Calibrating Covariate-Dependent Discrete-Time Markov Chain Models of Disease Progression to Prevalence Target Data

Hanie Eskandari[1] Jourdain Lamperski[1*] Mark Roberts[2]
Mary Krauland[2] Praveen Kumar[2] Nisha Nataraj[3]
Michaela Rikard[3]

[1]Department of Industrial Engineering, University of Pittsburgh
[2]School of Public Health, University of Pittsburgh
[3]Division of Overdose Prevention, Centers for Disease Control and Prevention[†]

January 7, 2024

Abstract

Statistical models of individual-level disease progression play an important role in a number of health studies. Typically, these studies use *simulation-based* methods (SBMs) to *calibrate* the parameters of the models, that is, to select parameters so that certain model *outputs* (under the parameters) agree with corresponding *target* data. SBMs provide studies with the freedom to construct complex models and to use diverse sources of target data to calibrate the models. SBMs, however, can require large amounts of computation time, due to the potential need to run many computationally expensive simulations. In this work, we restrict our attention to using *disease prevalence* target data to calibrate a class of discrete-time Markov chain models that have covariate-dependent transition probabilities. Disease prevalence data captures the proportion of individuals in the population who are in given states at given times. We formulate the calibration problem as a (deterministic) non-convex optimization problem and consider solving it with first order methods that just require relatively inexpensive matrix-vector multiplications (instead of simulations). We investigate the performance of our methods through computational experiments and apply them in a case study on Opioid Use Disorder. We show that our method reduced computational barriers to building more geographically accurate models of OUD at the county level, enabling improved decision making by public health practitioners at the state and local level.

*Corresponding author. Email: lamperski@pitt.edu.

[†]The findings and conclusions in this report are those of the authors and do not necessarily represent the official position of the Centers for Disease Control and Prevention.

1 Introduction

Statistical models of individual-level disease progression play an important role in a number of health studies. Typically, these studies model individual-level disease progression as a stochastic process on a finite state space, where each state in the state space captures a different stage of the disease in an individual. Developing these models can help better understand disease progression, evaluate treatment options and other evidence-based interventions, and identify opportunities for prevention and early intervention. As an example, consider Figure 1, which depicts the transition diagram of a Markov chain model of individual-level opioid use disorder (OUD); see Subsection 1.2 for a discussion about the motivation for developing and calibrating models of OUD and a description of the state space.

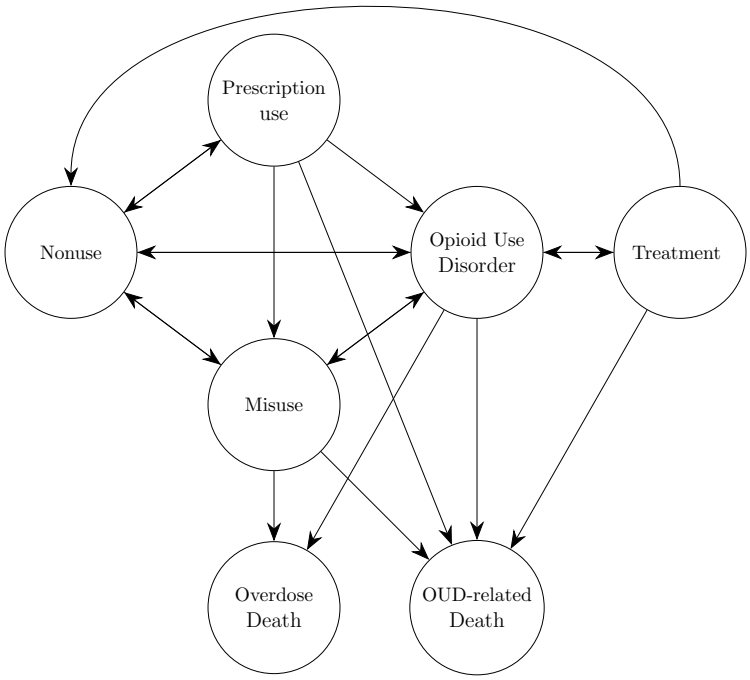

Figure 1: Transition diagram of a Markov chain model of Opioid Use Disorder (OUD). All states have self-transitions that are not illustrated in the diagram.

A number of studies aim to construct complex models that provide a comprehensive depiction of disease progression. These studies often consider non-Markovian models. Many other studies restrict their attention to Markov chain models. Early work in this direction dates back to several studies [11, 32] which assume that the transition rates of continuous-time Markov chain (CTMC) models (or the transition probabilities of discrete-time Markov chain (DTMC) models) do not vary across individuals or with time. In these models, the transition intensities (or transition probabilities) are the model parameters. Other more recent studies assume that some of the transition rates of CTMCs (or the transition probabilities of DTMCs) are parametric functions of covariates; see, for instance, [12, 28, 35]. In these models, the function parameters are model parameters.

The parameters of individual-level disease progression models are either determined from the literature or disease experts, *calibrated* so that model *outputs* agree with available *target* data, or both (e.g., using calibration methods to fine-tune parameter specifications determined from the literature/disease experts). Calibration refers to the adjusting of the model parameters until a certain model output (e.g., the probability that the model is in a given state at a given time) agrees with target data (e.g., the proportion of individuals observed to be in a given state at a given time). Typically, studies use pooled cross-sectional data (from a variety of different sources) as target data. In particular, many studies [16,17,21] use *disease prevalence* data that captures the proportion of individuals that are in given states at given times. Disease prevalence data is often readily available in contrast to panel data that captures the state of each individual in a cohort at certain times; running a clinical trial to obtain panel data can be expensive or infeasible.

Most studies use *simulation-based* methods (SBMs) to calibrate model parameters. SBMs run simulations to estimate model outputs under different parameters. SBMs fall into one of two categories: *empirical* (or *direct search*) methods and *Bayesian* methods. Empirical methods search the parameter space for a point estimate of the parameters that minimizes some form of error between the model outputs and target data. A number of studies employ the Nelder-Mead method [31] to search the parameter space. In optimization parlance, empirical methods can be interpreted as *simulation optimization* methods [8] (or more generally *stochastic zeroth-order* methods). Simulation optimization methods aim to minimize an objective function (in this case the error between the model outputs and target data) using objective function value estimates obtained from simulations. Bayesian methods, on the other hand, construct an estimate of the posterior distribution of the parameters. We direct the reader to [14, 33] for surveys of empirical and Bayesian methods.

In theory SBMs can be used to calibrate a model with any type of target data as long as we can run simulations to estimate the corresponding model output for that target data. Accordingly, in theory SBMs provide studies with the freedom to construct complex models and to use diverse sources of target data to calibrate the models. SBMs, however, can require a large amount of computation time, ultimately limiting the complexity of the models that can be calibrated in practice with these methods. In order to properly search the parameter space, empirical calibration methods must run enough simulations for each set of parameters considered in the search to ensure that estimates of model outputs are close to the true model outputs. Consequently, if there are a large number of targets, or if there is a lot of variance, these methods require a large number of simulations. Furthermore, methods like Nelder-Mead are not guaranteed to converge to a stationary point of the objective function. Bayesian calibration methods typically require an even larger number of simulations due to the fact that they construct an estimate of the entire posterior parameter distribution (instead of just a point estimate of the parameters) [25]. However, some recent studies attempt to address this. For instance, [17] recently proposed using a neural network metamodel of the map from model parameters to targets in the Bayesian calibration instead of using simulations to evaluate the map. However, training a neural network metamodel that is an accurate enough approximation of the map can still require a large number of simulations.

While simulations are needed to compute general outputs of general disease progression models, many studies restrict their attention to calibrating Markov chain models with disease prevalence target data. This begs the question of whether we can leverage the mathemati-

cal structure of Markov chain models to develop more efficient methods for calibrating this class of models using prevalence data? The aim of this paper is to explore this idea. We restrict our attention to calibrating the parameters of a class of DTMC models that we call *covariate-dependent discrete-time Markov chains* (CD-DTMCs). By covariate-dependent, we mean that some or all of the transition probabilities of CD-DTMCs are functions of covariates; we present a formal description of CD-DTMCs along with the calibration problem of interest in Subsection 1.1. We further discuss an application of the calibration problem in a case study on OUD in Subsection 1.2.

An overview of our approach to calibrating the models along with its advantages is as follows. We formulate the calibration problem as a non-convex optimization problem and consider solving it with (deterministic) first order methods (as opposed to stochastic zeroth order methods). Our approach does not require simulations to estimate objective function values (like SBMs), but it instead just requires (inexpensive, relatively speaking) matrix-vector multiplications to explicitly evaluate the objective function (and gradient). We also show that that certain first order methods are guaranteed (under mild conditions) to converge to a stationary point (unlike Nelder-Mead). We present a more detailed description of our approach and contributions in Subsection 1.3.

1.1 CD-DTMC model and calibration problem

CD-DTMC model. Let $S := [n] := \{1, \ldots, n\}$ for $n \in \mathbb{Z}_{>0}$ denote the state space of the DTMC, and let $G = (S, A)$ denote its transition diagram. That is, G is a directed graph, and A is the set of possible transitions. Define $m := |A|$ to be the number of transitions. For instance, Figure 1 depicts the transition diagram of a Markov chain model of OUD that has $n = 7$ states and $m = 26$ transitions.

Let $A_c \subset A$ and $A_f = A \setminus A_c$. We refer to A_c and A_f as the *covariate-dependent* and *fixed transitions*, respectively. Define $m_c := |A_c|$ and $m_f := |A_f|$. For each $ij \in A_c$, the probability that the DTMC transitions to state j at time $t \in \mathbb{Z}_{>0}$ given that the DTMC is in state i at time $t-1$ depends on *covariates* $x_{ij}^{(t)} \in \mathbb{R}^{d_{ij}}$. More precisely, the transition probability is – up to a scaling factor – given by $g_{ij}(\beta_{ij}^\top x_{ij}^{(t)})$, where $\beta_{ij} \in \mathbb{R}^{d_{ij}}$ are *coefficient parameters*, and $g_{ij} : \mathbb{R} \to \mathbb{R}_{>0}$ is a *transition function*. (A scaling factor is needed to ensure that the transition probabilities for transitions leaving a given state sum to 1. Below we present a full description of the transition probabilities that includes the scaling factors.) For each fixed transition $ij \in A_f$, the probability that the DTMC transitions to state j at time t given that the DTMC is in state i at time $t-1$ is – up to a scaling factor – equal to the *fixed transition probability* $\hat{p}_{ij} \in \mathbb{R}_{>0}$.

A full description of the transition probabilities that includes the scaling factors is as follows. We collect the coefficient parameters β_{ij}, $ij \in A_c$ into $\beta \in \mathbb{R}^d$, where $d := \sum_{ij \in A_c} d_{ij}$. For each transition $ij \in A$, we define a *weight function* $w_{ij} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_{>0}$ by

$$w_{ij}(x, \beta) := \begin{cases} g_{ij}(\beta_{ij}^\top x_{ij}) & \text{if } ij \in A_c \\ \hat{p}_{ij} & \text{if } ij \in A_f, \end{cases} \tag{1}$$

where $x_{ij} \in \mathbb{R}^{d_{ij}}$, and $x \in \mathbb{R}^d$ contains x_{ij}, $ij \in A_c$. And we define a *transition probability*

function $p_{ij} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_{>0}$ by

$$p_{ij}(x, \beta) := \frac{w_{ij}(x, \beta)}{\sum_{k \in N^+(i)} w_{ik}(x, \beta)}, \tag{2}$$

where $N^+(i) := \{k \in S : ik \in A\}$ is the set of *out-neighbors* of state i. The probability that the DTMC transitions to state j at time $t \in \mathbb{Z}_{>0}$ given that the DTMC is in state i at time $t-1$ is given by $p_{ij}(x^{(t)}, \beta)$.

Finally, let X_0 be the state of the DTMC at time $t = 0$, and denote the initial distribution of the DTMC by $\hat{p}^{(0)} \in [0, 1]^n$, i.e., $\hat{p}_i^{(0)} := \mathbb{P}(X_0 = i)$ for each $i \in S$.

Calibration to prevalence data. Let $\hat{p}_i^{(t)}$ denote an estimate of the proportion of individuals in the population of interest who are in state i at time $t \in \mathbb{Z}_{>0}$. We refer to $\hat{p}_i^{(t)}$ as a *disease prevalence estimate*. Disease prevalence data $\hat{p}_i^{(t)}$, $(i, t) \in D$, where $D \subset S \times [T]$ for some $T \in \mathbb{Z}_{>0}$, consists of disease prevalence estimates. In health studies, disease prevalence estimates are typically only available for a subset of state-time pairs. In particular, we will see in Subsection 1.2 that this is the case for our OUD case study. Thinking of the disease prevalence estimate $\hat{p}_i^{(t)}$ as a target, the corresponding model output is the (marginal) probability that the DTMC is in state i at time t, namely

$$p_i^{(t)}(x^{(1:t)}, \beta) := \mathbb{P}(X_t(x^{(1:t)}, \beta) = i),$$

where $X_t(x^{(1:t)}, \beta)$ equals the state of the DTMC at time t, and the columns of $x^{(1:t)} \in \mathbb{R}^{d \times t}$ are $x^{(1)}, \ldots, x^{(t)}$.

In this work we consider the calibration problem of determining coefficient parameters $\beta \in \mathbb{R}^d$ such that

$$\hat{p}_i^{(t)} \approx p_i^{(t)}(x^{(1:t)}, \beta), \quad (i, t) \in D.$$

We assume that a modeler has specified a transition diagram G, the covariate-dependent transitions A_c, the fixed transitions A_f, and the transition functions g_{ij}, $ij \in A_c$. We also assume that we have covariate data $x^{(t)}$, $t \in [T]$, fixed transition probabilities $\hat{p}_{ij}$, $ij \in A_f$, an initial distribution $\hat{p}^{(0)}$, and disease prevalence data $\hat{p}_i^{(t)}$, $(i, t) \in D$. In Appendix A we discuss how we determined these quantities for the OUD case study.

1.2 Case study: OUD in the United States at the county level

Opioid overdose deaths continue to remain at high levels in the United States (U.S.), driven primarily by synthetic opioids such as illicitly manufactured fentanyl. Provisional data from the National Center for Health Statistics [7] estimates more than 83,000 opioid-involved overdose deaths in 2022. Data from the 2021 National Survey on Drug Use and Health indicate that approximately 5.6 million people in the U.S. had an OUD in the past year [6]. There is substantial geographic variation in overdose mortality across the U.S. with states such as West Virginia, Delaware, and Kentucky seeing the highest burden of overdose deaths in 2021 [5]. Developing and refining disease progression models of OUD, particularly with increased geographic specificity, can help public health practitioners at the state and local level better understand how the disease progresses in a continually evolving overdose crisis

[29] and inform opportunities for intervention. However, models with increased geographic specificity, such as models at the county level, also have increased data and calibration needs in order to be accurate.

Our goal in this case study is to model how certain covariates impact OUD progression at the county level in the U.S. We consider four county level covariates of interest that were identified by subject matter experts: the annual (i) naloxone (an opioid overdose reversal medication) dispensing rate, (ii) opioid prescribing rate, (iii) proportion of drug seizures with illicitly manufactured fentanyl, and (iv) buprenorphine (a medication approved by the Food and Drug Administration for OUD treatment) prescribing rate. We have data for each of these covariates at the county or state level for years 2007 to 2018; see Appendix A for a detailed description of the data. To model the impact of these covariates, we consider constructing CD-DTMC models of OUD at the county level. Because we seek to construct a model for each of the 3,142 counties in the U.S., it is important to have an efficient calibration method. We restrict our attention to developing methods to calibrate the models (as opposed to also extracting insights from the calibrated models).

