Counterfactual Policy Introspection using Structural Causal Models

by

Michael Karl Oberst

B.A., Harvard University (2012)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2019

Author .....
Department of Electrical Engineering and Computer Science
August 30, 2019

Certified by .....
David Sontag
Associate Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Counterfactual Policy Introspection using Structural Causal Models

by

Michael Karl Oberst

Submitted to the Department of Electrical Engineering and Computer Science
on August 30, 2019, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Inspired by a growing interest in applying reinforcement learning (RL) to healthcare, we introduce a procedure for performing qualitative introspection and 'debugging' of models and policies. In particular, we make use of *counterfactual trajectories*, which describe the implicit belief (of a model) of 'what would have happened' if a policy had been applied. These serve to decompose model-based estimates of reward into specific claims about specific trajectories, a useful tool for 'debugging' of models and policies, especially when side information is available for domain experts to review alongside the counterfactual claims. More specifically, we give a general procedure (using structural causal models) to generate counterfactuals based on an existing model of the environment, including common models used in model-based RL. We apply our procedure to a pair of synthetic applications to build intuition, and conclude with an application on real healthcare data, introspecting a policy for sepsis management learned in the recently published work of Komorowski et al. (2018).

Thesis Supervisor: David Sontag
Title: Associate Professor of Electrical Engineering and Computer Science

Acknowledgments

It is a blessing to get to work on hard, interesting, and meaningful problems, and I cannot possibly thank all the people who have supported me along the way.

First and foremost, I am grateful for the love and support of my parents (Tom and Carla) and my siblings (Sarah, Aidan, Matthew, and Aaron). Thank you for being with me through everything, and know that I'll always be there for you.

I've been blessed with and supported by more friends than I can mention here, so I'll just pick one: Tim Chin, thanks for being a true friend and fellow adventurer in life. I wouldn't have gotten here today without you.

To everyone in the Clinical ML group: I couldn't have imagined a more welcoming, kind, and fun group of people with whom to spend an inordinate amount of time. You laugh at my jokes, tolerate my spontaneous and rambling musings on all things, and I couldn't ask for more than that. Special thanks to Fredrik D. Johansson, for putting up with my research-related mood swings, being a mentor in all things causal inference, and above all for being a friend.

Naturally, I would also like to thank my advisor, David Sontag. Thank you for your commitment to working on problems that matter, for encouraging me when I needed encouragement, for pushing me onwards when I needed to be pushed, and above all for your genuine care for me and the rest of your students.

Above all, I am grateful for the grace of God.

Contents

Chapter 1

Introduction

1.1 Reinforcement Learning in Healthcare: A Challenging Task

There is a long tradition of using data to improve healthcare and public health, from randomized trials to test the efficacy of new drugs, post-market surveillance for adverse drug interactions, and the practice of epidemiology more broadly, e.g., the use of observational studies to understand the public health impact of everything from cigarettes to air pollution. Over the past decade in the United States, there has also been an ever-expanding amount of raw healthcare data, driven by the rapid adoption of electronic medical records (EMRs). As the available data has expanded, so have the ambitions of some segments of the research community, fuelled by the hope that larger and richer datasets can lead to breakthroughs in personalized medicine.

With that in mind, there has been a growing interest in the application of machine learning to healthcare, not only for diagnostic purposes (e.g., image processing in radiology and pathology), but also for learning better treatment policies, tailored to individual patients. This requires solving two closely related subproblems: First, how to learn a policy from observational (that is, retrospective) data, and second, how to evaluate it.

For sequential decision-making settings in healthcare, where a *dynamic treatment*

policy[1] is required, several recent papers have used techniques from reinforcement learning (RL) to try and learn optimal policies for treating everything from sepsis (Raghu et al., 2017, 2018; Komorowski et al., 2018; Peng et al., 2018) to HIV (Parbhoo et al., 2017) and epilepsy (Guez et al., 2008). This is a challenging task, in ways that are quite different from modern success stories in reinforcement learning, such as achieving super-human performance at board games (Silver et al., 2018). The latter is a task that can be perfectly simulated, allowing for the (massive-scale) exploration and direct evaluation of different policies in a deterministic setting. In contrast, medicine is a stochastic, partially observable environment where direct experimentation by an algorithm would not be tolerable. As a result, we cannot simply try many policies and see if they work, but need to infer how a new policy would perform, using data collected under an older, different policy. In the RL literature, this is known as *off-policy evaluation*.

Of course, researchers in RL are not the first to have encountered this challenge. The evaluation of dynamic treatment policies (using observational data) is a well-studied causal inference problem in epidemiology and biostatistics, which is generally addressed with the application of g-methods, first introduced by Robins (1986). Lodi et al. (2016) and Zhang et al. (2018) are two recent examples, using g-methods to evaluate HIV treatment and anemia management strategies respectively. The techniques used in RL to evaluate novel treatment policies have much in common with these techniques, such as modelling the environment directly or re-weighting the observed data, as discussed in Chapter 2.

Quantitative evaluation is nonetheless fraught with difficulties that no mathematical method can address without making assumptions. For instance, if important variables are not measured (such as confounding variables, discussed in Section 2.1), then quantitative evaluation can give misleading results. These and other challenges, such as small effective sample sizes and miss-specification of reward, are discussed at length in Gottesman et al. (2019a).

[1] A dynamic treatment policy is one which takes intermediate outcomes into account, like stopping a medical treatment when a patient has an adverse reaction

Finally, a wealth of data exists in settings (e.g., EMRs, mobile health) that are not curated by any means, and are certainly not designed primarily for research purposes. This complicates matters further, and stands in contrast to research done with curated data registries, such as the US Renal Data System, used in Zhang et al. (2018), or sequentially randomized trials, such as the Strategic Timing of AntiRetroviral Treatment (START) trial, analyzed in Lodi et al. (2016).

1.2 Motivation: Debugging Policies and Models

Quantitative evaluation of policies can therefore be misleading for any number of reasons: There may exist unmeasured confounding in the dataset, the reward function (that is, the objective to be optimized) may be poorly specified, or there may not exist sufficient samples to evaluate policies that diverge too much from existing practice. Creating more robust methods for off-policy evaluation is an area of active research (Gottesman et al., 2019b; Liu et al., 2018; Kallus & Zhou, 2018), but a fundamental uncertainty remains.

Moreover, it may be difficult to inspect a policy directly, to determine whether or not it seems reasonable: In contrast to the epidemiological studies mentioned earlier (Zhang et al., 2018; Lodi et al., 2016) which pre-specify a dynamic policy to evaluate based on domain knowledge, it is not always clear what a reinforcement-learned policy is doing. In Raghu et al. (2017), for instance, the policy is parameterized by a neural network, and in Komorowski et al. (2018), the policy associates an action with each of 750 patient state clusters derived via k-means clustering.

With that in mind, consider the following hypothetical: Suppose that you have the power to change medical practice, and are given a complex policy which is claimed (e.g., due to off-policy evaluation) to perform far better than existing clinical guidelines. How might you proceed? Given the challenges of retrospective evaluation, you might want to test the policy prospectively, perhaps using a randomized trial. But before you did that, you would want to better *understand* the policy, before investing a large amount of time and money in a gold-standard evaluation. In essence, you may

wish to search for 'bugs' in the policy (like a tendency to take dangerous actions), or the model used to generate it (like the omission of a critical input), and iterate until you are confident that the policy has learned something reasonable.

There are a variety of ways you could do this, even if the policy is too complex to be interpretable directly. For instance, a physician might randomly select some real patients, pull up their full medical record, and compare the actions taken by the doctors to the recommendations of the learned policy, to see if they seem reasonable. Jeter et al. (2019) perform such an analysis in their critique of Komorowski et al. (2018), highlighting a sepsis patient where the learned policy makes a counter-intuitive decision to withhold treatment during a critical hypotensive episode. However, manual inspection of randomly selected trajectories may be inefficient, and difficult to interpret without more information: If we are to discover new insights about treatment, shouldn't there be *some* disagreement with existing practice?

This poses two problems: First, how do you surface the 'rationale' of a policy? In an ideal world, we could elicit a justification for each action. We refer to this as the challenge of *policy introspection*. Second, supposing that you could elicit these justifications *en masse* across all trajectories, how would you select the most interesting case examples for manual inspection?

1.3 Counterfactual Policy Introspection

In this thesis, we give a procedure that uses *counterfactual* trajectories to address both of these questions, and refer to this procedure as *counterfactual policy introspection*. Given a policy and a learned model of the environment, we provide a post-hoc method to generate counterfactual trajectories for each observed (or 'factual') trajectory, which attempt to describe what the model expects would have happened, in hindsight, if that policy had been used. We note that this is most useful in applications that already require the learning of a model of the environment, such as in model-based reinforcement learning. We can then compare counterfactual trajectories with observed trajectories, potentially with additional side-information (e.g.,

chart review in the case of a patient) so that domain experts can "sanity-check" a policy and the model used to learn it. In a way that we make precise in Section 3.2, if these counterfactuals are obviously wrong, then it provides evidence that the learned model of the environment is flawed.

Thus, our end-to-end procedure for 'debugging' models and policies is as follows, illustrated in Figure 1-1: First, once we have counterfactual trajectories for each observed trajectory, we can highlight episodes where there are surprisingly large differences between the factual and counterfactual outcomes. Second, we can then perform manual examination of the observed and counterfactual trajectories, to identify disagreements between the learned policy and existing practice, and to try and understand the rationale for them. Critically, because these are real patients, we can also go look for additional information to 'kick the tires' of the counterfactual conclusions. Finally, we can use our findings to iterate on the model and policy. For instance, looking at the medical record may suggest new variables to include in our model of the environment, at which point we can repeat the process again.

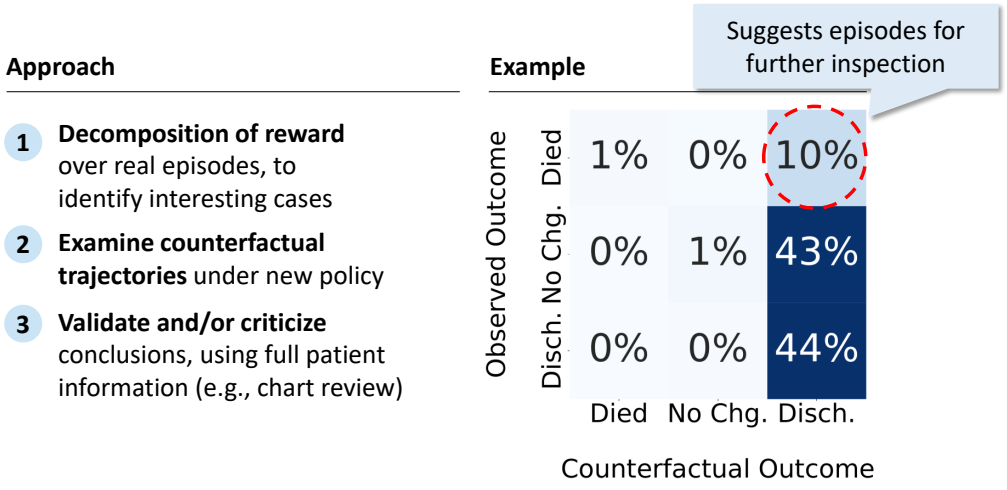


Figure 1-1: *Conceptual overview of our approach: First, counterfactual trajectories are generated for all observed trajectories, and are then used to guide manual inspection. The figure on the right is taken from a synthetic example of sepsis management in Section 6.2, and highlights patients who died, but who would have allegedly lived in the counterfactual.*

We stress that these counterfactuals are conceptually distinct from the simulation of new trajectories using a learned model of the environment. In particular, we don't want to know what the model believes might *generally* occur under a different policy: We want to know what would have been different in a *specific* trajectory. In Figure 1-2 we give a conceptual example of this distinction, in line with the medical use case described above. In this example, we imagine an observed trajectory where the patient had a rare, adverse reaction to an antibiotic. In a model-based simulation (or 'roll-out'), what might occur? Since the reaction is rare, then a model-based simulation might reasonably predict the most common outcome for patients *in general* (that the infection is cleared). Naturally, this does not satisfy our intuition for what would have happened to this specific patient (we already know!), but a model-based simulation is not designed to satisfy this intuition. A counterfactual trajectory, on the other hand, is designed to take into account what actually occurred to this patient, in a way that will be made precise in Section 2.3.

Moreover, counterfactual trajectories incorporate strictly more information about the observed trajectory, and thus exhibit less variance than a freshly simulated trajectory from a model. This is illustrated in a toy 2D grid-world setting in Figure 1-3, where the counterfactual trajectories in the left-hand figure (in blue) overlap perfectly with the observed trajectory (in black) when the actions are identical, and exhibit little variability even after actions diverge. This is in contrast to the simulated trajectories in the right-hand figure (in red), which borrow no information from the observed trajectory, and thus are different from the beginning, even under identical actions. This example is discussed in far more depth in Section 6.1.2.

Returning to our motivating example of evaluating a complex treatment policy, it is worth repeating that **these counterfactuals may be obviously wrong**, especially if we go to the medical record and use additional side information to check it against our intuition. This is a feature, not a bug, of our approach: In a setting where the model used for counterfactual evaluation is the same model that was used to train the policy, this can be used to confirm that suspicious actions (e.g., withholding treatment) are

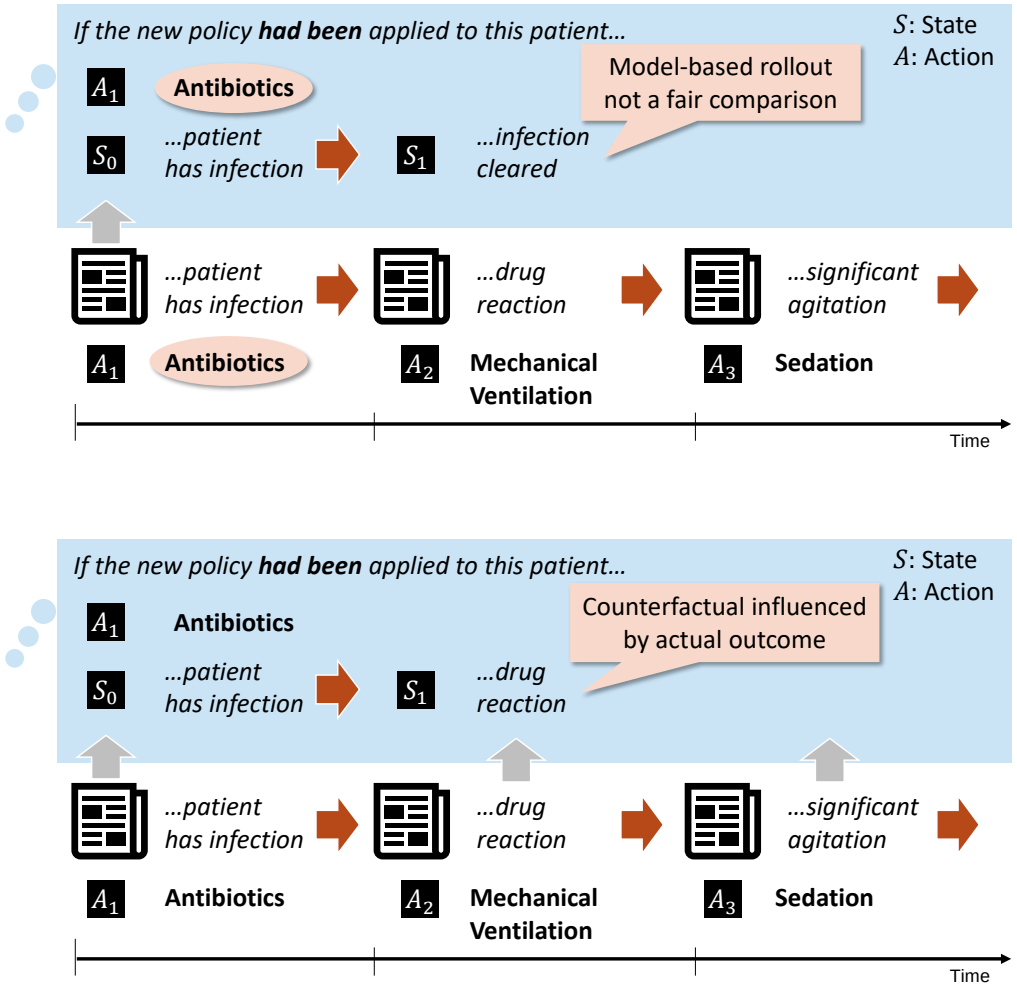
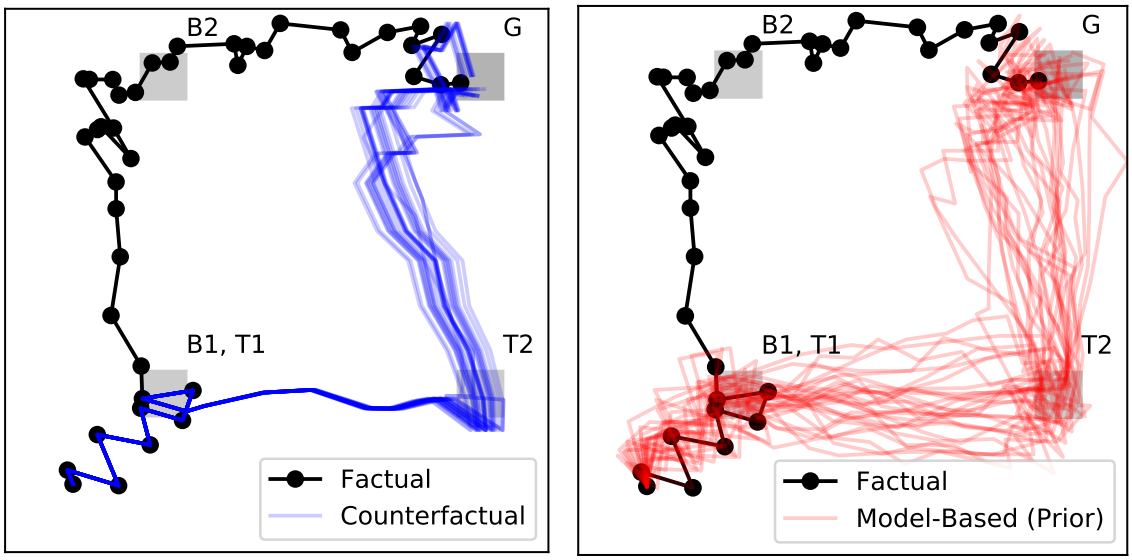


Figure 1-2: *In this example, we imagine an observed trajectory where the patient had a rare, adverse reaction to an antibiotic. In a model-based roll-out, even if the trajectory is started in the same state, with the same initial action, it is unlikely that all model-based roll-outs will include this adverse event. Thus, the model-based roll-out is harder to critique: Perhaps the model is correct, and this patient just got unlucky. A counterfactual trajectory, on the other hand, is designed to isolate differences which are due to differences in actions.*

based on a faulty model of the world, versus a real insight into the best treatment.[2] In a model-based simulation, by contrast, this is difficult to ascertain: Was the model wrong, or was this patient just one of the unlucky ones?

However, towards generating these counterfactual trajectories, we have to deal

[2]We make this intuition precise in Section 3.2



(a) **(b)**

Figure 1-3: *A visual example of how counterfactuals isolate differences that are due solely to divergence in actions from the factual, taken from Section 6.1.2. The black line represents an observed trajectory, whereas the blue and red lines represent counterfactual trajectories and model-based simulations, respectively*

with a fundamental issue of non-identifiability: As we show in Section 4.1, even with an infinite amount of interventional data, there are multiple structural causal models (as introduced in Section 2.3) which are consistent with with the data we observe, but which suggest different distributions of counterfactual outcomes on an individual level. This is not a new problem, and a common assumption in the binary setting to identify counterfactuals is the *monotonicity* condition (Pearl, 2000). However, to our knowledge, there is no analogous condition for the categorical case, as would be required to generate counterfactuals in discrete state-space models of the environment.

This motivates our main *theoretical* contribution, which is two-fold. First, we introduce a general condition of *counterfactual stability* for structural causal models (SCMs) with categorical variables and prove that this condition implies the monotonicity condition in the case of binary categories. Second, we introduce the *Gumbel-Max SCM*, based on the Gumbel-Max trick for sampling from discrete distributions, and demonstrate that it satisfies the counterfactual stability condition. We note that any discrete probability distribution can be sampled using a Gumbel-Max SCM; As

a result, drawing counterfactual trajectories can be done in a post-hoc fashion, given any probabilistic model of dynamics with discrete states. To conclude, we restate our main contributions, which are as follows:

1. **Using Counterfactuals for Policy Introspection and Model-Checking:** Our main conceptual contribution is the procedure described above, using counterfactual trajectories as a tool for introspection of learned policies and models. Additionally, we build on the theoretical results of (Buesing et al., 2019) in Section 3.2 to note that the expected counterfactual reward over all factual episodes (if the SCM is correctly specified) is in fact equal to the expected reward using freshly simulated trajectories. In this way, if counterfactual conclusions are incorrect on their face, it casts suspicion on the learned model of dynamics used in the first place, and any quantitative estimate of reward (as derived through e.g., the parametric g-formula, discussed in Section 2.1) that it yields.

2. **Counterfactual Stability and Gumbel-Max SCMs:** Our main theoretical contribution is twofold: First, we introduce the property of *counterfactual stability* for SCMs with categorical variables, and prove that this condition implies the monotonicity condition (Pearl, 2000) in the case of binary categories. Second, we introduce the Gumbel-Max SCM, a general SCM for categorical variables which we prove to satisfy the counterfactual stability condition. We note that any discrete probability distribution can be sampled using a Gumbel-Max SCM; As a result, drawing counterfactual trajectories can be done in a post-hoc fashion, given any probabilistic model of dynamics with discrete states.

3. **Application to a Real-World Setting**: In addition to a series of synthetic examples, we replicate the work of Komorowski et al. (2018) in learning a policy for sepsis management using EMR data. We apply counterfactual policy introspection with the assistance of a domain expert (in this case, a clinician), including the review of specific counterfactual trajectories using the full medical record as side information.

1.4 Structure of this Thesis

- **Background (Chapter 2):** We review the interrelated problems of learning and evaluating a dynamic policy, drawing connections between the literature on causal inference and model-based reinforcement learning. We also review the concepts necessary for generating counterfactuals, such as structural causal models. We draw a distinction between counterfactual and interventional distributions, and highlight both the inherent non-identifiability of counterfactuals, as well as the monotonicity assumption used to identify them in the binary case.

- **Counterfactual Decomposition of Reward (Chapter 3)**: We begin by demonstrating how common causal models assumed in the RL literature (MDPs and POMDPS) can be cast as structural causal models. We further discuss the connection between counterfactual estimates of rewards and notions like CATE and ITE in the causal inference literature. We conclude by building on the theoretical results of (Buesing et al., 2019) in Section 3.2 to note that the expected counterfactual reward over all factual episodes (if the SCM is correctly specified) is in fact equal to the expected reward using freshly simulated trajectories.

- **Gumbel-Max SCMs for Categorical Variables (Chapter 4)**: With the motivation from Chapter 3 in mind, in this chapter we introduce our core theoretical contributions. First, we introduce the property of categorical stability as a categorical analog of the montonicity assumption. Then, we introduce and motivate the Gumbel-Max SCM by proving that it satisfies this property. We also highlight connections to the discrete choice literature, which are useful for building intuition around the counterfactual stability condition.

- **SCMs with Additive Noise for Continuous Variables (Chapter 5)**: In this brief chapter, we highlight some possible approaches for developing general SCMs for continuous variables, by examining common continuous state-space models in RL and giving an SCM which is consistent with their formulation.

- **Illustrative Applications with Synthetic Data (Chapter 6)**: To build intuition,

we demonstrate the use of counterfactual trajectories in two idealized environments: A 2D grid-world and an illustrative simulator of sepsis. The former builds intuition for how counterfactual inference works in SCMs, while the latter demonstrates our proposed use of counterfactuals for policy introspection.

- **Real-Data Case Study: Sepsis Management (Chapter 7)**: In this chapter, we replicate the work of Komorowski et al. (2018) using real EMR data to learn a policy of sepsis management, and we apply our proposed methodology to perform introspection of the resulting policy. Most notably, we use the full medical record and the help of a clinician to examine counterfactuals for a particular trajectory, and discuss our insights from this exercise in Section 7.4.

Chapter 2

Background

In this chapter, we lay out the necessary background for the later chapters. Broadly speaking, we start by discussing the central problem of learning how to act from data. This is intrinsically a *causal* question: We would like to claim that if we acted in a particular way, this would bring about a particular outcome. Thus, in Section 2.1, we discuss some basic principles of causal inference, starting with the simplest case of estimating the effect of a binary action from interventional data (as in a randomized control trial), before moving on to techniques used to estimate the effect of dynamic treatment regimes from observational data. We highlight in particular some general classes of methods: Those which model the causal relationships directly, those which rely on re-weighting the data, and those which combine the two approaches.

