Offline Reinforcement Learning for Community-Acquired Pneumonia Management: A Feasibility Study

Alex Beeson[1,2] (✉), Keith Couper[2,3], and Giovanni Montana[1,4,5]

[1] Warwick Manufacturing Group, University of Warwick, Coventry, UK
{alex.beeson,g.montana}@warwick.ac.uk
[2] Warwick Medical School, University of Warwick, Coventry, UK
k.couper@warwick.ac.uk
[3] Critical Care Unit, University Hospitals Birmingham NHS Foundation Trust, Birmingham, UK
[4] Department of Statistics, University of Warwick, Coventry, UK
[5] Alan Turing Institute, London, UK

Abstract. Community-acquired pneumonia (CAP) remains a leading cause of hospital admission and mortality requiring dynamic clinical decision making as patients' conditions evolve. In this work, we formulate the management of CAP as a sequential decision-making problem and utilise reinforcement learning (RL) as a framework for discovering improved treatment strategies. We leverage a large-scale repository of routinely collected hospital data from the PIONEER hub and conduct an offline RL investigation under real-world complexities such as irregular sampling, missingness and variable treatment patterns. Through an extensive data transformation pipeline, we construct state-action trajectories suitable for RL and then train and evaluate policies via conservative Q-learning and fitted Q-evaluation, achieving initial—though modest—improvements in reducing 30-day mortality. In addition to these preliminary outcomes, our findings underscore the need for refined offline RL methods and rigorous validation to fully realize the potential of using large routine healthcare databases like PIONEER for clinical decision support.

Keywords: offline reinforcement learning · community-acquired pneumonia

1 Introduction

Pneumonia is an infection of the lungs that inhibits oxygen intake [13]. Although most cases are mild-to-moderate and patients respond well to standard treatments, serious complications can arise in vulnerable populations such as children, older adults and those with comorbidities. Pneumonia is typically classified by its site of initial infection—community-acquired pneumonia (CAP) versus hospital-acquired pneumonia (HAP). In the UK, CAP incidence rises sharply with age,

from 7.99 per 1,000 among those aged 65+ to 41.94 per 1,000 among those aged 90+ [25]. Pneumonia causes roughly 29,000 deaths per year, making it the third leading cause of lung-disease mortality. In hospitalized patients, 5–15% of patients die within 30 days, increasing to 30% for those admitted to intensive care units [5].

Dynamic treatment regimes (DTRs) provide individualized, time-adaptive strategies to manage patients as their clinical status evolves [4]. Conventional optimization often relies on sequential multiple assignment randomized trials (SMARTs), in which treatments are re-randomized at defined decision points [27]. However, these trials can be costly and typically require simplified intervention timings and treatment options that limit their applicability in practice [3]. Reinforcement learning (RL) has emerged as a powerful framework for discovering more flexible DTRs, showing promise in areas such as drug dosing, intervention timing, laboratory test scheduling and targeting, and more [41]. CAP presents a compelling case study for RL since it is both prevalent and clinically complex, requiring repeated, individualised treatment decisions that incorporate factors such as infection severity, comorbidities and changing clinical characteristics [31].

One advantage of modelling the management of CAP as a sequential decision-making problem is the existence of large repositories of retrospective data, which RL can exploit *offline*—i.e. purely from existing patient trajectories—rather than through prospective experimentation. Offline RL is particularly appealing in healthcare because ethical, logistical, and safety constraints typically preclude trial-and-error interactions or additional randomization [22]. By using historical data, the RL agent can learn policies that might improve health outcomes without placing patients at risk.

In this paper, we investigate whether offline RL applied to retrospective CAP patient data can yield improved policies for reducing 30-day mortality. By framing CAP management as a sequential decision-making task, we report initial, though modest, performance gains using RL-based approaches, highlighting how conservative RL strategies help mitigate extrapolation errors arising when real-world data only partially cover the space of possible actions. Our main contribution is an in-depth exploration of the challenges involved in applying RL to large-scale, routinely collected clinical data that exhibits features such as irregular sampling, extensive missingness and highly variable treatment patterns. Taken together, these findings underscore the promise of RL-based DTRs for pneumonia management and the substantial methodological effort required to transform retrospective health records into effective decision support tools.

2 Related work

Several studies have already investigated the use of RL to optimize distinct aspects of clinical care, demonstrating RL's potential in guiding medical decisions such as drug dosing and treatment timing. Examples include optimal intravenous fluid and vasopressor dosing for sepsis [14], morphine titration for pain relief [23],

heparin anticoagulation regimens [21], chemotherapy scheduling in cancer [40] and radiotherapy scheduling [33]. Researchers have also applied RL to weaning patients from mechanical ventilation [28], scheduling laboratory tests [6] and targeting specific laboratory test values [36]. While these works showcase the promise of RL in a variety of clinical contexts, they often either assume some degree of ongoing interaction with an environment (e.g. simulations) or depend on data-collection protocols not readily available in routine care.

Offline RL aims to learn optimal policies solely from retrospective data, addressing ethical, logistical and safety barriers that preclude further data gathering in many settings such as healthcare. However, directly applying established approaches such as Q-learning to fixed datasets can lead to severe overestimation bias, particularly when the learned policy evaluates actions outside the data distribution [19,9]. Many offline RL methods have thus been developed to mitigate this bias. For continuous action spaces, solutions often involve policy constraints [8], conservative value estimation [16], uncertainty estimation [1], in-distribution learning [15] or hybrid approaches [2]. Several have been successfully adapted to discrete settings. For example, discrete BCQ [7] limits actions for target Q-values to those with high probability under a learned behaviour policy. In CQL [16], Q-values for actions within the dataset are "pushed up" while out-of-distribution actions are "pushed down." Finally, discrete IQL [24] applies in-distribution learning via expectile regression and infers policies using advantage weighted behavioural cloning. These methods, while still emerging, offer promising tools to address the unique challenges of working with static hospital databases where patient safety and limited interaction make active data collection infeasible.

