Reliable Offline Policy Evaluation for Individualized Mechanical Ventilation

by

Bas Volkers

Thesis Committee:

Chair:	Marcel J.T. Reinders, Faculty EEMCS, TU Delft
Daily Supervisor:	Jesse H. Krijthe, Faculty EEMCS, TU Delft
External supervisor:	Jim M. Smit, Faculty EEMCS, TU Delft
Committee Member:	Frans A. Oliehoek, Faculty EEMCS, TU Delft

TU Delft

Reliable Offline Policy Evaluation for Individualized Mechanical Ventilation

Bas W.S. Volkers, Jim M. Smit, and Jesse H. Krijthe

Delft University of Technology

Abstract

Individualizing mechanical ventilation treatment regimes remains a challenge in the intensive care unit (ICU). Reinforcement Learning (RL) offers the potential to improve patient outcomes and reduce mortality risk, by optimizing ventilation treatment regimes.

We focus on the Offline RL setting, using Offline Policy Evaluation (OPE), specifically importance sampling (IS), to evaluate policies learned from observational data. Using a running example, we illustrate how a large difference between the learned policy and actual clinical behavior (behavior policy) limits the reliability of IS-based OPE. To assess this reliability, we use the Effective Sample Size (ESS) as a diagnostic.

To achieve reliable evaluation, we apply policy shaping, by incorporating a divergence constraint in the policy learning objective, aiming to reduce the difference between the evaluation and behavior policy. We consider both a Kullback-Leibler (KL) divergence constraint and introduce a new constraint, the ESS divergence. Since effective OPE relies on an accurate estimate of the true behavior policy, we address how such an estimate is acquired. Various classifiers for estimating the behavior policy are systematically evaluated, focusing on both discrimination and calibration performance.

Empirical results show the difficulty of learning policies that outperform existing clinical practices and generalize well to unseen patients. Although policy shaping improves the reliability of policy evaluations, no policies that consistently outperform clinician practice were found. The KL divergence constraint generalized better to unseen patients than the ESS divergence, which achieved large ESS without actually reducing the difference between the evaluation and behavior policy.

We underscore the necessity of a cautious approach to applying RL in healthcare, and advocate that assessing OPE reliability and behavior policy calibration becomes standard practice, to ensure that only effective and reliable RL policies are considered for real-world clinical trials.

1 Introduction

Individualized mechanical ventilation for patients in the intensive care unit (ICU) continues to be a challenging task[1]. Mechanical ventilation assists patients suffering from respiratory failure or pulmonary impairment with breathing. It helps stabilize patient conditions and lets other treatments and medications facilitate patient recovery. Current treatment regimes focus on setting the appropriate ventilator settings. Clinical research has led to established evidence-based ventilation strategies[2], where maintaining certain settings below specific thresholds is key. Notably, driving pressures (ΔP) are found to be associated with mortality[3]. However, optimal ventilator settings are often unknown for a specific individual, while suboptimal settings can lead to ventilator-induced lung injury (VILI)[4].

Mechanical ventilation is a continuous decision-making process, where patient state is frequently re-evaluated and ventilator settings are adjusted accordingly. There is a potential for the application of Artifical Intelligence in optimizing mechanical ventilation settings and improve patient outcomes. A particular area of Artificial Intelligence, called Reinforcement Learning (RL), aims to find optimal procedures for sequential decision-making[5].

RL can be categorized in two main approaches: online and offline[6]. Online RL is characterized as an iterative process, where a self-learning agent learns optimal decision-making strategies by actively interacting with its environment. In high-risk environments, such as healthcare, where direct interaction with the environment is unethical and unfeasible, the use of Offline RL is required, which involves optimizing a policy, or dynamic treatment regime, by learning from an observational dataset. The performance of a learned policy is then estimated through offline policy evaluation (OPE). A separate test set is held apart that is used to answer a counterfactual query: "What would patient outcomes have been if the proposed RL policy was applied?". Careful evaluation is necessary to safeguard against learning harmful policies[7].

A central challenge in OPE is the distributional shift: while the RL policy is learned under one distribution, it is evaluated on a different distribution[6]. We focus our attention primarily on importance sampling (IS) evaluation methods, a weighting-based category of OPE. Using a running example, we demon-

strate that when the proposed RL and observed policies disagree too much, the resulting policy value estimate depends on a limited number of patient trajectories, which undermines the reliability[1] of OPE. We highlight the need for using a diagnostic that is related to the variance of the estimated policy value, serving as a tool to assess the reliability of OPE. Since OPE relies on an estimate of the unknown true behavior policy, we investigate how to accurately estimate the behavior policy. Then, we explore how using this estimated behavior policy to shape the policy optimization process allows for policies that can be evaluated more reliably. We test the effectiveness of an existing policy constraint and introduce a new policy constraint, both aimed at improving the reliability of OPE. Finally, we discuss limitations of our approach and make recommendations for applying RL in heatlthcare.

2 Running Example: Predicting Optimal Ventilator Settings

To illustrate challenges associated with OPE, we use a running example based on the problem of finding individualized mechanical ventilation regimes. The goal is to find an optimal policy, the one with the lowest estimated mortality risk, recommending ventilator settings tailored to the individual state of a patient. The scenario is similar to previous work by Peine et al. [8], which based their approach on work by Komorowski et al. [9]. The scenario was later extended to continuous state space by Kondrup et al. [10].

We use a cohort of mechanically ventilated patients from the MIMIC-IV database[11]. The MIMIC-IV database contains 7,281 mechanically ventilated patients that met the inclusion criteria: age at least 18 upon admission, documented 90-day mortality, and mechanically ventilated for at least 24 hours. We use a collection of 33 features, including demographics, vital signs and lab values, to represent the patient state. For each ICU stay, we collect a week of data starting from the point of intubation, aggregated into 8-hour intervals. If less than a week of data is available, then data is collected up until the point of extubation.

Each patient stay i is modelled as a trajectory $(s_0, a_0, r_0, s_1, ..., s_{T_i}, a_{T_i}, r_{T_i})$: a historical sequence of states s, actions a, and rewards r, with varying episode length T_i. At every timestep t the patient is in a particular state s_t, which describes all relevant covariates of the patient. An action a_t is chosen by the attending clinician, after which the patient transitions to a new state s_{t+1}, 8 hours later. The entire observational dataset D consists of N such trajectories: $D = \{(s_0^i, a_0^i, r_0^i, s_1^i, ..., s_{T_i}^i, a_{T_i}^i, r_{T_i}^i)\}_{i=1}^N, \quad N = 7,281$.

The state space $\mathcal{S}$ consists of all possible states a patient can be in. A patient's state can either be represented as a single discrete value[8], or as a vector of feature values[10]. For discrete state representation, a clustering algorithm is used to map similar states to k discrete states.

In this work, we initially consider a discrete state representation in Chapter 3 to illustrate an issue in OPE. Then, in Chapter 4 onwards, we transition to a continuous state representation. This allows us to use more flexible RL algorithms, enabling us to incorporate constraints during policy optimization.

The action space $\mathcal{A}$ consists of a single ventilator setting which is found to be associated with mortality: driving pressure (ΔP). The driving pressure is discretized into five discrete bins, such that a policy has five available actions at each decision point throughout the episode. These action bins are detailed in Appendix C.

A reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is chosen that focuses on maximizing 90-day survival, in order to optimize for the long-term mortality risk. At the end of a patient trajectory, a positive reward (+100) is given if the patient survived, and a negative reward (-100) is given if the patient died. A discount factor γ of 0.99 is chosen, such that early deaths are punished nearly the same as late deaths.

The goal of RL is to find the best dynamic treatment regime; learn the optimal policy $\pi_e(a|s) : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$, that maximizes the expected return $\mathbb{E}_{\pi_e}[R]$. The return of trajectory i is the sum of cumulative discounted rewards $R_i := \sum_{t=0}^{T_i} \gamma^t r_t^i$.

3 Offline Policy Evaluation

OPE involves comparing an evaluation policy (π_e) with a behavior policy (π_b), representing the clinician's decision-making process that generated the observational dataset. In our running example, the evaluation policy, or AI policy, is learned through Reinforcement Learning. OPE enables the estimation of the performance of the evaluation policy without the need to execute the policy in a real-life setting.

In general, the value of a policy is defined as its expected return $\mathbb{E}_{\pi_e}[R]$. If we could execute the evaluation policy online, in either a simulated or real-life setting, then we could estimate the policy value by averaging over all rewards observed while executing the policy. This is known as the Monte Carlo (MC)

[1] Reliability: how much an estimator can be trusted. We regard an estimator that is based on a higher number of samples to be more reliable. OPE that is based on only a few trajectories is considered unreliable.

estimator for the policy value:

$$V_{MC} := \frac{1}{N}\sum_{i=1}^{N} R_i. \quad (1)$$

In this work, we consider importance sampling (IS) methods for OPE, a technique used to estimate the expected value of a distribution based on samples from another[12]. The return of each trajectory is weighted according to its importance sampling ratio: the relative probability of occurrence under the evaluation and behavior policy. The per-step importance ratio $\rho_t^{(i)}$ at time t and cumulative importance ratio $\rho_{1:T}^{(i)}$ of trajectory i are defined as:

$$\rho_t^{(i)} := \frac{\pi_e(a_t^i|s_t^i)}{\pi_b(a_t^i|s_t^i)}, \quad (2) \qquad \rho_{1:T}^{(i)} := \prod_{t=1}^{T} \rho_t^{(i)}. \quad (3)$$

The IS estimator for the policy value is a weighted average of the returns of all trajectories in the evaluation set, where the weight of a trajectory is its cumulative importance ratio:

$$V_{IS} := \frac{1}{N}\sum_{i=1}^{N} \rho_{1:T}^{(i)} R_i. \quad (4)$$

In the running example, we use the weighted importance sampling (WIS) estimator, which has lower variance than the IS estimator, at the expense of introducing bias. The normalized importance sampling weights w_i and the overall WIS estimator are defined in Eq. (5) and (6).

$$w_i := \frac{\rho_{1:T}^{(i)}}{\sum_{j=1}^{N} \rho_{1:T}^{(j)}} \quad (5) \qquad V_{WIS} := \sum_{i=1}^{N} w_i R_i \quad (6)$$

Intuitively, importance sampling can be thought of as creating a pseudo-population of trajectories, wherein each trajectory is replicated by a number of times according to its importance sampling weight. These weights enable the construction of the unbiased IS estimator for the expected policy value under the evaluation policy, based on samples obtained from the behavior policy.