With this background in hand, we turn to the problem of *learning* a policy from data, and highlight methods used in the reinforcement learning (RL) community for doing so in Section 2.2. We draw an explicit connection to the literature on dynamic treatment regimes, noting that RL methods can be viewed as assuming a particular causal graph with a certain Markov structure. With this assumption in mind, we discuss a basic method for learning an optimal policy, known as Policy Iteration, which falls under the general class of RL methods which are 'model-based', in that they assume access to a model of the environment. We then discuss two approaches in the RL literature for evaluating policies that are different from the one that generated the data, a problem known as *off-policy evaluation*: The first of

these methods, known as model-based off-policy evaluation (MB-OPE) bears some similarity to the g-formula used in the literature on evaluating dynamic treatment regimes. The second method is a re-weighting method, which is similar to inverse propensity (IP) weighting methods, another set of g-methods.

Finally, we introduce the notion of counterfactuals in Section 2.3, where we formalize the distinction between *interventional* questions, like 'what *will happen* if I apply policy X', and *counterfactual* questions, like 'what *would have happened* if I had applied policy X, given that I applied policy Y and observed outcome Z'. To do so, we introduce the mathematical framework of structural causal models, and highlight the challenges inherent in estimating counterfactuals, which are by definition never observed. We note that this is a different (and strictly more challenging) problem than the usual causal inference question, because it deals with individual-level counterfactuals (analogous to the individual treatment effect), instead of population-level causal effects (analogous to the conditional average treatment effect).

We refer the reader to several reference on the above topics for more detail, in lieu of attempting to reproduce the entirety of these fields within the confines of this thesis. In particular, we recommend Hernan & Robbins (2019) for an overview of causal inference with dynamic treatment regimes, and Pearl (2009); Peters et al. (2017) for an overview of causal graphs and structural causal models. For a general overview of reinforcement learning, we recommend Sutton & Barto (2017).

2.1 Causal Inference from Observational Data

2.1.1 Motivating Example: Binary Treatments

Suppose that we want to evaluate the causal effect of a binary action, such as taking an antibiotic, on a binary outcome, such as whether or not an infection is cleared. Let $T \in \{0, 1\}$ represent the action (whether or not we gave the treatment), and let $Y \in \{0, 1\}$ represent the outcome. Suppose we also have access to covariates / features X which describe potential confounding factors, so-called because they

influence both the treatment decision and the outcome. For any given individual, we can use Y_1 and Y_0 to represent the *potential outcomes* (Morgan & Winship, 2014) under the treatment and control respectively, of which we only observe one of the two, e.g., $Y = Y_1T + Y_0(1-T)$. We can also denote this set-up using a *causal graph*, a directed acyclic graph (DAG) which encodes the causal relationships between random variables (Pearl, 2009). In this case, the corresponding DAG is given in Figure 2-1, with arrows that represent the causal relationships between variables.

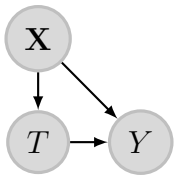


Figure 2-1: *Causal graph corresponding to the motivating example of a binary treatment and binary outcome*

In this example, we might be interested in the *average treatment effect* (ATE), which can be denoted by

$$\tau = \mathbb{E}[Y|do(T=1)] - \mathbb{E}[Y|do(T=0)],$$

where the $do(\cdot)$ operator is used to indicate an intervention. The $do(\cdot)$ operator is reviewed in (Pearl, 2009), and is accompanied by the rules of *do-calculus*, which give us a set of conditions which specify when (and how) it is possible to obtain causal relationships, like $\mathbb{P}(Y|do(T=t))$, from observed conditional relations like $\mathbb{P}(Y|T=t)$. Intuitively, the ATE corresponds to the expected difference in outcome between two policies, where we treat everyone $\mathbb{E}[Y|do(T=1)]$ or we treat no one $E[Y|do(T=0)]$. In the simplest case, if the treatment assignment is randomized such that $\mathbb{P}(T|X) = \mathbb{P}(T)$, then we have the equivalence $\mathbb{E}[Y|do(T=t)] = \mathbb{E}[Y|T=t]$. For instance, in an ideal randomized control trial with full compliance, we could estimate the causal effect by simply looking at the difference in outcome between the treatment and control groups.

2.1.2 Dealing with Observational Data

It should be noted that causal inference requires assumptions, which are often not empirically verifiable. For instance, if treatment assignment is not randomized, as is typical for observational data, a common approach is to first make the assumption of *no unmeasured confounding*: That is, we assume that we observe, through X, all of the variables which impact both the treatment and the outcome. We refer the reader to a variety of references (Hernan & Robbins, 2019; Pearl, 2009; Morgan & Winship, 2014; Imbens & Rubin, 2015) for a more comprehensive treatment of the topic, but we will briefly highlight three broad approaches, which have analogs in the reinforcement learning literature.

- First, we can model the conditional relationships directly, by estimating $\mathbb{P}(Y|X,T)$, which is equivalent to $\mathbb{P}(Y|X,do(T))$ under the assumption of no unmeasured confounding, and use this to calculate $\mathbb{P}(Y|do(T)) = \int \mathbb{P}(Y|X,T)\mathbb{P}(X)dx$ by marginalizing over X. This is known as *standardization* in epidemiology.

- Second, we can re-weight the data to create a *psuedo-population* that approximates the results of a randomized trial. For instance, we might use an estimate of the treatment probability $\mathbb{P}(T|X)$, known as the propensity score, and use this to re-weight our observations (Rosenbaum & Rubin, 1983), or stratify into sub-populations with similar propensity (Rubin & Rosenbaum, 1984). The more general form of this approach (discussed below) is known as *inverse probability (IP) weighting* in epidemiology.

- Finally, we can combine the two approaches above to develop *doubly-robust* estimators (Bang & Robins, 2005), which provide asymptotically correct estimates if we can correctly estimate either $\mathbb{P}(Y|X,T)$ or $\mathbb{P}(T|X)$.

2.1.3 ATE, CATE, and ITE

So far, we have implicitly focused on a very simple decision-making problem, by focusing on the estimation of the ATE. In effect, this corresponds to evaluating the

difference in the expected outcome between two policies: 'Treat everyone' and 'treat no one'. In the notation of potential outcomes, introduced in Section 2.1.1, the ATE corresponds to the quantity

$$\tau = \mathbb{E}[Y_1 - Y_0]$$

We can refine this further by investigating the *conditional average treatment effect* (CATE), which conditions on a specific subpopulation X, and can be denoted by the quantity

$$\tau_x = \mathbb{E}[Y_1 - Y_0|X]$$

In the causal graph given in Figure 2-1, this can (in principle) be estimated directly using regression models $\hat{f}(X,T) \approx \mathbb{E}[Y|X,T]$ since $P(Y|X,do(T)) = P(Y|X,T)$ in this case. How does this relate to learning a policy? In this simple setting, learning a policy follows naturally from evaluating the effect of the binary treatment. For instance, once we have learned the CATE, we can devise a policy which treats each patient (with covariates X) based on the sign of the estimated CATE $\hat{\tau}_x$.

Note that there is a conceptual distinction between the CATE and what we will refer to as the *individual treatment effect* (ITE), which is simply the difference in potential outcomes, denoted for an individual j by

$$\tau_{ite}^{(j)} = Y_1^{(j)} - Y_0^{(j)}$$

Unlike the ATE and CATE, this represents a statement about a specific individual, versus an expectation over a population. This can be a source of confusion when it comes to the use of counterfactual language: It is not uncommon to estimate the CATE and refer to this as a *counterfactual* or to refer to the CATE as the ITE (see Shalit et al. (2016) and discussion in Appendix B of Liu et al. (2018)).

Note that in this thesis, we will reserve the language of counterfactuals and counterfactual inference to refer to individual-level quantities, like $Y_0^{(j)}, Y_1^{(j)}$.

2.1.4 Extension to Dynamic Treatment Policies

Many of the methods which were originally developed for the simple setting described above do not work (when applied naively) to the setting where we wish to evaluate a dynamic treatment. In this setting, our initial action may have some intermediate effect which influences our choice of later actions, and so on. Robins (1986) introduced a class of general methods for adjustment in this setting, which are referred to *g-methods* in the dynamic treatment regime literature. Among these, we highlight two methods which are analogs to those discussed previously:

- First, the *g-computation algorithm formula*, typically referred to as the g-formula, is a generalization of the standardization approach given in Section 2.1.2. Simply put, the conceptual approach is to estimate the outcome under a specific policy by *simulating* from a model of the overall environment. The g-formula is widely used in epidemiology, where it is referred to as the parametric g-formula when it involves fitting a parametric model of the environment. For instance, Lodi et al. (2016) use this approach to evaluate a policy for HIV treatment, and Zhang et al. (2018) use it to evaluate a strategy for anemia management.

- Second, the class of inverse probability (IP) weighting methods, which generalize the re-weighting methods discussed previously, such as propensity score re-weighting (Rosenbaum & Rubin, 1983). See (Hernan & Robbins, 2019) for a more in-depth discussion, including the combination of IP weighting methods with marginal structural models.

2.2 Model-Based Reinforcement Learning

With all of this in mind, we shift gears to a different set of literature, namely that of reinforcement learning (RL). In contrast to the above sections, where our focus was on *evaluating* a policy based on observational data, reinforcement learning has its roots in trying to *learn* a policy efficiently, when given the ability to experiment freely in an environment. We cannot hope to summarize all the extant techniques

that exist for learning and evaluation in RL, but instead highlight those which are relevant for future chapters, as well as for understanding where our approach fits in.

Seen in relationship to the literature on dynamic treatment regimes, the reinforcement learning literature tends to assume a particular type of causal graph, a Markov Decision Process (MDP), which we describe in Section 2.2.1. While this assumption is shared across techniques used to learn a policy, there is a further distinction between methods which are *model-based*, which rely on learning to model the MDP, versus those that are 'model-free', in the sense that they do not model the environment directly. The techniques discussed in this thesis require a model of the environment, and thus we will focus our discussion in Section 2.2.2 on a simple model-based approach to learning a policy, known as Policy Iteration.

Finally, we discuss two broad types of evaluation, which have connections to the two classes of evaluation methods discussed in the previous section: First, model-based off-policy evaluation (MB-OPE), which can be seen as a specific instance of simulation via the g-formula, and importance re-weighting methods such as weighted importance sampling (WIS), which can be seen as instances of the inverse probability weighting approach described earlier.

2.2.1 Markov Decision Processes (MDPs and POMDPs)

The reinforcement learning literature tends to assume an underlying model of the world which can be represented as having a *Markov* structure, meaning that the state of the world in the future is independent of the past, given the present (observable) state. This leads to a representation which is known as a *Markov Decision Process* (MDP). This can be relaxed by assuming that there exists an underlying Markov structure, but we may not observe it, in which case it is considered a *partially observable Markov Decision Process* (POMDP). In this section we describe these general models, as a prelude to discussing their role in both learning and evaluation.

We follow the description of Finite Markov Decision Processes (MDPs) given in Sutton & Barto (2017), to which we refer the reader for more information. In this setting, the decision-maker (or *agent*) interacts with an *environment* at each discrete

time step. The decision maker is presented with a state $S \in \mathcal{S}$, and chooses an action $A \in \mathcal{A}$, which result in a new state $S' \in \mathcal{S}$ as well as a quantitative reward $R \in \mathcal{R}$, and the process continues until an absorbing state is reached, or until a fixed time (known as a fixed-horizon MDP). These states, actions, and rewards are typically indexed by time, and follow the conditional probability distribution (CPD) that governs the MDP, and which is referred to (in this work) as the *dynamics* of the process:

$$\mathbb{P}(S_{t+1}, R_t | S_t, A_t) \tag{2.1}$$

Note that the CPD in Equation (2.1) is Markov in the sense that the next state / reward only depend on the previous state and action, hence the moniker of a Markov Decision Process. Furthermore, this CPD is often assumed to be invariant to the time index, in which case we refer to this as a *homogenous* MDP. Finally, when the state space $\mathcal{S}$ has finite cardinality, we refer to this as a finite MDP.

The goal of the decision-maker at time t is typically to maximize the discounted expected reward over the future states. This is typically denoted as follows[1]

$$G_t \coloneqq \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.2}$$

In Equation (2.2), the discount factor $0 \leq \gamma \leq 1$ determines the degree to which future rewards are less valuable than immediate rewards, and this notation can be used to cover episodes which have a finite horizon or terminal states, using the assumption that after the horizon or a terminal state is reached, the subsequent rewards are all zero.

Thus, the goal of the decision-maker is to choose a policy π which maximizes the expected reward. This policy can either be deterministic, in which case $\pi : \mathcal{S} \to \mathcal{A}$ maps states to actions, or stochastic, in which case $\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ gives a probability density or mass function over the set of possible actions for each state, such that $\sum_{a \in \mathcal{A}} \pi(s, a) = 1, \forall s \in \mathcal{S}$. With a slight abuse of notation, we will sometimes write

[1] See equation 3.8 from Sutton & Barto (2017)

$\pi(a|s)$ in place of $\pi(s, a)$ to convey the fact that it describes a conditional probability distribution over actions.

An extension of this framework is to consider a *partially observable* MDP (POMDP), in which we distinguish between the true state S_t and the observation O_t at each time step, with the assumption that the true state S_t is unobserved. In this case, the generative model is augmented with the CPD $\mathbb{P}(O_t|S_t)$. In the case of a POMDP, the policy may depend on the entire *history* up to time point t, which is denoted as $H_t \coloneqq \{O_1, A_1, R_1, \ldots, O_{t-1}, A_{t-1}, R_{t-1}, O_t\}$, such that the policy is given by $\pi(a|h)$, with $h \in \mathcal{H}$ informing the action taken.

A *trajectory* or *episode*, denoted τ, is the full sequence of states, actions, and rewards, up to the terminal state or horizon. For a MDP, given a probability distribution over initial states $\mathbb{P}(S_1)$ and policy $\pi(a|s)$, the probability of any given trajectory $\tau = \{S_1, A_1, R_1, \ldots, S_T, A_T, R_T\}$ is given by

$$p(\tau) = \mathbb{P}(S_1) \prod_{k=2}^{T} \pi(A_{k-1}|S_{k-1}) \mathbb{P}(S_k, R_k|A_{k-1}, S_{k-1}) \tag{2.3}$$

With an analogous factorization in the case of a POMDP. Because this distribution depends on the policy π, we denote this distribution over τ by $p^\pi(\tau)$, and for any quantity which is a function of the trajectory (e.g., the total reward G), we will write $\mathbb{E}_\pi(\cdot)$ to denote the expected value over trajectories drawn from $p^\pi(\tau)$.

2.2.2 Policy Iteration Algorithm

There are a variety of techniques used to find an optimal policy in the case of a finite MDP, but for our purposes it will be sufficient to discuss the techniques used in (Komorowski et al., 2018), which use straightforward iterative optimization techniques that depend on knowledge of the MDP, which can be estimated from data.

First, we need to introduce the concept of the *value function* for each state, which

is defined with respect to a policy π by[2]

$$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s] \tag{2.4}$$

$$= \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a) \left[r + \gamma v_\pi(s')\right] \tag{2.5}$$

With this in hand, the *policy evaluation* problem is to estimate the value function for a given policy. Equation (2.5) defines a fixed point, and the following iterative update rule is known to converge to true value function

$$v_\pi^{(k+1)}(s) \leftarrow \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a) \left[r + \gamma v_\pi^{(k)}(s')\right], \tag{2.6}$$

where $v^{(k)}$ is the value function at the k-th iteration. Initializing a random value function and applying these updates until some desired tolerance is known as the *iterative policy evaluation* algorithm.

Using this technique for evaluating a policy as a subroutine, the *policy iteration* algorithm improves the policy at each step, using the update rule given by

$$\pi'(a|s) \leftarrow \max_a \sum_{s',r} p(s', r|s, a) \left[r + \gamma v_\pi(s')\right] \tag{2.7}$$

To summarize, policy improvement starts with a random (deterministic) policy and a randomly initialized value function, then alternates between policy evaluation and policy improvement, until it finds a stable policy. For more detail, we refer the reader to Chapters 4.1–4.3 of Sutton & Barto (2017).

2.2.3 Off-Policy Evaluation (OPE)

In the RL literature, it is commonly assumed that we are able to learn from experience. That is, we can experiment with different policies until we find a policy that maximizes our expected reward. From the perspective of healthcare applications, this is analogous to assuming that we can freely run our own randomized experiments

[2]See Equation 4.4 from Sutton & Barto (2017)

as we go along. Evaluation in this setting (the *on-policy* setting) is conceptually straightforward, similar to a randomized trial.

In this thesis, we deal with the setting where this type of experimentation is not possible, e.g., for ethical and practical reasons, and we are restricted to using observational data. This type of setting is referred to in the RL literature as *off-policy batch RL*, to reflect that the policy used to generate the data (the *'behavior' policy*) is different from the policy we wish to evaluate (the *'target'* or *'evaluation' policy*) and the fact that our dataset is restricted to a fixed batch of data.

Here we discuss two methods for off-policy evaluation, which have connections to the classes of evaluation methods discussed in Section 2.1:

- Model-based off-policy evaluation (MB-OPE) involves learning a parametric model of an underlying MDP, and then using this to estimate the value of a policy (see e.g., Chow et al. (2015); Hanna et al. (2017)), and can thus be seen as a specific instance of simulation via the g-formula.

- Importance sampling (IS) (Rubinstein, 1981) is the foundation for a series of techniques, such as weighted importance sampling (see e.g., Precup et al. (2000)). As discussed below, these are similar to IP weighting methods.

- There exist several methods for combining these approaches, whether to generate doubly robust estimates of performance (Jiang & Li, 2016; Bibaut et al., 2019; Farajtabar et al., 2018), or using a mixture of IS and MB estimates (Thomas & Brunskill, 2016; Gottesman et al., 2019a).

We take a moment here to describe the form of a basic IS estimator, as well as weighted importance sampling (WIS), as they will be relevant for our later experimental work replicating Komorowski et al. (2018). In general, importance sampling and related approaches (IP weighting, inverse propensity weighting) take advantage of the following relationship, where p, q are two different distributions

$$\mathbb{E}_p[Y] = \int y \cdot p(y)dy = \int y \cdot \frac{p(y)}{q(y)}q(y) = \mathbb{E}_q\left[\frac{p(y)}{q(y)}Y\right]$$

This is the same basic theory that underlies all the IP weighting methods discussed so far.[3] Thus, given samples of a random variable from a distribution q, we can approximate the expectation under the distribution p using the weights $p(y_i)/q(y_i)$ for each y_i, and taking a sample average $\mathbb{E}_p[Y] \approx n^{-1} \sum y_i \cdot p(y_i)/q(y_i)$

In an RL context, we want to estimate the expected reward of an evaluation policy π_e, given data sampled from an MDP under a behavior policy π_b. In this case the importance ratio is straightforward. Examining the probability of any given trajectory, given in Equation 2.3, we note that all the terms cancel in the importance sampling ratio, except for those which involve the policy. Thus, the importance sampling ratio is given by the following, where we use $\rho_{1:T}$ to denote the importance sampling ratio over T time steps

$$\rho_{1:T} = \prod_{i=1}^{T} \frac{\pi_e(a_t|s_t)}{\pi_b(a_t|s_t)}$$

Using importance sampling, we can get an unbiased and consistent estimator of the reward under the evaluation policy using $\mathbb{E}_{\pi_e}[G] \approx n^{-1} \sum_i \rho^{(i)} G^{(i)}$, where we drop the subscript on ρ, use the superscript to indicate observed trajectories, and write G as the total discounted reward. However, in practice the IS estimator can exhibit high variance, especially if some actions are rare under the behavior policy (such that $1/\pi_b(a_t|s_t)$ is very large).

Weighted importance sampling is an alternative estimator which exhibits much lower variance, albeit at the cost of introducing some bias.[4] The weighted importance sampling estimator performs a weighted average instead of a simple average, and is given by

$$\frac{\sum_i \rho^{(i)} \cdot G^{(i)}}{\sum_i \rho^{(i)}},$$

It is important to note that all variants of importance sampling are subject to the

[3]Note that this relationship is only well-defined if $p(y) > 0 \implies q(y) > 0$. This condition goes by various names depending on the field: In probability theory, it is referred to as *absolute continuity*. In the context of inverse propensity weighting, it is referred to as *overlap* or *positivity*. In the context of reinforcement learning, it is referred to as *coverage*.

[4]Weighted importance sampling is still consistent, in the sense that it converges to the correct value in the infinite data limit, with the bias asymptotically approaching zero.

same assumptions as any other causal analysis. That is, we typically need to estimate the behavior policy from data, and if there is some unmeasured confounding factor which cause our estimates of the behavior policy π_b to be incorrect, then our IS or WIS estimates will also be incorrect. This well-known fact is demonstrated in our synthetic experiments in Section 6.2.2.

2.3 Structural Causal Models and Counterfactuals

When we discussed binary treatments in Section 2.1.3 we discussed potential outcomes Y_1, Y_0. In that setting, we observe one of these, but the other is unknown, representing the theoretical counterfactual outcome. In many applications of causal inference, we wish to estimate some general effect of an intervention, such as the conditional average treatment effect $\mathbb{E}[Y_1 - Y_0|X]$ (e.g., Schulam & Saria, 2017; Johansson et al., 2016), because this represent general knowledge about interventions that we can apply to future patients. But we do not particularly care about e.g., estimating Y_0 given Y_1 for a particular patient that we have already treated, because we cannot go back in time and take a different action.

In a sense that we will make precise in Section 2.3.2, the CATE is a property of the *interventional* distribution of Y, describing how Y changes in response to interventions on other variables (in this case, T). However, we would like to go a step beyond this, as described in Section 1.3. We would like to take into account *what actually happened* to get a more precise estimate of *what would have happened* had a different action (or set of actions) been taken. This is a *counterfactual* question. In essence, we want to estimate something that is conceptually akin to the individual treatment effect $Y_1 - Y_0$, rather than just the CATE.

To do so, we need to introduce the mathematical formalism of structural causal models, which give a well-defined answer to these questions. In Section 2.3.1 we introduce the general framework, in Section 2.3.2 we formalize the conceptual distinction between interventional and counterfactual distributions, and in Sections 2.3.3-2.3.4 we discuss the fundamental challenge of non-identifiability, as well as some assumptions

that make identification possible in the binary case.

2.3.1 Structural Causal Models (SCMs)

As promised, we review the concept of *structural causal models*, and encourage the reader to refer to Pearl (2009) (Section 7.1) and Peters et al. (2017) for more details. A word regarding notation: As a general rule throughout, we refer to a random variable with a capital letter (e.g., X), the value it obtains as a lowercase letter (e.g., $X = x$), and a set of random variables with boldface font (e.g., $\mathbf{X} = \{X_1, \ldots, X_n\}$). Consistent with Peters et al. (2017) and Buesing et al. (2019), we write P_X for the distribution of a variable X, and p_x for the density function.

Definition 1 (Structural Causal Model (SCM))**.** A structural causal model $\mathcal{M}$ consists of a set of independent random variables $\mathbf{U} = \{U_1, \ldots, U_n\}$ with distribution $P(\mathbf{U})$, a set of functions $\mathbf{F} = \{f_1, \ldots, f_n\}$, and a set of variables $\mathbf{X} = \{X_1, \ldots, X_n\}$ such that $X_i = f_i(\mathbf{PA}_i, U_i), \forall i$, where $\mathbf{PA}_i \subseteq \mathbf{X} \setminus X_i$ is the subset of $\mathbf{X}$ which are parents of X_i in the causal DAG $\mathcal{G}$. As a result, the prior distribution $P(\mathbf{U})$ and functions $\mathbf{F}$ determine the distribution $P_X^{\mathcal{M}}$.

As a motivating example to simplify exposition, we will assume the causal graphs (and corresponding SCM) given in Figure 2-2. An astute reader will recognize this as the same binary setting discussed previously, representing (for example) the effect of a medical treatment T on an outcome Y in the presence of confounding variables $\mathbf{X}$.

2.3.2 Interventional vs. Counterfactual Distributions

The SCM $\mathcal{M}$ defines a complete data-generating processes, which entails the *observational distribution* $P(\mathbf{X}, Y, T)$. It also defines an *interventional distribution*, describing the effect of any possible intervention.