3 PIONEER data hub

PIONEER is a health data research hub for acute care within University Hospitals Birmingham (UHB) NHS Foundation Trust. The hub provide secure access to routinely collected healthcare records that undergo a rigorous curation procedure to ensure high quality, including removing malformed fields, de-duplicating records, mapping items to standard ontologies and other cleansing processes.

For our study, we requested records of adults (18+ years) diagnosed with pneumonia based on administrative coding of diagnoses (ICD-10 and SNOMED codes) or searches of key terms in medical records (i.e. CURB-65), who received antibiotics within 48 hours of admission. CURB-65 is a diagnostic metric used to determine the severity of CAP upon presenting in hospital [20] while ICD-10 and SNOMED are medical classification systems used by the NHS. A full list of codes is provided in the Appendix [1]. This initial extraction yielded 47,972 care spells for 36,885 patients for the time period April 2018 and September 2022.

Structure wise, each admission-to-discharge episode is identified by a *care spell id*, allowing multiple spells per patient. Data is split into tables that in-

[1] Appendix available here - `https://github.com/AlexBeesonWarwick/OfflineRLCAP/tree/main`

clude both time-dependent data (e.g. observations, interventions) and time-independent data (e.g. demographics). Further details and an example of the relational structure are provided in the Appendix.

As part of a cohort refinement procedure, we first excluded COVID-19 diagnoses to avoid data inconsistencies arising as a result of the global pandemic and the potential overlap with CAP. Next, to distinguish community- from hospital-acquired pneumonia, we required that both a pneumonia diagnosis and a CAP-specific antibiotic appeared within 24 hours of admission. These antibiotics were taken from the CAP section of the Trust's Adult Guidelines for Antimicrobial Prescribing [32], namely: Amoxicillin, Co-amoxiclav, Clarithromycin, Doxycycline and Levofloxacin. This refinement procedure resulted in a final cohort of 10,707 care spells for 9,147 patients. Of these, 88% had just one care spell, 9% had two, 2% had three, and fewer than 1% had four or more.

4 Methodology

4.1 Offline reinforcement learning

We start by defining a Markov Decision Process (MDP) $M =< S, A, T, R, \gamma >$ where S is the state space, A the action space, $T(s' \mid s, a)$ the environment dynamics, $R(s, a)$ the reward function and $\gamma \in [0, 1]$ the discount factor [29]. An autonomous agent interacts with this MDP by following a state-dependent policy $\pi(s)$, with the objective of discovering an optimal policy $\pi^*(s)$ that maximises the expected discounted sum of rewards, $\mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]$.

In discrete action spaces this objective can be achieved through Q-learning. The Q-function $Q^\pi(s, a)$ defines the value of taking action a in state s following policy π thereafter and optimal Q-values can be obtained by repeated application of the Bellman optimality equation:

$$Q^*(s, a) = r(s, a) + \gamma \, \mathbb{E}_{s' \sim T} \left[\max_{a'} Q^*(s', a') \right] .$$

The optimal policy can then be extracted by taking the action that maximises the optimal Q-value at each state, i.e. $\pi(s) = \arg\max_a Q(s, a)$. Alternatively, actions can be chosen stochastically based on Q-values, for example using a softmax. The probability of action a at state s is denoted $\pi(a \mid s)$.

The scale and complexity of real-world tasks often necessitates the use of function approximation methods. To this effect, Q-functions are paramaterised with learnable parameters θ, which are updated to minimise the following loss:

$$L(\theta) = \frac{1}{|B|} \sum_{(s,a,r,s') \sim \mathcal{B}} \left(Q_\theta(s, a) - y(r, s') \right)^2 ,$$

where $y(r, s') = r + \gamma \max_{a'} Q_\theta(s', a')$ is the target value and $\mathcal{B}$ is a replay buffer of transitions which is uniformly sampled during training [26].

In offline scenarios, an agent can no longer interact with the environment itself and must instead learn from a pre-existing set of interactions $\mathcal{B}$ collected

from some (potentially unknown) behaviour policy or set of policies π_β [17]. With environment interaction prohibited, errors in Q-values estimates are free to compound and propagate during training. Specifically, Q-value estimates for out-of-distribution (OOD) actions (i.e. those not present in $\mathcal{B}$) are prone to overestimation bias as a consequence of the maximisation carried out when determining target values [30]. The end result is spurious Q-value estimates and by extension highly sub-optimal policies. In order to mitigate the deterimental effects of overestimation bias, Q-values must be regularised by staying "close" to actions within the existing set.

In conservative Q-learning (CQL) [16] Q-value estimates are regularised by "pushing down" on estimates for out-of-distribution actions and "pushing up" on estimates for in-distribution actions. Such an adjustment is effectively "gap-expanding" in Q-values between in-distribution and out-of-distribution actions, leading the agent to favour actions more like those in the data when updating the Q-function.

The manner in which Q-values are pushed down can be varied. For each state we can for example use the average Q-value for the all actions, but in practice a more effective approach is to use a type of softmax, leading to the following modified loss:

$$L(\theta) = \frac{1}{|B|} \sum_{(s,a,r,s') \sim \mathcal{B}} \left(Q_\theta(s,a) - y(r,s')\right)^2 + \beta \sum_{(s,a) \sim \mathcal{B}} \left[\log \sum_{a_i \in A} \exp(Q_\theta(s,a_i)) - Q_\theta(s,a) \right] ,$$

where β is a hyperparameter that controls the level of conservatism.

