Dissertation

Machine Learning in Intensive Care Medicine

submitted in satisfaction of the requirements for the degree
Doctor of Science in Engineering Sciences
of the TU Wien, Faculty of Mathematics and Geoinformation

Maschinelles Lernen für den Einsatz in der Intensivmedizin

ausgeführt zum Zwecke der Erlangung des akademischen Grads
Doktor der technischen Wissenschaften
eingereicht an der TU Wien, Fakultät für Mathematik und Geoinformation

Dipl.-Ing. Lorenz Kapral

Matr.Nr.: 01326287

Betreuung:	Prof. Dipl.-Ing. Dr. Clemens Heitzinger
Betreuung:	Prof. Dr. Oliver Kimberger
Begutachtung:	Prof. Dr. Leif Saager
Begutachtung:	Prof. Dipl.-Ing. Dr. Christian Ringhofer

Wien, im März 2025 _____

Kapitel 1

Kurzfassung

Diese Dissertation untersucht fortschrittliche Ansätze der künstlichen Intelligenz zur Verbesserung der klinischen Entscheidungsfindung in der Intensivmedizin anhand dreier komplementärer Studien. In der ersten Publikation wurde ein Reinforcement-Learning-(RL)-Algorithmus entwickelt, um die Kortikosteroidtherapie bei septischen Patienten zu optimieren. Anhand von Daten aus 3.051 Intensivstationsaufenthalten im AmsterdamUMCdb wurden Patienten gemäß der Konsensdefinition von 2016 identifiziert. Ein auf einem Actor-Critic-Framework basierendes RL-Modell, das die Intensivstationsmortalität als Belohnungssignal nutzte, wurde mit zeitlich aufbereiteten Daten zu 277 klinischen Parametern trainiert. Off-Policy-Evaluierungen zeigten, dass die Behandlungspolitik des RL-Agenten – die durch eine restriktivere Kortikosteroidverordnung gekennzeichnet war (in 62 % der Patientenzustände ein Zurückhalten versus 52 % in der klinischen Praxis) – einen höheren erwarteten Reward und eine damit verbundende geringere Intensivstationsmortalität in der Evaluierung in einem Test-Datenset erzielte.

Im der zweiten Studie wurde RL angewendet, um die individualisierte Entscheidungsunterstützung für die Nierenersatztherapie (RRT) bei kritisch kranken Patient:innen mit akutem Nierenversagen (AKI) zu verbessern. Hierzu wurden Daten aus der öffentlich zugänglichen MIMIC-IV-Datenbank sowie einem externen Datensatz der Medizinischen Universität Wien (MUW) verwendet, wobei Patient:innen mit AKI ab Stadium I einbezogen und solche mit chronischer Nierenerkrankung oder vorangegangener Nierentransplantation ausgeschlossen wurden. Durch die Extraktion von 88 medizinischer Parametern und den Einsatz eines gewichteten K-Means-Clustering-Ansatzes zur Definition des Patientenzustands wurde ein Q-Learning-basierter RL-Ansatz entwickelt. Das Modell erreichte in beiden Kohorten eine Übereinstimmung mit den klinischen Entscheidungen von bis zu 98% und übertraf diese in beiden Evaluationsmethoden. Besonders hervorzuheben ist, dass das Modell eine Patientengruppe mit höherer Erkrankungsschwere identifizierte, die von einer früheren oder intensiveren RRT profitieren könnte, was das Potenzial einer KI-gestützten Entscheidungsunterstützung zur Verbesserung der Ergebnisse unterstreicht.

Die dritte Publikation dieser Dissertation befasst sich mit der Vorhersage intraoperativen Hypotonie. Hierzu wurde ein Temporal Fusion Transformer (TFT)-Modell eingesetzt, um die arterielle Blutdruckentwicklung während der Operation bis zu 7 Minuten im Voraus anhand von niedrig aufgelösten Daten (alle 15 Sekunden) von 73.009 Patient:innen, die sich einer Allgemeinanästhesie bei nicht-kardiothorakalen Eingriffen unterzogen, zu prognostizieren. Das TFT-Modell erreichte einen mittleren absoluten Fehler von ca. 4 mmHg im internen Test und 7 mmHg im externen Test. Zudem ermöglichte die binäre Vorhersage einer Hypotonie (mittlerer arterieller Druck unter 65 mmHg) eine hervorragende Diskriminierung mit AUROC-Werten von 0,933 im internen und 0,919 im externen Testdatensatz.

Insgesamt zeigen diese Studien, dass fortschrittliche KI-Techniken – insbesondere RL und transformerbasierte Modelle – Verbesserungen in der personalisierten und präzisen Gestaltung intensivmedizinischer Therapien bewirken könnten. Die Integration robuster Datenvorverarbeitung, dynamischer Zustandsraumrepräsentationen und Off-Policy-Evaluationsmethoden bildet

die Basis für die Generalisierbarkeit dieser Modelle. Letztlich legt diese Arbeit den Grundstein für zukünftige klinische Studien und ebnet den Weg zu KI-gestützten, individualisierten Behandlungsstrategien, die das Potenzial haben, die Patientensicherheit in unterschiedlichen intensivmedizinischen Settings nachhaltig zu verbessern.

Abstract

This dissertation explores advanced artificial intelligence methodologies to enhance clinical decision-making in critical care through three complementary studies. The first publication describes the development of a reinforcement learning (RL) algorithm to optimize corticosteroid therapy in septic patients. Using data from 3,051 ICU admissions in the AmsterdamUMCdb, septic patients were identified according to the 2016 consensus definition. An actor-critic RL model, which utilized ICU mortality as a reward signal, was trained on time-series data comprising 277 clinical parameters. Off-policy evaluations showed that the RL agent's treatment policy – characterized by a more restrictive use of corticosteroids (withholding in 62% of patient states versus 52% in clinician practice) – yielded a higher expected reward and lower ICU mortality when evaluated with an independent test set.

In the second study, we applied RL to support individualized decision making for renal replacement therapy (RRT) in critically ill patients with acute kidney injury (AKI). Data from the publicly available MIMIC-IV database and an external dataset from the Medical University of Vienna (MUW) were used, focusing on patients with AKI stage I or higher. By extracting 88 features and using weighted K-means clustering to define patient states, a Q-learning based RL model was developed. The model achieved up to 98% agreement with clinician decisions and outperformed the average clinician's treatment strategy in our evaluation methods. Notably, the model identified a subset of patients with higher disease severity who could benefit from earlier or more frequent RRT, highlighting the potential of AI-driven decision support to improve outcomes.

The third publication of this dissertation addresses the challenge of predicting intraoperative hypotension. We employed a Temporal Fusion Transformer (TFT) model to forecast intraoperative arterial blood pressure trajectories up to 7 minutes in advance using low-resolution data (sampled every 15 seconds) from 73,009 patients undergoing general anesthesia for non-cardiothoracic surgery. The TFT model achieved a very low mean absolute error of approximately 4 mmHg in internal testing and 7 mmHg in external testing. Additionally, binary prediction of hypotension (mean arterial pressure below 65 mmHg) yielded excellent discrimination, with AUROC values of 0.933 and 0.919 in internal and external test sets, respectively.

Collectively, these studies demonstrate that advanced AI techniques – including RL and transformer-based models – may lead to significant improvements in the personalization and precision of critical care therapies. The integration of robust data preprocessing, dynamic state-space representations, and off-policy evaluation methods supports the generalizability of these models. Ultimately, the work lays a solid foundation for future clinical trials and paves the way for AI-driven, individualized treatment strategies that promise to improve patient outcomes in various critical care settings. ay for AI-driven, individualized treatment strategies that promise to enhance patient outcomes across diverse critical care settings.

Chapter 2

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my advisors, Prof. Clemens Heitzinger and Prof. Oliver Kimberger, for their exceptional guidance, support, and expertise over the past four years. Without their mentorship and constructive feedback, this PhD would not have been possible.

I am equally thankful to the members of my dissertation committee, Prof. Leif Saager and Prof. Christian Ringhofer, for their insightful comments and valuable suggestions, which significantly enriched the quality of this work. I extend my appreciation to Dr. Aylin Bilir and Dr. Razvan Bologheanu for their continuous support and remarkable collaboration throughout these years. I also wish to thank the Ludwig Boltzmann Institute for Digital Health and Patient Safety for creating a supportive environment that has benefited many PhD students.

I owe special thanks to Sarah Fliegel for her unwavering encouragement, understanding, and incredible support. Finally, I am profoundly grateful to my family for their limitless patience, love, and belief in me throughout this demanding journey.

Contents

Chapter 3

Introduction

3.1 Overview

Artificial Intelligence (AI) has rapidly evolved from an emerging technology to a transformative tool in modern medicine. In particular, its applications in critical care medicine have received considerable attention in recent years, as evidenced by numerous PubMed-indexed studies and reviews. These reviews highlight the potential of AI to enhance clinical decision making and risk prediction [1, 2].

In the intensive care unit (ICU), clinicians are confronted with complex, multidimensional datasets—ranging from continuous physiological monitoring and laboratory values to imaging and electronic health records (EHRs). Traditional statistical methods often struggle to capture the nonlinear interactions inherent in such data. In contrast, machine learning (ML) and deep learning approaches have shown promise in processing these data streams, providing early warning signals for adverse events such as sepsis, mortality, and organ failure [3, 2]. Recent studies have highlighted advanced methodologies, including ensemble models and foundation models (e.g., transformers), which can integrate heterogeneous ICU data for improved prediction accuracy [4].

A critical factor in the adoption of AI in clinical practice is interpretability. In high-stakes environments such as the ICU, clinicians need transparent and explainable models that they can trust [5]. Several investigations have proposed frameworks for explainable AI that ensure machine learning models not only provide accurate predictions, but also actionable insights that are aligned with clinical reasoning [6].

The challenges of generalizability and external validation have also been extensively discussed in recent literature. The majority of AI models in intensive care have been developed using data from single centers or limited cohorts, which raises concerns about their applicability across diverse clinical settings [7, 8]. Efforts to standardize reporting and encourage multicenter validation are crucial for translating these innovations into routine practice.

Ethical and regulatory considerations further complicate the integration of AI in critical care. Issues related to bias, accountability, and patient safety have been rigorously debated, leading to proposals for robust ethical frameworks and regulatory pathways to ensure responsible AI deployment [9]. Moreover, future directions for research emphasize the need for transparency, replicability, and clinical integration to truly benefit patient care [10].

3.2 Introduction

The practice of critical care medicine is undergoing a fundamental transformation, driven by advances in artificial intelligence. Clinicians face time-sensitive decisions amidst dynamic patient conditions, heterogeneous disease trajectories, and incomplete physiological data in environments such as the ICU and operating room. Conditions such as sepsis, intraoperative hypotension, and AKI-each characterized by high morbidity, mortality, and therapeutic uncertainty-exemplify these challenges. Traditional protocols, often based on population-level evidence, struggle to address

the inherent complexity of individual patients. This dissertation advances AI-driven methods to bridge this gap, using reinforcement learning (RL) and transformer-based architectures to optimize personalized treatment strategies in three critical areas: corticosteroid therapy for sepsis, initiation of RRT for AKI, and proactive management of intraoperative hypotension-

Sepsis, defined as life-threatening organ dysfunction caused by a dysregulated host response to infection [11], remains a leading cause of global mortality despite decades of research. Corticosteroids, first proposed as adjunctive therapy in the 1950s [12], continue to provoke debate due to conflicting trial outcomes [13, 14]. This ambiguity reflects the syndrome's pathophysiological heterogeneity, where static treatment protocols fail to account for evolving patient states. Similarly, intraoperative hypotension (mean arterial pressure [MAP] <65 mmHg) is strongly associated with postoperative myocardial injury, renal impairment, and delirium [15, 16], yet reactive management remains standard due to the lack of reliable forecasting tools. AKI, affecting up to 50% of ICU patients [17], presents a parallel dilemma: RRT initiation hinges on balancing fluid overload, metabolic disturbances, and procedural risks, yet biomarkers and clinical thresholds offer limited guidance [18].

RL and TFT offer a paradigm shift. RL, which optimizes sequential decisions by maximizing cumulative rewards through environmental interactions [19], is uniquely suited to critical care's dynamic, high-dimensional data. TFTs, combining attention mechanisms with recurrent neural networks [20], enable precise multi-horizon forecasting even with sparse, low-resolution inputs. These methodologies address core limitations of traditional approaches: protocol rigidity, delayed interventions, and one-size-fits-all thresholds.

3.2.1 Publications

This work integrates three complementary studies to advance AI-driven critical care:

1. **Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis:** An actor-critic RL model was trained on 3,051 ICU admissions from the AmsterdamUMCdb using 277 clinical parameters and ICU mortality as a reward signal. Off-policy evaluation showed that the RL policy – characterized by more restrictive corticosteroid use (withheld in 62% of states vs. 52% clinician adherence) – achieved a 14% reduction in predicted mortality compared to standard clinical practice. This demonstrates the value of adaptive dosing strategies that respond to individual patient trajectories (see chapter 5):
   Bologheanu R, **Kapral L**, Laxar D, Maleczek M, Dibiasi C, Zeiner S, Agibetov A, Ercole A, Thoral P, Elbers P, Heitzinger C, Kimberger O. Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis. *Journal of Clinical Medicine*. 14 Feb. 2023, doi:10.3390/jcm12041513.

2. **Optimized Renal Replacement Therapy Decisions in Intensive Care: A Reinforcement Learning Approach:** Using the MIMIC-IV database and an external cohort from the Medical University of Vienna, a Q-learning algorithm combined 88 clinical features to derive RRT initiation guidelines. The model achieved 98% agreement with clinician decisions while identifying a high-risk subgroup that benefited from earlier intervention. By framing RRT timing as a Markov decision process, this approach dynamically balances changes in the patient state (see chapter 6):
   Kapral L, Bologheanu R, Azarbeik M, Bilir A, Bartos S, Weiss R, Schaller S, Heitzinger C, Schaden E, Kimberger O. Optimized Renal Replacement Therapy Decisions in Intensive Care: A Reinforcement Learning Approach. **Under Review** in *Intensive Care Medicine*. 2025, doi.org/10.21203/rs.3.rs-6243566/v1.

3. **Development and External Validation of Temporal Fusion Transformer Models for Continuous Intraoperative Blood Pressure Forecasting:** A TFT model trained on 73,009 non-cardiothoracic surgeries forecasted MAP trajectories 7 minutes ahead using low-resolution data (15-second sampling). Internal and external validation yielded mean absolute errors of 4 mmHg and 7 mmHg, respectively, with AUROCs of 0.933 and 0.919 for hypotension prediction. This precision enables proactive hemodynamic management without reliance on invasive arterial waveforms (see chapter 7):
Kapral L, **Dibiasi C**, Jeremic N, Bartos S, Behrens S, Bilir A, Heitzinger C, Kimberger O. Development and external validation of temporal fusion transformer models for continuous intraoperative blood pressure forecasting. *EClinicalMedicine*. 30 Aug. 2024, doi:10.1016/j.eclinm.2024.102797.

3.2.2 Contributions

This dissertation is written entirely by the author in terms of its scientific content and conceptual framework. No generative AI was used to produce the research questions, the results, or the interpretation of findings. Automated tools such as ChatGPT, DeepL, and Grammarly were employed solely for language checks, grammar corrections, minor stylistic refinements, and LaTeX formula transformation. Therefore, 0% of the topics, results, and statistics presented originate from AI generation, while all main arguments and analyses have been developed and executed by the author. With respect to theoretical derivations, no new mathematical proofs were introduced in this work, and any existing proofs or derivations relied upon are attributed to the appropriate literature as cited.

Numerical experiments and the associated software implementation were developed primarily by the author. Across all three publications and related work, the total code produced amounts to thousands of lines in Python, incorporating key libraries such as `TensorFlow 2.x` for neural network modeling, `scikit-learn` for machine learning utilities, and `pandas` for data handling. Visualization was carried out with `matplotlib` and `seaborn`. In instances where external codebases were adapted for specific functionalities, substantial modifications were performed by the author to align them with the particular research questions posed in this thesis. In a few instances, plotting routines were augmented by ChatGPT; however, all core model code and experimental design were manually scripted.

Regarding study 1, approximately 3,000 lines of code were developed for an actor-critic reinforcement learning framework including pre-processing and evaluation. All code was written by the author without AI assistance. The author was responsible for about 60% of the overall work, which included study design, data analysis, and writing. The code basis was created entirely for this specific study.

In study 2, around 4,000 lines of code were created, integrating Q-learning algorithms in Python for outcome analysis. The author contributed roughly 70% of the work, which included conceptual design, code development, and manuscript writing. Only minor AI help was used for plotting snippets. The foundations of this code were adapted from an existing repository, with about 50% rewritten. The code is available at `https://github.com/lorenzkap/MAP_TFT`, incorporating adaptations from `https://github.com/greatwhiz/tft_tf2`.

In study 3, about 4,000 lines of code were written to apply a transformer-based architecture for intraoperative blood pressure forecasting. The author carried out approximately 70% of the work, covering model adaptation, data handling, and drafting the text. AI was used only for minimal support in plotting. The code derives from a publicly accessible source, with approximately 40% modified to optimize the Temporal Fusion Transformer for predicting intraoperative blood

pressure. The code is available at `https://github.com/lorenzkap/RL4RRT`, adapted from `https://github.com/cmudig/AI-Clinician-MIMICIV/tree/main`.

3.2.3 Clinical and Methodological Implications

Together, these studies demonstrate how AI can integrate detailed patient data, real-time clinical insights, and advanced analytics to personalize critical care. By translating population-level medical evidence into tailored treatments, AI bridges the gap between generalized guidelines and individual patient needs. For example, the sepsis and AKI models illustrate how RL can dynamically adjust treatment targets—such as medication dosing or fluid thresholds—in response to a patient's changing condition. Similarly, the high predictive accuracy of the TFT highlights how transformer-based AI models can decode complex physiological patterns, such as unstable blood pressure during surgery, to guide clinical decisions. Methodologically, this work strengthens the reliability of AI in healthcare through testing across diverse patient datasets, transparent model explanations, and validation in real-world critical care settings.

The following chapters outline the medical and technical foundations of this research. Chapter 4 provides essential background on sepsis, AKI, and intraoperative hypotension, as well as the core principles of reinforcement learning and transformer architectures. Chapters 5–7 present the three central studies, each combining technical innovation with clinical relevance. Study 1 (Chapter 5) focuses on RL-driven treatment optimization for sepsis, study 2 (Chapter 6) explores AKI risk prediction using dynamic AI models, and study 3 (Chapter 7) details the TFT's ability to forecast perioperative complications. Collectively, this dissertation lays the groundwork for AI-driven critical care systems that are adaptive, precise, and proactive—ultimately aiming to improve outcomes for the most vulnerable patients.

Chapter 4

Fundamentals

4.1 Medical Background

4.1.1 Sepsis

Sepsis is a *heterogeneous clinical syndrome* defined as a life-threatening organ dysfunction caused by a dysregulated host response to infection [11, 21]. This definition underscores that sepsis, rather than being a single disease, is fundamentally driven less by the invading pathogen than by the host's maladaptive response, which, when uncontrolled, can lead to widespread tissue damage, organ failure, and ultimately death. Recent estimates indicate that sepsis is responsible for approximately 11 million deaths annually on a global scale [22], highlighting its enormous impact on public health. The heterogeneity of sepsis arises from the variable nature of the host's immune response, the diversity of infectious triggers, and the range of clinical phenotypes observed, which together pose significant challenges to both diagnosis and management [21, 23].

The pathophysiology of sepsis involves a complex interplay between pro-inflammatory and anti-inflammatory mechanisms. Initially, the infection triggers an exaggerated immune response characterized by the release of inflammatory cytokines, activation of complement pathways, and coagulation cascades. This early hyperinflammatory state can lead to endothelial dysfunction, capillary leak, and microvascular thrombosis, all of which contribute to tissue hypoperfusion and organ dysfunction. Paradoxically, many patients subsequently enter a phase of immune suppression, rendering them more vulnerable to secondary infections and complicating the clinical course [24, 25]. The interplay between these phases and the ensuing dysregulation underscores the complexity of developing targeted therapies for sepsis.

Current standard treatment for sepsis emphasizes early recognition and rapid intervention. The cornerstone of management care bundles developed in consensus by medical experts from existing evidence, include prompt initiation of broad-spectrum antibiotics, rigorous source control of the underlying infection, and supportive care measures aimed at maintaining adequate tissue perfusion through fluid resuscitation and vasopressor support when necessary [26, 27]. Despite these efforts, sepsis remains a leading cause of morbidity and mortality, and there is a pressing need for adjunctive therapies that can modulate the host response and improve clinical outcomes.

Corticosteroids have been investigated as one such adjunctive therapy for nearly seven decades, ever since early studies by Cook and colleagues first suggested their potential benefit in sepsis [12]. These agents are known for their potent anti-inflammatory and immunomodulatory properties, and their use in sepsis was further bolstered by observations of adrenal insufficiency in critically ill patients [28]. The rationale for corticosteroid therapy in sepsis is based on the concept that, in some patients, the endogenous stress response is insufficient to counteract the overwhelming inflammatory cascade, and that exogenous corticosteroids might help restore hemodynamic stability and modulate the dysregulated immune response.

Clinical investigations have demonstrated that corticosteroid treatment may hasten the resolution of septic shock, particularly by reducing the duration of vasopressor dependency [29, 30,

31]. However, these benefits have not consistently translated into a clear survival advantage, and the overall impact on mortality remains a subject of debate. The variability in outcomes observed across different studies may, in part, be attributable to differences in patient populations, the timing of intervention, dosage, and duration of corticosteroid administration [30]. Current guidelines suggest the use of corticosteroids in septic patients who remain hemodynamically unstable despite adequate fluid resuscitation and vasopressor support, yet they also acknowledge the uncertainty surrounding the optimal therapeutic regimen and the potential for adverse effects such as hyperglycemia, immunosuppression, and muscle weakness [26].

In recent years, the goal of refining corticosteroid therapy in sepsis has increasingly focused on identifying patient subgroups most likely to benefit from such treatment. Given the heterogeneity of the syndrome, a one-size-fits-all approach is unlikely to be effective, and there is growing interest in personalizing therapy using advanced tools such as transcriptomics and machine learning [32, 21]. In particular, RL, a branch of ML dedicated to sequential decision making in dynamic environments, has emerged as a promising method for analyzing high-resolution clinical data and optimizing treatment strategies in the intensive care unit [33, 34]. By exploiting the differences in individual patient trajectories and clinician treatment patterns, these algorithms aim to construct decision support tools that can recommend an optimal corticosteroid regimen tailored to the patient's evolving condition. This approach has the potential to overcome the limitations of traditional clinical trials, which are often challenged by the extreme variability of sepsis phenotypes, and may ultimately lead to more effective and personalized interventions.

4.1.2 Acute Kidney Injury and Renal Replacement Therapy

AKI is a complex and multifaceted syndrome characterized by a sudden decline in renal function, often developing over a period of hours to days. The kidneys, which are responsible for maintaining homeostasis through the regulation of fluid balance, electrolyte composition, and the removal of metabolic waste, can be severely compromised by insults such as ischemia, nephrotoxicity, or sepsis [35, 36]. In the ICU, AKI is a common complication, with studies indicating that up to 50% of critically ill patients experience some degree of renal dysfunction during their stay [17, 37]. The presence of AKI not only predisposes patients to immediate metabolic derangements, such as hyperkalaemia, acidosis, and fluid overload, but also increases the risk of multiorgan failure and death, with mortality rates exceeding 50% in severe cases [38, 39].

The management of AKI in critically ill patients frequently necessitates the use of RRT, a group of extracorporeal techniques designed to mimic the filtration functions of healthy kidneys. RRT modalities, which include intermittent hemodialysis, continuous renal replacement therapy (CRRT), and peritoneal dialysis, serve to remove toxins, regulate electrolyte imbalances, and correct acid-base disturbances [40]. However, it is important to recognize that while RRT can mitigate the biochemical consequences of AKI, it does not promote the recovery of injured renal tissue; rather, it functions as a supportive therapy until renal recovery occurs or until the patient is bridged to a more definitive therapy [36, 35]. This supportive nature of RRT underscores the critical challenge clinicians face: balancing the risks and costs inherent to the procedure against the potential harms of delayed intervention in the face of rapidly evolving renal dysfunction.

An important area of ongoing debate within critical care nephrology concerns the optimal timing of RRT initiation in patients with AKI. Early initiation of RRT refers to the start of therapy at a stage when renal impairment is recognized but before the full development of life-threatening complications. The rationale for this approach lies in the hypothesis that early intervention may prevent the accumulation of uremic toxins and stabilize metabolic imbalances, thereby forestalling further organ dysfunction [41, 42]. In contrast, a strategy of late initiation delays RRT until more definitive clinical or biochemical indications are present. While early

RRT might intuitively seem beneficial, several high-quality randomized controlled trials have failed to demonstrate a clear survival benefit with early intervention; in some cases, premature initiation has been associated with increased risks such as hemodynamic instability, infection, and the unnecessary use of resources [43, 42]. Conversely, delaying RRT until later stages may allow for spontaneous renal recovery in some patients, yet this delay can also expose patients to the detrimental effects of severe metabolic disturbances, including fluid overload and electrolyte imbalances, which themselves are associated with higher mortality rates [41]. Thus, the decision regarding the timing of RRT initiation remains a delicate balance, closely linked to the severity of the underlying pathology and the patient's overall clinical trajectory.

Despite the availability of various biomarkers and the assessment of urine output as tools to estimate renal function, their ability to accurately predict which patients will progress to severe, clinically significant AKI is limited. This diagnostic uncertainty complicates the determination of when to initiate RRT, as the conventional criteria based solely on kidney function markers may not capture the dynamic and heterogeneous progression of the injury. Additionally, the subsequent decision of when to discontinue RRT, once initiated, further contributes to the complexity of AKI management. In recent years, the application of ML, particularly RL, has emerged as a promising avenue for addressing these clinical challenges. Unlike traditional supervised learning methods, RL algorithms are adept at handling sequential decision-making problems by continuously learning and adapting policies based on the long-term outcomes of therapeutic interventions [33]. By framing the initiation and discontinuation of RRT as a series of interconnected decisions within a Markov decision process, RL techniques such as Q-learning and policy gradients have the potential to optimize individualized treatment strategies [44, 45]. Early studies have demonstrated that these approaches can lead to improved patient outcomes in related critical care scenarios, such as sepsis management and dialysis scheduling [33, 46].

The overarching hypothesis guiding current research is that the progression of the underlying pathological process in AKI, rather than isolated biomarkers or intermittent measures of urine output, is the key determinant of a patient's trajectory and subsequent need for RRT. In order to develop an optimal, individualized strategy for initiating RRT, a more refined understanding of the temporal evolution of AKI and its associated complications is essential.

4.1.3 Blood Pressure Management

Blood pressure is a fundamental physiological parameter that represents the force exerted by circulating blood on the surface of the walls of blood vessels. It is determined by two major components: cardiac output, which is the volume of blood ejected by the heart per unit time, and systemic vascular resistance, which reflects the degree of constriction of the peripheral vasculature [47]. The interaction between these components is determined by complex neurohumoral and renal mechanisms, which are continuously adjusted to meet the metabolic demands of the body. Under normal conditions, blood pressure is tightly regulated to ensure adequate tissue perfusion and oxygen delivery, but deviations from the norm can be the precursor to a range of clinical outcomes. Both hypotension (abnormally low blood pressure) and hypertension (abnormally high blood pressure) are associated with adverse outcomes; hypotension can lead to inadequate perfusion and organ ischemia, while hypertension is a major risk factor for cardiovascular events such as myocardial infarction and stroke.

The hemodynamic framework underlying blood pressure regulation involves the study of blood flow dynamics and the forces acting on the circulatory system. Hemodynamics is critical to understanding how variations in heart rate, myocardial contractility, blood volume, and vascular tone contribute to changes in blood pressure. For example, during surgery, particularly under general anesthesia, the administration of hypnotics and opioid analgesics can depress

cardiac contractility and reduce systemic vascular resistance, predisposing patients to hypotensive episodes [16]. These effects are often amplified by additional intraoperative stressors such as blood loss, hypovolemia, and patient positioning. Inadequate blood pressure during critical periods has been associated with adverse outcomes including myocardial injury, acute kidney injury, and neurological disorders such as delirium [48]. Therefore, understanding the hemodynamic responses during surgery is essential to reduce the risk of end-organ damage.

Pharmacological interventions play a central role in the management of blood pressure disorders, both chronic and perioperative. Blood pressure medications include a wide variety of agents with different mechanisms of action. For example, vasopressors – commonly used during episodes of intraoperative hypotension – work by inducing vasoconstriction and thereby increasing systemic vascular resistance to restore adequate perfusion pressure. Conversely, short-acting antihypertensives such as urapidil (an α1-receptor antagonist) and esmolol (a β1-adrenergic blocker) can be used intraoperatively to control acute increases in blood pressure by modulating vascular tone and heart rate. Meanwhile, other agents-such as β-adrenergic blockers, calcium channel blockers, and angiotensin-converting enzyme inhibitors-are often used in the long-term management of chronic hypertension to help prevent the harmful effects of persistently elevated blood pressure. The selection of these drugs requires a detailed understanding of their pharmacodynamics and pharmacokinetics, particularly in situations where rapid hemodynamic changes occur, such as during anesthesia-induced stress.

Moreover, patient-specific factors such as age, comorbidities, and inherent variations in vascular reactivity add layers of complexity to blood pressure management. Elderly individuals or patients with pre-existing cardiovascular diseases may exhibit impaired baroreceptor sensitivity and altered vascular compliance, rendering them more susceptible to rapid fluctuations in blood pressure during perioperative periods [49]. Recent advances in predictive modeling have sought to address these challenges by employing machine learning algorithms to hypotension based on both static and dynamic clinical data [20]. These innovative approaches, such as the temporal fusion transformer (TFT) algorithm, are designed to integrate heterogeneous data sources—ranging from demographic information to real-time hemodynamic measurements—to provide early warnings of impending hypotension. The ability to predict such episodes holds promise for transforming intraoperative management from a reactive to a proactive paradigm, thereby potentially reducing the incidence of hypotension-related complications [16].

A detailed understanding of blood pressure regulation, the hemodynamic principles underlying circulatory function, and the pharmacological strategies available to modulate blood pressure are essential for improving clinical outcomes, especially in high-risk scenarios such as surgery under general anesthesia. The integration of traditional physiological knowledge with modern predictive analytics offers a promising path towards individualized, timely interventions aimed at minimizing the risks associated with hemodynamic instability.

4.2 Basic Concepts of Reinforcement Learning

Reinforcement learning (RL) is a method of ML. Together with supervised learning and unsupervised learning, RL constitutes one of the fundamental approaches in machine learning. Unlike other methods, RL does not require a data set composed of analyzed example data. Instead, an agent interacts with an environment and autonomously learns a strategy to maximize the reward generated by the environment. The agent does not receive any information or instructions about which strategy is optimal; it only receives the reward.

Before looking at specific algorithms, it is important to define some basic concepts. The following definitions of RL basics are based on the discussion in reference [19].

4.2.1 Policy

The policy defines the agent's behavior at a given time. Specific actions are associated with the states of the environment that have been reached. The complexity of the policy heavily depends on the environment. In the simplest cases, it can be merely a table of all possible states, but it is usually of a stochastic nature. For complex applications, it can also demand substantial computational resources.

The policy can be considered the core of RL, as it alone suffices to determine the agent's behavior in specific situations.

In the literature, the policy is often denoted as $\pi(a, s)$ to indicate its dependence on the state s and the action a.

4.2.2 Reward Signal

The reward signal defines the ultimate goal of the RL problem. After each time step, the environment sends a real number, the reward, to the agent. The agent's sole objective is to maximize this reward over time. The reward sent is dependent on the current state of the environment and the agent's current action. The agent can influence the reward signal only by selecting specific actions.

Therefore, the reward signal represents the primary basis for modifying the policy. If a selected action leads to a low reward, the policy is adjusted so that, in the same initial situation in the future, a different action is more likely to be chosen.

4.2.3 Value Function

The value function indicates what is most beneficial in the long run. Essentially, the value of a state is equal to the total reward that an agent expects to accumulate starting from that state in the future. In other words, the reward defines the intrinsic and immediate desirability of a state. In contrast, the value describes the desirability in the long term, as it accounts for the rewards that will be obtained in subsequent states.

Values are therefore predictions of rewards and are given greater consideration in action selection than the rewards themselves. The agent seeks an action with the maximum value, not with the maximum reward. However, estimating the value is significantly more challenging than estimating the reward. Rewards can be directly obtained from the environment, whereas values must be estimated from observation sequences. The method of accurately estimating the value of a state is considered the most crucial component of Reinforcement Learning.

In the literature, the value of an action is typically denoted as $Q(s, a)$, which is also the namesake of Q-Learning. The value of a state is denoted as $V(s)$.

4.2.4 The Model

Some RL systems use models of the environment. The model's task is to mimic the environment's behavior so that this behavior can be predicted. For example, using the model, it should be possible to determine the next value $V(s_{t+1})$ based on a pair consisting of the current state and the chosen action. This allows for the planning of future actions. Methods that utilize models are referred to as *model-based methods*. Without using models, learning must occur solely through trial and error.

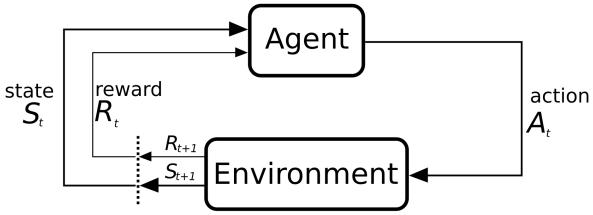


Fig. 4.1: The interaction between the agent and the environment [19]

4.2.5 Exploration and Exploitation

RL requires a mechanism to explore the environment. During the learning process, the agent must balance exploration and exploitation. Exploration involves investigating the environment and gaining information that can inform future decisions. In contrast, exploitation refers to selecting the best decision based on the currently available information, specifically choosing the action with the highest Q-value [50].

This balance has been extensively studied, particularly in the context of the *multi-armed bandit problem*, where simple exploration methods have proven to be practical. [51]

To manage this balance, a parameter ϵ, where $0 < \epsilon < 1$, is commonly used. The parameter ϵ represents the probability with which the agent chooses to explore. When exploration occurs, a random action is taken, allowing the agent to gather information about actions that might potentially lead to losses. Conversely, with a probability of $1-\epsilon$, the agent engages in exploitation, selecting the action with the highest long-term value based on the current information.

In practical applications, ϵ is typically set to a high value at the beginning to enable the agent to gather as much information about the environment as possible. Depending on the system's complexity, ϵ is then gradually decreased after several training iterations, usually by dividing it by a constant greater than zero. This process is repeated until ϵ approaches zero.

4.3 Finite Markov Decision Process

The foundation of RL is based on the Markov Decision Process (MDP). An MDP represents a strategy for decision-making in a system that is partially stochastic and partially controllable [52].

After each time step, a state s is reached. The decision-maker, who in the context of RL is the agent, must choose an action a that is permissible in state s. This action transitions the process to state s' and the agent receives a reward $r_a(s, s')$. The probability of transitioning to state s' depends on the action taken. If s and a are independent of previous states and actions, the Markov property is satisfied, allowing the application of rules to solve the Markov problem (see Figure 4.1).

The MDP is an extension of the *Markov chain*, as adding multiple actions introduces a decision-making capability.

RL closely resembles the MDP: stochastic rules determine the information the agent receives after each time step. In most cases, the agent is provided with a scalar directly associated with the previously chosen action. At each time step t, the agent receives an observation o_t, which typically includes the reward r_t. Based on this observation, the agent selects an action a_t from the set of possible actions in state s_t. This decision depends on the agent's policy $\pi(a, s)$; the agent maps the state s to selection probabilities for each possible action a. Subsequently, the

environment transitions to a new state s_{t+1}. The reward r_{t+1} associated with the transition (s_t, a_t, s_{t+1}) is also determined. The agent's goal is to maximize the sum of all received rewards. This is achieved by continuously adjusting the policy $\pi(a, s)$ based on the agent's experiences. Therefore, RL is also suitable for problems where a trade-off between short-term and long-term reward optimization is necessary.

4.4 Return

The agent aims to maximize the cumulative reward in the long term; the sequence of rewards after t time steps can be written as $r_{t+1}, r_{t+2}, r_{t+3}, \ldots$. The *return* G_t can be defined in the simplest case as

$$G_t = r_{t+1} + r_{t+2} + r_{t+3} + \cdots + r_T, \tag{4.1}$$

where T represents the final time step. Since problems can consist of many time steps, the *discount rate* γ is introduced. Using γ, the *discounted return* G_t is defined as

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}, \tag{4.2}$$

where γ is a parameter such that $0 \leq \gamma \leq 1$.

To capture the long-term performance of a policy, we often consider the *expected cumulative reward*, denoted by J. Formally, we can write

$$J = \mathbb{E}[G_0] \;=\; \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k \, r_{k+1}\right], \tag{4.3}$$

where G_0 is the (discounted) return starting at time $t = 0$, and the expectation is taken with respect to the stochastic process induced by the policy and the environment. The goal of many reinforcement learning algorithms is to find a policy π that maximizes this expected return J.

4.5 Markov Decision Process

RL satisfies the Markov property and is thus modeled as a MDP.

A finite MDP is defined by a set of states s, actions a, and the probability of transitioning to any possible next state s' from a state-action pair (s, a). This transition is associated with a reward r. Formally, this can be expressed as

$$p(s', r|s, a) = \Pr(s_{t+1} = s', r_{t+1} = r|s_t = s, a_t = a), \tag{4.4}$$

where Pr denotes probability. From Equation 4.4, the expected reward of a state-action pair can be formulated as:

$$r(s, a) = \mathbb{E}[r_{t+1}|s_t = s, a_t = a] = \sum_r r \sum_{s'} p(s', r|s, a), \tag{4.5}$$

with the state transition probabilities

$$p(s'|s, a) = \Pr(s_{t+1} = s'|s_t = s, a_t = a) = \sum_r p(s', r|s, a), \tag{4.6}$$

where $\mathbb{E}[.]$ denotes the expectation value. The expected rewards are described by the value function, which is defined with respect to a particular policy. To recall: the policy π assigns a probability $\pi(a|s)$ to selecting a possible action a in state s. The expected reward obtained by following policy π in state s is called the value of state s under policy π and is denoted as $V_\pi(s)$. For MDPs, this value function can be formally written as

$$V_\pi(s) = \mathbb{E}_\pi[G_t|s_t = s] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|s_t = s\right]. \tag{4.7}$$

Similarly, the value of an action a in state s under policy π can be defined as $Q_\pi(s, a)$. This function corresponds to the expected reward resulting from choosing action a in state s and subsequently following policy π:

$$Q_\pi(s, a) = \mathbb{E}_\pi[G_t|s_t = s, a_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|s_t = s, a_t = a\right]. \tag{4.8}$$

This function is known as the *Action-Value Function* of policy π and plays a central role in RL methods.

V_π and Q_π are estimated from the agent's experience. If these functions are approximated by computing individual average values for each observed state s, the approach is referred to as *Monte Carlo* methods. Through numerous repetitions, the optimal values of these functions are gradually approximated.

A fundamental property of these value functions, which is utilized in RL and Dynamic Programming, is their recursive relationship. For any policy π and any state s, the following consistency condition holds:

$$\begin{aligned}
V_\pi(s) &= \mathbb{E}_\pi[G_t|s_t = s] \\
&= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \,\middle|\, s_t = s\right] \\
&= \mathbb{E}_\pi\left[r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \,\middle|\, s_t = s\right] \\
&= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left(r + \gamma \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \,\middle|\, s_t = s'\right]\right) \\
&= \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a) \left(r + \gamma V_\pi(s')\right).
\end{aligned} \tag{4.9}$$

Equation 4.9 is known as the *Bellman Equation* and expresses the relationship between the value V_π of a state and its possible subsequent states s' [53]. The Bellman Equation is a sum over the three variables a, s', and r. For each triplet, the probability $\pi(a|s)p(s', r|s, a)$ is calculated and then summed, where $\pi(a|s)$ represents the policy.

4.6 Dynamic Programming

The fundamental idea of Dynamic Programming (DP) and is the use of value functions to structure the search for an optimal policy [53, 54].

4.6.1 Policy Evaluation

The process of computing the state-value function V_π, as described by Equation 4.9, for a given policy π is known as Policy Evaluation. The Bellman Equation can be rewritten as a successive approximation for the next state-value function:

$$\begin{aligned} V_{k+1}(s) &= \mathbb{E}_\pi \left[r_{t+1} + \gamma V_k(s_{t+1}) \mid s_t = s \right] \\ &= \sum_a \pi(a \mid s) \sum_{s',r} p(s', r \mid s, a) \left(r + \gamma V_k(s') \right), \end{aligned} \tag{4.10}$$

where $V_k = V_\pi$ is a fixed point for this update rule. Starting from an initial state V_0, V_π can be iteratively approximated.

This algorithm allows for the determination of V_π for a given policy π. However, it remains unclear whether modifying this policy would be advantageous. One way to determine when a policy change is beneficial is by observing the Action-Value Function Q_π. Q_π provides the value of a state s after selecting an action a. Consequently, Equation 4.8 can be rewritten using Equation 4.9:

$$\begin{aligned} Q_\pi(s, a) &= \mathbb{E}_\pi \left[r_{t+1} + \gamma V_\pi(s_{t+1}) \mid s_t = s, a_t = a \right] \\ &= \sum_{s',r} p(s', r \mid s, a) \left(r + \gamma V_\pi(s') \right). \end{aligned} \tag{4.11}$$

The criterion for policy improvement is whether Q_π is greater than or less than V_π. If Q_π is greater than V_π, it is better to choose action a in state s and then follow policy π rather than always following policy π. In such cases, the policy should be updated.

By selecting the best action a in all states s based on $Q_\pi(s, a)$, the *Greedy Policy* π' can be introduced:

$$\begin{aligned} \pi'(s) &= \arg\max_a Q_\pi(s, a) \\ &= \arg\max_a \mathbb{E}_\pi \left[r_{t+1} + \gamma V_\pi(s_{t+1}) \mid s_t = s, a_t = a \right] \\ &= \arg\max_a \sum_{s',r} p(s', r \mid s, a) \left(r + \gamma V_\pi(s') \right), \end{aligned} \tag{4.12}$$

where $\arg\max_a$ denotes the value of a that maximizes the expression. The process of improving policy π by selecting the action with the highest value based on the original policy's value function π' is called *Policy Improvement*.

From Equation 4.12, an iterative algorithm can be constructed to improve the policy, as illustrated in Algorithm 1.

However, this algorithm has the drawback that each iteration step includes a policy evaluation, which is itself an iterative process. The next consideration for improving the algorithm is based on reducing the number of iterations. An important special case is known as *Value Iteration*, which is equivalent to terminating after the first iteration step. With this improvement, Algorithm 1 can be written more concisely as:

$$\begin{aligned} V_{k+1}(s) &= \max_a \mathbb{E}_\pi \left[r_{t+1} + \gamma V_k(s_{t+1}) \mid s_t = s \right] \\ &= \max_a \sum_{s',r} p(s', r \mid s, a) \left(r + \gamma V_k(s') \right). \end{aligned} \tag{4.13}$$

The major disadvantage of the methods presented so far is that these operations iterate over all existing states of the MDP. In complex environments, this becomes computationally infeasible due to the enormous number of possible states.

Algorithm 1 This algorithm presents a pseudocode for policy improvement from [19] p. 97. Before the policy π can be improved, it must be evaluated according to Equation 4.10. Subsequently, the policy can be improved using Equation 4.12 until the policy stabilizes. It is important to note that the process loops back to step 2 repeatedly, meaning that the policy must be re-evaluated continuously.

1. Initialization:
$V(s) \in \mathbb{R}$ and $\pi(s) \in \mathcal{A}(s)$ arbitrarily for all $s \in \mathcal{S}$.
2. Policy Evaluation:
repeat
  $\Delta \leftarrow 0$
  for each $s \in \mathcal{S}$ **do**
    $v \leftarrow V(s)$
    $V(s) \leftarrow \sum_{s',r} p(s', r \mid s, \pi(s)) \, (r + \gamma V(s'))$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
  end for
until $\Delta < \theta$ (a small positive number)
3. Policy Improvement:
`policy-stable` $\leftarrow$ `true`
for each $s \in \mathcal{S}$ **do**
  $a \leftarrow \pi(s)$
  $\pi(s) \leftarrow \arg\max_a \sum_{s',r} p(s', r \mid s, a) \, (r + \gamma V(s'))$
  if $a \neq \pi(s)$ **then**
    `policy-stable` $\leftarrow$ `false`
  end if
end for
if `policy-stable` **then**
  Stop and return V and π
else
  Go to step 2.
end if

Algorithm 2 This algorithm related to Value Iteration is sourced from reference [19] p.101 and demonstrates an improvement over the algorithm in Algorithm 1. Instead of repeatedly evaluating the policy, only a single iteration step is performed to update V according to Equation 4.13. This algorithm is referred to as *Value Iteration*.

Initialize array V arbitrarily (e.g., $V(s) = 0$ for all $s \in \mathcal{S}^+$).
repeat
  $\Delta \leftarrow 0$
  for each $s \in \mathcal{S}$ **do**
    $v \leftarrow V(s)$
    $V(s) \leftarrow \max_a \sum_{s',r} p(s', r \mid s, a)\,(r + \gamma V(s'))$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
  end for

until $\Delta < \theta$ (a small positive number)
Output a deterministic policy π, such that

$$\pi(s) = \arg\max_a \sum_{s',r} p(s', r \mid s, a)\,(r + \gamma V(s'))$$

4.7 Monte-Carlo Methods

In the previous chapters, comprehensive knowledge of the entire system was assumed. However, Monte-Carlo (MC) methods rely solely on *experience*—sequences of states, actions, and rewards generated from interactions with the environment—for the learning process [55]. MC methods provide approaches to solve problems by creating average values from the accumulated experiences.

To estimate the state-value function $V_\pi(s)$, an average of the returns obtained after visiting state s_t is calculated. This average approximates the expected value with a high number of repetitions. Consequently, the individual estimates for each state are independent of one another.

In the absence of a model, it is preferable to use estimated action-values (Q-values) over state-values (V-values). The estimation of action-values follows a similar procedure to that of state-values; however, instead of visiting states, state-action pairs (s, a) are observed. After each visit, the MC method estimates a value for the state-action pair as the average of the returns resulting from selecting that specific action in the given state.

However, this approach encounters the problem that some state-action pairs may never be visited.

4.8 Temporal-Difference Learning

Temporal-Difference (TD) Learning plays a central role in the understanding of RL. It is a hybrid approach that combines elements of DP and MC methods. On one hand, TD methods can learn directly from experience, similar to MC methods [56]. On the other hand, they create estimates based on other estimates, akin to DP.

4.8.1 TD Estimation

Like MC methods, TD uses experience to approximate V_π. MC methods wait until the return of a state visit is known and then use this return as the target for updating $V(s_t)$. This can be mathematically expressed as:

$$V(s_t) \leftarrow V(s_t) + \alpha \left(G_t - V(s_t)\right), \tag{4.14}$$

where G_t represents the actual return at time t, and α denotes the step size. The MC method would need to wait until the end of the episode to compute Equation 4.14 and determine G_t.

The advantage of TD methods is that only the next time step is required: at time $t+1$, the observed reward r_{t+1} and an estimate of $V(s_{t+1})$ are used to compute a useful update. Based on this insight, the simplest TD method can be formulated as:

$$V(s_t) \leftarrow V(s_t) + \alpha \left(r_{t+1} + \gamma V(s_{t+1}) - V(s_t)\right). \tag{4.15}$$

The value that the approximation aims to reach is referred to as the *target*. In the case of the MC method, the target is G_t, whereas for the TD method, the target is $r_{t+1} + \gamma V(s_{t+1})$. Compared to Equation 4.9, the target for the MC method corresponds to the first line, and the target for DP corresponds to the last line of this equation.

TD methods that generate updates based on existing estimates are known as *bootstrapping* methods [57].

The target in TD represents an estimated value: on one hand, it accumulates expected values as in MC methods; on the other hand, it uses $V(s_{t+1})$ instead of the actual V_π. For this reason, TD combines the accumulation of expected values from MC methods with the bootstrapping characteristic of DP.

4.8.2 SARSA: On-Policy TD

To address the exploration-exploitation dilemma discussed in Chapter 4.2.5, there are two different approaches: *On-Policy* learning algorithms evaluate and improve the same policy that is used to select actions. In short, the target policy is simultaneously the policy that determines the agent's behavior. In contrast, *Off-Policy* algorithms evaluate and improve a policy that is different from the policy used to select actions. Thus, in this case, the target policy is different from the behavior-determining policy [50]. The SARSA method is an example of an On-Policy method [58].

Unlike the methods discussed above, the SARSA method uses the action-value function Q_π instead of the state-value function V_π. This can be achieved using the theory already discussed, as an episode consists of an alternating sequence of states s_t and state-action pairs (a_t, r_{t+1}), which include all the necessary components for the SARSA algorithm.

The fundamental algorithm for iteratively calculating the action-values can be formulated as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left(r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right). \tag{4.16}$$

This update (Equation 4.16) is performed after each transition starting from a non-terminal state s_t. If s_{t+1} is a terminal state, $Q(s_{t+1}, a_{t+1})$ is defined as 0. This rule uses each element of the following sequence of events $(s_t, a_t, r_{t+1}, s_{t+1}, a_{t+1})$. This sequence of events includes all the necessary components for transitioning from one state-action pair to the next, which is why the algorithm is named SARSA.

To create a complete algorithm, as with all On-Policy methods, a Q_π is continuously estimated based on the behavior of policy π. Simultaneously, π is also modified, as π can be derived from Q. From this, a general form of the SARSA algorithm can be obtained; it is presented as pseudocode in Algorithm 3.

Algorithm 3 The SARSA algorithm from [19] p.155. After all Q-values are initialized, an action is chosen in state s according to the current policy. The environment then returns the values r and s'. Subsequently, an action a' is chosen according to the same policy as before. Using Equation 4.16, $Q(s,a)$ is updated. These steps are repeated until the terminal state is reached.

Initialize $Q(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$ arbitrarily, and $Q(\text{terminal-state}, \cdot) = 0$.
repeat
  Initialize s
  Choose a from s using policy derived from Q (e.g., ϵ-greedy)
  repeat
    Take action a, observe r, s'
    Choose A' from S' using policy derived from Q (e.g., ϵ-greedy)
    $Q(s,a) \leftarrow Q(s,a) + \alpha\left(r + \gamma Q(s',a') - Q(s,a)\right)$
    $s \leftarrow s'; a \leftarrow a'$
  until s is terminal
until convergence

4.9 Q-Learning

The Q-Learning algorithm sit one of the most significant algorithms in RL [59]. The simplest form, One-Step Q-Learning) and can be defined as follows:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left(r_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t)\right). \tag{4.17}$$

In this case, the learned action-value function Q is directly approximated from Q_*, the optimal action-value function, and is therefore independent of the policy. Q-Learning is an Off-Policy learning method because the policy used to select actions is different from the target policy. Instead of following the target policy, the action with the highest Q-value is always selected, which essentially corresponds to a greedy ϵ-policy with ϵ approaching 0. However, the target policy still has an effect because it determines which state-action pairs are visited and updated.

4.9.1 Bellman Optimality Equation

The objective is to find the optimal action-value function Q^* that satisfies the Bellman optimality equation

$$Q^*(s,a) \;=\; \mathbb{E}\big[r_{t+1} + \gamma \max_{b\in\mathcal{A}} Q^*(s_{t+1},b)\big] \quad \text{for all } (s,a),$$

with the expectation taken over the random next state s_{t+1} and reward r_{t+1} given current state s and action a. This Q^* is the unique fixed point of the Bellman optimality operator T, which acts componentwise:

$$(TQ)(s,a) \;=\; \mathbb{E}\big[r_{t+1} + \gamma \max_b Q(s_{t+1},b)\big]. \tag{4.18}$$

4.9.2 Key Assumptions for Convergence

We consider a MDP with finite state set $\mathcal{S}$, finite action set $\mathcal{A}$, discount factor $0 \le \gamma < 1$, transition kernel $T(\cdot \mid s, a)$, and reward function $r(s, a)$ (bounded). We state standard assumptions ensuring almost-sure convergence of Q-Learning to Q^* [59, 60]:

(A1) **Sufficient Exploration.** Every state-action pair (s, a) is updated infinitely often. Formally,

$$\sum_{t=0}^{\infty} \mathbf{1}\{(s_t, a_t) = (s, a)\} \;=\; \infty \quad \text{with probability 1.}$$

(A2) **Bounded Rewards.** There is a constant $R_{\max} < \infty$ such that $|r_{t+1}| \le R_{\max}$.