Definition 2 (Interventional Distribution)**.** Given an SCM $\mathcal{M}$, an intervention $I = do\left(X_i \coloneqq \tilde{f}(\tilde{\mathbf{PA}}_i, \tilde{U}_i)\right)$ corresponds to replacing the structural mechanism $f_i(\mathbf{PA}_i, U_i)$ with $\tilde{f}_i(\tilde{\mathbf{PA}}_i, U_i)$. This includes the concept of atomic interventions, where we may

Figure 2-2: *Example translation of a causal graph into the corresponding Structural Causal Model.* ***Left:*** *Causal DAG on an outcome Y, covariates X, and treatment T. Given this graph, we can perform do-calculus (Pearl, 2009) to estimate the impact of interventions such as $\mathbb{E}[Y|X, do(T = 1)] - \mathbb{E}[Y|X, do(T = 0)]$, known as the Conditional Average Treatment Effect (CATE).* ***Right:*** *All observed random variable are assumed to be generated via structural mechanisms f_x, f_t, f_y via independent latent factors U which cannot be impacted via interventions. Following convention of Buesing et al. (2019), calculated values are given by black boxes (and in this case, are observed), observed variables are given in grey, and unobserved variables are given in white.*

write more simply $do(X_i = x)$. The resulting SCM is denoted $\mathcal{M}^I$, and the resulting distribution is denoted $P^{\mathcal{M};I}$.

For instance, suppose that Y corresponds to a favorable binary outcome, such as 5-year survival, and T corresponds to a treatment. Then several quantities of interest in causal effect estimation, including (but not limited to) the ATE and the CATE, are defined by the interventional distribution, which is *forward-looking*, telling us what might be expected to occur if we applied an intervention. However, we can also define the *counterfactual distribution* which is *retrospective*, telling us what might have happened had we acted differently. For instance, we might ask: Having given the drug and observed that $Y = 1$ (survival), what *would have happened* if we had instead withheld the drug? This is formalized in an SCM as follows:

Definition 3 (Counterfactual Distribution)**.** Given an SCM $\mathcal{M}$ and an observed assignment $\mathbf{X} = \mathbf{x}$ over any set of observed variables, the counterfactual distribution $P_X^{\mathcal{M}|\mathbf{X}=\mathbf{x};I}$ corresponds to the distribution entailed by the SCM $\mathcal{M}^I$ using the posterior distribution $P(\mathbf{U}|\mathbf{X} = \mathbf{x})$.

Explicitly, given an SCM $\mathcal{M}$, the counterfactual distribution can be estimated by first inferring the posterior over latent variables, e.g., $P(\mathbf{U}|\mathbf{X} = \mathbf{x}, T = 1, Y = 1)$ in our running example, and then passing that distribution through the structural

mechanisms in a modified $\mathcal{M}^I$ (e.g., $I = do(T = 0)$) to obtain a counterfactual distribution over any variable[5]. In this way, we make precise the meaning of several terms we will use in this thesis. When we say *counterfactual inference*, we are referring to this process of obtaining a counterfactual distribution. Similarly, we sometimes use the term *counterfactual posterior* to refer to the counterfactual distribution, to reflect the fact that it is simply posterior inference in a particular type of causal model.

2.3.3 Non-Identifiability of Binary SCMs

So, given an SCM $\mathcal{M}$, we can compute an answer to our counterfactual question: Having given the drug and observed that $Y = 1$ (survival), what *would have happened* if we had instead withheld the drug? In the binary case, this corresponds to the *Probability of Necessity* (PN) (Pearl, 2009; Dawid et al., 2015), because it represents the probability that the exposure $T = 1$ was necessary for the outcome.

Intuitively, this is impossible to answer with certainty, even though we may ask ourselves these types of questions frequently in the real world. For instance, in medical malpractice, establishing fault requires just such a counterfactual claim, showing that an injury would not have occurred "but for" the breach in the standard of care (Bal, 2009; Encyclopedia, 2008).

Mathematics matches our intuition in this case: The answer to the question is not identifiable without further assumptions, a general property of counterfactual inference. That is, there are multiple SCMs which are all consistent with the interventional distribution, but which produce different counterfactual estimates of quantities like the Probability of Necessity (Pearl, 2009).

2.3.4 Monotonicity Assumption for Identification of Binary SCMs

Nonetheless, there are plausible (though untestable) assumptions we can make that identify counterfactual distributions. Consider our intuition in the following case: Suppose that a non-smoker develops lung cancer. What would have happened if they

[5]This process is called abduction, action, and prediction in Pearl (2009)

had (counterfactually) smoked a pack a day? Our intuition is that, at the very least, it would not have *helped*, and they would have developed the cancer regardless, all else being equal. This type of intuition is formalized mathematically as the *monotonicity assumption* (Pearl, 2000; Tian & Pearl, 2000), and is in fact sufficient to identify the Probability of Necessity and related quantities in epidemiology (Cuellar & Kennedy, 2018; Yamada & Kuroki, 2017).

Definition 4 (Monotonicity)**.** A SCM of a binary variable Y is monotonic relative to a binary variable T if and only if it has the following property[6,7]: $\mathbb{E}[Y|do(T=t)] \geq \mathbb{E}[Y|do(T=t')] \implies f_y(t,u) \geq f_y(t',u),\ \forall u$. We can write equivalently that the following event never occurs, in the case where $\mathbb{E}[Y|do(T=1)] \geq \mathbb{E}[Y|do(T=0)]$: $Y_{do(T=1)} = 0 \wedge Y_{do(T=0)} = 1$. Conversely for $\mathbb{E}[Y|do(T=1)] \leq \mathbb{E}[Y|do(T=0)]$, the following event never occurs: $Y_{do(T=1)} = 1 \wedge Y_{do(T=0)} = 0$.

In particular, this assumption restricts the class of possible SCMs to those which all yield equivalent counterfactual distributions over Y. For instance, the following SCM exhibits the monotonicity property, and replicates any interventional distribution where $g(x,t) = \mathbb{E}[Y|X=x, do(T=t)]$:

$$Y = \mathbb{1}\left[U_y \leq g(x,t)\right], \quad U \sim \text{Unif}(0,1)$$

In Figure 2-3 we demonstrate how this plays out for a binary treatment and outcome.

There is a wide range of literature in statistics, epidemiology, and machine learning which makes use of this assumption: In epidemiology, it implicitly appears in early work on estimating quantities like the 'relative risk ratio' (Miettinen, 1974), which are often imbued with causal interpretations (Pearl, 2009; Yamada & Kuroki, 2017). Formalizing the assumption of monotonicity, required to correctly impute causal meaning

[6] We could also write this property as conditional on X

[7] This definition differs slightly from the presentation of monotonicity in Pearl (2009), where $f_y(t,u)$ being monotonically increasing in t is given as the property, with the testable implication that $\mathbb{E}[Y|do(T=t)] \geq \mathbb{E}[Y|do(T=t')]$ for $t \geq t'$. Because the direction of monotonicity is only compatible with the corresponding direction of the expected interventional outcomes, we fold this into the definition of monotonicity directly, to align with our later definition of counterfactual stability. Also note that we use the notation $Y_{do(T=t)} \coloneqq f_y(t,u)$ here

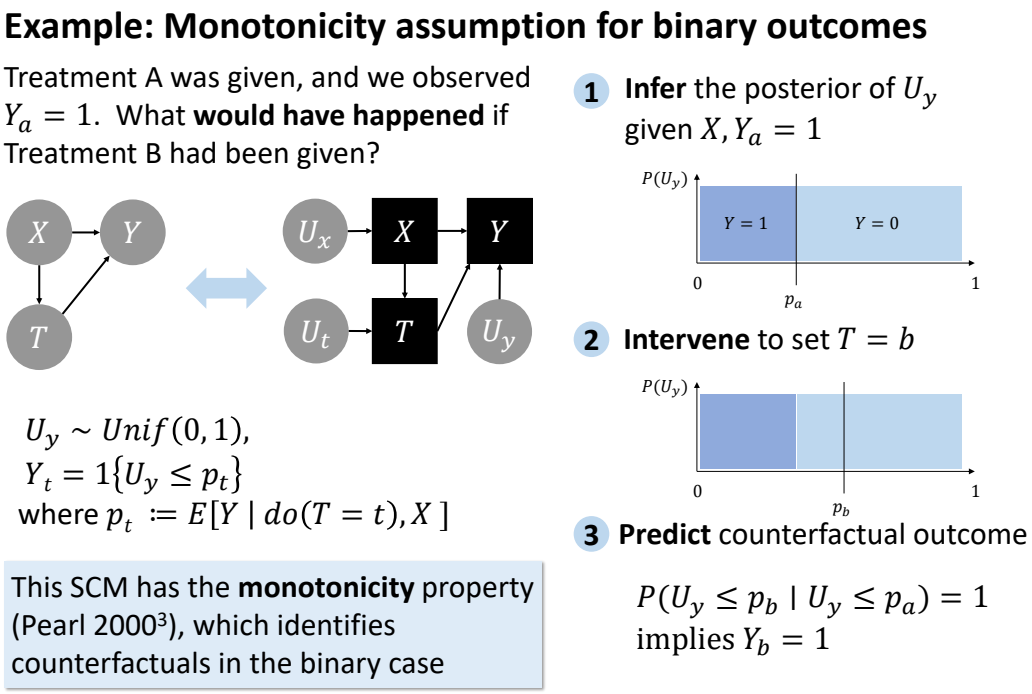


Figure 2-3: *Example of a structural causal model which satisfies the monotonicity assumption, and the process of performing counterfactual inference.*

to these quantities, is covered in Balke & Pearl (1994); Pearl (2000); Tian & Pearl (2000). More recent work in epidemiology uses the assumption of monotonicity explicitly, (e.g., to estimate the counterfactual effect of water sanitation in Kenya in Cuellar & Kennedy, 2018), and there has been ample discussion and debate regarding how this reasoning could apply (in principle) to legal cases, such as litigation around the toxic effects of drugs (Dawid et al., 2016). In statistics, monotonicity of treatment with respect to an instrumental is a core assumption of instrumental variable analysis (Imbens & Angrist, 1994). Finally, the monotonicity assumption has been used recently in the machine learning community by Kallus (2019) to classify treatment non-responders.

Chapter 3

Counterfactual Decomposition of Reward

3.1 Viewing MDPs and POMDPs as SCMs

In this section we will describe how to reformulate MDPs and POMDPs as structural causal models, retaining their implied interventional distributions while enabling the counterfactual inference procedure described previously. Critically, our results are not limited to MDPs and POMDPs, as any graphical model can be reformulated as a structural causal model. Thus, our results apply more generally wherever e.g., the parametric g-formula is used, but we focus primarily on MDPs and POMDPs in this thesis. Note that we will abuse language slightly throughout this thesis, referring to both (a) a structural causal model over all observed variables, as well as (b) the individual mechanisms for each variable (e.g., $S_{t+1} = f_s(s_t, a_t, u_{s_{t+1}})$) as structural causal models.

For a MDP, we can write the states, actions, and rewards as deterministic functions of their parents in the MDP (e.g., for any individual state, these are the previous state and action), as well as an independent exogenous variable. This is shown visually in Figure 3-1. If we are given a deterministic policy to evaluate, then the only SCMs and exogenous variables that we need to consider modelling (for the counterfactual) are those which impact the state transitions (as well as the rewards, if they are not

a deterministic function of state). For continuous state-space models, we will need a continuous SCM, as discussed in Chapter 5, and for discrete state-space models (e.g., a finite MDP), we will need a categorical SCM, as discussed in Chapter 4.

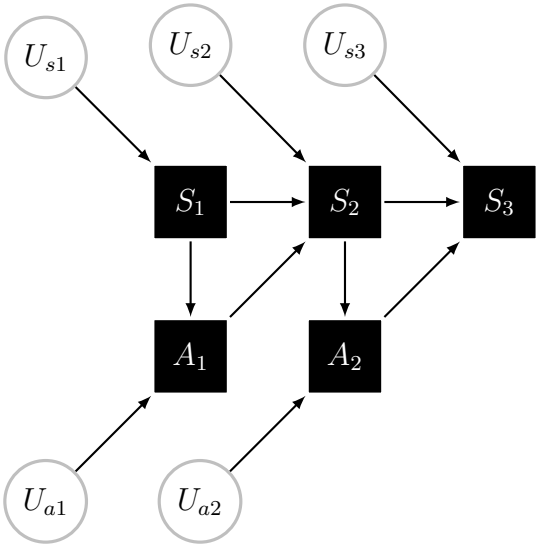


Figure 3-1: *SCM for a MDP, with states S_t and actions A_t, where the action is generated via the mechanism $\pi(U_a, S_t)$, or $\pi(S_t)$ if the policy is deterministic. Rewards are not shown for simplicity. Black squares are functions of their parents in the graph, and are observed, while white circles are unobserved random variables.*

Similarly, as noted in Buesing et al. (2019), we can view an episodic Partially Observable Markov Decision Process (POMDP) as an SCM, as shown in Figure 3-2, where S_t corresponds to states, A_t corresponds to actions, O_t corresponds to observable quantities (including reward R_t), H_t contains history up to time t, i.e., $H_t = \{O_1, A_1, \ldots A_{t-1}, O_t\}$, and stochastic policies are given by $\pi(a_t|h_t)$.

Thus, the only remaining task required to convert a MDP or POMDP into an SCM is to define the individual mechanisms in such a way that the conditional probability distributions are preserved. This will be discussed in more detail in Chapters 4-5.

For now, we will define some additional notation[1] that will prove useful later, and then discuss in Section 3.2 why this reformulation as an SCM is useful for understanding the model-based estimates of reward that a MDP or POMDP might produce. In the context of reinforcement learning with POMDPs, we are typically concerned with

[1]We also re-define some notation we used previously, for which we apologize profusely to the reader. From now on τ is a trajectory, not the ATE.

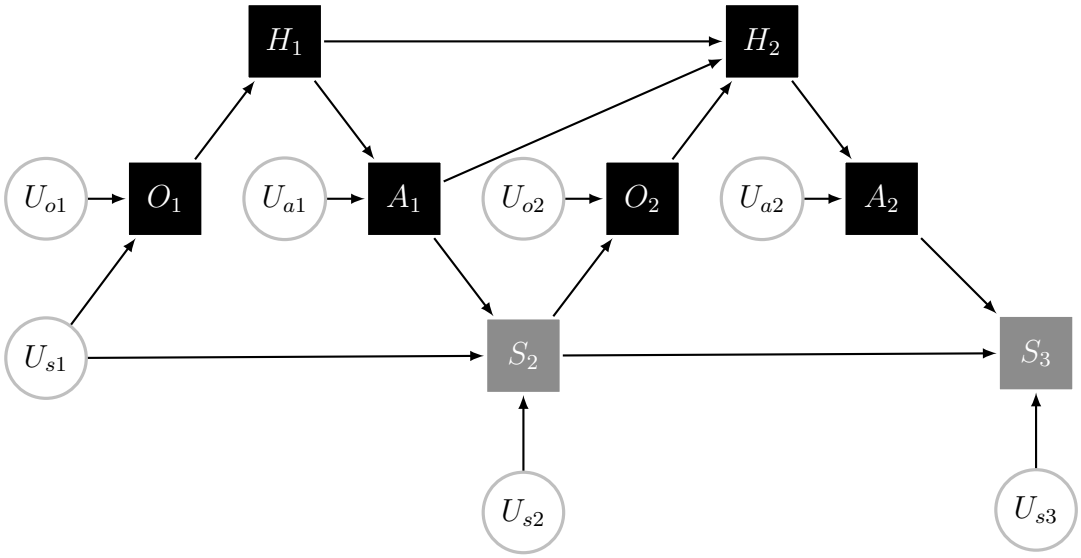


Figure 3-2: *SCM for a POMDP, slightly modified from a similar figure in Buesing et al. (2019), with initial state $U_{s1} = S_1$, states S_t, and histories H_t, where the action is generated via the mechanism $\pi(U_a, H_t)$, or $\pi(H_t)$ if the policy is deterministic. Rewards are captured as part of observed variables O for simplicity. Black and grey squares are functions of their parents in the graph, with black squares being observed and grey squares being unobserved. White circles still represent unobserved variables.*

estimating the expected reward of a proposed policy $\hat{\pi}$. To formalize notation, a given policy π implies a density over trajectories $\tau \in \mathcal{T} = (S_1, O_1, A_1, \ldots, A_{T-1}, S_T, O_T)$, which we denote as $p^{\pi}(\tau)$, and we let $R(\tau)$ be the total reward of a trajectory τ. For ease of notation, we sometimes write $\mathbb{E}_{\hat{\pi}}$ and $\mathbb{E}_{obs}$ to indicate an expectation taken with respect to $\tau \sim p^{\hat{\pi}}$ and $\tau \sim p^{\pi_{obs}}$ respectively, where $\hat{\pi}$ refers to the proposed ('target' or 'evaluation') policy, and π_{obs} to the observed ('behavior') policy.

3.2 Counterfactual Decomposition of Reward

3.2.1 Model-Based OPE as CATE Estimation

If we wish to compare the performance of a proposed policy $\hat{\pi}$ and the observed policy π_{obs}, we might compare the difference in expected reward. The expected reward under π_{obs} can be estimated in this case using observed trajectories, without a model of the environment. The difference in expected reward is conceptually similar to the average

treatment effect (ATE) of applying the proposed vs observed policy, and we denote it as δ:

$$\delta \coloneqq \mathbb{E}_{\hat{\pi}}[R(\tau)] - \mathbb{E}_{obs}[R(\tau)] \tag{3.1}$$

However, it may be useful to drill down into specific cases: Perhaps there are certain environments, for instance, in which the proposed policy would perform better or worse than the observed policy. One natural decomposition is to condition on the first observed state to estimate a conditional expected reward, e.g.,

$$\delta_o \coloneqq \mathbb{E}_{\hat{\pi}}[R(\tau)|O_1 = o] - \mathbb{E}_{obs}[R(\tau)|O_1 = o] \tag{3.2}$$

Equation 3.2 corresponds conceptually to CATE estimation, where we condition only on pre-treatment information (in this case, O_1, which occurs before the first action). However, we can go no further than that without a structural causal model, as we need a way to 'condition' on the entire observed trajectory.

3.2.2 Counterfactual OPE as ITE Estimation

Given a structural causal model, we can use information from the entire trajectory to decompose Equation (3.2) further, over actual trajectories that we have observed, to highlight differences between the observed and proposed policy. With an SCM in hand, we can decompose Equation 3.2 further as follows:

Lemma 1 (Counterfactual Decomposition of Expected Reward)**.** *Let trajectories τ be drawn from $p^{\pi_{obs}}$. Let $\tau_{\hat{\pi}}$ be a counterfactual trajectory, drawn from our posterior distribution over the exogenous U variables under the new policy $\hat{\pi}$. Note that under the SCM, $\tau_{\hat{\pi}}$ is a deterministic function of the exogenous U variables, so we can write $\tau_{\hat{\pi}}(u)$ to be explicit:*

$$\begin{aligned}
&\mathbb{E}_{\hat{\pi}}[R(\tau)|O_1 = o_1] \\
&= \int_{\tau} p^{\pi_{obs}}(\tau|O_1 = o_1)\mathbb{E}_{u\sim p^{\pi_{obs}}(u|\tau)}[R(\tau_{\hat{\pi}}(u))]d\tau
\end{aligned}$$

Proof. This proof is similar to the proof of Lemma 1 from (Buesing et al., 2019), but is spelled out here for the sake of clarity. Recall that the distribution of noise variables U is the same for every intervention / policy. Thus, $p^{\pi_{obs}}(u) = p^{\hat{\pi}}(u) = p(u)$. We will write p' and $\hat{p}$ for $p^{\pi_{obs}}$ and $p^{\hat{\pi}}$ respectively to simplify notation.

Furthermore, recall that all variables are a deterministic function of their parents in the causal DAG implied by the SCM. Most importantly, this means that the trajectory τ is a deterministic function of the policy π and the exogenous variables U. With that in mind, let $\tau_{\hat{\pi}}(u)$ indicate the trajectory τ as a deterministic function of $\hat{\pi}$ and u. We will occasionally use indicator functions to indicate whether or not a deterministic value is compatible with the variables that determine it, e.g., $\mathbb{1}\left[\tau | u, \pi\right]$ is equivalent to the indicator for $\mathbb{1}\left[\tau = \tau_{\pi}(u)\right]$. Note that the first observation is independent of the policy, and is just a function of the exogenous U, so we will write $\mathbb{1}\left[o_1 | u\right]$ in that case. For simplicity, we will remove the conditioning on O_1 to start with:

$$
\begin{aligned}
&\mathbb{E}_{\hat{p}}[R(\tau)] \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot \hat{p}(u) du && (3.3) \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot p'(u) du && (3.4) \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot \left(\int p'(\tau, u) d\tau \right) du && (3.5) \\
&= \int \int R(\tau_{\hat{\pi}}(u)) \cdot p'(u|\tau) \cdot p'(\tau) du d\tau && (3.6) \\
&= \mathbb{E}_{\tau \sim p'} \left[\int R(\tau_{\hat{\pi}}(u)) \cdot p'(u|\tau) du \right] && (3.7) \\
&= \mathbb{E}_{\tau \sim p'} \mathbb{E}_{u \sim p'(u|\tau)} \left[R(\tau_{\hat{\pi}}(u)) \right] && (3.8) \\
&= \int_{\tau} p^{\pi_{obs}}(\tau) \mathbb{E}_{u \sim p'(u|\tau)} \left[R(\tau_{\hat{\pi}}(u)) \right] d\tau && (3.9)
\end{aligned}
$$

In step (3.3) we are just using the definition of the expectation under $\hat{p}$, along with the notation $\tau_{\hat{\pi}}(u)$ to indicate that the trajectory is a deterministic function of the

exogenous u and the policy $\hat{\pi}$. In step (3.4) we replace $\hat{p}(u)$ with $p'(u)$ because they are equivalent, as noted earlier. In step (3.5) we expand $p'(u)$ over possible trajectories τ arising from the observed policy. In step (3.6) we rearrange terms and swap the order of the integral, and in step (3.7) we rewrite the outer integral as an expectation. In step (3.8) we further condense notation, and then expand in step (3.9) to match the notation in the Lemma. If we introduce the conditioning on O_1, we see that it is substantively the same.

$$
\begin{aligned}
&\mathbb{E}_{\hat{p}}[R(\tau)|o_1] \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot \mathbb{1}\left[o_1|u\right] \cdot \hat{p}(u) du && (3.10) \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot \mathbb{1}\left[o_1|u\right] \cdot p'(u) du && (3.11) \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot p'(u|o_1) du && (3.12) \\
&= \int R(\tau_{\hat{\pi}}(u)) \cdot \left(\int p'(\tau, u|o_1) d\tau \right) du && (3.13) \\
&= \int \int R(\tau_{\hat{\pi}}(u)) \cdot p'(u|\tau) \cdot p'(\tau|o_1) du d\tau && (3.14) \\
&= \int p'(\tau|o_1) \left[\int R(\tau_{\hat{\pi}}(u)) \cdot p'(u|\tau) du \right] d\tau && (3.15) \\
&= \int_{\tau} p'(\tau|o_1) \mathbb{E}_{u \sim p'(u|\tau)}[R(\tau_{\hat{\pi}}(u))] d\tau && (3.16)
\end{aligned}
$$

The main difference in this case is that is just that we carry the indicator into the prior on U at step (3.12), which we can do because O_1 does not depend on the policy that is applied. Note that Equation (3.16) matches the statement of the Lemma. □

Corollary 1 (Counterfactual Decomposition of δ_o)**.**

$$
\begin{aligned}
\delta_o &\coloneqq \mathbb{E}_{\hat{\pi}}[R(\tau)|O_1 = o_1] - \mathbb{E}_{obs}[R(\tau)|O_1 = o_1] \\
&= \int_{\tau} p^{\pi_{obs}}(\tau|O_1 = o_1) \mathbb{E}_{u \sim p^{\pi_{obs}}(u|\tau)}[R(\tau_{\hat{\pi}}(u)) - R(\tau)] d\tau
\end{aligned}
$$

Proof. By Lemma 1, we have it that

$$\begin{aligned}
\delta_o &\coloneqq \mathbb{E}_{\hat{\pi}}[R(\tau)|O_1 = o] - \mathbb{E}_{obs}[R(\tau)|O_1 = o] \\
&= \int_\tau p'(\tau|o_1)\mathbb{E}_{u\sim p'(u|\tau)}[R(\tau_{\hat{\pi}}(u))]d\tau \\
&\quad - \int_\tau p'(\tau|o_1)\mathbb{E}_{u\sim p'(u|\tau)}[R(\tau_{\pi_{obs}}(u))]d\tau \\
&= \int_\tau p^{\pi_{obs}}(\tau|O_1 = o_1)\mathbb{E}_{u\sim p^{\pi_{obs}}(u|\tau)}[R(\tau_{\hat{\pi}}(u)) - R(\tau)]d\tau
\end{aligned}$$

Note that in the last step, we recognize that $\mathbb{P}_{u\sim p'(u|\tau)}[\tau_{\pi_{obs}}(u) = \tau] = 1$, because the posterior density over u is zero for all u such that $\tau_{\pi_{obs}}(u) \neq \tau$. □

Corollary 1 implies that we can decompose the expected difference in reward between the policies into differences on *observed episodes* over counterfactual trajectories, if the SCM is correct. In the context of Buesing et al. (2019), this fact is used to argue that counterfactuals approximate draws from the interventional distribution, since efficient estimation of the latter is their ultimate goal.

