TIME AFTER TIME: DEEP-Q EFFECT ESTIMATION FOR INTERVENTIONS ON WHEN AND WHAT TO DO

Yoav Wald[1] **Mark Goldstein**[1] **Yonathan Efroni**[2] **Wouter A.C. van Amsterdam**[3]
yoav.wald@nyu.edu

Rajesh Ranganath[1]

ABSTRACT

Problems in fields such as healthcare, robotics, and finance requires reasoning about the value both of what decision or action to take and when to take it. The prevailing hope is that artificial intelligence will support such decisions by estimating the causal effect of policies such as how to treat patients or how to allocate resources over time. However, existing methods for estimating the effect of a policy struggle with *irregular time*. They either discretize time, or disregard the effect of timing policies. We present a new deep-Q algorithm that estimates the effect of both when and what to do called Earliest Disagreement Q-Evaluation (EDQ). EDQ makes use of recursion for the Q-function that is compatible with flexible sequence models, such as transformers. EDQ provides accurate estimates under standard assumptions. We validate the approach through experiments on survival time and tumor growth tasks.

1 INTRODUCTION

Sequential decision-making is common in healthcare, finance, and beyond (Chen et al., 2021a; Upadhyay et al., 2018). In hospitals, medical professionals administer treatments at different times based on the evolving observations of a patient's condition; in financial markets, traders execute orders based on sequential information flows. Algorithmic decision support systems can optimize these processes by evaluating different *policies* with respect to their expected outcomes. Estimating the difference in expected outcomes between various policies is a causal effect estimation question (Chen et al., 2021a; Joshi et al., 2025; van Amsterdam et al., 2024). This question involves several future treatment decisions taken at varying time points, hence it is a sequential decision-making problem.

Formally, this problem falls within the framework of off-policy evaluation (Fu et al., 2021; Uehara et al., 2022). A defining feature is that timings of observations and treatments are irregularly spaced, represented by a stochastic point process with intensity λ, whereas the *type* of the treatments at those times are specified as the marks of this process, governed by a distribution π. Unlike traditional formulations of off-policy evaluation that focus on action types, here the times must be accounted for, as they have a large effect on the outcome. This formulation is relevant to many decision-making scenarios, for example, in transplants, where it is often desirable to delay treatment as much as possible to lower the risk of complications. However, delaying too much can result in a deterioration in the patient's condition. Here, the type of treatment is fixed and what matters is the timing.

Estimating the effect of intervention on treatment timing is a crucial part of evaluating sequential policies. With irregular times, frameworks for sequential decision-making that discretize time can be problematic, as discretization can be inaccurate or inefficient and requires choosing appropriate time scales. Further, length scales for decision-making within a single trajectory can vary dramatically. Consider for example a patient with heart failure; such a patient may be stable for months or years with occasional treatment adjustments made during visits to the cardiologist. However, at some point, they may experience acute decompensation (Felker et al., 2011), which requires rapid treatment and

[1] New York University
[2] Meta
[3] University Medical Center Utrecht

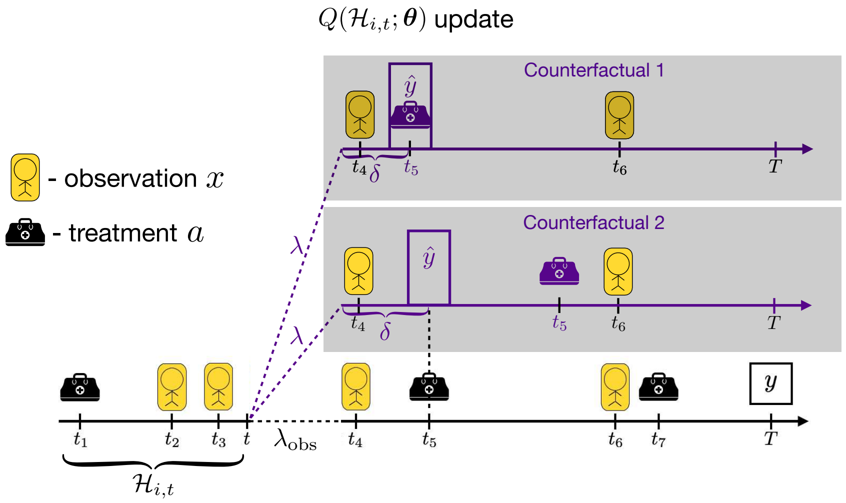


Figure 1: A summary of EDQ. Conditional expectations of the outcome, $\mathbb{E}_P[Y|\mathcal{H}_t]$, are estimated by $Q_t(\mathcal{H}_t;\boldsymbol{\theta})$. Given an observed trajectory $\mathcal{H}_i$ sampled under the training policy λ_{obs} (bottom trajectory), we fit values $Q_t(\mathcal{H}_{i,t};\boldsymbol{\theta})$ by regressing them on a label $\hat{y} := Q_{t+\delta}(\widetilde{\mathcal{H}}_{i,t+\delta})$ determined by a "counterfactual" trajectory $\widetilde{\mathcal{H}}$ sampled from the target policy λ. δ is the earliest disagreement time between the observed and counterfactual trajectories. It is either the time of the next observed treatment (counterfactual 1) or that of the treatment under the target policy (counterfactual 2).

intensive monitoring during hospitalization. Existing methods for continuous-time causal inference do not scale gracefully, since they solve complex estimation problems, such as integrating importance weights across time (Røysland, 2011). Those methods that scale to high-capacity models and large datasets do not handle dynamic policies (i.e., policies that take past states into account) and are implemented with differential equation solvers (Seedat et al., 2022), restricting architectural choices.

In this work, we give two methods for off-policy evaluation with irregularly sampled data. Our contributions are as follows:

- We define off-policy evaluation with *decision point processes* and develop Earliest Disagreement Q-Evaluation (EDQ), a model-free solution to the problem. While other methods are intractable in high dimensions or are limited to static treatments, EDQ eliminates these restrictions. EDQ is based on direct regressions and dynamic programming, which makes it easily applicable to flexible architectures including sequence models such as transformers.
- In Theorem 1, we prove that EDQ is an empirical estimator of the correct policy value. The estimator produces accurate causal effects under assumptions on causal validity based on Røysland (2011); Røysland et al. (2022).
- We validate the efficacy of EDQ through an experimental demonstration on time-to-failure prediction and tumor growth simulation tasks. For these tasks, we implement a transformer-based solution. The results show EDQ's advantage relative to baselines that rely on discretization.

We define the estimation problem, develop a solution, discuss related work and validate empirically.

2 Off-Policy Evaluation with Decision Point Processes

Consider a decision process defined by a marked point process P (Andersen et al., 2012; Snyder and Miller, 2012) over observations (which take values in $\mathcal{X}$), treatments (in $\mathcal{A}$ respectively) and real outcomes. We are interested in estimating an overall quantity $Y \in \mathbb{R}$ that is a function of a sequence $Y_1, Y_2 \ldots$ of observed rewards. For convenience, we let $Y = \sum_{k=1}^{\infty} Y_k$, where k is an index for observed outcomes along the trajectory.[1] Though, the methods extend to other outcome functions like discounted future outcomes. We assume the number of rewards in the segment $[0, T]$ is finite.

[1] We define $Y_k(\omega) = 0$ for an event ω where k is larger than the number of outcomes in the trajectory.