4.2 Off-policy evaluation

The most accurate and straightforward way to evaluate a policy is to deploy it in the environment and record its return $G(t) = \sum_{t=0}^{T} \gamma^t r_t$. However, in settings involving DTRs rolling out policies without prior assurances on their quality would be considered unacceptable, not least because of the concerns regarding patient safety. Instead, the policy must also be evaluated, not just learnt, in the offline setting. This is the motivation behind off-policy evaluation (OPE), which seeks to estimate the value of one policy using transitions collected from another. In the context of offline RL, this equates to estimating the value of an offline trained policy π_e using transitions from the data set $\mathcal{B}$, originally collected by behaviour policy π_β

Off-policy evaluation is an important and active research area in its own right and there have been many approaches put forward seeking to provide accurate estimators with desirable properties [34]. For the purpose of this study we make use of fitted Q-evaluation (FQE) [18], in which a Q-function is trained using the Bellman expectation equation and the property $V(s) = \sum_{a \in A} \pi(a \mid s) Q(s,a)$ used as the basis for a policy value estimate.

Specifically, a parameterised Q-function $Q_\phi(s, a)$ is learnt using the following loss:

$$L(\phi) = \frac{1}{|B|} \sum_{(s,a,r,s') \sim \mathcal{B}} \Big(Q_\phi(s, a) - r - \gamma V(s') \Big)^2 ,$$

where $V(s') = \sum_{a' \in A} \pi(a' \mid s') Q_\phi(s', a')$.

The estimate of the policy value is then:

$$\hat{V}(\pi_e) = \frac{1}{N} \sum_{i=1}^{N} \sum_{a \in A} \pi_e(a \mid s_0^i) Q_\phi(s_0^i, a) \ ,$$

where s_0 is the initial state.

Although this is a biased estimator of $V(\pi_e)$, in general it has low variance since it only requires a one-step estimate during training. This contrasts to other OPE methods such as importance sampling and its per-decision/weighted variants [29] which suffer from high variance due to importance weights becoming either vanishingly small or exponentially large in cases where the evaluation and behaviour policy are significantly different and/or trajectories are long. Furthermore, using FQE allows us to establish a relationship between Q-values and mortality rate, which we can utilise as part of an evaluation protocol.

In order to apply RL methods to the PIONEER data, we must define states, actions and rewards in an MDP framework. This necessitates a range of data preprocessing tasks, including the aforementioned cohort refinement, as well as variable selection and the derivation of clinically relevant features. We also need to align each patient record with appropriate action labels (e.g. administered medications) and design a reward function that captures 30-day mortality. Below, we provide an abridged explanation of these steps, and direct the reader to the Appendix for a full detailed description.

4.3 Variable selection and derivation

A crucial first step was to identify variables suitable for modelling patient states, actions and rewards. To this effect, we conducted a table-by-table review guided by four criteria:

1. **Clinical relevance:** Variables had to reflect a patient's health (e.g. vital signs, laboratory results) or administered treatments (e.g. antibiotics) relevant to managing CAP.
2. **Usability:** We required data amenable to deep learning, excluding unstructured free text without consistent numerical or categorical encoding.
3. **Coverage:** Items needed sufficient coverage across the final cohort to enable robust learning. Variables recorded in only a small fraction of care spells were generally discarded.
4. **Imputability:** For potentially informative items with lower coverage, we assessed whether missing data could be addressed with nominal values, derived values or other proxies.

Applying these criteria yielded 41 variables spanning observations, laboratory tests, radiography, comorbidities, demographics and drug administration. Where helpful, we derived additional features (e.g. mapping Troponin or D-dimer to "normal/high/not-recorded"). This process established a broad foundation for representing patient health and relevant clinical parameters.

4.4 Constructing the MDP

Once the pertinent variables were selected, we defined a MDP for offline RL. Unlike simulated settings, *real-world clinical data do not arrive in neatly spaced intervals*, nor does a reward event happen immediately after each clinical decision. Hence, reconciling real-world complexity with the structure of MDPs required several design decisions.

States. For each decision point, we aggregated the chosen variables into a single state representation of the patient. However, since hospitals record different measurements at different times, many entries appeared "missing" when pivoted and joined into a uniform table. To handle this, we adopted a *sample-and-hold* strategy [14], carrying the most recent observed value forward until a new measurement arrived. Where no prior value existed (e.g. early in a care spell) or the data were inherently sparse, we used nominal or median-based imputation.

Actions. We focused on antibiotic administration, concatenating a drug with its route (enteral or parenteral) into a single *drug-route* action. In reality, patients could receive multiple antibiotics simultaneously or switch among them at varying frequencies, which the data did not always capture explicitly. As a compromise, we either (a) limited actions to single antibiotic-route pairs in variable-length windows or (b) allowed up to two antibiotic–route pairs in fixed windows (Section 4.5). If no antibiotics were administered for an extended period (e.g. 36 hours), or no antibiotic was given in the first interval, we labeled the action "no treatment." These simplifications approximate real care patterns, although they inevitably lose detail about multi-drug regimens.

Reward and terminal state. We used a sparse reward keyed to 30-day mortality, a standard outcome metric for pneumonia [37]. If the patient was alive at 30 days, we assigned a reward of $+1$, if not, -1. The terminal state was the patient's final recorded state on or before day 30.