The bias of the WIS estimator arises from the normalization term in the denominator in Eq. (5). The act of normalizing the weights changes the overall expectation of the WIS estimator. In practice, however, the WIS estimator usually has lower variance and is therefore preferred[12].

3.1 Unreliable policy evaluation

We showcase, using the running example, a scenario in which the policy evaluation is dependent on only a few trajectories, making it unreliable. We consider the discrete state space setting, and randomly split the dataset into two parts: 80% training data, and 20% test data. For each random split, we build a model by clustering data with k-means ($k = 650$), learn an RL policy via Q-learning[13], derive a stochastic evaluation policy with a softmax over estimated Q-values, and use the WIS estimator for policy value estimation on the train and test set. We repeat this cycle 40 times to assess the variation across models.

Following Peine et al. [8] and Komorowski et al. [9], we plot the evolution of the highest estimated policy value (Figure 1) as more models are built, alongside a boxplot of the estimated policy values on the test set (Figure 2). While these Figures hint at potential improvement over clinician practice, they give an incomplete view of policy performance, since they don't address the uncertainty of policy evaluation within a single model.

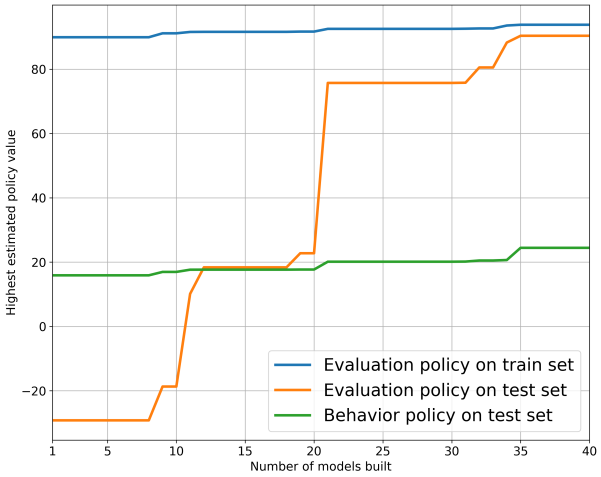


Figure 1: ***Policy evaluation across models.*** *Evolution of the highest estimated policy value on the train and test set versus the behavior policy (clinician practice). For each model, an optimal policy is learned through RL and then evaluated by the WIS policy value estimator.*

After building 20 models, it appears that we find improvement over current clinician practice. We might be inclined to choose the 'best' policy (green line, Figure 2), and conclude that we find improvement over the behavior policy, as Peine et al. [8] and Komorowski et al. [9] do. However, the large variation in policy value estimate ($\sigma = 53.1$) is concerning and requires investigation, as nearly half of the models (47.5%) don't show improvement over the behavior policy (Figure 2). We advise against relying solely on these two Figures, and recommend exploring the cause of variation across models.

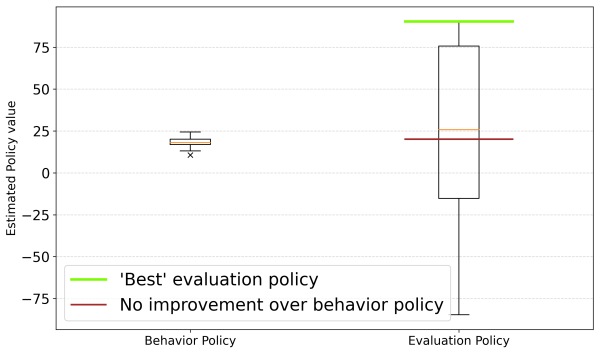


Figure 2: ***Policy evaluation across models.*** *Distribution of the estimated policy value of the behavior policy, and the AI policy on the test set. Estimated policy values for 40 models are shown. Colored lines show two example policies.*

To understand the variation across models, we examine the uncertainty of policy evaluation within a single model. By applying bootstrapping[14] on the test set, we gain insight into the variance of the estimated policy value. For instance, consider a policy that showed no improvement over the behavior policy (brown line, Figure 2). Bootstrapping in this case shows a large variance in the estimated policy value (Figure 3), indicating uncertainty in the policy value estimate.

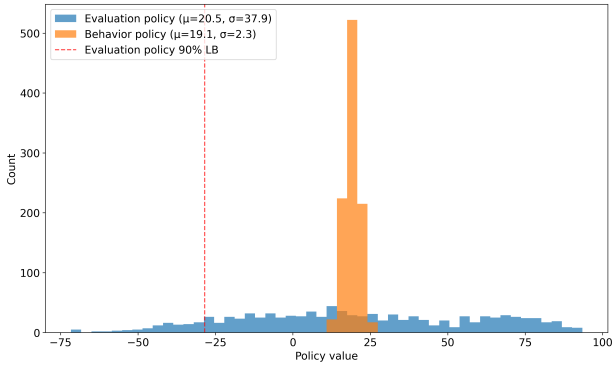


Figure 3: *Histogram of estimated policy value on the test set for 1000 bootstrap iterations. In each bootstrap iteration, a new population of patient trajectories is sampled by bootstrapping from the test set, and the WIS estimator is applied to compute the estimated policy value. The dashed red line shows the 90% lower bound (LB).*

Bootstrapping, however, is not always sufficient. Consider another policy, with a high estimated value (Figure 4a). The distribution of policy value estimates might look good, but the distribution of WIS weights reveals that two trajectories, having WIS weight 0.46 and 0.25, generally account for 71% of the estimated policy value. This indicates that the policy value estimation mostly relies on these two trajectories.

To illustrate the impact of a few trajectories on the estimated policy value, we remove the two influential trajectories (Figure 4b), and observe a shift in the estimated performance of the evaluation policy. This removal, introducing selection bias, causes an increase in the variance and a decrease in the 90% lower bound (LB) of the evaluation policy. Bootstrapping in the test set without these trajectories would have resulted in a much more pessimistic estimate with a higher estimated variance. This scenario highlights that a policy value estimate based on only a few influential trajectories cannot be considered reliable[15].

3.2 Policy mismatch

The limited number of influential trajectories is caused by a large variance in the WIS weights. The distribution of WIS weights (Figure 4a) is left-skewed: only 24 out of 1457 trajectory weights surpass the mean. This variance is explained by the mismatch between the evaluation policy and behavior policy, which agree on the best action for only 17% of the states. This dissimilarity leads to an increase in the variance in WIS weights, which grows exponentially in the episode length - a problem known as the curse of horizon[16]. The WIS estimator suffers from high variance when the evaluation and behavior policy differ significantly.

Remember that OPE tries to answer the question: "What would patient outcomes have been if the proposed RL policy would have been applied?". When the evaluation policy frequently proposes actions that differ from those taken by clinicians, then only a few patients remain that were treated in accordance with the proposed policy. Consequently, only those few patients can be used to make an estimation of the policy's performance. In such cases, we cannot draw reliable conclusions about the evaluation policy.

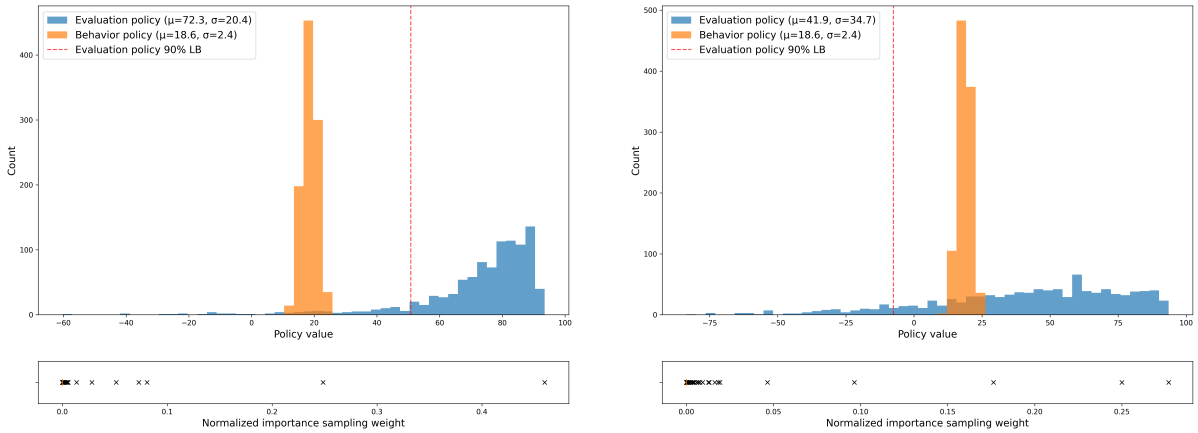
3.3 Effective Sample Size: a diagnostic

W can use a diagnostic known as the effective sample size (ESS) to assess the reliability of the WIS estimator [17, 18]. The ESS measures the relative efficiency between IS estimator and the MC estimator for the policy value:

$$ESS := N \frac{\text{var}_{\pi_e}[V_{MC}]}{\text{var}_{\pi_b}[V_{IS}]}. \tag{7}$$

Intuitively, the ESS indicates how many samples drawn from π_e would provide the same information as our N weighted samples from π_b. Through several assumptions and approximations[18], a practical ESS approximation is proposed by Kong[17]:

$$ESS \approx \frac{1}{\sum_{i=1}^{N} w_i^2}. \tag{8}$$



(a) *Entire test set*

(b) *Two most influential trajectories removed*

Figure 4: *Distribution of estimated policy values for a single model. The top figure shows a histogram of the estimated policy value on the test set for 1000 bootstrap iterations. In every iteration, a new population of patient trajectories is sampled by bootstrapping from the test set, and the WIS estimator is applied to compute the estimated policy value. The dashed red line shows the 90% lower bound (LB). The bottom figure shows a boxplot of the associated WIS weights computed over the entire test set (n=1457)*

Intuitively, an ESS value close to N means that all trajectories are equally weighted to determine the policy value, with individual weights uniformly distributed. Conversely, an ESS close to 1 means a single trajectory dominates the estimate, indicating high variance in WIS weights. For illustrative examples of the ESS, see Appendix A.

