [23, 29, 28]. We also plan to add a non-causal augmentation baseline so causal benefit can be separated from generic data-augmentation effects.

### Causal invariance diagnostic (planned extension).

To separate causal mechanism learning from associative fitting, a direct next step is an invariance diagnostic under controlled environment shifts (for example, changing gravity or wind in *LunarLander-v3*) [10, 11]. The intended plot compares one-step prediction error versus shift magnitude for BiCoGAN-SCM and Base-S.

## 4 Conclusion and Future Work

This study reimplements CTRL and evaluates it under a substantially broader matrix than the original reproduction baseline. Our evidence supports a *conditional* conclusion: counterfactual augmentation can improve offline RL performance, but gains depend strongly on regime and are not universal.

In controlled CartPole analysis, CF data improves clean-return performance in several settings and shows stronger effect when dataset size and generator quality are higher. However, noisy/CTRL robustness gains are modest, and cross-domain transfer is mixed (MuJoCo has both wins and losses; D4RL remains underpowered at current budget and seed count).

**Practical takeaway.** Counterfactual augmentation should be treated as a high-variance lever that requires careful control of data quality, generator quality, and mixing ratio. It is promising, but not yet a drop-in replacement for standard offline RL pipelines.

**Future work.** To strengthen external validity, the next phase should prioritize compute-matched seed expansion and broader MuJoCo/D4RL coverage with normalized reporting. Methodological upgrades (monotonic/identifiable SCMs, uncertainty-aware CF selection, and stronger offline RL baselines) are detailed in the preceding diagnostic subsection.

**Open-Source Implementation.** All experiments described above can be fully reproduced using our public notebook and code is available [Causal-RL-Study](#).

## References

- [1] Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., Zaremba, W.: OpenAI Gym. arXiv preprint arXiv:1606.01540 (2016)
- [2] Jaiswal, A., AbdAlmageed, W., Wu, Y., Natarajan, P.: Bidirectional Conditional Generative Adversarial Networks. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. LNCS, vol. 11363, pp. 216–232. Springer, Cham (2019). 10.1007/978-3-030-20893-6\_14
- [3] Lu, C., Huang, B., Wang, K., Hernandez-Lobato, J.M., Zhang, K., Scholkopf, B.: Sample-Efficient Reinforcement Learning via Counterfactual-Based Data Augmentation. arXiv:2012.09092 (2020)
- [4] Pearl, J.: Causality. Cambridge University Press, Cambridge (2009)
- [5] Pitis, S., Creager, E., Garg, A.: Counterfactual Data Augmentation using Locally Factored Dynamics. In: Advances in Neural Information Processing Systems, vol. 33, pp. 3976–3987 (2020)
- [6] Hafner, D., Pasukonis, J., Ba, J., Lillicrap, T.: Mastering Diverse Domains through World Models. arXiv preprint arXiv:2301.04104 (2023)
- [7] Laskin, M., Srinivas, A., Abbeel, P.: Reinforcement Learning with Augmented Data. In: Advances in Neural Information Processing Systems, vol. 33, pp. 19884–19895 (2020)
- [8] Kumar, A., Zhou, A., Tucker, G., Levine, S.: Conservative Q-Learning for Offline Reinforcement Learning. In: Advances in Neural Information Processing Systems, vol. 33 (2020)
- [9] Vygotsky, L.S.: Mind in Society: The Development of Higher Psychological Processes. Harvard University Press, Cambridge (1978)
- [10] Peters, J., Buhlmann, P., Meinshausen, N.: Causal Inference by Using Invariant Prediction: Identification and Confidence Intervals. *Journal of the Royal Statistical Society: Series B* **78**(5), 947–1012 (2016). 10.1111/rssb.12167
- [11] Scholkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., Mooij, J.M.: On Causal and Anticausal Learning. In: Proceedings of the 29th International Conference on Machine Learning (2012)
- [12] Pitis, S., Creager, E., Garg, A.: Model-Based Counterfactual Data Augmentation for Offline Reinforcement Learning. In: Advances in Neural Information Processing Systems, vol. 35 (2022)
- [13] Sun, Y., Wu, T., Yu, X., Lu, C., Zhang, K.: Adversarial Causal Model-Based Data Augmentation for Offline Reinforcement Learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38(14), pp. 15653–15661 (2024). 10.1609/AAAI.V38I14.29442
- [14] Armengol Urpi, A., Hernandez-Lobato, J.M., Lu, C.: Counterfactual Augmentation via Action Influence in Offline Reinforcement Learning. In: Proceedings of the 41st International Conference on Machine Learning. PMLR, vol. 235, pp. 1709–1729 (2024)
- [15] Ameperosa, N., Curi, S., Janner, M., Garg, A.: RoCoDA: A Unified Framework for Invariance, Equivariance, and Causality in Robot Learning. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 17111–17118 (2025). 10.1109/ICRA55743.2025.11128694
- [16] Deng, Z., Zhang, D., Li, Y., Bareinboim, E.: Causal Reinforcement Learning: A Survey. *Transactions on Machine Learning Research* (2023)
- [17] Zeng, G., Li, X., Wang, Q., Zhang, K.: A Survey on Causal Reinforcement Learning. *IEEE Transactions on Neural Networks and Learning Systems* (2025). 10.1109/TNNLS.2024.3403001
- [18] Bareinboim, E., Zhang, J., Lee, S.: An Introduction to Causal Reinforcement Learning. Tech. rep., Causal AI Lab (2025). <https://causalai.net/r65.pdf>
- [19] Buesing, L., et al.: Woulda, Coulda, Shoulda: Counterfactually-Guided Policy Search. In: International Conference on Learning Representations (2019)
- [20] Huang, B., Lu, C., Wang, K., Zhang, K.: Learning Action-Sufficient State Representation for Control from Pixels. In: Proceedings of the 39th International Conference on Machine Learning (2022)
- [21] Huang, B., Lu, C., Zhang, K.: AdaRL: What, Where, and How to Adapt in Transfer Reinforcement Learning. In: International Conference on Learning Representations (2022)
- [22] Scholkopf, B., et al.: Toward Causal Representation Learning. *Proceedings of the IEEE* **109**(5), 612–634 (2021). 10.1109/JPROC.2021.3058954