Marked point processes. A marked point process is a distribution over event times, along with distributions over *marks*, or details of the events at each time (i.e. treatment times and which treatment was given). We consider multivariate counting processes $N(t) = (N^a(t), N^b(t), \ldots)$ on the time interval $[0, T]$. For a univariate process, e.g. N^b, $N^b(t)$ is the number of events of type b until time t. A trajectory of event times and their marks is a set $\mathcal{H} = \{(t_0, \mathbf{e}_0), (t_1, \mathbf{e}_1), \ldots, (t_n, \mathbf{e}_n)\}$. We denote events up to time t by $\mathcal{H}_t = \{(t_k, \mathbf{e}_k) \in \mathcal{H} : t_k \leq t\}$ and $\mathcal{H}_{(t,t+\delta]}$ for events in the interval $(t, t+\delta]$. We use $\mathcal{H}^b$ to refer to events of type b on the trajectory and $\mathcal{H}^{\setminus b}$ for events of all types other than b.

Finally, we assume intensity functions $\lambda(t|\mathcal{H}_t) = \mathbb{E}[dN(t)|\mathcal{H}_t]$ for processes exist, $N(t)$ is almost surely finite for any $t \in [0, T]$, and that the process can depend on its own history. That is, the filtration is the σ-algebra generated by random variables $N(t)$ and their marks (Aalen et al., 2008).

2.1 Problem Definition

We follow notation from the RL literature (Upadhyay et al., 2018). We begin with the data generating process and then summarize our goal of inferring causal effects. This involves off-policy evaluation under a distribution P, while observing samples from P_{obs}.

Definition 1. *A marked decision point process P is a marked point process with observed components N^e for $e \in \{x, y, a\}$ that have corresponding intensity functions λ^e, and mark spaces $\mathcal{X}, \mathbb{R}, \mathcal{A}$, and a multivariate unobserved process with intensity λ^u. By default, we omit unobserved events from the trajectories $\mathcal{H}$, hence $\mathcal{H} = \{(t_0, \mathbf{e}_0), (t_1, \mathbf{e}_1), \ldots, (t_n, \mathbf{e}_n)\}$ where $\mathbf{e}_k \in \{\mathcal{X} \cup \mathcal{A} \cup \mathbb{R}\}$. The intensity function and mark distribution $\lambda^a(t|\mathcal{H}_t), \pi(A_t|\mathcal{H}_t)$ are called the policy. The mark distributions for X, Y are denoted by $P_X(\mathbf{x}_t|\mathcal{H}_t), P_Y(y_t|\mathcal{H}_t)$.*

Off-policy evaluation. We are given a dataset of m trajectories, where trajectory $\mathcal{H}_i$ has n_i observations: $\{(t_{i,k}, \mathbf{e}_{i,k})\}_{i \in [m], k \in [n_i]}$. These are sampled from an observed decision process P_{obs} with policy $(\lambda^a_{\text{obs}}, \pi_{\text{obs}})$. Treatment times are samples from a counting process with intensity λ^a_{obs} and treatments at those times are sampled from π_{obs}. We reason about outcomes when $(\lambda^a_{\text{obs}}, \pi_{\text{obs}})$ is replaced with a target policy (λ^a, π), with other processes in P_{obs} fixed. The resulting decision process is denoted P, and our goal is to estimate the expected future outcome $\mathbb{E}_P\left[Y|\mathcal{H}_t\right]$ for all $t \in [0, T)$ and $\mathcal{H}_t$ in the natural filtration associated with the point process.

When to treat. To simplify notation, we omit the marks $\pi(A_t|\mathcal{H}_t)$ and focus on intensities $\lambda^a(t|\mathcal{H}_t)$. That is, we explore interventions on when to treat (*medicine weekly or monthly?*) instead of how to treat (*which medication?*). Technically, “when" is the more challenging and underexplored part of the problem, and solutions can be easily extended to incorporate interventions on π using existing methods (Chakraborty and Murphy, 2014; Li et al., 2021). In the ASCVD example, this corresponds to reasoning about questions like: “Consider prescribing statins to patients with characteristics $\mathcal{H}_t$, whose LDL cholesterol is below 180 mg/dl up to time t. What would be the expected change in 10-year ASCVD risk if going forward, we prescribe a daily dose of statins for patients whose LDL cholesterol goes above 180 mg/dL, instead of the existing policy followed in the population?".

2.2 Roadmap to identifiability via Local Independences

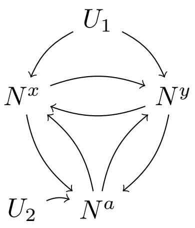


Figure 2: The assumed local independence graph for a decision point processes, where our estimand is identifiable from observed data (N^x, N^a, N^y).

The goal of this section is to elucidate the conditions under which the algorithm we present in section 3 estimates valid causal effects. We briefly summarize the essential conditions and supplement this summary in appendix D. Our assumptions to ensure identifiability of causal estimands follow Didelez (2008); Røysland (2011); Røysland et al. (2022), who study graphical models for point processes. In this setting, where the goal is to intervene on N^a and estimate $E_P[Y|\mathcal{H}_t]$ under P rather than P_{obs}, in the presence of unobserved processes U, Røysland (2012) define and analyze the following notions:

- A graphical condition called *causal validity* ensures that changing the treatment intensity from λ^a_{obs} to the interventional λ^a, while changing no other intensities, changes the joint distribution from P_{obs} to P. A graph may not be causally valid when it contains unobserved variables U.
- *Local independence* (Aalen, 1987; Schweder, 1970) adapts sequential exchangeability (Hernan and Robins, 2023; Robins, 1986) to continuous time. It is an asymmetric form of independence where one process' intensity may depend on the history of another, but not vice versa.
- A certain set of local independences together are referred to as *eliminability* (a generalization of the backdoor criterion), which implies casual validity, even with unobserved variables U.

Consider the graph in Figure 2, where an edge means that the history of the source node affects the future of the target node (Didelez, 2008). It is possible to show that it satisfies eliminability (see appendix D). To understand this condition, we start with the basic local independence requirements.

Definition 2. *For a multivariate process $N(t) = (N^a(t), N^b(t), \dots)$ on variables V we say that N^a is locally independent of N^b given $N^{\setminus b}$, or $N^b \not\rightarrow N^a | N^{\setminus b}$, if $\lambda^a(t|\mathcal{H}_t) = \lambda^a(t|\mathcal{H}_t^{\setminus b})$. A graphical local independence model $(\mathcal{P}, G)$ is a class of processes $\mathcal{P}$ on V and directed graph $G = (V, E)$, such that $(b \rightarrow a) \notin E \Rightarrow N^b \not\rightarrow N^a | N^{\setminus b}$ holds for all $P \in \mathcal{P}$.*

This condition means that the intensity of a process only makes use of certain information from other processes. Røysland et al. (2022) package together the set of local independences that imply causal validity under the name *eliminability*, defined below. They use a graphical criterion that is akin to using d-separation in graphical models, while we state the conditions in terms of the implied functional independencies of intensities. We expand on this in appendix D.

Definition 3. *Let U be the set of unobserved variables. Suppose U can be written as a sequence $(U_1, \dots, U_K)$ such that for each k, either*

- *$(N^y, N^x, U_{>k})$ is locally independent of U_k given $(N^x, N^y, N^a, U_{>k})$, or*
- *N^a is locally independent of U_k given $(N^x, N^y, N^a, U_{>k})$.*

Then the graph is said to satisfy Eliminability. Here, for sets A, C and variable b we say A is locally independent of b given $A \cup C$ if each $a \in A$ is locally independent of b given $A \cup C \setminus \{a\}$.

We summarize that *causal validity* holds under the *local independences* that satisfy *eliminability*. We assume the graph in Figure 2, which satisfies these assumptions. We also assume mutual independence of increments of all processes at a one time to rule out instantaneous effects. We refer to validity and this independence together as **ignorability**, in accordance with existing terminology on confounding.