Figure 1 depicts the transition diagram of a CD-DTMC model of OUD that was constructed by a panel of experts in addiction medicine and substance use disorder. The model has 7 states and 26 transitions (this includes the self-transitions at each state, not depicted). State Nonuse (NU) represents individuals who do not use opioids; state Prescription Use (PU) represents individuals who use prescription opioids, as prescribed; state Misuse (MU) represents individuals who misuse prescription opioids or use illicit opioids; state Opioid Use Disorder (OUD) represents individuals with OUD; state Treatment (T) represents individuals who are receiving medication for OUD; state Overdose Death (OD) represents people who have died from an opioid-involved overdose, and state OUD-related Death (RD) represents individuals who have died from an OUD-related cause other than an overdose.

We note that the model is a simple version of the full model that we ultimately aim to calibrate. In particular, we intend to consider a non-Markovian model. More specifically, we intend to consider a model in which some of the transition probabilities to some extent depend on the history of the model, instead of just the current state. Of course, naively converting such a model into a CD-DTMC model would result in a state space explosion; we discuss how we plan to handle this in future research in Section 6.

It is assumed that transitions occur on a monthly basis. We will think of X_0 as the state of OUD (or lack thereof) in an individual in some county at the beginning of the year 2007. And for $t > 0$, we will think of $X_t(x^{(1:t)}, \beta)$ as the state of OUD in an individual after t months have passed. Since we have covariate data for 12 years (i.e., 2007 to 2018), we have that $T = 144$ in this case. Recall that we have annual, not monthly, covariate data. The setup in Subsection 1.1 calls for (in this case) monthly covariate data. We will simply use the same annual data to obtain monthly rates for each given year; see Appendix A for a mathematical description of how we process the covariate data.

Observe that the CD-DTMC can transition from state OUD to state OD, capturing the fact that an individual with OUD could overdose and die. The probability of transitioning from state OUD to state OD at time t depends in part on illicitly manufactured fentanyl seizures, which serves as a proxy variable for the drug supply, around time t. We might expect that if a higher proportion of illicitly manufactured fentanyl is being seized, then (all else equal), more individuals with an OUD might be exposed to illicitly manufactured fentanyl,

leading to more transitions from state OUD to state OD. To capture this consideration, we could model the transition from state OUD to state OD as a covariate-dependent transition that depends on the illicitly manufactured fentanyl seizures. For example, we could use a *sigmoid* transition function $g_{\text{OUD,OD}}(z) = \frac{1}{1+\exp(-z)}$ and the weight function

$$w_{\text{OUD,OD}}(x^{(t)}, \beta) = \frac{1}{1 + \exp(-\beta_{\text{OUD,OD},0} - \beta_{\text{OUD,OD},1} x^{(t)}_{\text{OUD,OD},1})},$$

where $x^{(t)}_{\text{OUD,OD},1}$ is the proportion of illicitly manufactured fentanyl seized around time t. We could model the other transitions as covariate-dependent transitions (as well) or as fixed transitions. In the latter case, we set the corresponding fixed transition probabilities to be equal to time-homogeneous estimates of the transition probabilities that were informed by experts; see Table 5 in Appendix A for a description of these estimates.

In the computational experiments in Section 5, we consider three CD-DTMC models that have different covariate-dependent transitions. We summarize the covariate-dependent transitions for each model along with the covariates for each of these transitions in Table 1. Model 1 is the simplest model; it has 1 covariate-dependent transition, namely from state OUD to state OD. Model 2 is slightly more complicated and has 2 covariate-dependent transitions. Model 3 is the most complicated model; it has 3 covariate-dependent transitions. In each model, all covariate-dependent transitions have an intercept coefficient. It follows that in total (taking into account the intercept coefficients), Model 1 has 2 coefficients, Model 2 has 4, and Model 3 has 7. All three models have sigmoid transition functions (as described above).

Model	OUD to OD	NU to PU	OUD to T
1	Proportion of illicitly manufactured fentanyl seized	×	×
2	Proportion of illicitly manufactured fentanyl seized	Opioid prescribing rate	×
3	Proportion of illicitly manufactured fentanyl seized and naloxone dispensing rate	Opioid prescribing rate	Buprenorphine prescribing rate

Table 1: Summary of covariate-dependent transitions along with the covariates for each of these transitions for Model 1, 2, and 3.

We consider calibrating the coefficient parameters β of these three models to agree with disease prevalence targets for states OUD and OD. We have disease prevalence estimates for each of these states at times $t = 12, 24, \ldots, 144$ months; see Appendix A for a more detailed description of this data. It follows that, in this case,

$$D = \{(\text{OUD}, 12t)\}_{t\in[12]} \cup \{(\text{OD}, 12t)\}_{t\in[12]}.$$

Once the parameters β of the models are calibrated, we can use the model to predict the impact of certain covariate values on overdose outcomes.

1.3 Summary of our approach and main contributions

We consider finding coefficient parameters that minimize the sum of squared error between model outputs (the probability that the CD-DTMC is in given states at given times) and disease prevalence targets (estimates of the proportion of people in given states at given times):

$$\min_{\beta \in \mathbb{R}^d} \sum_{(i,t) \in D} \left(p_i^{(t)}(x^{(1:t)}, \beta) - \hat{p}_i^{(t)} \right)^2. \tag{3}$$

Let f denote the objective function of (3), i.e., $f(\beta) := \sum_{(i,t) \in D} \left(p_i^{(t)}(x^{(1:t)}, \beta) - \hat{p}_i^{(t)} \right)^2$.

Transition function assumptions. We study problem (3) under different assumptions about the transition functions g_{ij}, $ij \in A_c$. In Section 2 we discuss these assumptions and present three specific transition functions (the *logistic*, *sigmoid*, and *exponential* functions) that satisfy them.

Properties and evaluation of the objective. In Section 3 we focus on establishing properties of the objective f along with developing an algorithm for evaluating f. First, we provide an analytic formula for the probability $p_i^{(t)}(x^{(1:t)}, \beta)$ that a CD-DTMC model is in state i at time t. The formula generalizes a formula for time-homogeneous DTMCs. We use the analytic formula to show that f is non-convex. The formula also enables us to explicitly compute $p_i^{(t)}(x^{(1:t)}, \beta)$ with matrix-vector multiplications (instead of estimating the quantity with simulations), which in turn enables us to compute $f(\beta)$ (i.e., evaluate the objective). We present an algorithm for evaluating the objective that requires $O(\max\{d, n^2\}T)$ operations and $O(m_c T)$ transition function evaluations; we use the algorithm as a subroutine for solving (3) in the computational experiments in Section 5.

Properties and evaluation of the gradient. In Section 4 we study the gradient ∇f of the objective f. (The gradient is well-defined as long as we assume that the transition functions are differentiable). First, we establish an analytic formula for $\nabla f(\beta)$. Using the formula, we show that the gradient is Lipschitz continuous under certain assumptions about the transition functions that we discuss in Section 2. This result implies that various first order methods are guaranteed to converge to a local minima under mild conditions [22,23,34]. (Nelder-Mead, in contrast, is not even guaranteed to converge to a stationary point.) We also provide an algorithm for computing the gradient that uses matrix-vector multiplications. Running the algorithm requires $O(n \max\{d, n\}T^2)$ operations, $O(m_c T)$ transition function evaluations, and $O(m_c T)$ transition function derivative evaluations. Note that it is more expensive to evaluate the gradient than the objective function. We use the algorithm as a subroutine for solving (3) in computational experiments in Section 5.

Computational experiments. In Section 5 we carry out computational experiments. In particular, we compare the empirical performance of the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm (a first order method) with Nelder-Mead. (In our implementation of Nelder-Mead, we compute objective values with the algorithm that we develop in Section 3,

instead of estimating the objective values with simulations. Consequently, our implementation requires much less computation time.) We demonstrate that both of these methods can be used to calibrate models in a matter of seconds that would otherwise require hours (using SBMs). We also observe that, in almost all of our experiments, BFGS recovers *ground truth parameters*, but Nelder-Mead does not. By ground truth parameters, we mean parameters that we specified and then used to generate disease prevalence data. It is interesting that BFGS frequently recovers ground truth parameters in light of the fact that f is non-convex. Finally, we apply our methods in a case study on Opioid Use Disorder (OUD) and highlight some important practical considerations.

1.4 Related work

Below we discuss two related lines of work that develop special purpose methods for calibrating specific models to specific types of target data.

Multi-state models and panel data. A number of health studies consider *multi-state models* of disease progression. A multi-state state model is a continuous-time stochastic process on a finite state space. We direct the reader to the survey [9] for a review of fundamental multi-state models of disease progression. Recently an R package [15] was developed for estimating the parameters of certain multi-state models from panel data (which recall captures the state of each individual in a cohort at certain times). Certain covariate-dependent multi-state models can be calibrated with the package. The package utilizes maximum likelihood estimation (not SBMs) to estimate the parameters. Of course, the package cannot be used to estimate the parameters from disease prevalence data; SBMs are currently used in these scenarios.

Time-homogeneous DTMCs and aggregate data. The statistics literature considers a few methods for estimating the transition probabilities of time-homogeneous DTMCs from *fully observed* prevalence data (i.e., prevalence data that contains prevalence estimates for every state at every time). These methods are not specifically designed for calibrating disease progression models, and what we call prevalence data, they refer to as *aggregate data*. There are studies that consider using least squares [19, 24, 27, 30], ridge regression [20], maximum likelihood estimation [26], and method of moments [13]. In contrast to our work, however, none of these studies consider *partially observed* prevalence data or DTMC models with transition probabilities that depend on covariates.

1.5 Organization

The remainder of the paper is organized as follows. In Section 2 we discuss transition function related assumptions, and we present three types of transition functions that satisfy them. In Section 3 we present results related to the objective f, and in Section 4 we present results related to the gradient ∇f. We investigate the empirical performance of our methods through computational experiments in Section 5. Finally, in Section 6 we conclude with discussion and discuss future research directions.

2 Transition functions and related assumptions

Throughout we make a number of different assumptions about the transition functions $g_{ij} : \mathbb{R} \to \mathbb{R}_{>0}$, $ij \in A_c$. Here we summarize these assumptions along with where we use them. We also present three types of transition functions that satisfy all of the assumptions.

Transition function assumptions. First, we note that the assumption that the transition functions take on positive values ensures that the weights (1) take on positive values, in turn implying that the transition probabilities (2) are well-defined.

In Section 4 we will see that in order to ensure that the gradient ∇f is well-defined, it is sufficient to assume that the transition functions are differentiable:

Assumption 2.1. *For each $ij \in A_c$, the transition function g_{ij} is differentiable.*

In Section 4 we show that ∇f is Lipschitz continuous under Assumption 2.2 below. The assumption states that the transition functions are twice differentiable and that the absolute values of the first and second derivative values are bounded above by the function values.

Assumption 2.2. *For each $ij \in A_c$, the transition function g_{ij} is twice differentiable. Furthermore, $|g'_{ij}| \leq g_{ij}$ and $|g''_{ij}| \leq g_{ij}$.*

We apply zeroth and first order methods to (3) in our computational study in Section 5. Both of these types of methods require an initial point. Considering transition functions that have inverses will help us to choose a "good" initial point.

Assumption 2.3. *For each $ij \in A_c$, the transition function g_{ij} has an inverse.*

Note that if each transition function is either strictly increasing or strictly decreasing, then Assumption 2.3 holds.

Logistic, sigmoid, and exponential transition functions. Deciding which transition functions to use is ultimately up to the modeler. We provide three types of transition functions that satisfy the above assumptions, namely the *logistic*, *sigmoid*, and *exponential* transition functions. We present a description of each of these transition functions along with their first derivatives, second derivatives, and inverses in Table 2. What we call the sigmoid transition function is often referred to as the logistic function in the literature, and what we call the logistic function is often referred to as the logistic loss function.

The transition functions in Table 2 take on positive values. Furthermore, it is clear that they satisfy Assumption 2.1 and 2.3. In Proposition 2.1 below we verify that they also satisfy Assumption 2.2. We provide a straightforward proof of the proposition in Appendix B.

Proposition 2.1. *If $g : \mathbb{R} \to \mathbb{R}_{>0}$ is either the logistic, sigmoid, or exponential transition function (as defined in Table 2), then $|g'| \leq g$ and $|g''| \leq g$.*

Proof. See Appendix B. □

Transition function	$g(z)$	$g'(z)$	$g''(z)$	$g^{-1}(z)$
Logistic	$\ln(1+\exp(z))$	$\frac{1}{1+\exp(-z)}$	$\frac{\exp(-z)}{(1+\exp(-z))^2}$	$\ln(\exp(z)-1)$
Sigmoid	$\frac{1}{1+\exp(-z)}$	$\frac{\exp(-z)}{(1+\exp(-z))^2}$	$\frac{2\exp(-2z)}{(1+\exp(-z))^3} - \frac{\exp(-z)}{(1+\exp(-z))^2}$	$\ln\left(\frac{z}{1-z}\right)$
Exponential	$\exp(z)$	$\exp(z)$	$\exp(z)$	$\ln(z)$

Table 2: Transition functions $g : \mathbb{R} \to \mathbb{R}_{>0}$ of interest along with their first derivatives g', second derivatives g'', and inverses g^{-1}.

3 Properties and evaluation of the objective

In this section we focus on establishing properties of the objective f and developing an algorithm for evaluating it. Throughout we do not make any assumptions about the transition functions. Accordingly, the algorithm could in theory be used as a subroutine in zeroth-order methods in settings in which the transition functions are not differentiable.

Properties of the objective function. Let $\beta \in \mathbb{R}^d$. First, we present an analytic formula for the probability $p_i^{(t)}(x^{(1:t)}, \beta)$ that the CD-DTMC is in state i at time $t \in \mathbb{Z}_{>0}$. This will in turn provide us with an analytic formula for $f(\beta)$.

For $x \in \mathbb{R}^d$, define the *transition probability* matrix $P(x, \beta) \in \mathbb{R}^{n \times n}$ by

$$[P(x,\beta)]_{ij} = \begin{cases} p_{ij}(x,\beta) & \text{if } ij \in A \\ 0 & \text{otherwise,} \end{cases} \qquad i, j \in S,$$

and let e_i denote the i-th unit vector in $\mathbb{R}^n$. We present the analytic formula in Proposition 3.1 below. The proposition extends the following fact about time-homogeneous DTMCs to the setting of interest. If P and $p^{(0)}$ are the transition probability matrix and initial distribution vector of a time-homogeneous DTMC, respectively, then the probability that the DTMC is in state i at time t equals $e_i^\top (P^\top)^t p^{(0)}$; see, for example, [18].

Proposition 3.1. *For $\beta \in \mathbb{R}^d$ and $(i,t) \in S \times [T]$,*

$$p_i^{(t)}(x^{(1:t)}, \beta) = e_i^\top P(x^{(t)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)}.$$

Proof. See Appendix C. □

From Proposition 3.1 and the definition of f (see Subsection 1.3), we have that

$$f(\beta) = \sum_{(i,t) \in D} \left(e_i^\top P(x^{(t)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)} - \hat{p}_i^{(t)} \right)^2.$$

From this expression for $f(\beta)$ it is not hard to see that f can be non-convex. Example 3.1 below presents a simple instance of the calibration problem in which f is a 1-dimensional non-convex function.

Example 3.1. Suppose that $S = \{1, 2\}$ and $A = \{11, 12, 22\}$. Also suppose that $A_c = \{11\}$ and $d_{11} = 1$. That is, there is one covariate dependent transition (i.e., transition 11) that has one coefficient parameter $\beta_{11,1}$. Further suppose that $x_{11}^{(1)} = 1$, $\hat{p}_{12} = 1$, $g_{11}(z) = \exp(z)$, and $\hat{p}^{(0)} = (1, 0)$. Then

$$\begin{aligned} p_2^{(1)}(x^{(1)}, \beta) &= \hat{p}_1^{(0)} p_{12}^{(1)}(x^{(1)}, \beta) + p_2^{(0)} p_{22}^{(1)}(x^{(1)}, \beta) \\ &= p_{12}^{(1)}(x^{(1)}, \beta) \\ &= \frac{\hat{p}_{12}}{\hat{p}_{12} + g_{11}(\beta_{11,1} x_{11}^{(1)})} \\ &= \frac{1}{1 + \exp(\beta_{11,1})}. \end{aligned}$$

Finally, suppose that $T = 1$, $D = \{(2, 1)\}$, and $\hat{p}_2^{(1)} = 1/2$. Then it follows that

$$f(\beta) = (p_2^{(1)}(x^{(1)}, \beta) - \hat{p}_2^{(1)})^2 = \left(\frac{1}{1 + \exp(\beta_{12,1})} - \frac{1}{2} \right)^2,$$

and hence f is a 1-dimensional non-convex function.