In our case this fact serves an additional purpose: It *theoretically motivates the use of counterfactuals as a model-checking tool*. In principle, if the SCM is correct, then the counterfactuals can be used to identify how observed episodes contribute to overall estimates of reward, and thus ground the model-based conclusions in specific counterfactual claims that can be vetted by domain experts. In practice, we consider this decomposition a heuristic, as we do not believe the SCM is necessarily correct. That said, our empirical work in Chapters 6-7 gives anecdotal evidence that this equality holds approximately in some situations when the learned MDP is not correct.

Chapter 4

Gumbel-Max SCMs for Categorical Variables

In the previous chapter, we illustrated how to convert a model of the environment into a structural causal model, as well as the potential benefits of doing so for the purpose of decomposing model-based rewards into counterfactual claims. All that remained was to specify the specific causal mechanisms for each of the variables in the respective SCMs.

However, it is at this point that we face a non-identifiability issue: Multiple SCMs can all entail the same interventional distribution, but a different set of counterfactual trajectories, and therefore a different decomposition under Lemma 1. This motivates the theoretical work of this chapter: We must make our assumptions carefully, as they cannot be tested by data, so it is worth investigating which assumptions are consistent with our causal intuition. We illustrate this non-identifiability (with respect to categorical distributions) in Section 4.1. Then we introduce the condition of *counterfactual stability* (in Section 4.2) for a discrete distribution on k categories, and show that it is compatible with the monotonicity condition of Pearl (2000) in that it implies the monotonicity assumption when $k = 2$. Then we introduce the Gumbel-Max SCM for discrete variables in Section 4.3, and prove that it satisfies the counterfactual stability condition, and in Section 4.4 describe an intuitive connection to discrete choice models.

4.1 Non-Identifiability of Categorical SCMs

We will first illustrate that the non-identifiability of counterfactual distributions applies to categorical distributions as well. Consider the categorical distribution over k categories, e.g., the transition kernel $P(S'|S = s, A = a)$ over discrete states. Let $p_i \coloneqq P(S' = i|S = s, A = a)$. There are multiple ways that we could sample from this distribution with a structural mechanism f and latent variables U. For instance, we could define an ordering **ord** on the categories, and define k intervals of $[0, 1]$ as $[0, p_{\mathbf{ord}(1)}), [p_{\mathbf{ord}(1)}, \sum_{i=1}^{2} p_{\mathbf{ord}(i)}), \ldots, [\sum_{i=1}^{k-1} p_{\mathbf{ord}(i)}, 1]$. Then we could draw $U \sim Unif(0, 1)$, and return the interval that u falls into.

However, different permutations **ord** will yield equivalent interventional distributions but can imply different counterfactual distributions. For instance, consider the following example, shown visually in in Figure 4-1. Let $k = 4$ and $p_1 = p_2 = 0.25, p_3 = 0.3, p_4 = 0.2$ and consider an intervention $A = a'$ which defines a different distribution $p'_1 = 0, p'_2 = 0.25, p'_3 = 0.25, p'_4 = 0.5$. Now consider two permutations, $\mathbf{ord} = [1, 2, 3, 4]$ and $\mathbf{ord}' = [1, 2, 4, 3]$, and the counterfactual distribution under a' given that $S' = 2, A = a$. In each case, posterior inference over U implies that $P(U|S' = 2, S = s, A = a) \sim Unif[0.25, 0.5)$. However, under **ord** this implies the counterfactual $S' = 3$, while under $\mathbf{ord}'$ it implies $S' = 4$.

Note that in this example, the mechanism $f_{\mathbf{ord}}$ implied a non-intuitive counterfactual outcome: Even though the intervention $A = a'$ *lowered* the probability of $S' = 3$ (relative to the probability under $A = a$) without modifying the probability of $S' = 2$, it led to a delta distribution in the counterfactual posterior on $S' = 3$. Since all choices for **ord** imply the same interventional distribution, there is no way to distinguish between these mechanisms with data.

This motivates the following sections, where we posit a desirable property for categorical SCMs to possess, and which rules out this result (among others) and is compatible with the notion of monotonicity introduced by Pearl (2000). We then demonstrate that a mechanism based on sampling independent Gumbel variables satisfies this property, which motivates the use of the Gumbel-Max SCM for performing

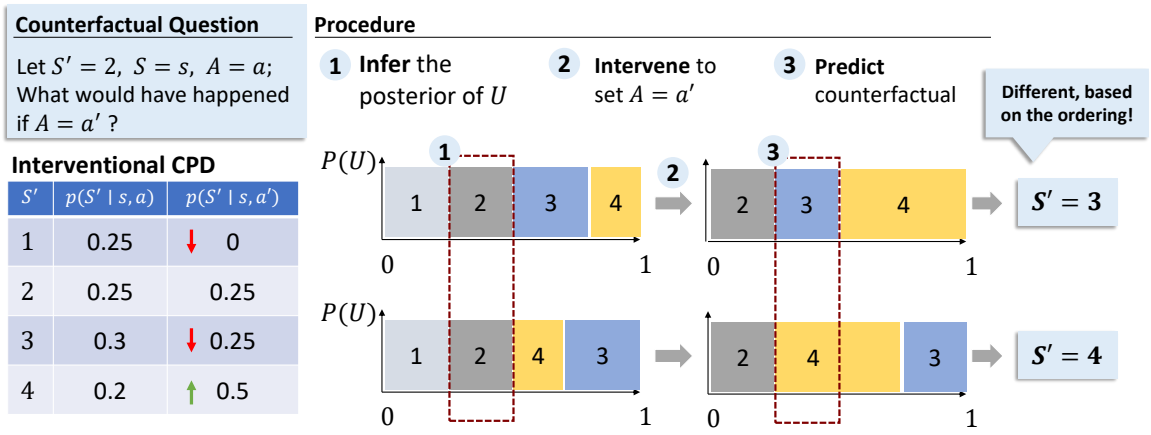


Figure 4-1: *Example of non-identifiability of categorical counterfactual outcomes. The table on the bottom right illustrates the difference in the conditional probability distribution (the 'interventional' distribution) as a function of actions a versus a'. The procedure is illustrated in the middle, where the two rows represent two possible orderings (**ord** and **ord'**) both of which define a causal mechanism $S' = f(S, A, U)$ with $U \sim Unif(0, 1)$ that replicates the interventional probability distribution. From left to right, we see the application of counterfactual inference: (1) Infer the posterior of U, represented by the red box, (2) intervene to set $A = a'$, and (3) predict the counterfactual by evaluating under the posterior of U. These two SCMs produce different counterfactual outcomes, with the outcome of $S' = 3$ being particularly unintuitive, since the interventional probability was reduced under the shift from a to a'.*

counterfactual inference in this setting.

4.2 Counterfactual Stability Property

We now introduce our first contribution, the desired property of *counterfactual stability* for categorical SCMs with k categories, laid out in in Definition 5. This property would rule out the non-intuitive counterfactual implications of $f_{\mathbf{ord}}$ in Section 4.1. We then demonstrate that this condition implies the monotonicity condition when $k = 2$.

First, with apologies to the reader, we will once again introduce some notation. Denote the interventional probability distribution of a categorical variable Y with k categories as $P^{\mathcal{M};I}(Y) = \mathbf{p}$ under intervention I, and $\mathbf{p}'$ under intervention I', where $\mathbf{p}, \mathbf{p}' \in \Delta^k$, the probability simplex over k categories. To simplify notation for interventional outcomes, we will sometimes denote by Y_I the observed outcome Y under intervention I, and denote by $Y_{I'}$ the counterfactual outcome under intervention I', such that p_i and $P(Y_I = i)$ are both equivalent to $P^{\mathcal{M};I}(Y = i)$, and similarly for I'. For counterfactual outcomes, we will write $P^{\mathcal{M}|Y_I=i;I'}(Y)$ for the counterfactual distribution of Y under intervention I' given that we observed $Y = i$ under the intervention I.

Definition 5 (Counterfactual Stability)**.** A SCM of a categorical variable Y satisfies *counterfactual stability* if it has the following property: If we observe $Y_I = i$, then for all $j \neq i$, the condition $\frac{p'_i}{p_i} \geq \frac{p'_j}{p_j}$ implies that $P^{\mathcal{M}|Y_I=i;I'}(Y = j) = 0$. That is, if we observed $Y = i$ under intervention I, then the counterfactual outcome under I' cannot be equal to $Y = j$ unless the multiplicative change in p_i is less than the multiplicative change in p_j.

Corollary 2. *If $\mathcal{M}$ is a SCM which satisfies counterfactual stability, then if we observe $Y_I = i$, and $\frac{p'_i}{p_i} \geq \frac{p'_j}{p_j}$ holds for all $j \neq i$, then $P^{\mathcal{M}|Y_I=i;I'}(Y = i) = 1$.*

This definition and corollary encode the following intuition about counterfactuals: If we had taken an alternative action that would have *only increased* the probability of $Y = i$, without increasing the likelihood of other outcomes, then the same outcome would have occurred in the counterfactual case. Moreover, in order for the outcome

to be different under the counterfactual distribution, the relative likelihood of an alternative outcome must have increased relative to that of the observed outcome. The connection to monotonicity is given in Theorem 1, whose proof is deferred to Section 4.5.

Theorem 1. *Let $Y = f_y(t, u)$ be the SCM for a binary variable Y, where T is also a binary variable. If this SCM satisfies the counterfactual stability property, then it also satisfies the monotonicity property with respect to T.*

4.3 Gumbel-Max SCMs Satisfy Counterfactual Stability

Unlike monotonicity with binary outcomes and treatments, the condition of counterfactual stability does not obviously imply any closed-form solution for the counterfactual posterior. Thus, we introduce a specific SCM which satisfies this property, and discuss how to sample from the posterior distribution in a straightforward fashion. We start by recalling the following fact, known as the Gumbel-Max trick (Luce, 1959; Yellott, 1977; Yuille. & L, 2011; Hazan & Jaakkola, 2012; Maddison et al., 2014; Hazan et al., 2016; Maddison et al., 2017):

Definition 6 (Gumbel-Max Trick)**.** We can sample from a categorical distribution with k categories as follows, where $\tilde{p}_i$ is the unnormalized probability $P(Y = i)$: First, draw $g_1, \ldots, g_k$ from a standard Gumbel, which can be achieved by drawing $u_1, \ldots, u_k$ iid from a Unif$(0, 1)$, and assigning $g_i = -\log(-\log u_i)$. Then, set the outcome j by taking $\arg\max_j \log \tilde{p}_j + g_j$.

Clearly, we can perform this for any categorical distribution, e.g., the transition distribution $p_i = P(S' = i|S, A)$; In particular, for any discrete variable Y whose parents in a causal DAG are denoted $\mathbf{X}$, a *Gumbel-Max SCM* assumes the following causal mechanism, where $\mathbf{g} = (g_1, \ldots, g_k)$ are independent Gumbel variables:

$$Y = f_y(\mathbf{x}, \mathbf{g}) \coloneqq \arg\max_j \{\log P(Y = j|\mathbf{X} = \mathbf{x}) + g_j\}$$

Like any mechanism which replicates the conditional distribution under intervention, this mechanism is indistinguishable from any other causal mechanism based on data alone. That said, it does satisfy the property given in Definition 5.

Theorem 2. *The Gumbel-Max SCM satisfies the counterfactual stability condition.*

The intuition is that, when we consider the counterfactual distribution, the Gumbel variables are fixed. Thus, in order for the argmax (our observed outcome) to change in the counterfactual, the log-likelihood of an alternative outcome must have increased relative to our observed outcome.

We note that posterior inference in the Gumbel-Max SCM is straightforward. Given a Gumbel-Max SCM as defined above, where $Y = \arg\max_j \log p_j + g_j$ and $p_j \coloneqq P(Y_I = j)$, we can draw Monte Carlo samples from the posterior $P(\mathbf{g}|Y_I = i)$ using one of two approaches: First, we can use rejection sampling, drawing samples from the prior $P(\mathbf{g})$ and rejecting those where $i \neq \arg\max_j \log p_j + g_j$. Alternatively, it is known (Maddison et al., 2014; Maddison & Tarlow, 2017) that in the posterior, the maximum value and the argmax of the shifted Gumbel variables $\log p_j + g_j$ are independent, and the maximum value is distributed as a standard Gumbel (in the case of normalized probabilities). Thus, we can sample the maximum value first, and then sample the remaining values from shifted Gumbel distributions that are truncated at this maximum value. Then, for each index j, subtracting off the location parameter $\log p_j$ will give us a sample of g_j. We can then add this sample $\mathbf{g}$ to the log-probabilities under I' (i.e., $\log \mathbf{p}'$) and take the new argmax to get a sample of the counterfactual outcome Y under intervention I'.

4.4 Intuition: Connection to Discrete Choice Models

The Gumbel-Max sampling mechanism was initially introduced in the discrete-choice literature (Luce, 1959), where it is used as a generative model for decision-making under utility maximization (Train, 2002; Aguirregabiria & Mira, 2010), where the log probabilities may be assumed to follow some functional form, such as being linear in

features. This is motivated by understanding the impact of different characteristics on consumer choices, see (Aguirregabiria & Mira, 2010, Example 1).

We discuss this connection further in this section, but note the contrast with our approach: Whereas the traditional discrete choice literature assumes a particular functional form (e.g., linear in features) for the log probabilities, we decouple this structural mechanism (for estimation of counterfactuals) from the statistical model used to estimate the conditional probability distributions under interventions. We encourage the reader to consult (Train, 2002, e.g., Chapter 2) for more details, but we highlight some relevant pieces of intuition below, with their connection to the counterfactual stability condition.

Discrete choice models that utilize Gumbel noise are known in the econometrics literature as *logit discrete choice models*, and are part of a broader class of discrete-choice models which are derived on the principle of utility maximization, known as *random utility models*. This literature is motivated by consumers as decision-makers, deciding between different discrete alternatives. In the context of modelling state transitions in an MDP, we can make the analogy that the 'decision-maker' is nature, and the choice is the next discrete state. First, we introduce two core assumptions: The concept of random utility maximization, which is introduced as a core assumption of discrete-choice models in (Train, 2002), and the assumption of additive separability.

Random Utility Maximization We assume that the decision-maker acts to optimize utility. In particular, the decision-maker associates some utility U_i with each discrete choice / alternative i, and chooses the alternative i if and only if $U_i > U_j \ \forall i \neq j$. Because U is not observed directly, we treat it as a random variable. We only observe the conditional probability distribution on Y, known as the *conditional choice probability*, given by

$$P(Y = i|X) = \int \mathbb{1}\left[U_i > U_j, \forall j \neq i\right] p(U|X) dU \tag{4.1}$$

Additive Separability Without loss of generality, the utility U can be re-written in terms of a deterministic component which depends on observable variables X, and an unobserved component ϵ, so that $U_j = V_j + \epsilon_j$, where V is assumed to be a function of observable variables, and is called the *representative utility*. With that in mind, Equation (4.1) can be rewritten as

$$\begin{aligned} &P(Y = i|X) \\ &= \int \mathbb{1}\left[V_i(x) + \epsilon_i > V_j(x) + \epsilon_i, \forall j \neq i\right] p(\epsilon|X) d\epsilon \end{aligned} \tag{4.2}$$

The assumption of *additive separability* states that the unobserved components ϵ are independent of the observed components, i.e., $\epsilon \perp\!\!\!\perp X$. Tying these assumptions back to our proposed notion of counterfactual stability, the implication from a counterfactual perspective is that if we intervene on the variables X, we do not change the values of ϵ as a result of additive separability. Thus, the assumption of random utility maximization implies that if we observe $Y_x = i$, then a necessary condition for substituting j for i is that

$$V(x')_j - V(x)_j > V(x')_i - V(x)_i \tag{4.3}$$

Different choices of discrete-choice models imply different functional forms for V and different distributions on ϵ. In the logit model, the ϵ_i variables are assumed to be drawn iid (over alternatives i) from a Gumbel distribution (also known as a Type 1 Extreme Value distribution). This implies that

$$P(Y = i|X) = \frac{\exp V_i(x)}{\sum_j \exp V_j(x)} \tag{4.4}$$

Because any scaling or shifting of the utility is irrelevant, we can set the scale of V such that $V_i = \log p_i$, consistent with Equation (4.4), and see that Equation (4.3) corresponds to the counterfactual stability condition.

4.5 Appendix: Proofs

Theorem 1. Let $Y = f_y(t, u)$ be the SCM for a binary variable Y, where T is also a binary variable. If this SCM satisfies the counterfactual stability property, then it also satisfies the monotonicity property with respect to T.

Proof. To simplify notation further, let $p^{t=1} \coloneqq P(Y = 1|do(T = 1))$, $p^{t=0} \coloneqq P(Y = 1|do(T = 0))$, and let $Y_t \coloneqq Y_{do(T=t)}$. Without loss of generality, assume that $p^{t=1} \geq p^{t=0}$.

To show that counterfactual stability implies monotonicity, we want to show that the probability of the event $(Y_1 = 0 \wedge Y_0 = 1)$ is equal to zero. We will do so by proving both cases: First that $P^{\mathcal{M}|Y_0=1;do(T=1)}(Y = 0) = 0$ and second that $P^{\mathcal{M}|Y_1=0;do(T=0)}(Y = 1) = 0$. We can start with the assumption that $p^{t=1} \geq p^{t=0}$ and write:

$$\begin{aligned} & p^{t=1} \geq p^{t=0} \\ \implies & p^{t=1}(1 - p^{t=0}) \geq p^{t=0}(1 - p^{t=1}) \\ \implies & \frac{p^{t=1}}{p^{t=0}} \geq \frac{(1 - p^{t=1})}{(1 - p^{t=0})} \end{aligned}$$

Using the counterfactual stability condition, the last inequality implies that if we observe $Y_0 = 1$, then the counterfactual probability of $Y_1 = 0$ is equal to $P^{\mathcal{M}|Y_0=1;do(T=1)}(Y = 0) = 0$, as desired. For the second case, where we observe $Y_1 = 0$, we can simply manipulate the inequality to see that

$$\frac{(1 - p^{t=0})}{(1 - p^{t=1})} \geq \frac{p^{t=0}}{p^{t=1}}$$

Which yields the conclusion that $P^{\mathcal{M}|Y_1=0;do(T=0)}(Y = 1) = 0$, as desired, completing the proof. □

Theorem 2. The Gumbel-Max SCM satisfies the counterfactual stability condition.

Proof. Recall that we write the shorthand $p_i \coloneqq P^{\mathcal{M};I}(Y = i)$, and $p'_i \coloneqq P^{\mathcal{M};I'}(Y = i)$. Suppose that Y is generated from a Gumbel-Max SCM $\mathcal{M}$ under intervention I, and

we observe that $Y_I = i$. The Gumbel-Max SCM implies that almost surely:

$$\log p_i + g^{(i)} > \log p_j + g^{(j)} \quad \forall j \neq i \tag{4.5}$$

To demonstrate that the Gumbel-Max SCM satisfies the counterfactual stability condition, we need to demonstrate that $\frac{p'_i}{p_i} \geq \frac{p'_j}{p_j} \implies P^{\mathcal{M}|Y_I=i;I'}(Y = j) = 0$ for all $j \neq i$.

We will proceed by proving the contrapositive, that for all $j \neq i$, $P^{\mathcal{M}|Y_I=i;I'}(Y = j) \neq 0 \implies \frac{p'_i}{p_i} < \frac{p'_j}{p_j}$.

Fix some index $j \neq i$. The condition $P^{\mathcal{M}|Y_I=i;I'}(Y = j) \neq 0$ implies that there exist values $g^{(i)}, g^{(j)}$ such that

$$\log p'_i + g^{(i)} < \log p'_j + g^{(j)} \tag{4.6}$$

Because the Gumbel variables $g^{(i)}, g^{(j)}$ are fixed across interventions, this implies there exist values for these variables which satisfy both inequalities (4.5) and (4.6). Thus, we proceed by subtracting inequality (4.5) from inequality (4.6), maintaining the direction of the inequality and cancelling out the Gumbel terms. The rest is straightforward manipulation using the monotonicity of the logarithm.

$$\begin{aligned} \log p'_i - \log p_i &< \log p'_j - \log p_j \\ \log(p'_i/p_i) &< \log(p'_j/p_j) \\ (p'_i/p_i) &< (p'_j/p_j) \end{aligned}$$

This demonstrates that $P^{\mathcal{M}|Y_I=i;I'}(Y = j) \neq 0 \implies (p'_i/p_i) < (p'_j/p_j)$ as desired, and taking the contrapositive completes the proof. □

Chapter 5

SCMs with Additive Noise for Continuous Variables

In this brief chapter, we collect some thoughts on structural causal models that reflect the conditional probability distribution of continuous random variables. Although this is not the primary focus of this thesis, we include it here for completeness, as a reference for how the conceptual ideas of this thesis (e.g., decomposition of reward and investigation of counterfactual trajectories) can be applied in the continuous setting.

In contrast to the categorical case, we do not have specific non-identifiability examples for continuous SCMs, nor do we have a corresponding assumption, analogous to counterfactual stability, which suggests specific SCMs for this case. However, we note that a common model assumed in this case takes the form of Equation 5.1, where the next state $s_{t+1} \in \mathbb{R}^n$ is assumed to follow a Gaussian distribution conditioned on the previous state $s_t \in \mathbb{R}^n$ and action $a \in \mathcal{A}$, and the mean and covariance are determined by arbitrary functions $\mu_\theta : \mathbb{R}^n \times \mathcal{A} \to \mathbb{R}^n$ and $\Sigma_\theta : \mathbb{R}^n \times \mathcal{A} \to \mathbb{R}^{n\ timesn}$ of the previous state and action. For instance, in Chua et al. (2018), these are parameterized by neural networks, and Σ_θ is a diagonal covariance.

$$\mathbb{P}(s_{t+1} \mid s_t, s_t) = \mathcal{N}(\mu_\theta(s_t, a_t), \Sigma_\theta(s_t, a_t)) \tag{5.1}$$

This particular model can be re-written equivalently as the following SCM with

additive noise that is drawn independently at each time step, where we write L_θ as the Cholesky decomposition of Σ_θ such that $L_\theta L_\theta^T = \Sigma_\theta$. In the case where Σ_θ is a diagonal covariance, as in Chua et al. (2018), this is simply the element-wise square root of Σ_θ.

$$s_{t+1} = \mu_\theta(s_t, a_t) + L_\theta(s_t, a_t) \cdot \epsilon_t \tag{5.2}$$

$$\epsilon_t \sim \mathcal{N}(0, I_n) \tag{5.3}$$

In a similar fashion, many models of dynamics used for reinforcement learning in continuous state spaces can be re-formulated as structural causal models with additive noise that follows some known distribution. Moreover, if the only source of stochasticity in the transitions is an additive term which is an invertible function of ϵ_t, as in Equation 5.2, then counterfactual inference is particularly simple, as the exogenous term ϵ_t can be identified exactly from the observable prediction error.

Chapter 6

Illustrative Applications with Synthetic Data

In this chapter, we develop some intuition for how counterfactuals could be used in practice, using some illustrative applications. First, in Section 6.1 we use a toy example of a 2D gridworld to illustrate the differences between counterfactual trajectories and model-based trajectories. Then we give an illustrative example of how counterfactuals could be used to 'debug' a policy in Section 6.2, using a synthetic environment of sepsis management. We note that all code required to replicate these synthetic experiments will be made available at `https://github.com/clinicalml/cf-policy-introspection`.

6.1 Building Intuition: 2D Gridworld

To illustrate the concepts behind counterfactual trajectories, we start with a simple 2D example, inspired by a similar experimental setup in (Gottesman et al., 2019b).[1] In Section 6.1.1, we describe the simulator setup, and in Section 6.1.2 we demonstrate how counterfactual inference proceeds in this setting. Finally, we show in Section 6.1.3 how this enables us to decompose differences in reward (between a target and behavior policy) across individual episodes.

[1]We thank Omer Gottesman for providing the original code used in his work

As an addendum, in Section 6.1.4 we demonstrate how counterfactuals take maximum advantage of the information present in a trajectory, by making inferences over all sources of variation, not only a single per-trajectory latent variable.