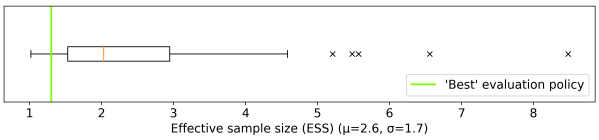


Figure 5: ***Effective sample size across models****. Distribution of the estimated effective sample size for the policy evaluation of on the test set. This figure shows the estimated ESS for the policy evaluation depicted in Figures 1 and 2, and serves as an informative addition regarding the policy evaluation reliability. The green line shows the ESS for the policy that yielded the highest estimated value in Figure.2.*

In our running example, the policy evaluation has a small ESS (Figure 5), indicating that the estimation of policy values relies on a handful of trajectories. The variation across models can be attributed to multiple factors, such as the random train/test split, convergence to local optima, etc. However, we find it most likely that the low ESS is the primary contributor to this variation.

In this case, our OPE becomes unreliable. We cannot draw meaningful conclusions about any of the policies we learned. Selecting the 'best' policy, the green line in Figure 2, without accounting for evaluation uncertainty, risks overestimating its performance. Choosing this policy gives no guarantees regarding prospective performance, and we must consider the evaluation of the policy as unreliable instead.

To improve the reliability of the policy evaluation of learned policies, a larger ESS is preferred. This requires reducing the difference between the evaluation and behavior policy. In Chapter 5, we explore how policy shaping can be applied to explicitly steer the learned policy towards closer alignment with the behavior policy. However, both for the WIS estimator, and to be able to stay close to the behavior policy, we need to accurately estimate what the behavior policy looks like. We will address the estimation of the behavior policy first, before delving into policy shaping.

4 Behavior Policy Estimation

For effective OPE it is crucial to have a good estimate of the behavior policy[7, 19]. The WIS estimator requires an estimate of the behavior policy $\pi_b(a|s)$: the probability that a clinician selects a particular ventilator setting a when the patient is in a specific state s. The true behavior policy is unknown, so it is required to estimate the behavior policy using our dataset. The behavior policy π_b is estimated through supervised learning. A classifier $C : \mathcal{S} \mapsto \mathcal{A}$ is trained to predict what discrete action a is chosen based on the patient state s. After fitting a classifier to the training data, an estimate of the behavior probabilities $\pi_b(a|s)$ can be obtained from the classifier.

In order to assess how 'good' the estimate of a behavior policy is, two different properties of a classifier are of interest: discrimination and calibration. Discrimination refers to how well the classifier is able to distinguish between different classes. A classifier that has good discrimination will correctly assign higher

probabilities to instances of the target class and lower probabilities to instances of the other classes. Calibration refers to how well the predicted probabilities match the observed probabilities in the dataset.

4.1 Methods

We evaluate discrimination in terms of accuracy, and evaluate calibration following the hierarchy as proposed by Van Calster et al. [20, 21]: four increasingly strict levels, referred to as mean, weak, moderate, and strong calibration. Moderate calibration is the level we are most interested in, as it ensures that probabilities are well calibrated across the entire probability range. We use calibration curves as a tool for assessing moderate calibration, by comparing the calibration curve with the ideal calibration curve, which is characterized by an exact correspondence between the predicted probability of a class and the observed proportion of the class. Additionally, we also use other metrics such as proper scoring rules and the calibration error[22] to quantitatively assess moderate calibration, which are defined in Appendix B.

In this chapter, we transition to continuous state representations to allow the classifiers to fully utilize the data's complexity. For the running example, we evaluate the effectiveness of four increasingly flexible classifiers as an estimator of the behavior policy: a Logistic Regression (LR) model[23], XGBoost (XGB) classifier[24], Random Forest (RF) classifier[25], and a Multilayer Perceptron (MLP) classifier[26]. The dataset is split into training, validation, and test set 20 times. Each classifier is trained on the training set, while the validation set is used for hyperparameter selection and early stopping.

The sequential nature of mechanical ventilation, where decisions at time t depend on those at time $t-1$, is reflected in our dataset, where 53% of decision moments show no change in ventilator settings. Recognizing this, we include the previous clinician action as a feature, in contrast to the approaches of Peine et al. [8] and Kondrup et al. [10]. We evaluate the performance impact of including versus excluding the previous clinician action on the performance of the behavior policy estimators, examining the effect of our decision to include this feature.

4.2 Results

Through visual inspection of the calibration curves for the first action, shown in Figure 6, we observe that the RF classifier deviates from the ideal calibration curve the most, clearly performing worst. The MLP, XGB, and LR models show similar performance, but the MLP classifier has a larger variation between the randomsplits of the dataset. Although the LR and XGB classifiers are both close to the ideal calibration curve, their manner of miscalibration is different. The LR models shows overestimation in the range $[0, 0.4]$ and underestimation in the range $[0.4, 1]$. For the XGB model we see a different trend: it shows underestimation in the range $[0.2, 0.7]$ and overestimation in the range $[0.7, 1]$.

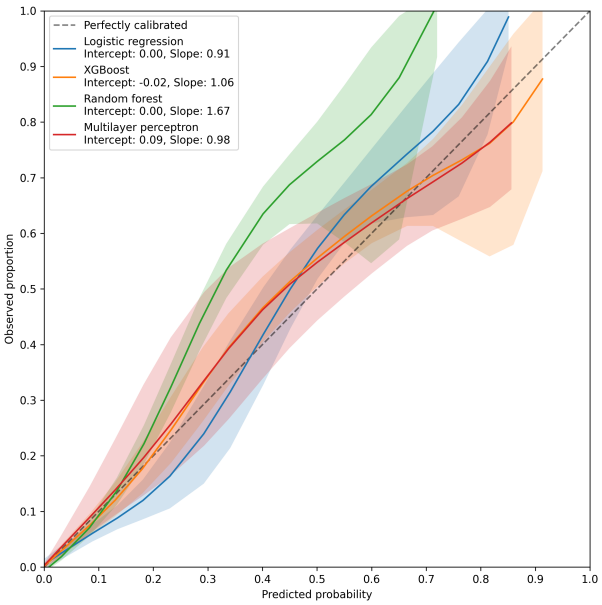


Figure 6: *Calibration curves of different classifiers for behavior policy estimation. Shown curves are for the first action. Shaded areas show the 0.05 and 0.95 quantiles, estimated by randomly splitting the dataset 20 times.*

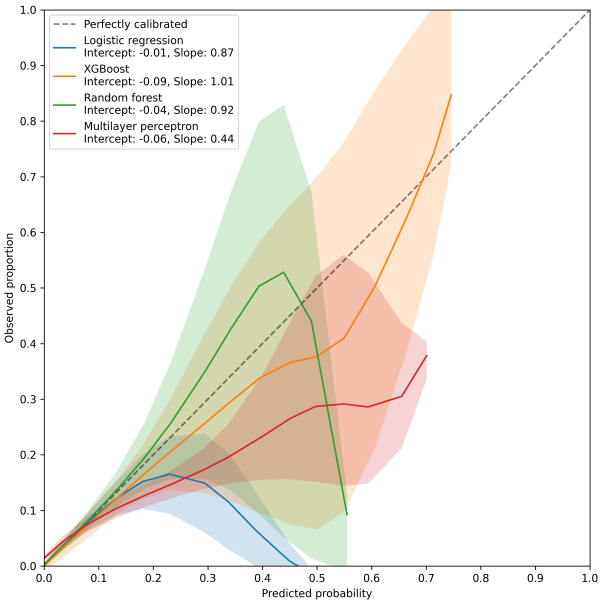


Figure 7: *Calibration curves of different classifiers for behavior policy estimation, when not including previous clinician action in the patient state. Shown curves are for the first action. Shaded areas show the 0.05 and 0.95 quantiles, estimated by randomly splitting the dataset 20 times.*

Table 1: *Performance of different classifiers for behavior policy estimation. Results were measured on the test set under 20 random splits of the dataset, and show 95% CI. ECE and MCE use 10 equal width bins. For their definitions, see Eq.* (26) *and Eq.* (27) *in Appendix B.*

a_{t-1} as feature	Classifier	Accuracy	Log Loss	Brier Score	ECE	MCE
Yes	LR	0.611 ± 0.002	1.070 ± 0.005	0.1131 ± 0.0003	0.039 ± 0.001	**0.19 ± 0.06**
	MLP	0.605 ± 0.003	1.039 ± 0.005	0.1113 ± 0.0005	0.017 ± 0.002	0.26 ± 0.19
	RF	0.592 ± 0.002	1.109 ± 0.003	0.1181 ± 0.0003	0.051 ± 0.001	0.26 ± 0.10
	XGB	**0.612 ± 0.002**	**1.033 ± 0.004**	**0.1103 ± 0.0004**	**0.011 ± 0.001**	0.22 ± 0.05
No	LR	0.303 ± 0.003	1.509 ± 0.003	0.1528 ± 0.0002	0.010 ± 0.001	0.31 ± 0.07
	MLP	0.289 ± 0.004	1.605 ± 0.009	0.1586 ± 0.0006	0.058 ± 0.002	0.32 ± 0.13
	RF	0.299 ± 0.002	1.512 ± 0.003	0.1531 ± 0.0002	0.013 ± 0.001	0.33 ± 0.11
	XGB	0.302 ± 0.003	1.508 ± 0.002	0.1529 ± 0.0002	0.013 ± 0.001	0.36 ± 0.04

Table 1 shows a quantitative evaluation of discrimination and calibration, aggregated for all five classes. For the discriminative measure, accuracy, the difference between the classifiers is small. The moderate calibration metrics, the expected calibration error (ECE) and maximum calibration error (MCE), dependent on a predetermined number of bins, have optimal values for different classifiers, respectively the XGB and LR classifier.

Excluding the previous clinician action from the patient state features results in a significant drop in performance in terms of discrimination and calibration. The calibration curves exhibit a larger spread and deviate from the ideal calibration curve (Figure 7). Furthermore, on all quantitative metrics, except for the ECE, the classifiers score worse (Table 1). In contrast to previous work, we will therefore always include the previous action in the state space.

Choosing the best behavior policy estimator is not trivial. So far, we looked only at the calibration curves of the first action. However, given that the first action is selected in only 9% of the decision instances, we need to visually assess the calibration curves for the other actions as well, which we show in Appendix E. As the degree, as well as the direction of miscalibration (e.g. underestimation vs overestimation) for each classifier varies over the different actions, it is hard to judge which classifier yields the best overall calibration.