Given that f can be non-convex, it is not clear that local search methods will converge to a globally optimal solution of (3). Note, however, that in the (albeit very simple) instance of Example 3.1, the objective f has a unique stationary point. Through computational experiments in Section 5 we investigate the extent to which local search methods empirically converge to a globally optimal solution.

Evaluating the objective function. Next, we turn our attention to evaluating f. To this end, we consider Algorithm 1. The algorithm exploits the following two facts that are apparent from Proposition 3.1. First, it exploits the fact that the quantities $p_i^{(t)}(x^{(1:t)}, \beta)$, $(i, t) \in D$ depend on similar products of transition probability matrices. Second, it exploits the fact that each of these quantities can be computed with matrix-vector multiplications.

Algorithm 1 Evaluating the objective f at $\beta \in \mathbb{R}^d$

Input: Transition probability matrices $P(x^{(t)}, \beta)$, $t \in [T]$ and initial distribution $\hat{p}^{(0)}$

```
obj ← 0
p^(1) ← P(x^(1), β)^⊤ p̂^(0)
for t = 2, ..., T do
    for i ∈ S such that (i, t) ∈ D do
        obj ← obj + (p_i^(t) − p̂_i^(t))^2
    end for
    p^(t) ← P(x^(t), β)^⊤ p^(t−1)
end for
Return obj
```

Note that Algorithm 1 takes the transition probability matrices $P(x^{(t)}, \beta)$, $t \in [T]$ as input. We review the total computational complexity of computing the transition probability matrices and running Algorithm 1 in Remark 3.1 below.

Remark 3.1. The computational complexity of computing the transition probability matrices $P(x^{(t)}, \beta)$, $t \in [T]$ is as follows. Fix $t \in [T]$. Computing the inner products $\beta_{ij}^\top x_{ij}^{(t)}$, $ij \in A_c$ requires $O(d)$ operations. Given these inner product values, computing the weight values $w_{ij}(x^{(t)}, \beta)$, $ij \in A_c$ requires $O(m_c)$ transition function evaluations. Furthermore, given these weight values, computing the transition probability matrix $P(x^{(t)}, \beta)$ requires $O(m)$ operations. Thus, computing all of the transition probability matrices requires $O(\max\{d, m\}T)$ operations and $O(m_c T)$ transition function evaluations.

Now let us consider the computational complexity of Algorithm 1. Steps 2 and 7 of Algorithm 1 dominate the computational complexity of the algorithm. Both of these steps require $O(n^2)$ operations. Thus Algorithm 1 uses $O(n^2 T)$ operations.

Thus, because $m \leq n^2$, computing the transition probability matrices and running Algorithm 1 in total requires $O(\max\{d, n^2\}T)$ operations and $O(m_c T)$ transition function evaluations.

4 Properties and evaluation of the gradient

In this section we focus on establishing properties of the gradient ∇f and developing an algorithm that primarily uses matrix-vector multiplications to evaluate it. Throughout we assume that the transition functions either satisfy Assumption 2.1 or Assumption 2.2; which assumption we assume to hold will be clear from context.

Let $x, \beta \in \mathbb{R}^d$ and $ij \in A_c$. For $k \in [d_{ij}]$, we write the k-th entry of β_{ij} and x_{ij} as β_{ijk} and x_{ijk}, respectively, and we take $\frac{\partial}{\partial \beta_{ijk}} P(x, \beta)$ to denote the $n \times n$ matrix that contains the partial derivatives (with respect to variable β_{ijk}) of the entries of $P(x, \beta)$; that is, $[\frac{\partial}{\partial \beta_{ijk}} P(x, \beta)]_{uv} = \frac{\partial}{\partial \beta_{ijk}} [P(x, \beta)]_{uv}$ for $u, v \in S$. (The partial derivatives are well-defined under Assumption 2.1.) Also, for a differentiable function $h : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, we collect the partial derivatives $\frac{\partial}{\partial \beta_{ijk}} h(x, \beta)$, $k \in [d_{ij}]$ into $\nabla_{ij} h(x, \beta) \in \mathbb{R}^{d_{ij}}$.

An analytic formula for the gradient. Here we provide a formula for $\nabla f(\beta)$. Let $uv \in A$. First, we provide an expression for the partial derivative $\frac{\partial}{\partial \beta_{ijk}} p_{uv}(x, \beta)$ of the transition probability function p_{uv} evaluated at (x, β).

Proposition 4.1. *Suppose that Assumption 2.1 holds. Let $x, \beta \in \mathbb{R}^d$. For $ij \in A_c$, $k \in [d_{ij}]$, and $uv \in A$,*

$$\frac{\partial}{\partial \beta_{ijk}} p_{uv}(x, \beta) = \begin{cases} \frac{g'_{ij}(\beta_{ij}^\top x_{ij}) \sum_{\ell \in N^+(i) \setminus \{j\}} w_{i\ell}(x, \beta)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x, \beta)\right)^2} x_{ijk} & u = i \text{ and } v = j \\ -\frac{g'_{ij}(\beta_{ij}^\top x_{ij}) w_{iv}(x, \beta)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x, \beta)\right)^2} x_{ijk} & u = i \text{ and } v \in N^+(i) \setminus \{j\} \\ 0 & \text{otherwise.} \end{cases}$$

Proof. The proposition follows from the definition of the transition probabilities (2), the definition of the weights (1), and the quotient and chain rule for derivatives. □

Remark 4.1. From Proposition 4.1, we see that all entries of $\frac{\partial}{\partial \beta_{ijk}} P(x, \beta)$ that are not in the i-th row of $\frac{\partial}{\partial \beta_{ijk}} P(x, \beta)$ are equal to zero.

We make note of the following expression for $\nabla_{ij} p_{uv}(x, \beta)$ that immediately follows from Proposition 4.1.

Corollary 4.1. *Suppose that Assumption 2.1 holds. Let $x, \beta \in \mathbb{R}^d$. For $ij \in A_c$ and $uv \in A$,*

$$\nabla_{ij} p_{uv}(x,\beta) = \begin{cases} \frac{g'_{ij}(\beta_{ij}^\top x_{ij}) \sum_{\ell \in N^+(i) \setminus \{j\}} w_{i\ell}(x,\beta)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)\right)^2} x_{ij} & u = i \text{ and } v = j \\ -\frac{g'_{ij}(\beta_{ij}^\top x_{ij}) w_{iv}(x,\beta)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)\right)^2} x_{ij} & u = i \text{ and } v \in N^+(i) \setminus \{j\} \\ 0 & \text{otherwise.} \end{cases}$$

Proposition 4.2 below establishes a formula for the partial derivatives of the objective function in terms of the partial derivatives of the transition probability functions. Accordingly, the proposition together with Proposition 4.1 provide us with formula for $\nabla f(\beta)$.

Proposition 4.2. *Assume that Assumption 2.1 holds. Let $\beta \in \mathbb{R}^d$. For each $ij \in A_c$ and $k \in [d_{ij}]$,*

$$\frac{\partial}{\partial \beta_{ijk}} f(\beta) = 2 \sum_{(s,t) \in D} (p_s^{(t)}(x^{(1:t)}, \beta) - \hat{p}_s^{(t)}) \frac{\partial}{\partial \beta_{ijk}} p_s^{(t)}(x^{(1:t)}, \beta),$$

where

$$\begin{aligned} &\frac{\partial}{\partial \beta_{ijk}} p_s^{(t)}(x^{(1:t)}, \beta) \\ &= \sum_{\ell=1}^{t} e_s^\top P(x^{(t)}, \beta)^\top \cdots P(x^{(\ell+1)}, \beta)^\top \left[\frac{\partial P(x^{(\ell)}, \beta)}{\partial \beta_{ijk}}\right]^\top P(x^{(\ell-1)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)}. \end{aligned}$$

Proof. The proposition follows from Proposition 3.1 along with the chain and product rule for derivatives. □

Lipschitz continuity of the gradient under Assumption 2.2. The gradient of a differentiable function $h : \mathbb{R}^d \to \mathbb{R}$ is *Lipschitz continuous* if there exists $L \in \mathbb{R}_{>0}$ such that $\|\nabla h(\beta_1) - \nabla h(\beta_2)\| \leq L\|\beta_1 - \beta_2\|$ for all $\beta_1, \beta_2 \in \mathbb{R}^d$. Furthermore, we say that ∇h is *L-Lipschitz continuous*. Whether or not the gradient of the objective of an unconstrained optimization problem is is Lipschitz continuous impacts the convergence properties of first order methods applied to that objective. For instance, if we apply gradient descent (with a sufficiently small constant step-size) to an objective h that has Lipschitz continuous gradient, and if the iterates of gradient descent remain bounded, then gradient descent is guaranteed to converge to a local minimizer of h [22, 23]. We direct the reader to [34] for a more general overview of the role that Lipshitz continuity plays in the convergence of first order methods.

Proposition 4.3 below establishes that the gradient ∇f is $(3|S|^2 T^2 \sum_{t \in [T]} \|x^{(t)}\|_1^2)$-Lipschitz continuous under Assumption 2.2. A sketch of the proof of Lemma 4.3 is as follows. First, using Assumption 2.2, we bound the absolute value of the first-order and second-order partial derivatives of the transition probabilities; see Lemma D.1. This part of the proof requires a good amount of case work. Next, using these bounds, we then bound the second-order partial derivatives of f, which in turn enables us to bound the maximum eigenvalue of the Hessian of f by $6n^3 \sum_{t \in [T]} t^2 \|x^{(t)}\|_1^2$.

Proposition 4.3. *Suppose that Assumption 2.2 holds. Then the gradient ∇f of the objective of (3) is $(6n^3 \sum_{t\in[T]} t^2 \|x^{(t)}\|_1^2)$-Lipschitz continuous.*

Proof. See Appendix D. □

Evaluating the gradient. Next, we turn our attention to evaluating the gradient ∇f with Algorithm 2. From Proposition 4.2, computing $\nabla f(\beta)$ boils down to computing terms of the form $p_s^{(t)}(x^{(1:t)}, \beta)$ and $\frac{\partial}{\partial \beta_{ijk}} p_s^{(t)}(x^{(1:t)}, \beta)$. Naturally we can compute terms of the form $p_s^{(t)}(x^{(1:t)}, \beta)$ with an algorithm similar to Algorithm 1, so let us focus our attention on how Algorithm 2 computes terms of the form $\frac{\partial}{\partial \beta_{ijk}} p_s^{(t)}(x^{(1:t)}, \beta)$. More precisely, we focus on computing terms of the form $\nabla_{ij} p_s^{(t)}(x^{(1:t)}, \beta)$.

Recall that $x, \beta \in \mathbb{R}^d$ and $ij \in A_c$. Define the *multiplier vector* $\alpha_{ij}(x, \beta) \in \mathbb{R}^n$ by

$$[\alpha_{ij}(x,\beta)]_v = \begin{cases} \frac{g'_{ij}(\beta_{ij}^\top x_{ij}) \sum_{\ell \in N^+(i)\setminus\{j\}} w_{i\ell}(x,\beta)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^2} & v = j \\ -\frac{g'_{ij}(\beta_{ij}^\top x_{ij}) w_{iv}(x,\beta)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^2} & v \in N^+(i)\setminus\{j\}\,, \\ 0 & \text{otherwise} \end{cases} \qquad v \in S.$$

From Proposition 4.1, the i-th row of $\frac{\partial}{\partial \beta_{ijk}} P(x,\beta)$ equals $\alpha_{ij}(x,\beta) x_{ijk}$. It follows from Remark 4.1 and Proposition 4.2 that

$$\begin{aligned} &\nabla_{ij} p_s^{(t)}(x^{(1:t)}, \beta) \\ &= \sum_{\ell=1}^{t} \left[P(x^{(t)},\beta)^\top \cdots P(x^{(\ell+1)},\beta)^\top \alpha_{ij}(x^{(\ell)},\beta)\right]_s \left[P(x^{(\ell-1)},\beta)^\top \cdots P(x^{(1)},\beta)^\top p^{(0)}\right]_i x_{ij}^{(\ell)}. \quad (4) \end{aligned}$$

Algorithm 2 maintains a list L_{ij} of vectors, and it also maintains a list R of vectors. Let $t \in [T]$ such that $t > 1$. Consider the t-th iteration of the main for loop in Algorithm 2. Let $\ell \in [t]$. After Step 11 has been completed, the ℓ-th entry in the list L_{ij} is

$$L_{ij}[\ell] = P(x^{(t)},\beta)^\top \cdots P(x^{(\ell+1)},\beta)^\top \alpha_{ij}(x^{(\ell)},\beta),$$

and the ℓ-th entry in the list R is given by

$$R[\ell] = P(x^{(\ell-1)},\beta)^\top \cdots P(x^{(1)},\beta)^\top p^{(0)}.$$

It follows that we compute (4) in Steps 15-17. Also notice that we use matrix-vector multiplication to update list L_{ij} in Steps 6-8, and we use matrix-vector multiplication to update the list R in Step 10.

Algorithm 2 takes the transition probability matrices $P(x^{(t)},\beta)$, $t \in [T]$ and multiplier vectors $\alpha_{ij}(x^{(t)},\beta)$, $(ij,t) \in A_c \times [T]$ as input. We review the total computational complexity of computing these inputs and running Algorithm 2 in Remark 4.2 below. In Remark 4.3 we compare the computational complexity of evaluating the gradient with the computational complexity of evaluating the objective.

Algorithm 2 Evaluating the gradient ∇f at $\beta \in \mathbb{R}^d$

Input: Transition probability matrices $P(x^{(t)}, \beta)$, $t \in [T]$, multiplier vectors $\alpha_{ij}(x^{(t)}, \beta)$, $(ij, t) \in A_c \times [T]$, and initial distribution $\hat{p}^{(0)}$

1: $g \leftarrow 0 \in \mathbb{R}^d$
2: $L_{ij} \leftarrow []$ for each $ij \in A_c$
3: $R \leftarrow [\hat{p}^{(0)}]$
4: **for** $t = 1, \ldots, T$ **do**
5: **for** $t' = 1, \ldots, t-1$ **do**
6: $L_{ij}[t'] = P(x^{(t)}, \beta)^\top L_{ij}[t']$ for each $ij \in A_c$
7: **end for**
8: Append $\alpha_{ij}(x^{(t)}, \beta)$ to L_{ij} for each $ij \in A_c$
9: **if** $t > 1$ **then**
10: Append $P(x^{(t-1)}, \beta)^\top R[t-1]$ to R
11: **end if**
12: $p^{(t)} \leftarrow P(x^{(t)}, \beta) R[t]$
13: **for** $i \in S$ such that $(i, t) \in D$ **do**
14: $\tilde{g} \leftarrow 0 \in \mathbb{R}^d$
15: **for** $t' = 1, \ldots, t$ **do**
16: $\tilde{g}_{ij} \leftarrow \tilde{g}_{ij} + (L_{ij}[t'])_s (R[t'])_i x_{ij}^{(t')}$ for each $ij \in A_c$
17: **end for**
18: $g \leftarrow g + 2(p_s^{(t)} - \hat{p}_s^{(t)})\tilde{g}$
19: **end for**
20: **end for**
21: Return g

Remark 4.2. From Remark 3.1, we already know that computing the transition probability matrices $P(x^{(t)}, \beta)$, $t \in [T]$ requires $O(\max\{d, m\}T)$ operations and $O(m_c T)$ transition function evaluations.