6.1.1 Simulator Setup

In this example, the agent is navigating a 2D domain, with state $s \in [0, 1]^2$ and four possible actions $a \in \{[0, 0.1], [0.1, 0], [0, -0.1], [-0.1, 0]\}$ corresponding to the four cardinal directions (north, east, south, and west). The goal of the agent is to reach the *goal region* $G = \{(x, y) : x \in [0.9, 1.0], y \in [0.9, 1.0]\}$, and the reward is -1 at each time point until the agent enters the goal region, when it receives a reward of $+10$. The dynamics are as follows

$$s_{t+1} = s_t + a_t + w(s_t; \beta) + \epsilon_t$$

Where $\epsilon_t \sim \mathcal{N}(0, I\sigma_\epsilon^2)$ represents time-varying gusts of wind, and $w(s_t; \beta) = [-\beta \cdot y_t, 0]$ is a cross-wind which pushes in either the western or eastern direction, with a magnitude that increases as the agent progresses north. The β parameter is drawn uniquely for each instance from a Gaussian $\beta \sim \mathcal{N}(\mu_\beta, \sigma_\beta^2)$. We will refer to this as the *prior* on β, but we note that this just represents the *population-level* distribution of β, and could be the posterior population distribution after many trajectories have been observed. We call it a prior to distinguish from the counterfactual posterior over the particular β in each trajectory, which we will seek to infer as part of performing counterfactual inference. Thus, the entire generative model is given by the following, where $\pi(s_t)$ is a deterministic policy which we describe in the next section.

$$\beta \sim \mathcal{N}(\mu_\beta, \sigma_\beta^2) \tag{6.1}$$

$$\epsilon_t \sim \mathcal{N}(0, I\sigma_\epsilon^2) \tag{6.2}$$

$$a_t = \pi(s_t) \tag{6.3}$$

$$s_{t+1} = s_t + a_t + w(s_t; \beta) + \epsilon_t \tag{6.4}$$

We can view the trajectories as arising from the POMDP / Structural Equation Model given in Figure 6-1, where we leave out the rewards for the sake of simplifying exposition.

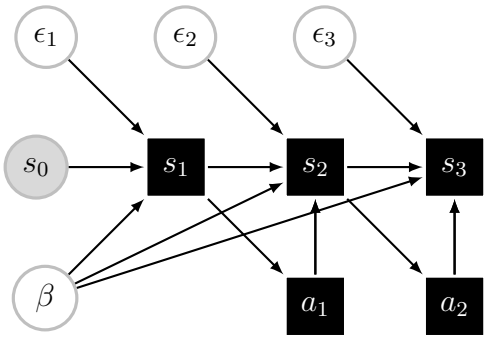


Figure 6-1: *The structural causal model model for our 2D sequences, where each black box is a deterministic function of its parents, and the initial state s_0 is an observed random variable. In practice, all of our sequences start at the same position, so s_0 is a deterministic value.*

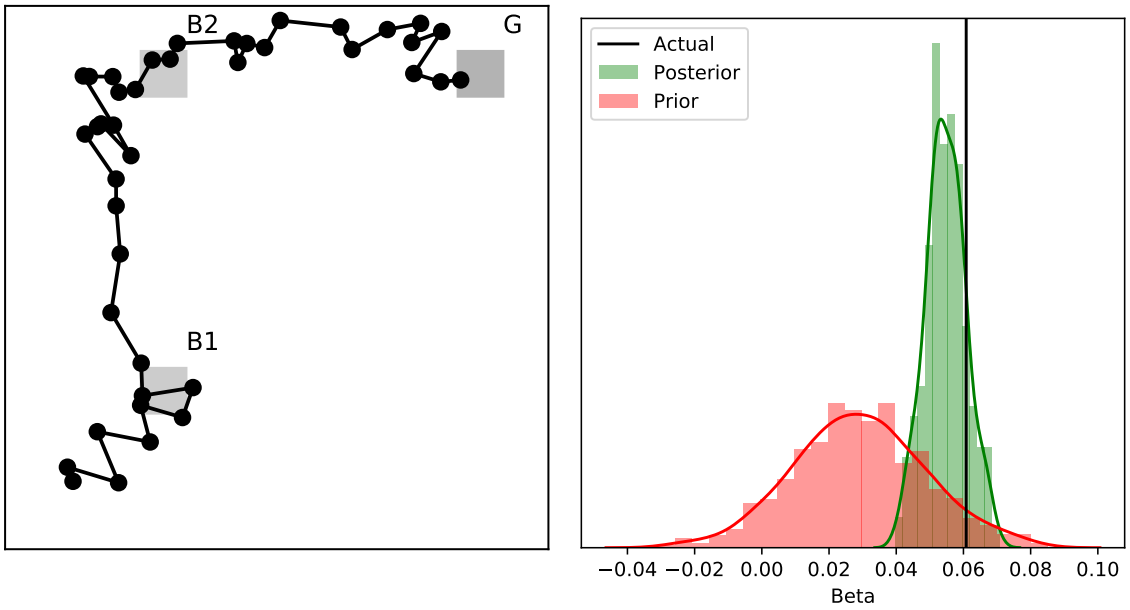
6.1.2 Generating Counterfactual Trajectories

In Figure 6-2a we plot a trajectory from this model, which we will use as a running example, where $\sigma_\epsilon = 0.001, \mu_\beta = 0.03, \sigma_\beta = 0.02$. This trajectory follows a myopic behavior policy $\pi_b(s_t)$, which is defined with respect to a series of 'checkpoint' regions that the agent must enter before heading to the goal region in the top right, and at each time point it takes the action which will minimizes the ℓ_2 distance between a naive prediction $s'_{t+1} = s_t + a_t$ and the center of the next region. In this case, the policy π_b seeks to traverse the regions denoted B1, B2 before seeking the region denoted G. The particular draw of β in this case is 0.061.

In this setting, counterfactual inference starts with posterior inference over β, ϵ, which factorizes as

$$p(\beta, \epsilon|\mathbf{x}, \mathbf{y}, \mathbf{a}) = p(\epsilon|\beta, \mathbf{x}, \mathbf{y}, \mathbf{a})p(\beta|\mathbf{x}, \mathbf{y}, \mathbf{a}).$$

Thus, the first step is posterior inference over β, the results of which are given in



(a) *An observed (factual) trajectory where $\beta = 0.061$, which traverses the regions B1, B2 before seeking the goal region G*

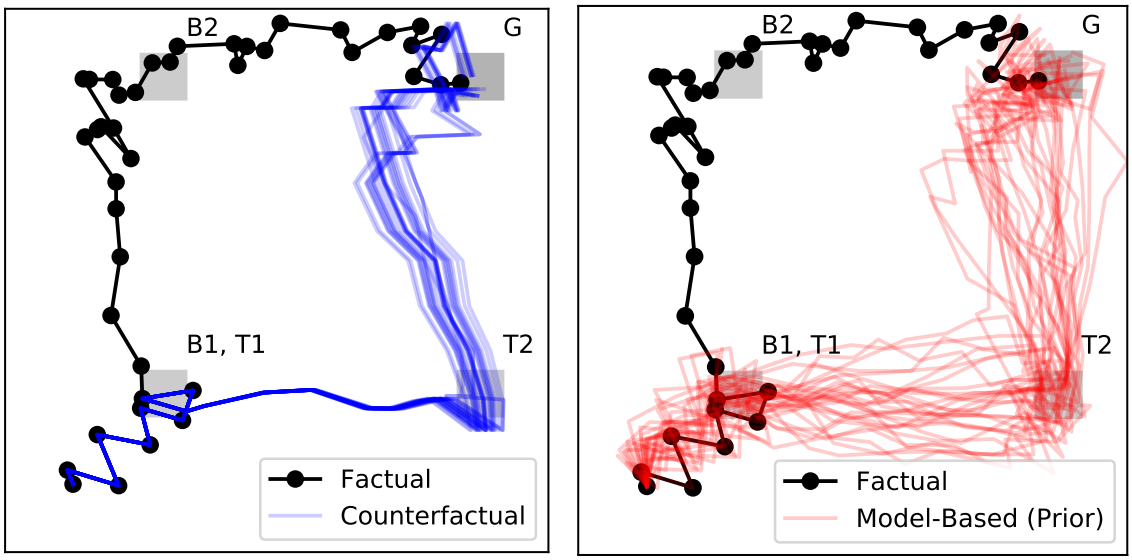
(b) *Prior versus posterior distribution over the value of β for this specific instance*

Figure 6-2: *Factual trajectory and posterior over latent variable β*

Figure 6-2b using MCMC.[2] Note that once we draw a sample of β from the posterior, we can uniquely identify ϵ from Equation 6.4, so the only uncertainty in the counterfactual is due to β.

The advantage of the counterfactual approach is that it allows us to associate a set of counterfactual trajectories with every factual trajectory. This is demonstrated in Figure 6-3a, where we generate counterfactual trajectories under a target policy π_t which seeks to traverse a different set of checkpoints (T1, T2) before heading to the goal region G. We make two notes about the counterfactual trajectories, contrasting them with model-based trajectories in Figure 6-3b (generated using the model given by Equations (6.1-6.4), starting at the same point): First, the counterfactual trajectories are identical to the factual trajectory up until the checkpoint B1/T1, because both policies take the same actions up until that point. Second, the counterfactual trajectories have much less variation because they incorporate all the information from the original trajectory (including both time-dependent and time-independent

[2]We use Pyro (Bingham et al., 2018) to perform MCMC.



(a) **(b)**

Figure 6-3: *In both cases, the target policy $\pi_t(s_t)$ is used, which seeks to pass through checkpoints T1 and T2 before proceeding to the goal region G. In (a) we see 30 trajectories from the counterfactual posterior, which can be contrasted with (b) where we see 30 trajectories sampled from the generative model given by Equations (6.1-6.4), starting from the same point.*

variation) through the posterior, whereas a model-based roll-out starting from the same point does not.

6.1.3 Decomposition of Reward via Counterfactuals

We can use these counterfactuals to associate with each factual trajectory an expected reward under the counterfactual 'had the target policy been used instead', and use this to examine where those differences are projected to be largest. Figure 6-4 demonstrates this over 100 factual trajectories (which follow the behavior policy) and their expected counterfactual reward (under the target policy)[3]. We plot the factual reward observed against the counterfactual reward[4], and shade each point according to the expected value of β under the posterior. This visually demonstrates that the difference in reward is greatest for larger values of β, but does so in a way that can

[3]We used 30 counterfactual trajectories for each factual trajectory, in order to compute the expected counterfactual reward.

[4]One point is excluded from the plot, with a factual / counterfactual reward of approximately -60 and -18 respectively.

be tied back to individual episodes.

In a real-data application, this type of analysis can be done in an exploratory fashion, to (a) search for trajectories where the difference in reward is estimated to be largest, and (b) examine what differentiates those trajectories from the others.

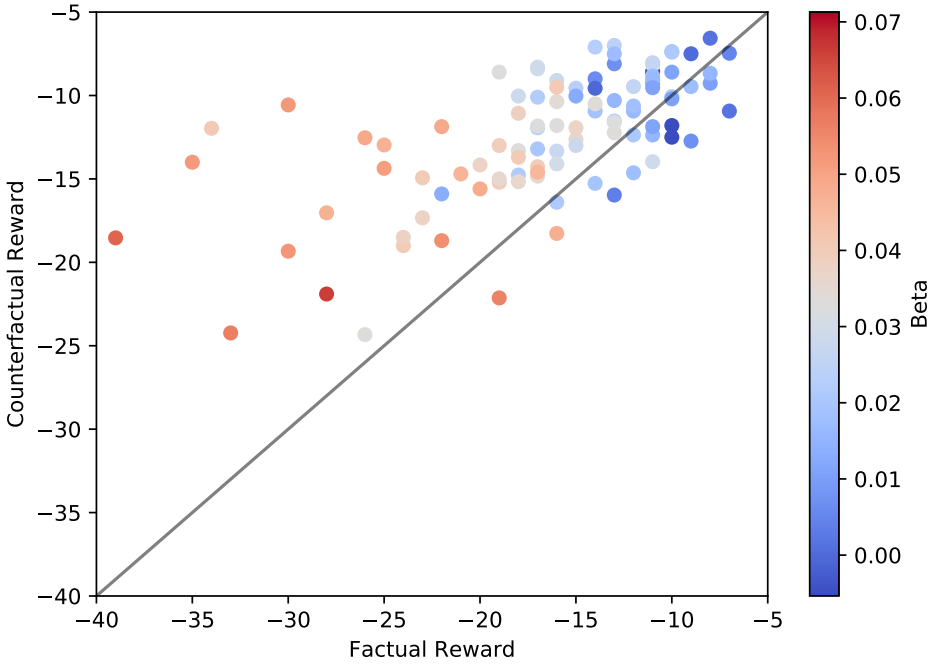


Figure 6-4: *Decomposition of reward*

6.1.4 Addendum: Counterfactual vs. Model-Based Trajectories

In this section, we demonstrate how counterfactuals take maximum advantage of the information present in a trajectory, by making inferences over all sources of variation. In this case, we make inferences over both β and ϵ, and this allows us to draw a contrast with two other ways that, conceptually, we could have generated trajectories from the same model.

1. **Model-based roll-out**: Sample a new $\beta \sim p(\beta)$, and then sample a new $\epsilon_t \sim p(\epsilon)$ at each time step. Given a deterministic policy, these random parameters imply a fixed trajectory.

2. **Model-based roll-out (posterior on β)**: Sample $\beta \sim p(\beta|\mathbf{s}, \mathbf{a})$, and then sample a new $\epsilon_t \sim p(\epsilon)$ at each time step.

3. **Counterfactual roll-out**: Sample $\beta, \boldsymbol{\epsilon} \sim p(\beta, \boldsymbol{\epsilon}|\mathbf{s}, \mathbf{a})$, which in this model is equivalent to sampling a value for β from the posterior, and then inferring the unique value of ϵ_t for each time step.

To demonstrate the differences between these approaches, we use two environments: The first environment is the same as the one described above (with $\sigma_\epsilon = 10^{-3}, \mu_\beta = 0.03, \sigma_\beta = 0.02$), and the second has a lower variance over ϵ but a higher prior variance on β (with $\sigma_\epsilon = 10^{-4}, \mu_\beta = 0.03, \sigma_\beta = 0.04$). A single trajectory is sampled from each environment[5], and are given in Figure 6-5, along with the resulting posterior over β.

With these two environments in hand, we can explore the differences between the three approaches given above. This is illustrated in Figure 6-6. In particular, we note the drawbacks of the second approach (generating a posterior over β alone), which has the appealing feature that it does not require a structural causal model with deterministic functions, only a graphical model with some time-independent latent factor β. Intuitively, this approach will face two drawbacks, which are illustrated in Figure 6-6:

- First, it is not guaranteed to replicate the same outcomes if the same actions are taken, violating our intuition for how a counterfactual should behave. This can be seen in Figure 6-6, where the counterfactuals are the only trajectories that exhibit this behavior.

- Second, it will ignore the information provided by ϵ, leading to unnecessary variance in the roll-out. If we have a SCM which is an accurate representation of the environment (as we do in this case), we can reduce the variance substantially by taking this information into account, especially when the variance of ϵ is high. This is also seen in Figure 6-6, where in the top row there is little (visual) difference between the two model-based approaches.

This concludes our conceptual example, which should drive home the idea that, if we have an accurate SCM of the environment, we can construct counterfactual

[5]The trajectory from the prior section is used for the first environment, for continuity

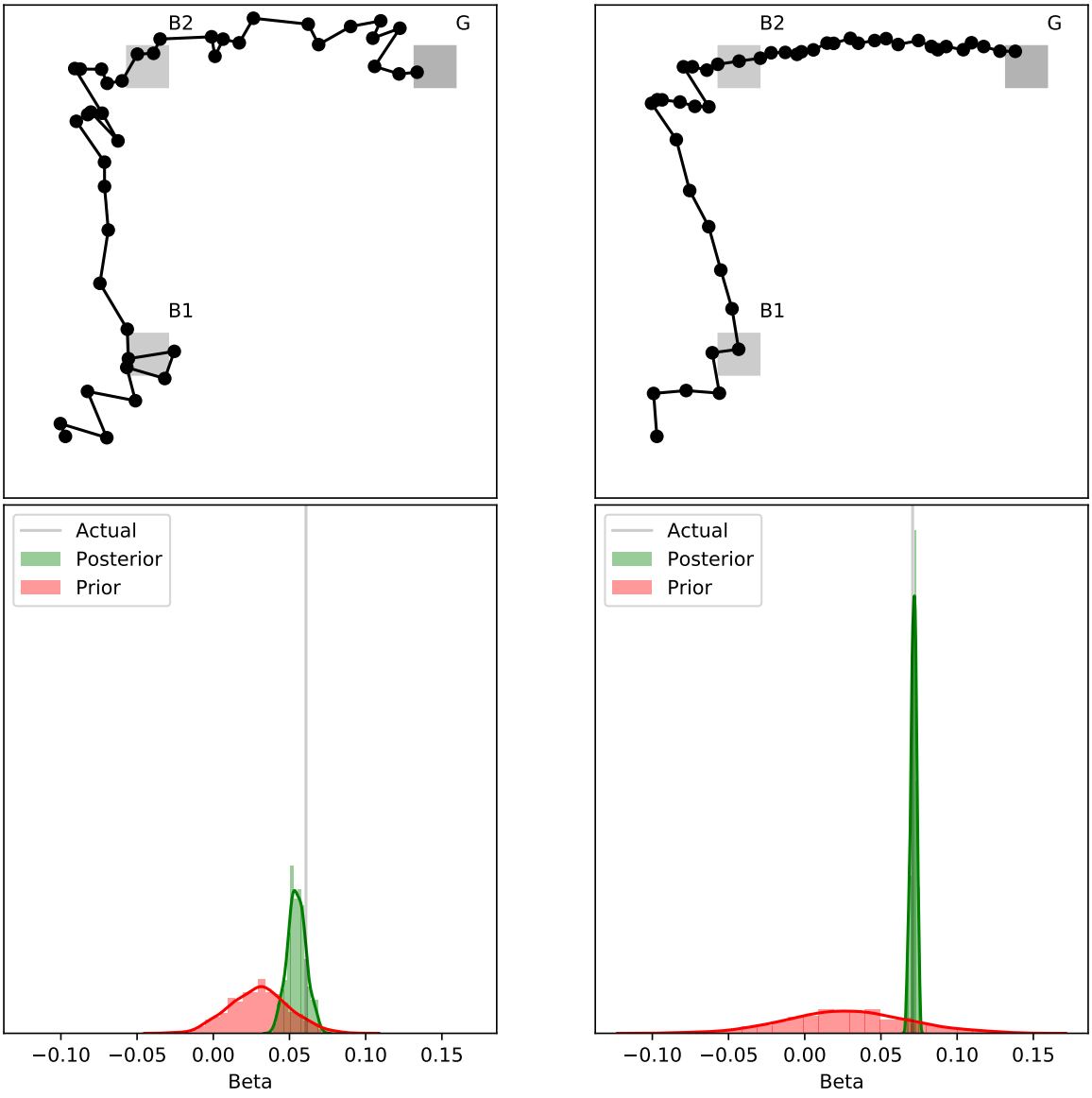


Figure 6-5: *A single trajectory sampled from each of the two environments. The latter environment has a higher prior variance over β, and a lower variance over ϵ. Below the trajectories are the corresponding prior and posterior distributions over β.*

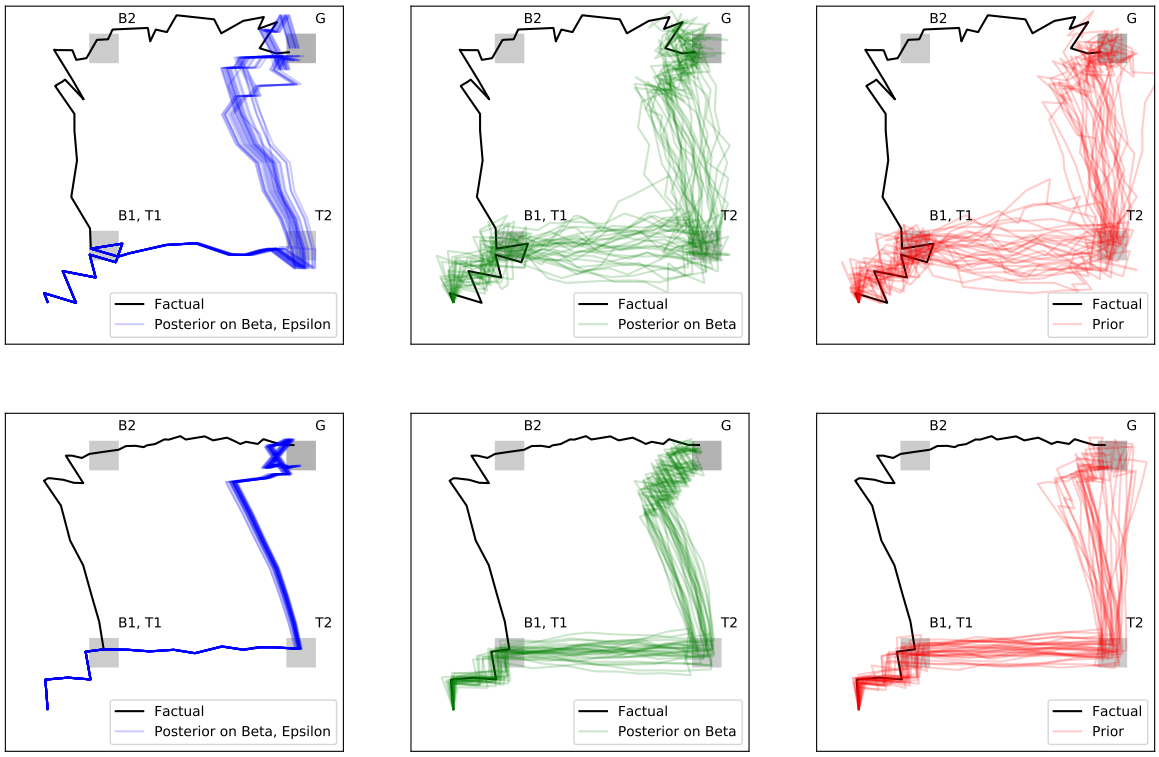


Figure 6-6: *Comparison of the three approaches given above. The first row represents the first environment, where* $\sigma_\epsilon = 10^{-3}, \sigma_\beta = 0.02$*, and the second row represents the second environment, where* $\sigma_\epsilon = 10^{-4}, \sigma_\beta = 0.04$*. The black trajectory represents a factual trajectory, and is constant across the columns. From left to right, we have counterfactual trajectories (which use the posterior on* β *and* ϵ*), model-based trajectories which only use a posterior on* β*, and model-based trajectories which use neither, just sampling from the prior. Note that in the first row, using the posterior on* β *does not reduce the variation as much as it does in the second row, due to the differences in* σ_ϵ*.*

trajectories which (a) allow us to decompose differences in reward across individual episodes, and (b) are easier to contrast with the original trajectory than other model-based trajectories (e.g., without using a SCM), through modelling all sources of variation in the factual trajectory. In particular, using a SCM allows us to isolate only the differences which are due to the change in policy, keeping all independent sources of variation constant. In the next section, we will take this a step further, and show how counterfactuals can help us 'debug' a policy and model of an environment, even if our SCM is not entirely correct.

6.2 Illustrative Example: Sepsis Management

As discussed in Chapter 1, our hope is to provide a method for qualitative introspection and 'debugging' of RL models, in settings where a domain expert could plausibly examine individual trajectories. We give an illustrative example of this use case here, motivated by recent work examining the use of RL algorithms for treating sepsis among intensive-care unit (ICU) patients. In particular, we use a simple simulator of sepsis and "debug" a RL policy that is learned on observed trajectories. This replicates an analysis originally presented in our publication (Oberst & Sontag, 2019).[6]

An analysis like this requires three ingredients: First, we are given *observed trajectories*, but cannot directly interact with the environment[7]. Second, we have access to a *structural causal model* of the environment. In this case, that model is a finite MDP, learned based on observed samples, combined with the assumption of a Gumbel-Max SCM for transition distributions. Finally, we need a *target policy* to evaluate. We refer to the policy which generated the data as the *behavior* policy, to distinguish it from the target policy.

In Sections 6.2.1-6.2.2 we describe our illustrative scenario, in which a target RL policy appears to perform well using off-policy evaluation methods such as weighted importance sampling, when it is actually much worse than the behavior policy. In Sections 6.2.3-6.2.4 we then demonstrate how our method could be used to identify a promising subset of trajectories for further introspection, and uncover the flaws in the target policy using side information (e.g., chart review of individual patients).

6.2.1 Setup of Illustrative Example

Environment: Our simulator includes four vital signs (heart rate, blood pressure, oxygen concentration, and glucose levels) with discrete states (e.g., low, normal, high),

[6]We also thank Christina X. Ji and Fredrik D. Johansson for their work on developing an earlier version of the sepsis simulator.

[7]We do not assume access to a simulator; In this example, it is used only for obtaining the initial observed trajectories

along with three treatment options (antibiotics, vasopressors, and mechanical ventilation), all of which can be applied at each time step. Reward is +1 for discharge of a patient, and -1 for death. Discharge occurs only when all patient vitals are within normal ranges, and all treatments have been stopped. Death occurs if at least three of the vital signs are simultaneously out of the normal range. In addition, a binary variable for diabetes is present with 20% probability, which increases the likelihood of fluctuating glucose levels.