- [23] Kostrikov, I., Nair, A., Levine, S.: Offline Reinforcement Learning with Implicit Q-Learning. In: International Conference on Learning Representations (2022)
- [24] Nasr-Esfahany, M., Madani, K., Shah, B.R.: Identifiability and Stability in Causal Structure Learning from Time Series Data. *Entropy* **25**(5), 761 (2023). [10.3390/e25050761](https://doi.org/10.3390/e25050761)
- [25] Chen, D., Du, Y.: Exogenous Isomorphism in Identifiable Causal Reinforcement Learning. In: Proceedings of the 42nd International Conference on Machine Learning (2025)
- [26] Lu, B., Fan, W., Ma, C., Cai, X., Wang, Y., Liu, Y.: OASIS: Offline Actor-Critic with Synthesis. arXiv preprint [arXiv:2410.03679](https://arxiv.org/abs/2410.03679) (2024)
- [27] Li, Z., et al.: BECAUSE: Bilinear Causal Representation for Generalization under Hidden Confounding. arXiv preprint [arXiv:2404.07108](https://arxiv.org/abs/2404.07108) (2024)
- [28] Huang, W., Rafailov, R., Van Hasselt, H., Silver, D.: JOWA: Joint-Policy World Model for Efficient Planning and Acting. ICLR 2025 OpenReview (2025). <https://openreview.net/forum?id=uA0YSMxK5N>
- [29] Anonymous: ReFORM: Reinforcement Learning with Flow-Based Support Constraints. ICLR 2026 OpenReview submission (2026). <https://openreview.net/forum?id=9mQAL4gjwz>

## A Appendix

---

**Algorithm 1** Policy Learning via Counterfactual-Based Data Augmentation — Part 1 ([3])

---

- 1: **Input:** Observed triplets  $(S_t, A_t, S_{t+1})$  from the offline dataset, for  $t = 1, \dots, T$ .
  - 2: **Estimation of a general policy:**
  - 3:     2.1. Estimate the SCM in Eq. 1 using BiCoGAN.
  - 4:     2.2. Generate counterfactual data for alternative actions according to the estimated SCM; denote the counterfactually augmented dataset by  $\tilde{D}$ .
  - 5:     2.3. Perform D3QN learning on  $\tilde{D}$  to obtain the learned policy  $\pi$ .
- 

*Note:* This pseudocode corresponds to **Part 1** of the CTRL framework in [3], adapted to our implementation pipeline.