Consider computing the multiplier vectors $\alpha_{ij}(x^{(t)}, \beta)$, $(ij, t) \in A_c \times [T]$. Fix $t \in [T]$. Computing the inner products $\beta_{ij}^\top x_{ij}^{(t)}$, $ij \in A_c$ requires $O(d)$ operations. Given these inner product values, computing the weight values $w_{ij}(x^{(t)}, \beta)$, $ij \in A_c$ requires $O(m_c)$ transition function evaluations. Furthermore, given these weight values along with the inner product values, computing the multiplier vectors $\alpha_{ij}(x^{(t)}, \beta)$, $ij \in A_c$ requires $O(m)$ operations and $O(m_c)$ transition function derivative evaluations. Accordingly, computing the multiplier vectors $\alpha_{ij}(x^{(t)}, \beta)$, $(ij, t) \in A_c \times [T]$ requires $O(\max\{d, m\}T)$ operations, $O(m_c T)$ transition function evaluations, and $O(m_c T)$ transition function derivative evaluations. (If the transition probability matrices and multiplier vectors are computed subsequently, then the inner product and weight calculations of course only need to be done once.)

Now we consider the computational complexity of running Algorithm 2. At the t-th iteration, implementing Steps 6-12 requires $O(n^2 t)$ operations, and implementing Steps 13-18 requires $O(dnt)$ operations. Therefore implementing Algorithm 2 requires $O(n \max\{n, d\}T^2)$ operations.

Thus, running Algorithm 2 and computing its inputs in total requires $O(n \max\{n, d\}T^2)$

operations, $O(m_c T)$ transition function evaluations, and $O(m_c T)$ transition function derivative evaluations.

Remark 4.3. Recall that running Algorithm 1 to compute the objective of Algorithm 3 r requires $O(\max\{d, n^2\}T)$ operations. If $d \leq n^2$, then Algorithm 2 requires a multiplicative factor of $O(T)$ more operations. If $d > n^2$, then Algorithm 2 requires a multiplicative factor of $O(nT)$ more operations.

5 Computational experiments

Our first goal in this computational study is to compare the computational performance of deterministic zeroth order methods and deterministic first order methods for solving problem (3). In all of the methods that we consider, we use Algorithm 1 to evaluate the objective f, and we use Algorithm 2 to evaluate the gradient ∇f. Recall from Section 1 that health studies use stochastic zeroth order methods that require running a large number of simulations to estimate the objective function values. Accordingly, these methods have significantly larger runtimes than the deterministic zeroth order methods that we consider herein (hours as opposed to seconds for problems with only a few parameters). Nonetheless, recall from Remark 4.3 that the gradient is more expensive to evaluate than the objective, so it is still of interest to understand the runtime differences between zeroth and first order methods. We, however, are primarily interested in comparing the extent to which deterministic zeroth and first order methods can recover ground-truth coefficient parameters $\beta^* \in \mathbb{R}^d$. That is, we specify ground-truth coefficient parameters β^*, then generate prevalence estimates $\hat{p}_i^{(t)} = p_i^{(t)}(x^{(1:t)}, \beta^*)$ for each $(i,t) \in D$, then apply zeroth and first order methods to the corresponding instance of (3), and finally observe whether these methods return parameters that are in some sense close to β^*. Recall that Example 3.1 establishes that the objective of (3) can be non-convex, so a priori it is not clear whether the local search methods that we consider will converge to the ground-truth parameters. (It is also not even clear that β^* is the unique optimal solution to (3).) Zeroth-order methods such as Nelder-Mead are not even guaranteed to converge to a stationary point, while Proposition 4.3 implies that first order methods such as gradient descent (with sufficiently small constant step-size) will (under mild conditions) converge to a local minima.

Our second goal is to use the methods developed herein to calibrate OUD models for different U.S. counties. Recall that we ultimately seek to calibrate a model for each of the 3142 U.S. counties. In this case study, we restrict our attention to 14 large counties across the US. We explore how much computation time is required to calibrate the models for these counties, and we investigate how well the calibrated models fit the prevalence data. We also discuss several practical considerations related to implementing the methods. In particular, both zeroth and first order methods require an initial point as input, and the choice of the initial point ultimately impacts the performance of the methods. We explain how this initial point is constructed from expert opinion. We also discuss how to use the methods to assess whether or not parameters are *identifiable* (i.e., whether or not two significantly different sets of parameters generate the same prevalence data).

We consider the following three methods below. (We use the implementation of these methods from the optimize package in the Python SciPy library. As subroutines for the

methods, we use Algorithm 1 to evaluate the objective f, and we use Algorithm 2 to evaluate the gradient ∇f.) Running each of these methods require an initial point. It will be clear from context how we choose this initial point.

1. **BFGS-g.** The algorithm (that we call) BFGS-g is an implementation of the BFGS algorithm. (We call it BFGS-g to distinguish it from the algorithm BFGS-f that we introduce below.) BFGS is a first order method that typically solves moderately-sized problems quite effectively in practice [34]. It is the only first order method that we consider. Throughout we run the algorithm until the norm of the gradient of the current iterate is less than 10^{-30}.

2. **Nelder-Mead.** Nelder-Mead is a zeroth order method. Recall from Subsection 1.4 that various health studies use Nelder-Mead. However, these studies use stochastic estimates of the objective function values instead of the exact values as in this study. Throughout we run the algorithm until the objective function values does not decrease by more than 10^{-30} from one iteration to the next or until the algorithm uses 10^4 functions calls.

3. **BFGS-f.** BFGS-f is an implementation of the BFGS algorithm in which objective values are used to approximate gradient values (via finite differences). It follows that BFGS-f is a zeroth order method. We consider BFGS-f in addition to Nelder-Mead so that we can more directly compare the impact of using derivative values (in BFGS-g) in addition to function values. Throughout we run the algorithm until the objective function values does not decrease by more than 10^{-30}.

In Subsection 5.1 we use these methods to calibrate certain synthetic models and compare their performance. In Subsection 5.2 we apply these methods in the case study on OUD.

Hardware and software. Our computational study is performed on a 2020 Macbook Pro with an M1 chip and 8 GB RAM. Our algorithms are coded using Python, specifically the NumPy and SciPy libraries.

5.1 Calibration of synthetic binary out-tree models

In this section we primarily evaluate the extent to which zeroth and first order methods can recover ground-truth coefficients of synthetic CD-DTMC models that we call *binary out-tree models*. We generate the synthetic calibration instances as follows.

To generate the transition diagram G, we generate a binary out-tree on N states (i.e., all transitions "point away" from the root state) such that states are only missing at the "lowest level." If a state is missing at the lowest level, then so are all of the states to its "right." Next, we add an additional state to the transition diagram that represents death from a cause unrelated to the underlying disease; we add transitions from every non-leaf state in the binary out-tree to the additional state. Finally, we add self-transitions to all of the states. The transition diagram could be thought of as a natural history of disease model for a disease that can progresses in different ways. Note that $n = N+1$. We uniformly at random set K of the outgoing transitions from non-absorbing states to be the covariate-dependent transitions

A_c. We set the other transitions to be the fixed transitions A_f. For each $ij \in A_c$, we set g_{ij} to be the sigmoid transition function; see Section 2. Each covariate-dependent transition has $M+1$ coefficient parameters (including an intercept coefficient), i.e., $d_{ij} = M+1$ for each $ij \in A_c$. For each $ij \in A_c$ and time $t \in [T]$, we set $x_{ij1}^{(t)} = 1$, and we generate $x_{ijk}^{(t)}$ from a standard normal distribution for each $k = 2, \ldots, M+1$. For each transition $ij \in A$, we generate the fixed transition probability $\hat{p}_{ij}$ from the uniform distribution on $[0, 1]$. To construction the initial distribution, we first draw a value $\tilde{p}_i^{(0)}$ from the uniform distribution on $[0, 1]$ for each non-absorbing state i in the transition diagram. For each absorbing state i, we set $\tilde{p}_i^{(0)} = 0$. Finally, for each $i \in S$, we set $\hat{p}_i^{(0)} = \tilde{p}_i^{(0)} / (\sum_{i \in S} \tilde{p}_i^{(0)})$. For each $ij \in A_c$ and $k \in [M+1]$, we generate a ground-truth coefficient parameter β_{ijk}^* from the standard normal distribution. We generate prevalence data using the ground-truth coefficient parameters β^*. Specifically, we set $\hat{p}_i = p_i^{(t)}(x^{(1:t)}, \beta^*)$ for each non-absorbing state $i \in S$ and each time $t \in [T]$. (Mortality data is often available for death states.) Finally, we generate an initial point $\beta^{(0)}$ for the zeroth and first order methods as follows. For each $ij \in A_c$ and $k \in [M+1]$, we draw $\beta_{ijk}^{(0)}$ from the standard normal distribution.

Generating the above requires specifying T, N, K, and M. We use $T = 20$ in all of our experiments. We generate 10 binary out-tree models (and corresponding calibration data) for each $(N, K, M) \in \{5, 10, 15\} \times \{1, 2, 4, 5\} \times \{1, 2\}$. It follows that there are 300 models to calibrate in total (10 for each of the 30 specifications of N, K, and M). We calibrate each of these models with each of the three methods, and we report average (across ten instances) results. Specifically, we report average *distance to ground truth* in Figure 2. By distance to ground truth, we mean the Euclidean distance $\|\tilde{\beta} - \beta^*\|_2$ between the parameters $\tilde{\beta}$ that the methods output and the ground-truth parameters β^*. We report average computation time (in seconds) in Figure 3. We also report average number of objective/function calls in Figure 7 in Appendix E.

We make the following observations:

- From Figure 2, we see that Nelder-Mead recovers ground truth parameters on instances with a smaller number of covariate-dependent transitions, while BFGS-f recovers parameters on instances with a smaller number of coefficient parameters. Albeit BFGS-f does not find parameters that are as close to the ground-truth coefficients as Nelder-Mead on instances with a smaller number of covariate-dependent transitions. BFGS-f in general does not find parameters that are as close to the ground-truth coefficients as BFGS-g because BFGS-f uses approximate gradient values instead of exact gradient values.

- Observe in Figure 2 that in general BFGS-g recovers the ground truth parameters. Furthermore, it in general finds parameters that are closer to the ground-truth parameters than the zeroth order methods. It is interesting that BFGS-g converges to the ground-truth parameters in light of the fact that problem (3) is non-convex. These results highlight the potential importance of considering first order methods in health studies instead of just zeroth order methods.

- From Figure 3, we see that the zeroth order methods require less computation time. This is particularly evident for instances with more coefficient parameters. However,

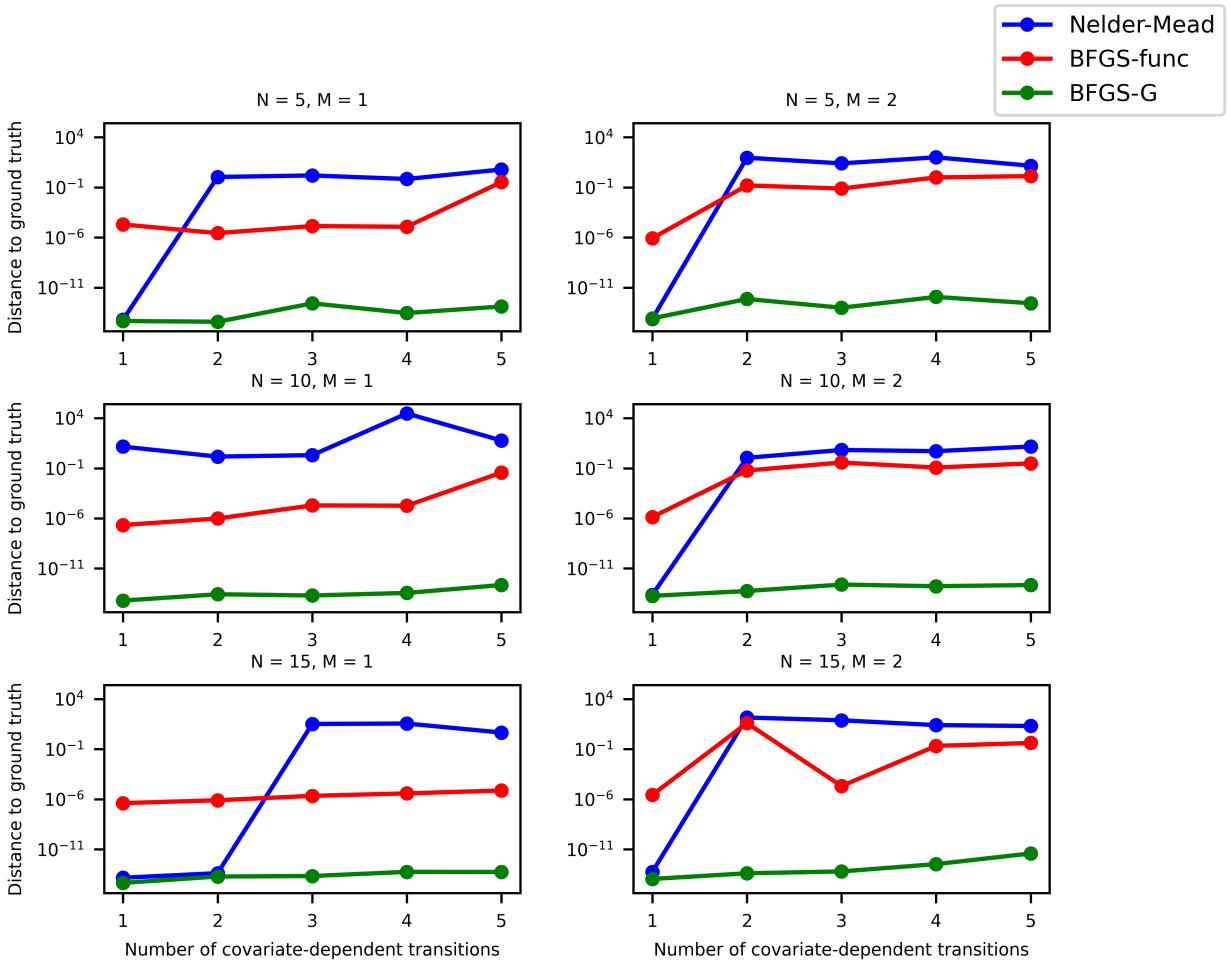

Figure 2: Average distance of method (Nelder-Mead, BFGS-f, and BFGS-g) outputs to ground truth coefficients for the 30 binary out-tree model instances. $N + 1$ equals the number of states of each model, and $M + 1$ equals the number of coefficient parameters for each covariate-dependent transition in each model.

from Figure 7, BFGS-g requires significantly less function and gradient calls. This could in part be explained by the fact that gradient is more expensive to evaluate than the objective; see Remark 4.3. Nonetheless, the additional computation time seems "worth it" in light of the fact that the BFGS-g can recovers ground-truth parameters in more general settings.

5.2 Calibration of OUD case study models

In this subsection we consider calibrating CD-DTMC models of OUD progression that were described in Subsection 1.2.