Observed Trajectories: For the purposes of this illustration, the behaviour policy was constructed using Policy Iteration (Sutton & Barto, 2017) with full access to the parameters of the underlying MDP (including diabetes state). This was done deliberately to set up a situation in which the observed policy performs well. To introduce variation, the policy takes a random alternative action w.p. 0.05. Using this policy, we draw 1000 patient trajectories from the simulator, with a maximum of 20 time steps. If neither death nor discharge is observed, the observed reward is zero.

Structural Causal Model: For this illustration, we 'hide' glucose and diabetes state in the observed trajectories; Given this reduced state-space, we learn the parameters of the finite MDP by using empirical counts of transitions and rewards from the 1000 observed trajectories, with death and discharge treated as absorbing states. For state / action pairs that are not observed, we assume that any action leads to death, and confirm that this results in a target policy which never takes an action that has never been observed. For counterfactual evaluation, we make the assumption that the transitions are generated by a Gumbel-Max SCM.

Target Policy: The target policy is learned using Policy Iteration on the parameters of the learned MDP. Because the target policy is learned using a limited number of samples, as well as an incomplete set of variables, it should perform poorly relative to the behavior policy.

Further details of the simulator can be found in the source code, which will be made available at `https://www.github.com/clinicalml/cf-policy-introspection`.

6.2.2 Off-Policy Evaluation Can Be Misleading

First, we demonstrate what might be done to evaluate this target policy without the use of counterfactual tools. In Figure 6-7, we compare the observed reward of the actual trajectories against the estimated reward of the target policy. Using weighted importance sampling on the given trajectories, the target policy appears superior to the behavior policy. We also use the parameters of the learned MDP to perform model-based off-policy evaluation (MB-PE), using the MDP as a generative model to simulate trajectories and their expected reward. Both of these suggest that the target policy is superior to the behavior policy. In reality, the target policy is inferior (as expected by construction), as verified by drawing new samples from the simulator under the target policy. This corresponds conceptually to what would happen if the target policy were deployed "in the wild".

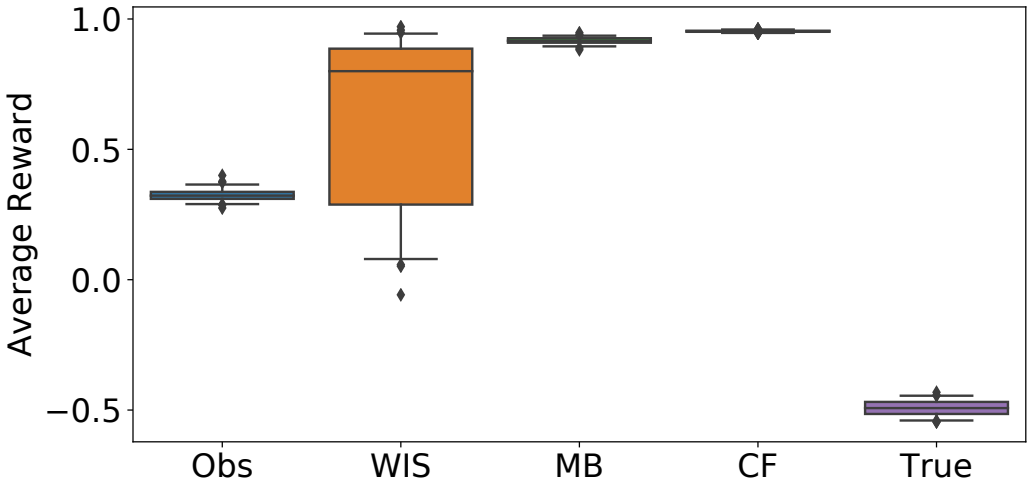


Figure 6-7: *Estimated reward under the target (RL) policy, with 95% uncertainty intervals generated through 100 bootstrapped samples (with replacement) of the same 1000 observed trajectories (for 1-4) and of 1000 new trajectories under the target policy (for 5).* ***(1) Obs****: Observed reward under the behavior policy.* ***(2) WIS****: Estimated reward under the target policy using weighted importance sampling.* ***(3) MB****: Estimated reward using the learned MDP as a generative model.* ***(4) CF****: Estimated reward over counterfactual trajectories (5 per observed trajectory).* ***(5) True****: Observed reward under the target policy, over 1000 newly simulated trajectories.*

With this in mind, we demonstrate how examining individual counterfactual tra-

jectories gives insight into the target policy. The first step is to apply counterfactual off-policy evaluation (CF-PE) using the same MDP and the Gumbel-Max SCM. This yields similarly optimistic results as MB-PE. However, by pairing counterfactual outcomes with observed outcomes of individual patients, we can investigate *why* the learned MDP concludes (wrongly) that the target policy would be so successful.

6.2.3 Identification of Informative Trajectories

To debug this model (without access to a simulator), we can start by drawing counterfactual trajectories for each individual patient under the target policy. With these in hand, we can assign each individual patient to one of nine categories, based on the most frequently occurring counterfactual outcome (death, no change, or discharge) in Figure 6-8. This highlights individual trajectories for further analysis, as discussed in the next section[8].

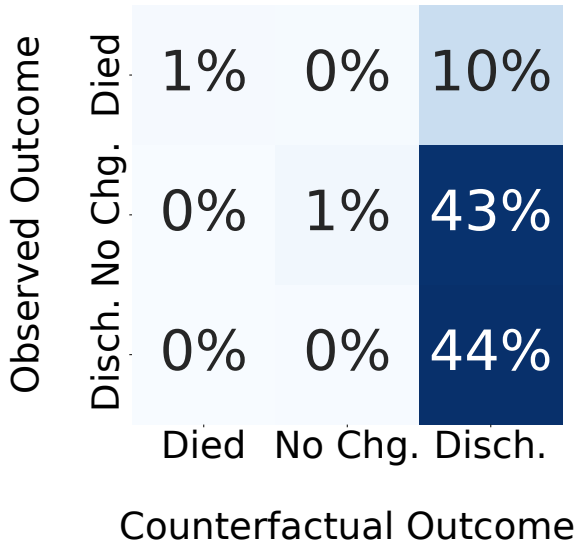


Figure 6-8: *Decomposition of 1000 observed patient trajectories based on observed outcome (Died, no change, and discharged) vs counterfactual outcome under the target policy, using the most common outcome over 5 draws from the counterfactual posterior.*

[8]We only draw 5 counterfactuals per observed trajectory for illustrative purposes here, but note that standard concentration arguments could be used to quantify how many of these independent draws are required to achieve a desired precision on counterfactual quantities of interest, e.g., the probability of death

6.2.4 Insights from Examining Individual Trajectories

Using this decomposition, we can focus on the 10% of observed trajectories where the model concludes that "if the physician had applied the target policy, these patients would have most likely lived".

This is a bold statement, but also one that is plausible for domain experts to investigate (e.g., through chart review of these specific patients), to try and understand the rationale. We illustrate this type of analysis in Figure 6-9, which shows both the observed trajectory and the counterfactual trajectories for a simulated patient.

This example illustrates a dangerous failure mode, where the target policy would have halted treatment despite the glucose vital being dangerously low (e.g., at $t = 5, 7, 8, 11$). Under the learned MDP, the counterfactual optimistically shows a speedy discharge as a result of halting treatment. To understand why, recall that discharge occurs when all four vitals are normal and treatment is stopped. Because diabetes and glucose fluctuations are relatively rare, and because the MDP does not observe either, the model learns that there is a high probability of discharge when the first three vitals are normal, and the action of 'stop all treatments' is applied.

6.2.5 Addendum: Impact of Hidden State

In the experiments given above, we hide the glucose and diabetes state from the model of dynamics used for the RL policy. In this section we explore the impact of that choice on the off-policy evaluations, as well as on the quality of the RL policy.

To demonstrate, in Figure 6-10, we replicate Figure 6-7, but with some important differences. First, instead of using 100 bootstrapped samples of the original 1000 trajectories, we instead repeat the entire process 100 times, with an independent set of trajectories drawn from the simulator in each case. These uncertainty intervals are wider, reflecting the variation which is not captured by bootstrapping alone. Second, we compare the use of a WIS estimator used on the training data (i.e., the original 1000 episodes used to learn the model of dynamics), with a WIS estimator used on a held-out set of 1000 independent episodes. While the example given in the Section 6.2.2 is

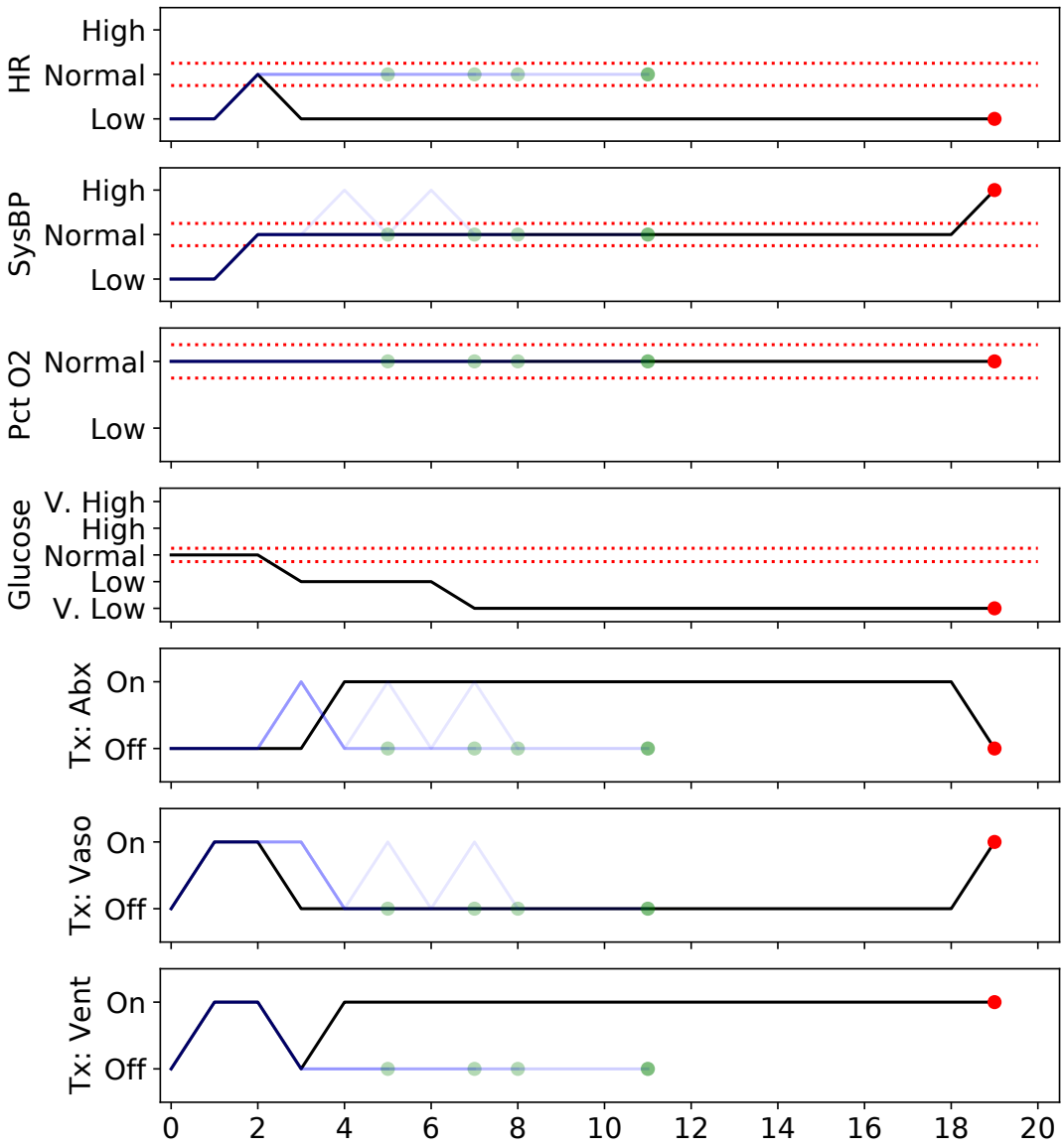


Figure 6-9: *Observed and counterfactual trajectories of a patient. The first four plots show the progression of vital signs, and the last three show the treatment applied. For vital signs, the normal range is indicated by red dotted lines. The black lines show the observed trajectory, which ends in death (signified by the red dot), and the blue lines show five counterfactual trajectories all of which end in discharge, signified by green dots. The glucose vital sign was not included in the model, and hence does not have a counterfactual trajectory. Note how this differs from a newly simulated trajectory of a patient with the same initial state, e.g., all the counterfactual trajectories are identical to the observed trajectory up to a divergence in actions ($t = 2$).*

meant to conceptually capture what might happen in a single analysis (where only a single set of trajectories is available), Figure 6-10 demonstrates the variability across

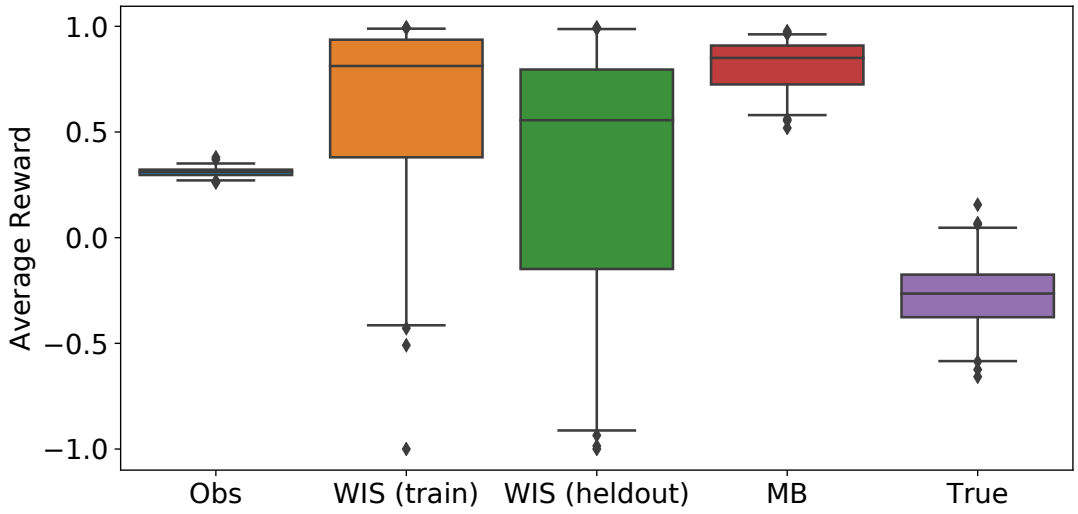


Figure 6-10: *Boxplots show the median and intervals which capture 95% of the 100 evaluations, each time with a newly simulated set of 1000 episodes used for training and 1000 episodes used for the held-out WIS estimator; WIS (train) is used on the training episodes, as in the previous sections, and WIS (held-out) is performed on the held-out set of 1000 episodes*

analyses, including those with access to a large held-out set of trajectories.

Towards understanding the impact of hiding variables from the RL policy, we performed the same experiment again, but giving the RL policy access to the entire state space. The results are shown in Figure 6-11, and the results from both figures are shown in Table 6.1

Table 6.1: *Performance given as Mean (95% CI) from Figures 6-10- 6-11*

	Hidden state	No hidden state
Observed Reward	0.31 (0.27, 0.35)	0.31 (0.27, 0.35)
WIS (train)	0.61 (-0.42, 0.99)	0.58 (-0.23, 0.92)
WIS (heldout)	0.32 (-0.92, 0.99)	-0.04 (-0.94, 0.80)
MB Estimate	0.81 (0.57, 0.96)	0.58 (0.37, 0.73)
True RL Reward	-0.27 (-0.59, 0.05)	-0.19 (-0.41, 0.00)

There are several reasons why weighted importance sampling, and other off-policy evaluation methods, could fail to capture the true performance of a target policy. These include issues like confounding and small sample sizes, as discussed in (Gottesman et al., 2019a). In this particular synthetic example, all of the following factors may play a role in the above results, but it is difficult to say conclusively how strong

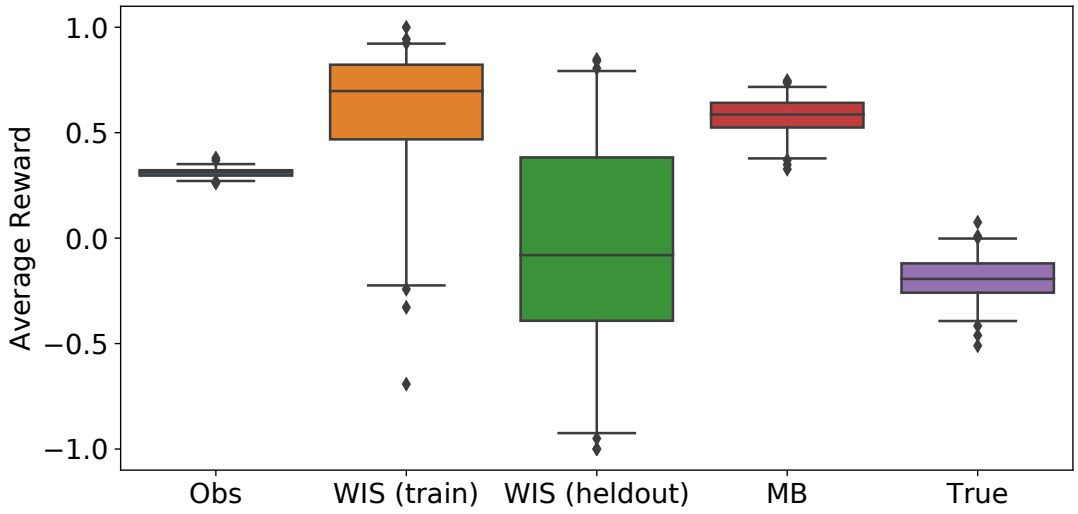


Figure 6-11: *Same setup as Figure 6-10, but allowing the model of dynamics (and the estimated behavior policy) to see the full state*

each factor is, and how they interact to produce the results: (i) Confounding due to unobserved states, (ii) sample complexity of learning the MDP, which is more pronounced when all state information is observed (144 states vs 1440 states), and (iii) small sample sizes in both the training and held-out datasets.

With that in mind, we believe that building a more comprehensive simulated environment, in which these various factors can be disentangled more precisely, would be a valuable direction for future work. In addition, we believe such an environment would be useful for evaluation of a variety of off-policy techniques beyond the limited set discussed in this thesis e.g., more recently developed methods such as Thomas & Brunskill (2016); Liu et al. (2018).

Chapter 7

Real-Data Case Study: Sepsis Management

In this chapter, we replicate the work of Komorowski et al. (2018), which seeks to learn an optimal policy for treating patients with sepsis in the ICU, using model-based RL techniques based on a finite MDP. We then apply our method of counterfactual policy introspection to the resulting policy and model, with the goal of understanding how well our approach works with a real-world example. We recapitulate a high-level overview of their methodology in Section 7.1, while deferring to the original paper for the full details of their setup. Having learned an MDP and corresponding policy following their approach, we perform a similar set of analyses to those we performed in Section 6.2: In Section 7.2 we estimate the reward using WIS on a held-out test set, and in Section 7.3 we decompose the counterfactual reward across trajectories in the test set.

Most notably, we find there are a very small number of patients who the model believes would have died in the counterfactual, and (as such) most of the patients who died in their observed trajectories are projected to have lived under the counterfactual. We select a random trajectory from this latter set for further analysis in Section 7.4, and review it alongside the full medical record, with the assistance of a clinician. In short, we find that it recommends actions which are not appropriate for this patient, based on information available in the clinical notes, and it expects

unrealistic outcomes in the counterfactual as a result of those actions. We discuss this case in more depth in Section 7.4.

Finally, in Section 7.5 we discuss some aspects of the original paper that made this analysis challenging, as well as some broader reflections on the exercise as a test-case for understanding where our approach works well, and where it has limitations.

7.1 Replicating Komorowski et al. (2018)

The authors of (Komorowski et al., 2018) seek to learn a better policy for treating patients in the ICU with sepsis, as discussed previously in this thesis. In this section, we describe their approach at a high level, as well as our methodology for replicating it. We would like to thank Matthieu Komorowski for his assistance in replicating the original paper.

Data Source There are two sources of data used in Komorowski et al. (2018); First, they use data from the MIMIC-III database (Johnson et al., 2016), which contains de-identified medical records from >50k admissions to critical care units at Beth Israel Deaconess Medical Center in Boston, Massachusetts. It also contains out-of-hospital mortality information using the Social Security Administration Death Master File. In their work, MIMIC-III is used for model development, and a separate dataset is used for model testing, the eICU Research Institute Database (eRI). We used the MIMIC-III database for both model development and testing, using a held-out test set of patients for evaluation, in part due to the availability of clinical notes.

Data Processing We used MATLAB code supplied by the authors at `https://github.com/matthieukomorowski/AI_Clinician` to process the raw data into the necessary format, which consists of one row of data for each 4-hour block of a patient's ICU stay, with a maximum of 20 rows per patient. We used slightly modified versions of the scripts, which will be made available at `https://github.com/clinicalml/cf-policy-introspection`. The original scripts are

1. `AIClinician_Data_extract_MIMIC3_140219.ipynb` to extract data from the MIMIC-III database

2. `AIClinician_sepsis3_def_160219.m` to create the sepsis cohort itself

3. `AIClinician_MIMIC3_dataset_160219.m` to construct the data table for downstream analysis

Learning and Evaluation We wrote our own python script to replicate the following procedure for selecting the best policy, using the code provided in `AIClinician_core_160219.m` as a guide when details were not clear from the main paper.

1. Center and scale all of the non-binary variables across the entire dataset, using log transformations where appropriate, and converting binary variables into $[-0.5, 0.5]$. For the two action variables (fluids and vasopressors), discretize into 5 bins, with the first reserved for zero treatment, and the remaining 4 based on quantiles over the entire dataset. Hold out 20% of the MIMIC-III data (by patient ID) as a test set.

2. Repeat 500 times, using a different 80/20 train / validation split on the remaining patient IDs:

   (a) Use K-Means clustering on 25% of the data[1] to assign each 4-hour block to one of 750 states

   (b) Use 90-day survival as the reward signal, with $100, -100$ corresponding to survival and death, respectively. This reward is obtained at the end of a trajectory (or after 20 steps, whichever is lower). Create two new absorbing states to reflect these outcomes.

   (c) Use empirical transition counts (state, action $\rightarrow$ state) to fill in the (three dimensional) transition matrix $P(S'|S, A)$, ignoring any state / action pair with fewer than 5 observations (we will refer to this later as 'truncation'). In the original paper, many of the state / action pairs have no observations,

[1] This was done in the original paper for computational reasons, and we do the same

or fewer than 5 observations, so the transition matrix is not fully defined. We resolved this by treating any observed state / action pair as leading to the 'death' absorbing state, towards the stated goal in Komorowski et al. (2018) of preventing the RL policy from taking any action which is rarely or never seen at a certain state. See Section 7.5 for more discussion on this point. Rewards are defined with respect to the absorbing state, so this suffices to define the MDP.

(d) Learn a policy from this MDP using Policy Iteration, and evaluate using Weighted Importance Sampling (WIS) on the validation set. In the original paper, the physician policy is estimated on the training set using the empirical transition counts, after truncation (described above), and then softened so that all actions have non-zero probability. The approach to softening could cause some probabilities to be negative, so we use a slightly different approach, described in Section 7.5. The RL policy is also softened to an ϵ-greedy policy for the purposes of WIS, where the learned action is taken with 99% probability, and otherwise a random alternative is taken.

(e) Calculate a 95% confidence interval using bootstrapped validation samples, and record the lower bound.

3. Using the k-means clustering, estimated MDP, and the resulting policy which obtained the highest WIS lower bound on the validation set, evaluate on the test set.

7.2 Off-Policy Evaluation with WIS

We give the results of our replication in Figure 7-1 and Tables 7.1 and 7.2. First, we note that there is a large variation in WIS performance on the validation set, with an average estimated reward which is lower than that of the behavior policy. Second, the test WIS results (using the 'best' policy) are highly variable as well, as revealed

through bootstrapping on the 4415 test samples in Table 7.2. This motivates the rest of this section, where we dig further into the counterfactual trajectories to better 'sanity check' this policy.

