Realistic CDSS Drug Dosing with End-to-end Recurrent Q-learning for Dual Vasopressor Control

Will Y. Zou[2] **Jean Feng**[1] **Alexandre Kalimouttou**[1]
Jennifer Yuntong Zhang[2,3] **Christopher W. Seymour**[4] **Romain Pirracchio**[1]
[1]University of California, San Francisco [2]Angle.ac
[3]Engineering Science, University of Toronto [4]University of Pittsburgh

Abstract

Reinforcement learning (RL) applications in Clinical Decision Support Systems (CDSS) frequently encounter skepticism because models may recommend inoperable dosing decisions. We propose an end-to-end offline RL framework for dual vasopressor administration in Intensive Care Units (ICUs) that directly addresses this challenge through principled action space design. Our method integrates discrete, continuous, and directional dosing strategies with conservative Q-learning and incorporates a novel recurrent modeling using a replay buffer to capture temporal dependencies in ICU time-series data. Our comparative analysis of norepinephrine dosing strategies across different action space formulations reveals that the designed action spaces improve interpretability and facilitate clinical adoption while preserving efficacy. Empirical results on *eICU* and *MIMIC* demonstrate that action space design profoundly influences learned behavioral policies. Compared with baselines, the proposed methods achieve more than $3\times$ expected reward improvements, while aligning with established clinical protocols [1].

1 Introduction

Clinical interventions in ICUs are inherently heterogeneous, spanning discrete choices and continuous adjustments [1, 2, 3, 4]. This diversity places strong demands on action space in Reinforcement Learning (RL): they should capture the full spectrum of clinical practice while remaining tractable for learning. Without careful action space design, RL policies risk generating unrealistic or impractical recommendations that clinicians are unlikely to adopt. Specifically, in Clinical Decision Support System (CDSS) [5, 6] for septic shock, clinicians apply norepinephrine (first-line vasopressor) and vasopressin (second-line vasopressor) over multiple phases—initiation, titration, and weaning. While reinforcement learning offers a natural framework for this sequential decision-making problem, prior work [7, 8] often focuses on single vasopressor control and early treatment phases, with fewer examining how titration evolves throughout the entire treatment period.

In this work, we study the effectiveness of action space design in influencing the learned dosing policies. Concretely, we adopt the offline settings of the Deep Q-learning (DQN) [9] with recurrent replay [10] to capture temporal dependencies, and incorporate conservative Q-learning (CQL) [11] to ensure policy safety. Integrating them together as a potential CDSS system, we validate the action space design for dosing in treatment trajectories. Using this integrated offline RL framework as a CDSS, we evaluate discrete, continuous, and directional dosing action spaces and analyze their impact on *both norepinephrine and vasopressin* control over the *full treatment horizon*.

Our results show that discrete and directional-discrete action spaces are not only more interpretable, aligned with clinician expectations, but they also significantly improve model performance compared to continuous dosing. Their sparsity and higher dimensionality, combined with the recurrent replay

[1]Code is publicly available at https://github.com/wzoustanford/vaso_rl/

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: NeurIPS 2025 Workshop on Time Series for Health.

framework enable more generalizable models that achieve more than $3\times$ expected reward improvements with weighted importance sampling offline-policy evaluation. These results show the potential of deep reinforcement learning and action space design for building effective CDSS systems.

2 Prior Work

Reinforcement learning techniques have been applied in several independent studies for treating sepsis shock. For instance, [12] proposed individualized treatment for sepsis using RL, and [8] provided an earlier comprehensive study of sepsis treatment with publicly available datasets. More recently, [1] studied ICU data and RL in medicine. [7] provided a focused study of effective dosing action space design in an end-to-end RL system for ICU sepsis treatment.

For action space design in RL, earlier work addressed factored [13], hierarchical [14], continuous action spaces [15, 16]. [10] addressed the importance of recurrent sequence learning in replays for application in Atari games, and [17] progressed recurrent learning into model-based RL applications. Offline RL methods such as conservative Q-learning [11], implicit Q-learning [18] and advantage weighted Actor-Critic [19] applied different regularization and policy constraints. Model-based offline RL approaches [20] focused on modeling transition dynamics. In the clinical literature, [2] applied RL algorithm to sedation dosing in ICUs. [4] provided acollection of dosing techniques focusing on referencing cancer and chemotherapy treatments. [3] applied a latent constrained batch algorithm to address the risks of incorrect dosing heparin in ICU. However, the use of RL to systematically design and evaluate vasopressor dosing strategies for sepsis shock remains under-explored.

3 Algorithms

We propose an offline Q-learning framework that employs dosing action space design with a conservative Q-learning objective, and incorporates a recurrent replay buffer to model ICU time-series.

3.1 Dosing Action Space Design

We introduce an action space design that bridges the gap between reinforcement learning optimization and clinical feasibility. The action space contains a binary decision for vasopressin (vp_1) and dosing decisions for norepinephrine (vp_2) which takes values in the range $(0, 0.5]$ (mcg/kg/min)[2]. Specifically, three action space designs for vp_2 are proposed: continuous dosage, clinically-aligned discrete intervals based on standard protocols, and directional dose adjustments that mirror clinical decision-making. Our action space $\mathcal{A}$=$\{(vp_1,vp_2)\}$ models dual vasopressor administration, maintaining model performance while preserving the tractability of real-world intensive care interventions. This design enables deployment within existing clinical workflows while maintaining the expressiveness required for effective policy learning.