Assumption 1. *Ignorability (in continuous time) is satisfied when:*

1. *the graph satisfies causal validity,*
2. *the increments of features, treatments, and outcome are mutually independent given the history, i.e., $((dN^x(t), X_t) \perp\!\!\!\perp (dN^a(t), A_t) \perp\!\!\!\perp (dN^y(t), Y_t))|\mathcal{H}_t$.*

In addition to ignorability, we require a second, standard assumption, **overlap**, for the conditional expectations we estimate to be well-defined. Recall that the interventional distribution P is defined by replacing the treatment distribution in P_{obs}, i.e., replacing λ^a_{obs} with λ^a.

Assumption 2. *Overlap is said to hold between the observational and interventional distributions, P_{obs} and P, if P is absolutely continuous with respect to P_{obs}, denoted by $P \ll P_{\text{obs}}$.*

Ignorability and overlap are the core assumptions that allow identification in our setting. Under these assumptions, we can now present algorithms for estimating causal effects in continuous time.

3 Model Free Off-Policy Evaluation for Decision Point Processes

To estimate $\mathbb{E}_P[Y|\mathcal{H}_t]$ for times t and $\mathcal{H}_t$ that overlap with P_{obs}, we express the expectation recursively as a function of expectations $\mathbb{E}_P[Y|\widetilde{\mathcal{H}}_{t+\delta}]$ for some $\delta > 0$ and trajectory $\widetilde{\mathcal{H}}$. Then, assuming that expectations at times larger than t have been learned correctly, this recursive expression allows us to propagate information for conditioning on earlier histories.

Let us describe this solution in more detail, as applied in the context of Q-evaluation.

Fitted Q evaluation (FQE) in discrete time. Q-evaluation relies on the tower property of conditional expectations, given below in eq. (1). In discrete time decision processes, where we consider $\delta = 1$, the property suggests a dynamic programming solution that we lay out in algorithm 1 (Le et al., 2019; Watkins and Dayan, 1992). Here, $\mathcal{H}_t$ includes all treatments and observations up to and including time t, and since they occur simultaneously, there are exactly t of each. $\widetilde{\mathcal{H}}_{t+1}$ is defined in the same manner, except that it includes $\tilde{a}_{t+1}$ sampled from the target policy π.

Algorithm 1 Fitted Q-Evaluation (discrete time)

```
Input: Trajectories $\{\mathcal{H}_i\}_{i=1}^m$,
    Policy $\pi : \cup_{t=1}^T \mathcal{X}^t \times \mathcal{A}^{t-1} \to \Delta^{|\mathcal{A}|}$
    Model class $\mathcal{F}$ where $f : \cup_{t=1}^T \mathcal{X}^t \times \mathcal{A}^t \to \mathbb{R}$
Initialize $Q$ randomly
Set $T_i \leftarrow \max\{t : (t, \mathbf{z}) \in \mathcal{H}_i\} \quad \forall i \in [m]$
for $N$ rounds do
    Draw $\tilde{a}_{t+1} \sim \pi(\cdot|\mathcal{H}_{i,t}, \mathbf{x}_{i,t+1}) \quad \forall i, t \in [T_i]$
    $\widetilde{\mathcal{H}}_{i,t+1} \leftarrow \mathcal{H}_{i,t} \cup \{(t+1, \mathbf{x}_{t+1}), (t+1, \tilde{a}_{t+1})\}$
    Set $\hat{y}_{i,t} = y_{i,t} + Q(\widetilde{\mathcal{H}}_{i,t+1}) \quad \forall i, t \in [T_i]$
    $Q \leftarrow \arg\min_{f \in \mathcal{F}} \sum_{i \in [m], t \in [T_i]} \left(f(\mathcal{H}_{i,t}) - \hat{y}_{i,t}\right)^2$
end for
Return $Q$
```

$$\mathbb{E}_P\left[Y|\mathcal{H}_t\right] = \mathbb{E}_{\widetilde{\mathcal{H}}_{t+\delta} \sim P(\cdot|\mathcal{H}_t)}\left[\mathbb{E}_P\left[Y|\widetilde{\mathcal{H}}_{t+\delta}\right]\right], \tag{1}$$

An attractive property of this algorithm is that it is *model-free*. That is, to form the label $\hat{y}_i$ we only need to sample $\tilde{a}_{t+1}$ from our target policy π, while a model of $P(\mathbf{x}_{t+1}|\mathcal{H}_t)$ is not necessary.

With accurate optimization over a sufficiently expressive hypothesis class and arbitrarily large datasets, algorithm 1 returns correct estimates. This is because if we fix $Q_{t+1}(\mathcal{H}_{t+1})$ and assume that it accurately estimates $\mathbb{E}_P[\sum_{s \geq t+1} Y_s|\mathcal{H}_{t+1}]$, then the minimizer of the regression is the conditional expectation, equal to $\mathbb{E}_P[\sum_{s \geq t} Y_s|\mathcal{H}_t]$ according to eq. (1). The model-free solution is enabled by the equality $P(\mathbf{x}_{t+1}|\mathcal{H}_t) = P_{\text{obs}}(\mathbf{x}_{t+1}|\mathcal{H}_t)$, which validates the use of $\mathbf{x}_{i,t+1}$ to form $\hat{y}_i$. In practice, we take gradient steps on randomly drawn times and training samples instead of walking backward from T to 1. Crucially, for $\delta > 1$, e.g. $\delta = 2$, we have $P(\mathbf{x}_{t+2}|\mathcal{H}_t) \neq P_{\text{obs}}(\mathbf{x}_{t+2}|\mathcal{H}_t)$. Hence, an algorithm using eq. (1) must either be model-based, or resort to solutions such as importance weights that suffer high variance (Hallak et al., 2016; Precup et al., 2000), or restrict the problem, e.g., by discounting rewards (Harutyunyan et al., 2016; Munos et al., 2016; Precup et al., 2000).

Challenges in application to continuous time. Moving to continuous time, the tower property turns into a differential equation, and solving it requires tools that go beyond common FQE (e.g., Jia and Zhou (2023)). Regressing to an outcome that is arbitrarily close to the observation at time t is ill-defined. While we may work under a fine discretization of time, this approach is wasteful, as a single update in the minimization for estimating Q_t takes into account the development of the process in the interval $[t, t+\delta]$, and for small values of δ this will usually yield a very small change to the estimate. Hence, intuitively, when updating Q_t, we would like to use estimates of $Q_{t+\delta}$ for a large δ. As explained above, this is seemingly difficult to achieve in a model-free fashion. However, for point processes, since the number of decisions over $[0, T]$ is countable, it seems plausible that a simple and efficient dynamic programming solution can be devised. In what follows, this is what we present.

3.1 EDQ: Fitted Q-Evaluation for Decision Point Processes via Earliest Disagreement Times

We wish to reason about what the outcome would have been for an observed trajectory sampled under policy λ^a_{obs}, had we treated it with policy λ^a from time t onward. Intuitively, it seems plausible that we can use a similar approach to FQE, but instead of going one time unit forward, we can move to the first time $t + \delta$ where the two policies differ in their treatments. That is, we will sample alternative treatment trajectories, asking what the target policy λ^a would have done at each time, given the observed history up until that time and find the earliest disagreement. The resulting algorithm is rather simple, and we summarize it graphically in fig. 1. The attractive property of this approach is that the “lookahead" time δ is adaptive. It will likely be short when applied in parts of trajectories where many treatments are applied, and longer when treatments are scarce.

To formalize the method, we present some additional notation, prove an appropriate variation of the tower property, and explain how it is operationalized by the implementation in algorithm 2.