(A3) **Step Sizes (Robbins–Monro).** The learning rates $\{\alpha_t\}$ satisfy

$$\sum_{t=0}^{\infty} \alpha_t \;=\; \infty, \quad \sum_{t=0}^{\infty} \alpha_t^2 \;<\; \infty,$$

as introduced in [61].

The infinite-visitations condition (A1) ensures that each pair (s, a) receives infinitely many updates. Bounded rewards (A2) help keep Q iterates bounded. The step-size conditions (A3) ensure stable yet persistent updates in the sense of classical stochastic approximation.

4.9.3 Convergence Proof

We present a convergence proof that merges standard boundedness with a four-stage convergence framework: (i) Bellman-operator contraction, (ii) update decomposition, (iii) martingale noise analysis, and (iv) a final stochastic-approximation argument [60, 62, 63].

4.9.3.1 Boundedness of Q-Learning Iterates

Lemma 1 (Boundedness)**.** *Under assumption (A2), the sequence $\{Q_t\}$ generated by (4.17) remains uniformly bounded almost surely. Specifically, if $|r_{t+1}| \le R_{\max}$ and $|Q_0(s, a)| \le C_0$ for all (s, a), then letting*

$$C \;=\; \max\Big\{ C_0,\; \tfrac{R_{\max}}{1-\gamma} \Big\},$$

we have $|Q_t(s, a)| \le C$ for all t and (s, a) almost surely.

Proof. We argue by induction. The base case $|Q_0(s, a)| \le C$ holds by definition. Assume $|Q_t(s, a)| \le C$ for all (s, a). Only (s_t, a_t) is updated at time t. Then

$$Q_{t+1}(s_t, a_t) \;=\; (1 - \alpha_t)\, Q_t(s_t, a_t) \;+\; \alpha_t\big(r_{t+1} + \gamma \max_b Q_t(s_{t+1}, b)\big).$$

Since $|r_{t+1}| \le R_{\max}$ and $|\max_b Q_t(s_{t+1}, b)| \le C$ by the induction hypothesis, we have

$$\big|r_{t+1} + \gamma \max_b Q_t(s_{t+1}, b)\big| \;\le\; R_{\max} + \gamma C.$$

Because $C \ge \frac{R_{\max}}{1-\gamma}$, it follows that $R_{\max} + \gamma C \;\le\; C(1 - \gamma) + \gamma C \;=\; C$. Hence

$$\big|Q_{t+1}(s_t, a_t)\big| \;\le\; (1 - \alpha_t)\, C + \alpha_t\, C \;=\; C.$$

For other pairs $(s,a) \neq (s_t,a_t)$, we have $Q_{t+1}(s,a) = Q_t(s,a)$, so they remain bounded by C by the induction hypothesis. □

4.9.3.2 Stage 1: Bellman Operator Contraction

A key property of the Bellman optimality operator T is its sup-norm contraction:

Proposition 1. *For any two action-value functions Q_1 and Q_2,*

$$\|TQ_1 - TQ_2\|_\infty \;\le\; \gamma\,\|Q_1 - Q_2\|_\infty.$$

Proof. Recall that

$$(TQ)(s,a) \;=\; \mathbb{E}\Big[r_{t+1} + \gamma \max_b Q(s_{t+1},b)\Big].$$

Hence,

$$(TQ_1)(s,a) \;-\; (TQ_2)(s,a) \;=\; \gamma\,\mathbb{E}\Big[\max_b Q_1(s_{t+1},b) \;-\; \max_b Q_2(s_{t+1},b)\Big].$$

Taking absolute values and noting that $\big|\max_b x_b \;-\; \max_b y_b\big| \;\le\; \max_b |x_b - y_b|$, we get

$$\big|(TQ_1)(s,a) - (TQ_2)(s,a)\big| \;\le\; \gamma\,\mathbb{E}\Big[\max_b |Q_1(s_{t+1},b) - Q_2(s_{t+1},b)|\Big].$$

Because $\max_{(s,a)}|Q_1(s,a) - Q_2(s,a)| = \|Q_1 - Q_2\|_\infty$ is a constant with respect to the expectation,

$$\big|(TQ_1)(s,a) - (TQ_2)(s,a)\big| \;\le\; \gamma\,\|Q_1 - Q_2\|_\infty.$$

Taking the supremum over all (s,a) then yields $\|TQ_1 - TQ_2\|_\infty \le \gamma\,\|Q_1 - Q_2\|_\infty$, as required. □

4.9.3.3 Stage 2: Update Decomposition

Starting from the one-step Q-Learning update:

$$Q_{t+1}(s,a) \;=\; Q_t(s,a) \;+\; \alpha_t(s,a)\Big(r_{t+1} + \gamma \max_b Q_t(s_{t+1},b) \;-\; Q_t(s,a)\Big),$$

we rewrite it in a way that cleanly separates the "contraction part" (the shift toward TQ_t) from a "noise term." Define

$$\Delta_t(s,a) \;=\; Q_t(s,a) \;-\; Q^*(s,a),$$

where Q^* is the fixed point of T (so $Q^* = TQ^*$). Then

$$Q_{t+1}(s,a) \;=\; Q_t(s,a) \;+\; \alpha_t(s,a)\Big((TQ_t)(s,a) \;-\; Q_t(s,a)\Big) \;+\; \alpha_t(s,a)\,w_t(s,a),$$

where the "noise" is given by

$$w_t(s,a) \;=\; \big(r_{t+1} + \gamma \max_b Q_t(s_{t+1},b)\big) \;-\; (TQ_t)(s,a).$$

Subtracting $Q^*(s,a)$ on both sides shows how the error Δ_{t+1} evolves:

$$\Delta_{t+1}(s,a) \;=\; (1 - \alpha_t(s,a))\,\Delta_t(s,a) \;+\; \alpha_t(s,a)\Big((TQ_t)(s,a) \;-\; (TQ^*)(s,a)\Big) \;+\; \alpha_t(s,a)\,w_t(s,a). \tag{4.19}$$

Since $TQ^* = Q^*$, the difference in brackets can be written as $(TQ_t)(s,a) - (TQ^*)(s,a)$. By the contraction property from Stage 1, we know

$$\|(TQ_t) - (TQ^*)\|_\infty \;\le\; \gamma\,\|Q_t - Q^*\|_\infty \;=\; \gamma\,\|\Delta_t\|_\infty.$$

Hence in subsequent stages, we will see that the contraction term pulls Q_t toward Q^*, while w_t is treated as a martingale-difference noise whose impact diminishes as $\alpha_t \to 0$.

4.9.3.4 Stage 3: Martingale Noise Analysis

Lemma 2 (Martingale Difference)**.** *Let* $\mathcal{F}_t = \sigma\big(Q_0,\,(s_0,a_0,r_1),\ldots,(s_t,a_t,r_{t+1})\big)$*. Then for each* (s,a)*,*

$$\mathbb{E}[\,w_t(s,a)\,|\,\mathcal{F}_t] \;=\; 0, \quad \textit{and} \quad \mathbb{E}\big[w_t(s,a)^2\,\big|\,\mathcal{F}_t\big] \;\le\; \sigma^2$$

for some finite constant σ^2*.*

Proof. **(i) Zero Mean.** If $(s,a) \neq (s_t,a_t)$, then $w_t(s,a) = 0$. If $(s,a) = (s_t,a_t)$, then

$$\mathbb{E}\big[w_t(s,a)\,\big|\,\mathcal{F}_t\big] \;=\; \mathbb{E}\Big[r_{t+1} + \gamma \max_b Q_t(s_{t+1},b)\,\Big|\,s_t,a_t\Big] \;-\; (TQ_t)(s,a) \;=\; 0,$$

by the definition of $(TQ_t)(s,a)$ (from equation 4.18).

(ii) Bounded Variance. From Lemma 1, the iterates $\{Q_t\}$ are almost surely bounded, say

$$|Q_t(s,a)| \le C.$$

Then,

$$|r_{t+1} + \gamma \max_b Q_t(s_{t+1},b)| \;\le\; R_{\max} + \gamma\,C.$$

This inequality implies that the sample term

$$X_t(s,a) = r_{t+1} + \gamma \max_b Q_t(s_{t+1},b)$$

is uniformly bounded. Since $(TQ_t)(s,a)$ is defined as the conditional expectation of $X_t(s,a)$ given $\mathcal{F}_t$, both $X_t(s,a)$ and $(TQ_t)(s,a)$ are restricted to lie within a finite range. As a result, their difference,

$$w_t(s,a) = X_t(s,a) - (TQ_t)(s,a),$$

is also uniformly bounded. Hence, there exists a finite constant σ^2 such that

$$\mathbb{E}\big[w_t(s,a)^2\,|\,\mathcal{F}_t\big] \le \sigma^2.$$

□

This bounded variance is critical because it guarantees that the stochastic fluctuations (noise) introduced at each update do not dominate the learning process. Consequently, under the Robbins–Monro conditions with diminishing step sizes, the effect of this noise averages out over time, ensuring convergence.

4.9.3.5 Stage 4: Stochastic Approximation Argument

Define $D_t = \|Q_t - Q^*\|_\infty$. The contraction property plus the noise decomposition imply (see equation 4.19)

$$D_{t+1} \;\le\; (1 - \alpha_t(1-\gamma))\,D_t \;+\; \alpha_t\,|w_t|,$$

where w_t is bounded and forms a martingale-difference sequence. By standard results in stochastic approximation (see [64, 65, 63]), if $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$, then $D_t \to 0$ almost surely.

Theorem 1 (Almost-Sure Convergence of Q-Learning)**.** *Under assumptions (A1)–(A3), the sequence* $\{Q_t\}$ *defined by* (4.17) *converges almost surely to* Q^**. Equivalently,* $\|Q_t - Q^*\|_\infty \to 0$ *with probability* 1*.*

Sketch of Proof. By Lemma 1, Q_t remains almost surely bounded. Lemma 2 shows the "noise" term w_t is a martingale difference with bounded variance, which, under the Robbins–Monro conditions (A3), implies that its impact diminishes over time. Meanwhile, Proposition 1 ensures T is a γ-contraction in the sup norm, so the deterministic part of the update pulls Q_t closer to Q^*. Assumption (A1) guarantees each component (s, a) is updated infinitely often. Putting these together in the standard stochastic-approximation recursion (from equation 4.17)

$$Q_{t+1} = Q_t + \underbrace{\alpha_t}_{\substack{\text{Step-size} \\ \text{(diminishing)}}} \Big[\underbrace{(TQ_t - Q_t)}_{\substack{\text{Deterministic component} \\ \text{(contraction term)}}} + \underbrace{w_t}_{\substack{\text{Stochastic noise} \\ \text{(martingale difference)}}} \Big]$$

yields $Q_t \to Q^*$ almost surely. □

Under the standard assumptions of (A1) infinite visits, (A2) bounded rewards, and (A3) proper stepsize decay, the iterates $\{Q_t\}$ converge almost surely to the unique fixed point Q^* of the Bellman optimality operator [59, 65]. This result underlies why Q-Learning remains a cornerstone of reinforcement learning.

4.9.3.6 Asynchronous and Parallel Implementations

Assumption (A1) simply requires that each state-action pair (s, a) is visited infinitely often, in any order. This setup is sometimes called "asynchronous" Q-Learning [60]. Convergence is guaranteed because the contraction $\|TQ - TQ'\|_\infty \leq \gamma \|Q - Q'\|_\infty$ applies componentwise, regardless of the update order or delays.

4.10 Backpropagation

The error minimization presented in Equation 4.20 illustrates the fundamental concept of backpropagation, which serves as the algorithm for updating the learning process in neural networks [66]. In the context of TD learning, the weight update is given by

$$\Delta w = \alpha \, \delta_t \, \nabla_w \hat{V}(s_t), \tag{4.20}$$

where α is the learning rate, $\hat{V}(s_t)$ is the network's current estimate (for example, the value of state s_t), and the TD error δ_t is defined as

$$\delta_t = r_t + \gamma \, \hat{V}(s_{t+1}) - \hat{V}(s_t).$$

This update rule can be derived from the standard TD update for state value functions given in equation 4.15. Assuming that the state value function is approximated by $\hat{V}(s_t, w)$ parameterized by w, we define the TD error as

$$\delta_t = r_{t+1} + \gamma \, \hat{V}(s_{t+1}, w) - \hat{V}(s_t, w).$$

To update the weights w so as to minimize the squared TD error, we can perform gradient descent on the loss function [67]

$$L(w) = \frac{1}{2}\delta_t^2.$$

Taking the gradient with respect to w gives

$$\nabla_w L(w) = \delta_t \, \nabla_w \delta_t.$$

In many TD methods (using a semi-gradient approach), the target $r_{t+1} + \gamma \, \hat{V}(s_{t+1}, w)$ is treated as constant with respect to w. Thus,

$$\nabla_w \delta_t = -\nabla_w \hat{V}(s_t, w),$$

and the gradient descent update becomes

$$\Delta w = -\alpha \, \nabla_w L(w) = \alpha \, \delta_t \, \nabla_w \hat{V}(s_t, w),$$

which is precisely Equation 4.20.

For supervised learning tasks, the error is often quantified using the Root Mean Squared Error (RMSE), defined as

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{j=1}^{N} (o_j - t_j)^2}, \tag{4.21}$$

where o_j represents the output of the jth neuron and t_j the corresponding target value.

A neural network typically is a matrix of weights $w_{i,j,t}$ rather than a simple vector w_t. Figure 4.2 shows an example of such a network; note that a practical network may include many more neurons, layers, and multiple outputs. In Figure 4.2, the connections between the neurons represent the weights $w_{i,j,t}$. Figure 4.2 depicts the mathematical framework of single neuron in Figure 4.2. The output o_j of a neuron is calculated as

$$o_j = \phi(\text{net}_j), \tag{4.22}$$

where ϕ is a differentiable activation function (commonly a sigmoid or ReLU function [66, 68, 69]) whose derivative is nonzero over most of its domain. This function maps the neuron's net input into a manageable range (typically between 0 and 1). The vector $\hat{V} = (o_1, o_2, \ldots, o_n)$ contains all the outputs of the neural network. Figure 4.3 illustrates how the output of a single neuron is formed.

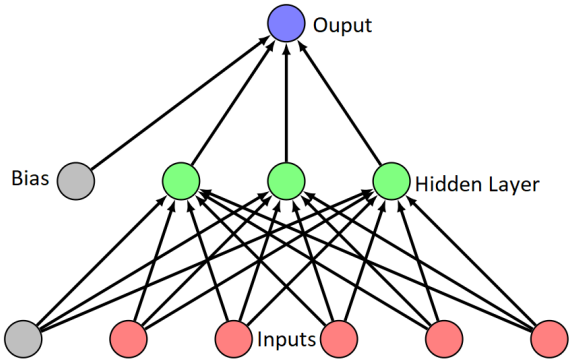


Fig. 4.2: Example of a neural network with input, hidden, and output layers.

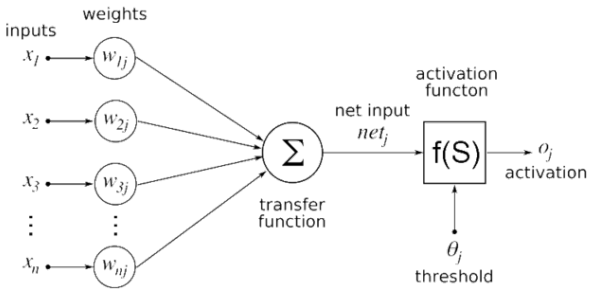


Fig. 4.3: Output of a single neuron.

The net input to a neuron, net_j, is calculated as

$$\text{net}_j = \sum_{i=1}^{n} x_i \, w_{i,j}, \tag{4.23}$$

where x_i denotes the input signals. This summation is performed for each neuron in the network.

The simplest form of the backpropagation update can be written as

$$w_{t+1} = w_t + \Delta w_{i,j}. \tag{4.24}$$

The term $\Delta w_{i,j}$ is computed from the derivative of the error function (for instance, the RMSE in Equation 4.21). By applying the chain rule, we obtain

$$\frac{\partial \text{RMSE}}{\partial w_{i,j}} = \frac{\partial \text{RMSE}}{\partial o_j} \cdot \frac{\partial o_j}{\partial \text{net}_j} \cdot \frac{\partial \text{net}_j}{\partial w_{i,j}}. \tag{4.25}$$

Due to the interdependence of neurons (since the output of one layer serves as the input to the next), the derivative in Equation 4.25 is split into parts. If a neuron resides in the output layer, the weight update can be computed directly; otherwise, it must be determined indirectly. The weight update is given by

$$\Delta w_{i,j} = -\alpha \frac{\partial \text{RMSE}}{\partial w_{i,j}} = -\alpha \, \delta_j \, o_i, \tag{4.26}$$

with

$$\delta_j = \begin{cases} \phi'(\text{net}_j) \, (o_j - t_j) & \text{if } j \text{ is an output neuron,} \\ \phi'(\text{net}_j) \sum_k \delta_k \, w_{j,k} & \text{if } j \text{ is not an output neuron.} \end{cases} \tag{4.27}$$

Here, t_j is the target value for the jth neuron, and the vector $G_t = (t_1, t_2, \ldots, t_n)$ represents the target outputs. Notice the strong similarity between Equation 4.27 and the TD gradient update in Equation 4.20.

It should be noted that, depending on the application domain, different error functions may be used, which in turn alter the weight update $\Delta w_{i,j}$. Likewise, the choice of activation function influences the form of Equation 4.27.

In many machine learning libraries, the error function is referred to as the *criterion* or simply the *loss*. The activation function is typically applied in the output layer of the neural network.

4.11 Actor-Critic Methods

Actor-Critic methods are a class of TD algorithms that explicitly separate the representation of the policy from that of the value function. In these methods, the *Actor* is responsible for selecting actions by maintaining a parameterized policy, while the *Critic* evaluates the actions taken by estimating the value of states (or state-action pairs). Rather than using action-values directly for decision making, the policy is adjusted based on feedback from the Critic [70, 71].

The Critic evaluates the current state (or the resulting state after an action) and computes a TD error—a scalar signal that reflects the discrepancy between the predicted and the observed outcomes. This TD error is analogous to Equation 4.15 and serves as the central feedback signal for both components of the algorithm. Specifically, the Critic uses the TD error to update its value estimates, and the Actor uses the same signal to modify the policy parameters in a direction that is expected to improve performance.

Typically, the Critic is implemented as a state-value function. After each action, the Critic evaluates the new state to determine whether the state value has increased or decreased, producing a TD error that is then used to update both the Actor and the Critic. More recent advancements have applied Actor-Critic architectures in deep reinforcement learning contexts.

$$\delta_t = r_{t+1} + \gamma V_t(s_{t+1}) - V(s_t), \tag{4.28}$$

where V_t is the value function implemented by the Critic at time t. In many variations of the Actor-Critic algorithm, instead of the state-value function $V_t(s_t)$, the action-value function $Q_t(s_t, a_t)$ is used to compute δ_t. This TD error can be used to evaluate the chosen action a_t in state s. If the TD error is positive, the tendency to choose action a_t should be strengthened; if negative, it should be weakened. This can be achieved using the *Gibbs Softmax* method [19], which is defined as:

$$\pi_t(a|s) = \Pr\{a_t = a | s_t = s\} = \frac{e^{H_t(s,a)}}{\sum_b e^{H_t(s,b)}}, \tag{4.29}$$

where $H_t(s, a)$ represents the modifiable policy parameters of the Actor, indicating the propensity to select each action a in each state s at time t. H corresponds to a vector containing all output values of the Actor's neural network, determined by the network's weights θ. By increasing or

decreasing $H_t(s, a_t)$, the propensity to select an action can be altered. This is typically achieved by adjusting the weights θ. Equation 4.30 shows how this update is performed:

$$\theta_{t+1} = \theta_t + \alpha_\theta Q(s_t, a_t) \nabla_\theta \ln \pi_\theta(s_t, a_t), \tag{4.30}$$

where α_θ is a step size parameter. This adjustment is analogous to the backpropagation described by Equation 4.24.

The policy parameters θ are initialized randomly, causing the Actor to execute random actions during the initial phase.

With this information, a pseudocode for the Actor-Critic algorithm (Algorithm 4) can be constructed.

Algorithm 4 The pseudocode of the Actor-Critic algorithm from [19] p.155. Initially, all parameters are initialized. At each time step, an action is chosen according to the current policy, followed by sampling the reward and next state. The policy parameters θ and action-value function parameters w are updated based on the TD error. These steps are repeated until the terminal state is reached.

Initialize s, θ, w at random; sample $a \sim \pi_\theta(a|s)$.
for $t = 1, \ldots, T$ **do**
  Sample reward $r_t \sim R(s, a)$ and next state $s' \sim P(s'|s, a)$;
  Then sample the next action $a' \sim \pi_\theta(a'|s')$;
  Update the policy parameters: $\theta \leftarrow \theta + \alpha_\theta Q_w(s, a) \nabla_\theta \ln \pi_\theta(a|s)$;
  Compute the TD error for action-value at time t: $\delta_t = r_t + \gamma Q_w(s', a') - Q_w(s, a)$;
  Use it to update the parameters of the Q-function: $w \leftarrow w + \alpha_w \delta_t \nabla_w Q_w(s, a)$;
  Update $a \leftarrow a'$ and $s \leftarrow s'$.
end for

4.12 Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) extends the Actor-Critic approach discussed in Section 4.11 by using deep neural networks to represent policies and value functions. This setup allows learning in high-dimensional or continuous state spaces frequently encountered in real-world domains such as sepsis management.

As outlined earlier, Actor-Critic methods split the policy representation (Actor) from the value function (Critic). The Critic estimates a state-value function $V_w(s)$ (see equation 4.13) or action-value function $Q_w(s, a)$ (see equation 4.8), while the Actor maintains a parameterized policy $\pi_\theta(a \mid s)$. The central feedback signal is the TD error (see equation 4.28), which measures how much better or worse the outcome is compared to the Critic's prediction. The Critic uses δ_t to refine its value estimates, and the Actor adjusts its policy parameters in a direction that ideally improves future performance.

4.12.1 Extending to Deep Learning

When the Actor and Critic are approximated by deep neural networks, they become:

$$\pi_\theta \quad \text{(parameterized by weights } \theta\text{)}, \quad V_w \quad \text{(parameterized by weights } w\text{)}.$$

These networks can manage state representations far larger than traditional tabular methods could handle. Hence, DRL introduces powerful function approximators for both policy and value functions. The goal is to learn π_θ that maximizes the expected discounted return (equation 4.3):

$$J(\theta) \;=\; \mathbb{E}_{\tau\sim\pi_\theta}\Big[\sum_{t=0}^{\infty}\gamma^t\, r(s_t,a_t)\Big],$$

where $\tau = (s_0, a_0, s_1, a_1, \dots)$ is a trajectory of states and actions drawn from π_θ.

Policy gradient methods optimize θ by following the gradient

$$\nabla_\theta\, J(\theta) \;=\; \mathbb{E}_{(s_t,a_t)\sim\pi_\theta}\Big[\nabla_\theta\, \ln \pi_\theta(a_t \mid s_t)\;\cdot\; Q_\pi(s_t,a_t)\Big].$$

In practice, Q_π is often replaced by an *Advantage* function $A_\pi(s_t,a_t) = Q_\pi(s_t,a_t) - V_\pi(s_t)$ to reduce variance (Advantage Actor Critic (A2C)). An unbiased estimator of A_π is the TD error δ_t, when $V_w(s)$ approximates the true $V_\pi(s)$. The Actor update minimizes

$$L_{\text{actor}}(\theta) \;=\; -\,\mathbb{E}\big[\delta_t\, \ln \pi_\theta(a_t \mid s_t)\big],$$

encouraging policy parameters to increase the probability of actions with positive TD error and decrease it for those with negative error. Often, an entropy term $\beta\, H(\pi_\theta(\cdot \mid s_t))$ is added to encourage exploration.

The Critic aims to approximate $V_\pi(s)$. A common strategy is minimizing the mean-squared TD error:

$$L_{\text{critic}}(w) \;=\; \mathbb{E}\big[(r_t + \gamma\, V_w(s_{t+1}) - V_w(s_t))^2\big].$$

Improving V_w yields a more accurate evaluation and better policy updates.

4.12.2 Application to Corticosteroid Optimization

In the sepsis-management study (Chapter 5), we adopt an A2C framework to optimize daily corticosteroid dosing for ICU patients. We define an MDP where the state s_t is a 379-dimensional vector of clinical features, the action a_t is one of five discrete steroid doses, and the reward r_t is terminal ($r = 0$ for survival, $r = 1$ for survival). The Actor $\pi_\theta(a \mid s)$ outputs dose probabilities, and the Critic $V_w(s)$ tracks expected future return. Algorithm 5 outlines the training procedure, which repeats sampling transitions, computing TD errors, and applying gradient-based updates to both networks.

Algorithm 5 Deep A2C Algorithm for Sepsis Management

1: Initialize Actor network π_θ and Critic network V_w.
2: Initialize replay buffer D with ICU patient trajectories.
3: **for** epoch = 1 to N **do**
4: Sample a batch $\{(s_t, a_t, r_t, s_{t+1})\}$ from D.
5: Compute $V_{\text{target}}(s_t) = r_t + \gamma\, V_w(s_{t+1})$.
6: Compute $\delta_t = V_{\text{target}}(s_t) - V_w(s_t)$.
7: Update Critic: $w \leftarrow w - \alpha_w\, \nabla_w\, L_{\text{critic}}(w)$.
8: Update Actor: $\theta \leftarrow \theta - \alpha_\theta\, \nabla_\theta\, L_{\text{actor}}(\theta)$.
9: **end for**

4.13 On-Policy vs. Off-Policy Evaluation

A central question in RL is *policy evaluation*, which involves estimating how well a given decision-making strategy (i.e., a policy π) will perform. There are two main approaches to this:

- **On-Policy Evaluation:** In this approach, the agent uses the same policy both to generate data and to evaluate its performance. This is analogous to taking a test using the same method you practiced with. Although the data and evaluation policy are identical, this method might not be efficient when exploring new strategies.
- **Off-Policy Evaluation:** Here, the goal is to estimate the performance of an *evaluation policy* (a new or improved strategy) using data that was collected by a single or multiple, potentially different, *behavior policy*. This is similar to predicting how well a new recipe might work by tasting meals prepared by another recipe. Off-policy evaluation is particularly important in real-world applications (such as healthcare or robotics) where experimenting directly with a new policy could be risky or expensive.

In what follows, we focus on off-policy evaluation and explore several methods that help estimate the effectiveness of an evaluation policy without requiring direct experimentation in the environment.

4.14 The Off-Policy Evaluation Problem

For a given trajectory of states and actions,

$$\tau = (s_0, a_0, s_1, a_1, \dots),$$

the *discounted policy value* (often denoted by ρ^π) can be defined as

$$\rho^\pi = (1-\gamma)\,\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t, a_t)\right], \tag{4.31}$$

which represents the expected performance of the evaluation policy when starting from the initial state distribution. (Note that this is distinct from the state value function $V^\pi(s)$, which is defined conditionally on $s_0 = s$.) In off-policy evaluation, our goal is to compute ρ^π using data generated by one or more behavior policies. Because the data originates from these behavior policies rather than from π, there is a distribution mismatch; the states and actions observed may not reflect those that would have occurred under the evaluation policy. To adjust for this discrepancy, it is necessary to reweight the observed data appropriately, effectively "translating" the behavior-policy data into an estimate of what the evaluation policy would have produced [72, 73].

4.15 Importance Sampling (IS) and Bias Properties

Importance Sampling (IS) is one of the simplest methods to address the off-policy evaluation problem [72]. It adjusts for the difference between the behavior and evaluation policies by applying a weight to each trajectory. For a given trajectory

$$\tau \;=\; (s_0, a_0, s_1, a_1, \dots, s_{T-1}, a_{T-1}),$$

the importance weight is computed as:

$$\rho(\tau) \;=\; \frac{\prod_{t=0}^{T-1} \pi(a_t \mid s_t)}{\prod_{t=0}^{T-1} \pi_b(a_t \mid s_t)},$$

where the numerator is the probability of the trajectory under the evaluation policy π, and the denominator is the probability of the same trajectory under the behavior policy π_b. Many terms typically cancel in this product due to Markov assumptions.

The Simple Importance Sampling (SIS) estimator for the policy value is then given by:

$$\hat{\rho}_{\text{SIS}} \;=\; (1-\gamma)\,\frac{1}{N}\sum_{i=1}^{N} \rho(\tau_i)\, G(\tau_i),$$

where $G(\tau_i)$ is the total (discounted) reward accumulated along the ith trajectory, N is the total number of trajectories, and γ is the discount factor. By the Law of Large Numbers, $\hat{\rho}_{\text{SIS}}$ is a consistent estimator of the true policy value $\rho(\pi)$. However, SIS can suffer from high variance in practice due to the so-called "curse of the horizon," where long trajectories cause the product of probabilities $\rho(\tau)$ to become extremely small or large. In some cases, the variance can even be unbounded [19].

SIS is *unbiased* provided the evaluation policy π has support over all actions in the dataset. Formally, if $\pi(a \mid s) > 0$ whenever the behavior policy $\pi_b(a \mid s) > 0$, then the expectation of each IS weight $\rho(\tau_i)$ is one, implying

$$\mathbb{E}\big[\rho(\tau_i)\, G(\tau_i)\big] \;=\; \rho(\pi) \quad \text{(unbiased estimator)}.$$

A practical approach to reduce the high variance of SIS is *Weighted Importance Sampling* (WIS) [72, 73]. WIS normalizes the importance weights, thus lowering variance at the expense of introducing a small, diminishing bias:

$$\hat{\rho}_{\text{WIS}} \;=\; (1-\gamma)\,\frac{\sum_{i=1}^{N} \rho(\tau_i)\, G(\tau_i)}{\sum_{i=1}^{N} \rho(\tau_i)}.$$

A key property is that the expected value of the IS weights is 1. This implies that if we denote the numerator as an estimator $\hat{a}$ of the true policy value a, and the denominator as an estimator $\hat{o}$ of 1, then by Slutsky's theorem, the ratio $\hat{a}/\hat{o}$ converges to a as the sample size increases. Thus, WIS is also a *consistent* estimator whose normalization ensures lower variance compared to SIS. Meanwhile, the finite-sample bias introduced by normalizing the weights vanishes with more data.

4.16 High Confidence Off-Policy Evaluation (HCOPE)

In many medical applications, the stakes for patient outcomes are extremely high, and deploying a reinforcement learning policy without stringent performance guarantees can be unsafe. *High-Confidence Off-Policy Evaluation* (HCOPE) [74] mitigates this risk by providing a statistically rigorous lower bound on the policy's expected return—even when the evaluation policy differs from the behavior policies used to collect data. Specifically, HCOPE ensures that, with probability at least $1-\delta$, the true performance of a proposed treatment or intervention policy exceeds the computed lower bound $\rho_{\text{HCOPE}-}$. This guarantee is critical in medical settings, where verifying that a new policy meets a minimum performance standard can help protect patients and build clinical confidence before real-world implementation.

4.16.1 Problem Setting and Notation

Let $\{\pi_b^i\}_{i=1}^n$ be behavior policies that generated trajectories $\{\tau_i\}_{i=1}^n$ with a bounded reward function . We wish to evaluate a different policy π_e, referred to as the *evaluation policy*. Our objective is to produce a high-confidence lower bound on $\rho(\pi_e)$, the (expected) discounted return of π_e.

4.16.2 Mathematical Formalization

Definition 1 (Importance Weighted Return)**.** *For a trajectory $\tau = (s_1, a_1, r_1, \ldots, s_T, a_T, r_T)$ generated by π_b, the importance weighted return is (see section 4.15):*

$$\hat{\rho}(\pi_e, \tau, \pi_b) := R(\tau) \prod_{t=1}^{T} \frac{\pi_e(a_t \mid s_t)}{\pi_b(a_t \mid s_t)}, \tag{4.32}$$

where

$$R(\tau) = \frac{\sum_{t=1}^{T} \gamma^{t-1} r_t - R_-}{R_+ - R_-} \quad \textit{and} \quad R_\pm = \frac{r_\pm(1-\gamma^T)}{1-\gamma}.$$

The quantity $R(\tau)$ is simply a normalized return in $[0,1]$ based on the accumulated discounted rewards in τ. This normalization step ensures better numerical stability and makes the bounding analysis more straightforward.

Given these normalized returns, define $\{X_i\}_{i=1}^n$ as independent random variables with

$$X_i = \hat{\rho}(\pi_e, \tau_i, \pi_b^i).$$

If we assume $\text{supp}(\pi_e) \subseteq \text{supp}(\pi_b^i)$ for all i (the coverage condition), we have $\mathbb{E}[X_i] = \rho(\pi_e)$. In other words, under coverage, the random variables X_i are unbiased estimators of $\rho(\pi_e)$.

4.16.3 Truncated Empirical Bernstein Inequality

HCOPE's fundamental tool is a carefully crafted *empirical Bernstein bound* that accounts for possibly heavy-tailed data via *truncation*. This truncation step ensures that excessively large random variables do not destroy the concentration inequality, which is critical in finite-sample regimes [74].

Theorem 2 (Truncated Empirical Bernstein Inequality)**.** *Let $X_1, \ldots, X_n$ be non-negative independent random variables with $\mathbb{E}[X_i] \le \mu$ for all i. For thresholds $\{c_i\}_{i=1}^n > 0$ and truncated variables $Y_i := \min(X_i, c_i)$, it holds with probability at least $1-\delta$ that*

$$\mu \;\ge\; \left(\sum_{i=1}^{n} \frac{1}{c_i}\right)^{-1} \left(\sum_{i=1}^{n} \frac{Y_i}{c_i} \;-\; \frac{7\ln(\frac{2}{\delta})}{3(n-1)} \;-\; \sqrt{\frac{2\ln(\frac{2}{\delta})}{n-1}\Big(\sum_{i,j=1}^{n} \frac{Y_i}{c_i} - \frac{Y_j}{c_j}\Big)^2}\right). \tag{4.33}$$

Proof Sketch 1. *The proof builds on the* empirical Bernstein bound *proposed by Maurer and Pontil [75]. Set $Z_i = Y_i/c_i$ so that $Z_i \in [0,1]$. Applying an empirical Bernstein-type inequality to $1 - Z_i$ yields a lower bound on $\mathbb{E}[Z]$. Since $\mathbb{E}[Z] \le \frac{\mu}{n}\sum_{i=1}^n \frac{1}{c_i}$ (because $\mathbb{E}[X_i] \le \mu$ and each $Y_i \le X_i$), we rearrange to solve for μ. Truncation is necessary to control the variance and ensure we do not rely on overly conservative assumptions for heavy-tailed data. A more rigorous measure-theoretic argument shows that limiting the domain of X_i to $[0, c_i]$ preserves concentration properties while capping extreme outcomes.*

4.16.4 Optimal Threshold Selection

To apply Theorem 2 practically, we choose a single threshold c^* (or multiple thresholds in some derivations) that controls the trade-off between *variance reduction* and *bias*. If the threshold is too small, we might truncate away significant parts of the distribution and underestimate $\rho(\pi_e)$. If the threshold is too large, the variance can blow up.

In HCOPE, the typical approach is to partition the data $\mathcal{D} = \{\tau_i, \pi_b^i\}$ into two sets:

$$\mathcal{D}_{\text{pre}} \quad (\text{size } n_{\text{pre}}) \quad \text{and} \quad \mathcal{D}_{\text{post}} \quad (\text{size } n_{\text{post}}).$$

The pre-partition $\mathcal{D}_{\text{pre}}$ is used to find an "optimal" threshold c^* via a search (often gradient-based). The post-partition $\mathcal{D}_{\text{post}}$ is then used to finalize the estimate of $\rho(\pi_e)$ (with high confidence).

Formally, we define:

$$Y(c) = \min\big(\hat{\rho}(\pi_e, \tau, \pi_b),\, c\big).$$

We then choose

$$c^* \;=\; \arg\max_{c \geq 1} \Bigg(\underbrace{\frac{1}{n_{\text{pre}}} \sum_{\tau \in \mathcal{D}_{\text{pre}}} Y(c)}_{\text{Truncated Mean}} \;-\; \underbrace{\frac{7c\,\ln(2/\delta)}{3\,n_{\text{post}}}}_{\text{Linear Penalty}} \;-\; \underbrace{\sqrt{\frac{2\,\ln(2/\delta)}{n_{\text{post}}}\,\widehat{\text{Var}}_{\mathcal{D}_{\text{pre}}}\big[Y(c)\big]}}_{\text{Variance Term}} \Bigg). \tag{4.34}$$

Choosing c in this manner reflects the fact that larger thresholds increase the penalty term in the empirical Bernstein bound; however, too small a threshold might clip important information. This optimization is usually solved numerically (e.g., grid search or gradient-based methods) for practical implementations.

4.16.5 Theoretical Guarantees

When the coverage assumption $\pi_e(a \mid s) > 0 \implies \pi_b^i(a \mid s) > 0$ for all i holds, importance sampling ensures that:

$$\mathbb{E}_{\tau \sim \pi_b^i}\big[\hat{\rho}(\pi_e, \tau, \pi_b^i)\big] \;=\; \rho(\pi_e).$$

Hence, in principle, one can recover the *true* return of π_e if enough samples and suitable bounding methods are available.

Theorem 3 (Unbiasedness Under Coverage)**.** *If $\pi_e(a \mid s) > 0 \implies \pi_b^i(a \mid s) > 0\ \forall (s,a), i$, then for all i:*

$$\mathbb{E}_{\tau \sim \pi_b^i}\big[\hat{\rho}(\pi_e, \tau, \pi_b^i)\big] = \rho(\pi_e).$$

As the number of samples n grows large, the HCOPE bounds tighten around the true value $\rho(\pi_e)$. In particular, if the threshold c^* grows more slowly than some polynomial rate (to ensure it effectively becomes large), one recovers asymptotic consistency at the standard $\mathcal{O}_p(\sqrt{\text{Var}(\hat{\rho})/n})$ rate typical of importance sampling estimators.

Theorem 4 (Asymptotic Consistency)**.** *As the number of samples n tends to infinity and the truncation threshold c^* grows unbounded (i.e. $c^* = \omega(1)$), the HCOPE lower bound satisfies*

$$\rho_- = \rho(\pi_e) - \mathcal{O}_p\Big(\sqrt{\frac{\text{Var}(\hat{\rho})}{n}}\Big).$$

In simpler terms, the difference between the HCOPE bound and the true expected return $\rho(\pi_e)$ decreases at the rate of $1/\sqrt{n}$.

Proof. The key ideas are as follows:

1. As c^* increases with more data, the truncated variable

$$Y_i := \min(X_i, c^*)$$

converges to the original importance sampling estimator X_i, since it becomes unlikely that X_i exceeds the large threshold c^*.

2. With the truncation effect diminishing, the empirical variance computed in the Bernstein inequality approaches the true variance $\mathrm{Var}(\hat{\rho})$.

3. By applying the Central Limit Theorem, the sample average of the X_i (or Y_i) converges to $\rho(\pi_e)$ at a rate proportional to $1/\sqrt{n}$. At the same time, the extra penalty terms in the bound shrink at this same rate.

Thus, the overall error in the HCOPE lower bound decreases as $\mathcal{O}_p\Big(\sqrt{\mathrm{Var}(\hat{\rho})/n}\Big)$, proving asymptotic consistency. For a detailed derivation, see Thomas et al. (2015) [74]. □

Theorem 5 (Finite-Sample Validity)**.** *For any $\delta \in (0, 1)$ and arbitrary behavior policies, the HCOPE bound satisfies:*

$$\mathbb{P}\Big(\rho(\pi_e) \geq \rho_-\Big) \geq 1 - \delta.$$

Proof. The high-level argument follows directly from Theorem 2, applied to the truncated samples in the post-partition. Since truncation only reduces each X_i to Y_i, it can never increase the mean. Thus, the one-sided (lower) bound on $\rho(\pi_e)$ holds in finite samples with probability $1 - \delta$. For a full measure-theoretic argument, see Appendix B of Thomas et al. (2015) [74]. □

This means that the evaluation policy is not required to be completely covered by the behavior policies; lacking full coverage only affects the magnitude of $\mathbb{E}[Y_i]$ and not the validity of the concentration argument.

4.16.6 Algorithm Specification

Below we summarize the entire procedure in pseudocode. The idea is to hold out a small portion of data to *learn* an appropriate threshold for truncation, and then use the remaining data to compute the final lower-confidence bound.

Algorithm 6 HCOPE (Thomas et al., 2015)

Require: Dataset $\mathcal{D} = \{\tau_i, \pi_b^i\}_{i=1}^n$, confidence level $\delta \in (0,1)$, evaluation policy π_e

1: **Partition** $\mathcal{D}$ into $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{post}}$ with $|\mathcal{D}_{\text{pre}}| = \lfloor 0.05\, n \rfloor$ (typically a small fraction)

2: **for** $\tau_i \in \mathcal{D}_{\text{pre}}$ **do**

3: Compute

$$X_i = R(\tau_i) \prod_{t=1}^{T} \frac{\pi_e(a_t^i \mid s_t^i)}{\pi_b^i(a_t^i \mid s_t^i)}.$$

4: **end for**

5: Optimize c^* via

$$c^* \;=\; \arg\min_{c \geq 1} \left(-\frac{1}{|\mathcal{D}_{\text{pre}}|} \sum_{X_i \in \mathcal{D}_{\text{pre}}} \min(X_i, c) + \frac{7\, c\, \ln(2/\delta)}{3\, |\mathcal{D}_{\text{post}}|} + \sqrt{\frac{2\, \ln(2/\delta)}{|\mathcal{D}_{\text{post}}|}\, \widehat{\text{Var}}(\min(X_i, c))} \right).$$

6: **for** $\tau_j \in \mathcal{D}_{\text{post}}$ **do**

7: Compute

$$Y_j = \min\Big(R(\tau_j) \prod_{t=1}^{T} \frac{\pi_e(a_t^j \mid s_t^j)}{\pi_b^j(a_t^j \mid s_t^j)},\, c^* \Big).$$

8: **end for**

9: Compute the weighted mean and variance:

$$\bar{Y} \;=\; \frac{\sum_{j=1}^{|\mathcal{D}_{\text{post}}|} \frac{Y_j}{c^*}}{\sum_{j=1}^{|\mathcal{D}_{\text{post}}|} \frac{1}{c^*}}, \quad S^2 \;=\; \frac{|\mathcal{D}_{\text{post}}| \, \sum \left(\frac{Y_j}{c^*}\right)^2 \;-\; \left(\sum \frac{Y_j}{c^*}\right)^2}{|\mathcal{D}_{\text{post}}| \;-\; 1}.$$

10: Return the final lower bound

$$\rho_- \;=\; \bar{Y} \;-\; \frac{7\, \ln(2/\delta)}{3\, (|\mathcal{D}_{\text{post}}| \;-\; 1)\, \sum \frac{1}{c^*}} \;-\; \sqrt{\frac{2\, \ln(2/\delta)\, S^2}{|\mathcal{D}_{\text{post}}| \;-\; 1}}.$$

This procedure ensures that the threshold c^* is adapted to the distribution of the sampled returns and the variability present in $\mathcal{D}_{\text{pre}}$. By decoupling threshold selection from final evaluation, we avoid "double-dipping" with the same data, thus preserving the validity of the resulting confidence bound.

4.16.7 Remarks on Theoretical Limits

The HCOPE bounds are known to be nearly minimax-optimal for heavy-tailed distributions, up to constants (Maurer and Pontil, 2009) [75]. This means that, in principle, we cannot hope for better asymptotic scaling than that achieved by HCOPE once we allow for possible large outliers.

One often-cited practical concern is the coverage condition. If there exist state-action pairs that π_e takes with positive probability but none of the behavior policies ever take, standard importance sampling ideas can fail catastrophically. However, HCOPE's truncated approach can still provide meaningful bounds on a *restricted* version of $\rho(\pi_e)$ ignoring those uncovered actions, because truncation prevents infinite or undefined importance weights from destroying the estimate.

Theorem 6 (Minimax Lower Bound)**.** *For any $\delta < 1/2$, there exists a distribution of X_i with $\mathrm{Var}(X_i) \le \sigma^2$ such that all valid lower bounds must satisfy:*

$$\rho(\pi_e) - \rho_- \;\ge\; \Omega\Big(\sqrt{\tfrac{\sigma^2 \ln(1/\delta)}{n}}\Big).$$

This result tells us that no method can fundamentally beat the $\sqrt{1/n}$ scaling (up to logarithmic factors) in the general case, demonstrating the inherent difficulty of off-policy evaluation with finite samples.

4.17 Dual Stationary Distribution Correction Estimation (DICE)

Dual Stationary Distribution Correction Estimation (DICE) is a modern method for off-policy evaluation that addresses some inherent challenges of traditional IS approaches [76, 73]. Standard IS techniques can suffer from high variance, particularly in long-horizon tasks, because they involve products of probabilities over full trajectories[72]. In contrast, DICE focuses on estimating a *state-action-level correction factor* that directly accounts for discrepancies between the evaluation policy and the data distribution. In many real-world settings, including the medical application of reinforcement learning, we often treat episodes as potentially unbounded in time, effectively leading to an infinite-horizon setting. This perspective justifies the stationarity assumption that underlies DICE.

4.17.1 Distribution Correction Estimation

For a policy π and discount factor $0 < \gamma < 1$, the *stationary distribution* $d^\pi(s,a)$ is given by

$$d^\pi(s,a) \;=\; (1-\gamma)\sum_{t=0}^{\infty} \mathbb{P}(s_t = s,\; a_t = a)\,\gamma^t, \tag{4.35}$$

where $s_0 \sim d_0$, $a_t \sim \pi(\cdot \mid s_t)$, $s_{t+1} \sim T(s_t, a_t)$. In words, $d^\pi(s,a)$ captures how frequently state-action pairs are visited under policy π. Here, $\mathbb{P}(\cdot)$ denotes the probability measure induced with initial state distribution d_0, policy π, and transition kernel T. The *policy value* for infinite-horizon tasks is typically written as (see equation 4.9)

$$\begin{aligned}\rho^\pi &= (1-\gamma)\,\mathbb{E}_{\tau\sim\pi}\Big[\sum_{t=0}^{\infty}\gamma^t\, r(s_t,a_t)\Big],\\ \rho^\pi &= \mathbb{E}_{(s,a)\sim d^\pi}\big[r(s,a)\big],\end{aligned} \tag{4.36}$$

An elementary result is that the *stationary distribution* d^π (discounted or undiscounted) is a fixed point of the corresponding backward Bellman operator, as described below.

We assume the standard setting in which we have a dataset $\mathcal{D}$

$$\mathcal{D} \;=\; \big((s_{0,i},\, s_i,\, a_i,\, r_i,\, s_i')\big)_{i=1}^{n}$$

(where each tuple may repeat states or actions already seen). Typically, $s_{0,i} \sim d_0$ represents the initial state for the i-th trajectory or episode, (s_i, a_i) is drawn according to some mixture of behavior policies, $r_i \sim R(s_i, a_i, s_i')$, and $s_i' \sim T(s_i, a_i)$. We want to evaluate a new policy π by using this dataset. In practice, one must ensure that every (s,a) for which $\pi(a \mid s) > 0$ is also adequately represented in $\mathcal{D}$.

4.17.2 Key Idea of DICE

The DICE family of methods [76, 77] centers on estimating a *state-action-level correction factor*

$$w_{\pi/D}(s,a) \;=\; \frac{d^{\pi}(s,a)}{d^{\mathcal{D}}(s,a)},$$

often called the *stationary distribution correction*. Once this function $w_{\pi/D}(s,a)$ is estimated, we can approximate the policy value by

$$\rho^{\pi}_{\mathrm{DICE}} \;=\; \mathbb{E}_{(s,a,r)\sim\mathcal{D}}\big[w_{\pi/D}(s,a)\cdot r\big].$$

A crucial coverage assumption is that π does not assign nonzero probability to any (s,a) lying outside the support of the dataset distribution $d^{\mathcal{D}}$. Formally, if $d^{\mathcal{D}}(s,a)=0$, then we require $d^{\pi}(s,a)=0$. Otherwise, the ratio

$$w_{\pi/D}(s,a) \;=\; \frac{d^{\pi}(s,a)}{d^{\mathcal{D}}(s,a)}$$

would not be well-defined.

A key feature of DICE is that it does *not* require access to the behavior policy; instead, it relies solely on dataset quintuplets $((s_0,\, s,\, a,\, r,\, s'))$ and knowledge of the evaluation policy π.

In principle, one may recognize the ratio $w_{\pi/D}(s,a)$ as playing a similar role to an IS weight, since it compares how frequently (s,a) arises under π vs. under $\mathcal{D}$. Traditional IS approaches, however, can suffer from high variance in long-horizon problems and require knowledge of the behavior policy. DICE mitigates these drawbacks by focusing on directly solving for the stationary correction factor $w_{\pi/D}(s,a)$ through Bellman-based objectives rather than explicit trajectory-level likelihood ratios.

The idea behind this is similar to the PageRank Algorithm [78].

4.17.3 Bellman Equations and Stationary Distributions

To derive DICE, it is helpful to understand the forward and backward Bellman operators. Recall that for a policy π and discount factor $\gamma\in(0,1)$, the *forward Bellman operator* $\mathcal{B}^{\pi}$ acts on a function $Q\colon S\times A\to\mathbb{R}$ as

$$\mathcal{B}^{\pi}Q \;=\; r \;+\; \gamma\,\mathcal{P}^{\pi}Q,$$

where $r(s,a)$ is the immediate reward function, and $\mathcal{P}^{\pi}$ (often called the *expected Bellman operator*) is given by

$$(\mathcal{P}^{\pi}Q)(s,a) \;=\; \mathbb{E}_{s'\sim T(s,a),\; a'\sim\pi(s')}\Big[\,Q(s',a')\Big].$$

Here, $T(s,a)$ is the transition kernel (i.e., $s'\sim T(s,a)$), and $\pi(s')$ is the policy's distribution over actions at state s'. An elementary result is the Bellman equations, which is a linear equation system which claims that the state-action value function $Q^{p}i$ is a fixed point of the Bellman operator

$$Q^{\pi} \;=\; \mathcal{B}^{\pi}Q^{\pi} \quad\Longleftrightarrow\quad Q^{\pi}(s,a) \;=\; r(s,a) \;+\; \gamma\,\mathbb{E}_{s',a'}\big[Q^{\pi}(s',a')\big],$$

which is the standard *forward Bellman equation* for the state–action value function Q^{π}.

The *backward Bellman operator* $\mathcal{T}^\pi$ instead acts on a measure or distribution over (s, a). Specifically, if d is any distribution over $S \times A$, then

$$\mathcal{T}^\pi d \;=\; (1-\gamma)\,(d_0 \times \pi) \;+\; \gamma\,\mathcal{P}_*^\pi\, d,$$

where d_0 is the initial state distribution, and $\mathcal{P}_*^\pi$ is the *adjoint operator* of $\mathcal{P}^\pi$. Concretely, $\mathcal{P}_*^\pi$ "pushes forward" a distribution d by the transition dynamics T and the policy π. One may think of $\mathcal{T}^\pi d$ as describing how the distribution d evolves under a single *backward* Bellman update step.

The expression $(d_0 \times \pi)$ refers to the joint distribution over (s, a) obtained by first sampling a state s from the initial state distribution d_0 and then drawing an action a according to the policy $\pi(\cdot \mid s)$. Formally, for any measurable set $B \subseteq S \times A$,

$$(d_0 \times \pi)(B) \;=\; \int_s d_0(s) \int_a \mathbb{1}\{(s,a) \in B\}\, \pi(a \mid s)\, \mathrm{d}a\, \mathrm{d}s.$$

The *discounted stationary distribution* d^π under π (with discount γ) is a distribution over (s, a) that is fixed by $\mathcal{T}^\pi$. Formally,

$$d^\pi \;=\; \mathcal{T}^\pi\, d^\pi.$$

In words, d^π remains unchanged (stationary) under the backward Bellman operator.