Continuous dosing. A continuous dosing variable is applied for vasopressor norepinephrine allowing it to freely move in a range. The action space is $\mathcal{A} = \{(vp_1, vp_2)\}$, where $vp_1 \in \{0, 1\}$ is a binary variable, and $vp_2 \in (0, 0.5]$ (mcg/kg/min).

Block discrete dosing. We define the dosing action to be the discrete values in the same range as in the continuous dosing. The discrete values only vary across meaningful discrete dosing blocks. We explore different numbers of blocks to validate the impact of effective dosing. The action space is $\mathcal{A} = \{(vp_1, vp_2)\}$, where $vp_2 \in \text{range}(\delta/2, 0.5, \delta)$ (mcg/kg/min) and δ is bin size computed from the number of bins.

Stepwise directional dosing. We define the the action space at each time step to be a directional and stepwise change. The action space is $\mathcal{A} = \{(vp_1, vp_2)\}$, where vp_2 is a stepwise change from the current dose. For example, $vp_2 \in \{-0.1, -0.05, 0, +0.05, +0.1\}$ (mcg/kg/min). The state space is extended to include a one-hot vector that stores and keeps track of the current discrete dosage of norepinephrine.

3.2 Q-Learning for Vasopressor Control

We formulate the vasopressor control problem with $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space of vasopressor decisions, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition dynamics, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward, and $\gamma \in [0, 1]$ is the discount factor. The state $s_t \in \mathcal{S} \subseteq \mathbb{R}^d$ comprises

[2]Due to international clinical practice, we ensure the dosing is above zero for the first-line vasopressor.

patient information, physiological measurements [3] , and organ support [4]. The action space $\mathcal{A}$ contains a binary decision for vasopressin (vp_1) and dosing decisions for norepinephrine (vp_2). The dosing decisions are formulated in Section 3.1. The reward function contains two objectives: maximizing the chance of survival and keeping vasopressors in dosage limit. Given a dataset $\mathcal{D} = \{(s_t, a_t, r_t, s_{t+1})\}$ of historical ICU trajectories collected under behavioral policy π_b (clinical practice), our goal is to learn an optimal policy π^* that maximizes expected cumulative reward.

Q-Learning. Q-learning seeks to learn the action-value function $Q(s, a)$ representing the expected cumulative reward by taking action a in state s and following the optimal policy thereafter. The Q-function satisfies the Bellman optimality: $Q^*(s, a) = \mathcal{R}(s, a) + \gamma \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)}[\max_{a'} Q^*(s', a')]$. In the offline setting, Q-learning minimizes the temporal difference (TD) mean-squared error:

$$\mathcal{L}_{\text{TD}}(\theta) = \mathbb{E}_{(s,a,r,s') \sim \mathcal{D}} \left[\Big(Q_\theta(s, a) - (r + \gamma \max_{a'} Q_{\theta^-}(s', a')) \Big)^2 \right], \tag{1}$$

where Q_θ is the learned Q-network with parameters θ, and Q_{θ^-} is a target network updated periodically for stability. We apply a version of double Q-learning [21] with two Q networks and two corresponding target networks.

As an improvement to Q-learning, we optionally add a conservative term [11] (conservative Q-Learning) to penalize Q-values for out-of-distribution actions, with α as strength of the penalty:

$$\mathcal{L}_{\text{CQL}}(\theta) = \alpha \cdot \mathbb{E}_{s \sim \mathcal{D}} \left[\log \sum_a \exp(Q_\theta(s, a)) - \mathbb{E}_{a \sim \mathcal{D}(a|s)}[Q_\theta(s, a)] \right] + \mathcal{L}_{\text{TD}}(\theta).$$

3.3 Recurrent Experience Replay with Effective Dosing

Our approach applies the recurrent experience replay algorithm [10] to address the temporal dependencies in ICU vasopressor control. The algorithm employs a prioritized sequential buffer $\mathcal{RB} = \{\mathcal{S}_e\}$, storing treatment episodes $\mathcal{S}_e = \{(s_t, a_t, r_t, s_{t+1})_{t \in e}\}$, to capture time-sensitive patterns of patient responses to vasopressor adjustments. In contrast to the sample-wise Q-function application in traditional Q-learning, we leverage an LSTM network to model the sequential nature of ICU states and interventions, recognizing that vasopressor effects unfold over time. In addition, we incorporate the *sum-tree* data structure to prioritize episodes with larger temporal difference (TD) errors [22], improving learning in challenging clinical scenarios. The recurrent model combines with specialized action space design to improve reinforcement learning for patient stability.

4 Experiments

We conduct multiple experiments demonstrating that our action space designs are both learnable and effective. We first validate learnability and model behavior through pilot evaluations with Δ Q-value and Fitted Q-Evaluation, using a simplified reward (Section 4.1). We then show concrete gains through full off-policy evaluation under a comprehensive clinical reward (Section 4.2).

4.1 Learnability of Clinically Aligned Action Spaces

We evaluate our dosing action space design with the reward function defined in Table 1. The objective is to evaluate whether each action space formulation supports meaningful value learning by tracking Q-value improvements and estimating policy quality using Fitted Q-Evaluation (FQE) on sepsis treatment data from eICU [23] and MIMIC-IV [24] datasets [5].