Definition 4. *For process P_{obs} and policy (λ^a, π), define an augmented process $\tilde{N} = (N^{a_{\text{obs}}}, N^y, N^x, N^a)$ where intensities are independent of the history of $N^{a_{\text{obs}}}$: (1) the intensity of N^a is $\lambda^a(t|\mathcal{H}_t^{\backslash a}) := \lambda^a(t|\mathcal{H}_t^{x,y}, \mathcal{H}_t^a = \mathcal{H}_t^{a_{\text{obs}}})$, i.e. λ^a where history of treatments is given by $\mathcal{H}_t^{a_{\text{obs}}}$,* [2] *and (2) $\lambda^e(t|\mathcal{H}_t) = \lambda^e_{\text{obs}}(t|\mathcal{H}_t^{\backslash a})$ for $e \in \{a_{\text{obs}}, x, y\}$. For a trajectory $\widetilde{\mathcal{H}} \sim \tilde{P}$ and time $t \in [0, T)$, define $\delta_{\widetilde{\mathcal{H}}}(t) = \min\{u - t : u > t, (u, \cdot) \in \widetilde{\mathcal{H}}^{a, a_{\text{obs}}}\}$, where $\delta_{\widetilde{\mathcal{H}}}(t) = T - t$ when the set is empty.*

The augmented process maintains an additional treatment trajectory, $\widetilde{\mathcal{H}}^a$, as an alternative to $\widetilde{\mathcal{H}}^{a_{\text{obs}}}$, the one observed under λ_{obs}. From definition 4, we observe that the intensities of the augmented process $\tilde{N}$ do not depend on N^a's history. It follows that the marginal over $\{x, a, y\}$ is P_{obs}, while $\widetilde{\mathcal{H}}^a$ has a similar role to the alternative treatment $\tilde{a}_{t+1}$ in algorithm 1. This notation is helpful for denoting sampled alternative trajectories over time intervals. Finally, $\delta_{\widetilde{\mathcal{H}}}(t)$ denotes the earliest disagreement between observed and target treatments after time t.[3] Our model-free evaluation method is based on the following result, which expresses our estimand as an expectation over trajectories $\widetilde{\mathcal{H}}$.

Theorem 1. *Let P_{obs} be a marked decision point processes, P the process obtained by replacing the policy with (λ^a, π), and $\tilde{P}$ the augmented process obtained from P, P_{obs} in definition 4. Further, let $t \in [0, T)$, and $\mathcal{H}_t$ measurable w.r.t P. Under Assumption 2, we have that*

$$\mathbb{E}_P\left[Y|\mathcal{H}_t\right] = \mathbb{E}_{\widetilde{\mathcal{H}} \sim \tilde{P}(\cdot|\mathcal{H}_t)}\left[\mathbb{E}_P\left[Y \,\middle|\, \mathcal{H}_{t+\delta_{\widetilde{\mathcal{H}}}(t)} = \mathcal{H}_t \cup \widetilde{\mathcal{H}}^{\backslash a_{\text{obs}}}_{(t, t+\delta_{\widetilde{\mathcal{H}}}(t)]}\right]\right]. \tag{2}$$

Takeaways from Theorem 1 and derivation of EDQ. Equation (2) suggests a method to calculate expectations $\mathbb{E}_P[Y|\mathcal{H}_t]$, similar to how FQE follows from eq. (1). The practical version of the resulting method, EDQ, is given in algorithm 2. To arrive at the method from eq. (2), let us examine the regressions solved in algorithm 2. Taking a random variable $Y_{>s}$ which is the sum of all outcomes after some time s, eq. (2) can be rewritten as follows (see Appendix A for a detailed derivation).

$$\underbrace{\mathbb{E}_P\left[Y|\mathcal{H}_t\right] - \sum_{(t_k, y_k) \in \mathcal{H}_t^y} y_k}_{Q(\mathcal{H}_t)} = \mathbb{E}_{\widetilde{\mathcal{H}}|\mathcal{H}_t}\Bigg[\sum_{\substack{(t_k, y_k) \in \widetilde{\mathcal{H}}^y: \\ t_k \in (t, t+\delta_{\widetilde{\mathcal{H}}}(t)]}} y_k + \underbrace{\mathbb{E}_P\left[Y_{>t+\delta_{\widetilde{\mathcal{H}}}(t)} \,\middle|\, \mathcal{H}_{t+\delta_{\widetilde{\mathcal{H}}}(t)} = \mathcal{H}_t \cup \widetilde{\mathcal{H}}^{\backslash a_{\text{obs}}}_{(t, t+\delta_{\widetilde{\mathcal{H}}}(t)]}\right]}_{Q(\mathcal{H}_{t+\delta_{\widetilde{\mathcal{H}}}(t)})}\Bigg] \tag{3}$$

Algorithm 2 uses gradient descent to fit Q functions that, at optimality, satisfy a self-consistency condition that appears in the equation above, $Q(\mathcal{H}_t) = \mathbb{E}_{\widetilde{\mathcal{H}}}[\sum y_k + Q(\mathcal{H}_{t+\delta_{\widetilde{\mathcal{H}}}(t)})]$. To this end, it solves regression problems where $Q(\mathcal{H}_{i,t})$ is fitted to a label $\hat{y}_i$. This label is defined in line 7 and coincides with the term in the expectation $\mathbb{E}_{\widetilde{\mathcal{H}}|\mathcal{H}_t}[\cdot]$ above. To conclude that the algorithm estimates the effect of interest, two simple arguments suffice: (1) a uniqueness argument showing that the self-consistency equation holds only when $Q(\mathcal{H}_t)$ matches $\mathbb{E}_P[Y_{>t}|\mathcal{H}_t]$, and (2) that despite ignoring the unobserved process λ^u, the estimation yields the desired causal effect due to the ignorability assumption. We summarize the conclusion below and provide detailed steps in appendix D.

Corollary 1. *Under assumption 1, a Q-function satisfying eq. (3) yields the causal effect of the intervention that replaces $(\lambda^a_{\text{obs}}, \pi_{\text{obs}})$ with (λ^a, π).*

An analogue of eq. (2) holds for discrete-time decision processes, where EDQ bears some resemblance to the eligibility traces approach of Precup et al. (2000). We provide this result in appendix B.3 for completeness but focus here on the point process case, as this is our main motivation and where the earliest disagreement approach is most fruitful.

4 Related Work

Our coverage of related work is divided into an overview of works that solve adjacent tasks to ours, before transitioning into a detailed discussion in section 4.1 about techniques more closely aligned with our goal of large scale causal inference in sequential decision making.

Causal inference with sequential decisions. Causal effect estimation for sequential treatments is usually studied in discrete time under the sequential exchangeability assumption (Hernan and

[2] Note that this is a slight abuse of notation, since $\mathcal{H}_t^a$ is not a random variable.

[3] The minimum treatment time is also the earliest disagreement, since under some regularity conditions, the processes N^a and $N^{a_{\text{obs}}}$ have probability 0 of jumping simultaneously.