If we denote by $d^{\mathcal{D}}$ the empirical (or nominal) distribution of state–action pairs from a dataset $\mathcal{D}$, and let $w(s, a) \;=\; \frac{d^\pi(s,a)}{d^{\mathcal{D}}(s,a)}$ be the ratio between the stationary distribution under π and the dataset distribution, then the stationarity condition implies

$$d^{\mathcal{D}}(s,a)\, w(s,a) \;=\; \mathcal{T}^\pi\big(d^{\mathcal{D}}\, w\big)(s,a), \quad \mathbb{E}_{(s,a)\sim d^{\mathcal{D}}}\big[w(s,a)\big] \;=\; 1. \tag{4.37}$$

The family of DICE algorithms aims to solve for w that satisfies these constraints in practice, thereby recovering the *correct* stationary ratio $w_{\pi/D}(s, a)$. With w in hand, one can perform off-policy evaluation by weighting observed rewards in $\mathcal{D}$ according to w.

4.18 Tabular DICE

In the tabular setting ($S \times A$ finite), we can solve for $\hat{w}$ directly via linear algebra or an eigenvalue problem. Observe that for a discount factor $0 < \gamma < 1$, the stationarity condition implies

$$(I - \gamma\,\mathcal{P}_*^\pi)\, D^{\mathcal{D}}\, w_{\pi/D} \;=\; (1-\gamma)\,(d_0 \times \pi),$$

where $D^{\mathcal{D}} \;=\; \mathrm{diag}(d^{\mathcal{D}}(s,a))$. By replacing the unknown distributions and transitions with empirical estimates, one obtains an approximate ratio $\hat{w}$. Algorithm 7 shows the step-by-step procedure for calculating the policy value.

Algorithm 7 Tabular DICE

Input: Dataset $\mathcal{D} = \left((s_{0,i},\, s_i,\, a_i,\, r_i,\, s_i')\right)_{i=1}^n$, discount factor $\gamma \in (0, 1]$, policy π.
Estimate empirical counts: $\hat{d}^{\mathcal{D}}(s,a) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}[(s_i,a_i) = (s,a)]$.
Estimate initial state-action distribution: $\widehat{d_0 \times \pi}(s,a) = \left(\frac{1}{n}\sum_{i=1}^n \mathbb{I}[s_0^i = s]\right)\pi(a|s)$.
Estimate transition probabilities $\widehat{\mathcal{P}^\pi}(s'|s,a)$ empirically.
Form diagonal matrix $D^{\mathcal{D}} = \operatorname{diag}(\hat{d}^{\mathcal{D}}(s,a))$.
Construct transition matrix $\widehat{\mathcal{P}^\pi}_*$ from estimates.
Solve linear system for $\hat{w}$: $(I - \gamma\,\widehat{\mathcal{P}^\pi}_*)D^{\mathcal{D}}\hat{w} = (1-\gamma)\widehat{d_0 \times \pi}$.
Compute standard DICE estimate: $\hat{\rho}^\pi_{\text{DICE}} = \frac{1}{n}\sum_{i=1}^n \hat{w}(s_i,a_i)r_i$.
Compute weighted DICE estimate (WIS-like): $\hat{\rho}^\pi_{\text{DICE,W}} = \frac{\sum_{i=1}^n \hat{w}(s_i,a_i)r_i}{\sum_{i=1}^n \hat{w}(s_i,a_i)}$.
Output: Approximate stationary ratio $\hat{w}(s,a)$, and off-policy value estimates $\hat{\rho}^\pi_{\text{DICE}}$, $\hat{\rho}^\pi_{\text{DICE,W}}$.

4.18.1 Practical Considerations and Advantages

DICE is *behavior-agnostic* and thus well-suited to off-line reinforcement learning scenarios where the dataset might have been generated by multiple or unknown behavior policies. It only needs tuples (s_0, s, a, r, s') sampled from the environment (through any policy or mixture of policies) along with the evaluation policy π. In many domains (e.g. healthcare), such a requirement is more realistic than assuming a single known behavior policy.

Additionally, by focusing on the stationary distribution correction rather than trajectory-level IS ratios, DICE methods can mitigate the large variance issues typically seen with long-horizon importance sampling. However, DICE still relies on learning a function $w_{\pi/D}(s,a)$ that accurately satisfies the stationary Bellman constraints, which can be challenging in high-dimensional or complex environments.

In summary, DICE provides a powerful framework for off-policy evaluation by directly estimating the ratio $d^\pi/d^{\mathcal{D}}$. The next sections will further explore theoretical guarantees, empirical performance, and extensions of the DICE family of methods.

4.19 Kullback-Leibler Divergence

The *Kullback-Leibler (KL) divergence* is a measure of how one probability distribution diverges from a second, reference probability distribution. In many contexts, one distribution (typically denoted by p) represents the true or target distribution of data, while the other (denoted by q) is an approximation or model distribution. The KL divergence is sometimes referred to as the *relative entropy* [79].

4.19.1 Definitions

For a discrete probability space, where the random variable X takes values in a countable set $\mathcal{X}$, the KL divergence from q to p is defined as

$$D_{\text{KL}}(p \parallel q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}. \tag{4.38}$$

Here, the logarithm is taken in the natural base, and by convention $0 \log(0/q) = 0$ for any $q \geq 0$.

For continuous random variables with probability density functions $p(x)$ and $q(x)$ defined over a domain $\mathcal{X} \subseteq \mathbb{R}$, the KL divergence is given by

$$D_{\mathrm{KL}}(p \parallel q) = \int_{\mathcal{X}} p(x) \log \frac{p(x)}{q(x)}\, dx. \tag{4.39}$$

Similarly, any region where $p(x) = 0$ contributes zero to the integral.

In essence, the KL divergence represents the extra amount of information (or coding length) required to describe samples drawn from p when using a coding scheme based on q instead of the optimal code based on p [80]. This quantity is widely used to quantify the inefficiency and information loss incurred when q is used to approximate the true distribution p [81].

4.20 K-means Clustering

Given a dataset $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$, k-means partitions X into k clusters $\{C_j\}_{j=1}^k$ by minimizing the within-cluster sum of squares (WCSS):

$$\min_{\{C_j\}} \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2,$$

where

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i.$$

The algorithm iterates between assigning points to the nearest centroids and updating centroids as cluster means [82, 83]. Initialization critically affects convergence; methods like k-means++ mitigate sensitivity [84]. While efficient, standard k-means assumes isotropic clusters and equal feature importance.

4.20.1 Weighted k-means

Assigning non-negative weights $\{w_i\}_{i=1}^n$ to points, weighted k-means minimizes:

$$\mathrm{WWCSS} = \sum_{j=1}^{k} \sum_{x_i \in C_j} w_i \|x_i - \mu_j\|^2,$$

with centroids updated via:

$$\mu_j = \frac{\sum_{x_i \in C_j} w_i x_i}{\sum_{x_i \in C_j} w_i}.$$

This prioritizes high-weight points. Let $J(C, \{\mu_j\}) = \mathrm{WWCSS}$. The assignment step minimizes J over C given $\{\mu_j\}$, while the update step minimizes J over $\{\mu_j\}$ given C.

Theorem 7 (Convergence)**.** *The sequence $\{J^{(t)}\}$ generated by weighted k-means converges monotonically to a local minimum in finite iterations.*

Proof. Each assignment step reduces or maintains the objective, since each point is reassigned to the cluster with the nearest weighted centroid, ensuring $J^{(t+1)} \leq J^{(t)}$. In the update step, each

centroid μ_j is recomputed as the weighted mean of points in C_j, which is the unique minimizer of the convex quadratic function

$$J_j(\mu) = \sum_{x_i \in C_j} w_i \|x_i - \mu\|^2.$$

As both steps monotonically decrease (or maintain) the objective and there are only finitely many distinct clusterings, the algorithm must converge in a finite number of iterations. However, due to the non-convexity of the full objective, the solution is guaranteed only to be a local minimum [84]. □

4.20.2 KL Divergence of State Transitions

Let $P, Q \in \mathbb{R}^{|S|\times|S|}$ be transition matrices for training/test cohorts over states S (clusters). The KL divergence (equation 4.38):

$$D_{\mathrm{KL}}(P\|Q) = \sum_{s,s'} P(s,s') \log \frac{P(s,s')}{Q(s,s')},$$

measures discrepancy, with $D_{\mathrm{KL}} \geq 0$ (Gibbs' inequality). Reduced D_{KL} indicates enhanced transition consistency across cohorts, validating cluster robustness [80]. Additivity allows decomposition across state subsets, isolating divergence sources. In context of study 2 (chapter 6), this measure serves as a quantitative metric for comparing the fidelity of the state transitions between two cohorts. A significant reduction in D_{KL} (e.g., reducing it to approximately one-third of its initial value) suggests that the clustering has improved the consistency of state transitions across cohorts. This is especially crucial when the clustering aims to capture clinically relevant heterogeneity.

4.21 General Transformer Models

Many state-of-the-art neural sequence transduction models employ an encoder-decoder architecture [85, 86, 87]. In such models, the *encoder* transforms an input sequence of symbols, $(x_1, \ldots, x_n)$, into a sequence of continuous representations that capture the essential features or meaning of the input. The *decoder* then uses this representation to generate an output sequence, $(y_1, \ldots, y_m)$, one symbol at a time in an auto-regressive fashion [88]—each new element is produced conditioned on the encoded input and the symbols generated so far.

The *Transformer* [89] follows this general encoder-decoder architecture but differ from earlier recurrent or convolutional models by relying entirely on self-attention and position-wise feed-forward networks. This design allows the model to directly capture dependencies between any two positions in the sequence, regardless of their distance, and to process sequences in parallel.

Figure 4.4 illustrates the overall architecture of the Transformer.

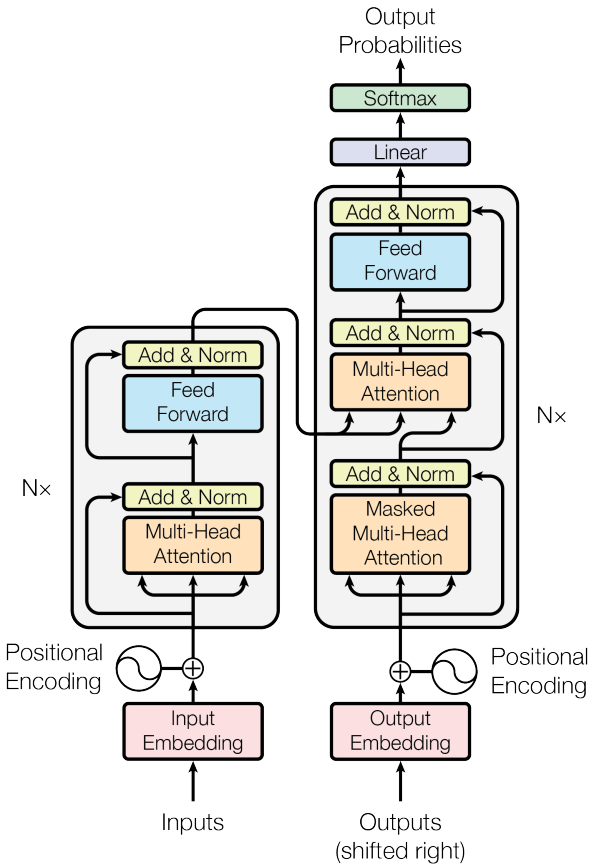


Fig. 4.4: The Transformer model architecture from [89].

4.21.1 Encoder and Decoder Stacks

Both the encoder and decoder are built as stacks of identical layers, allowing the model to gradually refine its representations through multiple levels of abstraction.

Encoder: The encoder is composed of a stack of identical layers. Each layer has two main sub-layers:

(i) A multi-head self-attention mechanism.

(ii) A position-wise fully connected feed-forward network.

The self-attention sub-layer allows every position in the input sequence to consider all other positions when forming its representation. This is particularly useful for capturing long-range dependencies. Following each sub-layer, residual connections [90] and layer normalization [91] are applied:

$$\text{LayerNorm}(x + \text{Sublayer}(x)).$$

These techniques help in stabilizing training and enable the construction of very deep models. The dimensionality of the embeddings and intermediate representations (often denoted as d_{model}) is kept consistent across layers.

Decoder: The decoder is similarly structured as a stack of identical layers, but with an additional twist. Each decoder layer includes three sub-layers:

(i) A self-attention mechanism over the output generated so far.

(ii) A multi-head attention mechanism over the encoder's output.

(iii) A position-wise feed-forward network.

The self-attention sub-layer in the decoder is modified with masking to prevent a position from attending to future positions. This ensures that the prediction for any given position depends only on the already generated output, thereby maintaining the autoregressive property. The additional attention sub-layer that attends over the encoder's output allows the decoder to incorporate contextual information from the input sequence into the generation process.

4.21.2 Attention Mechanisms

The attention mechanism is a fundamental component of the Transformer architecture, allowing the model to dynamically and selectively focus on different parts of the input sequence when building its representations. Rather than reducing the entire input to a single vector of fixed size, the attention mechanism constructs context-dependent representations by computing weighted sums of input features, where the weights indicate the relevance of each feature with respect to a particular query. This process allows the model to emphasise information that is most relevant to the task at hand, while downplaying less relevant details.

At the core of this mechanism lies the interaction between queries, keys, and values. In practice, the input embeddings are first transformed into these three distinct sets through learned linear projections. For each query, the model assesses its compatibility with all keys using a similarity measure, typically the dot product. This operation results in a set of scores that reflect how well each key matches with the query. To ensure that these scores remain at a manageable scale—especially when the dimension of the keys d_k is high—the dot products are divided by $\sqrt{d_k}$. This scaling prevents the softmax function, which converts the scores into a probability distribution, from saturating and producing extremely small gradients.

The softmax normalised scores serve as weights that determine the contribution of each corresponding value to the final output. That is, the mechanism computes a weighted sum of the values, where the weights indicate the degree of attention that each part of the input deserves. This process can be summarised by the equation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{4.40}$$

Here, the term QK^T represents the pairwise similarities between queries and keys, and the softmax function ensures that these similarities are normalized into a distribution that effectively weighs the values.

The beauty of this approach lies in its flexibility and efficiency. By allowing each query to dynamically aggregate information from different positions in the input, the attention mechanism is capable of capturing both local and long-range dependencies. Moreover, because these computations can be performed in parallel for all positions in the sequence, the attention mechanism is highly efficient and scalable.

Multi-head attention further extends this approach by projecting the inputs into multiple subspaces. Specifically, for h parallel heads, each head applies learned matrices:

$$W_h^Q,\, W_h^K \in \mathbb{R}^{d_{\text{model}} \times d_k}, \quad W_h^V \in \mathbb{R}^{d_{\text{model}} \times d_v},$$

and the outputs of all heads are concatenated and linearly transformed by $W^O \in \mathbb{R}^{(h\, d_v) \times d_{\text{model}}}$:

$$\text{head}_h = \text{Attention}(QW_h^Q,\, KW_h^K,\, VW_h^V), \tag{4.41}$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\, W^O. \tag{4.42}$$

This procedure allows each head to specialize in attending to different parts (subspaces) of the sequence.

4.21.3 Positional Encoding

Since the Transformer architecture does not include any recurrence or convolution, position information is provided via *positional encodings*:

$$PE \in \mathbb{R}^{n \times d_{\text{model}}}.$$

These are added directly to the input embeddings to indicate the position of each token. Concretely,

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{\text{model}}}), \tag{4.43}$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{\text{model}}}), \tag{4.44}$$

where pos is the position in the sequence (e.g., token index), and i ranges over the embedding dimensions. The sine and cosine functions enable the model to learn relationships based on relative positions via trigonometric identities.

4.21.4 Layer Normalization and Residual Connections

To facilitate stable training and preserve information from earlier sub-layers, each sub-layer (e.g., attention or feed-forward) is wrapped by a residual connection and a layer normalization:

$$\text{Output} = \text{LayerNorm}(x + \text{Sublayer}(x)). \tag{4.45}$$

LayerNorm is given by:

$$\text{LayerNorm}(x) = \gamma \odot \frac{x - \mu}{\sigma + \epsilon} + \beta, \tag{4.46}$$

where μ and σ are the mean and standard deviation of x, γ and β are learnable parameters, and ϵ is a small constant for numerical stability.

4.21.5 Position-wise Feed-Forward Networks

Each encoder and decoder layer also includes a position-wise feed-forward network (FFN) that is independently applied to each position in the sequence. A common choice is:

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)\, W_2 + b_2, \tag{4.47}$$

where $W_1 \in \mathbb{R}^{d_{\text{model}} \times 4d_{\text{model}}}$ and $W_2 \in \mathbb{R}^{4d_{\text{model}} \times d_{\text{model}}}$. This layer *expands* the hidden dimension to $4d_{\text{model}}$ internally, giving the model an extra capacity for learned transformations.

4.22 Temporal Fusion Transformer (TFT)

Multi-horizon forecasting aims to predict target values over multiple future time steps using heterogeneous inputs. These inputs typically consist of time-invariant features (static covariates), historical observations (past-observed inputs), and known future signals (a priori-known future inputs). Figure 4.5 illustrates this multi-source scenario, which is common in applications ranging from retail to healthcare and economics. This model was employed to predict MAP values in study 3 (chapter 7).

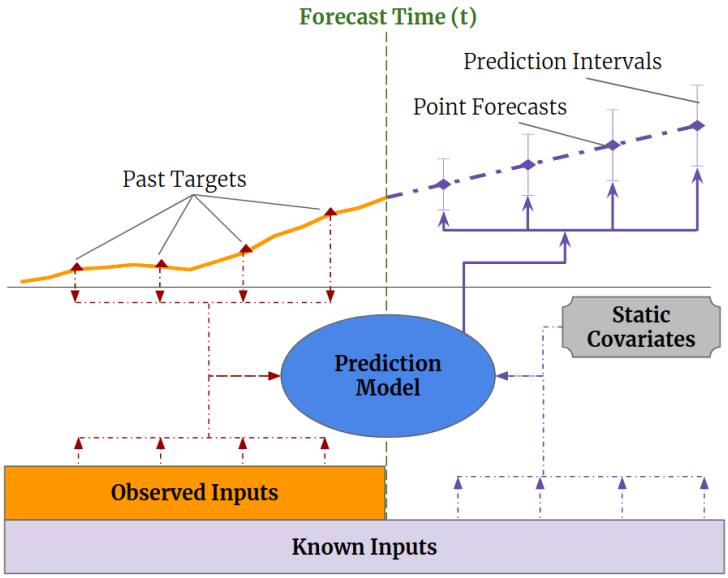


Fig. 4.5: Illustration of multi-horizon forecasting with static covariates, past-observed, and a priori-known future time-dependent inputs from [20].

The Temporal Fusion Transformer (TFT) is designed to address two main challenges: (i) the heterogeneous nature of the inputs, and (ii) the need for interpretability in the forecasts. An overview of the TFT architecture is shown in Figure 4.6 [20]. In the following sections, we describe the key components of the model along with the rationale behind their design.

4.22.1 Adaptive Gating Mechanisms

At the heart of TFT are adaptive gating mechanisms implemented via *Gated Residual Networks* (GRNs). GRNs enable the model to decide dynamically which parts of the input should undergo non-linear transformations. This selective processing is essential when dealing with complex, multi-modal data.

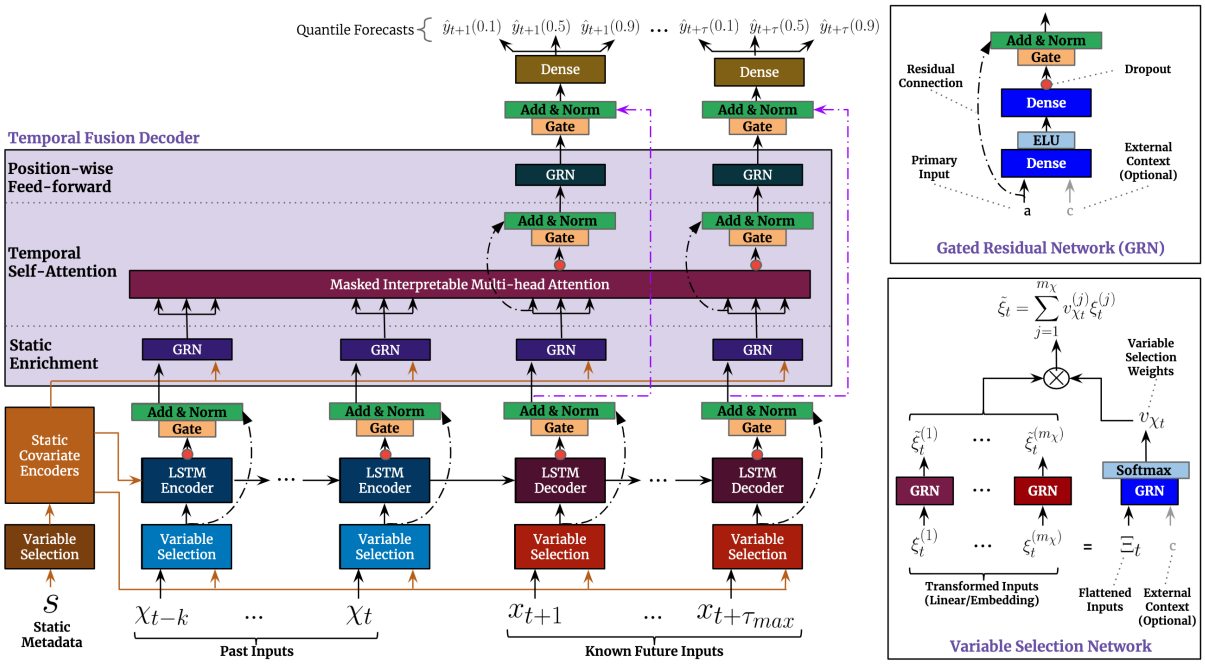


Fig. 4.6: Overview of the TFT architecture from [20]. The model ingests static covariates, historical observations, and known future inputs. Key modules include dynamic variable selection, adaptive GRNs for efficient non-linear processing, LSTM-based local temporal processing, and an interpretable multi-head attention mechanism for capturing long-term dependencies.

The GRN is computed as follows. Given a primary input vector $\boldsymbol{a}$ and an optional context vector $\boldsymbol{c}$, we first combine them linearly:

$$\boldsymbol{\eta}_1 = \boldsymbol{W}_1\,\boldsymbol{a} + \boldsymbol{W}_2\,\boldsymbol{c} + \boldsymbol{b}_1. \tag{4.48}$$

The combined signal is then passed through a non-linear activation:

$$\boldsymbol{\eta}_2 = \mathrm{ELU}\Big(\boldsymbol{\eta}_1\Big). \tag{4.49}$$

Finally, the output of the GRN is obtained by applying a Gated Linear Unit (GLU) with a residual connection followed by layer normalization:

$$\mathrm{GRN}(\boldsymbol{a},\boldsymbol{c}) = \mathrm{LayerNorm}\Big(\boldsymbol{a} + \mathrm{GLU}(\boldsymbol{\eta}_2)\Big), \tag{4.50}$$

where the GLU is defined as

$$\mathrm{GLU}(\boldsymbol{\gamma}) = \sigma\Big(\boldsymbol{W}_3\,\boldsymbol{\gamma} + \boldsymbol{b}_3\Big) \odot \Big(\boldsymbol{W}_4\,\boldsymbol{\gamma} + \boldsymbol{b}_4\Big).$$

This sequence—first a linear combination (producing $\boldsymbol{\eta}_1$), then a non-linear transformation (yielding $\boldsymbol{\eta}_2$), followed by gated residual learning—ensures that the GRN can model complex interactions while preserving the original signal.

4.22.2 Dynamic Variable Selection

In multi-horizon forecasting, many time-dependent inputs may be available, but not all are equally informative at every time step. TFT addresses this by using dynamic variable selection. Each time-dependent variable is first embedded into a d_{model}-dimensional vector. Denote by $\boldsymbol{\xi}_t^{(j)}$ the embedding of the jth variable at time t, and form the concatenated embedding:

$$\boldsymbol{\Xi}_t = \left[\boldsymbol{\xi}_t^{(1)^T} \quad \boldsymbol{\xi}_t^{(2)^T} \quad \cdots \quad \boldsymbol{\xi}_t^{(m_\chi)^T}\right]^T.$$

A GRN, conditioned on a static context vector $\boldsymbol{c}_s$, produces variable selection weights via a softmax:

$$\boldsymbol{v}_{\chi t} = \text{Softmax}\Big(\text{GRN}_{v_\chi}(\boldsymbol{\Xi}_t, \boldsymbol{c}_s)\Big).$$

Each variable is then individually processed:

$$\tilde{\boldsymbol{\xi}}_t^{(j)} = \text{GRN}_{\tilde{\xi}(j)}\left(\boldsymbol{\xi}_t^{(j)}\right),$$

and the final aggregated representation is formed as a weighted sum:

$$\tilde{\boldsymbol{\xi}}_t = \sum_{j=1}^{m_\chi} v_{\chi t}^{(j)}\, \tilde{\boldsymbol{\xi}}_t^{(j)}.$$

This dynamic selection not only mitigates the influence of irrelevant or noisy features but also enhances interpretability by revealing which inputs drive the forecasts at each time step.

4.22.3 Integration of Static Covariates

Static covariates such as store location or patient demographics are processed using dedicated GRN-based encoders. These encoders produce context vectors (e.g., $\boldsymbol{c}_s$, $\boldsymbol{c}_e$, $\boldsymbol{c}_c$, and $\boldsymbol{c}_h$) that are used to condition both the variable selection networks and subsequent temporal processing layers. This integration ensures that the model leverages invariant information to improve forecasting accuracy.

4.22.4 Fusion of Temporal Patterns

TFT captures both local and long-term temporal dependencies through a two-stage temporal fusion approach. First, a sequence-to-sequence Long Short-Term Memory (LSTM) network [92] processes recent inputs to model local temporal patterns. A gated skip connection is applied to the LSTM outputs to preserve important raw features:

$$\tilde{\boldsymbol{\phi}}(t,n) = \text{LayerNorm}\Big(\tilde{\boldsymbol{\xi}}_{t+n} + \text{GLU}(\boldsymbol{\phi}(t,n))\Big),$$

where $n \in [-k, \tau_{max}]$ indexes the relative time positions.

The LSTM outputs are then enriched with static context via another GRN:

$$\boldsymbol{\theta}(t,n) = \text{GRN}\Big(\tilde{\boldsymbol{\phi}}(t,n), \boldsymbol{c}_e\Big).$$

To capture long-range dependencies, TFT employs an interpretable multi-head attention mechanism inspired by [89]. The aggregated attention (see equation 4.21.2) output is computed as

$$\tilde{\boldsymbol{H}} = \frac{1}{m_H} \sum_{h=1}^{m_H} \text{Attention}\Big(\boldsymbol{Q}W_h^Q, \boldsymbol{K}W_h^K, \boldsymbol{V}W^V\Big). \tag{4.51}$$

In this formulation, the sharing of value weights across attention heads simplifies the interpretation of the attention scores, clearly indicating which past time steps are most influential. This dual strategy—using LSTMs for local context and multi-head attention for global dependencies—is central to the effectiveness of TFT and is depicted in Figure 4.6.

4.22.5 Quantile Forecasting for Uncertainty Estimation

A notable strength of TFT is its ability to generate prediction intervals via quantile forecasting. For each forecast horizon $\tau \in \{1, \ldots, \tau_{max}\}$ and quantile level q, the model outputs

$$\hat{y}(q, t, \tau) = \boldsymbol{W}_q \tilde{\boldsymbol{\psi}}(t, \tau) + b_q.$$

The training objective is to minimize the quantile loss:

$$QL(y, \hat{y}, q) = q\,(y - \hat{y})_+ + (1 - q)\,(\hat{y} - y)_+,$$

which explicitly accounts for the uncertainty in the predictions.

Chapter 5

Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis

Journal of
Clinical Medicine

MDPI

Article

Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis

Razvan Bologheanu [1,2,*,†], Lorenz Kapral [2,†], Daniel Laxar [2], Mathias Maleczek [1,2], Christoph Dibiasi [1], Sebastian Zeiner [1], Asan Agibetov [1], Ari Ercole [3], Patrick Thoral [4], Paul Elbers [4], Clemens Heitzinger [5] and Oliver Kimberger [1,2]

[1] Department of Anaesthesia, Intensive Care Medicine and Pain Medicine, Medical University of Vienna, 1090 Vienna, Austria
[2] Ludwig Boltzmann Institute for Digital Health and Patient Safety, 1090 Vienna, Austria
[3] Centre for Artificial Intelligence in Medicine, University of Cambridge, Cambridge CB2 0QQ, UK
[4] Department of Intensive Care Medicine, Laboratory for Critical Care Computational Intelligence, Amsterdam UMC, Vrije Universiteit, 1081 HV Amsterdam, The Netherlands
[5] Institute of Analysis and Scientific Computing, Department of Mathematics and Geoinformation, Technical University of Vienna, 1040 Vienna, Austria
* Correspondence: razvan.bologheanu@meduniwien.ac.at
† These authors contributed equally to the work.

Abstract: Background: The optimal indication, dose, and timing of corticosteroids in sepsis is controversial. Here, we used reinforcement learning to derive the optimal steroid policy in septic patients based on data on 3051 ICU admissions from the AmsterdamUMCdb intensive care database. Methods: We identified septic patients according to the 2016 consensus definition. An actor-critic RL algorithm using ICU mortality as a reward signal was developed to determine the optimal treatment policy from time-series data on 277 clinical parameters. We performed off-policy evaluation and testing in independent subsets to assess the algorithm's performance. Results: Agreement between the RL agent's policy and the actual documented treatment reached 59%. Our RL agent's treatment policy was more restrictive compared to the actual clinician behavior: our algorithm suggested withholding corticosteroids in 62% of the patient states, versus 52% according to the physicians' policy. The 95% lower bound of the expected reward was higher for the RL agent than clinicians' historical decisions. ICU mortality after concordant action in the testing dataset was lower both when corticosteroids had been withheld and when corticosteroids had been prescribed by the virtual agent. The most relevant variables were vital parameters and laboratory values, such as blood pressure, heart rate, leucocyte count, and glycemia. Conclusions: Individualized use of corticosteroids in sepsis may result in a mortality benefit, but optimal treatment policy may be more restrictive than the routine clinical practice. Whilst external validation is needed, our study motivates a 'precision-medicine' approach to future prospective controlled trials and practice.

Keywords: sepsis; corticosteroids; outcomes; artificial intelligence; reinforcement learning

Citation: Bologheanu, R.; Kapral, L.; Laxar, D.; Maleczek, M.; Dibiasi, C.; Zeiner, S.; Agibetov, A.; Ercole, A.; Thoral, P.; Elbers, P.; et al. Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis. *J. Clin. Med.* **2023**, *12*, 1513. https://doi.org/10.3390/jcm12041513

Academic Editor: Sergio Ruiz-Santana

Received: 11 January 2023
Revised: 30 January 2023
Accepted: 6 February 2023
Published: 14 February 2023

1. Introduction

Sepsis represents a significant cause of morbidity and is responsible for 11 million deaths globally each year [1]. Defined as "life-threatening organ dysfunction caused by a dysregulated host response to infection", sepsis is an umbrella term for a heterogeneous syndrome with many distinct phenotypes and wide variation in outcomes [2,3]. As a result, clinical trials have provided conflicting evidence concerning the benefit of specific therapies beyond source control, antibiotics, and maintenance of tissue perfusion [4,5].

Corticosteroids have been extensively investigated as a therapeutic option for sepsis ever since Cook et al. first advocated their use seven decades ago, but uncertainty regarding

their optimal use nevertheless persists [6]. More recently, the case for corticosteroids in sepsis was based on the evidence of adrenal insufficiency accompanying critical illness [7]. Since diagnostic criteria for adrenal insufficiency are missing, identifying patients that should receive corticosteroids is challenging [7]. In addition, several studies have found that corticosteroids can lead to a faster resolution of shock but provided equivocal results concerning survival [8–10].

Currently, guidelines for the management of sepsis suggest using corticosteroids in septic patients with ongoing vasopressor requirement [5]. However, the optimal treatment regimen, particularly timing, duration, and dose of corticosteroids, is not known, and the clinical significance of potential adverse effects of corticosteroid therapy is unclear [5]. Identifying patients who are likely to benefit from corticosteroids is essential and attempts at personalizing corticosteroid therapy using novel approaches, such as machine learning and transcriptomics, have been reported [11,12].

Since interventional studies in sepsis are challenging due to the extreme heterogeneity of its phenotypes, machine learning could represent a complementary evaluation method for specific treatments using observational data. In essence, the aim is to construct an algorithm that can exploit clinician variances in treatment policy over a large dataset in a way that it is possible to find the effects of the treatment on similar patients at a given time. Reinforcement learning, one of the three primary machine learning branches, can be applied to this type of problem [13,14]. Reinforcement learning algorithms can serve as the foundation for decision support tools in intensive care, where decision making is based on sequential, highly granular data [15,16]. In brief, such algorithms attempt to find an 'optimal' policy that maximizes some reward function (for example survival), given a particular treatment strategy with a comprehensive description of the state of the patient at that time [13]. In the present study, we describe the development of a reinforcement learning algorithm to find the optimal approach to corticosteroid therapy in septic patients based on high-resolution clinical data from an intensive care database.

2. Materials and Methods

2.1. Data Sources and Data Processing

All data were queried from the AmsterdamUMCdb database. Approval was obtained for 3rd party re-use of AmsterdamUMCdb data for research from its steering group, and the research was conducted according to the data use agreement. Such a study of deidentified data is not subject to the need for ethical review. The ethical approvals for the AmsterdamUMCdb have been previously described [17]. AmsterdamUMCdb contains high-resolution clinical data related to 23,106 ICU admissions of 20,109 patients from 2003 to 2016 [17]. Patients with sepsis were identified based on the Sepsis-3 criteria2 Accordingly, patients with new organ dysfunction as indicated by either a SOFA score ≥ 2 at admission or an increase of 2 points or more in the SOFA score during the ICU stay, in the context of suspected infection as described in Supplemental Table S1, were included in the sepsis cohort [2,18,19]. Patients aged <18 years at the time of the ICU admission and patients who stayed in the ICU less than 24 h were excluded. The onset of the septic episode was considered the day the change in the SOFA score occurred and patients remained in the sepsis cohort until discharge or death.

In total, 281 variables were extracted, of which 277 input variables were coded as a multidimensional time series with a time resolution of 24 h. Every ICU day was considered separately, and only current measurements available at that timepoint were included in each data point. Only numeric variables represented in more than 2% of the data points were included. The imbalance resulting from missing data and the variable sampling rate were addressed by preprocessing: missing laboratory values were imputed using forward fill, while missing medication doses were set to 0. Overall, 17.93% of all input values were imputed. Numeric data were normalized to values between -1 and $+1$; for frequently sampled parameters (e.g., heart rate), the mean, the maximum, the minimum, and standard deviation were calculated, and for others (e.g., continuously administered drugs), the sum,

i.e., the 24 h cumulative dose, was used as input instead. Therefore, the final number of extracted parameters increased to 379. The complete list of input features is provided in Supplemental Table S2.

2.2. Algorithm Development

Reinforcement learning is based on modeling a virtual decision-making 'agent' interacting with its environment described by a set of continuous states; the interaction between the agent and the environment predetermined as the action space (in this case, the finite number of treatment choices). At each step, the agent chooses an action, and the environment changes its state, returning a reward. The reward signal is used to train the agent, which gradually learns an optimal policy that maximizes return [20].

We implemented a reinforcement learning algorithm, consisting of two distinct neural networks, based on the Markov Decision Process using the temporal difference actor-critic method able to suggest the optimal corticosteroid dose for each septic patients by retrospectively analyzing clinical data [20–22]. The dataset was randomly split into a training set, consisting of 70% of all patients, and two smaller datasets for validation (20%) and testing (10%) (Figure 1). The algorithm was trained on trajectories of successive patient states, where a state corresponded to a vector of all features within a 24 h period, other than mortality and the administered corticosteroid dose. The reward signal associated with each transition was related to the ICU mortality. The action space consisted of five discrete actions, defined by converting the cumulative 24 h dose of systemic corticosteroids to the equivalent dose of hydrocortisone and binning the resulting values: the null ('no corticosteroids') action and four dose ranges: 1–100 mg, 101–200 mg, 201–300 mg, and over 300 mg hydrocortisone [23]. A detailed description of the reinforcement learning model is provided in Supplemental File S1 and Supplemental Figure S1. The reinforcement learning algorithm was built using the TensorFlow 2.7 Python library [24].

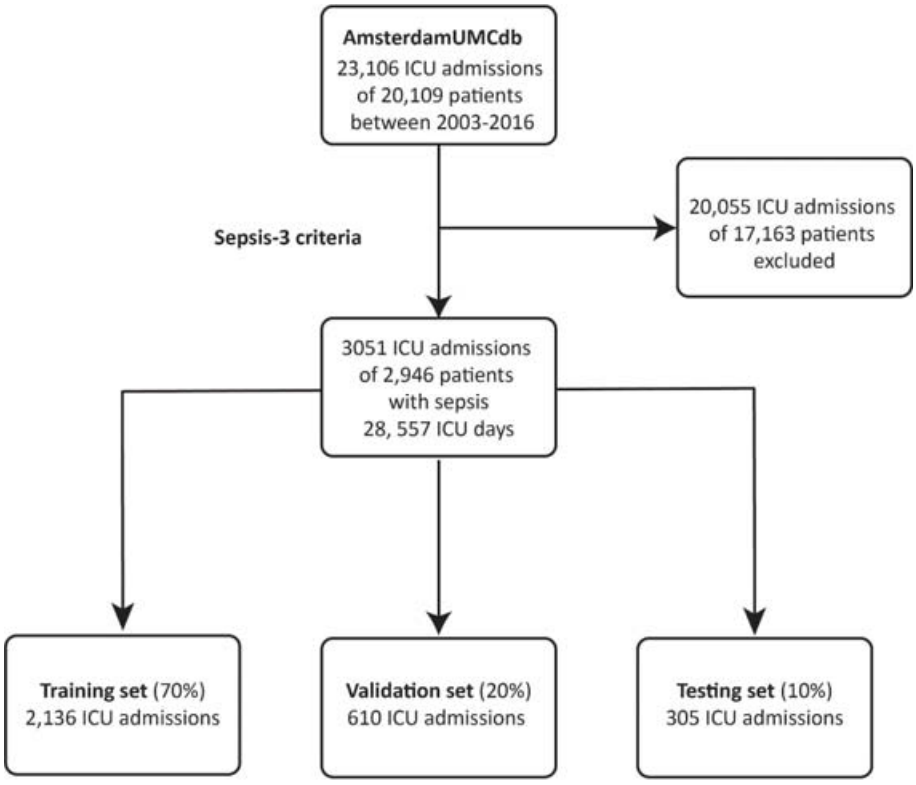


Figure 1. The Sepsis Cohort. Patients with sepsis from the AmsterdamUMC database were identified using the Sepsis-3 criteria. The sepsis cohort was randomly split in three distinct subsets used for training, evaluating, and testing the reinforcement learning algorithm.

2.3. Evaluation of the Algorithm

The reinforcement learning algorithm was initially evaluated by comparing the actual reward after concordant actions, i.e., when the actual treatment and the corticosteroid dose suggested by the agent were identical, with the reward after discordant actions in the testing set.

The performance of such reinforcement learning algorithms could not be directly evaluated by measuring the received reward of each action, since the reinforcement learning (evaluation) policy was different from the clinician (behavior) policy and the actual reward represented the performance of the clinician policy. We implemented a high-confidence off-policy evaluation (HCOPE) of the algorithm, a statistical method which compares the performance of the algorithm's policy with a baseline, the performance of the clinician policy, and computes the probability that the algorithm's policy has a performance below this baseline to select the best performing model. Using the clinician policy, a set of trajectories was generated and used to lower-bound the performance of the evaluation policy. The high-confidence off-policy evaluation (HCOPE) allowed for determining whether the 95% lower bound of the expected reward of the policy of the reinforcement learning agent exceeded the average reward of the clinician policy, i.e., the actual treatment the patients received [25,26].

Finally, we estimated the relative importance of each variable using a Layer-wise Relevance Propagation algorithm and ranked the input features of the RL algorithm according to their contribution to the agent's decision [27]. To allow for comparison between the relevance of the input features of agent's policy and the clinical practice, we developed a random forest model using the Scikit-learn Python library that predicts the clinicians' policy, simulating the clinician behavior, and we ranked the clinical variables supporting the average clinician behavior according to the parameters of the fitted model [28].

3. Results

A total of 3051 ICU admissions at the Amsterdam UMC corresponding to 2946 distinct patients were included (Figure 1).

Repeated admissions to the ICU, both remote and during the same hospital stay, were included if they met the sepsis definition and were analyzed as independent ICU stays. 1395 admissions were associated with vasopressor use and lactate values >2 mmol/l during the ICU stay, therefore meeting the criteria for septic shock. The cumulative length of stay from the onset of sepsis until ICU discharge was 28,557 days corresponding to as many data points. The training dataset comprised 2136 randomly selected ICU admissions, leaving a total of 610 and 305 admissions in the evaluation and testing datasets, respectively (Figure 1). Patients' characteristics are summarized in Table 1.

The relative error of the actor-critic model decreased over the training steps and converged after 250 epochs at 0.044 of the initial relative error (Figure 2a). The concordance between the virtual agent's action and the retrospective action by ICU physicians started at 22%, which was the expected value considering the dimension of the action space (five possible actions). The overall agreement between the virtual agent and the human clinicians reached 63% after convergence (Figure 2c). Similarly, the probabilities of choosing each action from the action space were equal initially. Over the training epochs, the virtual agent increasingly tended towards withholding corticosteroids. After convergence, in 65% of ICU days, the agent chose to withhold corticosteroids, and in patients where corticosteroids were prescribed, the suggested dose was low (Figure 2b). In contrast, the human clinicians prescribed corticosteroids in 45% of data points. Although the virtual agent displayed a tendency towards passive behavior, in 49% of the cases where the agent chose to administer glucocorticoids, the ICU physicians acted concordantly.

Table 1. Summary of patients' characteristics. Each ICU admission is considered separately.

Characteristics	Summary (Total)	Summary (Survivors)	Summary (Non-Survivors)
Total number of ICU admissions	3051	2336	715
Male sex, No. (%)	1758 (57.6%)	1353 (57.9%)	405 (56.6%)
Age group (years), No. (%)	–	–	–
18–39	342 (11.2%)	303 (12.9%)	39 (5.4%)
40–49	322 (10.5%)	265 (11.3%)	57 (7.9%)
50–59	518 (17.0%)	414 (17.7%)	104 (14.5%)
60–69	757 (24.8%)	591 (25.2%)	166 (23.2%)
70–79	709 (23.2%)	506 (21.6%)	203 (28.3%)
>80	403 (13.2%)	257 (11%)	146 (20.4%)
Highest SOFA score during the ICU stay, Median (IQR)	10 (6)	9 (6)	13 (6)
Sofa score at sepsis onset, Median (IQR)	9 (6)	8 (5)	11 (7)
Septic shock, No. (%)	1395 (45.7%)	845 (36.1%)	550 (76.9%)

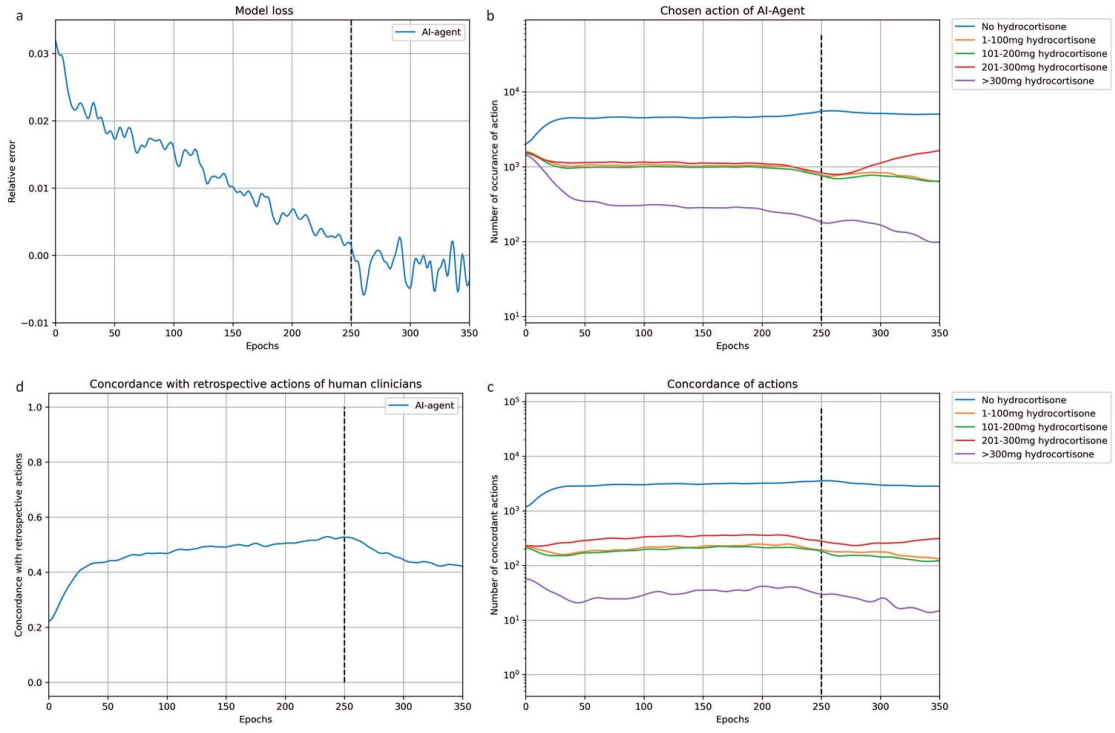


Figure 2. Training process of the virtual agent. Figure 2 shows how the performance and the behavior of the RL agent changed during the training process. On the X-axis, the number of epochs, i.e., how many times the algorithm had worked through the learning dataset, since the beginning of the training is displayed. The vertical dotted line marks the end of the training process. (**a**) The decrease in the relative error, which reflects the accuracy of the model's output, during the training process. (**b**) The number of occurrences for each action suggested by the algorithm during training is displayed in the (**b**). All five possible actions are equally represented at the beginning of the training. After 50 epochs, the algorithm's tendency to withheld corticosteroids becomes obvious. (**c**) The increasing overall agreement between the RL policy and the actual historic treatment. (**d**) The number of occurrences when agreement between the RL policy and the retrospective treatment was reached is displayed across the five possible actions in (**b**).

In the testing dataset, the treatment suggested by the virtual agent matched the retrospective action by ICU physicians in 59% of the data points. The agent's tendency to prescribe less corticosteroids was also confirmed in the testing dataset: corticosteroids were withheld in 62% of the ICU days, compared to 52% according to the ICU physicians. Accordingly, the average daily corticosteroid dose prescribed by the virtual agent was lower (Figure 3). Both ICU physicians and the RL agent tended to prescribe corticosteroids in the early phase of the septic episode and corticosteroid use dropped sharply after 10 days (Figure 3).

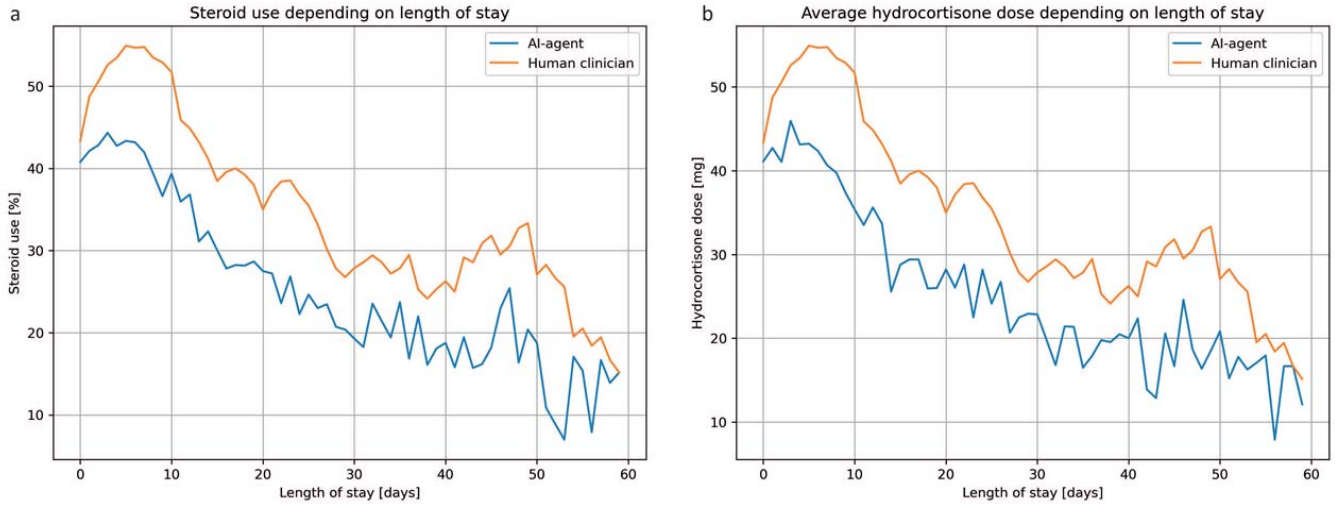


Figure 3. Comparison of corticosteroid use between ICU physicians and the RL agent. Use of corticosteroids as percentage of patients receiving corticosteroids (**a**) and average cortisone dose (**b**) is compared between the historic treatment in the ICU and the RL policy after adjusting for the ICU length of stay. Both ICU physicians and the RL agent tend to prescribe corticosteroids during the early phase of the septic episode. Notably, the RL policy is more restrictive compared to the actual treatment the patients received.

The ratio between the reward of the agent's policy and the clinicians' policy increased over the training process and high-confidence off-policy evaluation (HCOPE) demonstrated that the 95% lower bound of the expected average reward for the agent's policy was higher compared to the average reward for the historical decisions by clinicians after 200 epochs (Figure 4). Accordingly, the normalized expected mortality rate decreased and was lower than 0.7. Overall, when patients from the testing set received the same glucocorticoid therapy as suggested by the RL agent, mortality was lower: the mortality across all ICU days, i.e., the ICU days that eventually result in patient's death, when the decisions made by the RL agent and the ICU physician were identical was 22.38% compared to 28.33% in case the actions were different. This finding was consistent both when the RL agent withheld corticosteroids (25.85% of the data points compared to 32.22%) and when the RL agent suggested using corticosteroids (33.02% of the data points compared to 34.27%).

We modeled the retrospective treatment policy by the ICU physicians using a random forest model that predicted the clinicians' treatment decisions. The micro-average multiclass Area under the Receiver Operator Characteristic Curve for the random forest model was 0.8 (Supplemental Figure S2). The most relevant input features underlying the decisions of the reinforcement learning algorithm and the random forest model, respectively, are presented in Supplemental Tables S3 and S4 and Supplemental Figure S3. Both algorithms relied on vital parameters and laboratory values to determine the optimal treatment policy. However, vasopressor use and PEEP were distinctly more relevant for the clinician policy. Accordingly, although the reinforcement learning agent was consistently more restrictive

compared to human clinicians, the difference is more obvious in patients who met the criteria for septic shock (Figure 5).