Type	Rule of the reward function
Mortality:	$+10$ if the patient survived, and -10 otherwise
Vaso. penalty:	At each time step, aggregate to reward: $-0.1 \cdot (vp_1 + vp_2 \cdot 2 - 1.0))$

Table 1: Definition of the simple reward function.

Action Spaces. Following Section 3.1, we compare five formulation of action space: **(i)** the Binary vp_1 model, which treats vasopressin as a binary variable; **(ii)** the Dual Mixed model, which presents vp_2

[3] The measurements include mean blood pressure (MBP), lactate levels, blood urea nitrogen, serum creatinine, (total) fluid, urine of the hour, corticosteroid, and Sequential Organ Failure Assessment (SOFA) score.

[4] The support includes mechanical ventilation and renal replacement therapy.

[5] See Appendix A for more data and processing details.

as a continuous variable in the range $(0, 0.5]$ (mcg/kg/min); **(iii)** the Dual Block Discrete (Dual BD) model, which discretizes vp_2 into 3, 5, or 10 bins; **(iv)** the Dual Stepwise model, which encodes directional dose adjustments $vp_{2s} \in [-\text{max\_step}, ..., -0.05, 0, +0.05, ..., +\text{max\_step}]$ (mcg/kg/min); **(v)** the LSTM Block Discrete (LSTM BD), which discretizes vp_2 into 10 bins trained with sequence length 5 and a 2-step burn-in.

RL Model Implementation. The RL model leverages a fully-connected network as a Q-function which takes the combined state-action as input. For double-Q learning, during training and inference, Q-value is taken as the minimum of the target outputs, $q = \min(q_1, q_2)$. The Q networks are periodically updated by adding weighted parameters of the target networks. The model is trained with the TD error objective (Equation 1) [6].

Δ Q Evaluation. As shown in Table 2, action space designs are evaluated with Q improvement per transition (ΔQ /step), where ΔQ is the gain of model-recommended actions over clinician actions. We also measure concordance (C.) between model recommendations and clinician actions. For vp_1, concordance is binary agreement; for vp_2, concordance holds when the recommended dosage falls within the defined bin edges.

Table 2: **Comparison of RL models.** We include Binary vp_1, Dual Mixed, Dual BD, Dual Stepwise, and LSTM BD models with the best conservatism levels (α) selected with the validation set.

Model	Config	α	vp_1 (%)	Q/step	**ΔQ/step**	vp_1 C.(%)	vp_2 C.(%)
Clinician	–	–	38.8	–	0.000	100.0	–
Binary vp_1	–	0.001	79.8	0.869	0.028	44.5	–
Dual Mixed	–	0.010	77.8	1.345	0.185	52.5	–
Dual BD.	3 bins	0.010	76.8	1.576	0.127	53.1	56.3
Dual BD.	5 bins	0.000	96.3	2.383	0.265	39.7	26.4
Dual BD	10 bins	0.000	92.3	4.673	**0.309**	42.4	13.3
Dual Stepwise	max_step 0.1	0.0001	12.4	0.094	0.136	65.1	17.7
Dual Stepwise	max_step 0.2	0.0000	97.3	4.021	**0.511**	38.1	11.7
LSTM BD	10 bins	0.0000	100.0	0.878	**0.456**	–	24.5
LSTM BD	10 bins	0.0001	100.0	0.866	0.398	–	20.4

Fitted Q-Evaluation [7]. FQE is applied as an off-policy evaluation (OPE) [25, 26] method for all models. For each model, we compute test-set Q-values and fit a Gaussian to summarize their distribution. We compare model optimal policy with the clinician actions. As shown in Table 3, block discrete models perform as well as continuous variable for vp_2. Stepwise directional with a larger max_step has the largest 25.1% probability of improved rewards. Moreover, incorporating an LSTM with recurrent experience reply yields a 17.4% probability of improved rewards, which shows the benefits of capturing cross-time dependencies.

Table 3: **FQE OPE Results.** PIR (Probability of Improved Rewards) is defined as the probability of model recommended action reaching a higher reward value than the clinician mean reward, minus 50% (P(Model > Cli. Mean) - 50%); Δ Q Mean is the shift in the mean of fitted Gaussian for Model vs Clinician.

Model	α	Config	PIR	Δ Q Mean	Mod. Mean	Cli. Mean	Cohen's d
Binary vp_1	1e-3	–	0.8%	0.028	0.869	0.841	0.019
Dual Mixed	1e-2	–	6.3%	0.186	1.346	1.161	0.146
Dual BD	1e-2	3 bins	3.0%	0.127	1.576	1.450	0.073
Dual BD	0.0	5 bins	4.4%	0.265	2.383	2.119	0.106
Dual BD	0.0	10 bins	6.4%	0.309	4.673	4.364	0.151
Dual Stepwise	0.0	max_step 0.2	**25.1%**	**0.511**	4.021	3.510	**0.503**
LSTM BD	1e-4	10 bins	**17.4%**	0.398	0.866	0.468	**0.439**

[6] See Appendix F and G for model implementation and architecture. See Appendix H for hyper-parameters.
[7] See Appendix B and C for detailed FQE descriptions and figures .

4.2 Performance with Full Clinical Reward and Weighted Importance Sampling (WIS)

To show that the action space design concretely improves model performance, we implement the full reward function in [7], and leverage WIS for Offline-Policy Evaluation (OPE).