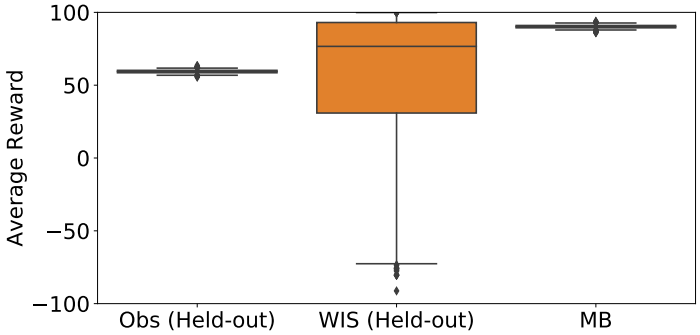


Figure 7-1: *Observed reward of the physician policy (Obs) versus the estimated reward of the learned RL policy using both weighted importance sampling on the validation set (WIS) as well as a model-based (MB) estimate derived from simulating 1000 trajectories, using the learned policy, on the learned MDP. Box-plots show the median and 95% range across 500 iterations. Higher is better.*

Table 7.1: *Results from 500 iterations of the procedure described in Section 7.1. Mean, median, and 95% range calculated over all iterations, and 1000 simulated trajectories were used to derive the model-based result, using the same MDP that was used to learn the policy. Higher is better.*

	Mean	Median	95% range
Observed (Validation)	59.33	59.43	(56.81, 61.85)
WIS (Validation)	53.00	76.64	(-73.00, 99.91)
Model-based	90.22	90.20	(87.85, 92.70)

Table 7.2: *Performance of the chosen policy on the held-out test set of 4415 trajectories, using bootstrapping (750 iterations) to estimate the distribution*

	Mean	2.5%	25%	50%	75%	97.5%
Observed	60.28	57.83	59.46	60.32	61.09	62.82
WIS	60.26	-28.42	47.72	69.42	83.50	96.59

7.3 Decomposition with Counterfactuals

First, we draw 5 counterfactual trajectories (under the chosen policy) for each of the test trajectories, using the techniques described in Chapter 4. In Figure 7-2 we

take the most common outcome across the counterfactual trajectories to assign each individual to one of six categories, based on their factual outcome of 90-day survival and their counterfactual outcome, which can include 'no outcome' (see Section 7.5 for more discussion on this point).

Most notably, we find that *very few patients* have a negative outcome in the counterfactual, and most of the patients who died would have lived. In the next section we investigate this further by selecting a random trajectory from the latter set of patients.

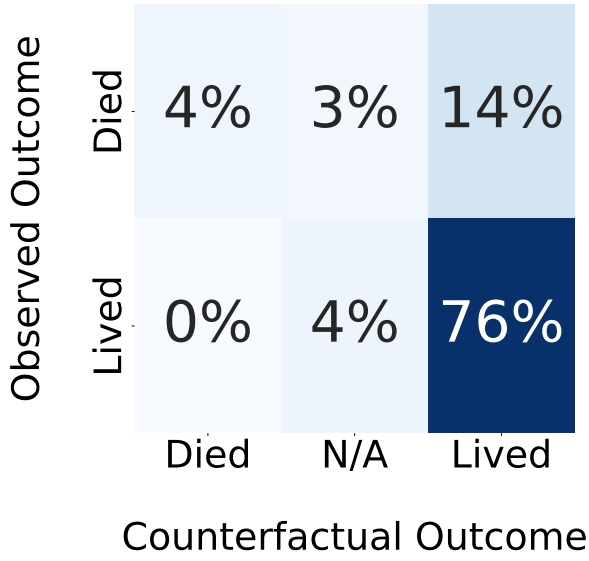


Figure 7-2: *Comparison of outcomes (90-day survival) between the observed and counterfactual trajectories, on the test set. Most notably, under the counterfactual it is estimated that very few patients would have died, and most of the patients who died would have lived. However, 7% of patients have no outcomes in the counterfactuals, due to a nuance discussed in Section 7.5*

7.4 Inspection of Counterfactuals using the Full Medical Record

As stated many times throughout this thesis, one of the main conceptual advantages of using counterfactuals is that they are conceptually easier to 'disprove', and that faults in the counterfactuals are a (heuristic) indication of faults in the learned model of the

environment. In particular, by forcing our model to make counterfactual claims about an actual patient, we can bring additional side-information to bear on scrutinizing the conclusions. To that end, we present an illustrative example in this section, where we review the medical record of a patient alongside their counterfactual trajectories. In particular, we take a randomly selected patient from among those who died but 'would have lived' under all their estimated counterfactual trajectories.

We began by reviewing the clinical notes for this patient (the de-identified notes are available in MIMIC-III) with an infectious disease clinician[2]. A summary of the major takeaways from reviewing those notes:

- *Cause of admission:* This patient was admitted after collapsing, with initial suspicion that this was due to either a respiratory or cardiac failure, and was taken immediately to the cath lab where cardiac causes were ruled out. Chest imaging showed a large amount of fluid around the right lung, and a large mass in the lower right lobe. This was discovered to be State IIIA lung cancer, suggesting the possible etiology of the patients' presentation to be cardiovascular collapse and a post-obstructive pneumonia secondary to compression from the mass.

- *Treatment before and during ICU:* Cardiovascular compromise and inflammation from pneumonia contributed to the build up of a large amount of fluid in the pleural space. Thus, clinicians elected to place a chest tube, which subsequently drained >1L of serous fluid. The patient's clinical status responded rapidly, suggesting the external compression from the fluid was a major contributor to his ICU course. Antibiotics and vasopressors in this setting act as temporizing measures until the definitive intervention of chest tube placement could be performed.

- *Cause of death:* Despite the placement of a chest tube, the underlying problem of a large lung mass leading to cardiovascular compromise remained unad-

[2]Dr. Sanjat Kanjilal, MD, MPH, the Associate Medical Director of Clinical Microbiology at Brigham & Women's Hospital. We thank Dr. Kanjilal for all of his help with this work.

dressed. Given the morbidity of the necessary chemotherapy, it was decided by the providers, the patient and the family that further aggressive intervention would not have been in the patient's interests.

After reviewing the notes, we reviewed the counterfactual trajectories alongside the factual trajectories. We present a condensed output in Figure 7-3, consisting of a few important vital signs, and defer the full output to Figures 7-4-7-7.[3] In particular, we make the following observations

- **No basis (in medical record) for proposed actions**: Recall that the patient was in fluid overload due to congestive heart failure and capillary leakage, which were themselves the result of the adjacent lung mass. The optimal approach in this setting is to carefully reduce the cardiac afterload using diuretics and anti-hypertensives, as well as drainage of the pleural effusion. Thus, while vasopressors and fluids are not grossly counter-indicated, they would have the opposite effect — increasing the work of the heart because they increase cardiac afterload, eventually resulting in worsening of the patients clinical status. Thus, while in the early admission period it is not unreasonable to provide vasopressors and fluids to maintain vital signs, there is a clinical trade-off, and there is no support in the medical record for giving *maximum dose of vasopressors* in the early stages, present in several of the counterfactual trajectories.

- **Consequences of proposed actions are not reflected in CF trajectories**: As noted, the alternative policy gives the maximum dose of vasopressers early on. However, the first 12 hours (first 3 time periods) look almost identical in the

[3] *How to read counterfactual trajectories:* To visualize the counterfactual trajectories, we map the patient state back to the original space of variables. To do so, we used the median of each feature in each cluster (across the entire dataset), though this is not entirely reliable, as can be seen by comparing the black solid lines (the median values for the corresponding state in k-means) with the black dotted lines, which indicate the true values of each variable. This mismatch is discussed further in Section 7.5. To read the trajectories, note that the observed trajectory is given in black, and the counterfactuals are given in light blue, with both derived from the medians (for each feature) of their respective states. Black dotted lines indicate the patient state without using k-means clustering. Red crosses and green dots both indicate the end of the trajectory, as well as the outcome, with green indicating 90-day survival and red indicating a lack thereof. Grey circles indicate no outcome in the counterfactual. Red dotted lines indicate the middle 90% across all patients, in the original data prior to k-means.

counterfactuals to the actual trajectory, and do not reflect the expected effect of additional vasopressors on blood pressure and other vital signs. In particular, maximum dose of vasopressors should have resulted in a significant blood pressure response, which is not evident in these counterfactual trajectories.

- **The anticipated outcomes are not credible given medical record**: Most glaringly, the anticipated outcomes (discharge from the ICU and 90-day survival) are not credible given what we know about the patient from their medical record. For instance, the first counterfactual trajectory ends in 8 hours (with subsequent 90-day survival). That stands in contrast to what we know from the medical record, that the death of this patient was due to irreversible lung damage caused by Stage IIIA lung cancer and pneumonia, neither of which would have been resolved by this treatment.

Our review suggests an important possible limitation of the underlying learned MDP and policy. Important features (such as the underlying infection and lung cancer in this case) are not included in the model, but could reasonably impact both the outcome of the patient as well as the treatment decisions of clinicians. This issue also arises in a second trajectory that we randomly sampled (not shown here), in which the clinical notes indicated that the patient died from complications due to pre-existing Hodgkin Lymphoma and treatment in the ER (prior to admission to the ICU) which triggered respiratory failure and irreversible lung injury. The counterfactuals all indicated 90-day survival, contradicting the clinical notes which suggest that by the time the patient entered the ICU, nothing more could be done.

In conclusion, if we are to fully trust a model of dynamics, and the policy that is derived from it, then we would like to see a series of counterfactuals that 'make sense' to a clinician, as a type of explanation and justification for why the RL policy might have performed better than existing practice. As always, it is possible that the structural causal model itself is incorrect in this case, but we present this method as a useful (and simple) heuristic to apply, for generating hypotheses which could be useful for iterating on the model and resulting policy.

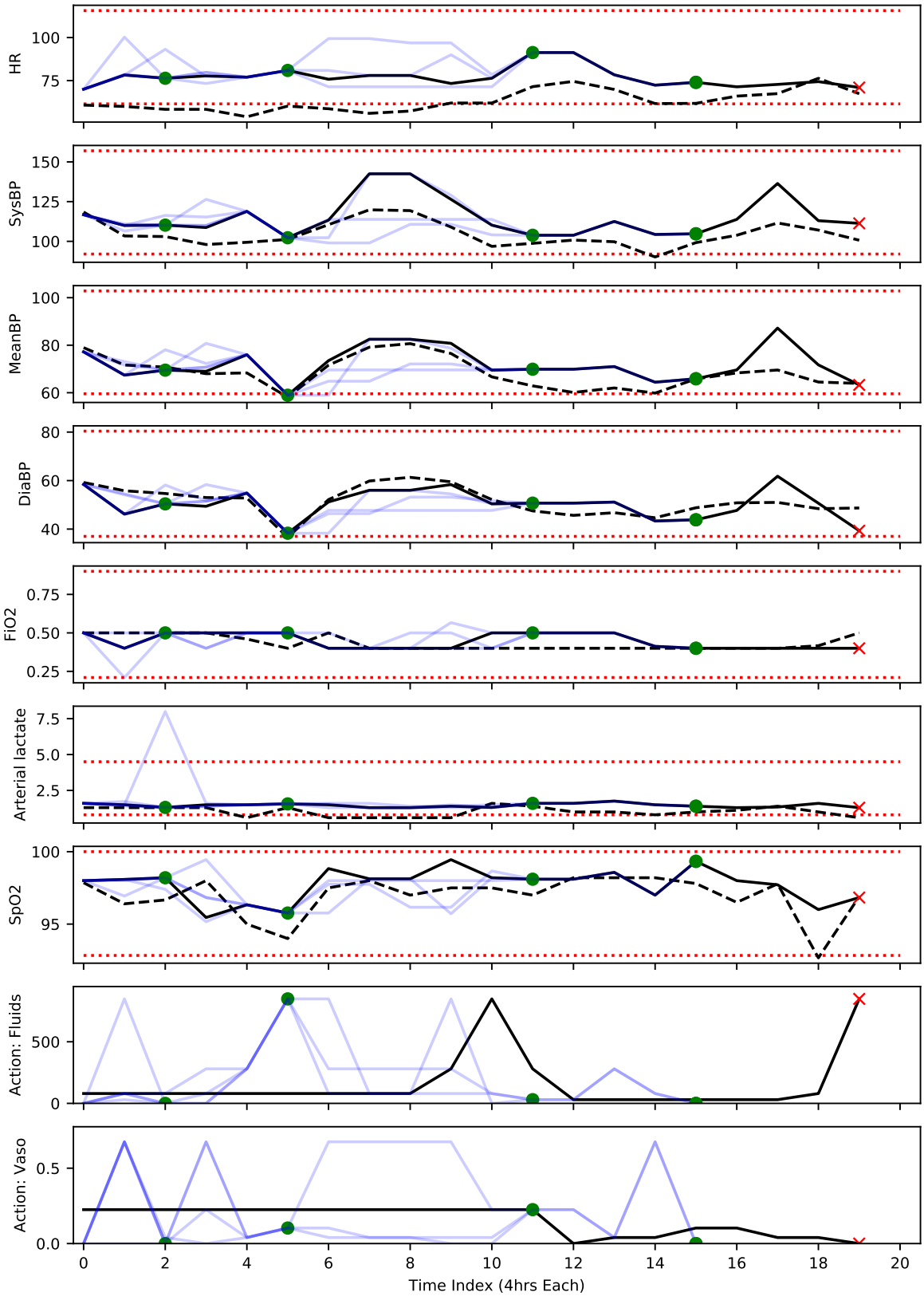


Figure 7-3: *Five counterfactual trajectories (for selected features), two of which end at $t = 15$. See description in the main text for how to read counterfactuals. HR: heart rate. BP: blood pressure. FiO2: fraction of inspired oxygen. SpO2: Peripheral oxygen saturation.*

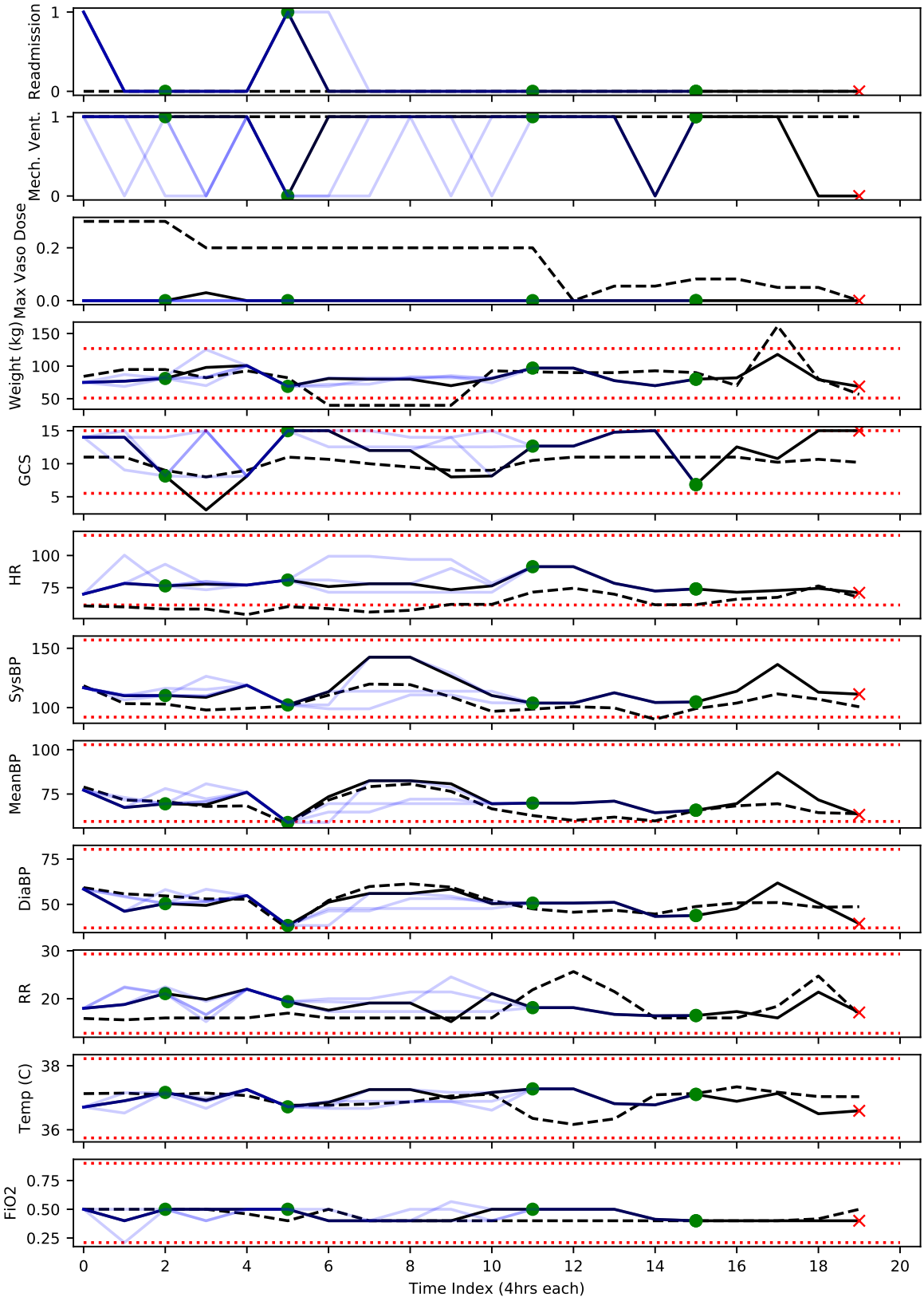


Figure 7-4: *Example Trajectory including all features (Part 1/4). See description in the main text.*

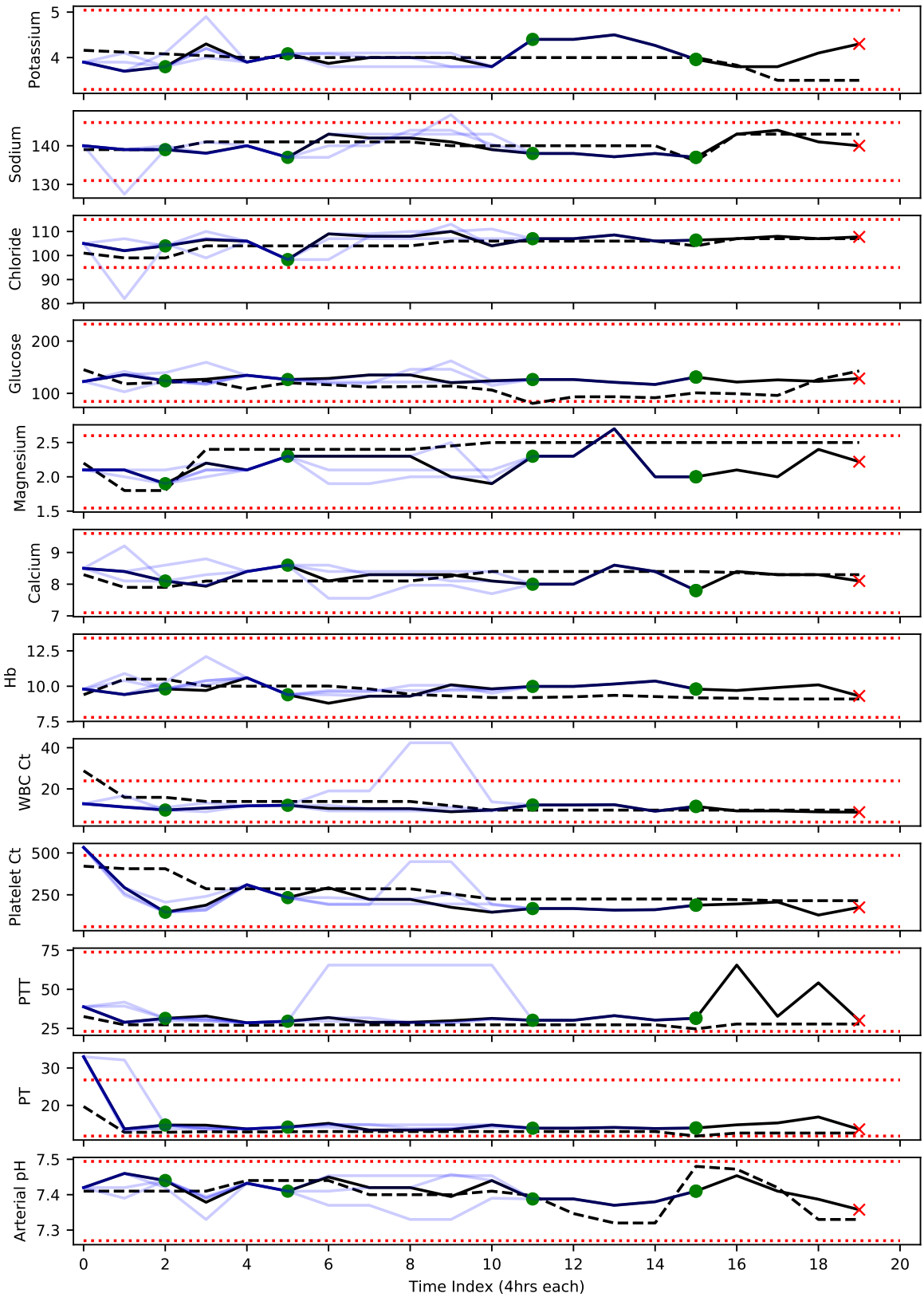


Figure 7-5: *Example Trajectory including all features (Part 2/4). See description in the main text.*

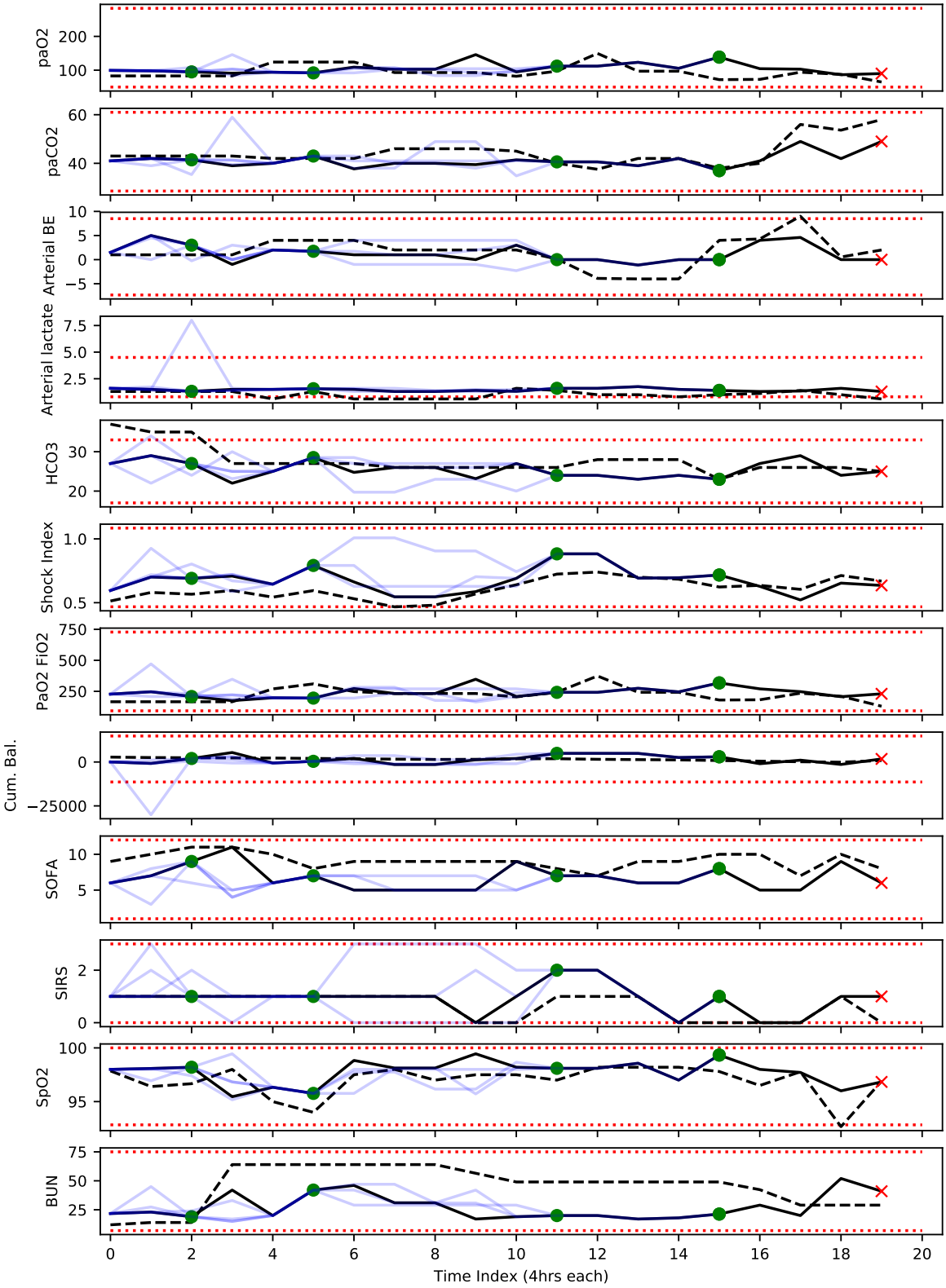


Figure 7-6: *Example Trajectory including all features (Part 3/4). See description in the main text.*