Algorithm 2 Earliest Disagreement Fitted Q-Evaluation

```
Input: Trajectories $\{\mathcal{H}_i\}_{i=1}^m$,
       Policy $\lambda_a(\cdot|\mathcal{H}_t), \pi(\cdot|\mathcal{H}_t)$
Initialize $\boldsymbol{\theta}$ randomly
for $N$ rounds do
    Draw $t \sim \text{Unif}([0,T])$ and $i \sim \text{Unif}([m])$
    Draw $\widetilde{\mathcal{H}} \sim \widetilde{P}(\cdot|\mathcal{H}_{i,t})$ and set $\mathcal{H}'_{i,t+\delta_{\widetilde{\mathcal{H}}}(t)} = \mathcal{H}_t \cup \widetilde{\mathcal{H}}^{\setminus a_{\text{obs}}}_{(t,t+\delta_{\widetilde{\mathcal{H}}}(t)]}$
    $\hat{y}_i = \sum_{\substack{(t_k,y_k)\in\widetilde{\mathcal{H}}^y:\\ t_k\in(t,t+\delta_{\widetilde{\mathcal{H}}}(t)]}} y_k + Q(\mathcal{H}'_{i,t+\delta_{\widetilde{\mathcal{H}}}(t)};\boldsymbol{\theta})$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta\nabla_{\boldsymbol{\theta}}\left(Q(\mathcal{H}_{i,t};\boldsymbol{\theta}) - \hat{y}_i\right)^2$
end for
Return $Q(\cdot;\boldsymbol{\theta})$
```

Robins, 2023; Robins, 1986). Addressing unobserved confounders is of interest (Namkoong et al., 2020; Tennenholtz et al., 2020), but this is beyond our scope here. As described in section 2.2, the framework of Røysland et al. (2022) draws a parallel to sequential exchangeability for continuous time, which we adopt here. For estimation in these continuous-time problems, several methods have been explored (Lin et al., 2004; Lok, 2008; Røysland, 2011; Rytgaard et al., 2022; Zhang et al., 2011). Most do not scale to large and high-dimensional datasets. For instance, Røysland (2011) requires estimating an integral of propensity weights over time; Rytgaard et al. (2022) propose a targeted estimator that fits each process in P_{obs}, but it is limited to interventions at fixed times (i.e. no interventions on treatment schedules) and it is unclear how to implement it with expressive models. Nie et al. (2021) study the effect of timing t when applying a fixed policy from time t onward. However, they do not discuss interventions on all treatment timings, as we do here.

Reinforcement learning (RL) techniques. As discussed in section 3, EDQ is related to FQE and Q-learning (Le et al., 2019; Murphy, 2005; Watkins and Dayan, 1992), and to n-step methods from RL (De Asis et al., 2018; Munos et al., 2016; Precup et al., 2000). In the causality literature, Q-learning often appears in the context of dynamic treatment regimes (Chakraborty and Moodie, 2013; Chakraborty and Murphy, 2014), where discrete-time policies are learned off-policy. All these methods are not applicable to irregularly sampled times. Works in RL that consider irregularly sampled times, either do not intervene on treatment times, or operate in the on-policy setting (Qu et al., 2023; Upadhyay et al., 2018), or incorporate continuous-time positional embeddings into decision transformers (Chen et al., 2021b). The latter facilitates the recommendation of sets of actions to arrive at a desired outcome (Zhang et al., 2023) rather than evaluating a policy of interest. Recent work suggests that goal-conditioned imitation learning methods such as decision transformers may fail to estimate the causal effect of actions (Malenica and Murphy, 2023) in some scenarios where there are no unobserved confounders, whereas Q-learning methods produce correct estimates.

We next discuss *scalable* causal estimation methods for sequential treatments. We delineate assumptions, implementation choices, and subsequent properties of solutions from recent work.

4.1 Large Scale Estimation Approaches for Sequential Treatments

Notable early work on machine learning for estimating counterfactual quantities related to treatment timelines, Schulam and Saria (2017), used Gaussian Processes for estimation. Limitations such as scalability and incorporating various features prompted the development of deep learning approaches.

Large scale models. One family of solutions (Bica et al., 2020; Lim, 2018; Melnychuk et al., 2022) took an important step forward by using RNNs and transformers for estimation. All of these methods build on the idea of learning balancing representations (Johansson et al., 2016). Roughly, these are representations under which the treatment is randomly assigned. This facilitates the training of high-capacity effect estimators on large datasets, but these works are restricted to discrete times.

Dynamic policies. The above methods can only estimate effects of *static* treatments, meaning the treatment plan cannot dynamically depend on future observations. For instance, consider a policy that prescribes a daily dose of statins, and if the patient develops side effects in the future, switches

[4] We mark TE-CDE with as non-scalable, since the algorithm relies on differential equation solvers which limits the scalability of methods one can use.

Table 1: Qualitative comparison of estimation methods under sequential treatments, compatible with large scale ML. Our method EDQ is marked in bold, while others can be found in (Bica et al., 2020; Chakraborty and Moodie, 2013; Le et al., 2019; Li et al., 2021; Lim, 2018; Melnychuk et al., 2022; Schulam and Saria, 2017; Seedat et al., 2022). The acronym DP stands for Dynamic Programming.

	Properties			**Estimation Method**			
	Irregular Times	Large Scale	Dynamic Policy	Prop. Weights	Model Based	Balancing Rep.	DP
CGP	✗	✗	✓		●		
CRN , CT	✗	✓	✗			●	
R-MSN	✗	✓	✗	●			
TE-CDE	✓	✗[4]	✗		●	●	
G-Net	✗	✓	✓		●		
FQE	✗	✓	✓				●
EDQ	✓	✓	✓				●

to another medication. This policy depends on possible future states, which the above methods do not accommodate. G-Net (Li et al., 2021; Xiong et al., 2024) takes a model-based approach, where models are fit for both $\pi_{\text{obs}}(A(t)|\mathcal{H}_{t-1}, X(t))$, and $P_{\text{obs}}(X(t), Y(t)|\mathcal{H}_{t-1})$. Then at inference time, a dynamic policy is estimated when π_{obs} is replaced with the desired policy π and conditional expectations of Y are estimated with Monte-Carlo simulations.

Irregular times. None of the above solutions address irregular observation and action times, a key focus of this paper. Some steps in this direction have been taken by Seedat et al. (2022); Vanderschueren et al. (2023). Past work on irregular times includes Seedat et al. (2022), who combine balanced representations with a neural controlled differential equation architecture suited for irregular sampling, while Vanderschueren et al. (2023) use reweighting to adjust for sampling times that are informative of the outcome. However, these methods mitigate sampling-induced bias rather than estimate outcomes under interventions on treatment times. We further discuss them in appendix C.

Table 1 summarizes the above techniques, along with FQE and EDQ. Notably, EDQ possesses the desirable qualities mentioned here and handles interventions on λ^a, which other solutions do not.

5 Implementation and Experiments

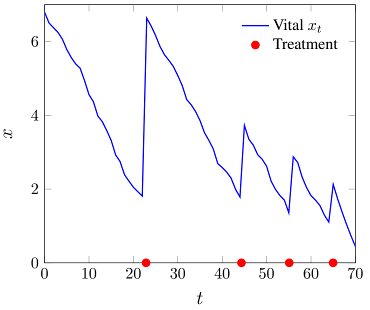
To implement EDQ for experimentation we use the GPT-2 architecture. Each token is a concatenation of embeddings of time t_i, value $\mathbf{z}_i$ and event type $e_i \in \{A, X, Y, \Delta T\}$. The event types A, X, Y correspond to actions, features, and outcomes, while ΔT is introduced for convenience to represent cases where time passes but no events occur. Both absolute times t_i and time gaps ΔT use continuous time positional embeddings: the k-th dimension is $\sin\left(tC^{-k/d_{\text{time}}}\right)$ for even k, and $\cos\left(tC^{-(k-1)/d_{\text{time}}}\right)$ for odd k. Here $C = 10^5$ and d_{time} is the embedding dimension. We also keep a target network and update it with soft-Q updates, as is common in Deep Q-Networks, e.g., Van Hasselt et al. (2016).