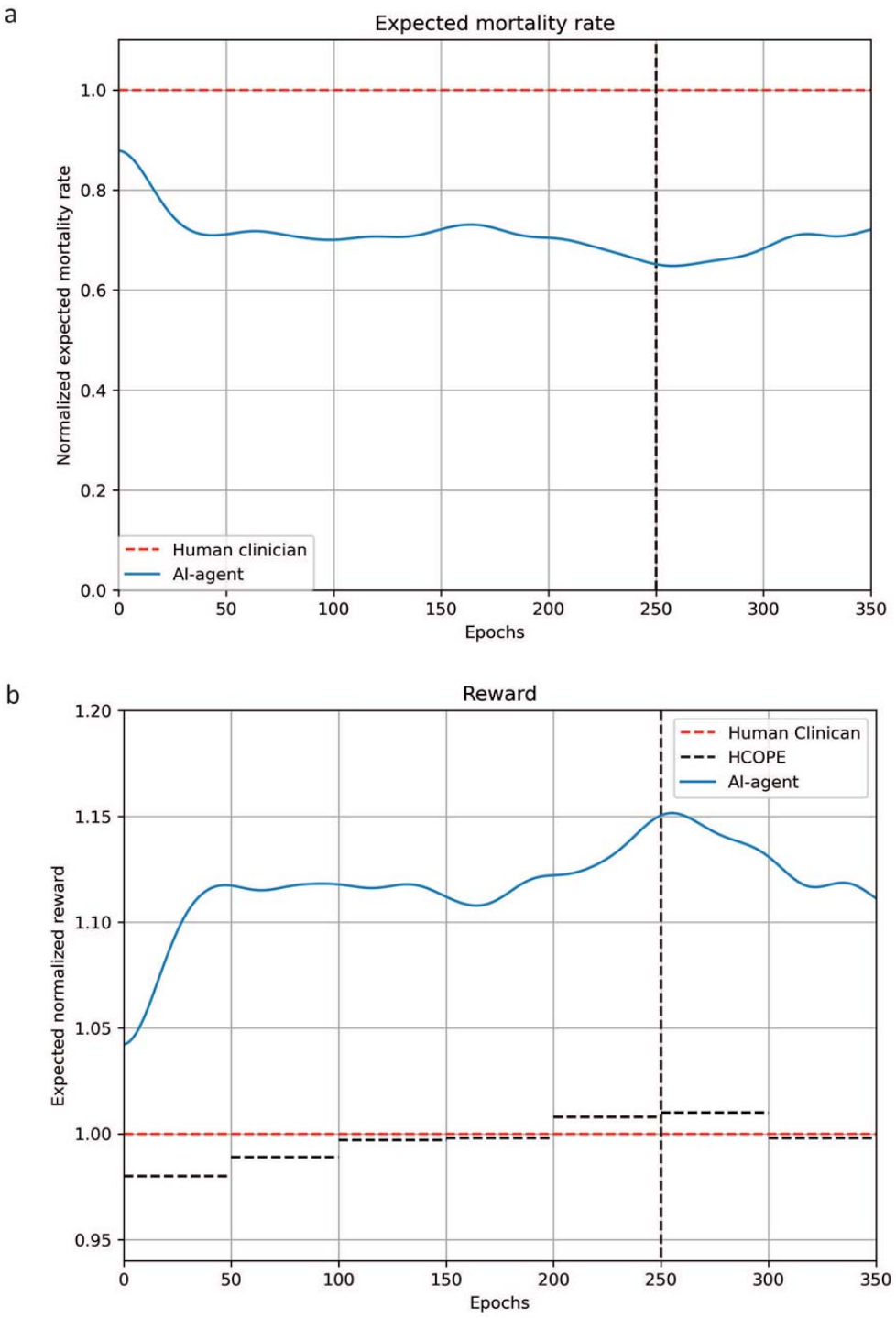


Figure 4. Comparison between the evaluation (RL) policy and the behavior policy (the actual treatment). (**a**) The change in the normalized expected mortality rate across training epochs, (i.e., the number of iterations or how many times the algorithm had worked through the learning dataset, since the beginning of the training) is represented in Figure 5a. (**b**) The 95% lower bound of the normalized expected reward of the RL policy (black dotted line) determined by high-confidence off-policy evaluation compared to the estimated reward of the clinician policy (red dotted line) is shown in (**b**).

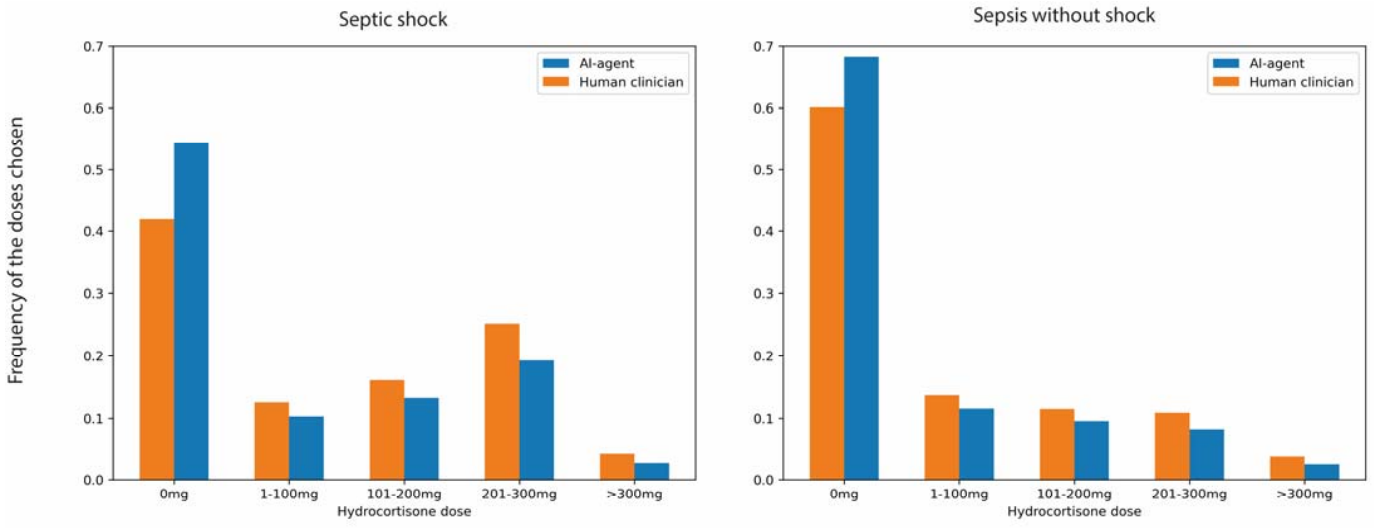


Figure 5. Comparison between the RL and physician policy in patient states grouped by septic shock criteria.

4. Discussion

We present a reinforcement learning algorithm trained to optimize the corticosteroid treatment strategy for a specific patient state in critically ill patients with sepsis. The novelty of our approach is that it potentially enables an individualized therapy to improve a highly relevant outcome based on clinical parameters routinely collected in the ICU. The goal of our reinforcement learning algorithm, determined by the reward signal, was to minimize mortality. Indeed, in the testing dataset, ICU mortality was the lowest in patients who received a treatment identical to the action suggested by the algorithm. Off-policy evaluation confirmed that the algorithm performed well within the given environment and even outperformed the clinician policy in the validation dataset.

Currently, the rationale for corticosteroids in sepsis is based on several studies suggesting faster resolution of shock in septic patients who require vasopressors despite adequate fluid resuscitation [5]. While earlier studies showed a mortality benefit, this was not consistently confirmed in subsequent trials [8,9,29–32]. This led to frequent changes in the clinical practice to accommodate new, often conflicting evidence, which have been likened to a "swinging pendulum" situation [30]. The most recent guidelines for the treatment of sepsis suggest corticosteroids as early as 4 h after the initiation of treatment in patients who require vasopressors. In the testing subset of our sepsis cohort, where 45.7% of patients met the criteria for septic shock, corticosteroids were suggested by the virtual agent in 38% of the ICU days. Conversely, ICU physicians used corticosteroids in 48% of the data points, yet only in 49% of the cases where the reinforcement learning agent suggested using corticosteroids, the actual treatment prescribed in the ICU was concordant. This difference may be a result of at least two factors. First, the reward signal used for training was related to the mortality and the reinforcement learning agent aimed to maximize survival. Second, corticosteroids have been historically reserved for patients who require more vasopressors and have higher severity of disease and, therefore, worse outcomes. Indeed, the random forest model we developed to simulate decision making by the ICU physicians showed that blood pressure and vasopressor use were most consistently associated with corticosteroid use. Furthermore, due to the retrospective nature of our study, we expected that the association between higher severity scores and corticosteroid use in the database would translate in a bias of the RL policy towards the null action.

We identified patients from the database with sepsis algorithmically, and this required a pragmatic operationalization of the Sepsis-3 criteria, using a data-driven approach, instead of relying on coding data to be defined [2,19]. This method has been used before and has the advantage of being more reliable; more reproducible; and therefore, appropriate for

epidemiological or database studies [33]. These operational criteria can provide consistent estimates of the sepsis incidence over longer periods, despite its inherent limitations, such as the assumptions about suspected infections being confirmed, pre-admission organ function, and the impact of the caregivers' decisions on the SOFA score [33–35].

Although traditionally, artificial intelligence algorithms have been often compared to a black box, several methods are available to provide insight into which variables contributed most to the algorithmic decisions [36]. We ranked the input features based on relevance, showing that our model was explainable and valid from a clinical standpoint and that the agent relied on plausible clinical variables to make its decisions. If the random forest model accurately simulates the decision-making process by ICU physicians, comparing the relative relevance of the input features between the reinforcement learning algorithm and the random forest model can reveal how a treatment policy can be developed to maximize ICU survival contrasts with actual care. Unlike current clinical practice, where refractory shock is the single most important factor considered to prescribe corticosteroids, vasopressor requirements and lactate only had a limited influence on the reinforcement learning policy while being highly relevant for the clinicians' policy. Similarly, the time elapsed since the onset of sepsis ranked distinctly higher amongst input features for the historical treatment by ICU physicians compared to the reinforcement learning treatment. These findings confirm the usual practice of prescribing corticosteroids early for patients in septic shock [5]. Interestingly, the machine learning policy resulted in a similar corticosteroid use pattern, characterized by an abrupt fall in steroid use after the 10th day since onset without explicitly relying as much on the time elapsed from the onset of sepsis. Conversely, total protein in cerebrospinal fluid (CSF) and the standard deviation of the heart rate ranked higher among the input parameters of the virtual agent only. It might seem surprising that a parameter that is rarely sampled is highly relevant for the output of the algorithm. Although corticosteroids are recommended for prevention of neurological sequelae in patients with bacterial meningitis, they have no effect on mortality [37]. Alternatively, lumbar puncture might be performed as a part of the work-up in patients with fever of unknown origin and subtle neurological symptoms [38]. In either case, since non-missing values are highly suggestive of a neurological diagnosis, informative missingness might explain its relevance for the reinforcement learning policy.

Arterial blood pressure, leucocyte count, serum sodium, and blood glucose levels were similarly influential in both algorithms. These findings seem biologically plausible, given the essential role of corticosteroids in regulating glucose metabolism and electrolyte homeostasis [39]. Corticosteroids also potentiate the effects of catecholamines and mobilize neutrophils, leading to leukocytosis and neutrophilia [40,41]. It is reasonable that clinical variables related to the physiological effects of corticosteroids could help guide therapy in septic patients by accurately predicting their effects in specific patient states. However, these results must be interpreted cautiously. The method we used to rank input variables estimates the overall contribution of all variables to the output of the model. Furthermore, unlike traditional statistical modeling, neural networks are less suitable for determining relationships between variables. Finally, all input variables were normalized between -1 and +1, and the relation between the normalized values, the actual values, and the reference range for each variable was determined by the variable's distribution and is not obvious or readily interpretable for clinicians.

We acknowledge several limitations of our study. First, we used a single database to develop our algorithm and our findings have not been externally validated, which considerably limits the clinical applicability of the model. Like most of the artificial intelligence research in the intensive care, our study is in the prototype phase, and broad implementation remains a distant goal [42] Although machine learning models could be transferred across ICUs, moving these models to the bedside proves challenging [42]. Artificial intelligence holds great promise to enhance the practice of intensive care and the management of sepsis in the ICU; however, the current state of AI in intensive care does not support its routine use due to regulatory reasons, but also because uncertainty

surrounds how these models could be included in daily practice, and good prospective studies still need to be included. Second, data used to train and test the model originate from a single medical center over several years. Changes in the best care practices over time and differences between local policies concerning ICU admission and sepsis management might result in relevant heterogeneity of the sepsis cohort and the outcomes. However, the aim of this study was to create an algorithm that can exploit these differences to derive an optimal treatment policy by analyzing several different suboptimal policies. Third, data were anonymized, and in the process, all notes were removed. Consequently, we could not account for the withdrawal of life-sustaining therapies. Fourth, by using a 24 h step to model the patients' trajectories, our model artificially creates data points that encompass more data than are available to the clinician at any given time. We considered the time resolution of 24 h and the action space defined as the cumulative 24 h dose of corticosteroids due to several reasons, since this approach allowed us to compare different treatment regimens, using different substances, doses, and intervals. Furthermore, in our experience, therapy goals and some therapeutic measures for the next 24 h are defined during the ICU rounds, once daily. Therefore, modelling clinical data as time-series data with a resolution of 24 h resembles, to some extent, clinical practice.

Decision making in the ICU typically takes place during the once-daily rounds and the cumulative 24 h dose allows for different treatment regimens to be compared regardless of substance and timing. Finally, we analyzed all clinical data from onset of sepsis until discharge from the ICU, which most likely covers a significantly longer period than the duration of the septic shock. However, clearly delineating between the acute critical illness, and subsequent organ dysfunction and persistent critical illness does not seem feasible in the context of the present study.

5. Conclusions

We developed and evaluated a reinforcement learning algorithm that used clinical data to derive the optimal corticosteroid therapy aimed at improving mortality in patients with sepsis. The algorithm performed well in the testing dataset, and the reinforcement learning policy was associated with a lower mortality than the clinician's policy. Due to the exploratory nature of our work, future research focusing on external validation of the model is required before prospective evaluation at the bedside. Our model suggests that a more targeted and individualized, reinforcement learning-driven approach to corticosteroids is possible and motivates prospective evaluation of treatment scenarios beyond refractory shock.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/jcm12041513/s1, Supplemental Table S1: Diagnosis of sepsis. Supplemental Table S2: Input features included in development of the algorithm. Supplemental Figure S1: Development of the RL Algorithm. Supplemental Figure S2: Micro-average ROC curve of the SVM algorithm. Supplemental Table S3: The most relevant predictors of the clinicians' policy according to the SVM algorithm ordered from the lowest to highest rank. Supplemental File S1: The 20 most relevant input features for the RL and random forest models [43–45].

Author Contributions: Conceptualization, R.B., M.M. and O.K.; methodology, L.K.; software, D.L, M.M. and L.K.; validation, L.K. and. A.A.; formal analysis, O.K.; investigation, R.B.; resources, A.E., P.E. and P.T.; data curation, S.Z., D.L. and C.D.; writing—original draft preparation, R.B.; writing—review and editing, A.E., O.K. and P.T.; visualization, L.K. and R.B.; supervision, O.K. and C.H.; project administration, R.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Approval was obtained for third party re-use of AmsterdamUMCdb data for research from its steering group, and the research was conducted according to the data use agreement. Such a study of deidentified data is not subject to the need for ethical review. The ethical approvals for the AmsterdamUMCdb have been previously described.

Data Availability Statement: Access to the dataset used in this manuscript may be requested from Amsterdam Medical Data Science (https:/amsterdammedicaldatascience.nl/ accessed on 1 January 2021).

Acknowledgments: The authors acknowledge the European Society of Intensive Care Medicine (ESICM) for support as part of the 2021 ESICM Datathon project.

Conflicts of Interest: The authors declare no conflict of interest regarding the contents of this submission.

References

1. Rudd, K.E.; Johnson, S.C.; Agesa, K.M.; Shackelford, K.A.; Tsoi, D.; Kievlan, D.R.; Colombara, D.V.; Ikuta, K.S.; Kissoon, N.; Finfer, S.; et al. Global, regional, and national sepsis incidence and mortality, 1990–2017: Analysis for the Global Burden of Disease Study. *Lancet* **2020**, *395*, 200–211. [CrossRef]
2. Singer, M.; Deutschman, C.S.; Seymour, C.W.; Shankar-Hari, M.; Annane, D.; Bauer, M.; Bellomo, R.; Bernard, G.R.; Chiche, J.-D.; Coopersmith, C.M.; et al. The Third International Consensus Definitions for Sepsis and Septic Shock (Sepsis-3). *JAMA* **2016**, *315*, 801–810. [CrossRef] [PubMed]
3. Seymour, C.W.; Kennedy, J.N.; Wang, S.; Chang, C.-C.H.; Elliott, C.; Xu, Z.; Berry, S.; Clermont, G.; Cooper, G.; Gomez, H.; et al. Derivation, Validation, and Potential Treatment Implications of Novel Clinical Phenotypes for Sepsis. *JAMA* **2019**, *321*, 2003–2017. [CrossRef]
4. Iwashyna, T.J.; Burke, J.F.; Sussman, J.B.; Prescott, H.C.; Hayward, R.A.; Angus, D.C. Implications of Heterogeneity of Treatment Effect for Reporting and Analysis of Randomized Trials in Critical Care. *Am. J. Respir. Crit. Care Med.* **2015**, *192*, 1045–1051. [CrossRef]
5. Evans, L.; Rhodes, A.; Alhazzani, W.; Antonelli, M.; Coopersmith, C.M.; French, C.; Machado, F.R.; Mcintyre, L.; Ostermann, M.; Prescott, H.C.; et al. Surviving sepsis campaign: International guidelines for management of sepsis and septic shock 2021. *Intensive Care Med.* **2021**, *47*, 1181–1247. [CrossRef]
6. Cook, C.; Smith, C. Sepsis and cortisone. *Nature* **1952**, *170*, 980. [CrossRef]
7. Annane, D.; Pastores, S.M.; Arlt, W.; Balk, R.A.; Beishuizen, A.; Briegel, J.; Carcillo, J.; Christ-Crain, M.; Cooper, M.S.; Marik, P.E.; et al. Critical illness-related corticosteroid insufficiency (CIRCI): A narrative review from a Multispecialty Task Force of the Society of Critical Care Medicine (SCCM) and the European Society of Intensive Care Medicine (ESICM). *Intensiv. Care Med.* **2017**, *43*, 1781–1792. [CrossRef] [PubMed]
8. Annane, D.; Bellissant, E.; Bollaert, P.E.; Briegel, J.; Confalonieri, M.; De Gaudio, R.; Keh, D.; Kupfer, Y.; Oppert, M.; Meduri, G.U. Corticosteroids in the treatment of severe sepsis and septic shock in adults: A systematic review. *JAMA* **2009**, *301*, 2362–2375. [CrossRef]
9. Annane, D.; Bellissant, E.; Bollaert, P.E.; Briegel, J.; Keh, D.; Kupfer, Y. Corticosteroids for treating sepsis. *Cochrane Database Syst. Rev.* **2015**, *12*, CD002243. [CrossRef]
10. Rygård, S.L.; Butler, E.; Granholm, A.; Møller, M.H.; Cohen, J.; Finfer, S.; Perner, A.; Myburgh, J.; Venkatesh, B.; Delaney, A. Low-dose corticosteroids for adult patients with septic shock: A systematic review with meta-analysis and trial sequential analysis. *Intensiv. Care Med.* **2018**, *44*, 1003–1016. [CrossRef] [PubMed]
11. Pirracchio, R.; Hubbard, A.; Sprung, C.L.; Chevret, S.; Annane, D.; for the Rapid Recognition of Corticosteroid Resistant or Sensitive Sepsis (RECORDS) Collaborators. Assessment of Machine Learning to Estimate the Individual Treatment Effect of Corticosteroids in Septic Shock. *JAMA Netw. Open* **2020**, *3*, e2029050. [CrossRef]
12. Antcliffe, D.B.; Burnham, K.L.; Al-Beidh, F.; Santhakumaran, S.; Brett, S.J.; Hinds, C.J.; Ashby, D.; Knight, J.C.; Gordon, A.C. Transcriptomic Signatures in Sepsis and a Differential Response to Steroids. From the VANISH Randomized Trial. *Am. J. Respir. Crit. Care Med.* **2019**, *199*, 980–986. [CrossRef] [PubMed]
13. Doya, K. Reinforcement learning: Computational theory and biological mechanisms. *HFSP J.* **2007**, *1*, 30–40. [CrossRef]
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. *Nat. Med.* **2018**, *24*, 1716–1720. [CrossRef]
15. 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. *J. Med. Internet Res.* **2020**, *22*, e18477. [CrossRef] [PubMed]
16. Liu, S.; Ngiam, K.Y.; Feng, M. Deep Reinforcement Learning for Clinical Decision Support: A Brief Survey. *arXiv* **2019**, arXiv:1907.09475.
17. Thoral, P.J.; Peppink, J.M.; Driessen, R.H.; Sijbrands, E.J.; Kompanje, E.J.; Kaplan, L.; Bailey, H.; Kesecioglu, J.; Cecconi, M.; Churpek, M.; et al. Sharing ICU Patient Data Responsibly Under the Society of Critical Care Medicine/European Society of Intensive Care Medicine Joint Data Science Collaboration: The Amsterdam University Medical Centers Database (AmsterdamUMCdb) Example. *Crit. Care Med.* **2021**, *49*, e563–e577. [CrossRef] [PubMed]

18. Lambden, S.; Laterre, P.F.; Levy, M.M.; Francois, B. The SOFA score—Development, utility and challenges of accurate assessment in clinical trials. *Crit. Care* **2019**, *23*, 374. [CrossRef]
19. Thoral, P.J.; Driessen, R.H.; Peppink, J.M. AmsterdamUMCdb Github Repository. 2020. Available online: https://github.com/AmsterdamUMC/AmsterdamUMCdb (accessed on 15 September 2021).
20. Shin, J.; Badgwell, T.A.; Liu, K.-H.; Lee, J.H. Reinforcement Learning—Overview of recent progress and implications for process control. *Comput. Chem. Eng.* **2019**, *127*, 282–294. [CrossRef]
21. Li, L.; Komorowski, M.; Faisal, A.A. The Actor Search Tree Critic (ASTC) for Off-Policy POMDP Learning in Medical Decision Making. *arXiv* **2018**, arXiv:1805.11548.
22. Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; et al. Soft Actor-Critic Algorithms and Applications. *arXiv* **2018**, arXiv:181205905.
23. Liu, D.; Ahmet, A.; Ward, L.; Krishnamoorthy, P.; Mandelcorn, E.D.; Leigh, R.; Brown, J.P.; Cohen, A.; Kim, H. A practical guide to the monitoring and management of the complications of systemic corticosteroid therapy. *Allergy Asthma Clin. Immunol.* **2013**, *9*, 30. [CrossRef] [PubMed]
24. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: Large-scale machine learning on heterogeneous systems. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, Savannah, GA, USA, 2–4 November 2015.
25. Thomas, P.; Theocharous, G.; Ghavamzadeh, M. High-Confidence Off-Policy Evaluation. In Proceedings of the AAAI Conference on Artificial Intelligence, Hollywood, FL, USA, 18–20 May 2015. [CrossRef]
26. Thomas, P.; Theocharous, G.; Ghavamzadeh, M. High Confidence Policy Improvement. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 7–9 July 2015; Proceedings of Machine Learning Research, PMLR. Francis, B., David, B., Eds.; MLResearch Press: San Francisco, CA, USA, 2015; Volume 37, pp. 2380–2388.
27. Montavon, G.; Binder, A.; Lapuschkin, S.; Samek, W.; Müller, K.-R. Layer-Wise Relevance Propagation: An Overview. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2019; pp. 193–209. [CrossRef]
28. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
29. Sligl, W.I.; Milner, J.D.A.; Sundar, S.; Mphatswe, W.; Majumdar, S.R. Safety and Efficacy of Corticosteroids for the Treatment of Septic Shock: A Systematic Review and Meta-Analysis. *Clin. Infect. Dis.* **2009**, *49*, 93–101. [CrossRef] [PubMed]
30. Vincent, J.-L. Steroids in sepsis: Another swing of the pendulum in our clinical trials. *Crit. Care* **2008**, *12*, 141. [CrossRef]
31. Sprung, C.L.; Annane, D.; Keh, D.; Moreno, R.; Singer, M.; Freivogel, K.; Weiss, Y.G.; Benbenishty, J.; Kalenka, A.; Forst, H.; et al. Hydrocortisone Therapy for Patients with Septic Shock. *New Engl. J. Med.* **2008**, *358*, 111–124. [CrossRef] [PubMed]
32. Venkatesh, B.; Finfer, S.; Cohen, J.; Rajbhandari, D.; Arabi, Y.; Bellomo, R.; Billot, L.; Correa, M.; Glass, P.; Harward, M.; et al. Adjunctive Glucocorticoid Therapy in Patients with Septic Shock. *New Engl. J. Med.* **2018**, *378*, 797–808. [CrossRef] [PubMed]
33. Shah, A.D.; MacCallum, N.S.; Harris, S.; Brealey, D.A.; Palmer, E.; Hetherington, J.; Shi, S.; Perez-Suarez, D.; Ercole, A.; Watkinson, P.J.; et al. Descriptors of Sepsis Using the Sepsis-3 Criteria: A Cohort Study in Critical Care Units Within the U.K. National Institute for Health Research Critical Care Health Informatics Collaborative. *Crit. Care Med.* **2021**, *49*, 1883. [CrossRef]
34. Rhee, C.; Murphy, M.V.; Li, L.; Platt, R.; Klompas, M.; for the Centers for Disease Control and Prevention Epicenters Program. Comparison of Trends in Sepsis Incidence and Coding Using Administrative Claims Versus Objective Clinical Data. *Clin. Infect. Dis.* **2015**, *60*, 88–95. [CrossRef]
35. Valik, J.K.; Ward, L.; Tanushi, H.; Müllersdorf, K.; Ternhag, A.; Aufwerber, E.; Färnert, A.; Johansson, A.F.; Mogensen, M.L.; Pickering, B.; et al. Validation of automated sepsis surveillance based on the Sepsis-3 clinical criteria against physician record review in a general hospital population: Observational study using electronic health records data. *BMJ Qual. Saf.* **2020**, *29*, 735–745. [CrossRef]
36. Wang, F.; Kaushal, R.; Khullar, D. Should Health Care Demand Interpretable Artificial Intelligence or Accept "Black Box" Medicine? *Ann. Intern. Med.* **2020**, *172*, 59–60. [CrossRef] [PubMed]
37. Brouwer, M.C.; McIntyre, P.; Prasad, K.; van de Beek, D. Corticosteroids for acute bacterial meningitis. *Cochrane Database Syst. Rev.* **2015**, *2015*, CD004405. [CrossRef] [PubMed]
38. Cunha, B.A.; Lortholary, O.; Cunha, C.B. Fever of unknown origin: A clinical approach. *Am. J. Med.* **2015**, *128*, 1138.e1–1138.e15. [CrossRef] [PubMed]
39. Teblick, A.; Peeters, B.; Langouche, L.; Van den Berghe, G. Adrenal function and dysfunction in critically ill patients. *Nat. Rev. Endocrinol.* **2019**, *15*, 417–427. [CrossRef]
40. Walker, B.R.; Yau, J.L.; Brett, L.P.; Seckl, J.R.; Monder, C.; Williams, B.C.; Edwards, C.R. 11 beta-hydroxysteroid dehydrogenase in vascular smooth muscle and heart: Implications for cardiovascular responses to glucocorticoids. *Endocrinology* **1991**, *129*, 3305–3312. [CrossRef]
41. Shoenfeld, Y.; Gurewich, Y.; Gallant, L.A.; Pinkhas, J. Prednisone-induced leukocytosis. *Am. J. Med.* **1981**, *71*, 773–778. [CrossRef]
42. Van de Sande, D.; van Genderen, M.E.; Huiskens, J.; Gommers, D.; van Bommel, J. Moving from bytes to bedside: A systematic review on the use of artificial intelligence in the intensive care unit. *Intensive Care Med.* **2021**, *47*, 750–760. [CrossRef]
43. Richard, S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction. *IEEE Trans. Neural Netw* **2015**, *9*, 1054.

44. Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; Zaremba, W. Openai gym. *arXiv* **2016**, arXiv:1606.01540.
45. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In Proceedings of the International Conference on Machine Learning, PMLR 2022, Virtual Event, 7–8 April 2022. [CrossRef]

Chapter 6

Optimized Renal Replacement Therapy Decisions in Intensive Care: A Reinforcement Learning Approach

Optimized Renal Replacement Therapy Decisions in Intensive Care: A Reinforcement Learning Approach

Lorenz Kapral

lorenz.kapral@lbg.ac.at

Medical University of Vienna: Medizinische Universitat Wien https://orcid.org/0000-0003-3316-9024

Razvan Bologheanu

Medical University of Vienna: Medizinische Universitat Wien

Mohammad Mahdi Azarbeik

Technical University of Vienna: Technische Universitat Wien

Aylin Bilir

Medical University of Vienna: Medizinische Universitat Wien

Richard Weiss

Technical University of Vienna: Technische Universitat Wien

Stefan Bartos

Medical University of Vienna: Medizinische Universitat Wien

Stefan Schaller

Medical University of Vienna: Medizinische Universitat Wien

Clemens Heitzinger

Technical University of Vienna: Technische Universitat Wien

Eva Schaden

Medical University of Vienna: Medizinische Universitat Wien

Oliver Kimberger

Medical University of Vienna: Medizinische Universitat Wien

Research Article

Keywords: Acute kidney injury (AKI), Renal replacement therapy (RRT), Reinforcement learning (RL), Intensive care, Clinical decision support (CDS)

Posted Date: March 27th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-6243566/v1

OPTIMIZED RENAL REPLACEMENT THERAPY DECISIONS IN INTENSIVE CARE: A REINFORCEMENT LEARNING APPROACH

Lorenz Kapral,[a,b,c] Razvan Bologheanu,[a] Mohammad Mahdi Azarbeik,[b,c] Aylin Bilir,[a,b] Richard Weiss,[c] Stefan Bartos,[a] Stefan Schaller,[a,d] Clemens Heitzinger,[c] Eva Schaden,[a,b,*] Oliver Kimberger [a,b,*]

[a]Medical University of Vienna, Department of Anaesthesia, Intensive Care Medicine and Pain Medicine, Clinical Division of General Anaesthesia and Intensive Care Medicine, Vienna, Austria, Währinger Gürtel 18-20, 1090 Vienna, Austria

[b]Ludwig Boltzmann Institute Digital Health and Patient Safety, Währinger Str. 104/10, 1180 Wien, Vienna, Austria

[c]Technical University Vienna, Department of Informatics, Research Unit Machine Learning, Favoritenstraße 9/11, 1040 Wien, Vienna, Austria

[d]Universitätsmedizin Berlin, Corporate Member of Freie Universität Berlin and Humboldt Universität zu Berlin, Department of Anaesthesiology and Intensive Care Medicine (CCM/CVK), Berlin, Germany

[*]These authors contributed equally to the work as the last authors.

Corresponding author: Lorenz Kapral
Medical University of Vienna
Department of Anaesthesia, Intensive Care Medicine and Pain Medicine
Währinger Gürtel 18-20
1090 Vienna
Austria
lorenz.kapral@lbg.ac.at

ABSTRACT

PURPOSE

Acute kidney injury (AKI) is highly prevalent in intensive care units (ICUs) and often requires renal replacement therapy (RRT). However, the optimal timing for initiating RRT remains controversial. The aim of this study was to develop a reinforcement learning (RL) model to support individualized RRT decision-making for critically ill AKI patients.

METHODS

We trained and validated our RL model using ICU data from two cohorts: the publicly available MIMIC-IV database and a dataset from the Medical University of Vienna (MUW). Patients with AKI of stage I or higher were included, and those with chronic kidney disease or prior kidney transplantation were excluded. We extracted 88 features, employing weighted K-means clustering for state definition. A Q-learning–based RL approach was applied, with off-policy evaluation to assess the policy's performance versus clinician decisions.

RESULTS

In both the MIMIC and MUW cohorts, the RL model demonstrated a high level of concordance (up to 98.5%) with clinicians but exhibited superior performance on key metrics. Notably, the model identified a reproducible patient subgroup with greater illness severity for whom earlier or more frequent RRT could improve outcomes, suggesting a beneficial role for AI-driven decision support.

CONCLUSIONS

Our RL model provides dynamic, data-driven recommendations for initiating and ceasing RRT, closely aligning with clinical practice and identifying high-risk patients who may benefit from earlier intervention.

TAKE-HOME MESSAGE

RL offers a promising approach to augment clinical decision-making for RRT timing, potentially improving outcomes for selected subgroups of critically ill AKI patients.

KEYWORDS

Acute kidney injury (AKI), Renal replacement therapy (RRT), Reinforcement learning (RL), Intensive care, Clinical decision support (CDS)

INTRODUCTION

Acute kidney injury (AKI) is a common and severe complication of critical illness characterized by a sudden decline in kidney function over hours to days. Up to 50% of critically ill patients develop AKI during their stay in the intensive care unit (ICU), and AKI is associated with worse clinician- and patient-reported outcomes [1]–[4]. Mortality rates among patients who develop AKI may exceed 50%, with those experiencing stage III AKI facing an 80% higher risk of mortality compared to non-AKI patients [5]–[7].

The immediate consequences of AKI include a decrease in urine output and disruption of acid–base and electrolyte homeostasis, potentially progressing to fluid overload with hypoxemia, severe acidosis, hyperkalemia, and uremic symptoms [8], [9]. Although blood purification techniques can effectively correct the biochemical disturbances accompanying AKI, renal replacement therapy (RRT) does not restore kidney function, and in the absence of absolute indications, the optimal strategy for RRT in AKI patients, remains unknown despite extensive studies.

The challenge in the optimal use of RRT is to balance the risks and costs associated with RRT against the severity and consequences of AKI. Although early RRT initiation was found to have no survival benefit in several high-quality trials, delaying RRT can also be associated with harm [10]–[13]. Furthermore, absolute indications for RRT do not correlate with AKI biomarkers or urine output [14], [15]. If elevated kidney function markers and oliguria cannot predict clinically relevant AKI complications, it follows that an RRT initiation strategy based on AKI stages and time windows will not effectively prevent these complications. Moreover, RRT discontinuation presents similar challenges, adding to the complexity of this conundrum [16].

As machine learning has been increasingly studied in the field of intensive care, there has been a growing interest in data-driven applications of machine learning to AKI [17]. Most research on this topic, however, focuses on early detection and prediction of AKI rather than on RRT initiation strategies [18], [19]. Reinforcement learning (RL) is a machine learning framework that focuses on sequential decision-making. Through iterative interactions with their environment, RL agents can learn optimal policies [20]. Unlike supervised learning, which relies on static, labeled datasets, RL dynamically adapts to evolving contexts by evaluating the long-term consequences of actions. This makes RL a particularly suitable paradigm for medical applications. By conceptualizing clinical decisions as Markov decision processes, RL algorithms, such as Q-learning [21] and policy gradients [22], can optimize interventions. This approach has proven transformative in critical care; for example, Komorowski et al. [23] leveraged RL to derive personalized sepsis treatment policies from electronic health records, reducing calculated mortality by dynamically adjusting vasopressor and fluid administration in response to patients' physiological trajectories. In the context of chronic disease management, RL has been shown to optimize interventions with delayed effects, such as insulin dosing for patients with diabetes [24] and dialysis scheduling for patients with renal failure [25]. Grolleau et al. introduced an RL-driven algorithm that analyzes patient data to recommend personalized RRT initiation thresholds, demonstrating improved survival rates in retrospective validation [26].

We hypothesize that the severity and progression of the underlying pathology, to a greater extent than biomarkers and complications, are determinants of AKI trajectories in the ICU, including the need for RRT. The aim of this study was to develop and validate an algorithm that can accurately suggest individualized RRT strategies for AKI patients by training an RL agent in an environment consisting of highly granular clinical data to reduce mortality.

METHODS

DATA SOURCES AND COHORT SELECTION

We developed and validated our RL algorithm using ICU patient data from two sources. For training and internal validation, we used the publicly available Medical Information Mart for Intensive Care IV (MIMIC-IV v3.1) database, which comprises 94,458 ICU stays from 65,366 individuals between 2008 and 2022 [27]. For external validation, we used a proprietary dataset from the Medical University of Vienna (MUW) and the University Hospital Vienna, one of Europe's largest teaching hospitals. Supplemental Material 1 illustrates the patient counts and sequential filtering steps for MIMIC-IV.

We included all ICU patients with AKI of stage I or higher according to the KDIGO criteria [28]. To reduce confounding due to long-term renal impairment, we excluded those with stage V chronic kidney disease or prior kidney transplantation. In accordance with these criteria, 54,285 patients from the MIMIC-IV cohort (2008-2022) and 10,219 patients from the MUW cohort (2016-2024) were included in the study. Table 1 shows the demographic and clinical characteristics of both populations.

FEATURE EXTRACTION AND PREPROCESSING

We extracted 88 clinically relevant features (patient demographics, vital signs, laboratory values, medications, etc.) from the MIMIC-IV database (see Supplementary Material 2). Missing data were imputed via forward filling, and outliers were capped at the 1st and 99th percentiles via an approach similar to that of Komorowski et al. [29]. All features were then standardized via z-score normalization and log transformation. The code basis of our work was a Python implementation[1] of the AI Clinician of Komorowski et al. [29].

Figure 1 illustrates a schematic of the modeling setup: Patient data were structured in 12-hour intervals up to 264 hours (1 day prior to and 10 days following RRT initiation). A random forest model, optimized to predict clinician-initiated RRT and patient outcomes, was trained to generate feature importance scores. These scores served as weights in a weighted K-means clustering algorithm, which was run on 80% of the data (with 20% reserved for testing). The optimal number of clusters and features was determined by comparing state-transition matrices from the training and test sets using the Kullback–Leibler (KL) divergence [30] and matrix norms [31]. The final clustering model was subsequently applied to the entire dataset, including the external MUW cohort, ensuring consistent state definitions across all analyses.

REINFORCEMENT LEARNING MODEL DEVELOPMENT

We developed a Q-learning–based RL [21] model to guide decisions regarding RRT initiation versus withholding in ICU patients, adhering to the TRIPOD-AI [32] reporting guidelines. Patient states were defined by clusters derived from weighted K-means clustering [33], and the reward structure allocated +100 for 90-day survival, −100 for mortality, and an additional penalty (ranging from 0 to −35) for RRT initiation. Each penalty value was evaluated by training 500 RL models to identify the best-performing policy (Figure 2).

The hyperparameters were tuned on a validation split (20% of training data). Convergence was assessed via weighted importance sampling (WIS) [34], which is an evaluation method that adjusts the influence of patient trajectories collected under standard clinical practice to estimate the performance of an RL algorithm. The model's performance was then evaluated internally on a held-out test set (20% of the data) and externally on an independent cohort (MUW). Additionally, the distribution correction estimation (DICE) [35] algorithm was applied to generate an unbiased second performance evaluation.

MODEL ASSESSMENT

[1] https://github.com/cmudig/AI-Clinician-MIMICIV/tree/main

We stratified patients into four groups based on whether RRT was initiated solely by clinicians, recommended solely by the AI, both, or neither. On the test set, we assessed the concordance between the RL model's recommendations and the clinicians' decisions. We quantified the proportion of patients receiving RRT under the AI-recommended policy, the frequency of AI-recommended RRT, and the decision variation stratified by Sequential Organ Failure Assessment (SOFA) score and ICU type (surgical, mixed, or medical).

Survival analysis [36], stratified by age and SOFA score, was performed to assess differences in 90-day mortality across groups. To identify the determinants of treatment decisions, we trained an additional random forest model [37] to evaluate feature importance for AI-recommended RRT versus clinician-initiated RRT.

RESULTS

MODEL SELECTION

We evaluated the WIS metric across various penalty levels for RRT initiation using the validation set and plotted the corresponding RRT initiation rates, as shown in Figure 2. The analysis suggests that higher penalty levels tend to be associated with lower WIS values (higher mortality).

We ultimately selected a penalty level of 22% of the reward associated with mortality for two key reasons. First, it yields a treatment rate that is similar to that of the MIMIC dataset, reflecting real-world constraints such as the limited availability of resources; this ensures that the model remains practical and implementable. Second, among the models with a treatment rate comparable to that of MIMIC, the 22% penalty model has the highest average WIS.

The optimal number of features was determined to be 40, and the optimal number of clusters was found to be 500.

PERFORMANCE

The evaluation results indicate that, on the MIMIC dataset, the AI model's 95% lower bound (86.0) is higher than the physicians' 95% upper bound (64.6). On the external MUW dataset, the AI's 95% lower bound (78.5) also exceeds the physicians' 95% upper bound (77.5). This is further supported by the DICE values: for MIMIC, the physician's DICE score is 62.3 versus the AI's 62.5; for MUW, the physician's DICE score is 70.5, compared to 71.6 under the AI policy. The difference between the scores of the clinicians' policies is probably due to the different cohorts and does not allow for a comparison between the hospitals.

COMPARISON OF RRT INITIATION

We compared the proportion of patients who received RRT under clinician-initiated protocols to the AI recommendations in both the MIMIC and MUW datasets. In MIMIC, clinicians initiated RRT in 3.6% of patients compared to 2.8% for the AI (98.5% concordance). In MUW, clinicians initiated RRT in 11.5% of patients, compared with 8.6% for the AI. The model predicted shorter RRT treatment durations (Figure 3), a finding that may be attributable to the retrospective study design.

In the MIMIC cohort, our model demonstrated a high level of concordance with clinicians regarding RRT initiation decisions (F1 score: 0.80). Among the 16,283 patients in the MIMIC test set, clinicians and the AI agreed on initiating RRT in 422 instances and on withholding it in 15,651 cases (see Table 1). Similarly, in the MUW cohort (F1 score: 0.81), concordance occurred in 9,847 of 10,219 patients; clinicians alone initiated RRT in 283 instances, compared to 89 recommendations made solely by the AI.

Across both test sets, the Clinician-Only RRT group was characterized by a higher prevalence of chronic comorbidities, including congestive heart failure, hypertension, and diabetes, along with lower bilirubin and elevated creatinine levels. In contrast, the AI-Only RRT recommendations targeted patients with higher SOFA scores, elevated blood pressure, reduced total urine output, an increased anion gap, elevated BUN, and increased WBC levels (see Table 1). Survival analyses (Figure 4) revealed that the "Neither" group (in which neither the AI nor a clinician recommended RRT initiation) consistently presented the lowest 90-day mortality, whereas the "AI-recommended only" group presented significantly higher mortality rates (Figure 4).

SUPPLEMENTARY ANALYSES

Further analyses, including action distributions, survival curves for the MUW cohort, a 3D clustering visualization, and feature importance ranking, were conducted for both MIMIC and MUW and are provided in Supplementary Materials 3-6.

DISCUSSION

We present an RL algorithm to guide RRT decisions in critically ill patients, developed using medical health data from the MIMIC-IV database and validated externally on data from MUW. The main finding of our study is that RL has the potential to guide RRT decisions for ICU patients with AKI. By integrating 40 routinely measured ICU variables through weighted K-means clustering, our RL model achieved 98.5% concordance with human clinicians and outperformed conventional approaches (as measured by WIS and DICE) in both internal and external validation. Updated every 12 hours, the model provides dynamic, data-driven recommendations for initiating or stopping RRT. These findings suggest that AI-driven policies can meaningfully augment clinical judgment, particularly by identifying patients who may benefit from an earlier start of RRT.

Another key insight from our analysis is that there exists a subgroup of patients for whom the model recommended treatment, but clinicians did not. These patients exhibited increased mortality, indicating that AI-recommended interventions could improve outcomes if validated prospectively. We further characterized these subgroups to identify which types of patients might benefit most from RRT.

A growing body of research supports the use of RL to guide decision-making in critical care. Komorowski et al. [23] introduced an RL-based framework and demonstrated how data-driven models could improve decision-making to treat patients with sepsis. In the context of AKI and RRT, Grolleau et al. [26] applied RL to the AKIKI trial data to optimize the initiation of RRT in the ICU, and Zhang et al. demonstrated the utility of RL in managing ICU-acquired AKI [38]. Subsequent investigations have explored the potential of RL in addressing CKD complications [39] and refining fluid management in hemodialysis [40].

Unlike Grolleau et al., we used real-world data for the validation dataset. Although the use of clinical trial data for secondary analyses is common, this approach has its downsides. The trial population may be less representative of the entire critical care population due to the predetermined sample size, the requirement for informed consent, and the strict inclusion criteria. When several trials with different inclusion criteria, designs, and interventions are combined, data heterogeneity can represent a source of bias. Clinical trial data, which are often limited in scope and granularity, may not align with secondary research objectives, particularly in AI studies that require extensive datasets with robust volumes of observations and data points. Additionally, our algorithm not only provides predictions for the initiation of RRT but also determines when RRT should be discontinued.

Notably, Grolleau et al. analyzed only nine variable-baseline characteristics: arterial blood pH, serum potassium, urine output, and blood urea nitrogen at 24 and 48 hours. This aligns with our identification of these parameters as key predictors (see Supplement Material 5). However, our analysis also highlights underappreciated variables, including platelet count and phosphorus. In contrast to the method of Grolleau et al., our methodology employs 88 variables, prioritizing the 40 that are most representative of patient condition. These variables are sampled more frequently, allowing higher resolution and therefore resulting in a more accurate representation of patient trajectories [26].

Our algorithm introduces several key methodological advancements to improve its reliability and applicability in real-world clinical settings. First, to ensure that our model remains accurate when applied to new patient populations, we implemented weighted K-means clustering to better categorize patient states, which decreased the KL divergence by approximately two-thirds compared with traditional techniques, outperforming the unweighted K-means clustering proposed by Komorowski et al. [23]. Second, to ensure that the model's performance assessments would be as objective as possible, we employed a dual evaluation strategy that combines two complementary statistical techniques (WIS and DICE); this reduces the risk of bias that may arise when a single evaluation method is used, as was the case in previous studies by Peine and Komorowski [23], [41]. Our approach aligns with best practices in machine learning for health care, as highlighted by Kaushik and Gottesman, who emphasized the importance of rigorous validation to ensure the robustness of AI-driven clinical decision tools [42], [43].

Furthermore, building on Liu's recommendation of a meaningful reward design, we introduced reward penalization, in which the initiation of RRT is penalized during training in a manner that reflects real-world constraints to systematically study how action frequencies (e.g., varying rates of RRT initiation) influence clinical outcomes [44]. By varying penalty weights (from low to high), we demonstrated that lower penalties reproduced European-style treatment rates, whereas higher penalties aligned with American-style protocols—although the model was trained exclusively on MIMIC data. This analysis also suggests that there is a light increase in mortality with increasing penalties (Figure 2).

The higher 90-day mortality among AI-recommended patients who did not receive RRT suggests that the model identified individuals with acute deterioration (e.g., elevated SOFA scores and reduced urine output) who might benefit from earlier intervention. Clinicians prioritized chronic comorbidities (e.g., heart failure and renal failure) and lowered bilirubin, potentially overlooking dynamic acuity. Elevated mortality in untreated AI-identified patients highlights an unmet therapeutic need and suggests that the model captured a high-risk cohort in which timely RRT could improve outcomes. While end-of-life decisions or unmodeled factors may explain some cases, the model's consistent identification of high-risk patient groups demonstrates its potential to benefit clinical decision-making.

Another strength of our study is the external validation using a dataset outside the US that differs from the training dataset. To date, only a few models have been validated in different geographical and cultural contexts, yet this approach leads to significantly more robust performance [45].

We acknowledge several limitations. First, our analysis relies on retrospective data from two centers with differing RRT practices—primarily continuous RRT in Vienna versus shorter RRT sessions in the United States—potentially introducing bias and limiting generalizability [46]. An important aspect of this limitation is the extent to which continuous RRT and intermittent techniques, such as slow, low-efficiency dialysis, are comparable. We divided patient trajectories into 12-hour steps, assuming that blood, dialysate, and replacement fluid flows accurately and fully characterize the techniques employed, but the effects of different modalities on hemodynamics and solute/fluid control are not accounted for.

Second, although weighted K-means clustering preserves more patient-level heterogeneity than standard clustering does, it still averages patient characteristics and may overlook rare but significant critical cases [47]. Third, the retrospective design inherently limits our evaluation: the AI-recommended actions were not actually executed, thus patient responses to these recommendations remain estimated [48]. Although we employed two state-of-the-art evaluation methods (WIS and DICE) to mitigate bias, unmeasured confounders could still influence performance. Finally, our model depends on data completeness and does not yet incorporate unstructured inputs (e.g., narrative notes) or intangible clinical considerations (e.g., end-of-life discussions). Prospective trials integrating AI-driven recommendations with clinical judgment are necessary to validate and refine these results.

Importantly, heterogeneous documentation practices across cohorts introduced variability in the reporting of comorbidities and patient characteristics, which were used exclusively for descriptive analyses; diagnoses were excluded from model training, whereas objective, computer-measured parameters such as vital signs remained unaffected by documentation variability.

In conclusion, this study demonstrates that an RL-based model can effectively support RRT decision-making for critically ill patients with AKI, offering higher overall performance than human clinicians and maintaining a high level of concordance overall. Notably, we identified a distinct subgroup of patients who did not receive RRT but were flagged by the AI as likely to benefit, suggesting that earlier or more targeted intervention could improve outcomes in this high-risk patient cohort. Through rigorous feature extraction, clustering, and off-policy evaluation, our model achieves robust performance across multiple datasets. However, further prospective validation and the incorporation of unstructured clinical data are imperative to ensure safe, transparent, and ethically responsible integration of AI-driven decision support into critical care.

DATA AVAILABILITY

The code is available at the following address: https://github.com/lorenzkap/RL4RRT.

REFERENCES

[1] M. Andonovic, J. P. Traynor, M. Shaw, M. A. B. Sim, P. B. Mark, and K. A. Puxty, "Short- and long-term outcomes of intensive care patients with acute kidney disease.," *EClinicalMedicine*, vol. 44, p. 101291, Feb. 2022, doi: 10.1016/j.eclinm.2022.101291.

[2] K. P. Mayer, V. M. Ortiz-Soriano, A. Kalantar, J. Lambert, P. E. Morris, and J. A. Neyra, "Acute kidney injury contributes to worse physical and quality of life outcomes in survivors of critical illness.," *BMC Nephrol.*, vol. 23, no. 1, p. 137, Apr. 2022, doi: 10.1186/s12882-022-02749-z.

[3] E. A. J. Hoste *et al.*, "Epidemiology of acute kidney injury in critically ill patients: the multinational AKI-EPI study.," *Intensive Care Med.*, vol. 41, no. 8, pp. 1411–1423, Aug. 2015, doi: 10.1007/s00134-015-3934-7.

[4] J. A. Kellum, N. Lameire, and KDIGO AKI Guideline Work Group, "Diagnosis, evaluation, and management of acute kidney injury: a KDIGO summary (Part 1).," *Crit. Care*, vol. 17, no. 1, p. 204, Feb. 2013, doi: 10.1186/cc11454.

[5] D.-H. Wang, J.-C. Zhao, X.-M. Xi, Y. Zheng, and W.-X. Li, "Attributable mortality of acute kidney injury among critically ill patients with sepsis: a multicenter, retrospective cohort study.," *BMC Nephrol.*, vol. 25, no. 1, p. 125, Apr. 2024, doi: 10.1186/s12882-024-03551-9.

[6] E. Alba Schmidt *et al.*, "Acute kidney injury predicts mortality in very elderly critically-ill patients.," *Eur. J. Intern. Med.*, vol. 127, pp. 119–125, Sep. 2024, doi: 10.1016/j.ejim.2024.05.007.

[7] R. Bellomo *et al.*, "Acute kidney injury in the ICU: from injury to recovery: reports from the 5th Paris International Conference.," *Ann. Intensive Care*, vol. 7, no. 1, p. 49, Dec. 2017, doi: 10.1186/s13613-017-0260-y.

[8] R. Wald *et al.*, "Fluid balance and renal replacement therapy initiation strategy: a secondary analysis of the STARRT-AKI trial.," *Crit. Care*, vol. 26, no. 1, p. 360, Nov. 2022, doi: 10.1186/s13054-022-04229-0.

[9] S. M. Bagshaw *et al.*, "Impact of renal-replacement therapy strategies on outcomes for patients with chronic kidney disease: a secondary analysis of the STARRT-AKI trial.," *Intensive Care Med.*, vol. 48, no. 12, pp. 1736–1750, Dec. 2022, doi: 10.1007/s00134-022-06912-w.