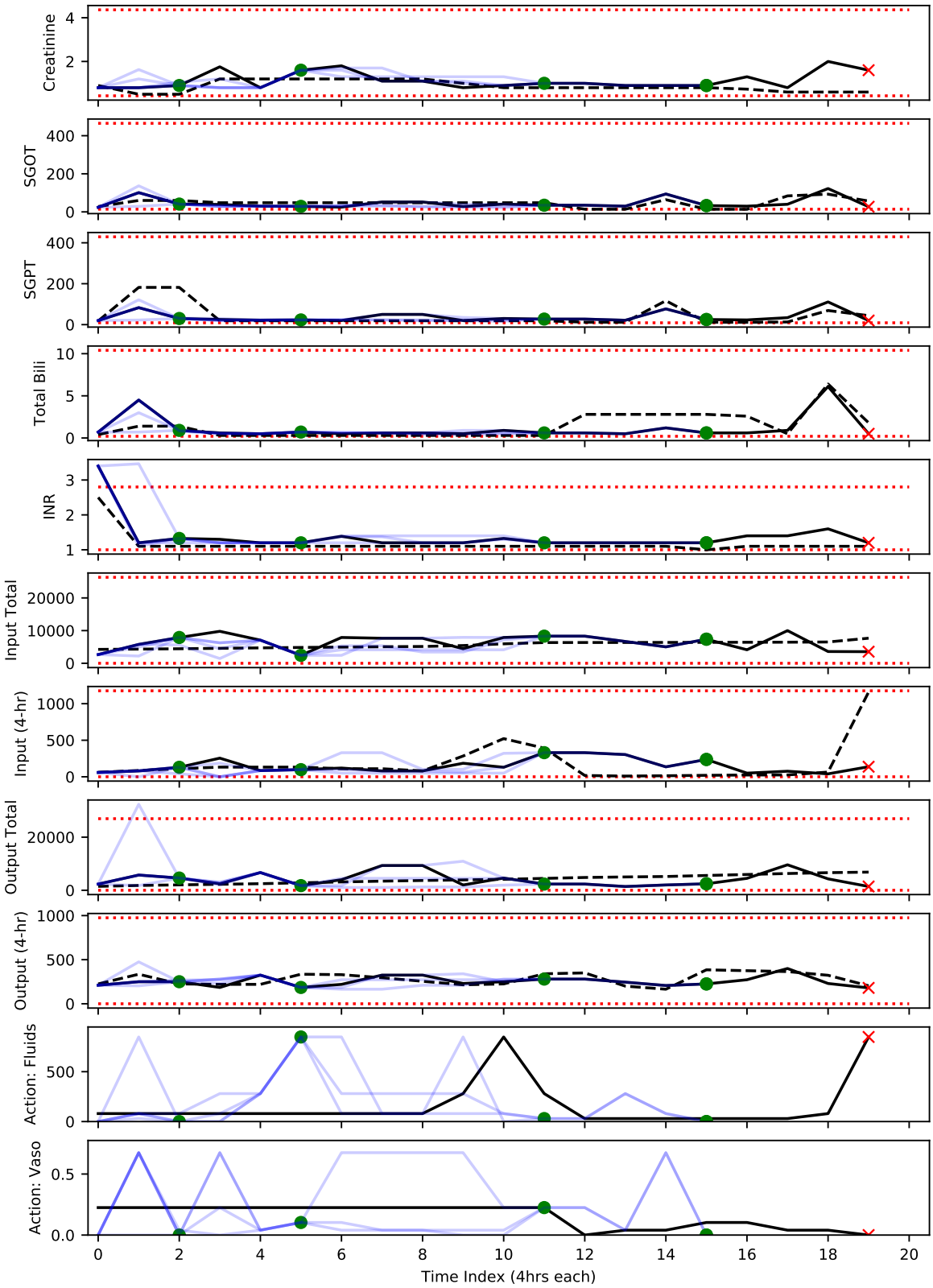


Figure 7-7: *Example Trajectory including all features (Part 4/4). See description in the main text.*

7.5 Challenges and Lessons Learned

There were a number of challenges in applying our methodology as imagined, some of which are due to idiosyncrasies with the approach in Komorowski et al. (2018).

Specification of outcome The outcome used in the original paper was 90-day mortality after discharge from the hospital, which was treated as an absorbing state. Moreover, for each patient, a maximum of 20 time-steps (of 4 hours each) were allowed, with the outcome always coming at the end of an observed trajectory. Thus, it has the implicit interpretation of 'discharge followed by [survival / death] after 90 days'. However, there is no guarantee that any model-based trajectory (including the counterfactuals) will end within 20 steps, leading to some instances where the counterfactual ends without an observed outcome.

Specification of states First, there are some idiosyncrasies with how state variables were encoded in the original paper. For instance, every variable is included in the k-means clustering, including those which should not fluctuate over the course of an ICU stay (such as gender and age). Second, we observed that our approach to visualization, of using the median value of each feature for each state, has some limitations. In particular, perhaps due to not having a large enough set of discrete states, when we 'impute' the factual trajectory based on the discrete states and compare it to the actual trajectory for those features, they are not always comparable. See Figure 7-8 for an example of this, taken from the same patient as above. This suggests that for our method to be most useful, the MDP should either operate in the original state space or operate in an invertible representation of it.

Estimation of behavior policy Because the behavior policy is derived using empirical counts, and because rare actions are truncated, it leads to an estimated zero-probability of several (observed) state/action pairs (including in the training set). This makes WIS impossible to use, because it relies on each observed action having non-zero probability. The solution to this taken in the original paper was to subtract

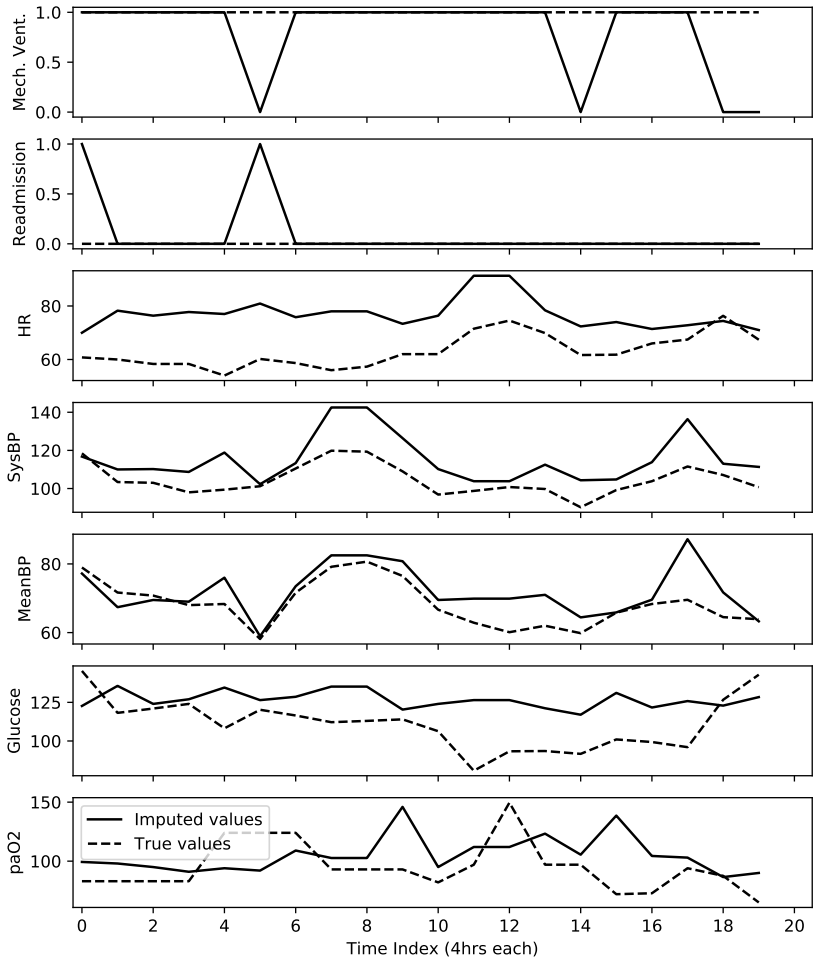


Figure 7-8: *Comparison of the imputed values (by taking the median of each feature for each cluster) and the actual values for the same patient.*

a small amount from every action that has a non-zero probability, and add it to the other actions evenly. The way it was implemented in the supplied MATLAB code, this could cause some actions to have negative probability, because the amount subtracted was equal across observed actions. We resolved this in two ways: First, we did not implement truncation for the purposes of learning the behavior policy. Second, we softened the policy by instead adding a pseudo-observation of 0.01 to every action/state pair which was never seen, in the empirical counts.

Empirical MDP In the original paper, empirical counts are used to estimate the MDP, but does not result in a valid set of conditional distributions, because some state/action pairs are never observed. This is critical for our approach, because

we need the observed trajectories to have non-zero probability under the MDP to calculate a counterfactual. Here we chronicle our efforts to resolve this, as well as explaining our final resolution:

1. As an initial attempt to resolve this, we first introduced the notion that you instantly die if you take an action that had never been taken, and used this to learn the policy (so that it avoids those actions). This approach forced us to re-learn the MDP on the test data for evaluation purposes, so avoid zero-probability trajectories.

2. This proved to be an inadequate solution for running on test data, because it results in a skewed model-based (and thus, counterfactual) estimate of reward; While the policy takes actions that were observed in the training data, they may not be observed in the test data, and by construction of our test MDP led to instant death.

3. Thus, we settled on using a softened MDP for the counterfactual evaluations, based only the training data, where we added a pseudo-observation of 10^{-3} for each transition, did not truncate observations, and did not use the 'instant death' rule. We confirmed (see Table 7.3) that this did not meaningfully impact the model-based estimate of reward under the RL policy, so we took it as a good proxy for the original MDP used to learn the policy.

Table 7.3: *Comparison of MDPs; 1000 model-based trajectories were averaged, and this was done 10 times to give 90% confidence intervals*

Approach	Average Reward	90% interval
Train	89.68	(88.08, 91.11)
Train (Soft)	85.32	(83.74, 86.52)

Chapter 8

Conclusion

Given the desire to deploy RL policies in high-risk settings (e.g., healthcare), it is important to develop more tools and techniques to introspect these models and the policies they learn. In this thesis, we have presented a general method for doing so, which we call *counterfactual policy introspection*. Our approaches requires two inputs: A policy to be inspected, and a model of the relevant decision-making problem. This model could be a MDP or POMDP, or it could be any other learned graphical model of the environment which can be represented as a directed acyclic graph. By making general assumptions regarding the structure of causal mechanisms, we convert such a model into a structural causal model which can be used to generate counterfactuals. These counterfactuals serve several purposes:

1. First, they can be used to get a sense for which patients are driving the overall model-based reward. Theoretically, if the SCM is well specified, the expected counterfactual reward will be equivalent to the model-based reward. Anecdotally, in both our real-data and synthetic experiments (where the model was presumably not well-specified), we also found this to hold approximately.

2. Second, they can be used to highlight particularly interesting trajectories for further manual inspection. In our experiments, we give the example of highlighting patients who the model believes would have lived under the counterfactual, despite dying in the real world.

3. Finally, they serve to provide a detailed 'rationale' for the estimated performance of the policy, in terms of an expected counterfactual trajectory. These trajectories seek to isolate the differences in intermediate and final outcomes that are due to difference in actions, and can be reviewed along with side information (e.g., chart review in the medical setting) to identify flaws in the conclusions, which may suggest flaws in the original model.

However, this approach does not come without its limitations. It requires knowing, or making an untestable assumption about, the structural causal model: Here we propose the Gumbel-Max SCM, which is an example of an SCM that may be realistic in some settings. As revealed through our real-data experiment, our approach may also work best when the environment is modelled directly in the original state space, and the model of dynamics is not too brittle to handle unseen trajectories that may arise in test data. Nonetheless, our real-data experiments give us hope that this might be useful to researchers in the future, as a relatively straightforward method to debug models and generate hypotheses for improving them.

Bibliography

Aguirregabiria, V. and Mira, P. Dynamic discrete choice structural models: A survey. *Journal of Econometrics*, 156(1):38–67, 2010. ISSN 03044076. doi: 10.1016/j.jeconom.2009.09.007. URL http://dx.doi.org/10.1016/j.jeconom.2009.09.007.

Bal, B. S. An introduction to medical malpractice in the United States. *Clinical Orthopaedics and Related Research*, 467(2):339–347, 2009. ISSN 0009921X. doi: 10.1007/s11999-008-0636-2.

Balke, A. and Pearl, J. Counterfactual Probabilities: Computational Methods, Bounds and Applications. *Proceedings of the Tenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-94)*, pp. 46–54, 1994. doi: 10.2202/1557-4679.1322.

Bang, H. and Robins, J. M. Doubly Robust Estimation in Missing Data and Causal Inference Models. *Biometrics*, 61:962–972, 2005.

Bibaut, A., Malenica, I., Vlassis, N., and Van Der Laan, M. More Efficient Off-Policy Evaluation through Regularized Targeted Learning. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 654–663, Long Beach, California, USA, 2019. PMLR.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*, 2018.

Buesing, L., Weber, T., Zwols, Y., Heess, N., Racaniere, S., Guez, A., and Lespiau, J.-B. Woulda, Coulda, Shoulda: Counterfactually-Guided Policy Search. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJG0voC9YQ.

Chow, Y., Petrik, M., and Ghavamzadeh, M. Robust Policy Optimization with Baseline Guarantees. *arXiv preprint*, pp. 1–25, 2015. URL http://arxiv.org/abs/1506.04514.

Chua, K., Calandra, R., McAllister, R., and Levine, S. Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models. In *32nd Conference on*

Neural Information Processing Systems (NeurIPS), Montreal, Canada, 2018. ISBN 1471-2458. doi: arXiv:1805.12114v1.

Cuellar, M. and Kennedy, E. H. A nonparametric projection-based estimator for the probability of causation, with application to water sanitation in Kenya. *Journal of the Royal Statistical Society: Series A (Special Issue on Causal Inference)*, pp. 1–24, 2018.

Dawid, P., Faigman, D. L., and Fienberg, S. E. On the Causes of Effects: Response to Pearl. *Sociological Methods and Research*, 44(1):165–174, 2015. ISSN 15528294. doi: 10.1177/0049124114562613.

Dawid, P., Musio, M., and Fienberg, S. E. From statistical evidence to evidence of causality. *Bayesian Analysis*, 11(3):725–752, 2016. ISSN 19316690. doi: 10.1214/15-BA968.

Encyclopedia, W. *West's Encyclopedia of American Law*. The Gale Group, 2nd edition, 2008. URL `https://legal-dictionary.thefreedictionary.com/proximate+cause`.

Farajtabar, M., Chow, Y., and Ghavamzadeh, M. More Robust Doubly Robust Off-policy Evaluation. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1447–1456, Stockholmsmässan, Stockholm Sweden, 2018. PMLR. URL `http://proceedings.mlr.press/v80/farajtabar18a.html`.

Gottesman, O., Johansson, F., Komorowski, M., Faisal, A., Sontag, D., Doshi-Velez, F., and Celi, L. A. Guidelines for reinforcement learning in healthcare. *Nature Medicine*, 25(1):16–18, 2019a. ISSN 1078-8956. doi: 10.1038/s41591-018-0310-5. URL `http://www.nature.com/articles/s41591-018-0310-5`.

Gottesman, O., Liu, Y., Sussex, S., Brunskill, E., and Doshi-Velez, F. Combining Parametric and Nonparametric Models for Off-Policy Evaluation. In *Proceedings of the 36th International Conference on Machine Learning*, Long Beach, California, 2019b. URL `http://arxiv.org/abs/1905.05787`.

Guez, A., Vincent, R. D., Avoli, M., and Pineau, J. Adaptive Treatment of Epilepsy via Batch-mode Reinforcement Learning. *AAAI*, 2008. URL `http://www.cs.mcgill.ca/~jpineau/files/guez-iaai08.pdf`.

Hanna, J. P., Stone, P., and Niekum, S. Bootstrapping with models: Confidence intervals for off-policy evaluation. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 1(May):538–546, 2017. ISSN 15582914.

Hazan, T. and Jaakkola, T. On the partition function and random maximum a-posteriori perturbations. In *ICML*, 2012.

Hazan, T., Papandreou, G., and Tarlow, D. *Perturbation, Optimization, and Statistics*. MIT Press, 2016.

Hernan, M. and Robbins, J. *Causal Inference*. Chapman & Hall/CRC, forthcoming, Boca Raton, 2019.

Imbens, G. W. and Angrist, J. D. Identification and Estimation of Local Average Treatment Effects. *Econometrica*, 62(2):467–475, 1994. ISSN 00129682, 14680262. doi: 10.2307/2951620. URL http://www.jstor.org/stable/2951620.

Imbens, G. W. and Rubin, D. B. *Causal Inference for Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015. ISBN 978-0-521-88588-1.

Jeter, R., Josef, C., Shashikumar, S., and Nemati, S. Does the Artificial Intelligence Clinician learn optimal treatment strategies for sepsis in intensive care? *arXiv preprint*, 2019. URL http://arxiv.org/abs/1902.03271.

Jiang, N. and Li, L. Doubly robust off-policy value evaluation for reinforcement learning. *33rd International Conference on Machine Learning, ICML 2016*, 2:1022–1035, 2016.

Johansson, F. D., Shalit, U., and Sontag, D. Learning Representations for Counterfactual Inference. In *ICML*, 2016. URL http://arxiv.org/abs/1605.03661.

Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L.-w. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., and Mark, R. G. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:160035, may 2016. ISSN 2052-4463. doi: 10.1038/sdata.2016.35. URL http://www.nature.com/articles/sdata201635.

Kallus, N. Classifying treatment responders under causal effect monotonicity. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3201–3210, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Kallus, N. and Zhou, A. Confounding-Robust Policy Improvement. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 9269–9279. Curran Associates, Inc., 2018.

Komorowski, M., Celi, L. A., Badawi, O., Gordon, A. C., and Faisal, A. A. The Artificial Intelligence Clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine*, 24(11):1716–1720, 2018. ISSN 1546170X. doi: 10.1038/s41591-018-0213-5. URL http://dx.doi.org/10.1038/s41591-018-0213-5.

Liu, Y., Gottesman, O., Raghu, A., Komorowski, M., Faisal, A. A., Doshi-Velez, F., and Brunskill, E. Representation Balancing MDPs for Off-policy Policy Evaluation. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and

Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 2644–2653. Curran Associates, Inc., 2018.

Lodi, S., Sharma, S., Lundgren, J. D., Phillips, A. N., Cole, S. R., Logan, R., Agan, B. K., Babiker, A., Klinker, H., Chu, H., Law, M., Neaton, J. D., and Hernán, M. A. The per-protocol effect of immediate versus deferred antiretroviral therapy initiation. *AIDS*, 30(17), 2016. ISSN 0269-9370.

Luce, R. D. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, New York, 1959.

Maddison, C. J. and Tarlow, D. Gumbel Machinery, 2017. URL `https://cmaddis.github.io/gumbel-machinery`.

Maddison, C. J., Tarlow, D., and Minka, T. A* Sampling. *Advances in Neural Information Processing Systems*, 2014. ISSN 02767783. doi: 10.1016/B978-0-12-803459-0.00005-4. URL `http://arxiv.org/abs/1411.0030`.

Maddison, C. J., Mnih, A., and Teh, Y. W. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables. In *International Conference on Learning Representations (ICLR)*, 2017. ISBN 0780365402. URL `http://arxiv.org/abs/1611.00712`.

Miettinen, O. S. Proportion of disease caused or prevented by a given exposure, trait or intervention. *American Journal of Epidemiology*, 99(5):325–332, 1974. ISSN 00029262. doi: 10.1093/oxfordjournals.aje.a121617.

Morgan, S. L. and Winship, C. *Counterfactuals and Causal Inference: Methods and Principles for Social Research*. Cambridge University Press, Cambridge, 2 edition, 2014. ISBN 9781107065079. doi: DOI:10.1017/CBO9781107587991.

Oberst, M. and Sontag, D. Counterfactual Off-Policy Evaluation with Gumbel-Max Structural Causal Models. *Proceedings of the 36th International Conference on Machine Learning*, 97, 2019. URL `http://arxiv.org/abs/1905.05824`.

Parbhoo, S., Bogojeska, J., Zazzi, M., Roth, V., and Doshi-Velez, F. Combining Kernel and Model Based Learning for HIV Therapy Selection. *AMIA Joint Summits on Translational Science Proc*, 2017:239–248, 2017. ISSN 2153-4063. URL `http://www.ncbi.nlm.nih.gov/pubmed/28815137%0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5543338`.

Pearl, J. Probabilities of causation: three counterfactual interpretations and their identification. *Synthese*, 121(1):93–149, 2000. ISSN 00397857. doi: 10.1023/A:1005233831499.

Pearl, J. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009. ISBN 052189560X. URL `https://dl.acm.org/citation.cfm?id=1642718`.

Peng, X., Ding, Y., Wihl, D., Gottesman, O., Komorowski, M., Lehman, L.-W. H., Ross, A., Faisal, A., and Doshi-Velez, F. Improving Sepsis Treatment Strategies by Combining Deep and Kernel-Based Reinforcement Learning. *AMIA Annual Symposium*, pp. 887–896, 2018. ISSN 1942-597X (Electronic).

Peters, J., Janzing, D., and Schölkopf, B. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, Cambridge, MA, 2017. ISBN 9780262037310.

Precup, D., Sutton, R. S., and Singh, S. P. Eligibility Traces for Off-Policy Policy Evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML)*, pp. 759–766, San Francisco, CA, USA, 2000. ISBN 1-55860-707-2.

Raghu, A., Komorowski, M., Celi, L. A., Szolovits, P., and Ghassemi, M. Continuous State-Space Models for Optimal Sepsis Treatment: a Deep Reinforcement Learning Approach. In *Machine Learning for Healthcare*, 2017.

Raghu, A., Komorowski, M., and Singh, S. Model-Based Reinforcement Learning for Sepsis Treatment. *Machine Learning for Health (ML4H) Workshop at NeurIPS 2018*, 2018. URL `http://arxiv.org/abs/1811.09602`.

Robins, J. M. A New Approach to Causal Inference In Mortality Studies with a Sustained Exposure Period - Application to Control of the Healthy Worker Survivor Effect. *Mathematical Modelling*, 7(9-12):1393–1512, 1986. ISSN 02700255. doi: 10.1016/0270-0255(86)90088-6.

Rosenbaum, P. R. and Rubin, D. B. The Central Role of the Propensity Score in Observational Studies for Causal Effects. *Biometrika*, 70(1):41–55, 1983.

Rubin, D. B. and Rosenbaum, P. R. Reducing Bias in Observational Studies Using Subclassification on the Propensity Score. *Journal of American Statistical Association*, 79:516–524, 1984.

Rubinstein, R. Y. *Simulation and the Monte Carlo Method*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1981. ISBN 0471089176.

Schulam, P. and Saria, S. Reliable Decision Support using Counterfactual Models. In *31st Conference on Neural Information Processing Systems*, 2017.

Shalit, U., Johansson, F. D., and Sontag, D. Estimating individual treatment effect: generalization bounds and algorithms. In *33rd International Conference on Machine Learning (ICML)*, jun 2016. URL `http://arxiv.org/abs/1606.03976`.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., and Hassabis, D. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018. ISSN 0036-8075. doi: 10.1126/science.aar6404. URL `http://science.sciencemag.org/content/362/6419/1140`.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 2017.

Thomas, P. S. and Brunskill, E. Data-Efficient Off-Policy Policy Evaluation for Reinforcement Learning. In *33rd International Confernece on Machine Learning (ICML)*, volume 48, 2016.

Tian, J. and Pearl, J. Probabilities of causation : Bounds and identification. *Annals of Mathematics and Artificial Intelligence*, 28(1-4):287–313, 2000. ISSN 1012-2443. doi: 10.1023/A:1018912507879.

Train, K. *Discrete choice methods with simulation*. Cambridge University Press, 2002.

Yamada, K. and Kuroki, M. Counterfactual-Based Prevented and Preventable Proportions. *Journal of Causal Inference*, 5(2), 2017. ISSN 2193-3677. doi: 10.1515/jci-2016-0020.

Yellott, J. I. The relationship between Luce's choice axiom, Thurstone's theory of comparative judgment, and the double exponential distribution. *Journal of Mathematical Psychology*, 15(2):109–144, 1977.

Yuille., G. P. and L, A. Perturb-and-map random fields: Using discrete optimization to learn and sample from energy models. In *ICCV*, 2011.

Zhang, Y., Young, J. G., Thamer, M., and Hernán, M. A. Comparing the Effectiveness of Dynamic Treatment Strategies Using Electronic Health Records: An Application of the Parametric g-Formula to Anemia Management Strategies. *Health services research*, 53(3):1900–1918, jun 2018. ISSN 1475-6773. doi: 10.1111/1475-6773.12718. URL `https://www.ncbi.nlm.nih.gov/pubmed/28560811`.