5.1 Baselines

We are unaware of baselines for effect estimation on treatment timing with high-dimensional or long-sequence data. Thus, we implement two baselines that let us glean important aspects of EDQ.

ERM / MC is an Empirical Risk Minimizer (ERM) trained to predict observed outcomes, which, also known as Monte-Carlo (MC) prediction in RL and policy evaluation. We use the same GPT-2 architecture and data representation as EDQ, but instead of running algorithm 2, we train $f_{\boldsymbol{\theta}}(\mathcal{H}_t)$ by minimizing prediction loss on observed data. Given each training trajectory $\mathcal{H}_i$ with an outcome label y_i, we solve $\min_\theta \sum_{i,(t_i,\mathbf{z}_i)\in\mathcal{H}_i} \left(\ell(f_{\boldsymbol{\theta}}(\mathcal{H}_{i,t}), y_i\right)$, where $\ell(\cdot,\cdot)$ is the squared loss. This method estimates outcomes under λ_{obs}, therefore we expect it to perform as well as or better than off-policy evaluation methods, including EDQ, when $\lambda = \lambda_{\text{obs}}$, and to suffer a drop otherwise.

FQE is implemented following section 3, but with discretized time and Q-updates using one timestep forward. At each iteration, we draw training example $i \in [m]$ and time $t \in [0, T]$, define $\hat{y}_i = y_{t+1} + Q_{t+1}(\widetilde{\mathcal{H}}_{t+1}; \boldsymbol{\theta}))$, and take a gradient step on loss $\ell(Q_t(\mathcal{H}_{i,t}), \hat{y}_i)$. Similarly, we define discrete-time approximations of our policies of interest, described later. The positional embeddings



	ERM / MC	FQE	EDQ
		$\lambda_{\text{int}} = 0.1$	
$\lambda_{\text{obs}} = 0.1$	0.20 ± 0.007	0.21 ± 0.02	0.20 ± 0.005
$\lambda_{\text{obs}} = 0.5$	0.38 ± 0.011	0.23 ± 0.04	$\mathbf{0.20 \pm 0.006}$
		$\lambda_{\text{int}} = 0.5$	
$\lambda_{\text{obs}} = 0.5$	0.11 ± 0.004	0.197 ± 0.013	0.10 ± 0.003
$\lambda_{\text{obs}} = 0.1$	0.28 ± 0.004	0.31 ± 0.01	$\mathbf{0.11 \pm 0.006}$

Figure 3: **Left.** An example of a trajectory from our simulation. The blue curve denotes the value of the vital x_t and red dots mark treatment times. **Right.** Normalized RMSE under the different simulation settings. The mean is taken over all points in the history of patients in the test data. Rows colored blue have $\lambda_{\text{obs}} = \lambda_{\text{int}}$, and we expect all methods to perform well since train and test distributions match. Red rows are those where the effect of an intervention needs to be estimated.

correspond to discrete times, and the representations of actions, features and outcomes at each time are concatenated. Other than this, we use the same architecture and hyperparameters of EDQ. This baseline examines the effects of time discretization on estimation and optimization.

Computational complexity: The per-iteration runtime of EDQ is similar to FQE, which is a common tool for large-scale offline RL problems, e.g. Paine et al. (2020); Voloshin et al. (2021). EDQ and FQE differ in computation times due to sampling methods from the target policy used to draw the treatments used in the Q-update. We discuss this in appendix A.

5.2 Simulations on Time to Failure and Cancer Tumor Growth Prediction

To validate our method, we construct two settings. The first is to predict the effect of treatment timing policies on patients' time-to-event. The second uses a cancer tumor growth simulator from Geng et al. (2017) to form a policy evaluation problem on applications of chemotherapy and radiotherapy.

Simulators. We use two simulators. **(i) Time-to-failure:** In this setting, each data point simulates the vital of a patient $x_t \in \mathbb{R}_+$ measured regularly at a frequency of one time unit, and treatments $a_t \in \mathbb{R}_+$ are assigned irregularly in time according to an observational policy. Without treatment, the vital drops linearly $dx_t/dt = -(\alpha + \xi_t)$ where $\alpha > 0$ and $\xi_t \sim \mathcal{N}(0, \sigma)$ is a noise term drawn at each time unit. Upon receiving treatment, the vital rises by an amount inversely proportional to the number of treatments, $1 \leq k \leq m$, applied up until that time, where m is the maximal number of treatments that a patient can receive. That is, the efficacy of treatment reduces with repeated applications. We also inject small noise terms into the treatment dosage that a patient receives, which further affect the vital and add randomness to the problem. Section 5.2 shows an example simulated patient trajectory. **(ii) Tumor growth:** We use the experimental setting from Bica et al. (2020), which other works use to study irregular sampling (Seedat et al., 2022; Vanderschueren et al., 2023). As this is a commonly used simulator, we defer its details to appendix A and focus on the type of irregular sampling and policies we use. The simulator works in discrete time $t \in [T]$, and irregular sampling is induced by the features being unobserved at certain times. Namely, the covariate $x_t \in \mathbb{R}_+$ which represents tumor volume is observed with probability $\sigma((\bar{x}_{t-d:t-1}/d_{\max}) - 1.5)$, where $\bar{x}_{t-d:t-1}$ is the average tumor volume over the last d timesteps, and $d_{\max}$ is the maximum considered volume.

Outcomes and policies. For **(i) time-to-failure**, our outcome of interest is failure time $y \in \mathbb{R}_+$, where a patient dies if the vital drops to a value of 0. [5] We focus on effect estimation for interventions on a rate parameter λ^a that controls the timing of treatment. At each time t where the observed vital crosses a threshold, i.e. $x_t < r$ for some predetermined $r \in \mathbb{R}_+$, a random time is drawn from an exponential distribution $\delta \sim \exp(\lambda^a)$ and treatment is applied at $t + \delta$. The threshold r and the dosage of treatment given are also part of the policy π, yet to focus on the effects of timing we do not intervene on them in this experiment. At each experiment we observe m patients treated under a policy with $\lambda^a = \lambda_{\text{obs}}$ and aim to reason about the expected failure times under the interventional λ_{int}. In **(ii) tumor-growth**, the goal is to predict x_T, where $T = 20$, under a policy that at time t assigns treatment a_t from four possible options: no-treatment, radiotherapy, chemotherapy, or combined

[5] Note that the vital changes outside measurement times, hence death time does not generally coincide with vital measurement times.

	ERM / MC	FQE	EDQ
		$(\gamma, \beta)_{\text{int}} = (6, 0.75)$	
$(\gamma, \beta)_{\text{obs}} = (6, 0.75)$	$\mathbf{0.034 \pm 0.001}$	0.048 ± 0.001	0.037 ± 0.001
$(\gamma, \beta)_{\text{obs}} = (10, 0.5)$	0.07 ± 0.004	0.080 ± 0.013	$\mathbf{0.052 \pm 0.006}$

	ERM / MC	FQE	EDQ
		$\lambda_{\text{int}} = 0.2$	
$\lambda_{\text{obs}} = 0.2$	$\mathbf{0.17 \pm 0.01}$	0.18 ± 0.004	0.178 ± 0.01
$\lambda_{\text{obs}} = 2$	0.28 ± 0.01	0.20 ± 0.03	$\mathbf{0.178 \pm 0.01}$
		$\lambda_{\text{int}} = 2$	
$\lambda_{\text{obs}} = 2$	0.22 ± 0.01	$\mathbf{0.197 \pm 0.013}$	0.22 ± 0.004
$\lambda_{\text{obs}} = 0.2$	0.32 ± 0.02	0.31 ± 0.01	$\mathbf{0.22 \pm 0.007}$

Figure 4: **Left.** Normalized RMSE on the **tumor-growth** simulation. All methods are affected by distribution shift. EDQ is the most robust out of the methods considered. **Right.** Normalized RMSE for **time-to-failure** simulation on short trajectories.

therapy. Therefore, the estimation here is both on "when" and "what" to do. Policies are determined by two parameters (γ, β) and assign each type of treatment with probability $\sigma(\gamma(x_{\text{last}} - \beta) + t - t_{\text{last}})$. Here, x_{last} is the last observed volume and β is an intercept controlling how often treatments are applied, while γ controls the dependence of treatment assignment on tumor volume. Finally t_{last} is the time of the last treatment, and the term $t - t_{\text{last}}$ induces a lag between consecutive treatments.