Comprehensive Reward Function. Following [7], we implement a comprehensive reward function with adjusted rewards that indicate patient progress. The reward is composed of a base survival reward, a set of benefit for improved clinical parameters (such as improved mean blood pressure or lactate levels), reward for decreased norepinephrine usage, and mortality penalty [8].

Weighted Importance Sampling. OPE is performed on the test-set trajectories using. First, we identify optimal actions recommended by our model and train a softmax model on both the target policy (optimal model actions, π_m) and baseline policy (clinician actions, π_c). Then, the trajectory-level WIS coefficients are computed as specified in the first line of Equation 2 to estimate the trajectory-level expected reward $R^{(i)}_{\text{traj}}$. Lastly, the overall WIS expected reward is computed using the renormalization shown on the second line of Equation 2.

$$
\begin{aligned}
R^{(i)}_{\text{traj}} &= T \cdot \frac{\sum_{t=0}^{T} w_t R_t}{\sum_{t=0}^{T} w_t}, \quad \text{where } w_t = \frac{\pi_m(a_t^{\text{cli.}} \mid s_t)}{\pi_c(a_t^{\text{cli.}} \mid s_t)}, \\
R_{\text{WIS}} &= \frac{\sum_{i=0}^{N} \tilde{w}_i R^{(i)}_{\text{traj}}}{\sum_{i=0}^{N} \tilde{w}_i}, \qquad \text{where } \tilde{w}_i = \frac{1}{T} \sum_{t=0}^{T} w_t.
\end{aligned}
\tag{2}
$$

We also calculate the trajectory-level WIS by computing the trajectory-level weight as the product of transition level weights across the trajectory time-steps. This alternative evaluation resulted in similar ranges of WIS improvements as in [7]. For our clinically-aligned discrete action spaces, we found the two-step WIS method described in Equation 2 to be reliable and stable.

Model implementation, architecture, and hyperparameters are the same as Section 4.1 except that we remove conservative Q-learning ($\alpha = 0.0$), and train all models for 500 epochs.

Results. As shown in Table 4, finer discretization in the action space yields better performance, and adding recurrent LSTMs with experience replay results in further substantial improvements. Both the Dual BD and LSTM BD models show over $3\times$ improved WIS expected rewards, compared with the the Dual Mixed model where vp_2 is a continuous variable.

Table 4: **Importance Sampling (IS) and Weighted Importance Sampling (WIS) OPE Results.** We show transition-level (Trans. Lvl) average reward $E(R)$, IS, WIS, and corrsponding trajectory-level (Traj. Lvl) average reward. We also present the improvements in trajectory-level WIS (Imp. WIS) between model recommended actions and clinician actions. Lastly, we show the 95% Confidence Interval (CI) for WIS improvement obtained by bootstrapping across test-set trajectories.

Model	Config	Trans. Lvl			Traj. Lvl			Imp. WIS	Imp.$\times$	95% CI
		$E[R]$	IS	WIS	$E[R]$	IS	WIS	(Mod.-Cli.)		
Binary vp_1	–	2.13	2.10	2.15	112.01	110.51	117.54	5.53	-	[2.0, 9.1]
Dual Mixed	–	2.13	1.71	2.23	112.01	89.69	118.85	6.84	1$\times$	[-0.8, 14.2]
Dual BD	3 bins	2.13	1.92	2.22	112.01	101.13	120.98	8.98	1.3$\times$	[2.6, 15.3]
Dual BD	5 bins	2.13	2.23	2.38	112.01	117.25	139.80	27.79	4.1$\times$	[19.8, 35.8]
Dual BD	10 bins	2.13	2.34	2.40	112.01	123.48	141.51	29.50	4.3$\times$	[20.1, 38.3]
LSTM BD	10 bins	2.13	**2.51**	**2.73**	112.01	**131.83**	**155.41**	**43.40**	**6.3**$\times$	[33.6, 54.1]

5 Conclusion

We establish action space design as a critical bridge between reinforcement learning optimization and clinical assistance in ICU settings. By developing a novel action space, we convert theoretical optimal policies into actionable clinical protocols in our proposed dual vasopressor control system for septic shock. Combined with recurrent experience replay to capture the complex temporal dependencies, our action space design achieves more than $3\times$ expected reward improvements with WIS offline-policy evaluation compared with baselines. These contributions move towards a blueprint for CDSS systems that clinicians could interpret and trust, and towards AI systems that not only compute optimal actions, but deliver them in forms that integrate into critical medical workflows.

[8] See Appendix D for more detailed reward description.

References

[1] Pushkala Jayaraman, Jacob Desman, Moein Sabounchi, Girish N Nadkarni, and Ankit Sakhuja. A primer on reinforcement learning in medicine for clinicians. *NPJ Digital Medicine*, 7(1):337, 2024.

[2] Niloufar Eghbali, Tuka Alhanai, and Mohammad M Ghassemi. Patient-specific sedation management via deep reinforcement learning. *Frontiers in Digital Health*, 3:608893, 2021.

[3] Xihe Qiu, Xiaoyu Tan, Qiong Li, Shaotao Chen, Yajun Ru, and Yaochu Jin. A latent batch-constrained deep reinforcement learning approach for precision dosing clinical decision support. *Knowledge-based systems*, 237:107689, 2022.