Parameter identifiability for a specific U.S. county. First, we evaluate the extent to which BFGS-g can recover ground-truth coefficients for a specific U.S. county ("County

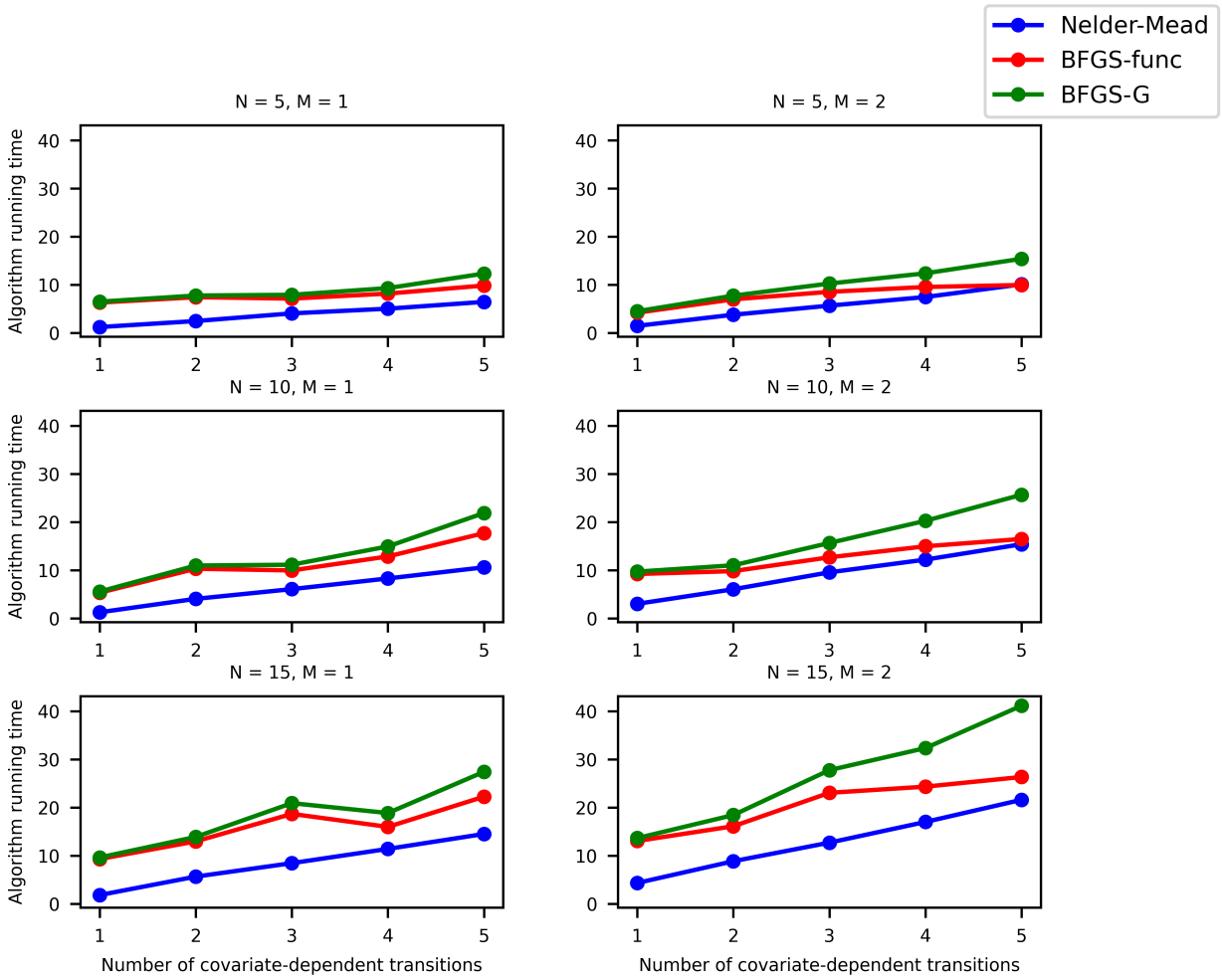

Figure 3: Average computation time (in seconds) of Nelder-Mead, BFGS-f, and BFGS-g for the 30 binary out-tree model instances. $N+1$ equals the number of states of each model, and $M+1$ equals the number of coefficient parameters for each covariate-dependent transition in each model.

A"). These experiments provide insight into whether or not parameters of Model 1, 2, or 3 are identifiable using the available disease prevalence data (assuming that the models are an accurate representation of reality).

We first generate the ground-truth coefficients β^*. For each $ij \in A_c$, we generate the intercept coefficient β^*_{ij1} from the uniform distribution on $[\text{logit}(P_{ij})-1, \text{logit}(P_{ij})+1]$, where P is the $n \times n$ matrix that contains the time-homogeneous estimates of the transition probabilities derived from experts (see Table 5 in Appendix A), and $\text{logit}(z) := \ln\left(\frac{z}{1-z}\right)$ for $z \in (0,1)$. (Note that the logit function is the inverse of the sigmoid function; see Table 2.) For each $k \in 2, \ldots, d_{ij}$, we generate β^*_{ijk} from the uniform distribution on $[-.5, .5]$. This choice of ground truth coefficients ensures that the ground truth transition probabilities (across all times) are not far from the time-homogeneous transition probabilities determined from experts. Next, we use the ground-truth coefficients β^* to generate synthetic disease prevalence data. We consider two scenarios: we compute $\hat{p}_i^{(t)} = p_i^{(t)}(x^{(1:t)}, \beta^*)$ for each $(i,t) \in D$, where D is either given by

$$D = \{(\text{OD}, 12t)\}_{t \in [12]},$$

or D is given by

$$D = \{(\text{OUD}, 12t)\}_{t \in [12]} \cup \{(\text{OD}, 12t)\}_{t \in [12]}.$$

In the first scenario, we have one calibration target (overdose death prevalence), and in the second scenario, we have two targets (overdose death prevalence and OUD prevalence). We use BFGS-g to calibrate Model 1, 2, and 3 under both scenarios to understand how incorporating more disease prevalence data impacts recovery performance; we would expect that more targets would enhance recovery performance.

We apply BFGS-g with a number of different starting points. Because the objective of (3) can be non-convex, it could be critical to choose a starting point that is closer to the ground truth coefficients in order to recover the ground truth coefficients, described below. To investigate this, we generate five different starting points $\beta^{(0)}$, each with a different (expected) distance from the ground-truth coefficients, reflecting varying levels of information that can be obtained from expert opinion or literature. Specifically, for each $ij \in A_c$ and $k \in [d_{ij}]$, we draw $\beta^{(0)}_{ijk}$ from the normal distribution with mean β^*_{ijk} and standard deviation σ. We generate a starting point for each value of $\sigma \in \{0.01, 0.05, 0.1, 0.5, 1\}$. Starting points whose entries are generated from a distribution with a smaller standard deviation σ will tend to be closer to the ground truth coefficients β^*.

Each time the SciPy implementation of BFGS-g calls Algorithm 1 or 2, we record the objective function value at the parameters at which the objective or gradient, respectively, is evaluated. We report the distance to ground truth against the number of function/gradient evaluations for each calibration that we run in Figure 4.

We make the following observations:

- BFGS-g is able to identify the ground truth parameters of Model 1 and 2 from just overdose death prevalence data. However, BFGS-g is not able to identify the parameters of Model 3 from just overdose death prevalence data; recall that Model 3 is the most complicated of the three models. Nonetheless, BFGS-g still moves the initial estimates of the coefficients closer to the ground truth coefficients for all choices of starting points.

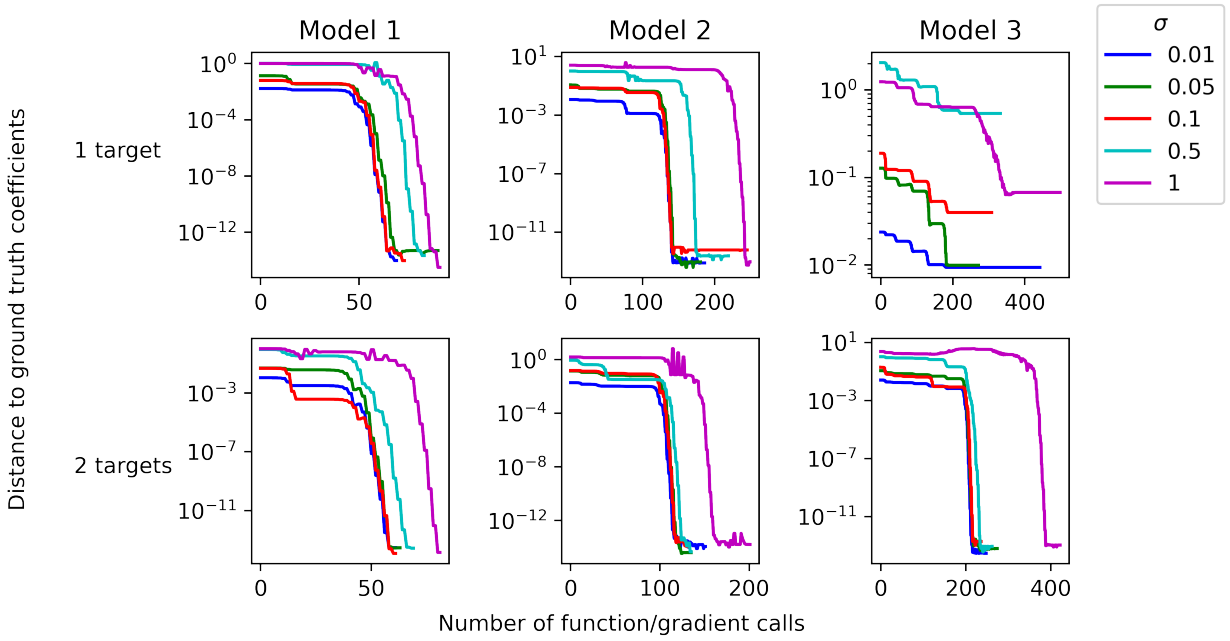

Figure 4: Ground truth recovery performance of BFGS-g applied to calibrating Model 1, 2, and 3 using 1 target (overdose deaths) and 2 targets (overdose deaths and OUD prevalence) for County A. Plots report distance to ground truth coefficients against the number of function/gradient calls.

- The methods tend to converge to the ground truth parameters faster if the initial point is closer. (The starting points are randomly generated, so this observation does not always hold.) Per calibrating Model 3 with 1 target, if the starting point is closer, BFGS-g tends to output parameters closer to the ground truth parameters. Accordingly, in situations in which we do not have a lot of target data, constructing a good initial starting point (e.g., informed by the literature or expert opinion) could be important. In all of the other plots, the starting point does not seem to affect recovery accuracy.

- Aside from the Model 1 calibration results, BFGS-g uses less function calls if more targets are available. Establishing whether or not this is typical behavior would require running more experiments.

Calibration for 14 counties in the United States. Next, we use BFGS-g to calibrate Model 1 and Model 3 for 14 U.S. counties to overdose death prevalence data. That is, we take $D = \{(\text{OD}, 12t)\}_{t\in[12]}$. We choose a starting point $\beta^{(0)}$ informed by experts. Specifically, for each $ij \in A_c$, we use the starting point with intercept coefficient $\beta^{(0)}_{ij1} := \text{logit}(P_{ij})$, and we set $\beta^{(0)}_{ijk} = 0$ for $k = 2, \ldots, d_{ij}$, where recall P is the $n \times n$ matrix that contains time-homogeneous estimates of the transition probabilities (see Table 5 in Appendix A). This choice of $\beta^{(0)}$ ensures that the transition probabilities at all time steps are equal to the time-homogeneous estimates of the transition probabilities.

We report the computation time (in seconds) of BFGS-g in Table 3, we report predicted

County	Model 1	Model 3
A	2.27	15.61
B	2.46	16.09
C	2.48	31.21
D	2.46	16.46
E	2.38	16.43
F	3.07	24.58
G	2.61	16.09
H	2.22	15.44
I	2.59	19.71
J	2.55	23.67
K	2.79	18.56
L	2.73	25.87
M	2.52	21.26
N	3.08	21.21
Average	**2.57**	**20.16**
Standard deviation	**0.26**	**6.94**

Table 3: Time in seconds used by BFGS-g to calibrate Models 1 and 3 for 14 example U.S. counties.

overdose deaths under Model 1 versus actual overdose deaths in Figure 5, and we report predicted overdose deaths under Model 3 versus actual overdose deaths in Figure 6. We make the following observations:

- From Table 3, we see that our method on average requires 20.16 seconds to calibrate Model 3. Thus, forecasting, it would require approximately $(20.16 \times 3142)/86400 = 0.7331$ days (as there are 86400 seconds in a day) to calibrate Model 3 for all 3142 counties.

- From Figure 5, we see that predicted overdose deaths under Model 1 nicely fits actual overdose deaths for some, but not all, counties. This is not too surprising in light of the fact that Model 1 only has 1 covariate-dependent transition. Accordingly, it is of interest to consider more complex models, such as Model 3, to improve fit to actual overdose deaths.

- Figure 6 shows an improvement over Model 1 in terms of fit to actual overdose deaths. However, recall from our previous experiment that we were not able to recover ground truth coefficients of Model 3 with just overdose death prevalence data. Accordingly, it would be worthwhile to explore calibrating Model 3 with both targets (i.e., overdose death prevalence and OUD prevalence) to prevent overfitting.

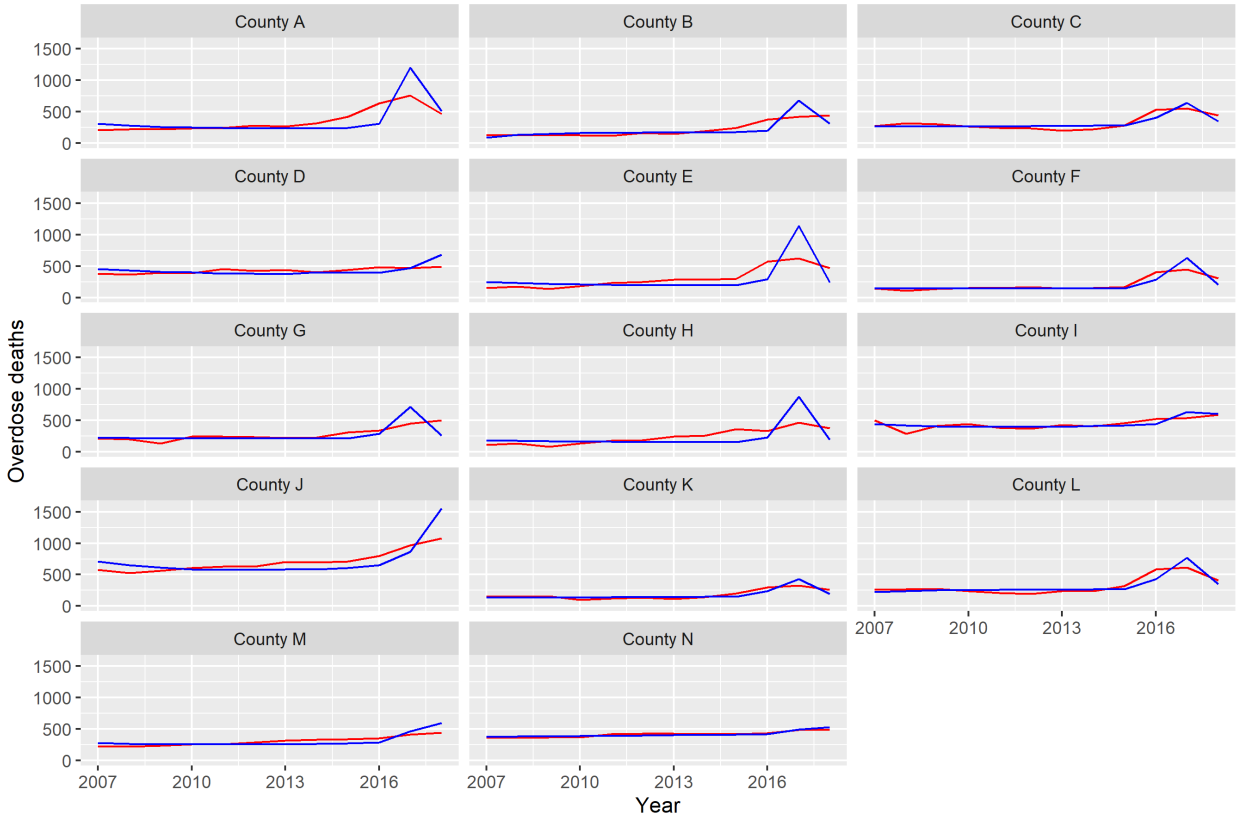

Figure 5: Comparison of predicted overdose deaths under Model 1 (calibrated with overdose death prevalence data) and actual overdose deaths for 14 example counties in the U.S. Blue lines correspond to actual deaths, while red lines correspond to predicted deaths.