4.5 Time-step definitions

Discretizing the data into steps for an MDP presents further challenges because antibiotic schedules and patient evolution rarely conform to uniform intervals. In light of this, we explored two main approaches:

Fixed time step We partitioned each patient's timeline into windows of 8, 12, or 24 hours. Within each window:

1. **State:** Continuous variables were aggregated by median, categorical variables by mode, and missing values imputed via sample-and-hold or nominal/median substitution.
2. **Action:** Up to two *drug-route* pairs were concatenated. If no antibiotics were administered for >36 hours or none appeared in the first window, the action was "no treatment."

This approach maintains a regular MDP structure but may misrepresent real antibiotic timing and overlooks finer nuances like overlapping regimens.

Variable time step We subdivided each patient's timeline at actual antibiotic events, creating a new window whenever a drug was administered. If 36 hours elapsed without any antibiotic, a "no treatment" window was inserted. States reflected the most recent measurement values at each event. This strategy captures real scheduling more accurately yet violates the fixed-step MDP assumption and does not easily allow antibiotic combinations in a single step.

4.6 Processed data sets

Having defined states, actions and rewards, we applied the above imputation and time-step procedures to yield four final data sets: **Fixed 8hr**, **Fixed 12hr**, **Fixed 24hr** and **Variable** windows (based on antibiotic events). They differ primarily in how time is discretized and how multi-antibiotic use is handled. In each, the terminal state is day 30 or earlier if the patient died or was discharged, and the reward is set by 30-day mortality. Table 1 outlines key cohort characteristics, with lists of ethnicity, comorbordities and antibiotic-routes provided in the Appendix. While these transformations approximate real patient trajectories for offline RL, they inevitably sacrifice some detail due to irregular sampling, missing data and partially documented regimens.

5 Experimental results

5.1 Set-up and implementation

We evaluate four offline policies for CAP management: a **random policy** that selects actions uniformly from the entire action space; a **behaviour policy** that replicates the empirical distribution of observed actions; a **DQN** policy trained via standard Q-learning; a **CQL** policy trained using conservative Q-learning as outlined Section 4.1. We use FQE as detailed in Section 4.2 to estimate each policy's expected return using the processed dataset.

For DQN, CQL and FQE each Q-function comprised a 2-layer MLP with ReLU activation functions and 256 nodes, taking as input a state and outputting a Q-value for each action. We set the discount factor $\gamma = 1$ and used the Huber

Table 1. Overview of the patient cohort. FiO_2 (OT): derived values for patients receiving oxygen therapy. D-dimer, Troponin-I, Troponin-T: N=Normal, H=High, NR=Not Recorded. See Table O in the Appendix for additional details on ethnicity groups, comorbidity definitions and antibiotics (ABX).

Category	Feature	Mean (SD)	Feature	Mean (SD)
Demographics	Age	73.9 (15.7)	Male (N, %)	5390 (50.3)
	Non-survivors (N, %)	1727 (16.1)	Ethnicity	7 types
	Comorbidity	22 types		
Observations	AVPU scale	3.96 (0.25)	Respiratory rate	18.6 (3.3)
	Diastolic BP	70.7 (12.9)	Systolic BP	125.1 (22.6)
	Heart rate	85.2 (17.4)	Temperature	36.4 (0.7)
	NEWS2	3.17 (2.6)	O_2 sats (%)	94.8 (3.3)
Lab Analysis	Base excess	0.15 (5.1)	Blood K	4.21 (0.74)
	Blood Na	138 (6.1)	pCO_2	6.01 (1.82)
	pO_2	8.04 (5.46)	Basophils	0.05 (0.06)
	Eosinophils	0.17 (0.38)	Haematocrit	0.34 (0.07)
	Haemoglobin	113 (22.5)	Lymphocytes	1.72 (8.9)
	Mean cell Hb	29.4 (3.0)	Mean cell volume	89.6 (7.7)
	Monocytes	0.87 (1.44)	Neutrophils	9.01 (7.9)
	Platelets	290 (143)	Red cell count	3.83 (0.75)
	White cell count	11.4 (11.2)	Albumin	29.1 (6.8)
	Alkaline phosphatase	134 (144)	Calcium	2.2 (0.2)
	Total protein	62.0 (8.8)	C-reactive protein	101 (91.7)
	Alanine transferase	35.7 (97.3)	Urea	9.12 (6.94)
Imaging	Chest X-ray (N, %)	8334 (77.9)		
Derived	FiO_2 (OT)	0.28 (0.23)	D-dimer	N, H, NR
	Trop-T	N, H, NR	Trop-I	N, H, NR
Treatment	ABX+Route	9 types		

loss to update network parameters. We used a dual critic approach, taking the mean across two Q-value estimates for target Q-values. We made use of separate target networks for estimating target Q-values, updating these networks according to Polyak averaging with update rate 0.005. Networks were trained via stochastic gradient descent using the Adam optimizer with learning rate $3e^{-4}$ and batch size 256. For CQL we trained networks for 500k gradient steps and for FQE we trained networks for 1M gradient steps. For the conservative hyperparameter in CQL we used $\beta \in \{1, 2, 5\}$.

We split the processed data into training and validation sets at a ratio of 80/20. Data was split based on trajectories as opposed to individual transitions as the goal is to generalise treatment policies to new care spells. For each of the processed data sets, we created five different training and validation splits and we report policy value estimates using means $\pm$ one standard error. Numerical state features were normalised to compensate for differences in scales between measurement types.

Table 2. Policy value estimates (mean ± standard error) computed by fitted Q-evaluation across 5 training/validation splits. The β in CQL is a hyperparameter controlling the level of conservatism (higher values equate to more conservatism).