Therefore, to select a classifier, we chose a calibration metric that aggregates over all actions. For a particular action, the MCE focuses on the single bin with the largest discrepancy between predicted probabilities and observed proportions. The ECE averages the errors across all bins, mitigating the impact of a single bin. We aim for the lowest error for any single bin, prioritizing the MCE. Based upon this criterion, the LR model emerges as the best behavior policy estimator. In the next chapter, we examine how this estimated behavior policy can shape the policy optimization process in RL, with as objective to obtain policies that are evaluated more reliably.

5 Policy Shaping

To decrease the variance in policy evaluation, one approach is to reduce the difference between the evaluation and behavior policy, which has been actively explored in recent offline RL literature[27]. The goal is to obtain policies that allow for more reliable evaluation, ensuring that a greater proportion of the trajectories in the evaluation set contribute meaningfully to the determination of the policy value. Actively shaping the policy in the policy optimization phase of RL may result in policies that are easier to evaluate.

Policy shaping can be implemented in multiple ways. In this work, we consider the direct policy constraint approach[27]. We use the typical actor-critic framework, in which the actor, responsible for action selection, defines the evaluation policy and is parameterized by a neural network π_θ. The critic, helping to stabilize the actor's reward signal, is parameterized by a neural network Q_ψ (Figure 8). Optimization of the critic involves minimizing Bellman errors, similarly to the process of Q-learning. The policy learning objective for the actor network is defined as:

$$\max_{\pi_\theta} \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim D}[\mathbb{E}_{a'\sim\pi_\theta(\cdot|s_t)}[Q_\psi(s_t,a')]]. \quad (9)$$

The direct policy constraint approach aims to reduce the distance between the evaluation policy and the behavior policy. This is accomplished by introducing a divergence penalty into the policy learning objective. Let $\hat{D}$ denote an estimate of the divergence between π_θ and π_b, and α a hyperparameter that represents the weight of the policy constraint. The constrained policy learning objective[28] is defined

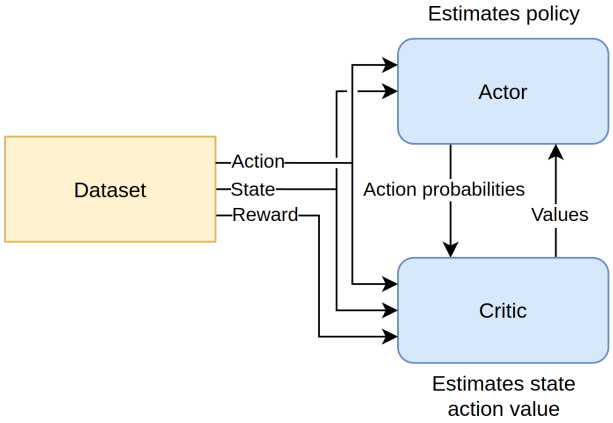


Figure 8: ***The actor-critic framework for offline RL.*** *The critic uses observed rewards to provide value estimates for states. The actor uses these estimates from the critic instead of the raw reward signal to improve stabilization during learning.*

as:

$$\max_{\pi_\theta} \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim D}[\mathbb{E}_{a'\sim\pi_\theta(\cdot|s_t)}[Q_\psi(s_t,a')] - \alpha \cdot \hat{D}(\pi_b(\cdot|s_t),\pi_\theta(\cdot|s_t))]. \quad (10)$$

This objective focuses on achieving high expected return, while staying close the behavior policy. The trade-off between exploitation (i.e. high expected return) and divergence (i.e. the distance to the behavior policy) is influenced by setting the hyperparameter α.

5.1 Methods

In this work, we consider a Kullback-Leibler (KL) divergence constraint and introduce a new constraint, directly linked to the ESS. The KL divergence constraint (Eq. (11)) has been found to be effective at regularizing the evaluation policy[29], and is applied across the entire action space. This constraint forces the evaluation policy to be closer to the behavior policy, even for actions not chosen by clinicians.

$$\hat{D}_{KL}(\pi_b||\pi_\theta) = \sum_{a\in\mathcal{A}} \pi_b(a|\cdot)(\log \pi_b(a|\cdot) - \log \pi_\theta(\cdot|a)) \quad (11)$$

While a minimum KL divergence of 0 implies identical policies, resulting in an estimated ESS equal to N, a lower KL divergence doesn't guarantee a larger ESS. This is because importance sampling weights only consider actions chosen by clinicians. Lower KL divergence may result from aligning the evaluation and behavior policy for state-action pairs not present in the dataset, which does not influence the estimated ESS.

Therefore, we propose an alternative metric, the 'ESS divergence', that is directly related to the ESS defined in Eq. (8). This metric focuses on directly minimizing the variance in WIS weights, by constraining the policy only on the actions chosen by clinicians. We define the ESS divergence as:

$$\hat{D}_{ESS} = 1 - \frac{ESS}{N} \quad (12)$$

When D_{ESS} reaches its minimum value 0, the ESS equals N, indicating an uniform distribution of WIS weights. Conversely, a maximum D_{ESS} value of 1 means the ESS is equal to 0, implying a high variance in the WIS weights.

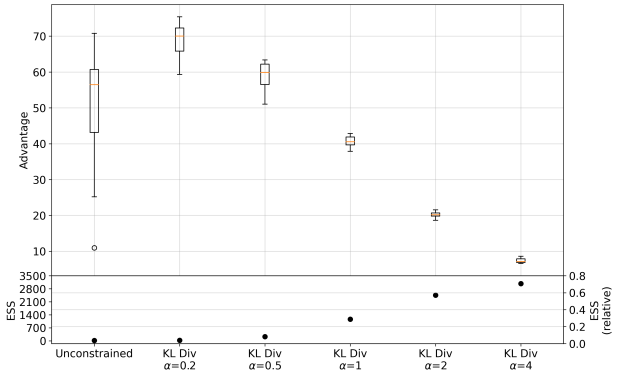
To assess the effectiveness of the evaluation policy relative to the behavior policy, we introduce the concept of 'advantage' as the difference between the policy value estimate of the policy and the behavior policy. An advantage above zero indicates that the evaluation policy reduces the estimated mortality risk in comparison to the clinician's current practice. An advantage below zero means that the evaluation policy underperforms relative to behavior policy.

The final policy is obtained through a two-phase process: an initial 'warm-up' phase with 10k steps of imitation learning, followed by 500k-step RL using the constrained policy objective defined in Eq. (10). In the imitation learning phase, only the divergence constraint is applied, such that the evaluation policy initially is a copy of the behavior policy. We compare the effect of imposing the constraints for varying values of α with a baseline in which no constraint is applied. For the KL divergence constraint, we test $\alpha \in \{0.2, 0.5, 1, 2, 4\}$, and for the ESS divergence constraint, we test $\alpha \in \{0.1, 0.2, 0.5, 1, 2\}$. Other hyperparameters such as learning rates and architecture are defined in Appendix D.

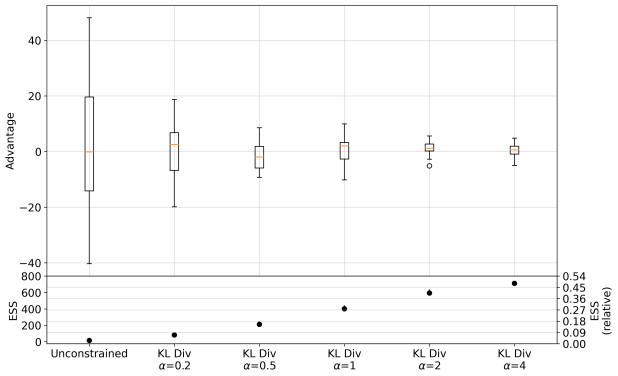
Results are gathered for 5 folds of the dataset into train, and test set. For each value of α, 3 runs with random initialization are done. In total, we learn and evaluate 15 policies for each value of α.

5.2 Results - KL Divergence

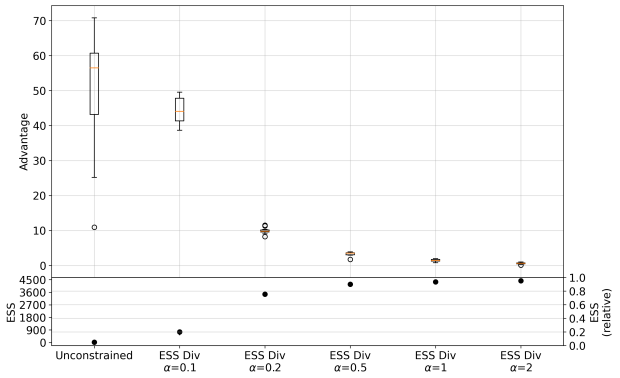
The effect of applying the KL divergence constraint during policy optimization, using varying levels of α, is illustrated for both the train (Figure 12a) and test set (Figure 12a). We start by analysing the effect of the KL divergence constraint on the estimated ESS and advantage as measured on the train set. In the unconstrained case ($\alpha = 0$), the estimated ESS ($\mu = 10$) is approximately the same as we saw in Chapter 3, suggesting that only a handful of trajectories contribute meaningfully to the estimated policy value. As α increases, and the policy regularization is stronger, there's a notable increase in ESS, alongside a tightening of the confidence intervals around the advantage. Applying the policy constraint reduces



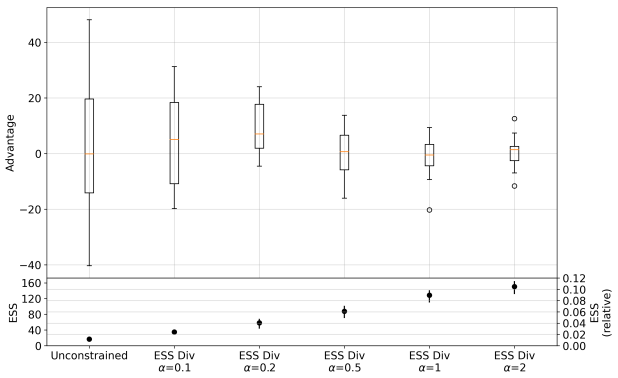
(a) *KL divergence constraint, policy evaluation on the train set*



(b) *KL divergence constraint, policy evaluation on the test set*



(c) *ESS divergence constraint, policy evaluation on the train set*



(d) *ESS divergence constraint, policy evaluation on the test set*

Figure 9: *The effect of applying the direct policy divergence constraint. For each part, the upper figure shows a boxplot of the estimated advantage for different values of α. The bottom figure shows the average ESS for different values of α, with on the right y-axis the ESS relative to the size of the set that was used. Results are gathered for 5 folds of the dataset into train, and test set. For each value of α, 3 runs with random initialization are done. In total, each boxplot consists of 15 evaluations of a policy.*

the policy search space, which explains why increasing α leads to a decrease in average advantage. An exception to this trend is a minimal KL divergence constraint ($\alpha = 0.2$), which maintains an ESS ($\mu = 12$) comparable to the unconstrained case, while achieving a high advantage with a tight confidence bound.