---

**Algorithm 2** BiCoGAN Training Pipeline (Our Implementation)

---

**Require:** Dataset  $\mathcal{D} = \{(s_t, a_t, \tilde{a}_t, r_t, s_{t+1})\}$ , hyperparameters  $(\alpha, \rho, \phi, \lambda_{\text{fwd}})$ .

**Ensure:** Trained  $(G, E, D)$  models.

- 1: **Stage 1: Pretrain Generator  $G$**
- 2: **for** epoch = 1 to  $N_{\text{pre}}$  **do**
- 3:     **for** minibatch  $(s_t, a_t, \tilde{a}_t, s_{t+1})$  from  $\mathcal{D}$  **do**
- 4:         Predict  $\hat{s}_{t+1} = G(s_t, a_t, u = 0)$ .
- 5:         Update  $G$  via  $\text{MSE}(\hat{s}_{t+1}, s_{t+1})$ .
- 6:     **end for**
- 7: **end for**
- 8: **Stage 2: Adversarial BiCoGAN Training**
- 9: **for** epoch = 1 to  $N_{\text{GAN}}$  **do**
- 10:     **for** minibatch  $(s_t, a_t, \tilde{a}_t, s_{t+1})$  **do**
- 11:         Sample latent  $u \sim \mathcal{N}(0, I)$ .
- 12:         Generate fake  $\hat{s}_{t+1} = G(s_t, a_t, u)$ .
- 13:         **Discriminator update:**

$$\mathcal{L}_D = \text{BCE}(D(s_t, a_t, s_{t+1}), 1) + \text{BCE}(D(s_t, a_t, \hat{s}_{t+1}), 0)$$

- 14:         Gradient step on  $D$ .
- 15:         **Encoder-Generator update:**
- 16:          $(\hat{s}_t, \hat{a}_t, \hat{u}_t) = E(s_{t+1})$ .
- 17:          $\hat{s}'_{t+1} = G(s_t, a_t, \hat{u}_t)$ .
- 18:         Compute

$$\mathcal{L}_{GE} = \mathcal{L}_{\text{adv}} + \gamma_t \mathcal{L}_{\text{EFL}} + \lambda_{\text{fwd}} \mathcal{L}_{\text{fwd}}.$$

- 19:         Gradient step jointly on  $G$  and  $E$ .
  - 20:     **end for**
  - 21: **end for**
- 

---

**Algorithm 3** Offline D3QN + CQL Training (Real or Real+CF Data)

---

**Require:** Normalized dataset  $(S, A, R, S', D)$ , hyperparameters  $\gamma, \alpha_{\text{CQL}}, \tau$ .

**Ensure:** Trained Q-network.

- 1: Initialize Q-network  $Q_\theta$  and target network  $Q_{\theta^-}$ .
- 2: **for** epoch = 1 to  $N_{\text{epochs}}$  **do**
- 3:     Shuffle dataset.
- 4:     **for** each minibatch  $(s, a, r, s', d)$  **do**
- 5:         Compute  $Q_\theta(s, a)$  and  $Q_\theta(s, \cdot)$ .
- 6:         Compute Double DQN target:

$$y = r + \gamma Q_{\theta^-} \left( s', \arg \max_{a'} Q_\theta(s', a') \right) (1 - d).$$

- 7:     TD loss:

$$\mathcal{L}_{\text{TD}} = \text{Huber}(Q_\theta(s, a), y).$$

- 8:     CQL penalty:

$$\mathcal{L}_{\text{CQL}} = \alpha_{\text{CQL}} \left( \log \sum_{a'} e^{Q_\theta(s, a')} - Q_\theta(s, a) \right).$$

- 9:     Total loss:

$$\mathcal{L} = \mathcal{L}_{\text{TD}} + \mathcal{L}_{\text{CQL}}.$$

- 10:     Gradient update on  $Q_\theta$ .