Data set	Behaviour	Random	DQN	CQL		
				$\beta = 1$	$\beta = 2$	$\beta = 5$
Fixed 8hr	0.75 ± 0.00	0.42 ± 0.01	0.12 ± 0.21	0.76 ± 0.00	0.78 ± 0.00	0.79 ± 0.00
Fixed 12hr	0.76 ± 0.00	0.22 ± 0.02	0.17 ± 0.19	0.79 ± 0.00	0.79 ± 0.00	0.78 ± 0.00
Fixed 24hr	0.73 ± 0.00	0.21 ± 0.01	0.59 ± 0.17	0.76 ± 0.00	0.75 ± 0.00	0.76 ± 0.00
Variable	0.76 ± 0.01	0.41 ± 0.01	0.72 ± 0.04	0.79 ± 0.00	0.80 ± 0.00	0.81 ± 0.00

5.2 Policy evaluation and mortality prediction

In Table 2 we report the policy value estimates for each of our chosen policies. Overall, we see CQL marginally outperforms the behaviour policy in most scenarios, whereas DQN often matches or falls below the random policy, especially for shorter fixed length windows (i.e. 8hr and 12hr windows). This underscores the importance of conservative regularization in alleviating overestimation when data coverage is incomplete.

In order to interpret policy values estimates in the context of mortality, we first have to establish a relationship between the two entities and then use this relationship to make predictions. To establish a relationship, similar to previous work [39] we create a set of bins for behaviour policy value estimates and within each bin calculate the mortality rate (number of deaths divided by number of care spells). In Figure 1 we visualise the results, which indicate an inverse relationship between value estimates and mortality rates for each data set. To quantify this relationship, we fit a logistic regression model and use it to predict the mortality rate for each of our policies. We superimpose the models onto Figure 1 and summarise predictions in Table 3 alongside the results of statistical significance testing between CQL and the behaviour policy using a two-sided paired t-test.

Table 3. Mortality rates (mean ± standard error) across 5 data splits. For CQL, the best-performing β value is shown. P-values obtained from two-sided paired t-test.

Data set	Behaviour	Random	DQN	CQL	p-value
Fixed 8hr	16.4 ± 0.3	21.2 ± 0.4	28.6 ± 4.0	15.6 ± 0.4	0.009
Fixed 12hr	16.4 ± 0.3	25.2 ± 0.6	28.2 ± 3.8	15.6 ± 0.3	<0.001
Fixed 24hr	16.5 ± 0.3	24.4 ± 0.6	19.1 ± 3.0	15.8 ± 0.4	0.004
Variable	16.6 ± 0.3	21.7 ± 0.8	17.3 ± 0.6	15.8 ± 0.4	0.003

Although the resulting estimated mortality reductions are statistically significant at the 5% significance level, they are relatively modest, consistently favouring CQL over the observed (behaviour) policy. A breakdown by patient age in Table 4 further suggests that older patients (those at higher risk) might benefit more, with slightly larger reductions in predicted mortality. However, it

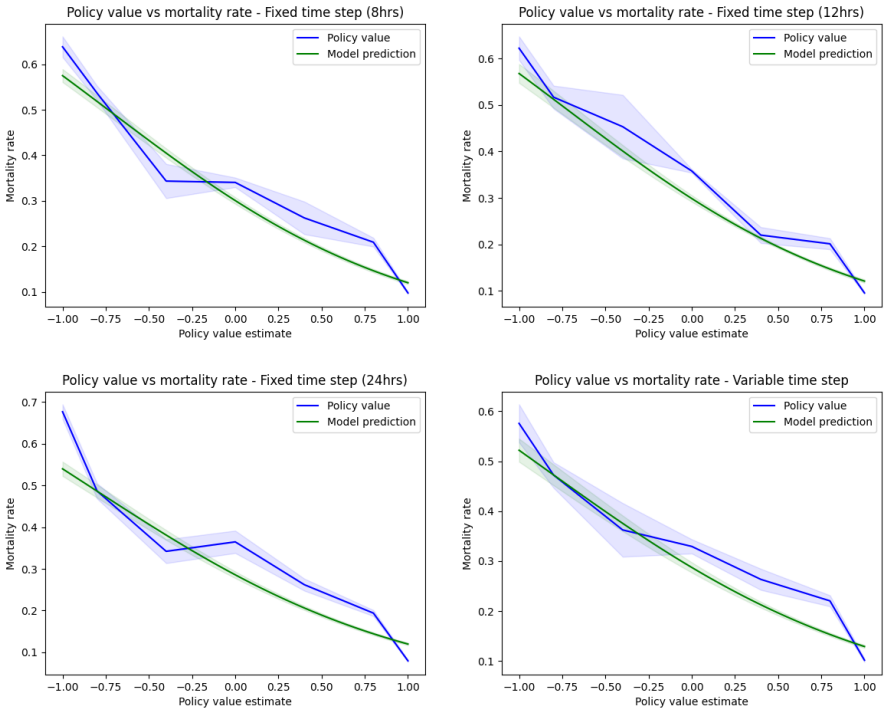

Fig. 1. Policy value estimate vs. mortality rate. For each data set there appears to be an inverse relationship between policy value estimates and mortality rate (blue). For predictive purposes we model this relationship using logistic regression (green).

should be pointed out these reductions are not always statistically significant at the 5% level when accounting for multiple testing using a Bonferroni correction in which the significance level is (conservatively) adjusted to 1%, reflecting the intrinsic complexities in real-world data and the offline setting's constraints.