[4] Elena Maria Tosca, Alessandro De Carlo, Davide Ronchi, and Paolo Magni. Model-informed reinforcement learning for enabling precision dosing via adaptive dosing. *Clinical Pharmacology & Therapeutics*, 116(3):619–636, 2024.

[5] Dereck L Hunt, R Brian Haynes, Steven E Hanna, and Kristina Smith. Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review. *Jama*, 280(15):1339–1346, 1998.

[6] Amit X Garg, Neill KJ Adhikari, Heather McDonald, M Patricia Rosas-Arellano, Philip J Devereaux, Joseph Beyene, Justina Sam, and R Brian Haynes. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. *Jama*, 293(10):1223–1238, 2005.

[7] Alexandre Kalimouttou, Jason N Kennedy, Jean Feng, Harvineet Singh, Suchi Saria, Derek C Angus, Christopher W Seymour, and Romain Pirracchio. Optimal vasopressin initiation in septic shock: the oviss reinforcement learning study. *Jama*, 333(19):1688–1698, 2025.

[8] Matthieu Komorowski, Leo A Celi, Omar Badawi, Anthony C Gordon, and A Aldo Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature medicine*, 24(11):1716–1720, 2018.

[9] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[10] Steven Kapturowski, Georg Ostrovski, John Quan, Remi Munos, and Will Dabney. Recurrent experience replay in distributed reinforcement learning. In *International conference on learning representations*, 2018.

[11] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in neural information processing systems*, 33:1179–1191, 2020.

[12] Suchi Saria. Individualized sepsis treatment using reinforcement learning. *Nature medicine*, 24(11):1641–1642, 2018.

[13] Arash Tavakoli, Fabio Pardo, and Petar Kormushev. Action branching architectures for deep reinforcement learning. In *Proceedings of the aaai conference on artificial intelligence*, volume 32, 2018.

[14] Tejas D Kulkarni, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

[15] Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In *International Conference on Machine Learning*, pages 6215–6224. PMLR, 2019.

[16] Olivier Delalleau, Maxim Peter, Eloi Alonso, and Adrien Logut. Discrete and continuous action representation for practical rl in video games. *arXiv preprint arXiv:1912.11077*, 2019.

[17] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

[18] Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

[19] Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

[20] Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel: Model-based offline reinforcement learning. *Advances in neural information processing systems*, 33:21810–21823, 2020.

[21] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[22] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

[23] Tom J Pollard, Alistair EW Johnson, Jesse D Raffa, Leo A Celi, Roger G Mark, and Omar Badawi. The eicu collaborative research database, a freely available multi-center database for critical care research. *Scientific data*, 5(1):1–13, 2018.

[24] Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

[25] Ruiqi Zhang, Xuezhou Zhang, Chengzhuo Ni, and Mengdi Wang. Off-policy fitted q-evaluation with differentiable function approximators: Z-estimation and inference theory. In *International Conference on Machine Learning*, pages 26713–26749. PMLR, 2022.

[26] Masatoshi Uehara, Chengchun Shi, and Nathan Kallus. A review of off-policy evaluation in reinforcement learning. *arXiv preprint arXiv:2212.06355*, 2022.

[27] Mervyn Singer, Clifford S. Deutschman, Christopher Warren Seymour, Manu Shankar-Hari, Djillali Annane, Michael Bauer, Rinaldo Bellomo, Gordon R. Bernard, Jean-Daniel Chiche, Craig M. Coopersmith, Richard S. Hotchkiss, Mitchell M. Levy, John C. Marshall, Greg S. Martin, Steven M. Opal, Gordon D. Rubenfeld, Tom van der Poll, Jean-Louis Vincent, and Derek C. Angus. The third international consensus definitions for sepsis and septic shock (sepsis-3). *JAMA*, 315(8):801–810, 02 2016.

Appendix

A Data Source and Processing

The patient features extracted from the eICU and MIMIC-IV databases serve as the state representation for the reinforcement learning (RL) algorithm, and the action space consists of vasopressin administration (binary) and norepinephrine dosing in the continuous range $(0, 0.5]$ (mcg/kg/min). The data are preprocessed such that missing values are imputed using forward filling, and norepinephrine dosages are clipped to the valid clinical range $(0, 0.5]$. We include unique patients experiencing their first episode of septic shock, defined using the Sepsis-3 criteria [27] and who were already receiving norepinephrine at the moment shock onset was identified. Shock onset could occur either while the patient was still in the emergency department or after ICU admission.

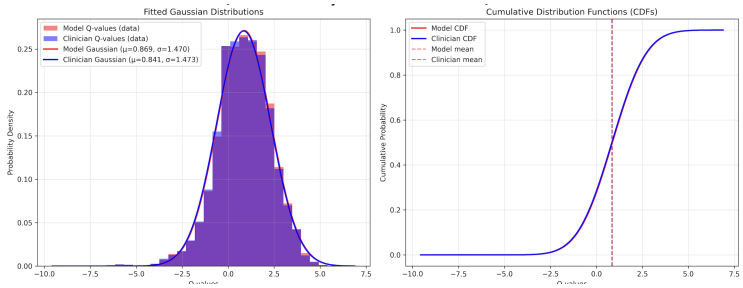
B Fitted Q-Evaluation (FQE)

To evaluate the learned policies against historical clinician decisions without online deployment, we employ Fitted Q-Evaluation (FQE) with Gaussian distribution modeling. This approach computes Q-values for both the learned policy and clinician actions on the held-out test set, enabling direct comparison of expected returns. For each trained model, we extract Q-values by evaluating state-action pairs from the test trajectories where actions are either from the learned policy (model Q-values) or from historical clinician decisions (clinician Q-values). The resulting Q-value distributions are then fitted with Gaussian distributions using maximum likelihood estimation, providing parametric representations characterized by mean (μ) and standard deviation (σ) for both the model and clinician policies. This parametric approach enables computation of key metrics including the mean improvement gap (difference in expected returns), probability of improvement (likelihood that model Q-values exceed clinician Q-values), and effect size (Cohen's d) to quantify the magnitude of policy differences.