On the test set, higher α values also lead to a larger ESS, tightening the confidence intervals for the estimated advantage (Figure 12a). However, the mean estimated advantage on the test set does not change as the value of α increases, which centers around zero, and means no significant improvement over current clinician practice is found. The application of the KL divergence constraint results in a larger ESS, and hence an improved reliability of the policy evaluation. With our reliable policy evaluation, we find that the learned policies do not outperform the clinician practice.

5.3 Results - ESS Divergence

Figure 12c and Figure 12d illustrate the effect of applying the ESS divergence constraint during policy optimization, for different values of α, measured on the train set and test set, respectively. On the train set, the ESS divergence proves effective at achieving a large ESS, even for small α values. The estimated advantage on the train set is similar to what we observed for the KL divergence constraint, showing a decrease in estimated advantage, and less variation, as α increases.

On the test set, a different effect of the ESS constraint is observed. The growth in estimated ESS with increasing α values is much slower, compared to the training set. For instance, at $\alpha = 2$, the evaluation policy and behavior policy are so similar on the train set that nearly all of its trajectories contribute equally to the policy value estimation. In contrast, the reliability of the policy evaluation on the test set is equivalent to having a random sample of trajectories from the evaluation policy with only 10% of the size of the test set.

This discrepancy suggests that achieving a large ESS for policy evaluation of the train set does not ensure an equivalently large ESS for policy evaluation on the test set. With the KL divergence constraint at ($\alpha = 2$) the difference in relative ESS — 0.6 on the training set versus 0.4 on the test set — was considerably smaller. Because of this, the confidence bounds

around the estimated advantage on the test set do not tighten as quickly for the ESS divergence constraint as they do under the KL divergence constraint when α grows larger.

5.4 Limitations of the ESS divergence

We now explain the limited generalizability of the ESS divergence constraint. The objective of policy shaping was to improve the ESS, by reducing the difference between the evaluation and behavior policies.

Table 2: *Example episode from the train set. π_e is obtained under application of the ESS divergence constraint.*

t	a_t	$\pi_b(a_t\|s_t)$	$\pi_e(a_t\|s_t)$	$\rho_t^{(i)}$	$\rho_{1:t}^{(i)}$
1	4	0.71	0.04	0.05	0.05
2	3	0.25	0.38	1.52	0.08
3	3	0.48	0.44	0.92	0.07
4	3	0.49	0.93	1.91	0.14
5	3	0.49	0.94	1.91	0.26
6	3	0.49	0.91	1.85	0.48
7	3	0.50	0.88	1.77	0.86
8	3	0.49	0.99	2.01	1.73
9	3	0.49	0.93	1.92	3.31

Table 2 shows an illustrative example from the train set, where the evaluation policy is obtained through application of the ESS divergence constraint ($\alpha = 2$). This episode receives a low cumulative importance sampling weight $\rho_{1:T}^{(i)}$ of 3.3, despite large differences between π_e and π_b at individual timesteps. The ESS divergence constraint, in this case, inadvertently forces the policy to exploit importance sampling ratio calculations to achieve a large ESS, rather than reducing the difference between the evaluation policy and the behavior policy.

Examination of all per-step importance sampling ratios $\rho_t^{(i)}$ on the train set reveals that for a policy obtained under the KL divergence constraint ($\alpha = 4$, $ESS = 3187$) the ratios appear normally distributed around 1.2 (Figure 10a), indicating that the behavior policy and evaluation policy are similar. However, the evaluation policy obtained under application of the ESS divergence constraint ($\alpha = 2$, $ESS = 5520$) shows a distribution with a mode at 1.4, and a noticeable left tail (Figure 10b). The left tail suggests that the large ESS here is not a result from similar evaluation and behavior policies.

While similar evaluation and behavior policies guarantee a larger ESS, the opposite is not necessarily true. The ESS divergence constraint meets our objective of achieving a larger ESS on the train set. However, it does not do so through the anticipated means of aligning the evaluation policy more closely with the behavior policy, and consequently this constraint fails on the test set. This distinction underscores a critical insight into the constraint's effectiveness and limited applicability in policy shaping.

To conclude, we enhanced the reliability of OPE through policy shaping, by applying a direct policy constraint, focused on reducing the difference between the evaluation policy and behavior policy. As the ESS increased, the variation in policy value estimates decreased, boosting our confidence in the policy evaluation. The KL divergence constraint proved more effective for achieving OPE reliability than the ESS divergence. However, a distinction emerged between the train and test data. While the constraints led to high advantage and improved ESS on the train data, no policies were learned that consistently outperformed the behavior policy on the test data.

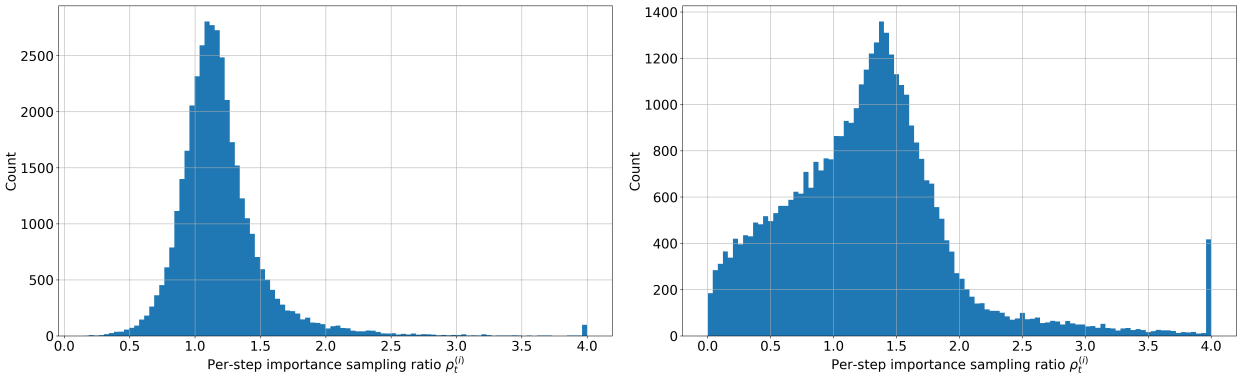
6 Discussion

6.1 Principal Findings

A central challenge in OPE is the distributional shift. For IS-based evaluation methods, we observe that when the evaluation policy and behavior policy differ too much, a few patient trajectories dominate the policy value estimation, undermining the reliability of the policy value estimate. We use the ESS as a diagnostic to assess this reliability. Applying (unconstrained) tabular Q-learning, similar to Peine et al., did not yield policies which could reliably be evaluated.

Policy shaping, through the introduction of a divergence constraint in the policy learning objective, successfully increases the ESS. This mitigates the problem of a few trajectories dominating the policy value estimation. However, with reliable policy evaluation, we found no policies that consistently outperformed clinician practice. The KL divergence constraint generalized better to unseen patients than the ESS divergence, which achieved large ESS in the train set without actually reducing the difference between the evaluation and behavior policy.

Accurately estimating the true behavior policy is key to effective OPE. We selected the optimal estimator for the behavior policy by comparing discrimination and calibration performance. We chose the MCE[22] as the decisive metric to compare the close calibration performance, as it captures moderate calibration performance in a single score. Including the previous clinician action as a feature was significantly beneficial for performance.



(a) *KL divergence constraint ($\alpha = 4$, ESS=3187)* **(b)** *ESS divergence constraint ($\alpha = 2$, ESS=5520)*

Figure 10: *Histogram of all individual importance sampling ratios ρ for an evaluation policy learned under a specific divergence constraint. For every (s_t, a_t) pair in the train set, the ratio $\rho = \pi_e(a|s)/\pi_b(a|s)$ is computed. These ratios are clipped at a maximum value of 4, and then binned into 100 equal-width bins.*

6.2 Connections to Related Work

Our findings challenge the conclusion of Peine et al. [8] and Komorowski et al. [9], who advocate for prospective clinical trials of the proposed RL policies. The lack of reported ESS in their work raises concerns about the reliability of their policy evaluations. As demonstrated by Gottesman et al. [7] and reinforced by our work, the lack of uncertainty reporting can lead to overestimating policy performance. Evaluating these policies in real-world clinical trials potentially results in harmful clinical decisions.

To our knowledge, there exists no standardisation for reporting uncertainty in OPE within healthcare, and this needs to be addressed. For this purpose, we propose examining WIS weights and using the ESS as a reliability diagnostic. The ESS is a practical diagnostic due to its simple computation and intuitive interpretation. In our running example, we observed a correlation between larger ESS values and reduced uncertainty in policy evaluation. Integrating a standard practice for reporting OPE uncertainty into the review process will help ensure the safety and trustworthiness of new applications of RL in healthcare.

We also build upon previous work emphasizing the importance of a well-calibrated estimated behavior policy for effective OPE[19]. Our study provides a practical example of behavior policy assessment, and its associated challenges. Any IS-based method requires an estimate of the behavior policy, yet evaluating the calibration of this estimate is not common practice. We advocate to make it standard practice in the field to evaluate the calibration in the moderate sense, e.g. by visual inspection of the calibration curve or by computing the ECE and MCE. As we found it difficult to choose the best metric of moderate calibration for multiple actions, further research may be necessary to establish best practices for evaluating multiclass moderate calibration.

6.3 Limitations

Choices in data extraction and preprocessing (inclusion criteria, sampling window, feature selection) impact the observational dataset and consequently affect the results. Changing the sampling frequency or maximum episode length influences the severity of the curse of horizon and observed variance in the WIS weights. Additionally, data quality (noise, missingness) was a limitation for some features. Our choice to aggregate data into 8-hour windows reduces information density, and certain features (lab values) accessible retrospectively might not be available in a prospective evaluation setting. Discretization of actions remove information about their ordinality, which could be leveraged by policies based on continuous actions. However, this would require major architectural changes. Despite these limitations, we believe our findings on the limited reliablity of WIS evaluation still hold when the behavior policy and evaluation policy differ too much.