6 Discussion and conclusions

In this work we have conducted a feasibility study looking at whether offline RL can be used to reduce 30-mortality for patients with CAP. We have undertaken an extensive set of data pre-processing procedures to convert source data into states, actions and rewards then used this processed data to train and evaluate policies offline. Our initial results indicate that agents trained offline using CQL are able to learn treatment policies that marginally improve overall mortality compared to the behaviour policy, as evaluated using FQE and a logistic regression model. Agents trained using DQN with no offline modifications fail to improve over the behaviour policy and in some cases perform no better than a random policy.

Table 4. Mortality rates (mean ± standard error) across 5 splits. For each data set, we list Behaviour vs. CQL (best β), the absolute reduction and associated p-values.

Data set	Age group	Behaviour	CQL	Reduction	p-value
Fixed 8hr	<65	13.5 ± 0.4	13.3 ± 0.5	0.2	0.362
	65–79	15.8 ± 0.4	15.1 ± 0.5	0.7	0.006
	80–84	17.2 ± 0.3	16.2 ± 0.3	1.0	0.007
	85–89	18.8 ± 0.4	17.4 ± 0.5	1.4	0.011
	90+	19.1 ± 0.3	17.9 ± 0.5	1.2	0.068
Fixed 12hr	<65	13.9 ± 0.4	13.4 ± 0.4	0.5	0.005
	65–79	15.7 ± 0.4	15.1 ± 0.3	0.6	0.004
	80–84	17.3 ± 0.1	16.1 ± 0.2	1.2	0.011
	85–89	18.4 ± 0.4	17.3 ± 0.3	1.1	0.007
	90+	18.7 ± 0.5	17.8 ± 0.1	0.9	0.170
Fixed 24hr	<65	13.6 ± 0.3	13.5 ± 0.3	0.1	0.413
	65–79	15.5 ± 0.4	15.4 ± 0.4	0.1	0.266
	80–84	17.2 ± 0.2	16.9 ± 0.3	0.3	0.226
	85–89	18.9 ± 0.5	17.3 ± 0.4	1.6	0.002
	90+	19.4 ± 0.3	17.6 ± 0.4	1.8	0.004
Variable	<65	14.2 ± 0.3	14.0 ± 0.4	0.2	0.300
	65–79	16.1 ± 0.3	15.4 ± 0.4	0.7	0.005
	80–84	17.4 ± 0.3	16.7 ± 0.3	0.7	0.034
	85–89	18.4 ± 0.4	17.1 ± 0.4	1.3	0.018
	90+	19.1 ± 0.3	17.7 ± 0.4	1.4	0.017

Future work. Our study has generated several lines of enquiry for future work, one of which is to investigate alternative approaches for aggregating state variables. In particular, including dispersion metrics (e.g. interquartile ranges, variances) or explicitly encoding temporal structure (e.g. concatenating previous states or tracking state changes across timesteps) might yield a more expressive representation of a patient's evolving health status. Additionally, architectures beyond a simple MLP, such as LSTMs [10] or transformers [35], could better capture the time-series nature of clinical data.

We also see scope for more nuanced action spaces. Our current setup merges an antibiotic with its route of administration, but escalation/de-escalation steps, dose variations, oxygen therapy, or ICU admission could be treated as distinct actions, possibly leading to more complete policies. Another direction lies in rethinking the reward structure. We used a sparse reward tied to 30-day mortality, but additional outcomes (e.g. hospital length-of-stay, ICU admissions) or intermediate measures (e.g. improving FiO_2) could create a denser, more informative signal for the agent.

Data preprocessing challenges. A key aspect of this work was the substantial and iterative data-processing effort required to align routine hospital records with the assumptions of RL. Although Section 4 points to an ordered sequence of steps, in practice we repeatedly revisited earlier decisions. For example, our initial patient selection excluded only COVID-19 cases, but this required mod-

ification upon discovering diagnosis codes did not clearly distinguish between CAP and HAP. Similarly, deriving variables like FiO_2 required cross-referencing multiple tables (Observations, Ventilation) that occasionally contradicted each other. Each iteration demanded thorough testing to ensure consistency and avoid introducing new errors.

Time-step definitions and antibiotic regimens. Defining the temporal granularity for state-action pairs proved especially challenging. We explored two main strategies: fixed-length windows (8, 12, or 24 hours) to preserve an approximate Markov property, and variable-length windows that aligned with recorded antibiotic administrations to reflect real-world treatment intervals. Each approach had drawbacks. Fixed windows often failed to capture the actual frequency or timing of drug administration, while variable windows lost regular time steps. Neither method could fully capture multi-drug regimens because the data only recorded drug administration at the singular level. Hence, our policy comparisons inevitably compare approximations rather than an exact record of patient care.

Imputation strategies. Although a sample-and-hold approach populated most state variables when new observations were temporarily absent, many transitions still lacked any recent value. We chose a median-based imputation in these scenarios, on the premise that clinicians might initially assume typical or population-level values before test results are available. Alternative approaches (e.g. regression or k-nearest neighbours) might yield more tailored estimates but would greatly increase the pipeline's complexity, requiring separate predictive models for each of the numerous clinical variables. In addition, sample-and-hold itself is subject to limitations since it assumes values remain constant in between time points, which may not necessarily be the case.

Comorbidities and cause of death. Including some comorbidity flags (i.e. past diagnoses) in the state space enhanced our representation of patient risk. Yet it remained difficult to fully represent the broader clinical ramifications of these comorbidities, such as additional medications or whether a patient's death was unrelated to CAP. Designing a more expansive action space, or refining the reward to account for other causes of death, would require careful modelling and extensive data given the multifactorial nature of hospital mortality.