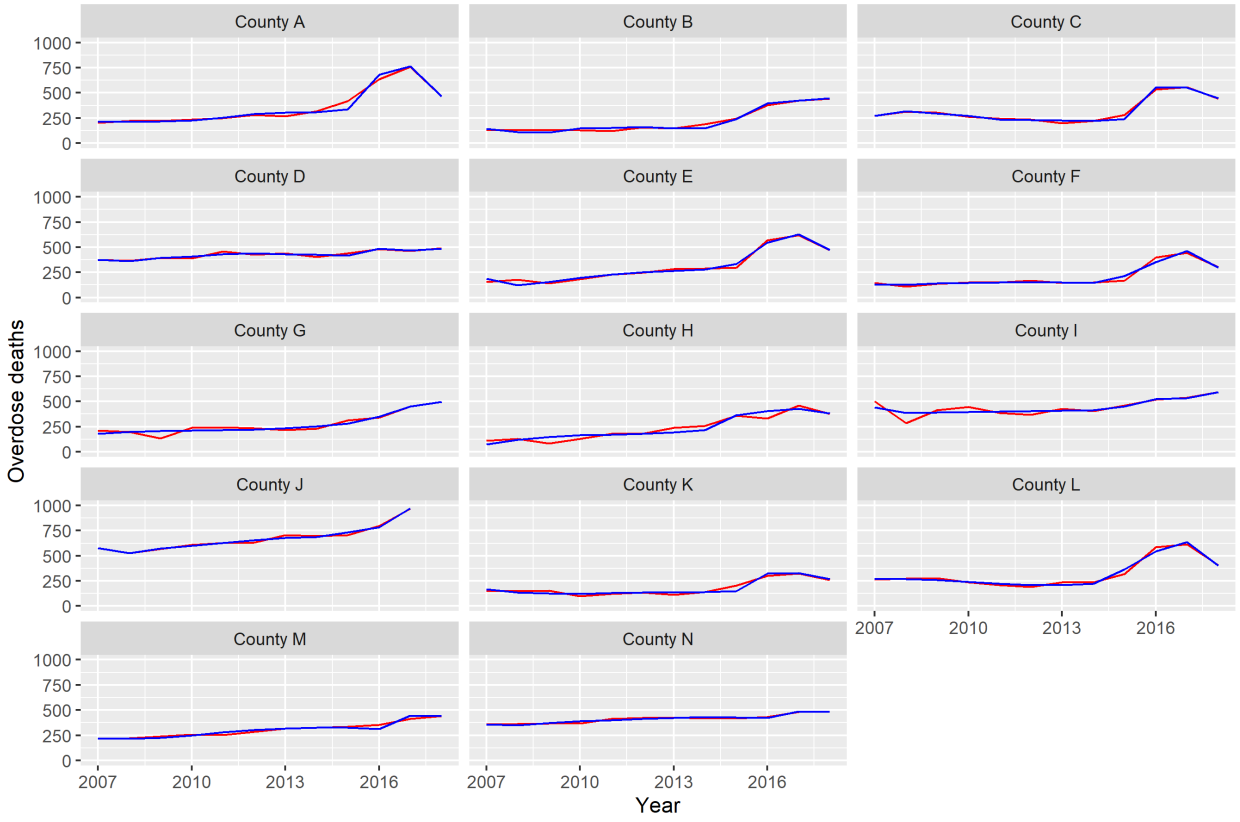

Figure 6: Comparison of predicted overdose deaths under Model 3 (calibrated with overdose death prevalence data) and actual overdose deaths for 14 example counties in the U.S. Blue lines correspond to actual deaths, while red lines correspond to predicted deaths.

6 Discussion

Statistical models of individual-level disease progression play an important role in a number of health studies. Typically, these studies use SBMs to calibrate the parameters of the models. SBMs provide studies with the freedom to construct complex models and to use diverse sources of target data to calibrate the models. SBMs, however, can require large amounts of computation time, due to the potential need to run many computationally expensive simulations. In this work we restricted our attention to using disease prevalence target data to calibrate a class of discrete-time Markov chain models that have covariate-dependent transition probabilities. We formulated the calibration problem as a (deterministic) non-convex optimization problem and considered solving it with (deterministic) first order methods (instead of stochastic zeroth order methods) that just require inexpensive (relatively speaking) matrix-vector multiplications (instead of simulations). We investigated the empirical performance of our zeroth and first order methods through computational experiments and applied them in a case study on OUD. We demonstrated that both of these methods can be used to calibrate models in a matter of seconds that would otherwise require hours (using SBMs). We also observed that, in almost all of our experiments, BFGS was able to recover ground truth parameters. This is an interesting observation in light of the fact that f is non-convex.

Our study has important implications for improved calibration of complex disease progression models. We demonstrated how our approach reduced computational barriers to building more geographically accurate models of OUD at the county level, providing actionable data on the overdose epidemic and thereby enabling improved decision making at the state and local level. More generally, this is an important methodological contribution enabling the continued development of such models to help better understand disease progression, evaluate treatment options and other evidence-based interventions, and more quickly identify opportunities for prevention and early intervention.

Future research directions include:

Related calibration problems. It is of interest to understand to what extent the ideas considered herein can be extended to other model classes and other target data. For instance, can we use these ideas to calibrate CTMC models with disease prevalence data? It is also of interest to extend the methodology to account for other types of commonly used target data, such as, for example, disease incidence target data.

Theoretical properties. Our study (in particular, Proposition 4.3) only touches the surface in regards to establishing theoretical properties of (3) and first order methods for solving (3). It is of interest to establish conditions on the covariate and prevalence data under which (3) has a unique optimal solution. It is also of interest to establish conditions under which certain first order methods converge to a globally optimal solution of (3).

Non-Markovian OUD models. As mentioned in Section 1.2, it is of interest to consider Non-Markovian OUD models. Naively converting such model into a CD-DTMC model would result in a model with a larger state space. This presents a challenge for existing simulation-based calibration methods that scale poorly with the number of states in the model, but as we have observed in Section 5, our model is quite fast and can potentially calibrate models with significantly larger state spaces. Furthermore, we can also introduce constraints and additional variables into optimization problem (3) to incorporate model history.

References

[1] https://www.nflis.deadiversion.usdoj.gov/overview.xhtml.

[2] https://www.iqvia.com/locations/united-states/library/fact-sheets/xponent.

[3] https://www.cdc.gov/nchs/nvss/drug-overdose-deaths.htm.

[4] https://www.census.gov/data/datasets/time-series/demo/popest/2010s-counties-total.html.

[5] *Centers for Disease Control and Prevention. State Unintentional Drug Overdose Reporting System (SUDORS). Atlanta, GA: US Department of Health and Human Services, CDC. Accessed at: https://www.cdc.gov/drugoverdose/fatal/dashboard*, 2023.

[6] Substance Abuse. Mental health services administration (samhsa).(2021). key substance use and mental health indicators in the united states: Results from the 2020 national survey on drug use and health (hhs publication no. pep21-07-01-003, nsduh series h-56). rockville, md: Center for behavioral health statistics and quality. *Center for Behavioral Health Statistics and Quality, Substance Abuse and Mental Health Services Administration. https://www. samhsa. gov/data*, 2022.

[7] Farida B Ahmad, Jodi A. Cisewski, Lauren M Rossen, and Paul Sutton. Provisional drug overdose death counts. *National center for health statistics. Accessed at: https://www.cdc.gov/nchs/nvss/vsrr/drug-overdose-data.html*, 2023.

[8] S Amaran, NV Sahinidis, B Sharda, and SJ Bury. Simulation optimization: a review of algorithms and applications. *Annals of Operations Research*, 240:351–380, 2016.

[9] Per Kragh Andersen and Niels Keiding. Multi-state models for event history analysis. *Statistical methods in medical research*, 11(2):91–115, 2002.

[10] Joshua A Barocas, Laura F White, Jianing Wang, Alexander Y Walley, Marc R LaRochelle, Dana Bernson, Thomas Land, Jake R Morgan, Jeffrey H Samet, and Benjamin P Linas. Estimated prevalence of opioid use disorder in massachusetts, 2011–2015: a capture–recapture analysis. *American Journal of Public Health*, 108(12):1675–1681, 2018.

[11] JR Beck and SG Pauker. The markov process in medical prognosis. *Medical decision making*, 3(4):419–458, 1983.

[12] A Begun, S Morbach, G Rümenapf, and A Icks. Study of disease progression and relevant risk factors in diabetic foot patients using a multistate continuous-time markov chain model. *PLoS One*, 11(1):e0147533, 2016.

[13] Garrett Bernstein and Daniel Sheldon. Consistently estimating markov chains with noisy aggregate data. In *Artificial Intelligence and Statistics*, pages 1142–1150. PMLR, 2016.

[14] CM Hazelbag, J Dushoff, EM Dominic, ZE Mthombothi, and W Delva. Calibration of individual-based models to epidemiological data: a systematic review. *PLoS Computational Biology*, 16(5):1–17, 2020.

[15] C Jackson. Multi-state models for panel data: the msm package for r. *Journal of statistical software*, 38:1–28, 2011.

[16] Christopher H Jackson, Mark Jit, Linda D Sharples, and Daniela De Angelis. Calibration of complex models through bayesian evidence synthesis: a demonstration and tutorial. *Medical decision making*, 35(2):148–161, 2015.

[17] H Jalal, TA Trikalinos, and F Alarid-Escudero. Streamlining bayesian calibration with artificial neural network metamodeling. *Frontiers in Physiology*, 12:1–12, 2021.

[18] Peter Watts Jones and Peter Smith. *Stochastic processes: an introduction*. CRC Press, 2017.

[19] J. D. Kalbfleisch and J. F. Lawless. Least-squares estimation of transition probabilities from aggregate data. *Canadian Journal of Statistics*, 12(3):169–182, 1984.

[20] Inderdeep Kaur and MB Rajarshi. Ridge regression for estimation of transition probabilities from aggregate data. *Communications in Statistics-Simulation and Computation*, 41(4):524–530, 2012.

[21] JJ Kim, KM Kuntz, NK Stout, S Mahmud, LL Villa, EL Franco, and SJ Goldie. Multiparameter calibration of a natural history model of cervical cancer. *American journal of epidemiology*, 166(2):137–150, 2007.

[22] JD Lee, I Panageas, G Piliouras, M Simchowitz, MI Jordan, and B Recht. First-order methods almost always avoid strict saddle points. *Mathematical programming*, 176:311–337, 2019.

[23] JD Lee, M Simchowitz, MI Jordan, and B Recht. Gradient descent only converges to minimizers. In *Conference on learning theory*, pages 1246–1257. PMLR, 2016.

[24] Tsoung-Chao Lee, George G Judge, and Arnold Zellner. Estimating the parameters of the markov probability model from aggregate time series data. 1970.

[25] D Lunn, D Spiegelhalter, A Thomas, and N Best. The bugs project: Evolution, critique and future directions. *Statistics in Medicine*, 28(25):3049–3067, 2009.

[26] Elizabeth Chase MacRae. Estimation of time-varying markov processes with aggregate data. *Econometrica: journal of the Econometric Society*, pages 183–198, 1977.

[27] Albert Madansky. Least squares estimation in finite markov processes. *Psychometrika*, 24(2):137–144, 1959.

[28] G Marshall and RH Jones. Multi-state models and diabetic retinopathy. *Statistics in medicine*, 14(18):1975–1983, 1995.

[29] Christine L Mattson, Lauren J Tanz, Kelly Quinn, Mbabazi Kariisa, Priyam Patel, and Nicole L Davis. Trends and geographic patterns in drug and synthetic opioid overdose deaths—united states, 2013–2019. *Morbidity and Mortality Weekly Report*, 70(6):202, 2021.

[30] George A Miller. Finite markov processes in psychology. *Psychometrika*, 17(2):149–167, 1952.

[31] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[32] FA Sonnenberg and JR Beck. Markov models in medical decision making: a practical guide. *Medical decision making*, 13(4):322–338, 1993.

[33] NK Stout, AB Knudsen, CY Kong, PM McMahon, and GS Gazelle. Calibration methods used in cancer simulation models and suggested reporting guidelines. *Pharmacoeconomics*, 27(7):533–545, 2009.

[34] S Wright and J Nocedal. Numerical optimization. *Springer Science*, 35(67-68):7, 1999.

[35] G Wu, Y Wang, A Yen, J Wong, H Lai, J Warwick, and T Chen. Cost-effectiveness analysis of colorectal cancer screening with stool dna testing in intermediate-incidence countries. *BMC cancer*, 6(1):1–12, 2006.

A OUD case study data

We present a more detailed description of the data that we use in the OUD case study, and we discuss how we process it. We use three *types* of data: covariate, prevalence, and other data. We summarize all of the data that we use in Table 4. We present time-homogeneous estimates of the transition probabilities of the OUD model in Table 5 and estimates of the initial distributions for the 14 counties in Table 6.

Type	Data	Description	Source
Covariate	IMF Seizures	Estimate of the number of times illicitly manufactured Fentanyl was seized in each U.S. state in each year during 2007 to 2018	DEA's National Forensic Laboratory Information System [1]
Covariate	Naloxone Prescribing	Estimate of the number of naloxone prescriptions dispensed at retail pharmacies in each U.S. county in each year during 2007 to 2018	IQVIA Xponent [2]
Covariate	Buprenorphine Prescribing	Estimate of the number of office-based buprenorphine prescriptions in each U.S. county in each year during 2007 to 2018	IQVIA Xponent [2]
Covariate	Opioid Prescribing	Estimate of the number of opioid prescriptions dispensed at retail pharmacies in each U.S. county in each year during 2007 to 2018	IQVIA Xponent [2]
Prevalence	Overdose deaths	Estimate of the number of overdose deaths in each U.S. county in each year during 2007 to 2018	National Vital Statistics System [3]
Prevalence	Opioid Use Disorder	Estimate of the number of individuals who had an opioid use disorder in each U.S. state in each year during 2007 to 2018	National Survey on Drug Use and Health and [10]
Other	Fixed values	Time-homogeneous estimates of the transition probabilities; see Table 5	Experts
Other	Initial distribution	Estimate of number of individuals in each U.S. county who are in each state of the OUD model at the beginning of 2007; see Table 6	Experts
Other	Population	Estimate of number of individuals who live in each U.S. county in each year during 2007 to 2018	U.S. Census Bureau [4]

Table 4: Summary of data used in the OUD case study.

	NU	PU	MU	OUD	T	OD	RD
NU	0.9904	0.0078	0.0010	0.0008	0.0000	0.0000	0.0000
PU	0.8150	0.1021	0.0500	0.0327	0.0000	0.0000	0.0002
MU	0.0390	0.0000	0.9378	0.0230	0.0000	0.0000	0.0002
OUD	0.0162	0.0000	0.0130	0.9597	0.0076	0.0007	0.0028
T	0.0090	0.0000	0.0000	0.0089	0.9819	0.0002	0.0000
OD	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
RD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000

Table 5: Derived estimates of transition probabilities for the OUD model informed by experts. For instance the probability of transitioning from Nonuse (NU) To Prescription Use (PU) is estimated to be .815. Other states are abbreviated as: Misuse (MU), Opioid Use Disorder (OUD), receiving Medication for OUD Treatment (T), Overdose Death (OD), OUD-Related Death (RD).