We did not individually tune hyperparameters for each divergence constraint and α value separately. Instead, we used a single hyperparameter set for all experiments. More extensive hyperparameter tuning could have led to finding better policies. For each α value, we chose to do 5 runs with 3 random initilizations. After 15 runs, a visual inspection of the results (Appendix F) suggested we had sufficient data to observe trends. Additional runs with random initialization could further validate the observed patterns and increase confidence in our conclusions.

Our current RL problem formulation highlights the difficulty of learning policies that both generalize effectively to unseen patients and reliably outperform

current clinician practice. Whether the task of finding individualized mechanical ventilation regimes is feasible with the current setup and dataset remains an open question.

We use Kong's ESS approximation (Eq. (8)) because of its simplicity and direct connection to the variance of the WIS weights. A key feature is that it depends only on the weights, and not on the reward function, while the theoretical definition does. Although a practical advantage, this makes the approximation not always accurate[18], such as in cases where the variance in the observed rewards is very small. In these cases, an ESS approximation that incorporates the reward function is more suitable. Nonetheless, given its nature as a diagnostic, it is still useful to provide an estimate of the reliability of the policy value estimates.

The direct policy constraint approach might limit generalization through overly constraining the policy based on the behavior policy estimated on the train set. Alternative approaches are worth exploring, such as implicit policy constraints or value regularization[22]. However, these methods do not explicitly reduce the difference between the evaluation and the behavior policy, and may fail to improve the ESS.

6.4 Future Directions

Optimizing directly for the ESS through the 'ESS divergence' does not give the desired result, as it does not lead to closer alignment of the evaluation and behavior policy. Perhaps the ESS is not the diagnostic for reliability we should consider, and we should explore alternative metrics for OPE reliability. While intuitive interpretation is challenging, the KL divergence between evaluation and behavior policies could be considered, as it directly quantifies the difference between the evaluation and behavior policy. Additionally, alternative divergence metrics, focused on reducing the difference in policies and the variance in weights simultaneously, are interesting to explore.

Testing the direct policy constraint in a simulated environment could validate whether policy evaluations deemed reliable truly lead to policies that consistently perform well in online evaluations. Such an environment would also enable assessing whether the direct policy constraint contributes to overfitting on training data and poor generalizability.

Further investigation into the influence of behavior policy miscalibration on the reliability of OPE is needed. Additionally, further research could examine the effect of behavior policy miscalibration on the effectiveness of direct policy constraints. This research could further motivate the critical assessment of behavior policy calibration as a standard practice in OPE.

In this work, we only considered IS-based methods, and excluded model-based evaluation methods. While bootstrapping can still be applied to model-based evaluation methods[30], the ESS diagnostic is specific to IS. Extending our work to other OPE methods would require reliability metrics for model-based evaluation. However, intuitive explanations for such metrics with model-based methods likely present a challenge.

6.5 Conclusion

We highlight the importance of a careful approach towards the application of RL in ICU settings, advocating for a critical assessment of the reliability of performed OPE. Future studies should be mindful to incorporate diagnostic tools for uncertainty, such as the ESS. They should refrain from drawing conclusion based on policies with small ESS, and instead consider the policy evaluation unreliable. Adopting such measures can enhance confidence, ensuring that only effective and reliable RL policies are considered for real-world clinical trials, thereby minimizing the risk of introducing ineffective or potentially harmful interventions into clinical practice.

6.6 Data & Code Availability

All experiments are done using the MIMIC-IV database[11], and the code used is openly available at github.com/BasVolkers/MechanicalVentilationRL.

References

[1] Ewan C Goligher, Niall D Ferguson, and Laurent J Brochard. "Clinical challenges in mechanical ventilation". In: *The Lancet* 387.10030 (Apr. 2016), pp. 1856–1866. ISSN: 0140-6736. DOI: 10.1016/s0140-6736(16)30176-3. URL: http://dx.doi.org/10.1016/S0140-6736(16)30176-3.

[2] Adnan Liaqat et al. "Evidence-Based Mechanical Ventilatory Strategies in ARDS". In: *Journal of Clinical Medicine* 11.2 (Jan. 2022), p. 319. ISSN: 2077-0383. DOI: 10.3390/jcm11020319. URL: http://dx.doi.org/10.3390/jcm11020319.

[3] Marcelo B.P. Amato et al. "Driving Pressure and Survival in the Acute Respiratory Distress Syndrome". In: *New England Journal of Medicine* 372.8 (Feb. 2015), pp. 747–755. ISSN: 1533-4406. DOI: 10.1056/nejmsa1410639. URL: http://dx.doi.org/10.1056/NEJMsa1410639.

[4] Tài Pham, Laurent J. Brochard, and Arthur S. Slutsky. "Mechanical Ventilation: State of the Art". In: *Mayo Clinic Proceedings* 92.9 (2017), pp. 1382–1400. ISSN: 0025-6196. DOI: https://doi.org/10.1016/j.mayocp.2017.05.004. URL: https://www.sciencedirect.com/science/article/pii/S0025619617303245.

[5] Martijn Otten et al. "Does Reinforcement Learning Improve Outcomes for Critically Ill Patients? A Systematic Review and Level-of-Readiness Assessment". In: *Critical Care Medicine* (Nov. 2023). ISSN: 0090-3493. DOI: 10.1097/ccm.0000000000006100. URL: http://dx.doi.org/10.1097/CCM.0000000000006100.

[6] Sergey Levine et al. "Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems". In: *CoRR* abs/2005.01643 (2020). arXiv: 2005.01643. URL: https://arxiv.org/abs/2005.01643.

[7] Omer Gottesman et al. "Evaluating Reinforcement Learning Algorithms in Observational Health Settings". In: *CoRR* abs/1805.12298 (2018). arXiv: 1805.12298. URL: http://arxiv.org/abs/1805.12298.

[8] Arne Peine et al. "Development and validation of a reinforcement learning algorithm to dynamically optimize mechanical ventilation in critical care". In: *npj Digital Medicine* 4.1 (Feb. 2021). ISSN: 2398-6352. DOI: 10.1038/s41746-021-00388-6. URL: http://dx.doi.org/10.1038/s41746-021-00388-6.

[9] Matthieu Komorowski et al. "The Artificial Intelligence Clinician learns optimal treatment strategies for sepsis in intensive care". In: *Nature Medicine* 24.11 (Oct. 2018), pp. 1716–1720. ISSN: 1546-170X. DOI: 10.1038/s41591-018-0213-5. URL: http://dx.doi.org/10.1038/s41591-018-0213-5.

[10] Flemming Kondrup et al. *Towards Safe Mechanical Ventilation Treatment Using Deep Offline Reinforcement Learning*. 2022. arXiv: 2210.02552 [cs.LG].

[11] Alistair E. W. Johnson et al. "MIMIC-IV, a freely accessible electronic health record dataset". In: *Scientific Data* 10.1 (Jan. 2023). ISSN: 2052-4463. DOI: 10.1038/s41597-022-01899-x. URL: http://dx.doi.org/10.1038/s41597-022-01899-x.

[12] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. Second. The MIT Press, 2018. URL: http://incompleteideas.net/book/the-book-2nd.html.

[13] Christopher J. C. H. Watkins and Peter Dayan. "Q-learning". In: *Machine Learning* 8.3–4 (May 1992), pp. 279–292. ISSN: 1573-0565. DOI: 10.1007/bf00992698. URL: http://dx.doi.org/10.1007/BF00992698.

[14] Josiah P. Hanna, Peter Stone, and Scott Niekum. *Bootstrapping with Models: Confidence Intervals for Off-Policy Evaluation*. 2018. arXiv: 1606.06126 [cs.AI].

[15] Omer Gottesman et al. "Interpretable Off-Policy Evaluation in Reinforcement Learning by Highlighting Influential Transitions". In: *CoRR* abs/2002.03478 (2020). arXiv: 2002.03478. URL: https://arxiv.org/abs/2002.03478.

[16] Qiang Liu et al. *Breaking the Curse of Horizon: Infinite-Horizon Off-Policy Estimation*. 2018. arXiv: 1810.12429 [cs.LG].

[17] Augustine Kong. "A Note on Importance Sampling using Standardized Weights". In: (1992). URL: https://victorelvira.github.io/papers/kong92.pdf.

[18] Víctor Elvira, Luca Martino, and Christian P. Robert. "Rethinking the Effective Sample Size". In: *International Statistical Review* 90.3 (Apr. 2022), pp. 525–550. ISSN: 1751-5823. DOI: 10.1111/insr.12500. URL: http://dx.doi.org/10.1111/insr.12500.

[19] Aniruddh Raghu et al. "Behaviour Policy Estimation in Off-Policy Policy Evaluation: Calibration Matters". In: *CoRR* abs/1807.01066 (2018). arXiv: 1807.01066. URL: http://arxiv.org/abs/1807.01066.

[20] Ben Van Calster et al. "A calibration hierarchy for risk models was defined: from utopia to empirical data". en. In: *J. Clin. Epidemiol.* 74 (June 2016), pp. 167–176.

[21] Ben Van Calster et al. "Calibration: the Achilles heel of predictive analytics". en. In: *BMC Med.* 17.1 (Dec. 2019), p. 230.

[22] Telmo Silva Filho et al. "Classifier calibration: a survey on how to assess and improve predicted class probabilities". In: *Machine Learning* 112.9 (May 2023), pp. 3211–3260. ISSN: 1573-0565. DOI: 10.1007/s10994-023-06336-7. URL: http://dx.doi.org/10.1007/s10994-023-06336-7.

[23] David W. Hosmer and Stanley Lemeshow. *Applied Logistic Regression*. Wiley, Sept. 2000. ISBN: 9780471722144. DOI: 10.1002/0471722146. URL: http://dx.doi.org/10.1002/0471722146.

[24] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *CoRR* abs/1603.02754 (2016). arXiv: 1603.02754. URL: http://arxiv.org/abs/1603.02754.

[25] Leo Breiman. In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 0885-6125. DOI: 10.1023/a:1010933404324. URL: http://dx.doi.org/10.1023/A:1010933404324.

[26] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York, 2009. ISBN: 9780387848587. DOI: 10.1007/978-0-387-84858-7. URL: http://dx.doi.org/10.1007/978-0-387-84858-7.

[27] Rafael Figueiredo Prudencio, Marcos R. O. A. Maximo, and Esther Luna Colombini. "A Survey on Offline Reinforcement Learning: Taxonomy, Review, and Open Problems". In: *IEEE Transactions on Neural Networks and Learning Systems* (2023), pp. 1–. DOI: 10.1109/TNNLS.2023.3250269.