Policy evaluation and mortality prediction. To evaluate our learned policies, we relied on FQE to estimate Q-values offline. We also modelled the relationship between Q-values and 30-day mortality using logistic regression, observing an inverse correlation that suggests our Q-function captures meaningful clinical signals. However, a more comprehensive validation could involve qualitative assessments of suggested actions by domain experts or prospective trials in controlled settings. The former would be a vital step in gaining assurances to facilitate the latter, assessing not just the safety of proposed actions but also the plausibility, both as a single treatment option and as part of a regimen. We did not have the

resources available to conduct a formal qualitative assessment of actions for this particularly study, however it is something we would seek to include as part of future work.

Implications and next steps. Despite the inherent complexity and partial success of our approach, this study highlights the promise of offline RL for the management of CAP. Building upon these findings might involve more advanced RL formulations, such as semi-Markov decision processes (SMDPs), partially observable Markov decision processes (POMDPs) and/or hierarchical reinforcement learning (HRL). In a SMDP [11] the time between actions can vary, which may better align with how treatment are administered in cases such as CAP. In a POMDP [38] the agent cannot directly observe the underlying state, which may better reflect the observational nature of healthcare data. In HRL [12] tasks are split into high-level and low-level sub-tasks, which are governed by different policies that operate together to achieve an overall objective. In a healthcare setting this could equate to having a high-level policy that determines when to treat a patient and a low-level policy that selects the specific treatment. Efforts could also be directed toward curating data sets specifically with RL needs in mind to avoid many of the pitfalls we encountered.

Concluding remarks. Overall, our feasibility study provides initial evidence that offline RL can yield marginally better outcomes than historical practice for CAP management, but it also illustrates the formidable challenges of adapting real-world healthcare data to the structure demanded by RL algorithms. Addressing these challenges—through richer state spaces, refined action definitions, improved reward signals, and more robust ways to handle missing data—will be essential for unlocking the full potential of RL in clinical settings. By continuing to refine data acquisition procedures and incorporate innovative RL techniques, we can move toward safer, more effective data-driven decision support for pneumonia and other complex conditions.

Acknowledgments. This work was supported by PIONEER, the Health Data Research Hub in Acute Care, which is affiliated with Health Data Research UK. PIONEER: Data curation and licensed access for this study through PIONEER has been approved by the East Midlands (Derby) REC (20/EM/0158) and is supported by the Confidentiality Advisory Group (Reference 20/CAG/0084). This study is also granted ethical approval by the University of Warwick's Biomedical and Scientific Research Ethics Committee (BSREC 82/21-22). AB acknowledges support from University of Warwick and University Hospitals Birmingham NHS Foundation Trust. GM acknowledges support from a UKRI AI Turing Acceleration Fellowship (EPSRC EP/V024868/1). The authors also acknowledge the support and guidance of Prof. Gavin Perkins from the University of Warwick.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. An, G., Moon, S., Kim, J.H., Song, H.O.: Uncertainty-based offline reinforcement learning with diversified q-ensemble. Advances in neural information processing systems **34**, 7436–7447 (2021)
2. Beeson, A., Montana, G.: Balancing policy constraint and ensemble size in uncertainty-based offline reinforcement learning. Machine Learning **113**(1), 443–488 (2024)
3. Bigirumurame, T., Uwimpuhwe, G., Wason, J.: Sequential multiple assignment randomized trial studies should report all key components: a systematic review. Journal of clinical epidemiology **142**, 152–160 (2022)
4. Chakraborty, B., Murphy, S.A.: Dynamic treatment regimes. Annual review of statistics and its application **1**(1), 447–464 (2014)
5. Chalmers, J., Campling, J., Ellsbury, G., Hawkey, P.M., Madhava, H., Slack, M.: Community-acquired pneumonia in the united kingdom: a call to action. Pneumonia **9**, 1–6 (2017)
6. Cheng, L.F., Prasad, N., Engelhardt, B.E.: An optimal policy for patient laboratory tests in intensive care units. In: BIOCOMPUTING 2019: Proceedings of the Pacific Symposium. pp. 320–331. World Scientific (2018)
7. Fujimoto, S., Conti, E., Ghavamzadeh, M., Pineau, J.: Benchmarking batch deep reinforcement learning algorithms. arXiv preprint arXiv:1910.01708 (2019)
8. Fujimoto, S., Gu, S.S.: A minimalist approach to offline reinforcement learning. Advances in Neural Information Processing Systems **34**, 20132–20145 (2021)
9. Fujimoto, S., Meger, D., Precup, D.: Off-policy deep reinforcement learning without exploration. In: International Conference on Machine Learning. pp. 2052–2062. PMLR (2019)
10. Hochreiter, S.: Long short-term memory. Neural Computation MIT-Press (1997)
11. Hu, Q., Yue, W.: Markov decision processes with their applications, vol. 14. Springer Science & Business Media (2007)
12. Hutsebaut-Buysse, M., Mets, K., Latré, S.: Hierarchical reinforcement learning: A survey and open research challenges. Machine Learning and Knowledge Extraction **4**(1), 172–221 (2022)
13. Kaplan, W., Wirtz, V., Mantel-Teeuwisse, A., Stolk, P., Duthey, Beatrice nd Laing, R.: Priority medicines for europe and the world 2013 update. Tech. rep., World Health Organisation (2013)
14. Komorowski, M., Celi, L.A., Badawi, O., Gordon, A.C., Faisal, A.A.: The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. Nature medicine **24**(11), 1716–1720 (2018)
15. Kostrikov, I., Nair, A., Levine, S.: Offline reinforcement learning with implicit q-learning. In: International Conference on Learning Representations (2021)
16. Kumar, A., Zhou, A., Tucker, G., Levine, S.: Conservative q-learning for offline reinforcement learning. Advances in Neural Information Processing Systems **33**, 1179–1191 (2020)
17. Lange, S., Gabel, T., Riedmiller, M.: Batch reinforcement learning. Springer pp. 45–73 (2012)
18. Le, H., Voloshin, C., Yue, Y.: Batch policy learning under constraints. In: International Conference on Machine Learning. pp. 3703–3712. PMLR (2019)
19. Levine, S., Kumar, A., Tucker, G., Fu, J.: Offline reinforcement learning: Tutorial, review, and perspectives on open problems. arXiv preprint arXiv:2005.01643 (2020)