County	NU	PU	MU	OUD	T	OD	RD
A	0.8873	0.0056	0.0379	0.0646	0.0046	0	0
B	0.8293	0.0082	0.0285	0.0207	0.1133	0	0
C	0.8946	0.0045	0.0311	0.0486	0.0212	0	0
D	0.8750	0.0093	0.0403	0.0653	0.0101	0	0
E	0.8899	0.0059	0.0365	0.0600	0.0077	0	0
F	0.8461	0.0116	0.0384	0.0490	0.0548	0	0
G	0.8668	0.0082	0.0370	0.0547	0.0333	0	0
H	0.8733	0.0075	0.0380	0.0572	0.0240	0	0
I	0.8981	0.0077	0.0346	0.0590	0.0005	0	0
J	0.8838	0.0087	0.0387	0.0644	0.0044	0	0
K	0.9174	0.0028	0.0282	0.0472	0.0045	0	0
L	0.8668	0.0078	0.0308	0.0415	0.0530	0	0
M	0.9006	0.0059	0.0339	0.0570	0.0026	0	0
N	0.8975	0.0042	0.0302	0.0463	0.0218	0	0

Table 6: Derived estimates of the initial distribution for 14 counties informed by experts. States are abbreviated as: Nonuse (NU), Prescription Use (PU), Misuse (MU), Opioid Use Disorder (OUD), receiving medication for OUD Treatment (T), Overdose Death (OD), and OUD-Related Death (RD).

Data preparation. Below we summarize how we prepare the data for a given county in the computational experiments in Subsection 5.2. Let $t \in [12]$, and let $N^{(t)}$ denote the estimate of the number of individuals who live in the county in year $2006 + t$.

- **Covariate data.** Let $\text{nal}^{(t)}$, $\text{trt}^{(t)}$, and $\text{pre}^{(t)}$ denote the estimate of the amounts of naloxone dispensed, buprenorphine prescribed, and opioids dispensed in the county in year $2006 + t$. Also, let $\text{fnt}^{(t)}$ denote the proportion of drug seizures with illicitly manufactured fentanyl in the state that the county belongs to in year $2006 + t$. First,

for each $t \in [12]$, we scale the covariate data to obtain the *proportional* covariate data:

$$\widetilde{\text{nal}}^{(t)} = \frac{\text{nal}^{(t)}}{N^{(t)}}, \quad \widetilde{\text{trt}}^{(t)} = \frac{\text{trt}^{(t)}}{N^{(t)}}, \quad \widetilde{\text{pre}}^{(t)} = \frac{\text{pre}^{(t)}}{N^{(t)}}, \quad \widetilde{\text{fnt}}^{(t)} = \frac{\text{fnt}^{(t)}}{N^{(t)}}.$$

Let us collect $\widetilde{\text{nal}}^{(t)}$, $t \in [12]$ into the vector $\widetilde{\text{nal}} \in \mathbb{R}^{12}$. Define $\widetilde{\text{trt}}, \widetilde{\text{pre}}, \widetilde{\text{fnt}} \in \mathbb{R}^{12}$ similarly. Next, for each $t \in [12]$, we normalize the proportional-covariate data to obtain *normalized proportional* covariate data:

$$\begin{aligned}
\widehat{\text{nal}}^{(t)} &= \frac{\widetilde{\text{nal}}^{(t)} - \mu(\widetilde{\text{nal}})}{\sigma(\widetilde{\text{nal}})}, \\
\widehat{\text{trt}}^{(t)} &= \frac{\widetilde{\text{trt}}^{(t)} - \mu(\widetilde{\text{trt}})}{\sigma(\widetilde{\text{trt}})}, \\
\widehat{\text{pre}}^{(t)} &= \frac{\widetilde{\text{pre}}^{(t)} - \mu(\widetilde{\text{pre}})}{\sigma(\widetilde{\text{pre}})}, \\
\widehat{\text{fnt}}^{(t)} &= \frac{\widetilde{\text{fnt}}^{(t)} - \mu(\widetilde{\text{fnt}})}{\sigma(\widetilde{\text{fnt}})},
\end{aligned}$$

where the functions μ and σ applied to a vector of numbers return the mean and standard deviation of the numbers, respectively. We are done preparing the covariate data. Recall that all of OUD CD-DTMC models require monthly covariate data. We simply use the same annual normalized proportional covariate data for each month in a given year. For instance, in Model 1 in Subsection 5.2, we use the covariate data defined by

$$x^{(t)}_{\text{OUD,OD,2}} = \widehat{\text{fnt}}^{(\lceil t/12 \rceil)}$$

for $t \in [144]$.

- **Prevalence data.** Recall that we use prevalence data $\{\hat{p}^{(12t)}_{\text{OUD}}\}_{t\in[12]} \cup \{\hat{p}^{(12t)}_{\text{OD}}\}_{t\in[12]}$. Let $t \in [12]$, and let $N^{(t)}_{\text{OUD}}$ denote the estimate of the number of individuals in the state that the county belongs to who had an opioid use disorder in year $2006 + t$. We set

$$\hat{p}^{(12t)}_{\text{OUD}} = \frac{N^{(t)}_{\text{OUD}}}{N^{(t)}}.$$

Let $N^{(t)}_{\text{OD}}$ denote the estimate of the number of overdose deaths in the county in year $2006 + t$. Recall that state OD is an absorbing state. Accordingly, we define the prevalence estimates $\hat{p}^{(12t)}_{\text{OD}}$, $t \in [12]$ recursively by

$$\hat{p}^{(12t)}_{\text{OD}} = \begin{cases} \frac{N^{(t)}_{\text{OD}}}{N^{(t)}} & t = 1 \\ \hat{p}^{(12(t-1))}_{\text{OD}} + \frac{N^{(t)}_{\text{OD}}}{N^{(t)}} & t \neq 1. \end{cases}$$

B Proofs for Section 2

Proof of Proposition 2.1. Let $z \in \mathbb{R}$. We show that $|g'(z)| \leq g(z)$ and $|g''(z)| \leq g(z)$. Clearly the desired result holds for the exponential transition function.

Consider the logistic transition function. Because $\ln(z') \geq 1 - \frac{1}{z'}$ for all $z' \in \mathbb{R}$, we see that

$$g(z) = \ln(1 + \exp(z)) \geq 1 - \frac{1}{1 + \exp(z)} = \frac{1}{1 + \exp(-z)} = |g'(z)|. \tag{5}$$

Now observe that

$$|g''(z)| = \frac{\exp(-z)}{(1 + \exp(-z))^2} = \frac{\exp(-z)}{1 + \exp(-z)} \frac{1}{1 + \exp(-z)} \leq \frac{1}{1 + \exp(-z)} = |g'(z)| \leq g(z),$$

where the last inequality follows from (5).

Now consider the sigmoid transition function. From the above argument for the logistic transition function, we see that $|g'(z)| \leq g(z)$. We also have that

$$\begin{aligned}
|g''(z)| &= \left| \frac{2\exp(-2z)}{(1 + \exp(-z))^3} - \frac{\exp(-z)}{(1 + \exp(-z))^2} \right| \\
&= \left| \frac{2\exp(-2z)}{(1 + \exp(-z))^3} - \frac{\exp(-z)}{(1 + \exp(-z))^2} \right| \\
&= \left| \frac{\exp(-z)}{1 + \exp(-z)} \right| \cdot \left| 2\frac{\exp(-z)}{1 + \exp(-z)} - 1 \right| \cdot \left| \frac{1}{1 + \exp(-z)} \right| \\
&\leq \frac{1}{1 + \exp(-z)} \\
&= g(z),
\end{aligned}$$

establishing the desired result. □

C Proofs for Section 3

Proof of Proposition 3.1. The proof proceeds by induction on $t \in [T]$. Observe that for any $j \in S$,

$$\begin{aligned}
p_j^{(1)}(x^{(1)}, \beta) &= \mathbb{P}(X_1(x^{(1)}, \beta) = j) \\
&= \sum_{i \in S} \mathbb{P}(X_1(x^{(1)}, \beta) = j \mid X_0 = i) \mathbb{P}(X_0 = i) \\
&= \sum_{i \in S} p_{ij}(x^{(1)}, \beta) p_i^{(0)} \\
&= [P(x^{(1)}, \beta)^\top p^{(0)}]_j \\
&= e_j^\top P(x^{(1)}, \beta)^\top \hat{p}^{(0)}.
\end{aligned}$$

Suppose that for some $t \in [T-1]$ it holds that

$$p_j^{(t)}(x^{(1:t)}, \beta) = e_j^\top P(x^{(t)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)}, \quad \text{for all } j \in S. \tag{6}$$

We have that

$$\begin{aligned}
p_j^{(t+1)}(x^{(1:t+1)}, \beta) &= \mathbb{P}(X_{t+1}(x^{(1:t+1)}, \beta) = j) \\
&= \sum_{i \in S} \mathbb{P}(X_{t+1}(x^{(1:t+1)}, \beta) = j \mid X_t(x^{(1:t)}, \beta) = i) \mathbb{P}(X_t(x^{(1:t)}, \beta) = i) \\
&= \sum_{i \in S} p_{ij}(x^{(t+1)}, \beta) p_i^{(t)}(x^{(1:t)}, \beta) \\
&= \sum_{i \in S} p_{ij}(x^{(t+1)}, \beta) e_i^\top P(x^{(t)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)} \\
&= \sum_{i \in S} p_{ij}(x^{(t+1)}, \beta) [P(x^{(t)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)}]_i \\
&= [P(x^{(t+1)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)}]_j \\
&= e_j^\top P(x^{(t+1)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)},
\end{aligned}$$

where the fourth equality follows from (6). Thus the desired result follows from induction. □

D Proofs for Section 4

Here we prove Proposition 4.3. Our strategy is to bound the absolute values of the second-order partial derivatives of f from above. First, consider Lemma D.1, which presents upper bounds on the absolute values of the first-order and second-order partial derivatives of the transition probabilities.

Lemma D.1. *Suppose that Assumption 2.2 holds. Let $x, \beta \in \mathbb{R}^d$. The following statements hold.*

1. *For $ij \in A_c$, $uv \in A$, and $k \in [d_{ij}]$,*

$$\left| \frac{\partial}{\partial \beta_{ijk}} p_{uv}(x, \beta) \right| \leq |x_{ijk}|.$$

2. *For $ij, pq \in A_c$, $uv \in A$, $k \in [d_{ij}]$, and $r \in [d_{pq}]$,*

$$\left| \frac{\partial^2}{\partial \beta_{pqr} \partial \beta_{ijk}} p_{uv}(x, \beta) \right| \leq 2|x_{ijk}||x_{pqr}|.$$

Proof. First, we establish the first statement of the lemma. From Proposition 4.1, there are two cases to consider:

Case 1. Suppose that $u = i$ and $v = j$. Then

$$\left| \frac{\partial}{\partial \beta_{ijk}} p_{uv}(x, \beta) \right| = \left| \frac{g'_{ij}(\beta_{ij}^\top x_{ij}) \sum_{\ell \in N^+(i) \setminus \{j\}} w_{i\ell}(x, \beta)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x, \beta) \right)^2} x_{ijk} \right|$$

$$\leq \left| \frac{w_{ij}(x,\beta)}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} \right| \left| \frac{\sum_{\ell \in N^+(i)\setminus\{j\}} w_{i\ell}(x,\beta)}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} \right| |x_{ijk}|$$
$$\leq |x_{ijk}|,$$

where the first inequality follows from $|g'_{ij}(\beta_{ij}^\top x_{ij})| \leq |g_{ij}(\beta_{ij}^\top x_{ij})| = |w_{ij}(x,\beta)|$.

Case 2. Suppose that $u = i$ and $v \in N^+(i) \setminus \{j\}$. Then

$$\left| \frac{\partial}{\partial \beta_{ijk}} p_{uv}(x,\beta) \right| = \left| \frac{g'_{ij}(\beta_{ij}^\top x_{ij}) w_{iv}(x,\beta)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta) \right)^2} x_{ijk} \right|$$
$$\leq \left| \frac{w_{ij}(x,\beta)}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} \right| \left| \frac{w_{iv}(x,\beta)}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} \right| |x_{ijk}|$$
$$\leq |x_{ijk}|,$$

where the first inequality follows from $|g'_{ij}(\beta_{ij}^\top x_{ij})| \leq |g_{ij}(\beta_{ij}^\top x_{ij})| = |w_{ij}(x,\beta)|$.

Now we establish the second statement of the lemma. From Proposition 4.1, if either $u \neq i$ or $u \neq p$, then $\frac{\partial^2}{\partial \beta_{pqr} \partial \beta_{ijk}} p_{uv}(x,\beta) = 0$, implying that the desired result holds. Accordingly, suppose that $u = i = p$. Below we consider five cases.

Case 1. Suppose that $v = j = q$. From Proposition 4.1 together with the chain and quotient rule for derivatives,

$$\left| \frac{\partial^2}{\partial \beta_{pqr} \partial \beta_{ijk}} p_{uv}(x,\beta) \right|$$
$$= \left| \frac{\left(\sum_{\ell \in N^+(i)\setminus\{j\}} w_{i\ell}(\beta,x) \right) \left(g''_{ij}(\beta_{ij}^\top x_{ij}) \sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta) - 2g'_{ij}(\beta_{ij}^\top x_{ij})^2 \right)}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta) \right)^3} x_{ijk} x_{ijr} \right|$$
$$= \left| \frac{\sum_{\ell \in N^+(i)\setminus\{j\}} w_{i\ell}(x,\beta)}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} \right| \left| \frac{g''_{ij}(\beta_{ij}^\top x_{ij})}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} - \frac{2g'_{ij}(\beta_{ij}^\top x_{ij})^2}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta) \right)^2} \right| |x_{ijk}||x_{ijr}|$$
$$\leq \left(\left| \frac{w_{ij}(x,\beta)}{\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta)} \right| + \left| \frac{2w_{ij}(x,\beta)^2}{\left(\sum_{\ell \in N^+(i)} w_{i\ell}(x,\beta) \right)^2} \right| \right) |x_{ijk}||x_{ijr}|$$
$$\leq 2|x_{ijk}||x_{ijr}|,$$

where the first inequality follows from the triangle inequality, $|g'_{ij}(\beta_{ij}^\top x_{ij})| \leq |w_{ij}(x,\beta)|$, and $|g''_{ij}(\beta_{ij}^\top x_{ij})| \leq |w_{ij}(x,\beta)|$.

Case 2. Suppose that $v \neq j = q$. From Proposition 4.1 together with the chain and quotient rule for derivatives,

$$\left| \frac{\partial^2}{\partial \beta_{pqr} \partial \beta_{ijk}} p_{uv}(x,\beta) \right|$$

$$
\begin{aligned}
&= \left| \frac{w_{iv}(x,\beta)\left(2g'_{ij}(\beta_{ij}^\top x_{ij})^2 - g''_{ij}(\beta_{ij}^\top x_{ij})\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^3} \right| |x_{ijk}||x_{ijr}| \\
&= \left| \frac{w_{iv}(x,\beta)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)} \right| \left| \frac{2g'_{ij}(\beta_{ij}^\top x_{ij})^2}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^2} - \frac{g''_{ij}(\beta_{ij}^\top x_{ij})}{\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)} \right| |x_{ijk}||x_{ijr}| \\
&\leq \left(\left| \frac{2w_{ij}(x,\beta)^2}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^2} \right| + \left| \frac{w_{ij}(x,\beta)}{\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)} \right| \right) |x_{ijk}||x_{ijr}| \\
&\leq 2|x_{ijk}||x_{ijr}|,
\end{aligned}
$$

where the first inequality follows from the triangle inequality, $|g'_{ij}(\beta_{ij}^\top x_{ij})| \leq |w_{ij}(x,\beta)|$, and $|g''_{ij}(\beta_{ij}^\top x_{ij})| \leq |w_{ij}(x,\beta)|$.