[10] B. A. Cooper *et al.*, "A randomized, controlled trial of early versus late initiation of dialysis.," *N. Engl. J. Med.*, vol. 363, no. 7, pp. 609–619, Aug. 2010, doi: 10.1056/NEJMoa1000552.

[11] STARRT-AKI Investigators *et al.*, "Timing of Initiation of Renal-Replacement Therapy in Acute Kidney Injury.," *N. Engl. J. Med.*, vol. 383, no. 3, pp. 240–251, Jul. 2020, doi: 10.1056/NEJMoa2000741.

[12] S. Gaudry *et al.*, "Initiation Strategies for Renal-Replacement Therapy in the Intensive Care Unit.," *N. Engl. J. Med.*, vol. 375, no. 2, pp. 122–133, Jul. 2016, doi: 10.1056/NEJMoa1603017.

[13] A. Zarbock *et al.*, "Effect of early vs delayed initiation of renal replacement therapy on mortality in critically ill patients with acute kidney injury: the ELAIN randomized clinical trial.," *JAMA*, vol. 315, no. 20, pp. 2190–2199, May 2016, doi: 10.1001/jama.2016.5828.

[14] T. P. Bleck, M. C. Smith, S. J. Pierre-Louis, J. J. Jares, J. Murray, and C. A. Hansen, "Neurologic complications of critical medical illnesses.," *Crit. Care Med.*, vol. 21, no. 1, pp. 98–103, Jan. 1993, doi: 10.1097/00003246-199301000-00019.

[15] R. W. Steiner, C. Coggins, and A. C. Carvalho, "Bleeding time in uremia: a useful test to assess clinical bleeding.," *Am. J. Hematol.*, vol. 7, no. 2, pp. 107–117, 1979, doi: 10.1002/ajh.2830070203.

[16] Y. P. Kelly, S. S. Waikar, and M. L. Mendu, "When to stop renal replacement therapy in anticipation of renal recovery in AKI: The need for consensus guidelines.," *Semin. Dial.*, vol. 32, no. 3, pp. 205–209, May 2019, doi: 10.1111/sdi.12773.

[17] Z. A. Miller and K. Dwyer, “Artificial Intelligence to Predict Chronic Kidney Disease Progression to Kidney Failure: A Narrative Review,” *Nephrology (Carlton)*, Jan. 2025.

[18] N. Hammouda and J. A. Neyra, “Can artificial intelligence assist in delivering continuous renal replacement therapy?,” *Adv. Chronic Kidney Dis.*, vol. 29, no. 5, pp. 439–449, Sep. 2022, doi: 10.1053/j.ackd.2022.08.001.

[19] T. T. Tran, G. Yun, and S. Kim, “Artificial intelligence and predictive models for early detection of acute kidney injury: transforming clinical practice.,” *BMC Nephrol.*, vol. 25, no. 1, p. 353, Oct. 2024, doi: 10.1186/s12882-024-03793-7.

[20] R. S. Sutton and A. G. Barto, “Reinforcement Learning: An Introduction (Solution Manual),” *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 1054–1054, 1998, doi: 10.1109/TNN.1998.712192.

[21] C. J. C. H. Watkins and P. Dayan, “Q-learning,” *Mach. Learn.*, vol. 8, no. 3–4, pp. 279–292, May 1992, doi: 10.1007/BF00992698.

[22] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, “Policy Gradient Methods for Reinforcement Learning with Function Approximation,” *Advances in Neural Information Processing Systems*, 1999.

[23] M. Komorowski, L. A. Celi, O. Badawi, A. C. Gordon, and A. A. Faisal, “The Artificial Intelligence Clinician learns optimal treatment strategies for sepsis in intensive care.,” *Nat. Med.*, vol. 24, no. 11, pp. 1716–1720, Nov. 2018, doi: 10.1038/s41591-018-0213-5.

[24] M. Oroojeni Mohammad Javad, S. O. Agboola, K. Jethwani, A. Zeid, and S. Kamarthi, “A Reinforcement Learning-Based Method for Management of Type 1 Diabetes: Exploratory Study.,” *JMIR Diabetes*, vol. 4, no. 3, p. e12905, Aug. 2019, doi: 10.2196/12905.

[25] C. Shi, D. Guan, and W. Yuan, “Deep learning preserving renal dialysis treatment recommendation,” in *2020 International Conference on Information Networking (ICOIN)*, Jan. 2020, pp. 49–54, doi: 10.1109/ICOIN48656.2020.9016472.

[26] F. Grolleau *et al.*, “Personalizing renal replacement therapy initiation in the intensive care unit: a reinforcement learning-based strategy with external validation on the AKIKI randomized controlled trials.,” *J. Am. Med. Inform. Assoc.*, vol. 31, no. 5, pp. 1074–1083, Apr. 2024, doi: 10.1093/jamia/ocae004.

[27] A. E. W. Johnson *et al.*, “MIMIC-IV, a freely accessible electronic health record dataset.,” *Sci. Data*, vol. 10, no. 1, p. 1, Jan. 2023, doi: 10.1038/s41597-022-01899-x.

[28] Kidney Disease: Improving Global Outcomes (KDIGO) CKD Work Group, “KDIGO 2024 clinical practice guideline for the evaluation and management of chronic kidney disease.,” *Kidney Int.*, vol. 105, no. 4S, pp. S117–S314, Apr. 2024, doi: 10.1016/j.kint.2023.10.018.

[29] L. Li, M. Komorowski, and A. A. Faisal, “The Actor Search Tree Critic (ASTC) for Off-Policy POMDP Learning in Medical Decision Making,” May 2018.

[30] S. Kullback and R. A. Leibler, “On information and sufficiency,” *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, Mar. 1951, doi: 10.1214/aoms/1177729694.

[31] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge: Cambridge University Press, 2012.

[32] G. S. Collins *et al.*, “TRIPOD+AI statement: updated guidance for reporting clinical prediction models that use regression or machine learning methods.,” *BMJ*, vol. 385, p. e078378, Apr. 2024, doi: 10.1136/bmj-2023-078378.

[33] S. Mannor *et al.*, “K-Means Clustering,” in *Encyclopedia of machine learning*, C. Sammut and G. I. Webb, Eds. Boston, MA: Springer US, 2010, pp. 563–564.

[34] A. R. Mahmood and H. P. Van Hasselt, “Weighted importance sampling for off-policy learning with linear function approximation,” *Advances in neural …*, 2014.

[35] “[2007.03438] Off-Policy Evaluation via the Regularized Lagrangian.” https://arxiv.org/abs/2007.03438 (accessed Jan. 10, 2025).

[36] E. L. Kaplan and P. Meier, “Nonparametric Estimation from Incomplete Observations,” *J. Am. Stat. Assoc.*, vol. 53, no. 282, pp. 457–481, Jun. 1958, doi: 10.1080/01621459.1958.10501452.

[37] S. J. Rigatti, “Random Forest.,” *J. Insur. Med.*, vol. 47, no. 1, pp. 31–39, 2017, doi: 10.17849/insm-47-01-31-39.1.

[38] H. Zhang *et al.*, “Reinforcement Learning-based Decision-making for Renal Replacement Therapy in ICU-acquired AKI Patients,” Jun. 2024.

[39] A. E. Gaweda, E. D. Lederer, and M. E. Brier, “Artificial intelligence-guided precision treatment of chronic kidney disease-mineral bone disorder.,” *CPT Pharmacometrics Syst. Pharmacol.*, vol. 11, no. 10, pp. 1305–1315, Oct. 2022, doi: 10.1002/psp4.12843.

[40] Z. Yang *et al.*, “Optimization of dry weight assessment in hemodialysis patients via reinforcement learning.,” *IEEE J. Biomed. Health Inform.*, vol. 26, no. 10, pp. 4880–4891, Oct. 2022, doi: 10.1109/JBHI.2022.3192021.

[41] A. Peine *et al.*, “Development and validation of a reinforcement learning algorithm to dynamically optimize mechanical ventilation in critical care.,” *npj Digital Med.*, vol. 4, no. 1, p. 32, Feb. 2021, doi: 10.1038/s41746-021-00388-6.

[42] O. Gottesman *et al.*, “Guidelines for reinforcement learning in healthcare.,” *Nat. Med.*, vol. 25, no. 1, pp. 16–18, Jan. 2019, doi: 10.1038/s41591-018-0310-5.

[43] P. Kaushik, S. Kummetha, and P. Moodley, “A conservative q-learning approach for handling distribution shift in sepsis treatment strategies,” *arXiv preprint arXiv …*, 2022.

[44] S. Liu, K. C. See, K. Y. Ngiam, L. A. Celi, X. Sun, and M. Feng, “Reinforcement learning for clinical decision support in critical care: comprehensive review.,” *J. Med. Internet Res.*, vol. 22, no. 7, p. e18477, Jul. 2020, doi: 10.2196/18477.

[45] P. Rockenschaub *et al.*, “The Impact of Multi-Institution Datasets on the Generalizability of Machine Learning Prediction Models in the ICU.,” *Crit. Care Med.*, vol. 52, no. 11, pp. 1710–1721, Nov. 2024, doi: 10.1097/CCM.0000000000006359.

[46] S. M. Bagshaw, L. R. Berthiaume, A. Delaney, and R. Bellomo, “Continuous versus intermittent renal replacement therapy for critically ill patients with acute kidney injury: a meta-analysis.,” *Crit. Care Med.*, vol. 36, no. 2, pp. 610–617, Feb. 2008, doi: 10.1097/01.CCM.0B013E3181611F552.

[47] A. E. Fohner *et al.*, “Assessing clinical heterogeneity in sepsis through treatment patterns and machine learning.,” *J. Am. Med. Inform. Assoc.*, vol. 26, no. 12, pp. 1466–1477, Dec. 2019, doi: 10.1093/jamia/ocz106.

[48] E. J. Topol, “High-performance medicine: the convergence of human and artificial intelligence.,” *Nat. Med.*, vol. 25, no. 1, pp. 44–56, Jan. 2019, doi: 10.1038/s41591-018-0300-7.

FUNDING

The authors received no financial support for the research, authorship, and/or publication of this article.

ACKNOWLEDGMENTS

The computational results presented have been achieved using the Vienna Scientific Cluster (VSC).

AUTHOR INFORMATION

AUTHORS AND AFFILIATIONS

Medical University of Vienna, Department of Anaesthesia, Intensive Care Medicine and Pain Medicine, Clinical Division of General Anaesthesia and Intensive Care Medicine, Vienna, & Ludwig Boltzmann Institute Digital Health and Patient Safety, Austria

Lorenz Kapral, Razvan Bologheanu, Aylin Bilir, Stefan Bartos, Eva Schaden, Oliver Kimberger

Technical University Vienna, Department of Informatics, Research Unit Machine Learning, Austria

Lorenz Kapral, Mohammad Mahdi Azarbeik, Richard Weiss, Clemens Heitzinger

Medical University of Vienna, Department of Anaesthesia, Intensive Care Medicine and Pain Medicine, Clinical Division of General Anaesthesia and Intensive Care Medicine, Vienna, & Charité - Universitätsmedizin Berlin, Corporate Member of Freie Universität Berlin and Humboldt Universität zu Berlin, Department of Anaesthesiology and Intensive Care Medicine (CCM/CVK), Berlin, Germany

Stefan J Schaller

CONTRIBUTIONS

Study conception and design: LK, RB, MMA, AB, ES, and OK; data curation and visualization: LK, MMA, RB, and SB; formal analyses: LK, MMA, and RW; methodology: LK, MMA, RW, and CH; supervision: SJS, CH, ES* and OK*; writing—original draft: LK, MMA, RB, and AB; writing—review and editing: all authors.

* These authors contributed equally to the work as the last authors.

CORRESPONDING AUTHOR

Correspondence to Lorenz Kapral.

ETHICS DECLARATIONS

This study was conducted in compliance with relevant ethical guidelines and received approval from the Institutional Review Board (IRB) of MUW (EK Nr. 2180/2023). All patient data were anonymized and handled in accordance with data security and privacy protocols, including compliance with the Health Insurance Portability and Accountability Act (HIPAA).

CONFLICTS OF INTEREST

SJS received grants and non-financial support from Reactive Robotics GmbH (Munich, Germany), ASP GmbH (Attendorn, Germany), STIMIT AG (Biel, Switzerland), ESICM (Geneva, Switzerland), grants, personal fees, and non-financial support from Fresenius Kabi Deutschland GmbH (Bad Homburg, Germany), grants from the Innovationsfond of The Federal Joint Committee (G-BA), personal fees from Springer Verlag GmbH (Vienna, Austria) for educational purposes and Advanz Pharma GmbH (Bielefeld, Germany), non-financial support from national and international societies (and their congress organizers) in the field of anesthesiology and intensive care medicine, outside the submitted work. Dr. Schaller holds stocks in small amounts from Alphabet Inc., Ascendis Pharma Inc., Bayer AG, and Siemens AG; these holdings have not affected any decisions regarding his research or this study. All other authors declare that they have no conflicts of interest.

FIGURES AND TABLES

TABLE 1. Distribution of comorbidities and vital parameters for all patients in the Medical Information Mart for Intensive Care IV (MIMIC IV, 2008-2022) and Medical University of Vienna (MUW, 2016-2024) datasets. Patients are categorized into four groups: Both renal replacement therapy (RRT), Neither RRT, Clinician-initiated RRT, and AI-recommended RRT—to highlight the distinct characteristics of each treatment category.

	MIMIC					MUW				
	All	**Both RRT**	**Neither RRT**	**Clinician-initiated RRT**	**AI-recommended RRT**	**All**	**Both RRT**	**Neither RRT**	**Clinician-initiated RRT**	**AI-recommended RRT**
Unique ICUs (n)	54285	370	15574	223	116	10219	788	9059	283	89
Unique ICU admissions (n)	41283	356	14079	221	110	9586	717	8534	252	83
Age, years (std)	65.5 (16.7)	61.4 (15.6)	65.7 (16.7)	63.6 (15.8)	67.0 (16.5)	60.0 (16.3)	60.4 (14.9)	59.9 (16.4)	60.3 (16.7)	57.0 (17.6)
Female sex (n (%))	24373 (44.9%)	141 (38.1%)	7072 (45.4%)	97 (43.5%)	45 (38.8%)	4180 (40.9%)	214 (27.2%)	3810 (42.1%)	117 (41.3%)	39 (43.8%)
Congestive heart failure	15806 (29.1%)	159 (43.0%)	4489 (28.8%)	122 (54.7%)	34 (29.3%)	1355 (13.3%)	129 (16.4%)	1166 (12.9%)	53 (18.7%)	7 (7.9%)
Hypertension	35269 (65.0%)	250 (67.6%)	10140 (65.1%)	176 (78.9%)	68 (58.6%)	4444 (43.5%)	359 (45.6%)	3939 (43.5%)	118 (41.7%)	28 (31.5%)
Chronic pulmonary	14024 (25.8%)	78 (21.1%)	4016 (25.8%)	58 (26.0%)	25 (21.6%)	1733 (17.0%)	162 (20.6%)	1493 (16.5%)	67 (23.7%)	11 (12.4%)
Diabetes	16408 (30.2%)	170 (45.9%)	4648 (29.8%)	115 (51.6%)	41 (35.3%)	1497 (14.6%)	161 (20.4%)	1279 (14.1%)	50 (17.7%)	7 (7.9%)
Renal failure	10580 (19.5%)	194 (52.4%)	2854 (18.3%)	149 (66.8%)	42 (36.2%)	905 (8.9%)	164 (20.8%)	666 (7.4%)	70 (24.7%)	5 (5.6%)
Liver disease	7487 (13.8%)	161 (43.5%)	1924 (12.4%)	61 (27.4%)	34 (29.3%)	460 (4.5%)	111 (14.1%)	317 (3.5%)	32 (11.3%)	0 (0.0%)
Cancer	7243 (13.3%)	27 (7.3%)	2094 (13.4%)	26 (11.7%)	21 (18.1%)	1112 (10.9%)	61 (7.7%)	1026 (11.3%)	20 (7.1%)	5 (5.6%)
Coagulopathy	12365 (22.8%)	227 (61.4%)	3372 (21.7%)	104 (46.6%)	65 (56.0%)	178 (1.7%)	27 (3.4%)	137 (1.5%)	10 (3.5%)	4 (4.5%)
Obesity	7256 (13.4%)	60 (16.2%)	2036 (13.1%)	40 (17.9%)	11 (9.5%)	2110 (20.6%)	210 (26.6%)	1854 (20.5%)	32 (11.3%)	14 (15.7%)
Fluid electrolyte	24320 (44.8%)	318 (85.9%)	6737 (43.3%)	171 (76.7%)	77 (66.4%)	4654 (45.5%)	555 (70.4%)	3877 (42.8%)	192 (67.8%)	30 (33.7%)
Alcohol abuse	4204 (7.7%)	62 (16.8%)	1155 (7.4%)	23 (10.3%)	17 (14.7%)	436 (4.3%)	40 (5.1%)	375 (4.1%)	15 (5.3%)	6 (6.7%)
Aids	284 (0.5%)	3 (0.8%)	73 (0.5%)	1 (0.4%)	0 (0.0%)	44 (0.4%)	3 (0.4%)	37 (0.4%)	4 (1.4%)	0 (0.0%)
Initial SOFA (std)	4.3 (2.5)	7.6 (3.1)	4.1 (2.4)	6.3 (2.6)	7.0 (3.1)	9.4 (4.0)	11.3 (3.3)	9.1 (3.9)	9.0 (3.1)	16.3 (6.1)
Mechanical ventilation	17759 (32.7%)	207 (55.9%)	4994 (32.1%)	97 (43.5%)	31 (26.7%)	7273 (71.2%)	669 (84.9%)	6302 (69.6%)	219 (77.4%)	83 (93.3%)
Vasopressors	0.1 (2.5)	0.1 (3.1)	0.1 (2.4)	0.1 (2.6)	0.1 (3.1)	1.8 (4.0)	10.1 (3.3)	0.9 (3.9)	6.7 (3.1)	1.8 (6.1)
Length of stay, days	4.0 (3.1)	8.3 (3.3)	3.8 (2.9)	6.7 (3.8)	4.9 (3.3)	5.5 (3.4)	8.6 (2.5)	5.2 (3.3)	8.2 (2.7)	6.9 (3.2)
Hospital mortality	7.2%	32.4%	5.1%	18.9%	15.4%	8.9%	36.0%	5.7%	29.0%	37.1%
90-day mortality	16.5%	35.7%	15.2%	34.1%	48.3%	15.7%	43.4%	12.5%	33.9%	39.3%
Heart rate, beats/min	85.1 (17.1)	89.7 (18.9)	85.0 (17.0)	87.2 (17.0)	88.6 (17.8)	79.4 (16.7)	87.3 (18.2)	78.5 (16.3)	87.0 (17.5)	83.1 (19.6)
Systolic blood pressure, mmHg	120.1 (17.9)	116.2 (19.0)	120.0 (17.8)	119.6 (20.5)	113.8 (15.5)	119.7 (18.2)	111.6 (18.4)	120.6 (18.0)	116.6 (19.6)	110.9 (18.5)
Mean blood pressure, mmHg	81.1 (12.6)	78.0 (13.0)	81.2 (12.6)	79.1 (13.8)	76.5 (11.1)	80.4 (11.6)	75.6 (10.5)	80.9 (11.6)	78.9 (12.4)	76.8 (11.5)
Respiratory rate, breaths/min	19.2 (4.2)	20.9 (5.4)	19.1 (4.2)	19.7 (4.6)	20.6 (4.7)	17.9 (7.1)	17.9 (7.1)	17.9 (7.2)	17.3 (7.0)	17.3 (7.4)
Temperature, °C	36.8 (0.6)	36.7 (0.8)	36.8 (0.6)	36.7 (0.6)	36.8 (0.5)	36.5 (0.8)	36.2 (0.9)	36.5 (0.7)	36.3 (0.8)	36.6 (1.0)
12-hour total output, mL	519.0 (574.0)	185.5 (313.5)	526.4 (568.7)	270.6 (451.3)	397.7 (494.4)	669.9 (617.3)	656.9 (1102.5)	674.0 (551.3)	620.4 (723.2)	524.1 (625.1)
RASS score	-0.8 (1.5)	-1.6 (1.9)	-0.7 (1.4)	-1.0 (1.4)	-0.6 (1.3)	-2.0 (2.0)	-2.9 (1.9)	-1.9 (2.0)	-2.4 (1.9)	-3.4 (1.7)
GCS score	13.3 (3.0)	11.4 (4.3)	13.4 (3.0)	12.9 (3.0)	13.7 (2.4)	8.9 (5.4)	7.4 (4.9)	9.1 (5.4)	5.7 (4.9)	6.4 (5.9)
Anion gap, mEq/L	13.6 (3.5)	17.3 (6.3)	13.5 (3.4)	15.7 (4.5)	15.4 (3.5)	12.4 (3.5)	14.2 (3.3)	12.3 (3.7)	13.6 (1.9)	11.1 (1.6)
Chloride, mEq/L	103.6 (5.7)	100.9 (7.1)	103.7 (5.6)	100.6 (7.1)	103.0 (5.4)	106.8 (4.8)	106.2 (5.6)	106.8 (4.7)	105.9 (5.0)	109.0 (5.5)
Cumulative balance, mL	3585.6 (12727.5)	6327.9 (15015.8)	3490.8 (12503.7)	4430.8 (9635.8)	3407.3 (12119.0)	2025.4 (1520.3)	2838.8 (2890.6)	1936.8 (1278.6)	2459.6 (1974.7)	2454.5 (2318.6)
Hematocrit, %	31.9 (5.6)	29.5 (5.9)	32.0 (5.6)	29.2 (5.5)	28.4 (5.8)	32.7 (5.6)	31.3 (5.4)	32.9 (5.6)	30.9 (5.4)	31.3 (5.8)
Total bilirubin, mg/dL	1.9 (3.8)	4.1 (7.5)	1.8 (3.4)	2.3 (4.7)	5.0 (7.7)	1.5 (2.6)	3.1 (4.7)	1.3 (2.1)	2.1 (2.8)	2.2 (2.9)
Phosphorus, mg/dL	3.6 (1.1)	4.7 (2.0)	3.5 (1.0)	4.6 (1.8)	4.1 (1.3)	3.9 (1.3)	5.2 (1.8)	3.7 (1.1)	4.7 (1.7)	4.3 (1.2)
Creatinine, mg/dL	1.4 (1.3)	3.4 (2.5)	1.3 (1.1)	3.2 (2.5)	2.1 (1.6)	1.3 (1.2)	2.5 (1.8)	1.1 (0.9)	2.2 (1.7)	1.5 (0.9)
BUN, mg/dL	25.9 (19.1)	46.3 (29.6)	25.1 (18.0)	46.3 (29.5)	40.0 (23.4)	22.6 (17.6)	40.0 (25.4)	19.8 (14.3)	36.2 (23.0)	28.9 (30.7)
Hemoglobin, g/dL	10.4 (1.9)	9.6 (2.0)	10.4 (1.9)	9.5 (1.9)	9.2 (1.9)	10.7 (1.9)	10.2 (1.9)	10.7 (1.9)	10.1 (1.8)	10.2 (1.9)
WBC count, ×10^3/µL	11.9 (7.9)	15.6 (17.6)	11.6 (5.5)	12.9 (7.3)	45.5 (67.4)	12.8 (7.4)	13.7 (9.5)	12.7 (7.0)	13.1 (9.3)	13.1 (8.7)
Platelet count, ×10^3/µL	212.4 (98.8)	183.9 (100.4)	214.0 (99.3)	201.0 (102.0)	162.7 (130.9)	203.7 (114.4)	177.2 (135.3)	208.9 (111.0)	153.9 (99.8)	209.0 (139.4)

ICU – intensive care unit; SOFA – Sequential Organ Failure Assessment; RASS – Richmond Agitation-Sedation Scale; GCS – Glasgow Coma Scale; BUN - Blood Urea Nitrogen; WBC – White Blood Cells.

Figures

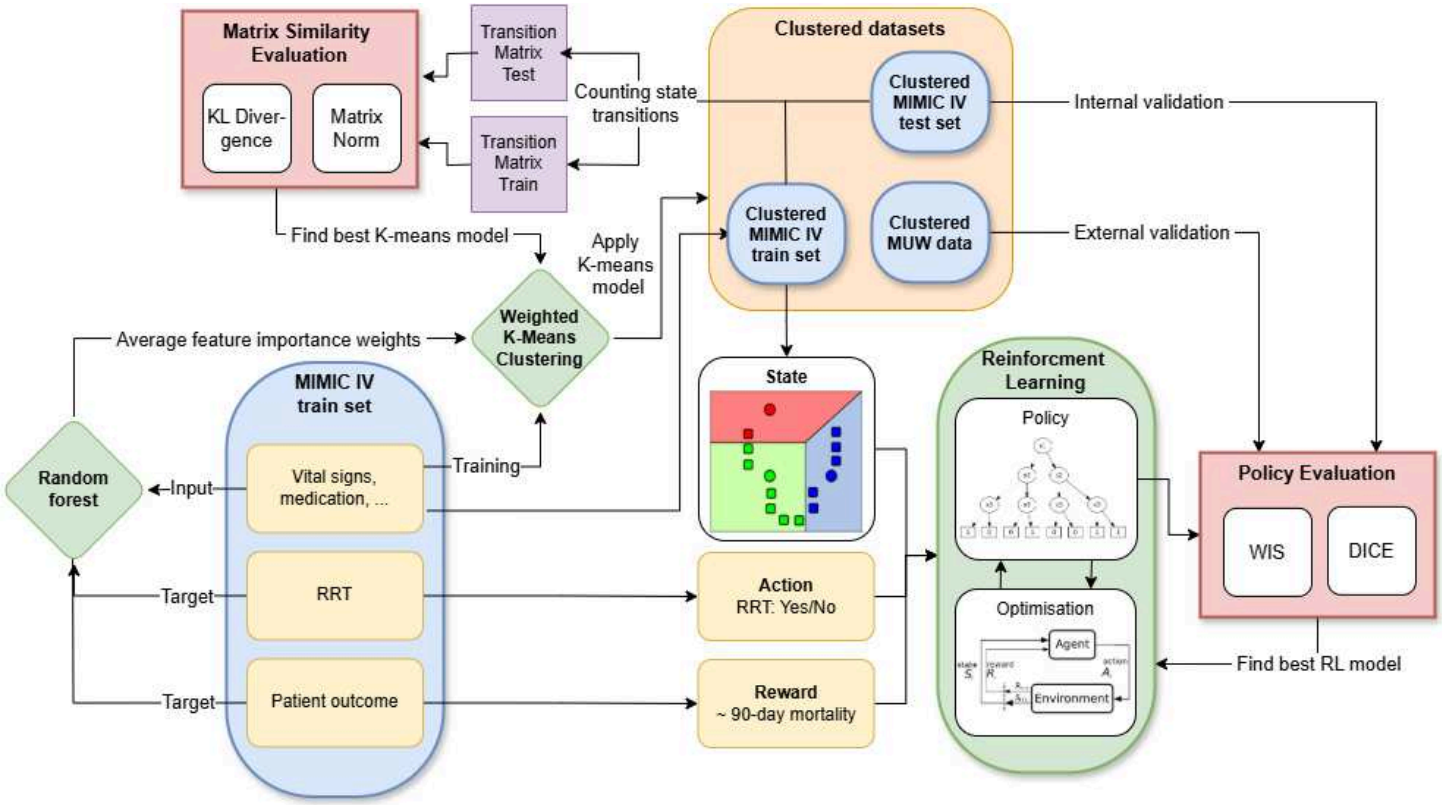


Figure 1

Reinforcement learning algorithm including evaluation and clustering optimization. Vital parameters, drug administration details, and renal replacement therapy (RRT) indications were extracted from the MIMIC dataset, which was split into training and test sets. A random forest model was trained on the training set to rank feature importance. Kullback-Leiber divergence and matrix norms were then used to compare train and test state-transition probabilities and determine the optimal number of features and clusters. Based on these results, a weighted k-means algorithm was applied to cluster all subsequent data. A reinforcement learning (RL) model was trained to optimize RRT timing based on a reward linked to 90-day mortality, and it was evaluated using off-policy evaluation methods (WIS and DICE). Finally, the RL model was validated using both an internal (MIMIC test) and an external (MUW) dataset.

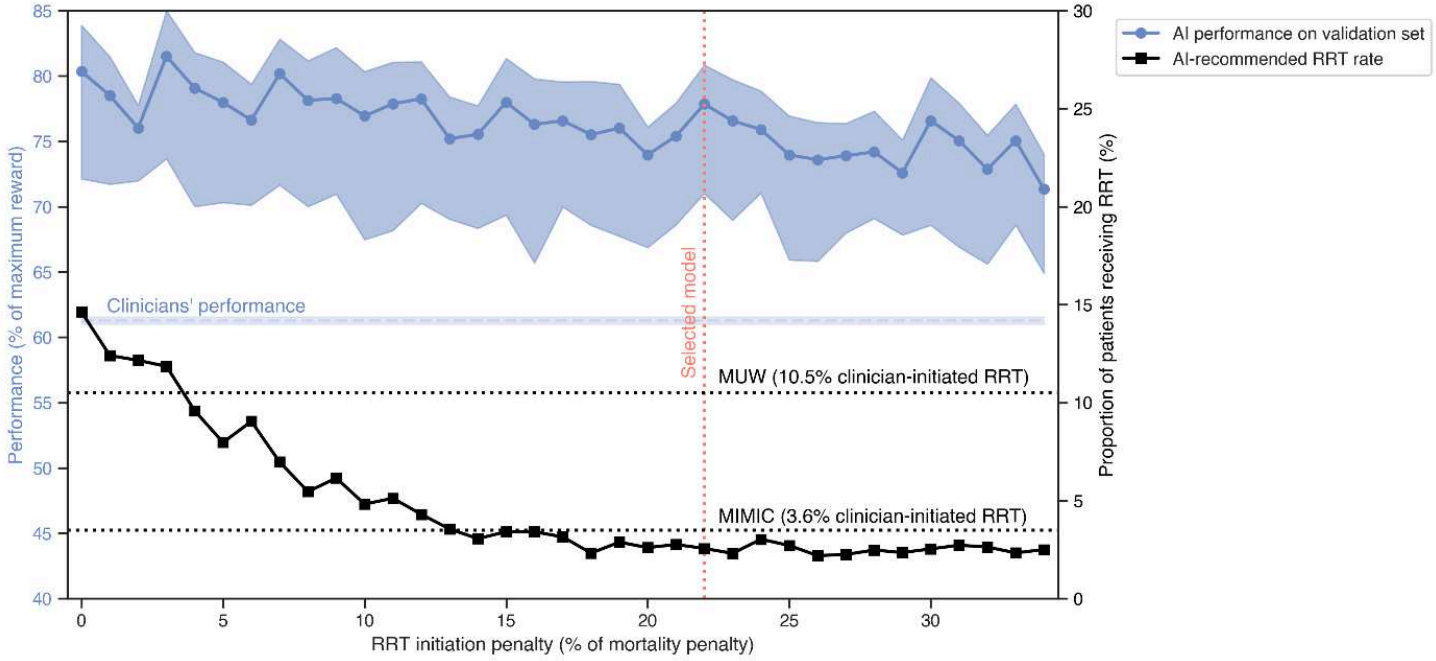


Figure 2

This figure illustrates the model's performance (dark blue) at varying levels of RRT initiation penalty within the validation set. The clinicians' performance is shown in light blue. The black curve represents the proportion of patients receiving AI-RRT recommendations as the penalty changes, and the black dashed lines show the average RRT treatment rates in the MIMIC and MUW test sets. The red dashed line represents the selected model, which we selected for 2 reasons: Firstly, the treatment rate yielded is analogous to that of the MIMIC dataset, thereby reflecting real-world constraints. Secondly, among the models exhibiting a treatment rate comparable to that of MIMIC, the 22% penalty model demonstrates the highest average WIS.

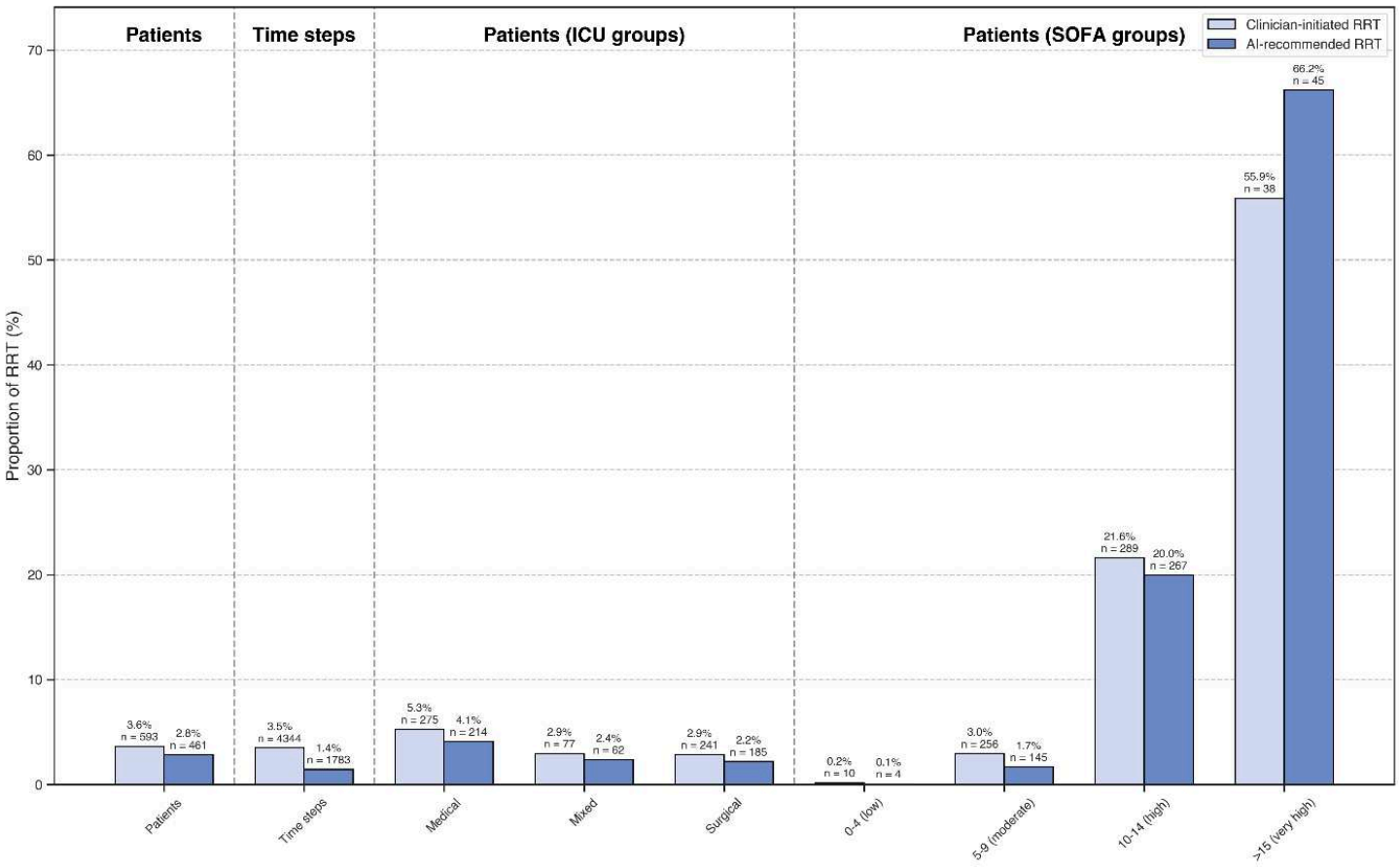


Figure 3

This figure demonstrates the proportion of AI-recommended renal replacement therapy (RRT) in the test set compared to actual clinical practice, displayed across all time steps and for all patients, stratified by ICU groups and SOFA score categories. The term "time steps" is used to denote the proportion of 12-hour steps in which the model or clinician recommended RRT in the data set. While the overall treatment strategies appear similar, the AI tends to recommend RRT less frequently and demonstrates a more pronounced response to increasing SOFA scores.

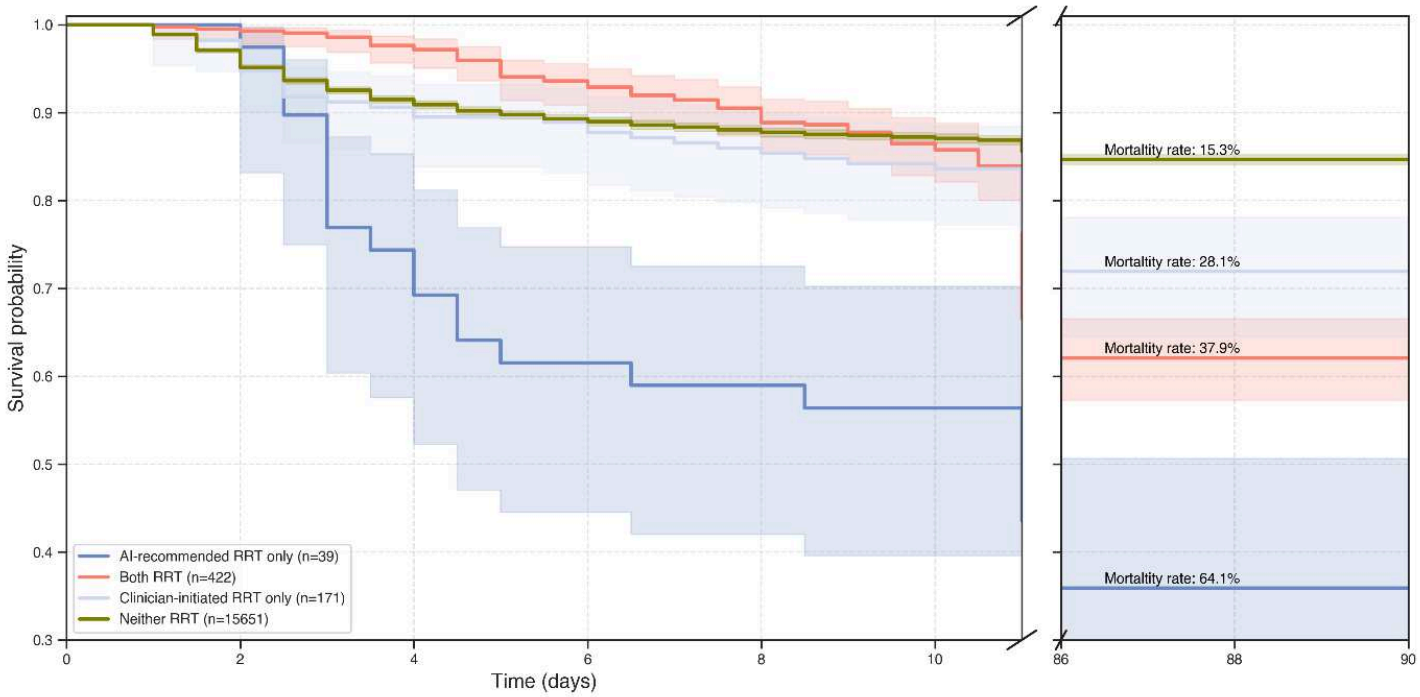


Figure 4

This figure presents the survival probability of four patient groups categorized by treatment type: AI-recommended RRT, Clinician-initiated RRT, Both AI- and Clinician-initiated RRT, and Neither RRT. Additionally, the 90-day mortality rate is shown on the right axis. The legend indicates the number of patients in each group. Notably, the group receiving only AI-recommended RRT exhibits higher mortality, suggesting a potential benefit from optimized treatment strategies.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- RRTfiguressupplement.docx
- TRIPODAI.pdf

Chapter 7

Development and External Validation of Temporal Fusion Transformer Models for Continuous Intraoperative Blood Pressure Forecasting

Development and external validation of temporal fusion transformer models for continuous intraoperative blood pressure forecasting

Lorenz Kapral,[a,b,c,e] Christoph Dibiasi,[a,b,e] Natasa Jeremic,[d] Stefan Bartos,[a,b] Sybille Behrens,[a,b] Aylin Bilir,[a,b] Clemens Heitzinger,[c] and Oliver Kimberger[a,b,*]

[a]Medical University of Vienna, Department of Anaesthesia, Intensive Care Medicine and Pain Medicine, Währinger Gürtel 18-20, Vienna 1090, Austria
[b]Ludwig Boltzmann Institute Digital Health and Patient Safety, Währinger Straße. 104/10, Vienna, 1180 Wien, Austria
[c]Technical University Vienna, Department of Informatics, Research Unit Machine Learning, Favoritenstraße 9/11, Vienna 1040 Wien, Austria
[d]Medical University of Vienna, Department of Ophthalmology and Optometry, Währinger Gürtel 18-20, Vienna 1090 Wien, Austria

eClinicalMedicine 2024;75: 102797
Published Online 30 August 2024
https://doi.org/10.1016/j.eclinm.2024.102797

Summary

Background During surgery, intraoperative hypotension is associated with postoperative morbidity and should therefore be avoided. Predicting the occurrence of hypotension in advance may allow timely interventions to prevent hypotension. Previous prediction models mostly use high-resolution waveform data, which is often not available.

Methods We utilised a novel temporal fusion transformer (TFT) algorithm to predict intraoperative blood pressure trajectories 7 min in advance. We trained the model with low-resolution data (sampled every 15 s) from 73,009 patients who were undergoing general anaesthesia for non-cardiothoracic surgery between January 1, 2017, and December 30, 2020, at the General Hospital of Vienna, Austria. The data set contained information on patient demographics, vital signs, medication, and ventilation. The model was evaluated using an internal (n = 8113) and external test set (n = 5065) obtained from the openly accessible Vital Signs Database.

Findings In the internal test set, the mean absolute error for predicting mean arterial blood pressure was 0.376 standard deviations—or 4 mmHg—and 0.622 standard deviations—or 7 mmHg—in the external test set. We also adapted the TFT model to binarily predict the occurrence of hypotension as defined by mean arterial blood pressure < 65 mmHg in the next one, three, five, and 7 min. Here, model discrimination was excellent, with a mean area under the receiver operating characteristic curve (AUROC) of 0.933 in the internal test set and 0.919 in the external test set.

Interpretation Our TFT model is capable of accurately forecasting intraoperative arterial blood pressure using only low-resolution data showing a low prediction error. When used for binary prediction of hypotension, we obtained excellent performance.

Funding No external funding.

Keywords: Intraoperative hypotension; Continuous prediction; Machine learning; Temporal fusion transformer; Haemodynamic monitoring; Blood pressure forecasting

Introduction

General anaesthesia for surgical interventions routinely involves administrating hypnotics and opioid analgesics to induce a loss of consciousness and tolerance to surgery. Commonly used anaesthetics interfere with the cardiovascular system by reducing cardiac inotropy and systemic vascular resistance, ultimately leading to hypotension.[1] This is further amplified by additional stressors such as hypovolemia, blood loss during surgery or intraoperative positioning (e.g., Trendelenburg position). Intraoperative hypotension, which is commonly defined as mean arterial pressure (MAP)

*Corresponding author. Medical University of Vienna, Department of Anaesthesia, Intensive Care Medicine and Pain Medicine, Währinger Gürtel 18-20, Vienna 1090, Austria.
E-mail address: oliver.kimberger@meduniwien.ac.at (O. Kimberger).
[e]Contributed equally.

Research in context

Evidence before this study

We searched PubMed database, from January 01, 2000, to June 01, 2024, for papers published in English using the terms "blood pressure", "prediction", "hypotension", and "forecasting". Our search yielded 131 results, indicating that intraoperative hypotension is a common occurrence during anaesthesia for non-cardiac surgery that is thought to be associated with postoperative morbidity. Predicting intraoperative hypotension before its occurrence could help anaesthesiologists to initiate prophylactic measures and thereby reduce the incidence of intraoperative hypotension. Existing machine learning algorithms mostly rely on the presence of high-resolution waveform data, which may not be available in many settings.

Added value of this study

We implemented the temporal fusion transformer (TFT) algorithm to predict intraoperative blood pressure trajectories using low-resolution data sampled at 15-s intervals from a large cohort of patients undergoing non-cardiothoracic surgery. We obtained robust predictive performance using low-resolution data, which renders our algorithm potentially more practical in clinical use. In addition to predicting continuous blood pressure values, the TFT model also provides binary predictions of hypotension with excellent discrimination and calibration. In contrast to previous studies, we incorporated data on intraoperative medication.

Implications of all the available evidence

The prediction algorithm developed by us is capable of accurately predicting intraoperative hypotension using low-resolution data. Implementation of our algorithm into clinical practice could help reduce the incidence of intraoperative hypotension, and thereby potentially reduce postoperative morbidity. Future research should prioritise integrating this predictive model into the clinical workflow and evaluating its impact on patient outcomes.

below 65 mmHg,[2] is potentially harmful, being linked to conditions such as myocardial injury,[3] kidney injury,[3,4] delirium[5] and postoperative nausea and vomiting.[6] Therefore, anaesthesiologists monitor patients under general anaesthesia and typically respond to hypotension when it occurs, for example, by administering vasopressors, by giving an intravenous fluid bolus, or by adjusting the depth of anaesthesia, in a *reactive* fashion.[3] However, the actual *prevention* of hypotensive episodes may be advantageous, yet this requires accurate *prediction* of hypotension in advance.

As a result, several tools for predicting intraoperative hypotension in advance have been developed through the use of conventional machine learning methods[7–9] and neural networks.[10–12] These models do not forecast actual MAP values but either make binary predictions (i.e., the patient will be hypotensive or not)[9,13] or provide a dimensionless number indicating the probability of hypotension.[14] In addition, those models are limited in terms of the input variables used for prediction because they mainly employ past vital signs and data on patient demographics. There is also discussion whether existing prediction models are superior to simply extrapolating the MAP trajectory.[15] Finally, most of the existing models require the use of high-quality arterial blood pressure waveform data and cannot be used when invasive arterial blood pressure monitoring is not in use.

There have been recent technical advances in time series data forecasting: The novel temporal fusion transformer (TFT) algorithm is an attention-based model that is designed for advanced multi-horizon forecasting.[16] It employs recurrent layers to effectively process short-term temporal patterns while using interpretable self-attention layers to understand long-term dependencies.[17] Hence, it can appropriately integrate static, time-stamped and time series data. In addition, the TFT algorithm can selectively focus on the relevant data points that are the most important for its forecast while filtering out nonessential elements.[18]

We hypothesised that the TFT algorithm would be well suited to predict intraoperative blood pressure trajectories and that it could be used to predict the occurrence of intraoperative hypotension, even with low resolution vital sign data. Therefore, we trained a TFT model to predict intraoperative MAP using a data set consisting of pre- and intraoperative data collected during routine patient care. To evaluate our model's performance, we assessed discrimination and calibration in both internal and external validation.

Methods

This retrospective observational study was performed after approval of the Ethics Committee of the Medical University of Vienna (reference number 2387/2020, January 19, 2021). Given the retrospective nature of the study, the requirement for informed consent was waived.

We screened all patients who underwent anaesthesia at the General Hospital of Vienna between January 1, 2017, and December 30, 2020, for eligibility. The General Hospital of Vienna is a tertiary academic hospital in Vienna, Austria. Anaesthesia is conducted by resident and consultant anaesthetists from the Department of Anaesthesia, Intensive Care Medicine and Pain Medicine of the Medical University of Vienna.

Patients older than 18 years at the time of surgery who had general anaesthesia performed for a diagnostic

or surgical intervention were included. We excluded patients who had cardiac, thoracic and/or vascular surgery and patients who had neuraxial, regional or local anaesthesia without general anaesthesia. We defined general anaesthesia as the administration of sedatives and invasive mechanical ventilation (either via laryngeal mask, endotracheal intubation or tracheostomy).

Preprocessing

We generated the data set from pre-, intra-, and postoperative data recorded for routine patient care in the patient data management system (IntelliSpace Critical Care and Anaesthesia, Philips Austria GmbH, Vienna, Austria). The following variables were static: age, sex, weight, American Society of Anaesthesiologists (ASA) score and surgical urgency (elective/urgent/emergency). The following variables were time series: heart rate (beats per minute), pulse rate (beats per minute), peripheral transcutaneous oxygen saturation (SpO2, %), non-invasive systolic, diastolic, and mean blood pressures (each in mmHg), invasive systolic, diastolic, and mean blood pressures (each mmHg) and end-tidal partial pressure of carbon dioxide (etCO2; mmHg). Anaesthetic agents, ventilation parameters and perfusion parameters were time-stamped but processed as time series; Supplemental Table S1 lists all the input variables.

The vital parameters heart rate, pulse rate, and SpO2 were available at a 15-s resolution. Invasive blood pressure was also available at a 15-s resolution while non-invasive blood pressure was available at a 3-min interval. We sampled all other time series variables including non-invasive blood pressure up to a 15-s resolution.

We grouped input features by type, differentiating between categorical and numerical variables as well as time-dependent and static variables. We checked the values of the input features for plausibility by analysing the maximum, minimum, and frequency distribution. Using the 'forward fill' method,[19] we replaced implausible and missing values, as detailed Supplemental Table S2, which lists their frequency of missingness. We scaled numerical variables to a standard deviation of 1 and a mean value of 0. Categorical variables underwent a one-hot encoding process, transforming each categorical variable into a dichotomous variable.

We split the complete data set into training set (70%), validation set (20%) and holdout test set (10%). This was done by randomly assigning patient IDs to each set. To prevent any potential leakage of data between different patients, we grouped each patient's data independently.

Model development

Google DeepMind's GitHub repository served as the foundational framework for the development of this TFT model.[20] We modified the model to handle data sets lacking future-known time points. To enhance the model's performance evaluation, we incorporated the metrics discussed in model evaluation. We integrated TensorBoard—a tool to visualise metrics—to track the training process.

We configured the TFT model to use the previous 32 values, corresponding to an input time interval of 8 min, for each variable to predict the subsequent 28 MAP values spanning 7 min. If the surgery duration was shorter than the combined duration of the input and output time intervals, the patients were excluded from training. When less than 8 min of history were available, we padded the oldest data point to form a complete input window for prediction.

We trained the model on the training set and evaluated its performance on the validation set every 10 epochs. To prevent overfitting, we stopped the training early if the error in the validation set did not reach a new optimal value for three consecutive iterations.

The TFT model was optimised using a 'RandomSearch' algorithm, focusing on the optimisation of several parameters, including batch size, learning rate, number of attention heads, number of hidden neurons, dropout rate and length of the input sequence; the final hyperparameters can be found in Supplemental Information S1.

Internal and external validation

We evaluated both MAP predictions themselves as well as binary predictions of whether hypotension will occur (defined by MAP < 65 mmHg). We used the holdout test set for internal validation and generated an external test set using the open public database 'Vital Signs DataBase' (VitalDB),[21] which contains high-resolution intraoperative data from 6388 patients. We transformed VitalDB data to match the format of our training data set.

Continuous MAP prediction

We evaluated continuous MAP predictions using two different metrics: mean squared error (MSE) and mean absolute error (MAE). MSE is the average of the squared differences between predicted and actual values, and MAE is the average of the absolute differences between predicted and actual values. MSE emphasises large errors, whereas MAE treats all errors equally, is easy to interpret and can be directly translated into units such as mmHg. We calculated the cumulative average of these metrics across all patients in the holdout test sets. This involved calculating the mean of all errors from the 28 predicted values for each data point of each patient in the test set.