The FQE analysis also generates visualizations including overlapping histograms with fitted Gaussian curves, cumulative distribution functions (CDFs) showing the probability mass distribution of Q-values, and improvement probability curves that illustrate the likelihood of the model outperforming various Q-value thresholds. The probability of improvement at the clinician's mean Q-value serves as a critical metric, indicating how often the learned policy is expected to achieve better outcomes than historical treatment decisions. Additionally, the analysis computes Cohen's d as a standardized effect size measure. The larger the value of Cohen's d, the more meaningful are the difference between the means of two groups. This framework provides quantitative evidence for policy improvement while accounting for the inherent uncertainty in Q-value estimates. The FQE analysis was performed for policy performance across different model architectures and hyperparameter configurations. The FQE numerical results are given in Table 3. The histogram and sigmoid CDF visualizations are shown in Appendix C.

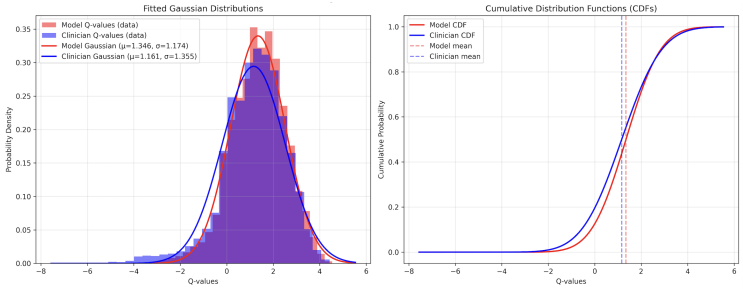
C Fitted Q-Evaluation Illustrations

Figure 1: FQE comparison between learned patient model and clinician baseline (Binary vp_1).



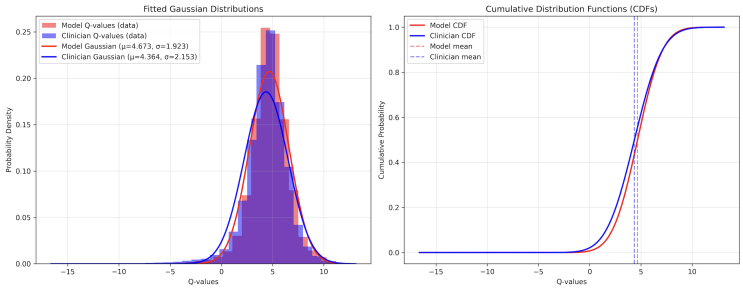
Binary vp_1 Model: distribution of Q-values from model (red) and clinician (blue).

Figure 2: FQE comparison between the learned patient model and clinician baseline (Dual Mixed).



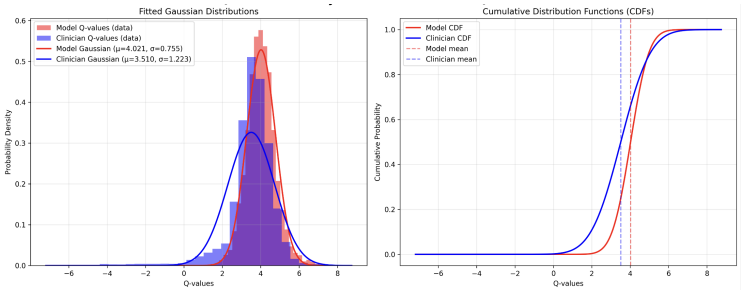
Dual Mixed Model: distribution of Q-values from model (red) and clinician (blue).

Figure 3: FQE comparison between the learned patient model and clinician baseline (Dual BD).



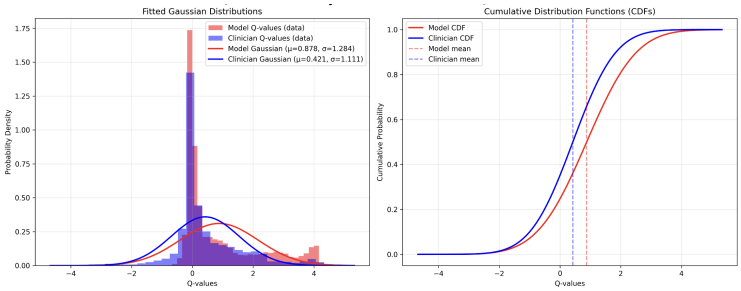
Block Discrete Model (10 bins): distribution of Q-values from model (red) and clinician (blue).

Figure 4: FQE comparison between the learned patient model and clinician baseline (Dual Stepwise).