[28] Yifan Wu, George Tucker, and Ofir Nachum. "Behavior Regularized Offline Reinforcement Learning". In: *CoRR* abs/1911.11361 (2019). arXiv: 1911.11361. URL: http://arxiv.org/abs/1911.11361.

[29] Natasha Jaques et al. *Way Off-Policy Batch Deep Reinforcement Learning of Implicit Human Preferences in Dialog*. 2019. arXiv: 1907.00456 [cs.LG].

[30] Botao Hao et al. *Bootstrapping Fitted Q-Evaluation for Off-Policy Inference*. 2022. arXiv: 2102.03607 [stat.ML].

[31] Luca Martino, Víctor Elvira, and Francisco Louzada. "Effective sample size for importance sampling based on discrepancy measures". In: *Signal Processing* 131 (Feb. 2017), pp. 386–401. ISSN: 0165-1684. DOI: 10.1016/j.sigpro.2016.08.025. URL: http://dx.doi.org/10.1016/j.sigpro.2016.08.025.

[32] Cátia M. Salgado et al. "Missing Data". In: *Secondary Analysis of Electronic Health Records*. Springer International Publishing, 2016, pp. 143–162. ISBN: 9783319437422. DOI: 10.1007/978-3-319-43742-2_13. URL: http://dx.doi.org/10.1007/978-3-319-43742-2_13.

A Effective Sample Size

The effective sample size is a measure of relative efficiency between a Monte carlo estimator for the policy value and an Importance sampling estimator for the policy value. We want to estimate the policy value for a policy π_e, while having only trajectories generated by a different policy π_b.
Let $h(x_i)$ be a function that measures the return of a trajectory x_i: $h(x_i) = \sum_{t=0}^{T_i} \gamma^t r_t^i$. If we had trajectories generated by the evaluation policy, as we do in the online RL setting, then we could compute the policy value using a Monte Carlo (MC) estimate $\hat{I}$:

$$\hat{I} = \frac{1}{N} \sum_{i=1}^{N} h(x_i) \quad \text{where} \quad x_i \sim \pi_e \tag{13}$$

Since we don't have trajectories generated by π_e, but instead have trajectories generated by π_b, we have to use an importance sampling (IS) estimate. Let w_i be the normalized importance sampling weight of trajectory i as defined in equation 5. The (W)IS estimate $\bar{I}$ for the policy value is then given by:

$$\bar{I} = \sum_{i=1}^{N} w_i h(x_i) \quad \text{where} \quad x_i \sim \pi_b \tag{14}$$

The effective sample size is then defined as the ratio of the variances of the estimators[31]:

$$ESS = N \frac{\text{var}_{\pi_e}[\hat{I}]}{\text{var}_{\pi_b}[\bar{I}]} \tag{15}$$

Informally, the ESS represents the number of samples from π_e required to obtain a Monte Carlo estimator $\hat{I}$ with the same efficiency as the IS estimator $\bar{I}$. It measures how many samples drawn from π_e are equivalent to the N weighted samples drawn from π_b.

With several assumptions and approximations, a pratical ESS approximation can be found (see also equation 8):

$$\widehat{ESS} = \frac{1}{\sum_{i=1}^{N} w_i^2} \tag{16}$$

A derivation of this approximation and a list of all assumptions can be found in the work by Elvira et al.[18], and the original introduction to the ESS in the work by Kong et al.[17].

A.1 Illustrative examples

Consider three cases that illustrate the estimator: a uniform distribution over all weights, a single weight dominating all other weights, and a uniform distribution over a select number over weights.
When $w_i = \frac{1}{N}$ for $1 \leq i \leq N$, i.e. all weights are uniformly distributed, then the ESS is computed as:

$$ESS = \frac{1}{\sum_{i=1}^{N} w_i^2} = \frac{1}{\sum_{i=1}^{N} (\frac{1}{N})^2} = \frac{1}{N \frac{1}{N^2}} = \frac{1}{\frac{1}{N}} = N \tag{17}$$

When $w_1 = 1$ and $w_i = 0$ for $2 \leq i \leq N$, i.e. a single weight dominates all other weights, then the ESS is computed as:

$$ESS = \frac{1}{\sum_{i=1}^{N} w_i^2} = \frac{1}{\sum_{i=1}^{1} 1^2 + \sum_{i=2}^{N} 0^2} = \frac{1}{1} = 1 \tag{18}$$

When $w_i = \frac{1}{10}$ for $1 \leq i \leq 10$ and $w_i = 0$ for $11 \leq i \leq N$, i.e. a uniform distribution over a select number over weights, then the ESS is computed as:

$$ESS = \frac{1}{\sum_{i=1}^{N} w_i^2} = \frac{1}{\sum_{i=1}^{10} (\frac{1}{10})^2 + \sum_{i=11}^{N} 0^2} = \frac{1}{10 \frac{1}{10^2} + 0} = \frac{1}{\frac{1}{10}} = 10 \tag{19}$$

Note that the ESS does not depend on N. Having a dataset of $N = 10$ trajectories, all having weight $w_i = 0.1$ results in the same ESS as having a dataset with $N = 100$ trajectories, of which 10 trajectories have weight $w_i = 0.1$ ($1 \leq i \leq 10$).

A.2 Effective Sample Size as a Divergence Metric

The ESS divergence is defined as:

$$\hat{D}_{ESS} = 1 - \frac{ESS}{N} \tag{20}$$

If $\pi_b(a|s) = \pi_e(|s)$ for $(s, a) \sim D$, i.e. the behavior policy and evaluation policy are equal on the state action pairs in the dataset, then all importance sampling weights are uniformly distributed, having weight $w_i = \frac{1}{N}$. The ESS is then at its maximum value N (Eq. 17), and the ESS divergence at its minimum value $\hat{D}_{ESS} = 0$.

Conversely, if the behavior policy and evaluation policy are different and consequently a single importance sampling weight dominates all other weights, then the ESS is at its minimum value of 1 (Eq. 18). The ESS divergence is then at its maximum value $\hat{D}_{ESS} = 1$.

B Behavior policy evaluation metrics

In this work we evaluate an estimated behavior policy on discrimination and calibration. For discrimination we only evaluate the logloss. For calibration we evaluate the brier score, estimated calibration error (ECE), and maximum calibration error (MCE).

B.1 Proper scoring rules

Both Brier score and logloss are proper scoring rules[22], meaning that optimal values are obtained when the classifier predicts the true probabilities of class occurrence. We start by defining the logloss for the binary case. For the binary case it is defined at the negative log-likelihood of a logistic model. For N data points with true label $y_i \in \{0,1\}$ and probability estimate $p_i = \mathrm{P}(y_i = 1)$, the log loss is:

$$\text{Log Loss} = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(p_i) + (1-y_i)\log(1-p_i)) \tag{21}$$

It can be extended to the multiclass case with K classes by considering the true label $y_{ij} = 1$ if data i has class j, and a probability estimate for class j: p_{ij}= $\mathrm{P}(y_i = j)$. The log loss (or cross entropy loss) is then defined as:

$$\text{Log Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K} y_{ij}\log p_{ij} \tag{22}$$

Using the same notation, we can easily define the brier score:

$$\text{Brier score} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}(p_{ij} - y_{ij})^2 \tag{23}$$

B.2 Calibration error

The calibration error measures the discrepancy between binned predicted probabilities and true observed proportions. We again start by defining the calibration error for the binary case, and then extend it to the multiclass. The predicted probabilities are binned into m bins: $\mathcal{B}_m$. For each bin we can compute the average predicted probability $s(\mathcal{B}_m) = \frac{1}{|\mathcal{B}_m|}\sum_{i\in\mathcal{B}_m} p_i$ and proportion of observed positives $y(\mathcal{B}_m) = \frac{1}{|\mathcal{B}_m|}\sum_{i\in\mathcal{B}_m} y_i$. A perfectly calibrated classifier would have that for every bin, the average predicted probability and proportion of observed positives is equal. The calibration error is expressed in terms of the gap between these. The binary estimated calibration error is computed as the average gap across all bins, and the binary maximum calibration error is computed as the maximum gap across all bins:

$$\text{ECE}_{\text{binary}} = \sum_{m=1}^{M}\frac{|\mathcal{B}_m|}{N}|y(\mathcal{B}_m) - s(\mathcal{B}_m)| \tag{24}$$

$$\text{MCE}_{\text{binary}} = \max_m |y(\mathcal{B}_m) - s(\mathcal{B}_m)| \tag{25}$$

In the multiclass case, we take class j as the positive class and all other classes as the negative class. With $\mathcal{B}_{m,j}$ as the m-th bin of class j we again compute the average predicted probability $s(\mathcal{B}_{m,j}) = \frac{1}{|\mathcal{B}_{m,j}|}\sum_{i\in\mathcal{B}_{m,j}} p_{ij}$ and proportion of observed positives $y(\mathcal{B}_{m,j}) = \frac{1}{|\mathcal{B}_{m,j}|}\sum_{i\in\mathcal{B}_{m,j}} \mathbb{1}[y_i = j]$. The classwise estimated calibration error is defined as the mean of all class-j estimated calibration errors:

$$\text{ECE}_{\text{classwise}} = \frac{1}{K}\sum_{j=1}^{K}\sum_{m=1}^{M}\frac{|\mathcal{B}_{m,j}|}{N}|y(\mathcal{B}_{m,j}) - s(\mathcal{B}_{m,j})| \tag{26}$$

The classwise maximum calibration error can be defined in multiple ways. In this work we use the mean of the class-j maximum calibration errors:

$$\text{MCE}_{\text{classwise}} = \frac{1}{K}\sum_{j=1}^{K}\max_m |y(\mathcal{B}_{m,j}) - s(\mathcal{B}_{m,j})| \tag{27}$$

The ECE and MCE represent the calibration error differently. For a particular class, the MCE focuses on the single bin with the largest discrepancy between predicted probabilities and observed proportions. An outlier in one bin can drastically influence the MCE value. The ECE averages the errors across all bins, mitigating the impact of a single outlier bin.

C Data Extraction & Preprocessing

The MIMIC-IV database contains a total of 61,532 ICU admissions, of which 7,281 met the inclusion criteria. A 'mechanical ventilation event' is defined by the following criteria: the first presence of a set tidal volume starts the event, and the event is continued until the ventilator is switched to a mode that is not of interest. Modes of interest include all modes where a tidal volume has to be set and thus a driving pressure is available. If the mode is switched back to a mode of interest within a timeframe of 2 hours, the mechanical ventilation event is considered to continue.