Case 3. Suppose that $v = j \neq q$. From Proposition 4.1 together with the chain and quotient rule for derivatives,

$$
\begin{aligned}
&\left| \frac{\partial^2}{\partial\beta_{pqr}\partial\beta_{ijk}} p_{uv}(x,\beta) \right| \\
&= \left| \frac{\left(w_{ij}(x,\beta) - \sum_{\ell\in N^+(i)\setminus\{j\}} w_{i\ell}(x,\beta)\right) g'_{ij}(\beta_{ij}^\top x_{ij}) g'_{iq}(\beta_{iq}^\top x_{iq})}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^3} \right| |x_{ijk}||x_{iqr}| \\
&\leq \left(\left| \frac{w_{ij}(x,\beta)w_{ij}(x,\beta)w_{iq}(x,\beta)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^3} \right| + \left| \frac{w_{ij}(x,\beta)w_{iq}(x,\beta)\sum_{\ell\in N^+(i)\setminus\{j\}} w_{i\ell}(x,\beta)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^3} \right| \right) |x_{ijk}||x_{iqr}| \\
&\leq 2|x_{ijk}||x_{iqr}|
\end{aligned}
$$

where the first inequality follows from the triangle inequality, $|g'_{ij}(\beta_{ij}^\top x_{ij})| \leq |w_{ij}(x,\beta)|$, and $|g'_{iq}(\beta_{iq}^\top x_{iq})| \leq |w_{iq}(x,\beta)|$.

Case 4. Suppose that $v = q \neq j$. The desired result follows from an identical argument to the one used in **Case 3**.

Case 5. Suppose that $v \neq j \neq p$. From Proposition 4.1 together with the chain rule for derivatives,

$$
\begin{aligned}
\left| \frac{\partial^2}{\partial\beta_{pqr}\partial\beta_{ijk}} p_{uv}(x,\beta) \right| &= \left| \frac{2w_{iv}(x,\beta)g'_{ij}(\beta_{ij}^\top x_{ij})g'_{iq}(\beta_{iq}^\top x_{iq})}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^3} \right| |x_{ijk}||x_{iqr}| \\
&\leq \left| \frac{2w_{iv}(x,\beta)w_{ij}(x,\beta)w_{iq}(x,\beta)}{\left(\sum_{\ell\in N^+(i)} w_{i\ell}(x,\beta)\right)^3} \right| |x_{ijk}||x_{iqr}| \\
&\leq 2|x_{ijk}||x_{iqr}|.
\end{aligned}
$$

where the first inequality follows from $|g'_{ij}(\beta_{ij}^\top x_{ij})| \le |w_{ij}(x,\beta)|$ and $|g'_{iq}(\beta_{iq}^\top x_{iq})| \le |w_{iq}(x,\beta)|$. □

We are now prepared to prove Proposition 4.3.

Proof of Proposition 4.3. Let $\beta \in \mathbb{R}^d$. Also let $ij, pq \in A_c$, $uv \in A$, $k \in [d_{ij}]$, and $r \in [d_{pq}]$. From Proposition 4.2 and the product rule for derivatives,

$$\begin{aligned}&\frac{\partial^2}{\partial\beta_{pqr}\partial\beta_{ijk}} f(\beta)\\ &= 2\sum_{(s,t)\in D} \frac{\partial}{\partial\beta_{pqr}} p_s^{(t)}(x^{(1:t)},\beta)\frac{\partial}{\partial\beta_{ijk}} p_s^{(t)}(x^{(1:t)},\beta) + (p_s^{(t)}(x^{(1:t)},\beta) - \hat{p}_s^{(t)})\frac{\partial^2}{\partial\beta_{pqr}\partial\beta_{ijk}} p_s^{(t)}(x^{(1:t)},\beta).\end{aligned} \tag{7}$$

Let $(s,t) \in D$. We claim that it is sufficient to show:

$$\left|\frac{\partial}{\partial\beta_{ijk}} p_s^{(t)}(x^{(1:t)},\beta)\right| \le tn|x_{ijk}^{(t)}| \tag{8}$$

$$\left|\frac{\partial^2}{\partial\beta_{pqr}\partial\beta_{ijk}} p_s^{(t)}(x^{(1:t)},\beta)\right| \le 2t^2n^2|x_{ijk}^{(t)}||x_{pqr}^{(t)}|. \tag{9}$$

To see this, observe that if (8) and (9) hold, then from (7) and the triangle inequality,

$$\begin{aligned}\left|\frac{\partial^2}{\partial\beta_{pqr}\partial\beta_{ijk}} f(\beta)\right| &\le 2\sum_{(s,t)\in D} t^2n^2|x_{ijk}^{(t)}||x_{pqr}^{(t)}| + |p_s^{(t)}(x^{(1:t)},\beta) - \hat{p}_s^{(t)}|2t^2n^2|x_{ijk}^{(t)}||x_{pqr}^{(t)}|\\ &\le 6n^2\sum_{(s,t)\in D} t^2|x_{ijk}^{(t)}||x_{pqr}^{(t)}|\\ &\le 6n^3\sum_{t\in[T]} t^2|x_{ijk}^{(t)}||x_{pqr}^{(t)}|,\end{aligned}$$

where the second inequality follows from $|p_s^{(t)}(x^{(1:t)},\beta) - \hat{p}_s^{(t)}| \le 1$. Let $H_f(\beta)$ denote the $d \times d$ hessian matrix of f evaluated at $\beta \in \mathbb{R}^d$. Observe that

$$\begin{aligned}\max_{x\in\mathbb{R}^d:\|x\|=1} x^\top H_f(\beta)^\top x &\le \sum_{ij\in A_c}\sum_{k\in[n_{ij}]}\sum_{pq\in A_c}\sum_{r\in[n_{pq}]}\left|\frac{\partial^2}{\partial\beta_{ijk}\beta_{pqr}} f(\beta)\right|\\ &\le \sum_{ij\in A_c}\sum_{k\in[n_{ij}]}\sum_{pq\in A_c}\sum_{r\in[n_{pq}]}\sum_{t\in[T]} 6n^3t^2|x_{ijk}^{(t)}||x_{pqr}^{(t)}|\\ &= 6n^3\sum_{t\in[T]} t^2\sum_{ij\in A_c}\sum_{k\in[n_{ij}]}\sum_{pq\in A_c}\sum_{r\in[n_{pq}]}|x_{ijk}^{(t)}||x_{pqr}^{(t)}|\\ &= 6n^3\sum_{t\in[T]} t^2\|x^{(t)}\|_1^2,\end{aligned}$$

and hence f is $(6n^3\sum_{t\in[T]} t^2\|x^{(t)}\|_1^2)$-smooth.

First, we establish (8). From Proposition 4.2, we have that $\frac{\partial}{\partial \beta_{ijk}} p_s^{(t)}(x^{(1:t)}, \beta) = \sum_{\ell=1}^{t} \theta_\ell$, where

$$\theta_\ell := e_s^\top P(x^{(t)}, \beta)^\top \cdots P(x^{(\ell+1)}, \beta)^\top \left[\frac{\partial P(x^{(\ell)}, \beta)}{\partial \beta_{ijk}}\right]^\top P(x^{(\ell-1)}, \beta)^\top \cdots P(x^{(1)}, \beta)^\top \hat{p}^{(0)}$$

for each $\ell \in [t]$, so it is sufficient to show that $|\theta_\ell| \leq n|x_{ijk}^{(t)}|$. It follows that for each $\ell \in [t]$, there are $n \times n$ *stochastic matrices* $Q_1^{(\ell)}$, $Q_2^{(\ell)}$ (i.e., matrices whose columns are probability vectors) and an $n \times n$ matrix $R_1^{(\ell)}$ such that $\theta_\ell = e_s^\top Q_1^{(\ell)} R_1^{(\ell)} Q_2^{(\ell)} \hat{p}^{(0)}$. By Remark 4.1 and Lemma D.1, the absolute value of each entry in the i-th column of $R_1^{(\ell)}$ is bounded above by $|x_{ijk}^{(t)}|$, and the entries in all other columns are equal to 0. Because $Q_2^{(\ell)} \hat{p}^{(0)}$ is a probability vector, the absolute value of each entry of $R_1^{(\ell)} Q_2^{(\ell)} \hat{p}^{(0)}$ is bounded above by $|x_{ijk}^{(t)}|$. Hence the absolute value of each entry of $Q_1^{(\ell)} R_1^{(\ell)} Q_2^{(\ell)} \hat{p}^{(0)}$ is bounded above by $n|x_{ijk}^{(t)}|$, so $|\theta_\ell| = |e_s^\top Q_1^{(\ell)} R_1^{(\ell)} Q_2^{(\ell)} \hat{p}^{(0)}| \leq n|x_{ijk}^{(t)}|$.

Next, we establish (9). It follows from Proposition 3.1 and 4.2 together with the product rule for derivatives and induction that

$$\frac{\partial^2}{\partial \beta_{pqr} \partial \beta_{ijk}} p_s^{(t)}(x^{(1:t)}, \beta) = \sum_{\ell, v \in [t]} \phi_{\ell v s}, \tag{10}$$

where for $\ell < v$,

$$\begin{aligned}
&\phi_{\ell v s} \\
&:= e_s^\top \left[\prod_{\substack{u \in [t]: \\ u > v}} P(x^{(u)}, \beta)^\top\right] \frac{\partial P(x^{(v)}, \beta)}{\partial \beta_{pqr}}^\top \left[\prod_{\substack{u \in [t]: \\ \ell < u < v}} P(x^{(u)}, \beta)^\top\right] \frac{\partial P(x^{(\ell)}, \beta)}{\partial \beta_{ijk}}^\top \left[\prod_{\substack{u \in [t]: \\ u < \ell}} P(x^{(u)}, \beta)^\top\right] \hat{p}^{(0)},
\end{aligned}$$

for $\ell = v$,

$$\phi_{\ell v s} := e_s^\top \left[\prod_{\substack{u \in [t]: \\ u > v}} P(x^{(u)}, \beta)^\top\right] \left[\frac{\partial^2}{\partial \beta_{pqr} \partial \beta_{ijk}} P(x^{(\ell)}, \beta)\right]^\top \left[\prod_{\substack{u \in [t]: \\ u < \ell}} P(x^{(u)}, \beta)^\top\right] \hat{p}^{(0)},$$

and for $\ell > v$,

$$\begin{aligned}
&\phi_{\ell v s} \\
&:= e_s^\top \left[\prod_{\substack{u \in [t]: \\ u > \ell}} P(x^{(u)}, \beta)^\top\right] \frac{\partial P(x^{(\ell)}, \beta)}{\partial \beta_{pqr}}^\top \left[\prod_{\substack{u \in [t]: \\ v < u < \ell}} P(x^{(u)}, \beta)^\top\right] \frac{\partial P(x^{(v)}, \beta)}{\partial \beta_{ijk}}^\top \left[\prod_{\substack{u \in [t]: \\ u < v}} P(x^{(u)}, \beta)^\top\right] \hat{p}^{(0)}.
\end{aligned}$$

It is sufficient to show that $|\phi_{\ell v s}| \leq n^2 |x_{ijk}^{(t)}||x_{pqr}^{(t)}|$. We consider two cases:

Case 1. Suppose that $\ell < v$ or $\ell > v$. Then from above there are $n \times n$ stochastic matrices $Q_1^{(\ell)}, Q_2^{(\ell)}, Q_3^{(\ell)}$ and $n \times n$ matrices $R_1^{(\ell)}, R_2^{(\ell)}$ such that $\phi_{\ell v s} = e_s Q_1^{(\ell)} R_1^{(\ell)} Q_2^{(\ell)} R_2^{(\ell)} Q_3^{(\ell)} \hat{p}^{(0)}$. Inequality (9) then follows from a similar argument to the one used above to establish (8).

Case 2. Suppose that $\ell = v$. Then from above there are $n \times n$ stochastic matrices $Q_1^{(\ell)}, Q_2^{(\ell)}$ and an $n \times n$ matrix $R_1^{(\ell)}$ such that $\phi_{\ell vs} = e_s Q_1^{(\ell)} R_1^{(\ell)} Q_2^{(\ell)} \hat{p}^{(0)}$. Inequality (9) then follows from a similar argument to the one used above to establish (8). □

E Additional figures for Section 5

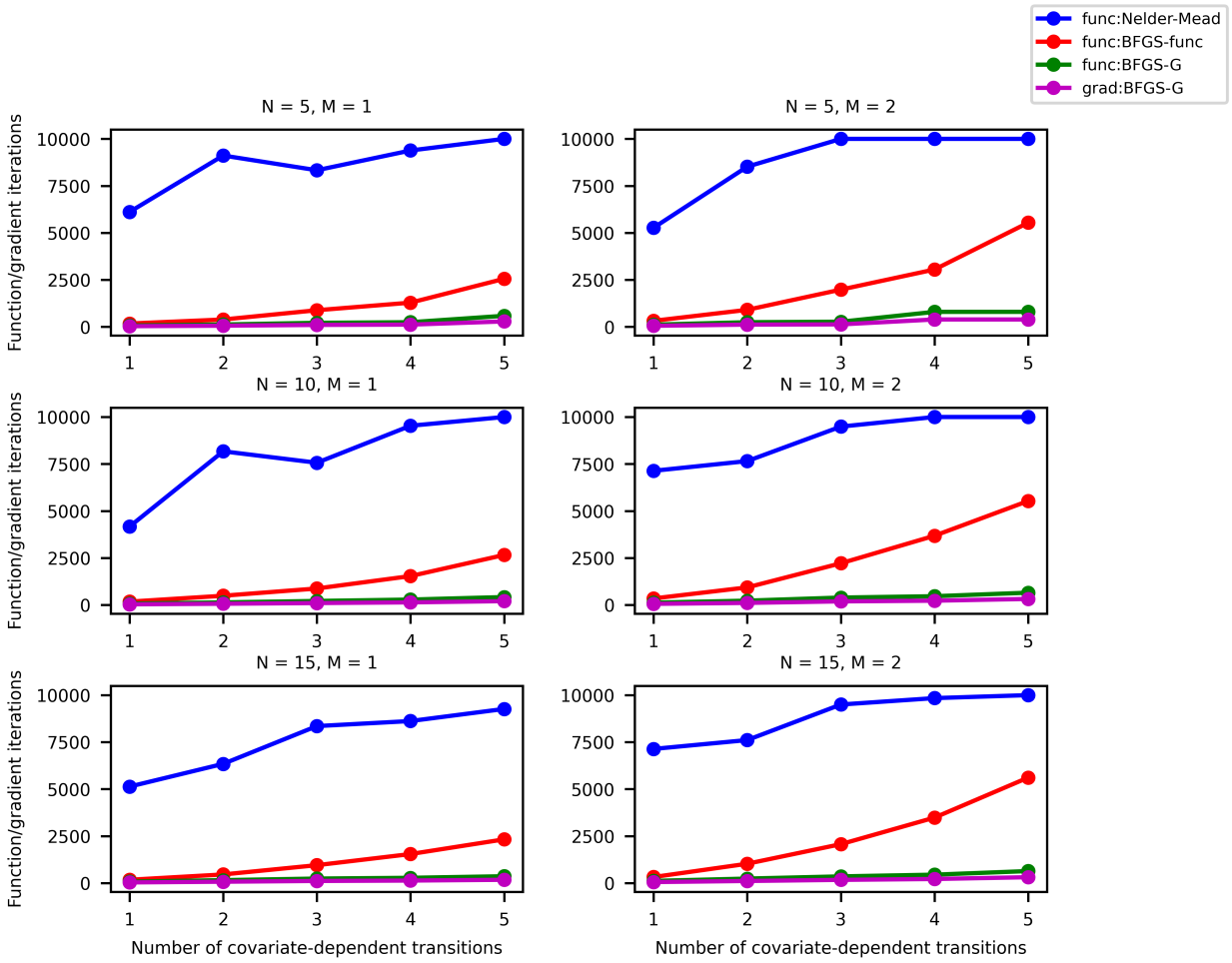

Figure 7: Average number of function/gradient evaluations of Nelder-Mead, BFGS-f, and BFGS-g for the 30 binary out-tree model instances.