Dual Stepwise Model (max_step=0.2): distribution of Q-values from model (red) and clinician (blue).

Figure 5: FQE comparison between the learned patient model and clinician baseline (LSTM BD).



LSTM BD (10 bins): distribution of Q-values from model (red) and clinician (blue).

D Comprehensive Reward Function

We directly follow the OVISS [7] study to implement a full and comprehensive reward function with adjusted rewards. The reward measures patient progress and response to vasopressor administration in a quantifiable manner. The reward definition involves calculating an "adjusted reward" for each patient at different time points, which is a composite measure reflecting both survival and physiological benefits.

- **Survival Reward.** Each patient receives a base reward of $+1$ at every time step, reflecting the fundamental importance of maintaining survival.
- **Death Penalty.** If in-hospital mortality occurs, a penalty of -20 is applied at the terminal state of the trajectory. This large negative reward emphasizes the critical nature of mortality when evaluating treatment effectiveness.
- **Clinical Improvement Rewards.** Additional positive rewards are assigned when key physiological markers improve within prespecified future windows:
  - **Lactate Decrease (+1):** A decrease in lactate levels within the next six hours, indicating improved metabolic status.
  - **Mean Blood Pressure Increase (+1):** MBP rising to ≥ 65 mmHg within the next four hours, for cases where the initial MBP is below this threshold, reflecting improved hemodynamic stability.
  - **SOFA Score Decrease (+3):** A reduction in SOFA score within the next six hours, highlighting improved multi-organ function.
  - **Reduced Norepinephrine Usage (+1):** A decrease in norepinephrine dosage within the next four hours, suggesting better cardiovascular stability.

E Detailed Experiment Results

We present detailed comparisons between RL models and baselines.

Table 5: Comprehensive comparison of Q-learning models including Binary, Dual Mixed, and Dual Block Discrete (with 3, 5, 10 bins) variants across different conservatism levels (α).

Model	Config	α	vp_1 (%)	Q/step	**ΔQ/step**	vp_1 C. (%)	vp_2 C. (%)
Clinician	–	–	38.8	0.000	0.000	100.0	–
Binary vp_1	–	0.000	77.3	0.792	0.023	51.6	–
Dual Mixed	–	0.000	87.7	1.367	0.177	41.2	–
Dual BD	3 bins	0.000	71.4	1.734	0.123	55.2	57.1
Dual BD	5 bins	0.000	96.3	2.383	0.265	39.7	26.4
Dual BD	10 bins	0.000	92.3	4.673	**0.309**	42.4	13.3
Binary vp_1	–	0.001	79.8	0.869	0.028	44.5	–
Dual Mixed	–	0.001	69.4	1.302	0.184	56.6	–
Dual BD	3 bins	0.001	82.2	1.614	0.110	46.9	55.5
Dual BD	5 bins	0.001	93.6	2.281	0.217	41.3	27.2
Dual BD	10 bins	0.001	96.6	4.445	0.292	39.1	14.1
Binary vp_1	–	0.010	66.2	0.774	0.024	58.7	–
Dual Mixed	–	0.010	77.8	1.345	0.185	52.5	–
Dual BD	3 bins	0.010	76.8	1.576	0.127	53.1	56.3
Dual BD	5 bins	0.010	84.9	2.091	0.119	43.9	40.3
Dual BD	10 bins	0.010	92.8	4.072	0.245	41.4	16.6

Table 6: Comparison of Binary vp_1 and Dual Stepwise models with vasopressor persistence policy at different maximum step sizes.

Model	α	vp_1 (%)	Q/step	ΔQ/step	vp_1 C. (%)	vp_2 C. (%)
Clinician	–	38.8	0.000	0.000	100.0	–
Binary vp_1	0.001	79.8	0.869	0.028	44.5	–
Dual Mixed	0.010	77.8	1.345	0.185	52.5	–
max_step = 0.1 (mcg/kg/min)						
Dual Stepwise	0.000000	34.6	0.046	0.103	61.1	17.9
Dual Stepwise	0.000100	12.4	0.094	0.136	65.1	17.7
Dual Stepwise	0.001000	82.9	0.252	0.128	47.4	24.5
Dual Stepwise	0.010000	45.9	-0.337	0.114	60.7	22.3
max_step = 0.2 (mcg/kg/min)						
Dual Stepwise	0.000000	97.3	4.021	**0.511**	38.1	11.7
Dual Stepwise	0.000100	31.3	0.134	0.143	70.8	16.4
Dual Stepwise	0.001000	62.0	0.177	0.130	61.2	24.1
Dual Stepwise	0.010000	38.7	-0.184	0.254	64.6	13.5

F Model Implementation and Training

F.1 Double Q-learning

We use double-Q learning for all RL models in this paper. For our simplified version of the double-Q learning for stability, we adopt two Q-networks, and when Q-value is needed, the minimum scalar output across the two network is taken. Similarly, we adopt two target networks and take the minimum across them when computing the temporal difference (TD) error.

F.2 Binary vp1 Model

The Binary vp_1 model represents the simplest action space, treating vasopressin administration as a binary decision (on/off). The vp_2 levels are used as part of the state. Training employs standard Q-learning TD loss with conservative logsumexp regularization over both action choices to prevent

overestimation of out-of-distribution actions. The model processes states through fully connected networks with ReLU activations, using double Q-learning with target networks for stability.