Experiments. We perform two sets of experiments for **time-to-failure**. In the first set, trajectory lengths range in $10 - 100$, and there are 5 treaments. In the second set (results in Figure 4, right), we change the parameters of the problem by taking a high slope α and capping at one treatment. This creates short trajectories of length between 3 and 10. To evaluate the performance of the estimator, we sample trajectories $(\mathcal{H}_i, y_i) \sim P_{\lambda_{\text{int}}}$ under the target policy and treat every $(\mathcal{H}_{i,t_i}, y_i)$ as a labeled data point. We then evaluate normalized RMSE between $f_{\boldsymbol{\theta}}(\mathcal{H}_{i,t_i})$ and the true labels y_i. For **tumor growth**, we evaluate a policy that increases the likelihood of treatments (i.e increases β) and reduces γ, the correlation to the observed volume. Error is also calculated with normalized RMSE.

Results. The tables in fig. 3 and fig. 4 present results. They show that for **time-to-failure**, EDQ solves the estimation problem both when $\lambda_{\text{obs}} = \lambda_{\text{int}}$ (blue rows, no intervention performed), and when $\lambda_{\text{obs}} \neq \lambda_{\text{int}}$ (red rows). This is evident by comparing to ERM under the setting where $\lambda_{\text{obs}} = \lambda_{\text{int}}$, where ERM should be nearly optimal. [6] ERM takes a significant performance drop when $\lambda_{\text{obs}} \neq \lambda_{\text{int}}$, as expected. For FQE, while in the first set of experiments (fig. 3), discretization should not result in significant information loss, it does create a more difficult optimization problem. This is because the updates to $Q(\mathcal{H}_t)$ need to propagate backwards and most updates get noisy gradient signals by fitting to the Q value of a trajectory sampled one time step ahead, $Q(\widetilde{\mathcal{H}}_{t+1})$. This challenge for FQE is most evident in Figure 3, where $\lambda_{\text{int}} = 0.5$ and FQE incurs a significant loss both when $\lambda_{\text{obs}} = \lambda_{\text{int}}$ and when $\lambda_{\text{obs}} \neq \lambda_{\text{int}}$. The results in Figure 4 (right) demonstrate the potential effects of information loss due to discretization. Here, since trajectories are short, the optimization problem of losses propagating is likely less pronounced. However, we see that there is still a significant drop when $\lambda_{\text{int}} = 2$ and $\lambda_{\text{obs}} = 0.2$ (approximating a high rate of treatment when data was sampled under low rates). Taken together, these two experiments demonstrate two possible drawbacks of discretization. For **tumor-growth**, EDQ still outperforms the alternatives but suffers a decrease in performance due to distribution shift between observational and interventional distributions.

6 Limitations and Future Work

We have developed a method for off-policy evaluation with irregular treatment and observation times, which facilitates interventions on treatment intensities. We connected the setting to identifiability results from causal inference to highlight the conditions under which the estimates are meaningful, and proved the estimator's correctness. EDQ is a "direct" method based on regression and, as demonstrated in experiments, is readily applicable to high-capacity sequence modeling architectures. To the best of our knowledge, it is the first available solution to this estimation problem that is applied with such architectures. Several limitations motivate exciting future research. Empirically, we plan to apply the method to large real-world datasets and additional simulators (Namkoong et al., 2020; Oberst and Sontag, 2019). The method does not handle censoring, which is required for reliable application in survival analysis and real trial data. Further developements include policy optimization in the setting we studied here and deriving bounds on errors due to unobserved confounding.

[6] this is up to numerical optimization issues, as we see FQE can outperform it in certain cases

Acknowledgments

We thank Stefan Groha for helpful discussion in early stages of the project. This work was supported by National Science Foundation Award 1922658, NIH/NHLBI Award R01HL148248, NSF CAREER Award 2145542, NSF Award 2404476, ONR N00014-23-1-2634, IITP with a grant from ROK-MSIT in connection with the Global AI Frontier Lab International Collaborative Research, Apple, and Optum.

References

Aalen, O., Borgan, O., and Gjessing, H. (2008). *Survival and event history analysis: a process point of view*. Springer Science & Business Media.

Aalen, O. O. (1987). Dynamic modelling and causality. *Scandinavian Actuarial Journal*, 1987(3-4):177–190.

Andersen, P. K., Borgan, O., Gill, R. D., and Keiding, N. (2012). *Statistical models based on counting processes*. Springer Science & Business Media.

Bica, I., Alaa, A. M., Jordon, J., and van der Schaar, M. (2020). Estimating counterfactual treatment outcomes over time through adversarially balanced representations. In *International Conference on Learning Representations*.

Chakraborty, B. and Moodie, E. E. (2013). Statistical methods for dynamic treatment regimes. *Springer-Verlag. doi*, 10(978-1):4–1.

Chakraborty, B. and Murphy, S. A. (2014). Dynamic treatment regimes. *Annual review of statistics and its application*, 1:447–464.

Chen, I. Y., Joshi, S., Ghassemi, M., and Ranganath, R. (2021a). Probabilistic machine learning for healthcare. *Annual review of biomedical data science*, 4(1):393–415.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. (2021b). Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097.

De Asis, K., Hernandez-Garcia, J., Holland, G., and Sutton, R. (2018). Multi-step reinforcement learning: A unifying algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Didelez, V. (2008). Graphical models for marked point processes based on local independence. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 70(1):245–264.

Felker, G. M., Lee, K. L., Bull, D. A., Redfield, M. M., Stevenson, L. W., Goldsmith, S. R., LeWinter, M. M., Deswal, A., Rouleau, J. L., Ofili, E. O., et al. (2011). Diuretic strategies in patients with acute decompensated heart failure. *New England Journal of Medicine*, 364(9):797–805.

Fu, J., Norouzi, M., Nachum, O., Tucker, G., Wang, Z., Novikov, A., Yang, M., Zhang, M. R., Chen, Y., Kumar, A., et al. (2021). Benchmarks for deep off-policy evaluation. *arXiv preprint arXiv:2103.16596*.

Geng, C., Paganetti, H., and Grassberger, C. (2017). Prediction of treatment response for combined chemo-and radiation therapy for non-small cell lung cancer patients using a bio-mathematical model. *Scientific reports*, 7(1):13542.

Hallak, A., Tamar, A., Munos, R., and Mannor, S. (2016). Generalized emphatic temporal difference learning: Bias-variance analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Harutyunyan, A., Bellemare, M. G., Stepleton, T., and Munos, R. (2016). Q () with off-policy corrections. In *International Conference on Algorithmic Learning Theory*, pages 305–320. Springer.

Hernan, M. and Robins, J. (2023). *Causal Inference: What If*. Chapman & Hall/CRC Monographs on Statistics & Applied Probab. CRC Press.

Jia, Y. and Zhou, X. Y. (2023). q-learning in continuous time. *Journal of Machine Learning Research*, 24(161):1–61.