- 11:     Soft target update:

$$\theta^- \leftarrow \tau \theta + (1 - \tau) \theta^-.$$

- 12:     **end for**

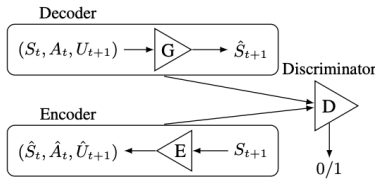
- 13:     **if** epoch mod eval\_every = 0 **then**

- 14:         Evaluate policy for 50 rollouts in the SD/CTRL evaluation environment.

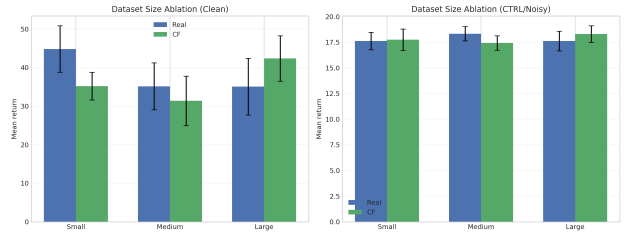
- 15:     **end if**

- 16:     **end for**
-

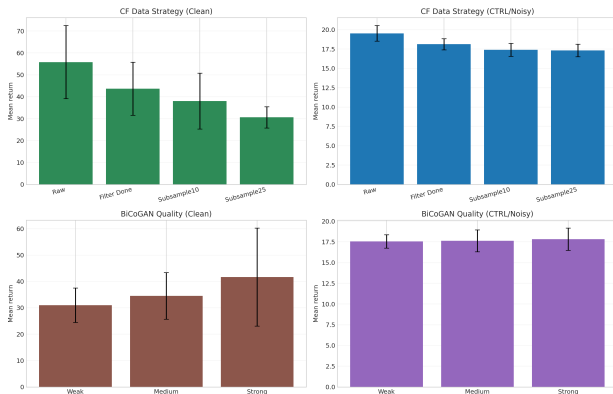
# Evaluating Counterfactual Data Augmentation in Reinforcement Learning



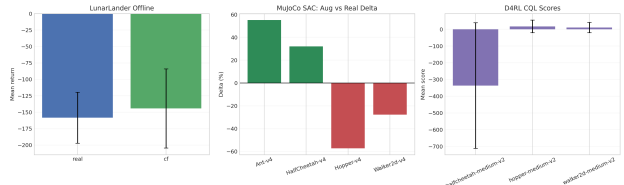
(a) Generator G, Encoder E, and Discriminator D in CTRL [3].



(b) Dataset-size ablation: CF vs real-only under clean and CTRL/noisy evaluation. CF benefit appears in the large-data regime only.



(c) CF strategy and BiCoGAN-quality ablations. Clean performance depends strongly on augmentation/generator quality; noisy returns remain closely clustered.



(d) External-validity summary across LunarLander, MuJoCo, and D4RL.

Table 3: Cross-domain summary with task-level outcomes and run budgets.

Domain	Task/Setting	Budget	Outcome
LunarLander offline	real vs CF	250 dataset eps, 400 DQN epochs, $n = 8$	-158.25 vs -144.31
MuJoCo SAC	Ant-v4 (real vs aug)	200k steps, 20 eval eps, $n = 2$	857.32 $\rightarrow$ 1330.07
MuJoCo SAC	HalfCheetah-v4 (real vs aug)	200k steps, 20 eval eps, $n = 3$	3974.77 $\rightarrow$ 5246.36
MuJoCo SAC	Hopper-v4 (real vs aug)	200k steps, 20 eval eps, $n = 3$	1785.47 $\rightarrow$ 763.75
MuJoCo SAC	Walker2d-v4 (real vs aug)	200k steps, 20 eval eps, $n = 2$	2692.83 $\rightarrow$ 1945.52
D4RL CQL (baseline)	halfcheetah-medium-v2	100k steps, 10 eval eps, $n = 2$	-336.17
D4RL CQL (baseline)	hopper-medium-v2	100k steps, 10 eval eps, $n = 2$	17.80
D4RL CQL (baseline)	walker2d-medium-v2	100k steps, 10 eval eps, $n = 3$	10.86