Binary prediction of hypotension

To generate binary predictions of hypotension, we extracted the continuous MAP predictions at one, three, five, and 7 min (Fig. 1). We used these values to

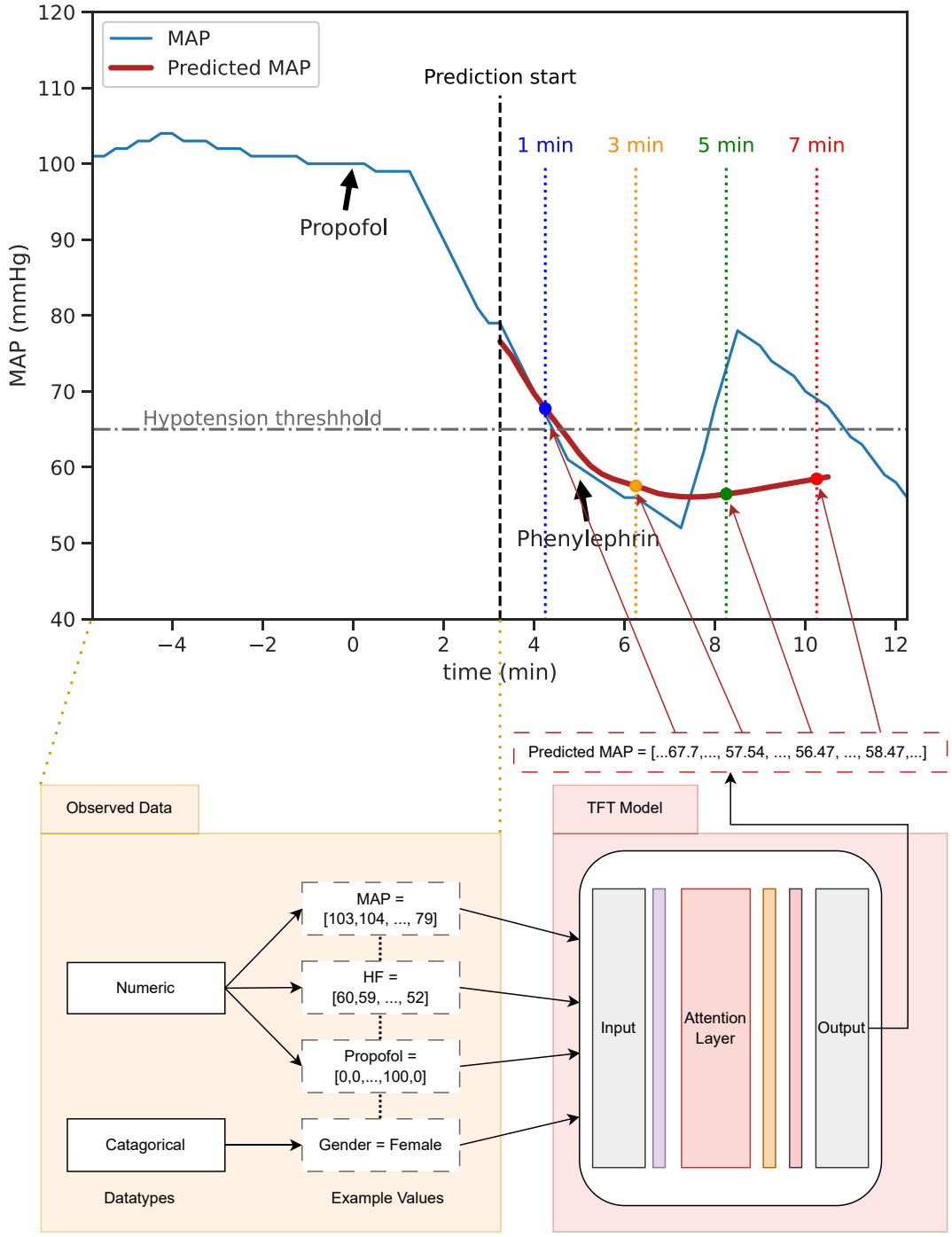


Fig. 1: Prediction of mean arterial pressure. A graphical representation of the temporal fusion transformer (TFT) model prediction process for mean arterial pressure (MAP). The top graph shows the observed MAP over time, the model predicted values and expected future MAP. The lower left section details the data input structure, separating real values and categorical data, with example values given. The bottom right shows a simplified architecture of the TFT model, highlighting the input, attention layer and output. The blue, orange, green and red lines indicate the specific time points used to assess hypotension, corresponding to predictions made 1, 3, 5, and 7 min into the future, respectively. The hypotension threshold was set at 65 mmHg. Propofol leads to arterial hypotension which is counteracted by the alpha-adrenergic agent phenylephrine. As the administration of phenylephrine occurs after the prediction start, it cannot be taken into account for forecasting MAP.

construct a binary prediction model that could estimate the likelihood of hypotension, defined as a MAP < 65 mmHg.

The model provided a range (lower and upper limits) for each of the 28 values. To evaluate the model using metrics such as which require probabilities rather than 'true' or 'false', we fit a Gaussian curve with the lower and upper limits. This allowed us to calculate probabilities.

For example, in the scenario shown in Fig. 1, the MAP values [68, 58, 57, 59] over four consecutive time points translated into a binary sequence of [false, true, true, true] with a decision threshold of 0.5, meaning that any probability greater than 50% was interpreted as a prediction of hypotension.

We calculated the following metrics for evaluating the binary hypotension predictions: Accuracy quantified the overall correctness of the model across all classes. Sensitivity (true positive rate) and specificity (true negative rate) measured the model's ability to correctly identify positive and negative cases, respectively. The positive predictive value (PPV) and negative predictive value (NPV) reflected the accuracy of positive and negative predictions. The area under the receiver operating characteristic curve (AUROC) assessed the ability of the model to discriminate between classes. Calibration slope, intercept and expected calibration error (ECE) together measured the agreement of the predicted probabilities with the observed outcomes and indicated the probabilistic accuracy of the model. To visualise these metrics, we plotted the receiver operating characteristic (ROC) curve and calibration plot.

Comparison with the XGB model

To establish a benchmark for the TFT model, we also used the extreme gradient boosting (XGB)[22] algorithm on the same training data set used for the TFT model as a way to train several models predicting the binary occurrence of hypotension at one, three, five, and 7 min.

We vectorised time-dependent variables into sequences and transformed them into unit scale. Separate XGB models were trained and optimised to predict occurrences of hypotension at one, three, five, and 7 min into the future.

We assessed the performance of the XGB model using the same metrics as those applied to the TFT model.

Interpretability

The attention mechanism allowed the model to focus on the most relevant aspects of the input data by assigning different levels of attention to different input parameters and acting as a filtering mechanism.[17]

To visualise the model's focus and determine the importance of temporal inputs, we computed the sum of the attention values assigned to all features at each time point. This allowed us to visualise the importance of each time step within the input sequence. In parallel, we assessed the weight of each input parameter across the data set by summing its attention values across all time points, thereby ranking its overall importance to the model's output.

In addition, we conducted experiments to investigate the influence of medication data on the behaviour of the model. After completing the training, we artificially manipulated the input data by omitting medication information and measured the effect of these differences on the predicted MAP over the next 3 min. This approach was only undertaken to provide insight into the extent to which the model was being influenced by medication data.

Statistical analysis

Because of patient privacy concerns and the regulations of the Medical University of Vienna, all data used to train the model are not available for public release in their current format. The external database, which was utilised for validation purposes, is openly available, enabling replication of the validation process.[21] The code for model training and evaluation is available (https://github.com/lorenzkap/MAP_TFT). We performed all calculations with R and Python 3.11.3, TensorFlow 2.12.0, and Scikit-learn 1.2.2.

Role of the funding source

This study was funded by institutional funds of the Medical University of Vienna, Department of Anaesthesia, Intensive Care Medicine and Pain Medicine and the Ludwig Boltzmann Institute Digital Health and Patient Safety.

Results

We screened data from 88,016 anaesthesia cases and included data from 81,122 cases in the final data set. The baseline characteristics of the anaesthesia cases analysed are given in Table 1. The internal data set was split randomly into training (70%), validation (20%), and holdout (10%) test sets, consisting of 56,785, 16,224 and 8113 cases. We tested the final algorithm in an external test set consisting of 5065 cases. Details of the external test set are given in Supplemental Table S3.

Continuous MAP prediction

We trained the TFT model to predict the continuous MAP trajectory for the next 7 min (Fig. 1; Supplemental Fig. S1), here by utilising 52 input features (Supplemental Table S2). In the internal test set, MSE was 0.405 standard deviations and MAE 0.376 standard deviations, corresponding to an average prediction error of 4 mmHg off the actual measurements. In the external test set, the average MSE was 1.165 standard deviations, and the average MAE was 0.622 standard deviations, or 7 mmHg. In both the internal and the external test sets, MAE was reduced when the forecast distance was lower and vice-versa (Fig. 2).

	N = 81,121
Age (years)	52 (34, 70)
Male sex (–)	35,730 (44%)
ASA score	
1	36,272 (27%)
2	36,272 (45%)
3	20,251 (25%)
4	2066 (2.5%)
5	730 (0.9%)
Surgical urgency (–)	
Elective	64,855 (80%)
Emergency	3459 (4.6%)
Urgent	12,808 (16%)
Duration of surgery (min)	132 (6, 296)
Surgical discipline	
General surgery	20,881 (26%)
Orthopaedics/Trauma surgery	16,098 (20%)
Plastic surgery	3153 (3.9%)
ENT	6110 (7.5%)
Maxillofacial surgery	3532 (4.4%)
Neurosurgery	5240 (6.5%)
Gynaecology	8905 (11%)
Obstetrics	5569 (6.9%)
Urology	6849 (8.4%)
Ophthalmology	4123 (5.1%)
Dermatology	656 (0.8%)
Undefined	6 (<0.1%)
1 Median (IQR); n (%)	

Table 1: **Patient characteristics: primary data set.**

A key feature of the TFT model was considering past data to predict blood pressure. The TFT model utilised medication data, for example, intravenous anaesthetics or vasopressors, to predict blood pressure. The model reacted to medication and its predictions became better when medication data was present (Fig. 3, Panel a). The model's attention mechanism can filter the data for more relevant time stamps (Fig. 3, Panel c). The top features selected for blood pressure predictions are shown in Fig. 3, Panel b, and the influence of the most common drugs on the prediction of the model in the data set is depicted in Fig. 3, Panel d.

Binary prediction of hypotension

We predicted the likelihood of blood pressure falling below 65 mmHg at one, three, five, and 7 min in the future by using specific quantiles of blood pressure predictions and compared these predictions with those from an XGB model (Fig. 4). In the internal test set, both the TFT and XGB models had area under the receiver operating characteristic curve (AUROC) scores above 0.9 (Table 2; Fig. 4) although the XGB model had slightly superior discrimination compared with the TFT model at the five- and 7-min marks. For both models, discrimination was reduced in the external test set. The TFT model was consistently able to discriminate between timepoints with and without hypotension when the forecast distance was increased from one to 7 min, but discrimination of the XGB model declined with increasing forecast distance, as evidenced by lower AUROC (Table 2; Supplemental Tables S4 and S5).

The calibration plots are shown in Fig. 5. The TFT model demonstrated an ECE ranging from 0.05 to 0.11 in the internal test set and from 0.06 to 0.08 in the external test set (Table 3). In both test sets, the TFT model had a calibration slope of less than one, indicating a tendency to overestimate the likelihood of hypotension (Fig. 5 Panel a, c; Supplemental Table S6). The XGB models showed good calibration in the internal test set (ECE < 0.03). However, the XGB models were poorly calibrated in the external test set (ECE >

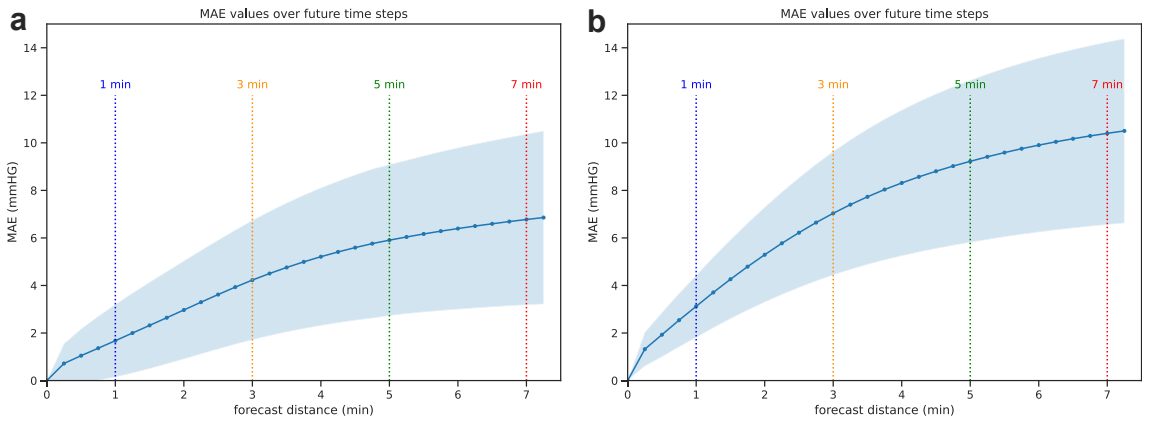


Fig. 2: Performance for continuous blood pressure prediction. Mean absolute error (MAE) of the temporal fusion transformer model for continuous prediction of intraoperative blood pressure in the internal (**a**) and external (**b**) test sets. The standard deviation of all MAEs is indicated by the lighter blue area.

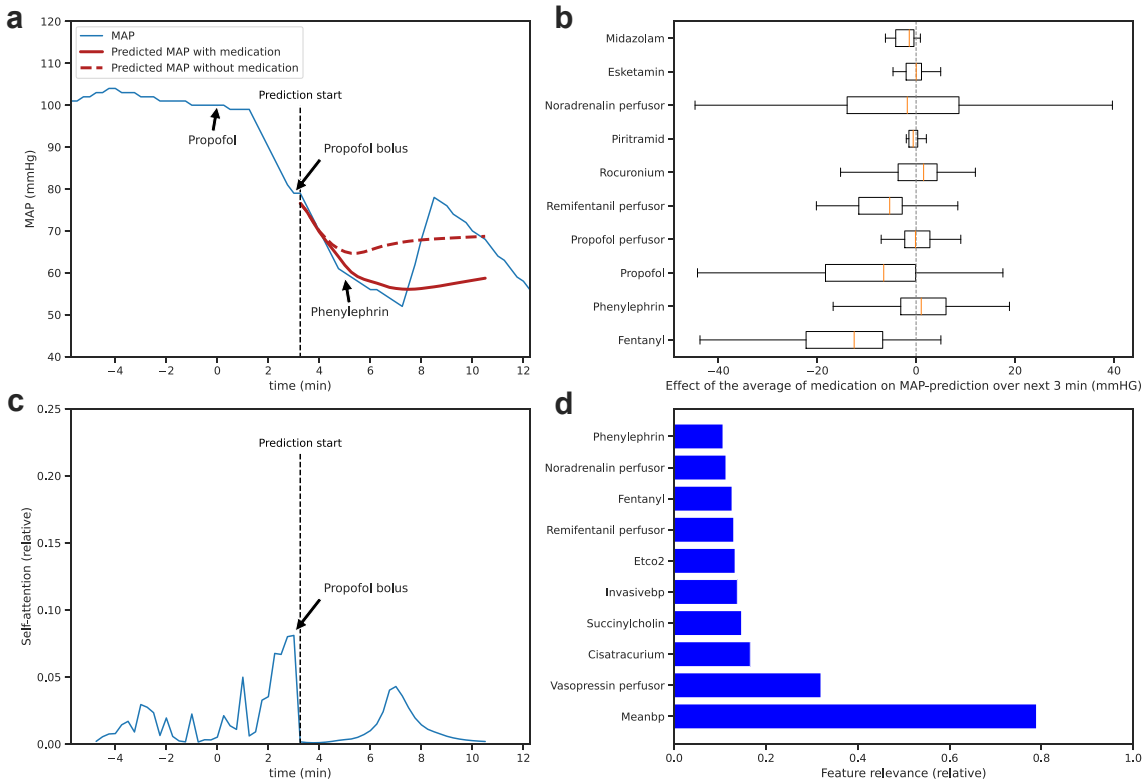


Fig. 3: Importance of medication and attention mechanism. (**a**) is a representative example of continuous mean arterial pressure (MAP) predictions using the temporal fusion transformer (TFT) model and shows that predicted MAP varies significantly when data on the use of propofol is included in the model vs. when these data are omitted. (**b**) shows the impact of the 10 most administered drugs on the predicted MAP over the next 3 min as predicted by the TFT model. The drugs are normalised by their average dosage because of their varying effects per milligram. (**c**) shows the relative importance of each time step in the model's input window. Self-attention in transformer models selectively focuses on the most relevant parts of the input. It highlights a significant increase in the importance of the time steps when propofol is administered, underscoring its influence on the model's output. (**d**) depicts the top 10 features that the model considers as being the most critical to its blood pressure predictions. Among these, historical MAP data stand out as the most influential factor for subsequent MAP predictions.

0.15 for 7 min) and overestimated the occurrence of hypotension (Fig. 5 Panels b, d; Supplemental Table S6).

Discussion

In the present study, we used the TFT algorithm to develop a predictive model 1) for continuously forecasting intraoperative blood pressure trajectories for the next 7 min and 2) for binarily predicting the occurrence of hypotension (defined as MAP below 65 mmHg) within the next one, three, five, and 7 min. We validated our model using internal and external test sets and found that our model predicted MAP with a low predictive error of 4 mmHg, respectively 7 mmHg in the internal and external test sets. Using the dichotomised TFT model, we obtained excellent discrimination and reasonable calibration for binary prediction of the occurrence of hypotension.

Predicting vital sign derangements, such as hypotension, is a well-established problem, and multiple studies from the field of anaesthesia and critical care medicine have used different study designs and computational algorithms to solve it.[8,10,11] For instance, Kendale et al. utilised multiple machine learning techniques to binarily predict the occurrence of hypotension (defined as a single MAP value below 55 mmHg) after the induction of anaesthesia. Jo et al. used deep learning models trained on high-resolution waveform data from VitalDB to predict intraoperative hypotension.[23] Hatib et al. and Davies et al. similarly applied deep learning to binarily predict hypotension (defined by them as MAP below 65 mmHg). Their model, which is commercially available[11,14], provides users with the hypotension prediction index (HPI), a dimensionless number ranging from 0 to 100, which indicates the likelihood of hypotension within the next 15 min. One of the key features distinguishing our TFT model from those works is the fact that our model directly predicts the course of MAP together with an uncertainty interval. In theory, this could be more readily interpretable by clinicians than an

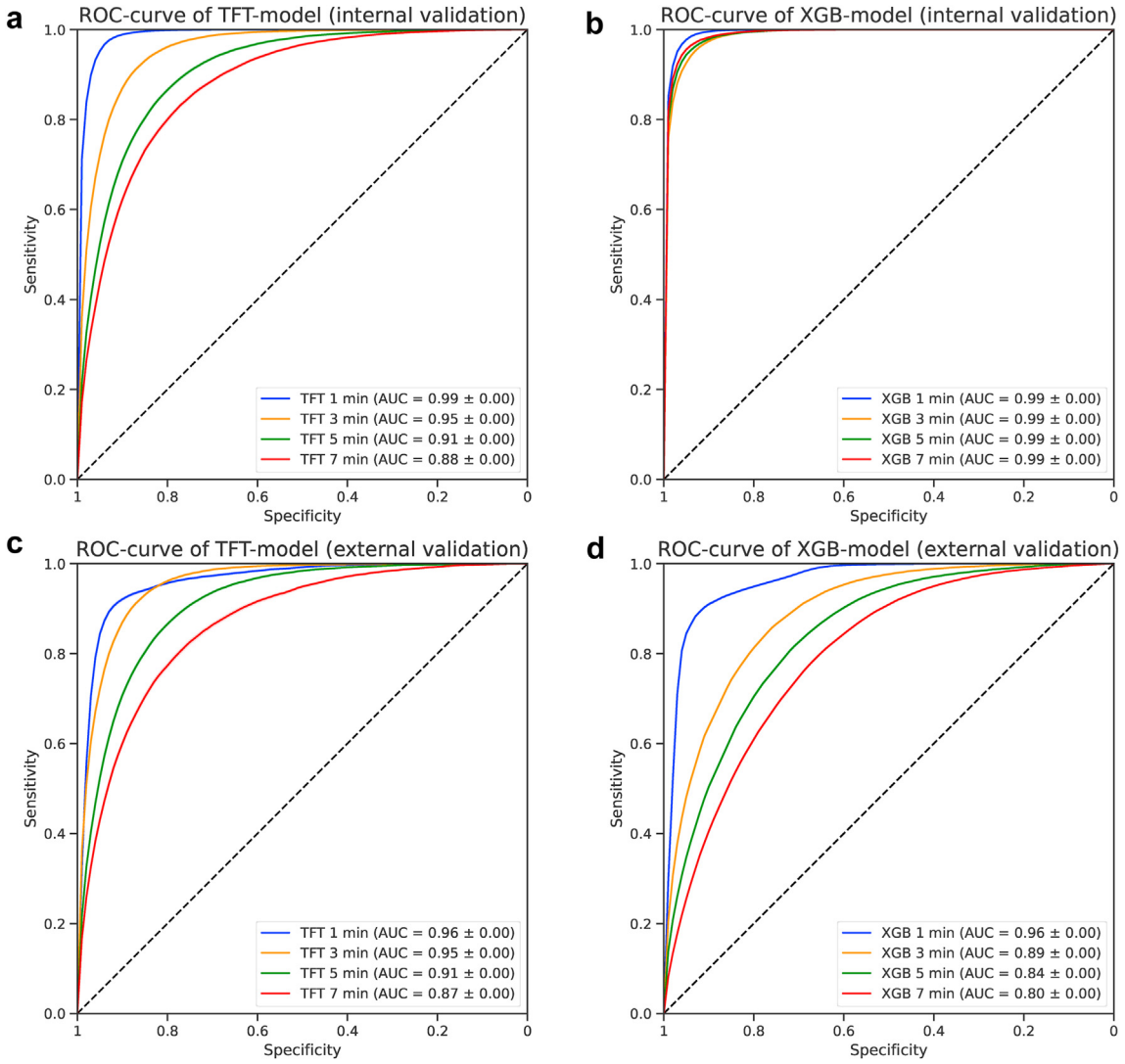


Fig. 4: Performance in binary prediction of hypotension. Receiver operating characteristic (ROC) curves for the temporal fusion transformer (TFT) model (**a, c**) and extreme gradient boosting (XGB) model (**b, d**) across time frames of 1, 3, 5 and 7 min for the prediction of hypotension in the internal and external validation. Area under receiver operating characteristic (AUC) values demonstrate high accuracy for both classifiers internally, with a modest decline externally. The TFT classifier shows a small drop in performance over time, while the XGB-classifier exhibits excellent internal but diminished external performance.

Forecast time	Internal validation		External validation	
	TFT	XGB	TFT	XGB
1 min	0.9883 (0.9880, 0.9886)	0.9941 (0.9939, 0.9943)	0.9598 (0.9590, 0.9607)	0.9607 (0.9602, 0.9612)
3 min	0.9544 (0.9536, 0.9553)	0.9874 (0.9871, 0.9878)	0.9453 (0.9444, 0.9462)	0.8909 (0.8900, 0.8918)
5 min	0.9095 (0.9083, 0.9107)	0.9893 (0.9890, 0.9896)	0.9032 (0.9017, 0.9046)	0.8420 (0.8409, 0.8432)
7 min	0.8800 (0.8785, 0.8816)	0.9908 (0.9905, 0.9910)	0.8667 (0.8648, 0.8686)	0.7981 (0.7968, 0.7994)

Area under the receiver operating characteristic (AUROC) of the temporal fusion transformer (TFT) and the extreme gradient boosting (XGB) model in internal and external validation. The forecast time indicates the time before a hypotensive event. The 95% confidence interval is indicated by the values within the brackets.

Table 2: **AUROC in internal and external test set.**

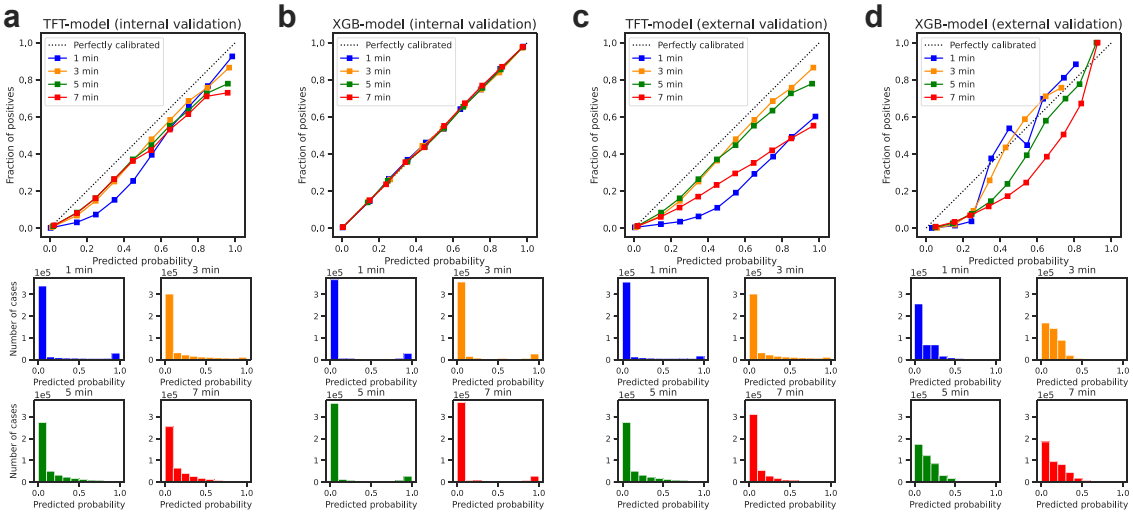


Fig. 5: Calibration curves for binary prediction of hypotension. Calibration curves for the temporal fusion transformer (TFT) model (**a, c**) and extreme gradient boosting (XGB) model (**b, d**) at 1, 3, 5 and 7 min for both internal and external validation for the prediction of hypotension. The graphs compare the predicted probabilities of positives against the actual proportion of positives, with the dotted line representing perfect calibration. The corresponding histograms below the calibration curves show the distribution of predicted probabilities at each time interval. The closer the calibration curve is to the dotted line, the better the calibration of the model. The histograms give an indication of the frequency and confidence of the classifier's predictions.

arbitrary index, and in addition, the length and severity of hypotension is easily visible, which is not the case with the models from Kendale et al. and with the HPI. The second key feature of the TFT model is the use of low-resolution data. In contrast to previous works, which have used waveform data that requires the invasive placement of an arterial line, we utilised vital signs data that is sampled every 15 s. Still, our TFT model showed similar discriminative performance compared with the HPI for predicting hypotension 5 min before it occurred, with an AUROC of 0.909 (TFT) compared with 0.926 (HPI). The higher specificity of the TFT model (0.960 compared with 0.858 for the HPI) could be advantageous because false positive predictions are less likely with our model, potentially reducing alarm fatigue. Notably, the HPI has recently been criticised for selection bias being present during training and validation, leading to data leakage which potentially falsely elevates its performance metrics.[24] As such, it has been suggested that HPI may not be superior to setting the mean blood pressure alarm threshold in the range of 70–75 mmHg.[24] Because our model utilises the TFT algorithm that is specifically designed for the prediction of time series data, we avoided such bias.

We conducted a series of tests on a range of models (LSTM, ARIMA, XGB, transformers) for the continuous MAP forecast. However, the results indicated that these models were not optimal. The TFT model demonstrated superior performance when applied to medical data. Consequently, we concentrated our efforts on the TFT model in our publication.

To the best of our knowledge, only the prediction model from Lee et al. could forecast continuous intraoperative blood pressure values similar to our TFT model; they applied a deep learning technique to predict blood pressure as well as hypotension (i.e., the

Forecast time	Internal validation		External validation	
	TFT	XGB	TFT	XGB
1 min	0.0259 (0.0255, 0.0264)	0.0008 (0.0004, 0.0011)	0.0529 (0.0524, 0.0533)	0.0791 (0.0788, 0.0793)
3 min	0.029 (0.0283, 0.0298)	0.0008 (0.0005, 0.0012)	0.0478 (0.0473, 0.0482)	0.1060 (0.1057, 0.1063)
5 min	0.0346 (0.0337, 0.0353)	0.0007 (0.0004, 0.0010)	0.0465 (0.0459, 0.0469)	0.1076 (0.1073, 0.1079)
7 min	0.0398 (0.0391, 0.0404)	0.0008 (0.0005, 0.0011)	0.0471 (0.0466, 0.0475)	0.1166 (0.1163, 0.1169)

Expected calibration error (ECE) of the temporal fusion transformer (TFT) and the extreme gradient boosting (XGB) model in the internal and external validation. The forecast time indicates the time before a hypotensive event. Low values represent a low error, thus better calibration. The 95% confidence interval is indicated by the values within the brackets.

Table 3: **Expected calibration error in the internal and external test sets.**

occurrence of blood pressure below 65 mmHg) within the next 5, 10, and 15 min using data from VitalDB, the database we used for external validation.[12] However, in the present study, we obtained lower predictive errors than in their study (MAE 4 mmHg in the internal test set and 7 mmHg in the external test set vs. 7 mmHg in the study from Lee et al.), even though we utilised lower-resolution data (sampled once every 15 s) as opposed to high-quality waveform data. In addition, their model was limited to predicting a single MAP value, whereas our model predicted an entire curve consisting of 28 different values, which can facilitate easier interpretation in the operating room. The TFT model also incorporates data on administered medication, such as hypnotics, analgesics and vasoactive agents, intraoperative ventilation parameters and intraoperative positioning. These features set our model apart from previous studies and are—in our opinion—the most important factor explaining the model's good performance. Our analysis also showed that data on administered medications were critical for the TFT model in predicting the blood pressure trajectory. Data on the use of propofol were especially used to improve MAP predictions, and the predictive error increases, for example, when data on the use of propofol were missing.

Fig. 3, Panel b, illustrates the directional influence of commonly administered drugs on blood pressure. The graph, created by excluding these drugs from the test set and analysing the prediction curves from Fig. 3, Panel a, shows an expected decrease in blood pressure when fentanyl or propofol are administered; however, the effect of noradrenaline varies widely, despite its known pressure-increasing effect. This variability may be attributed to patients entering the dataset with an active noradrenaline perfusor or the fact that the noradrenaline perfusor is often initiated and adjusted early to stabilise blood pressure, then maintained at a consistent level, resulting in minimal fluctuations during surgery. This may mask the actual influence of noradrenaline on blood pressure. Another potential use case of our TFT model could be the calculation of the 'optimal' dose of hypnotics/analgesics during the induction of anaesthesia. Furthermore, the black box problem of machine learning algorithms was alleviated by indicating the probability of the occurrence of hypotension as well as the time-resolved representation (Fig. 3, Panel d) of the essential features for decision-making.[25] For example, the model primarily uses the past MAP-values (Fig. 3, Panel d) for predicting MAP. Furthermore, it can identify significant occurrences such as the administration of propofol (see Fig. 3, Panel c).

To facilitate a comparison with previous studies, we used the results of the TFT model for binary predictions of the occurrence of hypotension. The discrimination of our model was superior to previously published works.[8,11,12] The generalisability of an algorithm was a persistent challenge that complicated the implementation of machine learning algorithms in clinical practice.[26] Our approach to predicting hypotension by directly calculating the MAP curve rather than providing an index offered additional robustness, as confirmed by external validation. Compared with the XGB models, which have previously been shown to have excellent performance in binary classification tasks, such as predicting hypotension[13,27] trained on the same data set, our model demonstrated greater robustness. This was evidenced by its superior performance on the external test set, even though the XGB models performed better on the internal test set and were trained on simpler tasks (hypotension: yes/no).

Although the model was reasonably calibrated in the internal test set, it overestimated the occurrence of hypotension in the external test set (Fig. 5). Miscalibration is a common phenomenon when predictive models are tested in a population that they were not developed in28, highlighting that predictive models should be carefully tested prior to implementation into clinical practise.[29] However, this overestimation is not necessarily an error of the TFT model but is rather a reflection that the model is not anticipating future medical interventions, even though we trained the model on retrospective surgical cases in which clinicians intervened during adverse events. For instance, if the model detected a potential drop in blood pressure, it could predict the onset of hypotension. However, in an actual OR scenario, clinicians often intervene to prevent such events. Therefore, a 'good' model should overestimate the likelihood of hypotension because it does not know these interventions at the time of prediction and therefore cannot and should not take them into account. In addition, these concerns were alleviated by the fact that our model could directly output blood pressure values.

Our study has several strengths and limitations. First, TFT is a state-of-the-art, novel algorithm that can utilise data on administered medication, a factor plausibly related to the occurrence of hypotension. We assembled a large and diverse patient cohort and had surgical cases from many specialties. We adhered to the transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD) guidelines[30] for the development and validation of predictive models and performed internal and external validation. However, the present retrospective study used data recorded for routine patient care, which likely introduced errors in our data set. Some data highly relevant for changes in blood pressure were not captured in our data set, such as bleeding, surgical compression of blood vessels or incorrectly documented medication regarding timing of data entry. Similarly, VitalDB lacks information on bolus drugs, which affects the performance of the models in the external validation. A 15-s sampling interval was employed to benefit

from higher resolution data and accurately time the effects of medication. However, this may result in inaccuracies due to the mismatch with standard 3-min blood pressure measurements. Although this approach offers increased detail, the use of forward filled values between actual measurements may impact the performance of the model and introduce noise into the learning process. In addition, medical interventions such as administration of vasopressors in response to hypotension were captured in our data set, which may have biased the TFT model towards an expectation of these interventions. Due to the extensive training time requirements, cross-validation was not employed to train the TFT model, which may have an impact on the final results' accuracy. While the TFT model performs well in continuous prediction tasks, the XGB model demonstrated superior results in binary predictions during internal validation, highlighting the importance of selecting the appropriate model for specific needs.

In summary, we applied the novel TFT algorithm to predict intraoperative blood pressure trajectories for the upcoming 7 min. Our model used easily obtainable input data available during routine care—most importantly, data on intraoperatively administered medications—and only required low-resolution data, which can be obtained without the placement of an arterial line. We obtained a low predictive error for continuous blood pressure predictions and—regarding the binary prediction of hypotension—and excellent discrimination with reasonable calibration. Future studies should investigate how our prediction model could be integrated into the anaesthesiologist's workflow and how this would affect patient outcomes.

Contributors

L.K., C.D., N.J., C.H., and O.K. were responsible for the conceptualization of the paper. L.K., C.D., and S.B.1 (Stefan Bartos) accessed and verified the data. The investigation was conducted by L.K., C.D., N.J., S.B.1, S.B.2 (Sybille Behrens), A.B., C.H., and O.K. Methodology was developed by L.K., C.D., N.J., C.H., and O.K. Software was developed by L.K. and N.J. Visualization was done by L.K. The original draft was written by L.K., C.D., N.J., S.B.2, and A.B. Supervision was provided by C.H. and O.K., who also reviewed and edited the paper. All authors agree to be fully accountable for ensuring the integrity and accuracy of the work and have read and approved the final manuscript. The corresponding author had full access to all data in the study and assumed final responsibility for the decision to submit the manuscript for publication.

Data sharing statement

The external database, which was utilised for validation purposes, is openly available, enabling replication of the validation process.[21] The code for model training and evaluation is available (https://github.com/lorenzkap/MAP_TFT).

Declaration of interests

We declare no competing interests.

Acknowledgements

The computational results presented have been achieved using the Vienna Scientific Cluster (VSC).

Appendix A. Supplementary data

Supplementary data related to this article can be found at https://doi.org/10.1016/j.eclinm.2024.102797.

References

1 Jor O, Maca J, Koutna J, et al. Hypotension after induction of general anesthesia: occurrence, risk factors, and therapy. A prospective multicentre observational study. *J Anesth*. 2018;32:673–680.
2 Wesselink EM, Kappen TH, Torn HM, Slooter AJC, van Klei WA. Intraoperative hypotension and the risk of postoperative adverse outcomes: a systematic review. *Br J Anaesth*. 2018;121:706–721.
3 Sessler DI, Bloomstone JA, Aronson S, et al. Perioperative Quality Initiative consensus statement on intraoperative blood pressure, risk and outcomes for elective surgery. *Br J Anaesth*. 2019;122:563–574.
4 Nadim MK, Forni LG, Bihorac A, et al. Cardiac and vascular surgery-associated acute kidney injury: the 20th international consensus conference of the ADQI (acute disease quality initiative) group. *J Am Heart Assoc*. 2018;7:e008834.
5 Wachtendorf LJ, Azimaraghi O, Santer P, et al. Association between intraoperative arterial hypotension and postoperative delirium after noncardiac surgery: a retrospective multicenter cohort stud y. *Anesth Analg*. 2022;134:822–833.
6 Maleczek M, Laxar D, Geroldinger A, Kimberger O. Intraoperative hypotension is associated with postoperative nausea and vomiting in the PACU: a retrospective database analysis. *J Clin Med*. 2023;12:2009.
7 Kang AR, Lee J, Jung W, et al. Development of a prediction model for hypotension after induction of anesthesia using machine learning. *PLoS One*. 2020;15:e0231172.
8 Kendale S, Kulkarni P, Rosenberg AD, Wang J. Supervised machine-learning predictive analytics for prediction of post-induction hypotension. *Anesthesiology*. 2018;129:675–688.
9 Solomon SC, Saxena RC, Neradilek MB, et al. Forecasting a crisis: machine-learning models predict occurrence of intraoperative bradycardia associated with hypotension. *Anesth Analg*. 2020;130:1201–1210.
10 Hatib F, Zhongping J, Sai B, et al. Machine-learning algorithm to predict hypotension based on high-fidelity arterial pressure waveform analysis. *Anesthesiology*. 2018;129:663–674.
11 Davies SJ, Vistisen ST, Jian Z, Hatib F, Scheeren TWL. Ability of an arterial waveform analysis-derived hypotension prediction index to predict future hypotensive events in surgical patients. *Anesth Analg*. 2020;130:352–359.
12 Lee S, Lee H-C, Chu Y, et al. Deep learning models for the prediction of intraoperative hypotension. *Br J Anaesth*. 2021;126:808–817.
13 Kang MW, Kim S, Kim YC, et al. Machine learning model to predict hypotension after starting continuous renal replacement therapy. *Sci Rep*. 2021;11:17169.
14 Maheshwari K, Shimada T, Fang J, et al. Hypotension Prediction Index software for management of hypotension during moderate- to high-risk noncardiac surgery: protocol for a randomized trial. *Trials*. 2019;20:255.
15 Vistisen ST, Enevoldsen J. CON: the hypotension prediction index is not a validated predictor of hypotension. *Eur J Anaesthesiol*. 2024;41:118–121.
16 Lim B, Arık SÖ, Loeff N, Pfister T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int J Forecast*. 2021. https://doi.org/10.1016/j.ijforecast.2021.03.012.
17 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. *Adv Neural Informat Process Syst*. 2017.
18 Gu A, Gulcehre C, Paine T, Hoffman M, Pascanu R. *Improving the gating mechanism of recurrent neural networks*. 2020.
19 Li J, Yan XS, Chaudhary D, et al. Imputation of missing values for electronic health record laboratory data. *NPJ Digital Med*. 2021;4:147.
20 GitHub - greatwhiz/tft_tf2: temporal fusion transformers for tensorflow 2.x. https://github.com/greatwhiz/tft_tf2.
21 Lee H-C, Park Y, Yoon SB, et al. VitalDB, a high-fidelity multi-parameter vital signs database in surgical patients. *Sci Data*. 2022;9:279.
22 Li Z. Extracting spatial effects from machine learning model using local interpretation method: an example of SHAP and XGBoost. *Comput Environ Urban Syst*. 2022;96:101845.
23 Jo Y-Y, Jang J-W, Kwon J-M, et al. Predicting intraoperative hypotension using deep learning with waveforms of arterial blood pressure, electroencephalogram, and electrocardiogram: retrospective study. *PLoS One*. 2022;17:e0272055.

24 Enevoldsen J, Vistisen ST. Performance of the hypotension prediction index may be overestimated due to selection bias. *Anesthesiology*. 2022;137:283–289.
25 Petch J, Di S, Nelson W. Opening the black box: the promise and limitations of explainable machine learning in cardiology. *Can J Cardiol*. 2022;38:204–213.
26 Zhang A, Xing L, Zou J, Wu JC. Shifting machine learning for healthcare from development to deployment and from models to data. *Nat Biomed Eng*. 2022;6:1330–1345.
27 Fernandes MPB, Armengol de la Hoz M, Rangasamy V, Subramaniam B. Machine learning models with preoperative risk factors and intraoperative hypotension parameters predict mortality after cardiac surgery. *J Cardiothorac Vasc Anesth*. 2021;35:857–865.
28 Van Calster B, McLernon DJ, van Smeden M, et al. Calibration: the Achilles heel of predictive analytics. *BMC Med*. 2019;17:230.
29 Alba AC, Agoritsas T, Walsh M, et al. Discrimination and calibration of clinical prediction models: users' guides to the medical literature. *JAMA*. 2017;318:1377–1384.
30 Collins GS, Reitsma JB, Altman DG, Moons KGM. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): the TRIPOD statement. *BMJ*. 2015;350:g7594.

Chapter 8

Discussion

8.1 Overview

In our research, we have taken a step-by-step approach to using machine learning to improve decision-making in the ICU. We started with a project focused on optimizing the administration of corticosteroids in patients with sepsis, moved on to improving decision support for starting or stopping RRT in patients with AKI, and finally turned our attention to predicting blood pressure changes during surgery using a transformer-based deep learning model.

In our first study (chapter 5), we developed an RL algorithm to guide corticosteroid dosing in critically ill sepsis patients. We used a rich dataset of routinely collected information from the ICU, and divided the data into 24-hour segments to match the daily cycle of ICU rounds. Traditionally, clinicians have relied on markers such as blood pressure and vasopressor use to make decisions about corticosteroid therapy. However, our RL model learned from the data that other factors were also important. One particularly interesting finding was that the algorithm responded to the presence of septic shock, even though septic shock was not a direct input. In other words, the model adjusted its treatment strategy to reflect the severity of the patient's condition, demonstrating that it could recognize the underlying signals associated with shock. In addition, when we compared the algorithm's recommendations with historical clinician decisions, we found a high level of agreement. This means that the model often made similar treatment decisions as experienced clinicians, but also provided new insights into patient outcomes. Off-policy evaluations showed that when the model's recommendations matched the historical clinicians' treatment strategies, patient outcomes improved compared to standard practice. These results suggest that our data-driven approach could provide a more personalized treatment strategy, even when working with retrospective, single-center data.

Building on these results, our second study focused on guiding RRT decisions for patients with AKI. In this project, we recognized that ICU patients are very diverse, so we needed a way to capture these differences more effectively. To do this, we incorporated weighted k-means clustering into our RL framework. By grouping patients based on 40 commonly measured clinical variables, clustering allowed us to create meaningful patient states that simplified the data while preserving important differences in each patient's condition. The refined RL model not only achieved very high concordance (96%) with clinicians' treatment policy, but also identified a subset of high-risk patients who might have benefited from earlier initiation of RRT. To ensure that our model was reliable, we used more advanced off-policy evaluation techniques. In both internal and external validation, the model achieved higher scores on metrics such as WIS and DICE. These results confirmed that our approach was robust and that the model's recommendations were consistent with, and sometimes superior to, conventional clinical decision making. We also experimented with different penalty values for initiating RRT to simulate different clinical practices (for example, differences between European and American treatment styles).

Our third study took a different direction, using a deep learning model to predict blood pressure changes during surgery. Being able to accurately predict blood pressure is critical, so instead of simply classifying whether a patient might become hypotensive, we aimed to predict the entire blood pressure trajectory. We used a TFT model, a neural network based on the transformer architecture designed for time-series data, and fed it low-resolution vital sign data recorded every 15 seconds, along with information about medications administered, such as hypnotics, analgesics, and vasoactive agents. The TFT model was able to generate a continuous prediction of MAP for the next few minutes, complete with uncertainty intervals that indicate how confident the prediction is. Not only did the model accurately predict hypotension, but it also appeared to be more robust than a model based on XGB. The TFT model achieved low prediction errors, around 4 mmHg on internal validation and 7 mmHg on external validation, and was consistent in identifying the risk of hypotension. It was also able to highlight the effects of specific interventions, such as the antihypertensive effects of drugs such as propofol or fentanyl.

8.2 RL for Optimizing Corticosteroid Therapy in Intensive Care

In this work, we developed an RL algorithm to optimize corticosteroid therapy in critically ill patients with sepsis. Our approach was designed to address the inherent challenges of applying RL in the medical domain by carefully constructing the RL environment, weighing the benefits of clustering versus continuous representations for the state space, and rigorously evaluating the resulting policy using OPE methods. In the following discussion, we elaborate on these aspects and suggest future directions for improving clinical RL systems.

8.2.1 Constructing RL Environments in Medical Applications

Creating an RL environment for medical applications starts with a clear definition of the state space. In clinical settings, the state should capture the complete physiological state of a patient at any point in time. This includes vital signs (heart rate, blood pressure, respiratory rate, and oxygen saturation), laboratory values (blood counts, electrolyte levels, and biochemical markers), and clinical observations (from imaging studies, physical examinations, and clinician notes). It is also important to include the patient's treatment history, such as previous interventions, medications, and how they have responded. Furthermore, deciding how much historical data to include – i.e. choosing an appropriate time window that covers enough past events – is crucial to ensure that the state representation reflects both current and relevant past conditions [33, 93].

As clinical datasets often have different data types and missing values, robust pre-processing, normalization and imputation techniques are required [94]. The challenge is to balance a detailed representation of the patient with the limits of computational resources and the need for a model that generalizes well.

In addition to the state space, the design of the action space is critical. The action space lists the clinical interventions available to healthcare providers. Some actions are discrete (e.g. deciding whether or not to start a particular treatment), while others are continuous (e.g. setting the exact dosage of a drug or adjusting the rate of an intravenous infusion). It is important that the action space closely resembles real clinical workflows so that the RL model remains interpretable and can be more easily integrated into clinical practice [95].

The reward function is at the heart of the RL environment. In medical applications, rewards are typically based on a mix of short-term and long-term outcomes. Short-term outcomes might include immediate changes in vital signs or lab results after an intervention, while long-term outcomes might relate to overall survival, reduction in morbidity, or improved quality of life. Designing a good reward function is challenging because interventions can have both immediate

benefits and delayed adverse effects. Therefore, the reward function must be carefully balanced – often with input from clinical experts – to encourage actions that are beneficial in both the short and long term [33, 96].

Finally, it is important to consider temporal dynamics and data granularity. Patient conditions evolve over time, and clinical decisions often depend on this dynamic process. It is important to choose a temporal resolution that captures meaningful changes in a patient's condition without adding too much noise. Data may be available in high-frequency formats (such as continuous waveform recordings) or as lower-resolution summaries (such as hourly or daily readings). In addition, as many clinical interventions have delayed effects, the RL environment needs to account for these delays with appropriate reward timing [93].

8.2.2 Clustering for RL Environments

When developing an RL environment for clinical applications, a key decision is whether to simplify the state space by clustering or to retain the raw, high-dimensional data. Each approach has distinct trade-offs in performance, interpretability and practicality.

Clustering reduces complexity by grouping similar patient states, making RL algorithms easier to train and faster to converge. It can reveal clinical patterns – such as disease stages – while smoothing out noise from measurement error or missing data. For example, clustering techniques have been successfully applied to derive clinically meaningful sepsis phenotypes [21]. Weighted clustering (as in paper 2) improves this further by prioritizing clinically relevant data points and better capturing transitions between states over time. This method also simplifies the creation of environments by estimating transition probabilities between clusters, which is critical for off-policy evaluation methods [93].

However, there are challenges to clustering medical data. Extreme values (e.g. dangerously high glucose levels) are often clinically significant, but may be grouped with less critical cases, masking urgent scenarios [34]. Clustering algorithms also require careful tuning of parameters (e.g. number of clusters) and validation against medical expertise to ensure meaningful groupings. These limitations highlight the risk of oversimplifying nuanced patient states.

Retaining raw, continuous data preserves fine-grained patient detail, which is critical when subtle parameter changes (e.g., small shifts in blood pressure) significantly affect outcomes [3]. Modern deep learning methods excel at processing high-dimensional data directly, enabling end-to-end learning without manual feature engineering [67]. In other domains, such techniques outperform manual processing by automatically identifying hidden patterns, suggesting similar potential in clinical RL [44]. A continuous state space could capture complex relationships in patient data that clustering might miss, potentially improving overall model performance.

Disadvantages include higher computational costs, risk of overfitting due to the large state space, and reduced interpretability. Clinicians often require transparent models, and the "black box" nature of deep learning can be a concern for trust. However, the benefits of detailed data may outweigh these drawbacks for applications where precision is essential, such as personalized treatment plans [97].

The decision depends on the clinical context and data characteristics. Clustering works well when there are clear patient subgroups, providing an interpretable, efficient framework for policy evaluation. However, a continuous state space may better reflect clinical reality in scenarios where granular data drives decisions – or where extremes and subtle patterns are critical. Advances in deep learning continue to address computational and interpretability challenges, making this approach increasingly viable for complex healthcare tasks [98].

8.2.3 Evaluation of RL Environments in Medical Applications

The evaluation of RL policies in medical applications is a unique challenge. It ensures that the application of new models meets the ethical obligation to avoid harm [99]. Unlike domains such as gaming or robotics, where suboptimal policies can lead to recoverable failures, errors in clinical settings can have irreversible consequences. This means that we need to make sure that we test the policies carefully using data from past clinical practices. OPE is not just a technical step, it is also a moral safeguard. It helps to ensure that new medical treatments are safe and effective without putting patients at risk. However, while OPE is fundamental, its application in medicine is associated with complexities that require careful consideration [100].

Direct experimentation with RL-derived policy in live clinical settings is ethically unacceptable [101]. OPE avoids this risk by using retrospective data, such as EHRs, to simulate how a new policy might work relative to established practice [93]. This approach is consistent with the principle of primum non nocere ("first, do no harm"), ensuring that innovations are tested against historical benchmarks before being implemented in the real world. Without OPE, there would be no viable way to translate RL research into clinical practice, as prospective trials of unvalidated interventions would pose unacceptable risks.

Despite its significance, OPE in health care is far from straightforward. First, confounding factors inherent in observational data – such as unmeasured variables (e.g., socioeconomic status, patient compliance) or selection bias – can introduce error into performance estimates [102]. For example, a policy to reduce steroid use in asthma patients may appear safe in historical data if healthier patients (who require fewer steroids) are overrepresented. Failure to adjust for such biases could lead to false conclusions and could advocate policies that harm vulnerable subgroups.

Second, rare but critical outcomes such as sepsis or cardiac arrest present statistical challenges. Methods such as WIS, which reweight trajectories based on the difference between target and historical policy, suffer from high variance when assessing rare events [74]. Even advanced techniques such as distribution correction methods (e.g. DICE) require large datasets to stabilize estimates, which are often not available for rare conditions or underrepresented populations [76].

Third, temporal dependencies in medical decision making complicate evaluation. Treatments such as chronic disease management have delayed effects that require OPE to consider long-term trajectories [103]. Simplifying these into short-term interactions risks misrepresenting outcomes, as critical consequences (e.g., drug toxicity) may occur months after initial interventions. Similarly, sparse reward signals, such as 1-year survival rates, make it difficult to attribute success or failure to specific decisions, further blurring performance estimates [104].

Finally, discrepancies between historical and target policies can destabilize the OPE. If clinicians have historically avoided a high-risk treatment, evaluating a policy that frequently recommends it may produce unreliable estimates due to extreme sampling weights. This problem, known as “weight collapse” or “variance explosion”, shows the risk of traditional OPE methods when policies deviate significantly from historical norms [74].

Counterfactual reasoning frameworks apply causal inference tools, such as inverse propensity scoring, to estimate “what-if” scenarios under a new policy while adjusting for observed confounders [105]. While promising, these methods need validation against clinical experience. For example, a policy that prioritizes early ICU transfer for deteriorating patients must be evaluated not only for statistical accuracy but also for its consistency with clinician intuition and hospital workflow [93].

The complexity of OPE in medicine reflects broader tensions between innovation and caution. In fields such as finance or advertising, suboptimal strategies can be iteratively refined through A/B testing [106]. In health care, however, such trial-and-error approaches are ethically prohibited,

putting immense pressure on OPE to deliver "first-time, right" validation. This challenge necessitates interdisciplinary collaboration – RL researchers must work closely with clinicians to ground OPE in the medical context, while statisticians ensure methodological consistency.

However, even the most advanced OPE cannot fully replicate real-world performance. Historical data may not capture edge cases, and unobserved confounders may persist. OPE should therefore be seen as a risk mitigation tool, not a guarantee of safety. A careful deployment pipeline could progress from OPE to simulated environments (e.g. digital twin patients [107]) and ultimately to tightly controlled prospective trials, with iterative post-deployment monitoring [108].