F.3 Dual Mixed Model

The Dual Mixed model handles two action dimensions: vp_1 (binary, 0 or 1) and vp_2 (continuous, 0-0.5 mcg/kg/min). This continuous action space allows for fine-grained dosing control. For action selection of vp2, the model samples 50 candidate actions per state during both training and inference, evaluating Q-values for each to select optimal doses. The conservative Q-learning penalty is computed through importance sampling over randomly sampled actions from the continuous space.

F.4 Dual Block Discrete Model

The Dual Block Discrete model discretizes the continuous vp_2 action space into configurable bins while maintaining vp_1 as binary, creating a factored discrete action space with $2 \times N$ (combined with the binary vp_1 action) total actions where N is the number of vp_2 bins. The discretization uses uniform binning from 0 to 0.5 (mcg/kg/min) with bin centers used for continuous dose reconstruction. Q-values are computed for all discrete action combinations using one-hot encoding, enabling efficient batch evaluation. The conservative Q-learning penalty applies logsumexp over all valid discrete actions to maintain conservative value estimates.

F.5 Dual Stepwise Model

The Dual Stepwise model implements an incremental dosing approach where vp_2 changes are limited to fixed steps (e.g., single step $\pm$0.05, max step$\pm$0.1 mcg/kg/min) from the current dose. This creates a context-dependent action space that respects clinical constraints on dose adjustments. The model augments state representations with one-hot encoded current vp_2 dose bins and maintains valid action masks to prevent out-of-bounds doses. The action space consists of vp_1 (binary) $\times$ vp_2 changes, with conservative Q-learning penalties computed only over valid actions given the current dose context.

F.6 LSTM Block Discrete Model

The LSTM Block Discrete model extends the block discrete approach with sequential modeling capabilities. It processes patient trajectories through LSTM layers to capture temporal dependencies in treatment responses. The model uses sequence lengths of 5 timesteps with 2-step burn-in periods for hidden state initialization. Training employs a specialized replay buffer that maintains overlapping sequences with mortality-weighted sampling priorities. The architecture combines LSTM encoders (2 layers, 32 hidden units) with Q-networks that process concatenated LSTM outputs and state features. Actions are discretized into 10 norepinephrine dosing levels, with conservative Q-learning penalties computed over the discrete action space.

G Model Architectures

The model architectures for general Q-learning networks is shown in Table 7. The model architectures for LSTM Block Discrete networks is shown in Table 8.

H Hyper-parameter Configurations

The shared hyper-parameters are shown in Table 9. The hyper-parameters specific to different action space models are shown in Table 10.

Table 7: Q Network Architecture (Binary/Mixed/Block/Stepwise)

Component	Architecture Details
Input Layer	State (`state_dim`) + Action (`action_dim`)
Hidden Layers	[State, Action] $\xrightarrow{\text{FC (ReLU)}}$ 128 $\xrightarrow{\text{FC (ReLU)}}$ 128 $\xrightarrow{\text{FC (ReLU)}}$ 64
Output Layer	64 $\xrightarrow{\text{FC}}$ 1 (Q-value)
Initialization	Xavier Uniform
Function	$Q(s, a) \to \mathbb{R}$
Networks	Dual Q-networks (Q1, Q2) with dual target networks

Table 8: LSTM Q Network Architecture (LSTM Block Discrete CQL)

Component	Architecture Details
Input	State (`state_dim`)
Feature Extractor	[State] $\xrightarrow{\text{FC (ReLU)}}$ 32 $\xrightarrow{\text{Dropout(0.1)}}$ $\xrightarrow{\text{FC (ReLU)}}$ 32 $\xrightarrow{\text{Dropout(0.1)}}$
LSTM Layers	32 $\xrightarrow{\text{LSTM}}$ `2 layers` $\times$ 32 units (with dropout=0.1)
Output Layer (Q-head)	32 $\xrightarrow{\text{FC}}$ `num_actions` (Q-values)
Initialization	Xavier Uniform
Function	$Q(s, a) \to \mathbb{R}^{\text{num\_actions}}$
Networks	Dual Q-networks with LSTM, dual target networks

Table 9: Common Hyperparameters Across All Models

Hyperparameter	Value	Description
Learning Rate (lr)	10^{-3}	Adam optimizer learning rate
Batch Size	128	Training batch size
Gamma (γ)	0.95	Discount factor for future rewards
Tau (τ)	0.8	Soft target network update rate
Gradient Clipping	1.0	Maximum gradient norm
Epochs	100	Number of training epochs
Validation Batches	10	Batches used for validation
Random Seed	42	Ensures reproducibility

Table 10: Model Configurations: Action Spaces and Hyperparameters

Action Space		Key Parameters	
Model	**Configuration**	**Model**	**Hyperparameters**
Binary	1D binary (0/1)	Binary	—
Dual Mixed	2D vp_1 binary, vp_2 continuous vp_1: {0,1}, vp_2: [0,0.5]	Dual Mixed	`num_samples`: 50 (continuous vp_2 selection)
Block Discrete	2×N discrete vp_1: binary, vp_2: N bins	Block Discrete	—
Stepwise	2×M discrete vp_1: binary, vp_2: M steps	Stepwise	`max_step`: {0.1, 0.2} `step_size`: 0.05
LSTM Block	N discrete (vp_2 only)	LSTM Block	`seq_len`: 5, `burn_in`: 2