20. Lim, W.S., Van der Eerden, M.M., Laing, R., Boersma, W.G., Karalus, N., Town, G.I., Lewis, S., Macfarlane, J.: Defining community acquired pneumonia severity on presentation to hospital: an international derivation and validation study. Thorax **58**(5), 377–382 (2003)
21. Liu, J., Xie, Y., Shu, X., Chen, Y., Sun, Y., Zhong, K., Liang, H., Li, Y., Yang, C., Han, Y., et al.: Value function assessment to different rl algorithms for heparin treatment policy of patients with sepsis in icu. Artificial Intelligence in Medicine **147**, 102726 (2024)
22. Liu, S., See, K.C., Ngiam, K.Y., Celi, L.A., Sun, X., Feng, M.: Reinforcement learning for clinical decision support in critical care: comprehensive review. Journal of medical Internet research **22**(7), e18477 (2020)
23. Lopez-Martinez, D., Eschenfeldt, P., Ostvar, S., Ingram, M., Hur, C., Picard, R.: Deep reinforcement learning for optimal critical care pain management with morphine using dueling double-deep q networks. In: 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC). pp. 3960–3963. IEEE (2019)
24. Luo, J., Dong, P., Wu, J., Kumar, A., Geng, X., Levine, S.: Action-quantized offline reinforcement learning for robotic skill learning. In: 7th Annual Conference on Robot Learning (2023)
25. Millett, E.R., Quint, J.K., Smeeth, L., Daniel, R.M., Thomas, S.L.: Incidence of community-acquired lower respiratory tract infections and pneumonia among older adults in the united kingdom: a population-based study. PloS one **8**(9), e75131 (2013)
26. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602 (2013)
27. Murphy, S.A.: An experimental design for the development of adaptive treatment strategies. Statistics in medicine **24**(10), 1455–1481 (2005)
28. Prasad, N., Cheng, L.F., Chivers, C., Draugelis, M., Engelhardt, B.E.: A reinforcement learning approach to weaning of mechanical ventilation in intensive care units. arXiv preprint arXiv:1704.06300 (2017)
29. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
30. Thrun, S., Schwartz, A.: Issues in using function approximation for reinforcement learning. In: Proceedings of the Fourth Connectionist Models Summer School. vol. 255, p. 263. Hillsdale, NJ (1993)
31. Torres, A., Chalmers, J.D., Dela Cruz, C.S., Dominedò, C., Kollef, M., Martin-Loeches, I., Niederman, M., Wunderink, R.G.: Challenges in severe community-acquired pneumonia: a point-of-view review. Intensive care medicine **45**, 159–171 (2019)
32. Trust, U.H.B.N.F.: Adult guidelines for antimicrobial prescribing. `https://www.uhb.nhs.uk/Downloads/pdf/controlled-documents/AntimicrobialPrescribingGuidelines.pdf`, 2023-09-01
33. Tseng, H.H., Luo, Y., Cui, S., Chien, J.T., Ten Haken, R.K., Naqa, I.E.: Deep reinforcement learning for automated radiation adaptation in lung cancer. Medical physics **44**(12), 6690–6705 (2017)
34. Uehara, M., Shi, C., Kallus, N.: A review of off-policy evaluation in reinforcement learning. arXiv preprint arXiv:2212.06355 (2022)
35. Vaswani, A.: Attention is all you need. Advances in Neural Information Processing Systems (2017)

36. Weng, W.H., Gao, M., He, Z., Yan, S., Szolovits, P.: Representation and reinforcement learning for personalized glycemic control in septic patients. arXiv preprint arXiv:1712.00654 (2017)
37. Wiemken, T.L., Furmanek, S.P., Mattingly, W.A., Guinn, B.E., Cavallazzi, R., Fernandez-Botran, R., Wolf, L.A., English, C.L., Ramirez, J.A.: Predicting 30-day mortality in hospitalized patients with community-acquired pneumonia using statistical and machine learning approaches. The University of Louisville Journal of Respiratory Infections **1**(3), 10 (2017)
38. Wiering, M.A., Van Otterlo, M.: Reinforcement learning. Adaptation, learning, and optimization **12**(3), 729 (2012)
39. Wu, X., Li, R., He, Z., Yu, T., Cheng, C.: A value-based deep reinforcement learning model with human expertise in optimal treatment of sepsis. NPJ Digital Medicine **6**(1), 15 (2023)
40. Yang, C.Y., Shiranthika, C., Wang, C.Y., Chen, K.W., Sumathipala, S.: Reinforcement learning strategies in cancer chemotherapy treatments: A review. Computer Methods and Programs in Biomedicine **229**, 107280 (2023)
41. Yu, C., Liu, J., Nemati, S., Yin, G.: Reinforcement learning in healthcare: A survey. ACM Computing Surveys (CSUR) **55**(1), 1–36 (2021)