The inclusion criteria were: patient age at least 18, documented 90-day mortality, documented vital signs, documented driving pressure, and mechanical event of at least 24 hours. From each ICU stay, the first mechanical ventilation event is selected. A week of data starting from the point of intubation is collected, and sampled in 8-hour timesteps.

Data was extracted using Google Bigquery. A collection of 34 features, including demographics, vital signs, lab values, and fluid balance, was selected, based on clinical relevance. If multiple values were present in the 8-hour time window, a time-weighted average was computed. Outliers were removed by considering clinically impossible values.

Category	Features
Demographics	Age, gender, height, weight, bmi
Vital signs	Heart rate, SpO2, temperature, diastolic blood pressure, systolic blood pressure, mean arterial pressure, shock index, respiratory rate
Lab values	PaO2, PaCO2, PF ratio, pH, base excess, lactace, carbon dioxide, SO2, glucose, creatine, bilirubin, hemoglobin, hematocrit
Scores	Charslon comorbidity index, GCS, SOFA, SIRS
Other	First ICU stay, respiratory rate set on ventilator, cumulative fluid balance since admission
Outcome	90-day mortality

Table 3: *Patient features*

To address problems with missing data, a mixed method of imputation was used. First, a (time-limited) sample-and-hold / last-value-carried forward approach was used to impute the majority of missing values. Then, a k-nearest-neighbours (KNN)[32] imputation method was used to interpolate for the remaining missing data. Before applying KNN imputation, normally distributed data was standardized, log-normal data was log-transformed before standardizing, and binary data was centred around zero. The distribution of each feature was assessed visually with frequency histograms. KNN imputation was applied in blocks of 10,000 rows of patient data.

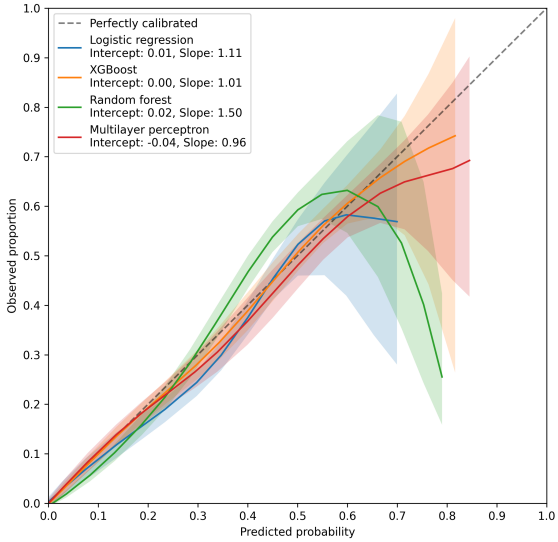
Bin	0	1	2	3	4
Driving Pressure	≤ 7.5	≤ 10	≤ 12.5	≤ 15	> 15
Count	7070	18142	21145	17508	14529

Table 4: *Distribution of the chosen action by clinicians. Total amount of decision time instances: 78394*

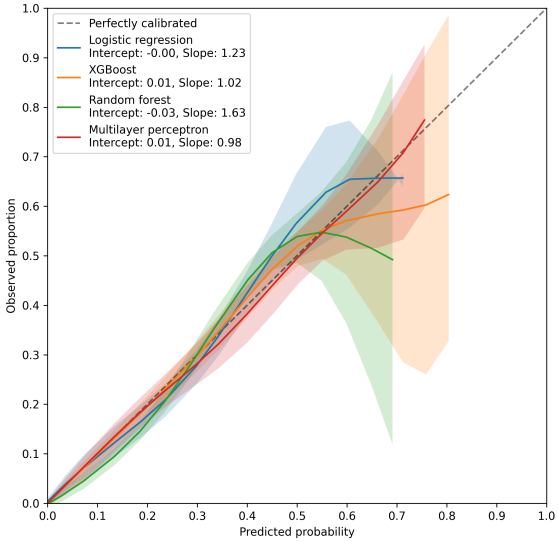
D Deep Reinforcement Learning Hyperparameters

actor learning rate	1e-4
critic learning rate	1e-4
batch size	32
number of critics	4
hidden layers	[256, 256]
activation	ReLU

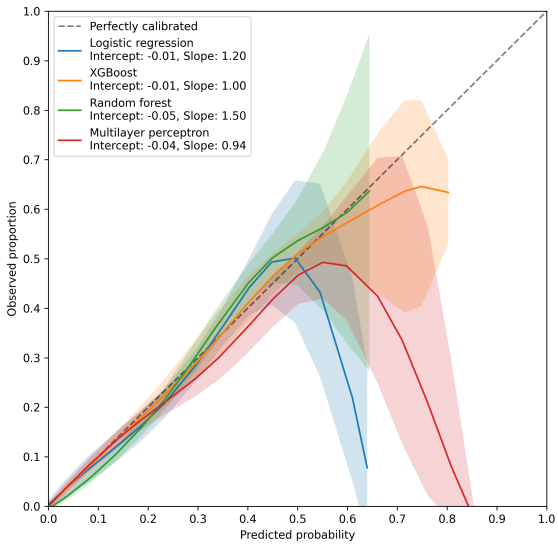
E Classifier calibration performance



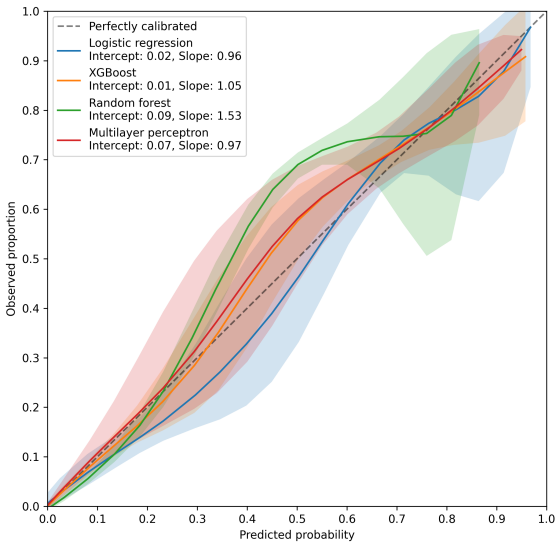
(a) *Action 2*



(b) *Action 3*



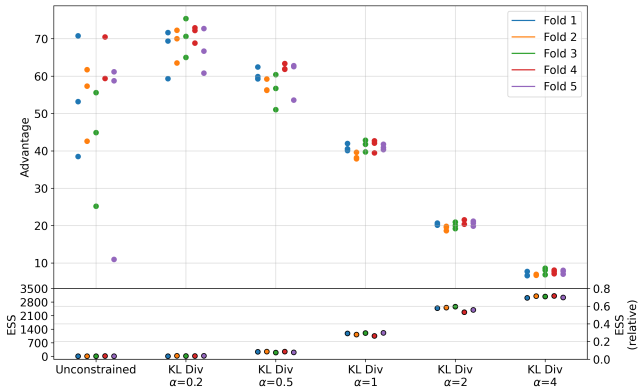
(c) *Action 4*



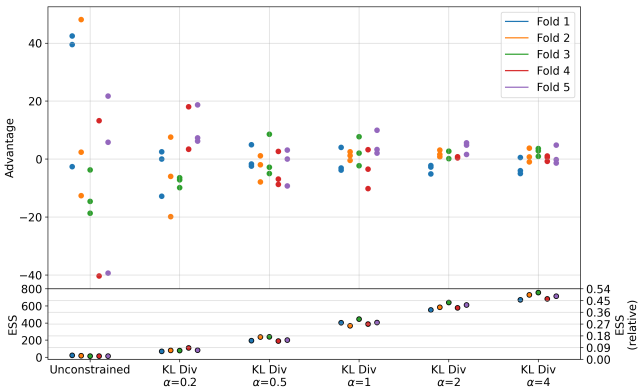
(d) *Action 5*

Figure 11: *Calibration curves of different classifiers for behavior policy estimation. Shown curves are for the actions not shown in the main text. Shaded areas show the 0.05 and 0.95 quantiles, estimated by randomly splitting the dataset 20 times.*

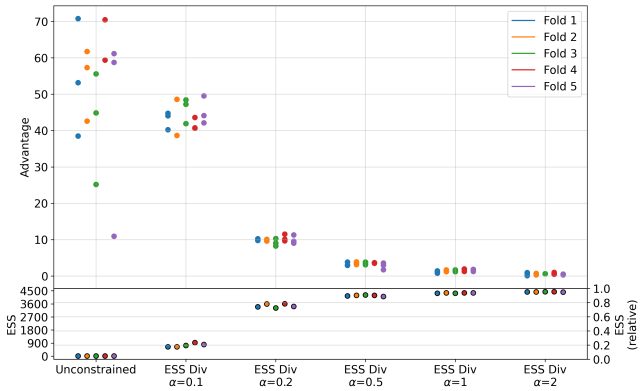
F Policy shaping results



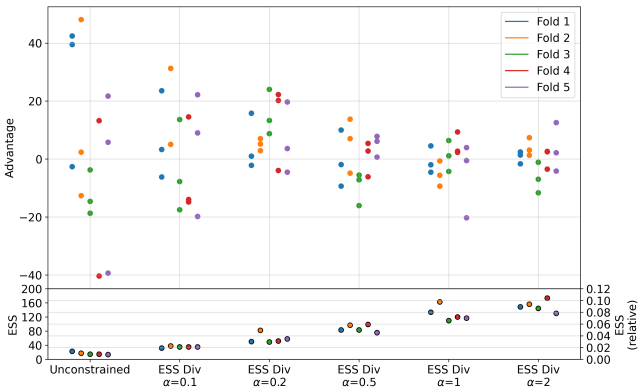
(a) *KL divergence constraint, policy evaluation on the train set*



(b) *KL divergence constraint, policy evaluation on the test set*



(c) *ESS divergence constraint, policy evaluation on the train set*



(d) *ESS divergence constraint, policy evaluation on the test set*

Figure 12: *The effect of applying the direct policy divergence constraint. For each part, the upper figure shows the estimated advantage for different values of α, and the five folds of the dataset. The bottom figure shows the average ESS each value of α and fold of the dataset, with on the right y-axis the ESS relative to the size of the set that was used. For each value of α and fold, 3 runs with random initialization are done.*