In our first study, we chose to train the RL model using continuous clinical data rather than applying an initial clustering step. This decision was driven by our desire to capture the full granularity of patient trajectories and to develop nuanced, patient-specific treatment policies. By leveraging an actor-critic framework, we allowed the critic network to implicitly learn an effective state representation directly from raw, continuous inputs. This approach enabled our model to capture subtle clinical nuances and generate individualized treatment recommendations, as evidenced by promising performance on real-world clinical datasets.

However, the continuous data approach brings its own challenges. The increased complexity of the state space requires robust off-policy evaluation methods to ensure reliable performance estimates. In our experience, traditional OPE methods such as High Confidence Off-Policy Evaluation sometimes struggle when applied to high-dimensional continuous data, possibly due to insufficient concordance between the behavior and target policies. To overcome this limitation, we have incorporated the DICE method into our subsequent work, which has demonstrated improved stability and accuracy in policy evaluation.

Notably, while our primary analysis in this paper used a single-centre dataset, we explicitly addressed this limitation by performing external validation (e.g., the MIMIC-IV data set) in the subsequent studies.

While our current actor-critic model using continuous data has yielded promising results, further improvements can be achieved by exploring new architectures. Transformer models, equipped with advanced embedding layers and self-attention mechanisms, have the potential to better capture the long-term temporal dependencies and complex interactions within clinical data. Particularly promising are decision transformers, which integrate sequential decision making with transformer architectures to directly map trajectories to actions. In paper 3 of this research series, we explicitly investigate transformer-based models and demonstrate their superior ability to model extended temporal contexts in treatment histories. Such architectures may prove particularly beneficial when scaling the model to external datasets, where variations in data quality, collection protocols and patient demographics are common. Scaling our approach to these diverse datasets presents additional challenges related to heterogeneity in clinical practice, making external validation essential to confirm the generalizability. Our ongoing work, which will be detailed in a third publication, addresses these issues by applying the model to external datasets and refining our evaluation techniques to ensure robust performance across different clinical settings.

8.3 RL for RRT Decision Support in AKI

In this paper (chapter 6), we present an RL algorithm developed using real-world medical data from the MIMIC-IV database [109], with external validation using data from the MUW. Our main finding is that RL shows considerable promise in guiding RRT decisions for ICU patients with AKI [110]. By integrating 40 routinely measured ICU variables through a weighted K-means clustering approach [96], our model achieved 98.5% agreement with human clinicians

and outperformed conventional off-policy assessment methods such as WIS [74] and the DICE algorithm [76] in both internal and external validation. The model is updated every 12 hours, reflecting the clinical decision interval, and dynamically provides data-driven recommendations on when to initiate and discontinue RRT. These results suggest that AI-driven strategies can significantly complement clinical decision-making, particularly by identifying high-risk patients who may benefit from earlier intervention.

There are several innovative aspects to our methodology. The use of real-world data from MIMIC-IV for model training and MUW for external validation sets our study apart from others that rely solely on clinical trial data, which often have strict inclusion criteria and limited sample sizes [111]. We coregistered a comprehensive set of 88 variables, prioritising the 40 most representative. The weighted K-means clustering method was instrumental in reducing state space complexity while preserving critical heterogeneity, in particular by placing strong emphasis on the RRT feature to effectively distinguish between patients who had already received RRT and those who had not.

To assess the quality of our clustering, we analysed the transitions between patient states using the Kullback-Leibler (KL) divergence as a metric to compare the state transition probability matrices between the training and test sets [79]. Notably, weighted K-means clustering reduced the KL divergence to approximately one-third of its original value, an improvement that exceeded that achieved by simply changing the number of clusters or features. After clustering, the significant increase in the clinician's WIS estimate indicated that patient trajectories were well represented [93]. This also had an interesting effect: The improved capture of the dynamics of patient states also improved the clinician's WIS estimation, making it difficult to find a model that outperformed the clinician's performance.

Traditional methods such as HCOPE face challenges when applied to high-dimensional continuous data [93]. Furthermore, evaluating the effectiveness of off-policy algorithms on a single dataset can be problematic. To address these limitations, we introduced an additional off-policy evaluation method to facilitate direct comparisons and provide a more reliable overall assessment. Specifically, we used the DICE algorithm [76], which provides more stable performance estimates.

Our analysis also revealed clinically significant findings. The model showed high agreement with human clinicians in both internal and external test sets; a notable finding given the rarity of RRT events, which typically reduces the likelihood of high agreement [3]. We performed a detailed subgroup analysis focusing on cases where AI recommendations and clinical decisions differed. Although these subgroups were relatively small, one subgroup of patients, those who did not receive RRT based on clinical judgement but would have been treated according to the model recommendations, had higher 90-day mortality. Conversely, another subgroup, consisting of patients treated by clinicians but not recommended for RRT by the AI, showed higher ICU mortality compared with patients with concordant AI and clinical decisions, suggesting a potential adverse effect of therapy in this cohort.

In addition, our model was able to simulate different treatment protocols by adjusting the penalty for RRT initiation. Lower penalties resulted in strategies more similar to European practice [17], whereas higher penalties resulted in strategies more similar to US protocols. Notably, despite being trained exclusively on US data, the model was able to emulate European-style treatment patterns.

This analysis also addressed an important issue regarding treatment strategies for RRT in Europe compared to the United States. In the US, treatment typically involves shorter durations, higher doses and an overall lower rate of RRT, whereas in Europe continuous RRT is the standard [17].

Our results showed a subtle trend suggesting that a lower rate of RRT may be associated with higher mortality. Although this trend is subtle and did not reach statistical significance

when using the limits of the WIS estimates [74], it remains observable and could potentially be significant in a study specifically designed to detect these differences.

Looking ahead, the impact of reinforcement learning on clinical medicine appears promising [112]. Our study demonstrates that AI-driven decision support can not only mirror but sometimes exceed human clinician performance, particularly in complex scenarios such as RRT initiation and discontinuation. Although we initially decided against predicting the dosage of RRT – given that clinicians already have robust methods for dosage determination – a logical next step could involve directly forecasting the dosage rather than relying on a binary on/off decision. This refinement may further enhance personalized patient care. Moreover, future work should include the development of an RL environment where different algorithms can be trained and rigorously compared using clinical data [97]. Such an environment would facilitate a systematic evaluation of various algorithmic strategies, promoting the advancement of AI-driven decision support systems in critical care.

In conclusion, our study demonstrates that an RL-based model can effectively support RRT decision-making in critically ill patients with AKI by achieving high concordance with human clinicians and identifying high-risk patients who might benefit from earlier intervention. While continuous data models capture detailed patient information, the weighted clustering approach enhances interpretability and reduces complexity [96]. Despite limitations such as reliance on retrospective data and variability in RRT practices across centers, our findings underscore the promise of RL in critical care. Future research should focus on integrating continuous and clustering-based methods, exploring advanced architectures like decision transformers [112], conducting prospective clinical trials to validate AI-driven systems [97], and developing comparative RL environments. These efforts will be essential for safely and effectively incorporating reinforcement learning into clinical practice to ultimately improve patient outcomes in the ICU.

8.4 TFT for Blood Pressure Forecasting

Our journey in developing a model to predict blood pressure during surgery (see chapter 7) started with Long Short-Term Memory (LSTM) networks, a standard method for time-series forecasting [92]. Early experiments with LSTMs, however, revealed a key problem: the model tended to pull predictions toward an average blood pressure value, missing important fluctuations – especially sudden changes after events like medication administration. This likely happens because LSTMs naturally smooth out noisy data over time, which makes them less sensitive to brief yet critical events [113]. Because of this limitation, we switched to transformer-based architectures. Unlike LSTMs, transformers use self-attention to highlight important moments – such as the precise time propofol is given – by assigning different weights to past data points [89]. This ability proved essential for capturing the cause-and-effect relationships between interventions (like vasopressor doses) and subsequent changes in blood pressure.

One surprising observation during transformer training was that the model could overfit our data, even though we had more than 40 million data points. This challenges the idea that larger datasets automatically prevent overfitting [114] and suggests that medical time-series data contain complex, hidden patterns that powerful models can learn. Overfitting here underscores the richness of perioperative data, where subtle interactions – such as the delayed drop in blood pressure from analgesics or the body's response to fluid administration – create intricate patterns [1]. Interestingly, increasing the number of attention heads did not improve performance, which indicates that our task may not require highly detailed separation of features or that even a few attention heads are enough to capture the most important temporal dependencies [115].

XGBoost (XGB) performed very well on our internal validation set for predicting hypotension as a yes-or-no outcome [116]. However, when we tested it on external data, its performance dropped significantly. We believe this discrepancy comes from two factors. First, reducing continuous blood pressure changes to a simple binary outcome lets XGB focus on static features, like baseline hypertension, rather than the timing of events [117]. Second, our internal dataset might have contained indirect clues about clinicians' responses to early signs of hypotension – such as the timing of vasopressor use – which XGB could exploit. These clues, however, are less likely to appear in external datasets where recording practices differ [111].

To improve the model's reliability and its ability to generalize across different datasets, we propose using data augmentation. Just as computer vision models benefit from rotating, scaling, and adding noise to images [118], medical time-series models might improve by incorporating synthetic scenarios that mimic expected drug effects. For example, creating artificial segments where propofol administration is paired with a controlled drop in blood pressure could reinforce the model's understanding of cause and effect [119]. Similarly, tweaking medication timing or dosage in existing records might simulate rare but clinically important events, such as an overdose. Although there is a risk of introducing biases if not done carefully, using synthetic data that follows known pharmacological principles could help fill gaps in real-world data [119].

We are also exploring the idea of adding constraints to our transformer models. Since we know that certain medications have a predictable impact on blood pressure, we could force these known effects into the model's output. This approach may make the predictions more reliable by grounding them in established physiological principles [120]. Looking ahead, combining several of these specialized models could lead to the development of a foundational model that serves multiple medical applications [121]. As more data and models are integrated over time, such foundational systems could offer a holistic view of a patient's current condition and provide doctors with more efficient decision support.

Beyond blood pressure prediction, our work highlights the transformative potential of transformer architectures in perioperative medicine. Their strength in integrating various data sources – vital signs, medication logs, ventilator settings – positions them as key elements in building comprehensive clinical AI systems. Future models might combine intraoperative data with preoperative risk factors and postoperative outcomes, effectively creating digital twins that simulate a patient's trajectory under different treatment strategies [107]. These systems could guide personalized hemodynamic management, predict complications like sepsis or delirium, or even assist with clinical documentation by highlighting key events, such as a norepinephrine dose before a predicted blood pressure rise. Importantly, the ability to visualize attention weights helps reduce concerns about AI as a "black box," allowing clinicians to understand which events influenced the predictions [122].

8.5 Conclusion

Our research has demonstrated the potential of RL and deep learning to improve decision-making in critical care. By applying machine learning techniques to optimize corticosteroid therapy in sepsis, guide RRT decisions in AKI and predict blood pressure fluctuations during surgery, we have highlighted key methodological advances and practical challenges in using AI in clinical practice.

The development of our RL algorithm for corticosteroid therapy is an example of how data-driven approaches can refine treatment strategies beyond traditional clinical heuristics. Our model not only aligned with established medical knowledge, but also uncovered subtle relationships between patient states and treatment efficacy. The ability of the algorithm to recognize septic

shock, despite not being explicitly coded, underlines the promise of RL in recognizing complex physiological patterns. Similarly, our work on RRT initiation showed that a clustering-based state-space representation can improve policy learning and interpretability while maintaining high agreement with clinical decisions. Furthermore, our analysis of different policy penalties provided insights into how RL frameworks can be adapted to reflect different clinical preferences.

Beyond RL, our exploration of deep learning for blood pressure prediction during surgery illustrated the power of neural networks in modeling complex physiological dynamics. The TFT model achieved high predictive accuracy while maintaining interpretability through uncertainty estimation. This capability is critical for translating AI-driven recommendations into real-time clinical decision support systems. The results suggest that deep learning can improve perioperative patient management by enabling proactive interventions to mitigate haemodynamic instability.

Despite this progress, significant challenges remain. Constructing reliable RL environments in medicine requires careful definition of state and action spaces, robust reward design, and comprehensive evaluation methods. Clustering methods facilitate interpretability but may obscure critical patient-specific nuances, while high-dimensional continuous representations improve precision but increase computational complexity. The evaluation of RL interventions using OPE techniques remains an ongoing challenge due to biases in historical data, the rarity of critical clinical events, and temporal dependencies in treatment outcomes. While techniques such as WIS and DICE provide robust validation frameworks, no single approach can completely remove the uncertainties associated with retrospective analyses.

Looking ahead, the successful integration of RL and deep learning into clinical workflows will depend on interdisciplinary collaboration. RL researchers, clinicians and statisticians need to work together to refine state space representations, optimize reward structures and improve methods for evaluating interventions. The adoption of AI in healthcare should follow a phased deployment strategy, including digital twin simulations and prospective trials before real-world implementation. In addition, addressing ethical and regulatory considerations will be critical to ensure that AI-driven recommendations augment rather than replace clinical expertise.

Ultimately, our research underscores the transformative potential of AI in critical care medicine. By using data-driven methods, we can move towards more personalized, adaptive and effective treatment strategies, improving patient outcomes while supporting clinical decision making. Future work should focus on refining these models, expanding their applicability to different patient populations, and integrating them seamlessly into clinical practice to realize the full potential of AI in healthcare.

Bibliography

[1] Alistair E W Johnson et al. "Machine learning and decision support in critical care." In: *Proceedings of the IEEE. Institute of Electrical and Electronics Engineers* (Feb. 2016), pp. 444–466. ISSN: 0018-9219. DOI: 10.1109/JPROC.2015.2501978.

[2] Duncan Shillan et al. "Use of machine learning to analyse routinely collected intensive care unit data: a systematic review." In: *Critical Care* (Aug. 2019), pp. 284–284. DOI: 10.1186/s13054-019-2564-9.

[3] Alvin Rajkomar, Jeffrey Dean, and Isaac Kohane. "Machine learning in medicine." In: *The New England Journal of Medicine* (Apr. 2019), pp. 1347–1358. ISSN: 0028-4793. DOI: 10.1056/NEJMra1814259.

[4] Michael Moor et al. "Foundation models for generalist medical artificial intelligence." In: *Nature* (Apr. 2023), pp. 259–265. ISSN: 0028-0836. DOI: 10.1038/s41586-023-05881-4.

[5] Cynthia Rudin. "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead." In: *Nature Machine Intelligence* (May 2019), pp. 206–215. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0048-x.

[6] Sana Tonekaboni et al. "What Clinicians Want: Contextualizing Explainable Machine Learning for Clinical End Use". In: *CoRR* (2019). arXiv: 1905.05134.

[7] Christopher M Sauer et al. "Leveraging electronic health records for data science: common pitfalls and how to avoid them." In: *The Lancet Digital Health* (Dec. 2022), pp. 893–898. ISSN: 25897500. DOI: 10.1016/S2589-7500(22)00154-6.

[8] Gary S Collins and Karel G M Moons. "Reporting of artificial intelligence prediction models." In: *The Lancet* (Apr. 2019), pp. 1577–1579. ISSN: 01406736. DOI: 10.1016/S0140-6736(19)30037-6.

[9] Jenna Wiens et al. "Do no harm: a roadmap for responsible machine learning for health care." In: *Nature Medicine* (Sept. 2019), pp. 1337–1340. DOI: 10.1038/s41591-019-0548-6.

[10] Sebastian Vollmer et al. "Machine learning and artificial intelligence research for patient benefit: 20 critical questions on transparency, replicability, ethics, and effectiveness". In: *BMJ* (2024). Ed. by British Medical Journal Publishing Group. DOI: 10.1136/bmj.q2390. eprint: https://www.bmj.com/content/387/bmj.q2390.full.pdf.

[11] Mervyn Singer et al. "The Third International Consensus Definitions for Sepsis and Septic Shock (Sepsis-3)." In: *The Journal of the American Medical Association* (Feb. 2016), pp. 801–810. DOI: 10.1001/jama.2016.0287.

[12] Charles Cook and Charles Smith. "Sepsis and Cortisone". In: *Nature* (Dec. 1952), pp. 980–980. ISSN: 0028-0836. DOI: 10.1038/170980b0.

[13] Djillali Annane et al. "Corticosteroids for treating sepsis." In: *Cochrane Database of Systematic Reviews* (Dec. 2015). DOI: 10.1002/14651858.CD002243.pub3.

[14] Sofie Louise Rygaard et al. "Low-dose corticosteroids for adult patients with septic shock: a systematic review with meta-analysis and trial sequential analysis." In: *Intensive Care Medicine* (July 2018), pp. 1003–1016. DOI: 10.1007/s00134-018-5197-6.

[15] Ondrej Jor et al. “Hypotension after induction of general anesthesia: occurrence, risk factors, and therapy. A prospective multicentre observational study.” In: *Journal of Anesthesia* (Oct. 2018), pp. 673–680. DOI: 10.1007/s00540-018-2532-6.

[16] Daniel I Sessler et al. “Perioperative Quality Initiative consensus statement on intraoperative blood pressure, risk and outcomes for elective surgery.” In: *British Journal of Anaesthesia* (May 2019), pp. 563–574. ISSN: 00070912. DOI: 10.1016/j.bja.2019.01.013.

[17] Eric A J Hoste et al. “Epidemiology of acute kidney injury in critically ill patients: the multinational AKI-EPI study.” In: *Intensive Care Medicine* (Aug. 2015), pp. 1411–1423. DOI: 10.1007/s00134-015-3934-7.

[18] Mark Andonovic et al. “Short- and long-term outcomes of intensive care patients with acute kidney disease.” In: *EClinicalMedicine* (Feb. 2022). DOI: 10.1016/j.eclinm.2022.101291.

[19] R.S. Sutton and A.G. Barto. “Reinforcement Learning: An Introduction (Solution Manual)”. In: *IEEE Transactions on Neural Networks* (1998), pp. 1054–1054. ISSN: 1045-9227. DOI: 10.1109/TNN.1998.712192.

[20] Bryan Lim et al. “Temporal Fusion Transformers for interpretable multi-horizon time series forecasting”. In: *International Journal of Forecasting* (June 2021). ISSN: 01692070. DOI: 10.1016/j.ijforecast.2021.03.012.

[21] Christopher W Seymour et al. “Derivation, validation, and potential treatment implications of novel clinical phenotypes for sepsis.” In: *The Journal of the American Medical Association* (May 2019), pp. 2003–2017. ISSN: 0098-7484. DOI: 10.1001/jama.2019.5791.

[22] KE Rudd et al. “Global, regional, and national sepsis incidence and mortality, 1990-2017: analysis for the Global Burden of Disease Study.” In: *The Lancet* (Jan. 2020), pp. 200–211. ISSN: 01406736. DOI: 10.1016/S0140-6736(19)32989-7.

[23] Derek Angus, Carlos Pires Pereira, and Eliezer Silva. “Epidemiology of severe sepsis around the world”. In: *Endocrine, Metabolic & Immune Disorders-Drug Targets* (June 2006), pp. 207–212. ISSN: 18715303. DOI: 10.2174/187153006777442332.

[24] Richard S Hotchkiss, Guillaume Monneret, and Didier Payen. “Sepsis-induced immunosuppression: from cellular dysfunctions to immunotherapy.” In: *Nature Reviews. Immunology* (Dec. 2013), pp. 862–874. DOI: 10.1038/nri3552.

[25] Tom van der Poll et al. “The immunopathology of sepsis and potential therapeutic targets.” In: *Nature Reviews. Immunology* (July 2017), pp. 407–420. DOI: 10.1038/nri.2017.36.

[26] Andrew Rhodes et al. “Surviving sepsis campaign: international guidelines for management of sepsis and septic shock: 2016.” In: *Intensive Care Medicine* (Mar. 2017), pp. 304–377. DOI: 10.1007/s00134-017-4683-6.

[27] Mitchell M Levy, Laura E Evans, and Andrew Rhodes. “The Surviving Sepsis Campaign Bundle: 2018 update.” In: *Intensive Care Medicine* (June 2018), pp. 925–928. DOI: 10.1007/s00134-018-5085-0.

[28] Djillali Annane et al. “Corticosteroids for severe sepsis and septic shock: a systematic review and meta-analysis.” In: *BMJ (Clinical Research Ed.)* (Aug. 2004), p. 480. DOI: 10.1136/bmj.38181.482222.55.

[29] Charles L Sprung et al. “Hydrocortisone therapy for patients with septic shock.” In: *The New England Journal of Medicine* (Jan. 2008), pp. 111–124. ISSN: 1533-4406. DOI: 10.1056/NEJMoa071366.

[30] Djillali Annane et al. “Corticosteroids in the treatment of severe sepsis and septic shock in adults: a systematic review.” In: *The Journal of the American Medical Association* (June 2009), pp. 2362–2375. ISSN: 1538-3598. DOI: 10.1001/jama.2009.815.

[31] B Venkatesh et al. “Adjunctive Glucocorticoid Therapy in Patients with Septic Shock.” In: *The New England Journal of Medicine* (Mar. 2018), pp. 797–808. ISSN: 0028-4793. DOI: 10.1056/NEJMoa1705835.

[32] Timothy E Sweeney et al. “Unsupervised analysis of transcriptomics in bacterial sepsis across multiple datasets reveals three robust clusters.” In: *Critical Care Medicine* (June 2018), pp. 915–925. DOI: 10.1097/CCM.0000000000003084.

[33] Matthieu Komorowski et al. “The Artificial Intelligence Clinician learns optimal treatment strategies for sepsis in intensive care.” In: *Nature Medicine* (Nov. 2018), pp. 1716–1720. ISSN: 1078-8956. DOI: 10.1038/s41591-018-0213-5.

[34] Aniruddh Raghu et al. “Deep Reinforcement Learning for Sepsis Treatment”. In: *CoRR* (2017). arXiv: 1711.09602.

[35] Rinaldo Bellomo et al. “Acute kidney injury in the ICU: from injury to recovery: reports from the 5th Paris International Conference.” In: *Annals of Intensive Care* (Dec. 2017), p. 49. DOI: 10.1186/s13613-017-0260-y.

[36] Kidney Disease: Improving Global Outcomes (KDIGO) CKD Work Group. “KDIGO 2024 clinical practice guideline for the evaluation and management of chronic kidney disease.” In: *Kidney International* (Apr. 2024), pp. 117–314. ISSN: 00852538. DOI: 10.1016/j.kint.2023.10.018.

[37] Shigehiko Uchino et al. “An assessment of the RIFLE criteria for acute renal failure in hospitalized patients.” In: *Critical Care Medicine* (July 2006), pp. 1913–1917. ISSN: 0090-3493. DOI: 10.1097/01.CCM.0000224227.70642.4F.

[38] Glenn M Chertow et al. “Acute kidney injury, mortality, length of stay, and costs in hospitalized patients.” In: *Journal of the American Society of Nephrology* (Nov. 2005), pp. 3365–3370. DOI: 10.1681/ASN.2004090740.

[39] Ravindra L Mehta et al. “Spectrum of acute renal failure in the intensive care unit: the PICARD experience.” In: *Kidney International* (Oct. 2004), pp. 1613–1621. DOI: 10.1111/j.1523-1755.2004.00927.x.

[40] Claudio Ronco et al. “Renal replacement therapy in acute kidney injury: controversy and consensus.” In: *Critical Care* (Apr. 2015), p. 146. DOI: 10.1186/s13054-015-0850-8.

[41] Alexander Zarbock et al. “Effect of early vs delayed initiation of renal replacement therapy on mortality in critically ill patients with acute kidney injury: the ELAIN randomized clinical trial.” In: *The Journal of the American Medical Association* (May 2016), pp. 2190–2199. DOI: 10.1001/jama.2016.5828.

[42] Stéphane Gaudry et al. “Initiation Strategies for Renal-Replacement Therapy in the Intensive Care Unit.” In: *The New England Journal of Medicine* (July 2016), pp. 122–133. DOI: 10.1056/NEJMoa1603017.

[43] Saber D Barbar et al. “Timing of Renal-Replacement Therapy in Patients with Acute Kidney Injury and Sepsis.” In: *The New England Journal of Medicine* (Oct. 2018), pp. 1431–1442. ISSN: 0028-4793. DOI: 10.1056/NEJMoa1803213.

[44] Volodymyr Mnih et al. “Human-level control through deep reinforcement learning.” In: *Nature* (Feb. 2015), pp. 529–533. DOI: 10.1038/nature14236.

[45] David Silver et al. “Mastering the game of Go with deep neural networks and tree search.” In: *Nature* (Jan. 2016), pp. 484–489. DOI: 10.1038/nature16961.

[46] François Grolleau et al. “Personalizing renal replacement therapy initiation in the intensive care unit: a reinforcement learning-based strategy with external validation on the AKIKI randomized controlled trials.” In: *Journal of the American Medical Informatics Association* (Apr. 2024), pp. 1074–1083. DOI: 10.1093/jamia/ocae004.

[47] John E Hall and Michael E Hall. *Textbook of Medical Physiology*. Elsevier Health Sciences, 2020.

[48] Michael Walsh et al. “Relationship between intraoperative mean arterial pressure and clinical outcomes after noncardiac surgery”. In: *Survey of Anesthesiology* (Aug. 2014), pp. 184–185. ISSN: 0039-6206. DOI: 10.1097/SA.0000000000000064.

[49] Giuseppe Mancia and Guido Grassi. “The autonomic nervous system and hypertension.” In: *Circulation Research* (May 2014), pp. 1804–1814. DOI: 10.1161/CIRCRESAHA.114.302524.

[50] Shin Ishii, Wako Yoshida, and Junichiro Yoshimoto. “Control of exploitation–exploration meta-parameter in reinforcement learning”. In: *Neural Networks* (June 2002), pp. 665–687. ISSN: 08936080. DOI: 10.1016/S0893-6080(02)00056-4.

[51] Michael N. Katehakis and Arthur F. Veinott. “The Multi-Armed Bandit Problem: Decomposition and Computation”. In: *Mathematics of Operations Research* (May 1987), pp. 262–268. ISSN: 0364-765X. DOI: 10.1287/moor.12.2.262.

[52] Richard Bellman. “A Markovian Decision Process”. In: *Indiana University Mathematics Journal* (1957), pp. 679–684. ISSN: 0022-2518. DOI: 10.1512/iumj.1957.6.56038.

[53] Richard Bellman. “The theory of dynamic programming”. In: *Bulletin of the American Mathematical Society* (Nov. 1954), pp. 503–516. ISSN: 0002-9904. DOI: 10.1090/S0002-9904-1954-09848-8.

[54] R Bellman. “Dynamic programming.” In: *Science* (July 1966), pp. 34–37. DOI: 10.1126/science.153.3731.34.

[55] N Metropolis and S Ulam. “The Monte Carlo Method”. In: *Journal of the American Statistical Association* (Sept. 1949), pp. 335–341. ISSN: 01621459. DOI: 10.2307/2280232.

[56] Gerald Tesauro. “Temporal difference learning and TD-Gammon”. In: *Communications of the ACM* (Mar. 1995), pp. 58–68. ISSN: 00010782. DOI: 10.1145/203330.203343.

[57] Christopher Z Mooney, Robert D Duval, and Robert Duvall. *Bootstrapping: A nonparametric approach to statistical inference*. Sage, 1993.

[58] Gavin A Rummery and Mahesan Niranjan. *On-line Q-learning using connectionist systems*. University of Cambridge, Department of Engineering Cambridge, UK, 1994.

[59] Christopher J. C. H. Watkins and Peter Dayan. “Q-learning”. In: *Machine learning* (May 1992), pp. 279–292. ISSN: 0885-6125. DOI: 10.1007/{BF00992698}.

[60] John N Tsitsiklis. “Asynchronous stochastic approximation and Q-learning”. In: *Machine learning* (1994), pp. 185–202.

[61] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *The annals of mathematical statistics* (1951), pp. 400–407.

[62] Vivek S Borkar and Vivek S Borkar. *Stochastic approximation: a dynamical systems viewpoint*. Springer, 2008.

[63] Tommi Jaakkola, Michael Jordan, and Satinder Singh. "Convergence of Stochastic Iterative Dynamic Programming Algorithms". In: *Advances in Neural Information Processing Systems*. 1993, pp. 703–710.

[64] Herbert Robbins and David Siegmund. "A convergence theorem for non negative almost supermartingales and some applications". In: *Optimizing Methods in Statistics*. Elsevier, 1971, pp. 233–257.

[65] Vivek S Borkar and Sean P Meyn. "The ODE method for convergence of stochastic approximation and reinforcement learning". In: *SIAM Journal on Control and Optimization* (2000), pp. 447–469.

[66] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors". In: *Nature* (1986), pp. 533–536. ISSN: 0028-0836. DOI: 10.1038/323533a0.

[67] Y. LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* (1998), pp. 2278–2324. ISSN: 00189219. DOI: 10.1109/5.726791.

[68] Vinod Nair and Geoffrey E Hinton. "Rectified linear units improve restricted boltzmann machines". In: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. 2010, pp. 807–814.

[69] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. "Deep sparse rectifier neural networks". In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings. 2011, pp. 315–323.

[70] Volodymyr Mnih et al. "Asynchronous Methods for Deep Reinforcement Learning". In: *arXiv* (2016). DOI: 10.48550/arxiv.1602.01783.

[71] Vijay Konda and John Tsitsiklis. "Actor-critic algorithms". In: *Advances in Neural Information Processing Systems* (1999).

[72] Doina Precup, Richard S Sutton, and Satinder Singh. "Eligibility traces for off-policy policy evaluation." In: *ICML*. Citeseer. 2000, pp. 759–766.

[73] Philip Thomas and Emma Brunskill. "Data-efficient off-policy policy evaluation for reinforcement learning". In: *International Conference on Machine Learning*. PMLR. 2016, pp. 2139–2148.

[74] Philip Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. "High-Confidence Off-Policy Evaluation". In: *Proceedings of the AAAI Conference on Artificial Intelligence* (Feb. 2015). ISSN: 2374-3468. DOI: 10.1609/aaai.v29i1.9541.

[75] Andreas Maurer and Massimiliano Pontil. "Empirical bernstein bounds and sample variance penalization". In: *arXiv preprint arXiv:0907.3740* (2009).

[76] Ofir Nachum et al. "[1906.04733] DualDICE: Behavior-Agnostic Estimation of Discounted Stationary Distribution Corrections". In: *arXiv* (June 2019).

[77] Mengjiao Yang et al. "Off-Policy Evaluation via the Regularized Lagrangian". In: *Advances in Neural Information Processing Systems* (2020), pp. 6551–6561.

[78] Lawrence Page et al. "The PageRank Citation Ranking: Bringing Order to the Web." In: *Stanford InfoLab* (Nov. 1999).

[79] S. Kullback and R. A. Leibler. "On information and sufficiency". In: *The Annals of Mathematical Statistics* (Mar. 1951), pp. 79–86. ISSN: 0003-4851. DOI: 10.1214/aoms/1177729694.

[80] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Hoboken, NJ, USA: John Wiley & Sons, Inc., Sept. 2005. ISBN: 9780471241959. DOI: 10.1002/047174882X.

[81] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Springer, 2006.

[82] James MacQueen. *Some methods for classification and analysis of multivariate observations*. 1967, pp. 281–298.

[83] S. Lloyd. "Least squares quantization in PCM". In: *IEEE Transactions on Information Theory* (Mar. 1982), pp. 129–137. ISSN: 0018-9448. DOI: 10.1109/TIT.1982.1056489.

[84] David Arthur and Sergei Vassilvitskii. "k-means++: the advantages of careful seeding". In: SODA '07. USA: Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035. ISBN: 9780898716245.

[85] Kyunghyun Cho et al. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2014, pp. 1724–1734. DOI: 10.3115/v1/D14-1179.

[86] D Bahdanau. "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473* (2014).

[87] I Sutskever. "Sequence to Sequence Learning with Neural Networks". In: *arXiv preprint arXiv:1409.3215* (2014).

[88] A Graves. "Generating sequences with recurrent neural networks". In: *arXiv preprint arXiv:1308.0850* (2013).

[89] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems* (2017).

[90] Kaiming He et al. "Deep residual learning for image recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2016, pp. 770–778. ISBN: 978-1-4673-8851-1. DOI: 10.1109/CVPR.2016.90.

[91] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. "Layer Normalization". In: *arXiv* (2016). DOI: 10.48550/arxiv.1607.06450.

[92] S Hochreiter and J Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[93] Omer Gottesman et al. "Guidelines for reinforcement learning in healthcare." In: *Nature Medicine* (Jan. 2019), pp. 16–18. ISSN: 1078-8956. DOI: 10.1038/s41591-018-0310-5.

[94] Roderick J. A. Little and Donald B. Rubin. *Statistical Analysis with Missing Data*. John Wiley & Sons, Inc., Aug. 2002. ISBN: 9780471183860. DOI: 10.1002/9781119013563.

[95] Edward H Shortliffe and Martin J Sepúlveda. "Clinical decision support in the era of artificial intelligence." In: *The Journal of the American Medical Association* (Dec. 2018), pp. 2199–2200. DOI: 10.1001/jama.2018.17163.

[96] Suchi Saria and Anna Goldenberg. "Subtyping: what it is and its role in precision medicine". In: *IEEE Intelligent Systems* (July 2015), pp. 70–75. ISSN: 1541-1672. DOI: 10.1109/MIS.2015.60.

[97] Eric J Topol. "High-performance medicine: the convergence of human and artificial intelligence." In: *Nature Medicine* (Jan. 2019), pp. 44–56. ISSN: 1078-8956. DOI: 10.1038/s41591-018-0300-7.

[98] Andre Esteva et al. "A guide to deep learning in healthcare." In: *Nature Medicine* (Jan. 2019), pp. 24–29. ISSN: 1078-8956. DOI: 10.1038/s41591-018-0316-z.

[99] Ziad Obermeyer and Ezekiel J Emanuel. "Predicting the future – Big data, machine learning, and clinical medicine". In: *The New England Journal of Medicine* (Sept. 2016), pp. 1216–1219. DOI: 10.1056/NEJMp1606181.

[100] Mattia Prosperi et al. "Causal inference and counterfactual prediction in machine learning for actionable healthcare". In: *Nature Machine Intelligence* (July 2020). ISSN: 2522-5839. DOI: 10.1038/s42256-020-0197-y.

[101] T L Beauchamp. "Methods and principles in biomedical ethics." In: *Journal of Medical Ethics* (Oct. 2003), pp. 269–274. DOI: 10.1136/jme.29.5.269.

[102] Judea Pearl. "An introduction to causal inference." In: *The International Journal of Biostatistics* (Feb. 2010), Article 7. DOI: 10.2202/1557-4679.1203.

[103] Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. "Time-series Generative Adversarial Networks". In: *Advances in Neural Information Processing Systems* (2019).

[104] Siqi Liu et al. "Reinforcement learning for clinical decision support in critical care: comprehensive review." In: *Journal of Medical Internet Research* (July 2020). DOI: 10.2196/18477.

[105] PAUL R. Rosenbaum and DONALD B. Rubin. "The central role of the propensity score in observational studies for causal effects". In: *Biometrika* (1983), pp. 41–55. ISSN: 0006-3444. DOI: 10.1093/biomet/70.1.41.

[106] Ron Kohavi, Diane Tang, and Ya Xu. *Trustworthy online controlled experiments: a practical guide to A/B testing*. Cambridge University Press, Mar. 2020. ISBN: 9781108653985. DOI: 10.1017/9781108653985.

[107] Jorge Corral-Acero et al. "The 'Digital Twin' to enable the vision of precision cardiology." In: *European Heart Journal* (Dec. 2020), pp. 4556–4564. DOI: 10.1093/eurheartj/ehaa159.

[108] Safa Elkefi and Onur Asan. "Digital twins for managing health care systems: rapid literature review." In: *Journal of Medical Internet Research* (Aug. 2022), e37641. DOI: 10.2196/37641.

[109] Alistair E W Johnson et al. "MIMIC-IV, a freely accessible electronic health record dataset." In: *Scientific Data* (Jan. 2023), p. 1. DOI: 10.1038/s41597-022-01899-x.

[110] Sergey Levine et al. "Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems". In: *arXiv* (2020). DOI: 10.48550/arxiv.2005.01643.

[111] Thomas Davenport and Ravi Kalakota. "The potential for artificial intelligence in healthcare." In: *Future Healthcare Journal* (June 2019), pp. 94–98. DOI: 10.7861/futurehosp.6-2-94.

[112] Lili Chen et al. "Decision Transformer: Reinforcement Learning via Sequence Modeling". In: *arXiv* (2021). DOI: 10.48550/arxiv.2106.01345.

[113] Slawek Smyl. "A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting". In: *International Journal of Forecasting* (July 2019). ISSN: 01692070. DOI: 10.1016/j.ijforecast.2019.03.017.

[114] Chiyuan Zhang et al. "Understanding deep learning (still) requires rethinking generalization". In: *Communications of the ACM* (Mar. 2021), pp. 107–115. ISSN: 0001-0782. DOI: 10.1145/3446776.

[115] Krzysztof Choromanski et al. “Rethinking Attention with Performers”. In: *arXiv* (2020). DOI: 10.48550/arxiv.2009.14794.

[116] Pattharawin Pattharanitima et al. “Comparison of Approaches for Prediction of Renal Replacement Therapy-Free Survival in Patients with Acute Kidney Injury.” In: *Blood Purification* (Feb. 2021), pp. 621–627. DOI: 10.1159/000513700.

[117] Ewout W. Steyerberg. *Clinical prediction models: A practical approach to development, validation, and updating*. Statistics for biology and health. Cham: Springer International Publishing, 2019. ISBN: 978-3-030-16398-3. DOI: 10.1007/978-3-030-16399-0.

[118] Connor Shorten and Taghi M. Khoshgoftaar. “A survey on Image Data Augmentation for Deep Learning”. In: *Journal of Big Data* (Dec. 2019), p. 60. ISSN: 2196-1115. DOI: 10.1186/s40537-019-0197-0.

[119] Mauro Giuffrè and Dennis L Shung. “Harnessing the power of synthetic data in healthcare: innovation, application, and privacy.” In: *npj Digital Medicine* (Oct. 2023), p. 186. DOI: 10.1038/s41746-023-00927-3.

[120] Laura von Rueden et al. “Informed Machine Learning - A Taxonomy and Survey of Integrating Prior Knowledge into Learning Systems”. In: *IEEE Transactions on Knowledge and Data Engineering* (2021), pp. 1–1. ISSN: 1041-4347. DOI: 10.1109/TKDE.2021.3079836.

[121] Tao Tu et al. “Towards generalist biomedical AI”. In: *NEJM AI* (Feb. 2024). ISSN: 2836-9386. DOI: 10.1056/AIoa2300138.

[122] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “"Why Should I Trust You?": Explaining the Predictions of Any Classifier”. In: *arXiv* (2016). DOI: 10.48550/arxiv.1602.04938.

Appendices

Appendix A

Appendix: Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis

Development of a Reinforcement Learning Algorithm to Optimize Corticosteroid Therapy in Critically Ill Patients with Sepsis

Supplemental Material. Table of Contents

Supplemental Table S1. Diagnosis of sepsis

Non-prophylactic anti-infective drugs used	Cultures drawn suggestive of sepsis	Admission diagnosis suggestive of infection
Amikacin	Urine culture	Pneumonia
Amoxicillin	MRSA swab	Meningitis
Benzylpenicillin	Blood culture	Endocarditis
Ceftazidime	Catheter tip culture	Cholangitis
Cefotaxime	Drain fluid culture	Pancreatitis
Ciprofloxacin	Stool culture	Abscess
Rifampicin	CSF culture	Fasciitis
Clindamycin	Nasal swab	Peritonitis
Tobramycin	Perineal swab	GI perforation/rupture
Vancomycin	Rectal swab	GI ischemia
Imipenem	Wound swab	Diverticulitis
Doxycycline	Ascites culture	Sepsis
Metronidazole	Legionella urinary antigen	Infection
Erythromycin		Inflammatory
Flucloxacillin		
Fluconazole		
Ganciclovir		
Flucytosine		
Gentamicin		
Foscarnet		
Amphotericin B		
Meropenem		
Myambutol		
Co-Trimoxazole		
Voriconazole		
Amoxicillin/Clavulanic acid		
Aztreonam		
Chloramphenicol		
Fusidic acid		
Piperacillin		
Ceftriaxone		
Cefuroxime		
Cefazoline		
Caspofungin		
Itraconazole		
Levofloxacin		
Anidulafungin		
Linezolid		
Tigecycline		
Daptomycin		
Colistin		

MRSA: Methicillin-resistant Staphylococcus aureus; CSF: cerebrospinal fluid; GI: gastro-intestinal.

Patients with sepsis were identified based on the Sepsis-3 criteria. Accordingly, patients with new organ dysfunction as indicated by either a SOFA score ≥2 at admission or an increase of 2 points or more in the SOFA score during the ICU stay in the context of suspected infection were included in the sepsis cohort used to develop the RL algorithm. The definition of suspected infection, which has been previously described (Thoral et al., AmsterdamUMCdb GitHub repository), was operationalized by identifying antibiotic therapy (other than prophylactic use), cultures drawn, sepsis flagged by admitting physicians or admission diagnosis suggestive for severe infection. The onset of the septic episode was considered the day the change in the SOFA score occurred and patients remained in the sepsis cohort until discharge or death.

Supplemental Table S2. Input features included in development of the algorithm

Category	Variable	Type	Preprocessing and derived features
Patient characteristics	Age (years)	Discrete	Bins*
	Male gender	Boolean	-
	Weight (kg)	Continuous	-
	Admission count	Discrete	-
Vital parameters	Respiratory rate (min^{-1})	Continuous	Mean, minimum, maximum, standard deviation
	Heart rate (min^{-1})	Continuous	
	Invasive systolic, diastolic, and mean blood pressure (mmHg)	Continuous	
	Non-invasive systolic, diastolic, and mean blood pressure (mmHg)	Continuous	
	SpO_2	Continuous	
	Temperature (°c)	Continuous	
Laboratory values	AG (mEq/l), BE (mEq/l), Bicarbonate (mEq/l), pH, Lactate (mmol/l), $PaCO_2$ (mmHg), PaO_2 (mmHg), SaO_2	Continuous	Mean, minimum, maximum, standard deviation
	ACTH (pmol/l), Cortisol (nmol/l), TSH (mIU/l), fT_3 (pmol/l)	Continuous	
	Albumin (g/l), Ammonia (µmol/l), Bilirubin (µmol), GOT (U/l), GPT (U/l)	Continuous	
	Blood glucose (mmol/l)	Continuous	
	CRP (mg/l), PCT (ug/l)	Continuous	
	Ca (mg/dl), Cl (mmol/l), Fe (µmol/l), K (mmol/l), Mg (mg/dl), Na (mmol/l), Phosphate (mg/dl), *i*Ca	Continuous	
	Total cholesterol, HDL-cholesterol, LDL-cholesterol (mmol/l) Triglycerides (mg/dl)	Continuous	
	Serum creatinine (µmol/l), Serum urea (mmol/l), eGFR (ml/min)	Continuous	
	Fibrinogen (mg/dl), PTT (s), PT (s)	Continuous	
	GSF Glucose (mmol/l), CSF Leucocytes (ml^{-1}), CSF Protein (mg/dl)	Continuous	

	Hematocrit, Hemoglobin (g/l), RBC count, Leucocyte count, Lymphocytes count, Neutrophils count, MCH (pg), MCV (fl), Thrombocyte count	Continuous	
	Urinary Na (mEq/l), Urinary K (mEq/l), Urinary Creatinine (mmol/day), Urinary Urea (mmol/day)	Continuous	
Ventilation parameters	FiO_2	Continuous	Mean, minimum, maximum, standard deviation
	PEEP (cmH_2O)	Continuous	
	Set respiratory rate (min^{-1})	Continuous	
Fluid balance	Urine output (ml)	Continuous	-
	Net fluid balance (ml)	Continuous	-
	Ultrafiltration rate (ml/h)	Continuous	-
Commonly used medication	Dose of fast-acting insulins: Actrapid, Novorapid, Velosulin (IU)	Continuous	Mean, minimum, maximum, standard deviation, and sum for continuously administered drugs
	Dose of benzodiazepines: Alprazolam, Lorazepam, Midazolam, Oxazepam, Temazepam	Continuous	
	Dose of other sedatives and analgesics: Clonidine, Fentanyl, Haloperidol, Morphine, Propofol	Continuous	
	Dose of antiplatelet and anticoagulant drugs: Clopidogrel, Heparin	Continuous	
	Dose of antiarrhythmic drugs: Metoprolol, Amiodarone	Continuous	
	Dose of vasopressors and inotropic agents: Digoxin, Dopamine, Noradrenaline	Continuous	
	Diuretics: Furosemide, Spironolactone	Continuous	
	Highest antibiotic rank**	Discrete	
	Antiviral drugs	Boolean	
	Antifungal drugs	Boolean	

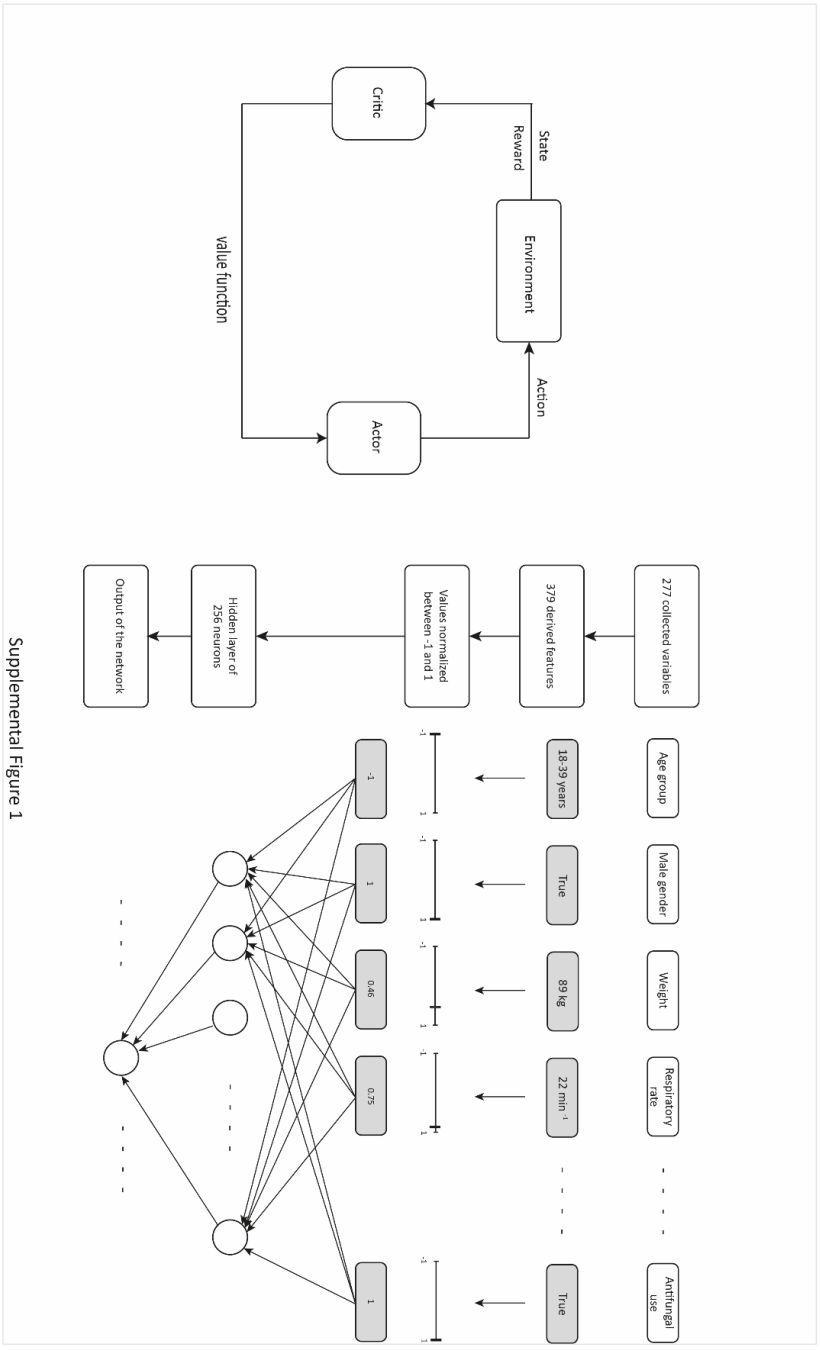
Supplemental Table S2 presents all the input variables collected and the derived features used for developing the reinforcement learning algorithm, after excluding the variables not represented < 2% of the datapoints. SpO2: peripheral oxygen saturation; SOFA: sequential organ failure assessment; AG: anion gap; BE: base excess; pH: potential of hydrogen; $PaCO_2$, PaO_2: partial pressure of carbon dioxide and oxygen, respectively, in arterial blood; SaO_2: arterial oxygen saturation; ACTH: adrenocorticotropic hormone; TSH: thyroid-stimulating hormone; fT_3: free triiodothyronine; GOT: serum glutamic-oxaloacetic transaminase; GPT: serum glutamic-pyruvic transaminase; CRP: C reactive protein; PCT: procalcitonin; Ca: total calcium; Cl: chloride; Fe: serum iron; K: serum potassium; Mg: serum magnesium; Na: serum sodium; *i*Ca: ionized Calcium; HDL: high-density lipoprotein; LDL: low-density lipoprotein; eGFR: estimated glomerular filtration rate; PTT: partial thromboplastin time; PT: prothrombin time; CSF:

cerebrospinal fluid; RBC: red blood cell; MCH: mean corpuscular hemoglobin; MCV: mean corpuscular volume; FiO_2: fraction of inspired oxygen.

*Age was sorted into bins: 18-39 years, 40-49 years, 50-59 years, 60-69 years, 70-79 years, 80+ years

**Antibiotics were classified by rank after Braykov et al. and the highest rank of antibiotics administered was used as input feature.

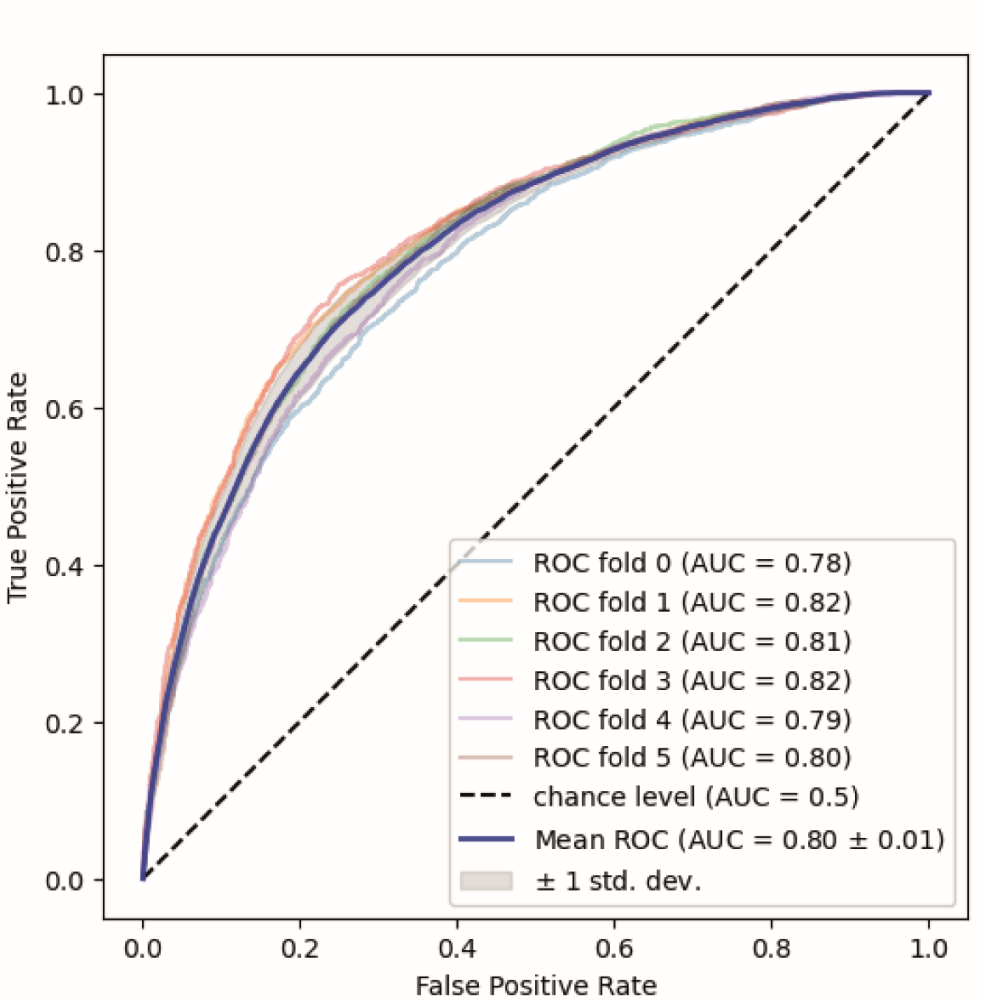
Supplemental Figure S1. Development of the RL Algorithm



For each day of the ICU stays included in the sepsis cohort, a set of 281 variables were collected, of which 277 were used as inputs. Using imputation and normalization, we derived a balanced dataset of

379 input variables. The RL algorithm consisted of 2 neural networks, with a similar structure that includes a hidden layer of 256 hidden neurons, but different outputs. The actor network has 5 potential outputs, corresponding to the 5 possible actions. The critic network ends in one node terminates with a single node. The 2 networks interact to determine the optimal policy. the actor network proposes an action based on the current state of the environment and the environment changes its state. The critic network evaluates the actor network based on the reward that the chosen action returns.