Johansson, F., Shalit, U., and Sontag, D. (2016). Learning representations for counterfactual inference. In *International conference on machine learning*, pages 3020–3029. PMLR.

Joshi, S., Urteaga, I., van Amsterdam, W. A., Hripcsak, G., Elias, P., Recht, B., Elhadad, N., Fackler, J., Sendak, M. P., Wiens, J., et al. (2025). Ai as an intervention: improving clinical outcomes relies on a causal approach to ai development and validation. *Journal of the American Medical Informatics Association*, page ocae301.

Le, H., Voloshin, C., and Yue, Y. (2019). Batch policy learning under constraints. In *International Conference on Machine Learning*, pages 3703–3712. PMLR.

Lewis, P. W. and Shedler, G. S. (1979). Simulation of nonhomogeneous poisson processes by thinning. *Naval research logistics quarterly*, 26(3):403–413.

Li, R., Hu, S., Lu, M., Utsumi, Y., Chakraborty, P., Sow, D. M., Madan, P., Li, J., Ghalwash, M., Shahn, Z., and Lehman, L.-w. (2021). G-net: a recurrent network approach to g-computation for counterfactual prediction under a dynamic treatment regime. In Roy, S., Pfohl, S., Rocheteau, E., Tadesse, G. A., Oala, L., Falck, F., Zhou, Y., Shen, L., Zamzmi, G., Mugambi, P., Zirikly, A., McDermott, M. B. A., and Alsentzer, E., editors, *Proceedings of Machine Learning for Health*, volume 158 of *Proceedings of Machine Learning Research*, pages 282–299. PMLR.

Lim, B. (2018). Forecasting treatment responses over time using recurrent marginal structural networks. *Advances in neural information processing systems*, 31.

Lin, H., Scharfstein, D. O., and Rosenheck, R. A. (2004). Analysis of longitudinal data with irregular, outcome-dependent follow-up. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 66(3):791–813.

Lok, J. J. (2008). Statistical modeling of causal effects in continuous time. *The Annals of Statistics*, pages 1464–1507.

Malenica, I. and Murphy, S. (2023). Causality in goal conditioned rl: Return to no future? In *NeurIPS 2023 Workshop on Goal-Conditioned Reinforcement Learning*.

McDermott, M., Nestor, B., Argaw, P., and Kohane, I. (2023). Event stream gpt: A data pre-processing and modeling library for generative, pre-trained transformers over continuous-time sequences of complex events. *arXiv preprint arXiv:2306.11547*.

Melnychuk, V., Frauen, D., and Feuerriegel, S. (2022). Causal transformer for estimating counterfactual outcomes. In *International Conference on Machine Learning*, pages 15293–15329. PMLR.

Munos, R., Stepleton, T., Harutyunyan, A., and Bellemare, M. (2016). Safe and efficient off-policy reinforcement learning. *Advances in neural information processing systems*, 29.

Murphy, S. A. (2005). A generalization error for q-learning. *Journal of Machine Learning Research*.

Nagpal, C., Jeanselme, V., and Dubrawski, A. (2021). Deep parametric time-to-event regression with time-varying covariates. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 184–193. PMLR.

Namkoong, H., Keramati, R., Yadlowsky, S., and Brunskill, E. (2020). Off-policy policy evaluation for sequential decisions under unobserved confounding. *Advances in Neural Information Processing Systems*, 33:18819–18831.

Nie, X., Brunskill, E., and Wager, S. (2021). Learning when-to-treat policies. *Journal of the American Statistical Association*, 116(533):392–409.

Oberst, M. and Sontag, D. (2019). Counterfactual off-policy evaluation with gumbel-max structural causal models. In *International Conference on Machine Learning*, pages 4881–4890. PMLR.

Ogata, Y. (1981). On lewis' simulation method for point processes. *IEEE transactions on information theory*, 27(1):23–31.

Paine, T. L., Paduraru, C., Michi, A., Gulcehre, C., Zolna, K., Novikov, A., Wang, Z., and de Freitas, N. (2020). Hyperparameter selection for offline reinforcement learning. *arXiv preprint arXiv:2007.09055*.

Precup, D., Sutton, R. S., and Singh, S. P. (2000). Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 759–766.

Qu, C., Tan, X., Xue, S., Shi, X., Zhang, J., and Mei, H. (2023). Bellman meets hawkes: Model-based reinforcement learning via temporal point processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 9543–9551.

Robins, J. (1986). 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.

Røysland, K. (2011). A martingale approach to continuous-time marginal structural models. *Bernoulli*, 17(3):895 – 915.

Røysland, K. (2012). Counterfactual analyses with graphical models based on local independence. *The Annals of Statistics*, 40(4):2162 – 2194.

Røysland, K., Ryalen, P., Nygaard, M., and Didelez, V. (2022). Graphical criteria for the identification of marginal causal effects in continuous-time survival and event-history analyses. *arXiv preprint arXiv:2202.02311*.

Rytgaard, H. C., Gerds, T. A., and van der Laan, M. J. (2022). Continuous-time targeted minimum loss-based estimation of intervention-specific mean outcomes. *The Annals of Statistics*, 50(5):2469–2491.

Schulam, P. and Saria, S. (2017). Reliable decision support using counterfactual models. *Advances in neural information processing systems*, 30.

Schweder, T. (1970). Composable markov processes. *Journal of applied probability*, 7(2):400–410.

Seedat, N., Imrie, F., Bellot, A., Qian, Z., and van der Schaar, M. (2022). Continuous-time modeling of counterfactual outcomes using neural controlled differential equations. *arXiv preprint arXiv:2206.08311*.

Snyder, D. L. and Miller, M. I. (2012). *Random point processes in time and space*. Springer Science & Business Media.

Tennenholtz, G., Shalit, U., and Mannor, S. (2020). Off-policy evaluation in partially observable environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10276–10283.

Uehara, M., Shi, C., and Kallus, N. (2022). A review of off-policy evaluation in reinforcement learning. *arXiv preprint arXiv:2212.06355*.

Upadhyay, U., De, A., and Gomez Rodriguez, M. (2018). Deep reinforcement learning of marked temporal point processes. *Advances in neural information processing systems*, 31.

van Amsterdam, W. A., de Jong, P. A., Verhoeff, J. J., Leiner, T., and Ranganath, R. (2024). From algorithms to action: improving patient care requires causality. *BMC Medical Informatics and Decision Making*, 24(1):111.

Van Hasselt, H., Guez, A., and Silver, D. (2016). Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Vanderschueren, T., Curth, A., Verbeke, W., and van der Schaar, M. (2023). Accounting for informative sampling when learning to forecast treatment outcomes over time. *arXiv preprint arXiv:2306.04255*.

Voloshin, C., Le, H. M., Jiang, N., and Yue, Y. (2021). Empirical study of off-policy policy evaluation for reinforcement learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Watkins, C. J. and Dayan, P. (1992). Q-learning. *Machine learning*, 8:279–292.

Xiong, H., Wu, F., Deng, L., Su, M., and Lehman, L.-w. H. (2024). G-transformer: Counterfactual outcome prediction under dynamic and time-varying treatment regimes. *arXiv preprint arXiv:2406.05504*.

Zhang, M., Joffe, M. M., and Small, D. S. (2011). Causal inference for continuous-time processes when covariates are observed only at discrete times. *Annals of statistics*, 39(1).

Zhang, Z., Mei, H., and Xu, Y. (2023). Continuous-time decision transformer for healthcare applications. In *International Conference on Artificial Intelligence and Statistics*, pages 6245–6262. PMLR.