Supplemental Figure S2. Micro-average ROC curve of the random forest model



Supplemental Figure 2

The micro-average multiclass AUROC for the random forest model was 0.8. Other performance metrics for the random forest model were:

True positive rate (TPR): 0.7936205665317781

True negative rate (TNR): 0.9255359157578037

Positive predictive value (PPV: 0.8938337801608579

Negative predictive value (NPV): 0.8502332008982553

False positive rate (FPR): 0.07446408424219632

False negative rate (FNR): 0.20637943346822185

False discovery rate (FDR): 0.1061662198391421

Accuracy: 0.8673179955877718

F1-score: 0.8407514815281806

F2-score: 0.8718163275979292

Supplemental Table S3. The most relevant predictors of the clinicians' policy according to the random forest model ordered from the lowest to highest rank

Feature	**Component of the unit vector (normalized vectors)**
PTT min	0.124653211
Fentanyl max	0.128485455
Blood glucose max	0.134108057
Length of stay	0.137129421
Thrombocytes min	0.139307112
Urea max	0.142143406
Leucocytes mean	0.142262137
PEEP mean	0.14305812
Blood glucose std	0.147403945
Midazolam (Dormicum) max	0.149040607
PEEP max	0.167538026
PEEP min	0.16780183
PTT max	0.175419476
PTT mean	0.17657839
Highest antibiotic rank	0.177066982
Thrombocytes mean	0.180251789
Noradrenaline (Norepinephrine) sum	0.187155279
Leucocytes max	0.202227827
Thrombocytes max	0.274195479
Noradrenaline (Norepinephrine) max	0.379869603

The normalized vector is the unit vector with a length of 1, which is defined by

$u_e = \frac{u}{||u||}$, where u_e is the normalized vector, u the original vector, and $|u|$ the norm of vector u. The norm of a vector is defined by $||u|| = \sqrt{u_1^2 + u_2^2 + \cdots + u_n^2}$, where u_i is an entry of the vector.

The normalized vector indicates how strongly a single input feature affects the decision in comparison to the other features.

Supplemental Table S4. The most relevant feature for the RL policy listed from the lowest to highest rank

Feature	**Component of the unit vector (normalized vectors)**
Velosulin (Insulin) max	0.078966455
Urinary sodium std	0.079285593
Serum sodium max	0.079325068
CSF protein std	0.079650287
CSF protein mean	0.079713876
Magnesium mean	0.081478474
invasive mean BP min	0.081885501
Midazolam max	0.084812754
Leucocytes max	0.086287831
Serum sodium mean	0.088578701
invasive diastolic BP min	0.10005845
Respiratory Rate min	0.105951594
Blood glucose max	0.113368019
Blood glucose std	0.114456484
Heartrate std	0.114888807
Blood glucose mean	0.119304028
Leucocytes mean	0.120227263
Leucocytes min	0.127948494
invasive mean BP mean	0.131318385
invasive diastolic BP mean	0.136660705

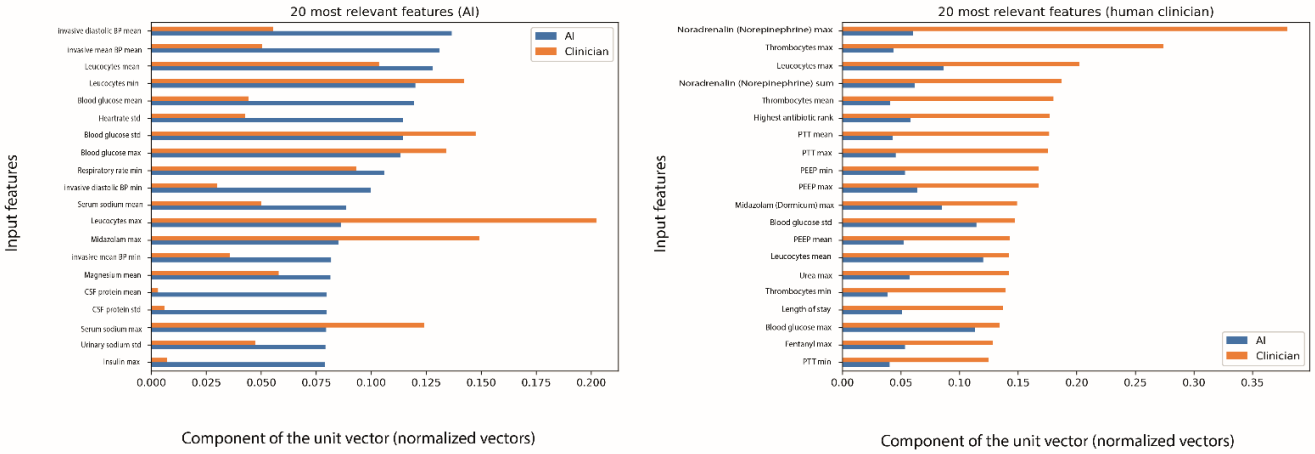
PTT: partial thromboplastin time; *i*Ca: ionized Calcium; PaCO2: partial pressure of carbon dioxide in arterial blood; CSF: cerebrospinal fluid; BP: blood pressure; GOT: serum glutamic-oxaloacetic transaminase; GPT: serum glutamic-pyruvic transaminase; PCT: procalcitonin; K: serum potassium; iO2: fraction of inspired oxygen.

The normalized vector is the unit vector with a length of 1, which is defined by

$u_e = \frac{u}{||u||}$, where u_e is the normalized vector, u the original vector, and $|u|$ the norm of vector u. The norm of a vector is defined by $||u|| = \sqrt{u_1^2 + u_2^2 + \cdots + u_n^2}$, where u_i is an entry of the vector.

The normalized vector indicates how strongly a single input feature affects the decision in comparison to the other features.

Supplemental Figure S3. The 20 most relevant input features for the RL and random forest models



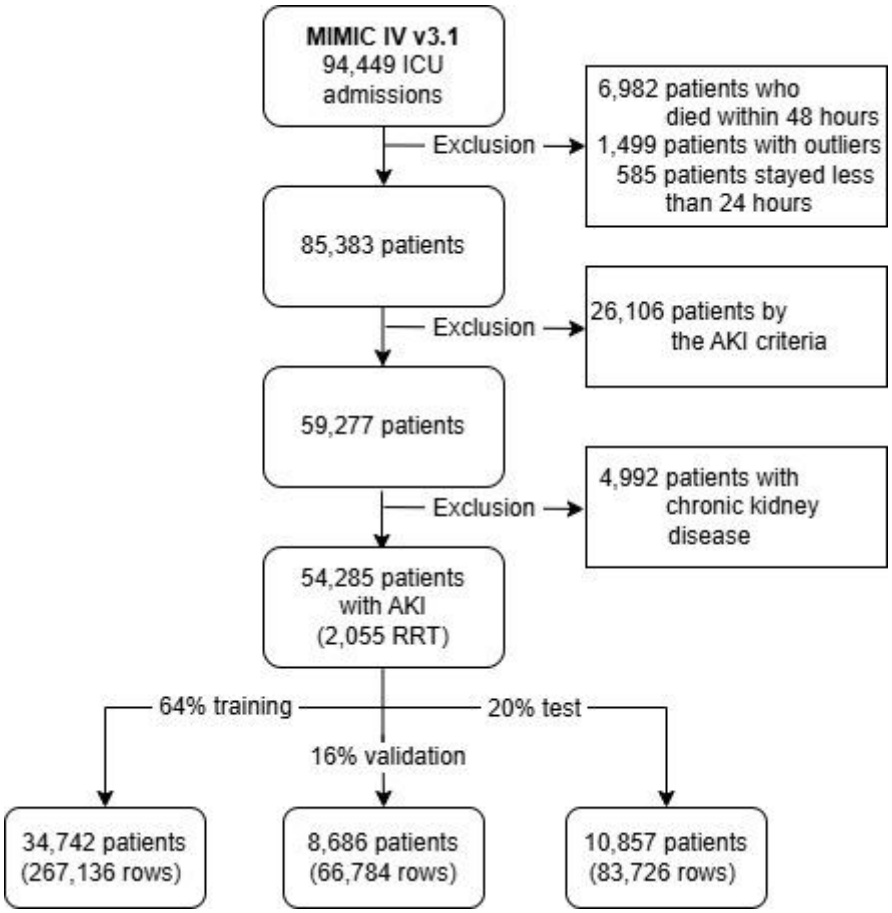
Supplemental Figure 3

The normalized vectors for 20 most relevant features for each model, sorted by rank, are displayed together with the normalized vectors of the same features in the other model.

Appendix B

Appendix: Optimized Renal Replacement Therapy Decisions in Intensive Care: A Reinforcement Learning Approach

SUPPLEMENTARY INFORMATION



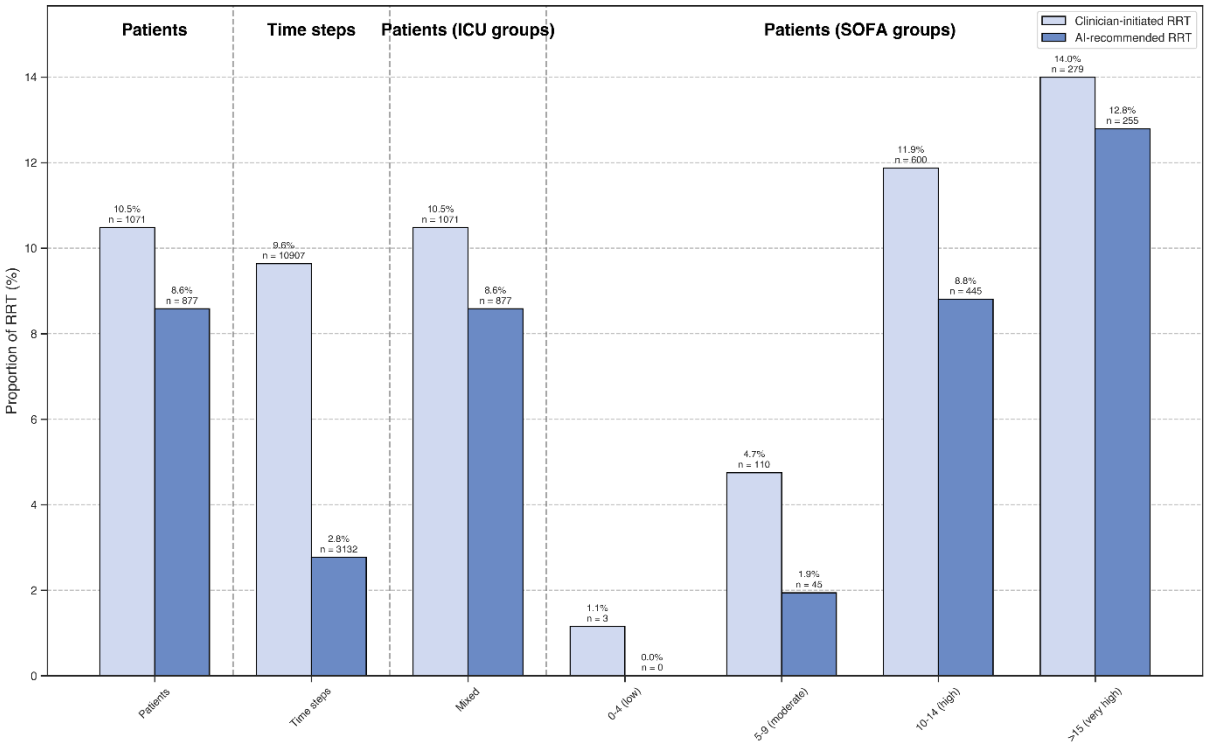
Supplemental Material 1. Patients with acute kidney injury from the MIMIC IV v3.1 database were identified using the AKI cohort.

Feature	Weight	MIMIC		MUW	
		Mean (SD)	Missingness (%)	Mean (SD)	Missingness (%)
12-hour total output, mL	0,3944	765.77 ± 807.75	0	1274.57 ± 866.63	0
SOFA score	0,2776	4.3 ± 2.7	0	8.87 ± 4.0	0
Cumulative balance, mL	0,2434	15884.29 ± 30463.42	0	23672.98 ± 20214.82	0
Creatinine, mg/dL	0,2399	1.34 ± 1.24	0	1.13 ± 0.92	48,5
Platelet count, ×10^3/µL	0,2141	210.07 ± 113.65	0	202.12 ± 127.7	48,4
Chloride, mEq/L	0,1887	103.52 ± 6.08	0	106.57 ± 5.27	4,9
BUN, mg/dL	0,1836	28.06 ± 21.91	0	25.93 ± 17.44	48,4
Anion gap, mEq/L	0,1802	13.16 ± 3.5	0	11.72 ± 3.81	99,8
Calcium, mg/dL	0,171	8.46 ± 0.69	0	8.35 ± 0.62	48,7
Total input, mL	0,1679	10466.24 ± 27521.2	0	13308.13 ± 11358.36	0
WBC count, ×10^3/µL	0,1614	11.74 ± 7.83	0	11.41 ± 6.47	48,3
Total bilirubin, mg/dL	0,1594	1.92 ± 4.18	0	1.4 ± 2.75	49
Phosphorus, mg/dL	0,1537	3.49 ± 1.15	0	3.25 ± 1.18	48,7
O2 flow, L/min	0,1529	11.6 ± 15.88	0	7.72 ± 12.2	52,6
Total output, mL	0,1507	5418.05 ± 7215.22	0	10364.86 ± 9537.48	0
Weight, kg	0,1478	83.21 ± 24.68	8,3	78.85 ± 20.03	0
RASS score	0,1439	-0.66 ± 1.33	0	-1.51 ± 1.9	16,6
Sodium, mEq/L	0,1434	138.97 ± 5.05	0	140.4 ± 5.04	4,9
Temperature, °C	0,1417	36.9 ± 0.51	0	36.64 ± 0.69	40,1
Age, years	0,1401	65.58 ± 16.25	0	59.23 ± 16.3	0
Maximum vasopressor dose, µg/kg/min	0,1178	0.11 ± 1.21	0	0.16 ± 1.66	0
Mean airway pressure, cmH2O	0,1151	9.07 ± 3.1	0	11.08 ± 3.48	48,8
GCS score	0,1143	13.65 ± 2.66	0	8.94 ± 5.34	92,7
AST (SGOT), U/L	0,1132	109.23 ± 386.12	0	139.07 ± 648.46	48,9
PT, s	0,1112	15.49 ± 6.0	0	17.71 ± 8.87	99,8
PTT, s	0,1109	38.66 ± 18.75	0	41.88 ± 11.06	48
RBC count, ×10^6/µL	0,1085	3.35 ± 0.63	0	3.32 ± 0.57	48,2
LDH, U/L	0,1066	357.1 ± 404.55	0	340.91 ± 601.05	49,2
Hematocrit, %	0,106	30.98 ± 5.76	0	30.77 ± 4.91	5,8
Respiratory rate, breaths/min	0,1057	19.71 ± 4.37	0	19.93 ± 7.29	16
Bicarbonate, mEq/L	0,1028	24.58 ± 4.56	0	27.2 ± 3.52	16,6
SpO2, %	0,1024	96.57 ± 2.26	0	97.52 ± 2.73	0,7
Ionized calcium, mmol/L	0,1018	1.13 ± 0.08	0	1.17 ± 0.07	5,2
Hemoglobin, g/dL	0,1011	10.1 ± 1.94	0	10.0 ± 1.64	4,9
FiO2, %	0,0999	0.36 ± 0.15	0	0.44 ± 0.14	43,3
ALT (SGPT), U/L	0,0978	115.29 ± 378.13	0	105.11 ± 338.9	49,2

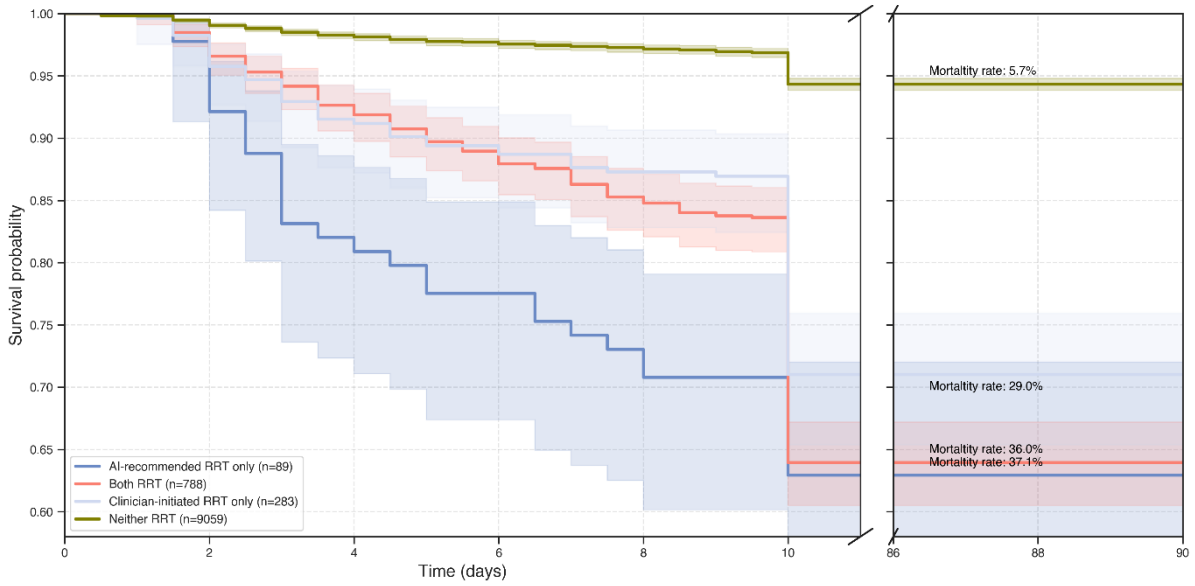
Shock index	0,0972	0.72 ± 0.23	0	0.66 ± 0.19	4,2
Glucose, mg/dL	0,0971	141.27 ± 47.66	0	137.39 ± 33.57	5,1
Heart rate, beats/min	0,0961	84.4 ± 16.19	0	80.61 ± 15.82	1,1
Minute ventilation, L/min	0,0951	8.4 ± 2.31	0	7.72 ± 3.48	48,4
Mean blood pressure, mmHg	0	80.77 ± 12.34	0	83.75 ± 12.89	3,6
INR	0	1.42 ± 0.61	0	1.27 ± 0.36	49,1
Potassium, mEq/L	0	4.07 ± 0.49	0	4.16 ± 0.43	5,1
Fibrinogen, mg/dL	0	362.07 ± 175.11	3,1	488.9 ± 192.91	48,6
Arterial pH	0	7.41 ± 0.05	0	7.43 ± 0.06	16,6
PaO2/FiO2 ratio	0	331.81 ± 224.32	0	248.07 ± 98.65	26,1
Tidal volume, mL	0	452.67 ± 112.05	0	496.04 ± 180.16	47,7
PaO2, mmHg	0	106.66 ± 61.77	0	96.24 ± 23.81	16,6
Albumin, g/dL	0	3.05 ± 0.57	0	2.85 ± 0.46	48,6
Diastolic blood pressure, mmHg	0	60.61 ± 13.07	0	62.05 ± 11.1	3,6
12-hour total input, mL	0	1352.12 ± 3401.3	0	1519.81 ± 1025.09	0
Magnesium, mg/dL	0	2.11 ± 0.33	0	2.14 ± 0.36	48,7
Systolic blood pressure, mmHg	0	120.54 ± 17.42	0	125.82 ± 18.96	3,6
Peak airway pressure, cmH2O	0	17.8 ± 5.96	0	18.38 ± 6.08	48,4
Extubated (yes/no)	0	0.13 ± 0.33	64	0.52 ± 0.5	0
Arterial base excess, mEq/L	0	1.05 ± 4.33	0	0.4 ± 4.04	16,6
Plateau airway pressure, cmH2O	0	18.0 ± 3.96	0	21.22 ± 4.83	90,4
Height, cm	0	168.88 ± 12.79	41,7	171.46 ± 10.88	0
cCntral venous pressure, mmHg	0	12.55 ± 18.56	0,9	12.73 ± 10.57	74,8
PaCO2, mmHg	0	43.47 ± 10.63	0	42.02 ± 7.66	16,6
Arterial lactate, mmol/L	0	1.8 ± 1.08	0	1.25 ± 1.21	5,2
PEEP, cmH2O	0	5.94 ± 2.31	0	8.05 ± 2.51	47,2
CK-MB, ng/mL	0	12.79 ± 33.32	4,4	6.86 ± 22.41	81
End-tidal CO2, mmHg	0	36.96 ± 7.06	54	36.98 ± 0.0	0
Troponin, ng/mL	0	0.35 ± 0.83	0	0.17 ± 0.28	97
Mechanical ventilation (yes/no)	0	0.34 ± 0.47	0	0.64 ± 0.48	0
Absolute neutrophil count, ×10^3/μL	0	11.14 ± 8.15	94,5	7.22 ± 2.66	81,2
SIRS criteria	0	1.31 ± 0.96	0	0.73 ± 0.73	0
SaO2, %	0	95.65 ± 3.91	91,9	96.51 ± 4.12	16,6
Triglycerides, mg/dL	0	213.55 ± 224.51	93,2	215.5 ± 0.0	0
SvO2, %	0	65.54 ± 10.64	95,3	68.04 ± 11.38	98,5
Pulmonary artery systolic pressure, mmHg	0	39.99 ± 12.78	94,5	38.84 ± 22.05	96,7

Pulmonary artery diastolic pressure, mmHg	0	19.82 ± 6.64	94,5	21.44 ± 17.98	96,7
re-admission (yes/no)	0	0.31 ± 0.46	0	0.07 ± 0.26	0
Mean pulmonary artery pressure, mmHg	0	28.47 ± 19.51	94,5	28.22 ± 18.69	96,7
Urine creatinine, mg/dL	0	83.74 ± 59.71	96,7	61.21 ± 39.58	56,2
Gender (M/F)	0	0.44 ± 0.5	0	0.39 ± 0.49	0
BNP, pg/mL	0	7797.93 ± 11153.81	98,2	4444.9 ± 6366.23	97,1
CRP, mg/L	0	101.46 ± 86.39	97,8	115.12 ± 94.43	48,5
Urine urea nitrogen, mg/dL	0	491.05 ± 311.08	97,9	540.96 ± 336.14	56,2
Urine sodium, mEq/L	0	65.29 ± 45.85	97,1	95.7 ± 46.47	57
Urine potassium, mEq/L	0	39.77 ± 20.49	98	42.37 ± 21.65	96,3
Iron, µg/dL	0	46.18 ± 40.87	99	39.48 ± 35.01	99,1
Ammonia, µg/dL	0	48.78 ± 41.24	99,2	55.97 ± 41.29	99,1
TSH, mIU/L	0	3.42 ± 5.44	98,7	3.37 ± 5.08	98
Total protein, g/dL	0	5.47 ± 0.96	99,5	5.18 ± 0.85	94,4
Cardiac index, L/min/m²	0	3.15 ± 0.95	99,4	3.32 ± 1.16	96,5
ACT, s	0	157.45 ± 33.78	99	232.13 ± 151.92	99,8
T3, ng/dL	0	72.17 ± 38.45	99,8	58.16 ± 26.64	100
GGT, U/L	0	361.12 ± 438.95	99,9	283.31 ± 416.3	48,7
Low molecular weight heparin (yes/no)	0	0.47 ± 0.36	99,9	0.47 ± 0.0	0
APACHE II renal failure score	0	0.2 ± 0.39	100	0.0 ± 0.0	0
Urine osmolality, mOsm/kg	0	nan ± nan	100	490.09 ± 154.73	55,5

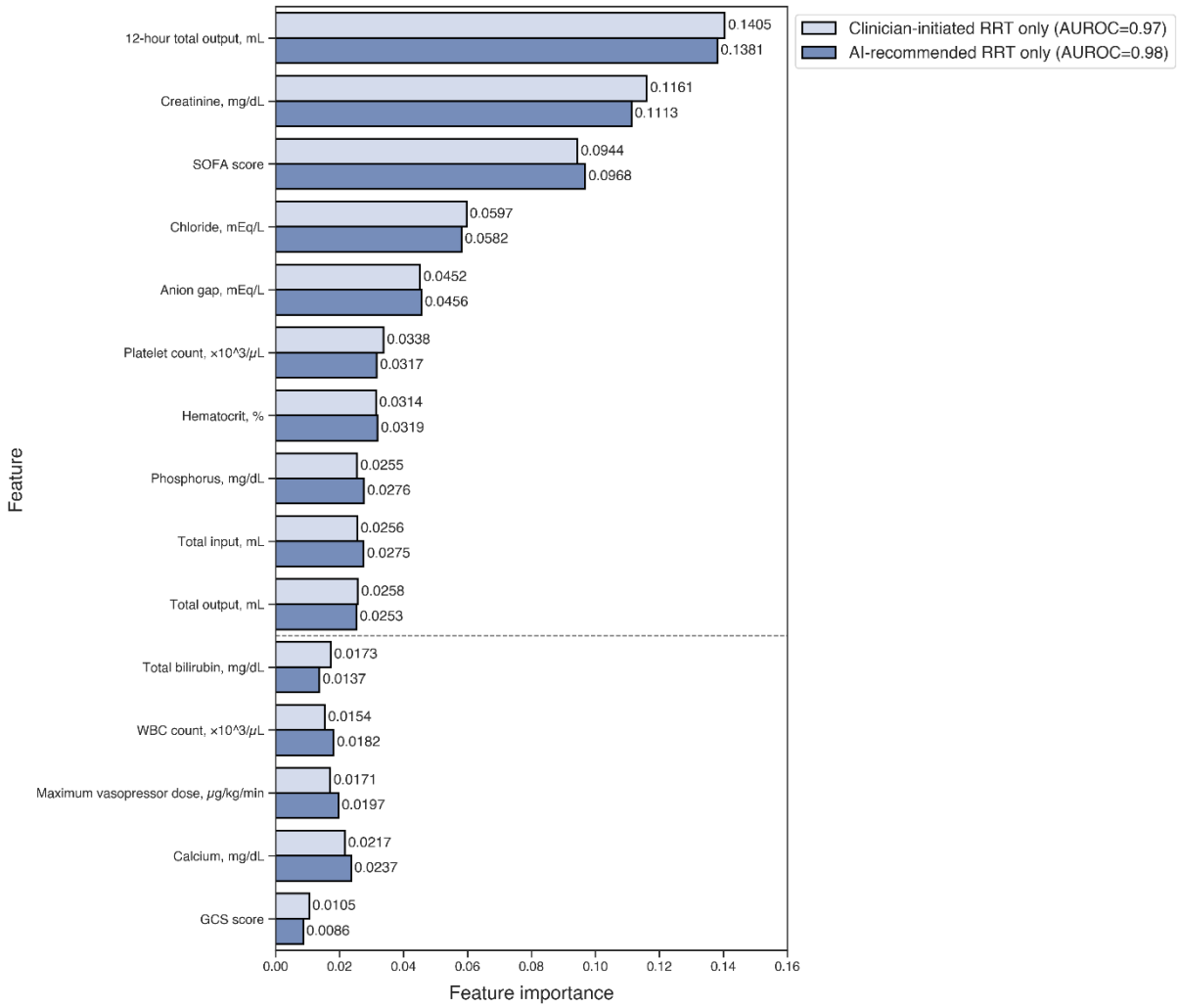
Supplemental Material 2: Feature weight, feature distribution and number of missingness in both data sets. SOFA: Sequential Organ Failure Assessment; RASS: Richmond Agitation-Sedation Scale; GCS: Glasgow Coma Scale; BUN: Blood Urea Nitrogen; WBC: White Blood Cells; AST (SGOT): Aspartate Aminotransferase (Serum Glutamic-Oxaloacetic Transaminase); PT: Prothrombin Time; PTT: Partial Thromboplastin Time; RBC: Red Blood Cells; LDH: Lactate Dehydrogenase; SpO_2: Peripheral Capillary Oxygen Saturation; FiO_2: Fraction of Inspired Oxygen; ALT (SGPT): Alanine Aminotransferase (Serum Glutamic-Pyruvic Transaminase); INR: International Normalized Ratio; PEEP: Positive End-Expiratory Pressure; Central Venous Pressure: Central Venous Pressure; $PaCO_2$: Arterial Partial Pressure of Carbon Dioxide; PaO_2: Arterial Partial Pressure of Oxygen; CK-MB: Creatine Kinase-MB Isoenzyme; BNP: B-type Natriuretic Peptide; CRP: C-Reactive Protein; TSH: Thyroid-Stimulating Hormone; APACHE II: Acute Physiology and Chronic Health Evaluation II; ACT: Activated Clotting Time; T3: Triiodothyronine; GGT: Gamma-Glutamyl Transferase; SIRS: Systemic Inflammatory Response Syndrome; SaO_2: Arterial Oxygen Saturation; SvO_2: Mixed Venous Oxygen Saturation



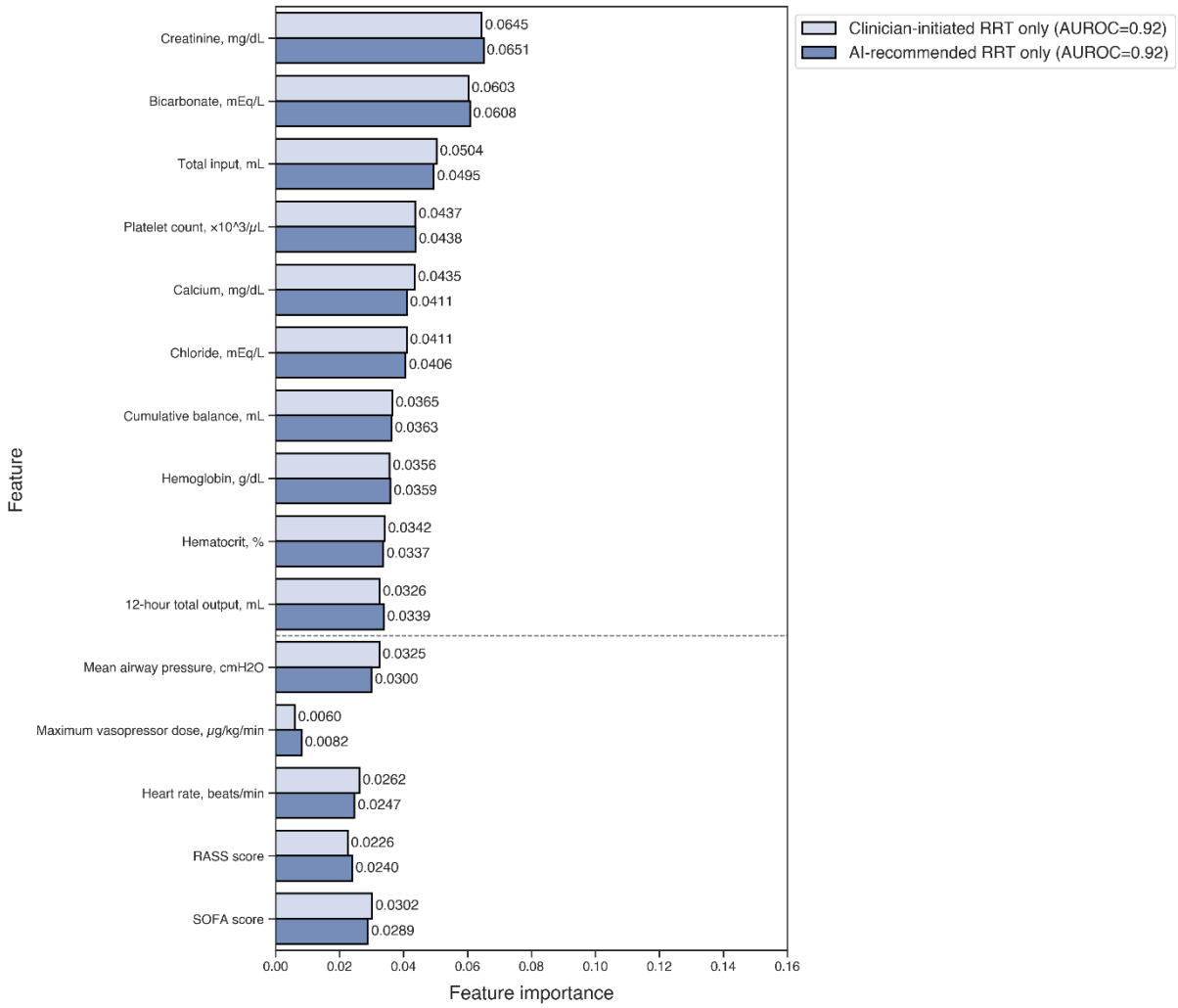
Supplemental Material 3: Proportion of AI-recommended renal replacement therapy (RRT) in the Medical University of Vienna (MUW) data set. SOFA: Sequential Organ Failure Assessment



Supplemental Material 4: Survival probability in the Medical University of Vienna (MUW) data set. RRT: Renal Replacement Therapy



Supplemental Material 5: Feature importance in the MIMIC data set. SOFA: Sequential Organ Failure Assessment; BUN: Blood Urea Nitrogen; RBC: Red Blood Cells; RASS: Richmond Agitation-Sedation Scale; GCS: Glasgow Coma Scale



Supplemental Material 6: Feature importance in the MUW data set.

Appendix C

Appendix: Development and External Validation of Temporal Fusion Transformer Models for Continuous Intraoperative Blood Pressure Forecasting

Supplemental Tables

Supplemental Table 1 Input features

Variable	Data type	Input type
Age of Patient	Real	Static
Gender of Patient	Categorical	Static
ASA Score: American Society of Anesthesiologists Physical Status Classification	Categorical	Static
Urgency of Procedure	Categorical	Static
Type of Surgery	Categorical	Static
Mean Arterial Pressure (MAP)	Real	Target
Pulse Rate (bpm)	Real	Observed
Oxygen Saturation (SpO2%)	Real	Observed
End-tidal Carbon Dioxide (EtCO2 mmHg)	Real	Observed
Systolic Blood Pressure (mmHg)	Real	Observed
Diastolic Blood Pressure (mmHg)	Real	Observed
Heart Rate (bpm)	Real	Observed
Invasive Blood Pressure (mmHg)	Categorical	Observed
Inhaled Sevoflurane (Insevo)	Real	Observed
Exhaled Sevoflurane (Exsevo)	Real	Observed
Inhaled Desflurane (Indes)	Real	Observed
Exhaled Desflurane (Exdes)	Real	Observed
Berodual (Combination of Ipratropium and Fenoterol)	Real	Observed
Cisatracurium (Neuromuscular Blocking Agent)	Real	Observed
Esketamine (S-enantiomer of Ketamine)	Real	Observed
Etomidate (Hypnotic Agent)	Real	Observed
Fentanyl (Opioid Analgesic)	Real	Observed
Midazolam (Benzodiazepine)	Real	Observed
Noradrenaline (Vasopressor)	Real	Observed
Phenylephrine (Vasopressor)	Real	Observed
Piritramide (Opioid Analgesic)	Real	Observed
Propofol (Anaesthetic)	Real	Observed
Remifentanil (Opioid Analgesic)	Real	Observed
Rocuronium (Neuromuscular Blocking Agent)	Real	Observed
Succinylcholine (Neuromuscular Blocking Agent)	Real	Observed
Sufentanil (Opioid Analgesic)	Real	Observed

Compliance of the Respiratory System (ml/cmH2O)	Real	Observed
Fraction of Inspired Oxygen (FiO2%)	Real	Observed
Positive End-Expiratory Pressure (PEEP cmH2O)	Real	Observed
Plateau Pressure (cmH2O)	Real	Observed
Maximum Airway Pressure (Pmax)	Real	Observed
Peak Inspiratory Pressure (Ppeak cmH2O)	Real	Observed
Mean Airway Pressure (Pmean cmH2O)	Real	Observed
Respiratory System Resistance (cmH2O/L/sec)	Real	Observed
Ventilation Frequency (Ventfreq bpm)	Real	Observed
Ventilation Mode (Ventmode)	Categorical	Observed
Tidal Volume (Vt ml)	Real	Observed
Dobutamine Infusion (Dobutamin Perfusor µg/kg/min)	Real	Observed
Epinephrine Infusion (Epinephrin Perfusor µg/kg/min)	Real	Observed
Levosimendan Infusion (Levosimendan Perfusor µg/kg/min)	Real	Observed
Noradrenaline Infusion (Noradrenalin Perfusor µg/kg/min)	Real	Observed
Phenylephrine Infusion (Phenylephrin Perfusor µg/kg/min)	Real	Observed
Propofol Infusion (Propofol Perfusor µg/kg/min)	Real	Observed
Remifentanil Infusion (Remifentanil Perfusor µg/kg/min)	Real	Observed
Sufentanil Infusion (Sufentanil Perfusor µg/kg/min)	Real	Observed
Vasopressin Infusion (Vasopressin Perfusor IU/min)	Real	Observed
Phase of Surgery	Categorical	Observed

Footnote Supplemental Table 1: The input features for the TFT model. The ‘Input Type’ column describes how the variables are processed: static variables (input once), observed variables (processed as time series) and the target variable (input as time series and predicted by the algorithm).

Supplemental Table 2 Missing and unplausible values

Variable	Missing Values (%)		Unplausible Values (%)		Ranges for plausibility check	
	Internal	External	Internal	External	Max	Min
Age of Patient	0·00	0·00	0·00	0·00	112	18
Gender of Patient	0·00	0·00	0·00	0·00	1	0
ASA Score	0·00	0·00	0·00	0·00	6·0	1
Urgency of Procedure	0·00	0·00	0·00	0·00	3	1
Type of Surgery	0·00	2·14	0·00	0·00	17	0
Mean Arterial Pressure (MAP)	0·23	0·00	0·00	0·00	300·0	0·0
Pulse Rate (bpm)	0·03	0·00	0·00	0·00	300·0	4·0
Oxygen Saturation (SpO2%)	0·00	5·72	0·00	0·00	100·0	24·0
EtCO2 mmHg	22·21	4·22	0·00	0·00	99·0	0·0
Systolic Blood Pressure (mmHg)	0·02	4·21	0·00	0·00	490·0	0
Diastolic Blood Pressure (mmHg)	0·02	5·70	0·00	0·00	350·0	0
Heart Rate (bpm)	0·00	7·15	0·00	0·01	350·0	0
Invasive Blood Pressure (mmHg)	0·00	7·14	0·00	0·07	1	0
Inhaled Sevoflurane (Insevo)	0·00	0·00	0·00	0·00	12·0	0·0
Exhaled Sevoflurane (Exsevo)	0·00	0·00	0·00	0·00	12·2	0·0
Inhaled Desflurane (Indes)	0·00	46·23	0·00	0·00	17·0	0·0
Exhaled Desflurane (Exdes)	0·00	46·23	0·00	0·00	26·3	0·0
Berodual	0·00	75·98	0·00	0·00	30	0
Cisatracurium	0·00	75·98	0·00	0·00	200·0	0·0
Esketamine	0·00	100·00	0·00	0·00	350·0	0·0
Etomidate (Hypnotic Agent)	0·00	100·00	0·00	0·00	150	0
Fentanyl (Opioid Analgesic)	0·00	100·00	0·00	0·00	1000·0	0·0
Midazolam (Benzodiazepine)	0·00	100·00	0·00	0·00	250·0	0·0
Noradrenaline (Vasopressor)	0·00	100·00	0·00	0·00	4000	0
Phenylephrine (Vasopressor)	0·00	100·00	0·00	0·00	4·0	0·0
Piritramide (Opioid Analgesic)	0·00	100·00	0·00	0·00	37·5	0·0
Propofol (Anaesthetic)	0·00	100·00	0·00	0·00	1000·0	0·0
Remifentanil (Opioid Analgesic)	0·00	100·00	0·00	0·00	400	0
Rocuronium	0·00	100·00	0·00	0·00	200·0	0·0
Succinylcholine	0·00	100·00	0·00	0·00	200	0
Sufentanil (Opioid Analgesic)	0·00	100·00	0·00	0·00	500·0	0·0
Compliance	89·56	100·00	0·00	0·00	200·0	0·0

FiO2%	88·01	100·00	0·00	0·00	100·0	0·0
Positive End-Expiratory Pressure	89·09	10·64	0·00	0·00	88·0	0
Plateau Pressure (cmH2O)	89·47	5·77	0·00	0·00	63·0	0·0
Maximum Airway Pressure (Pmax)	90·29	12·69	0·00	0·03	50·0	0·0
Peak Inspiratory Pressure	88·92	12·66	0·00	0·00	81·0	0
Mean Airway Pressure	88·54	100·00	0·00	0·00	40·0	0
Respiratory System Resistance	99·56	11·71	0·00	0·00	900·0	0·0
Ventilation Frequency	90·18	6·92	0·00	0·04	85·0	0·0
Ventilation Mode	93·94	100·00	0·00	0·00	2·0	0·0
Tidal Volume (Vt ml)	89·62	100·00	0·00	0·00	2000·0	0·0
Dobutamine Infusion	0·00	100·00	0·00	0·00	52·5	0·0
Epinephrine Infusion	0·00	100·00	0·00	0·00	8·0	0·0
Levosimendan Infusion	0·00	100·00	0·00	0·00	7·5	0·0
Noradrenalin Perfusor	0·00	99·95	0·00	0·00	53·4	0·0
Phenylephrin Perfusor	0·00	100·00	0·00	0·00	200	0·0
Propofol Perfusor	0·00	98·93	0·00	0·00	20000	0·0
Remifentanil Perfusor	0·00	98·51	0·00	0·00	104·4	0·0
Sufentanil Perfusor	0·00	47·30	0·00	0·01	0·6	0·0
Vasopressin Perfusor	0·00	22·81	0·00	0·00	120·0	0
Phase of Surgery	-	-	-	-	-	-

Footnote Supplemental Table 2: The input features for the TFT model. The high number of missingness of the ventilation parameters is due to the resampling from 2 minutes to 15 seconds.

Supplemental Table 3 Patient characteristics: external data set

	N = 5,065
Age (years)	58 (44, 72)
Male sex (-)	2,545 (50%)
ASA Score	
1	1,590 (31%)
2	2,933 (58%)
3	512 (10%)
4	30 (0·6%)
5	0 (0·0%)
Surgical urgency (-)	
Elective	4,466 (88%)
Emergency	733 (12%)
Urgent	0 (0%)
Duration of surgery (min)	187 (77, 296)
Surgical discipline	
General surgery	4,727 (93%)
Orthopaedics/Trauma surgery	0 (0·0%)
Plastic surgery	0 (0·0%)
ENT	0 (0·0%)
Maxillofacial surgery	0 (0·0%)
Neurosurgery	0 (0·0%)
Gynaecology	222 (4%)
Obstetrics	0 (0·0%)
Urology	0 (0·0%)
Ophthalmology	0 (0·0%)
Dermatology	116 (2·2%)

	N = 5,065
Undefined	0 (0·0%)
Vascular surgery	0 (0·0%)
1 Median (IQR); n (%)	

Supplemental Table 4 Performance metrics for the TFT model

Forecast time	Accuracy	Sensitivity	Specificity	PPV	NPV	AUROC
1 min (internal)	0·958 (0·957–0·958)	0·920 (0·918–0·923)	0·962 (0·962–0·963)	0·747 (0·743–0·750)	0·990 (0·990–0·990)	0·988 (0·988–0·988)
3 min (internal)	0·928 (0·927–0·929)	0·678 (0·673–0·682)	0·958 (0·957–0·958)	0·656 (0·652–0·661)	0·961 (0·961–0·962)	0·954 (0·954–0·955)
5 min (internal)	0·908 (0·907–0·908)	0·463 (0·458–0·467)	0·960 (0·959–0·960)	0·571 (0·566–0·576)	0·939 (0·938–0·939)	0·909 (0·908–0·910)
7 min (internal)	0·903 (0·902–0·903)	0·352 (0·347–0·356)	0·965 (0·965–0·966)	0·536 (0·530–0·541)	0·929 (0·928–0·930)	0·879 (0·877–0·880)
1 min (external)	0·942 (0·941–0·942)	0·856 (0·852–0·859)	0·946 (0·946–0·947)	0·458 (0·454–0·462)	0·992 (0·992–0·992)	0·960 (0·959–0·961)
3 min (external)	0·946 (0·946–0·947)	0·572 (0·567–0·577)	0·966 (0·965–0·966)	0·465 (0·460–0·470)	0·978 (0·977–0·978)	0·945 (0·944–0·946)
5 min (external)	0·944 (0·944–0·945)	0·377 (0·372–0·383)	0·973 (0·973–0·974)	0·418 (0·413–0·424)	0·969 (0·968–0·969)	0·903 (0·902–0·905)
7 min (external)	0·944 (0·944–0·945)	0·275 (0·270–0·280)	0·978 (0·977–0·978)	0·379 (0·372–0·385)	0·965 (0·964–0·965)	0·867 (0·865–0·869)

Footnote Supplemental Table 4: Summary of hypotension performance metrics of the TFT model for different time frames (one, three, five, and seven minutes into the future) using different test sets. ‘Internal’ refers to the internal validation, while ‘external’ refers to the external validation. The 95% confidence interval is indicated by the values within the brackets.

Supplemental Table 5 Performance metrics for the XGB model

Forecast time	Accuracy	Sensitivity	Specificity	PPV	NPV	AUROC
1 min (internal)	0·975 (0·975–0·976)	0·872 (0·869–0·875)	0·988 (0·987–0·988)	0·894 (0·892–0·897)	0·985 (0·984–0·985)	0·994 (0·994–0·994)
3 min (internal)	0·966 (0·965–0·966)	0·807 (0·804–0·811)	0·985 (0·984–0·985)	0·865 (0·862–0·868)	0·977 (0·976–0·977)	0·987 (0·987–0·988)
5 min (internal)	0·970 (0·969–0·970)	0·825 (0·821–0·828)	0·987 (0·987–0·987)	0·883 (0·880–0·886)	0·979 (0·979–0·980)	0·989 (0·989–0·990)
7 min (internal)	0·972 (0·971–0·972)	0·851 (0·848–0·855)	0·987 (0·986–0·987)	0·885 (0·882–0·888)	0·982 (0·982–0·982)	0·991 (0·991–0·991)
1 min (external)	0·956 (0·956–0·957)	0·304 (0·301–0·307)	0·990 (0·989–0·990)	0·597 (0·593–0·601)	0·965 (0·965–0·966)	0·961 (0·960–0·961)
3 min (external)	0·954 (0·953–0·954)	0·098 (0·096–0·100)	0·997 (0·997–0·997)	0·604 (0·596–0·613)	0·956 (0·956–0·957)	0·891 (0·890–0·892)
5 min (external)	0·951 (0·951–0·952)	0·104 (0·102–0·106)	0·994 (0·994–0·994)	0·462 (0·455–0·470)	0·957 (0·956–0·957)	0·842 (0·841–0·843)
7 min (external)	0·944 (0·944–0·944)	0·106 (0·104–0·108)	0·986 (0·986–0·986)	0·270 (0·265–0·275)	0·957 (0·956–0·957)	0·798 (0·797–0·799)

Footnote Supplemental Table 5: Summary of hypotension performance metrics of the XGB model for different time frames (one, three, five, and seven minutes into the future) using different test sets. ‘Internal’ refers to the internal validation, while ‘external’ refers to the external validation. The 95% confidence interval is indicated by the values within the brackets.

Supplemental Table 6 Calibration slope and intercept

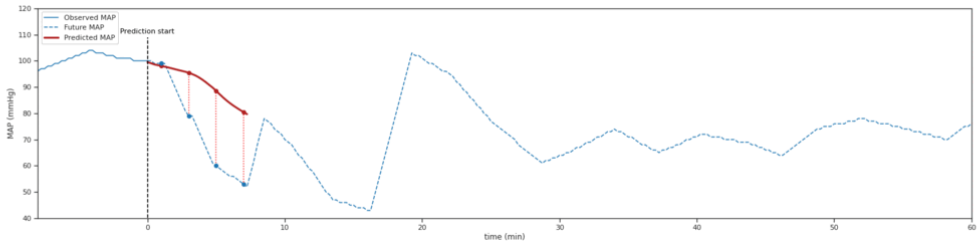
	Internal validation				External validation			
	TFT		XGB		TFT		XGB	
Forecast time	Slope	Intercept	Slope	Intercept	Slope	Intercept	Slope	Intercept
1 min	0·9	-0·01	1·00	0·0	0·54	-0·01	0·84	-0·05
3 min	0·88	-0·01	1·00	0·0	0·88	-0·01	0·78	-0·07
5 min	0·82	-0·01	1·00	0·0	0·82	-0·01	0·55	-0·04
7 min	0·79	-0·01	1·00	0·0	0·52	-0·0	0·42	-0·02

Footnote Supplemental Table 6: Calibration intercept and slope. A slope of 1 and intercept of 0 signify perfect model calibration, representing an exact match between predicted probabilities and observed fractions of positive outcomes.

Supplemental Information 1 Hyperparameters

Dropout rate: 0.3
The dropout rate, which is expressed as a percentage, indicates the proportion of neurons that are randomly turned off during each training step. In this case, the dropout rate is 0.3, which means that 30% of the neurons will be randomly turned off during each training step.
Hidden layer size: 240
The hidden layer size, which is expressed in units, defines the size of the hidden layers in the neural network. Each hidden layer consists of 240 units.
Learning rate: 0.0002
The learning rate parameter, with a value of 0.0002, specifies the rate at which the model's parameters are updated in relation to the loss gradient.
Max gradient norm: 100.0
The max gradient norm parameter, with a value of 100.0, is used for gradient clipping, which prevents the gradients from becoming excessively large during training, potentially leading to more stable training.
Minibatch size: 128
The minibatch size parameter indicates the size of each mini-batch used during training. A minibatch size of 128 means that the model processes 128 samples before updating the weights.
Number of attention heads: 32
The number of attention heads parameter defines the number of attention heads in multi-head attention mechanisms, which are often used in models such as transformers.
Stack size: 1
A stack size of 1 indicates a single layer. The total time steps parameter represents the total number of time steps in the sequence being processed
Total time steps: 62
This parameter represents the total number of time steps in the sequence being processed.
Num of encoder steps: 32
The number of encoder steps parameter indicates the number of time steps used by the encoder in sequence-to-sequence models.
Early stopping patience: 10
The early stopping patience parameter defines the patience for early stopping, meaning that the training will stop if the validation performance does not improve for 10 consecutive epochs.

Supplemental Figure 1 Dynamic blood pressure prediction



The TFT model's continuous prediction of a patient's blood pressure over time. Key moments at 1, 3, 5 and 7 minutes are highlighted with red dashed lines to indicate points of absolute prediction error. Initially predicting hypotension, the TFT model faces an unexpected challenge when an unanticipated intervention causes a sudden rise in blood pressure, significantly increasing the prediction error. However, the model demonstrates its adaptability by quickly adjusting to these changes, demonstrating the complexity and resilience of predictive models in medical scenarios.