Learning Insulin-Glucose Dynamics in the Wild

Andrew C. Miller ACMILLER@APPLE.COM
Apple
Seattle, WA, USA

Nicholas J. Foti NICHOLAS_FOTI@APPLE.COM
Apple
Seattle, WA, USA

Emily Fox EMILY_FOX@APPLE.COM
Apple
Seattle, WA, USA

Abstract

We develop a new model of insulin-glucose dynamics for forecasting blood glucose in type 1 diabetics. We augment an existing biomedical model by introducing time-varying dynamics driven by a machine learning sequence model. Our model maintains a physiologically plausible inductive bias and clinically interpretable parameters — e.g., insulin sensitivity — while inheriting the flexibility of modern pattern recognition algorithms. Critical to modeling success are the flexible, but structured representations of subject variability with a sequence model. In contrast, less constrained models like the LSTM fail to provide reliable or physiologically plausible forecasts. We conduct an extensive empirical study. We show that allowing biomedical model dynamics to vary in time improves forecasting at long time horizons, up to six hours, and produces forecasts consistent with the physiological effects of insulin and carbohydrates.

1. Introduction

Type one diabetes (T1D) is an incurable chronic condition in which the pancreas produces little to no insulin. This lack of insulin frustrates the regulation of blood glucose levels. Left unmanaged, glucose will elevate, ushering in a host long- and short-term health complications. There is no method of prevention, reversal, nor cure; T1D requires constant management. Of the estimated 1.25 million Americans with T1D, 75% are diagnosed in childhood, resulting in a life-long burden of disease management. Management typically entails the injection of subcutaneous insulin to regulate glucose.

T1D is rife with complications. Insufficient insulin leads to chronically elevated blood glucose; common complications including kidney disease, cardiovascular disease, eye disease, and nerve disease. Patients diagnosed with T1D before age ten have a thirty-fold increased risk of coronary heart disease and acute myocardial infarction compared to matched con-

trols. Early-diagnosed T1D patients face a 14-18 year loss in life expectancy (Rawshani et al., 2018). Children and adolescents with T1D begin to show signs of cardiovascular disease after only ten years of disease duration (Singh et al., 2003; Järvisalo et al., 2004; Margeirsdottir et al., 2010; Rawshani et al., 2018).

Excess insulin, on the other hand, can lead to acute complications, such as hypoglycemia. Too much insulin lowers blood glucose to dangerous levels resulting in loss of consciousness or even death (Snell-Bergeon and Wadwa, 2012). Existing insulin delivery systems may target higher than desirable levels of blood glucose to avoid hypoglycemic events. This reflects the asymmetry of negative effects of glucose levels — chronically elevated glucose has negative long term consequences, while low glucose levels can be immediately catastrophic.

Such complications can be avoided by delivering "just enough" insulin. The determination of "just enough" at any given moment is a challenge. One impediment is the unknown (and time-varying) state of the T1D subject — e.g., How sensitive is she to insulin? How many grams of carbohydrates has she absorbed? How do absorbed carbohydrates translate into increased blood glucose?

Patients with T1D endure the constant burden of tuning insulin delivery. Fewer than one-third of T1D patients in the US consistently achieve target blood-glucose levels (Miller et al., 2015). To ease the burden and improve glucose regulation, automatic insulin delivery systems are becoming the new standard for T1D management. Continuous glucose monitors (CGMs) and insulin pumps facilitate the management of T1D. Additionally, these devices present the opportunity to develop more effective insulin delivery algorithms.[1]

Like manual insulin delivery, automatic systems use CGM and insulin pump information to determine the appropriate dose of insulin at any given moment. The insulin pump controller uses forecasts of blood glucose a few hours into the future to deliver the appropriate basal insulin dose. Improved forecasts enable finer control over glucose levels.

Improving forecasts Models of insulin-glucose interaction are used to predict future glucose values. Traditionally, such models of insulin-glucose dynamics have been based in physiology. The seminal Bergman *minimal model* is a multi-compartment insulin model that describes the interaction between active insulin, blood glucose, and endogenous insulin production (Bergman et al., 1981; Bergman, 2005). The more sophisticated UVA/Padova T1D simulator includes a model for oral ingestion of carbohydrates to describe changes in blood glucose (Dalla Man et al., 2014). The UVA/Padova simulator is a useful tool for validating empirical models — it is FDA approved to test the efficacy and risk of insulin delivery policies in automated delivery systems. These simulators, however, were developed to be accurate in highly controlled experimental settings, not for modeling real-world data with noise and missingness.

In contrast, data driven approaches — both traditional statistical methods and machine learning tools — offer the promise of uncovering and leveraging patterns found in the data.

1. Automated delivery systems have been recently developed, measuring glucose every five minutes and automatically adjusting insulin delivery; bolus doses are manually requested. See the Tidepool Loop project (https://www.tidepool.org) for an additional example of an open automated insulin delivery system.

The challenge, of course, is to find models that can infer the intricate (and unobserved) dynamics of the observed data.

Contributions and generalizable insights In this work, we develop a hybrid statistical and physiological model of insulin-glucose dynamics for producing long-term forecasts from real-world T1D management data — CGM, insulin pump, and carbohydrate logs. Our model strikes a balance between purely statistical and purely physiological approaches. We show that statistical machine learning model components — e.g., neural networks and state-space models — can be part of a larger, physiologically-grounded model. This fused model inherits the realistic inductive biases from the physiological model and the flexibility and predictive power of modern machine learning sequence methods.

We show that this hybrid approach can improve forecasts over purely mechanistic or purely statistical approaches on real-world T1D data. Additionally, we show that our model produces physiologically plausible counterfactual predictions under alternative insulin and meal schedules, whereas statistical approaches do not. Importantly, we do not claim that our model *solves* glucose forecasting for T1D management — all models struggle with forecast accuracy at long time horizons. Rather, our contribution is the first in a new family of hybrid statistical-and-physiological models and evidence that this approach can better describe long term structure in real-world T1D data, an important step toward better management of T1D.

The biomedical and epidemiological literature is replete with structured models of complex biological phenomena that may be too inflexible or under-specified to use with real-world sensor data, but nevertheless provide a useful inductive bias (Anderson et al., 1992; Dalla Man et al., 2002; McSharry et al., 2003; Kotani et al., 2005; Trayanova, 2011). Our statistical-and-physiological hybrid approach has the potential to generalize to other biomedical applications. As such, the presented methodology holds promise in many domains beyond T1D.

Section 2 details the data and subjects incorporated into this work. Section 3 details the proposed hybrid model, including the underlying physiological T1D simulator and statistical time series model. Section 4 describes our experimental setup, evaluation metrics, and empirical results. We conclude with a discussion of related work and future research directions in Section 5.

2. Cohort and Data

Our data are observational measurements from two T1D participants using a continuous glucose monitor (CGM) and an insulin pump throughout daily life. The blood glucose level measurements are synchronized and collected using Apple's HealthKit framework.[2] Additionally, we consider the energy spent by the participant throughout the day, as summarized by HealthKit. For every five minute period the following quantities are recorded:

- instantaneous continuous glucose monitor measurement (mg/dL),
- basal and bolus insulin delivered (insulin units),

2. https://developer.apple.com/documentation/healthkit

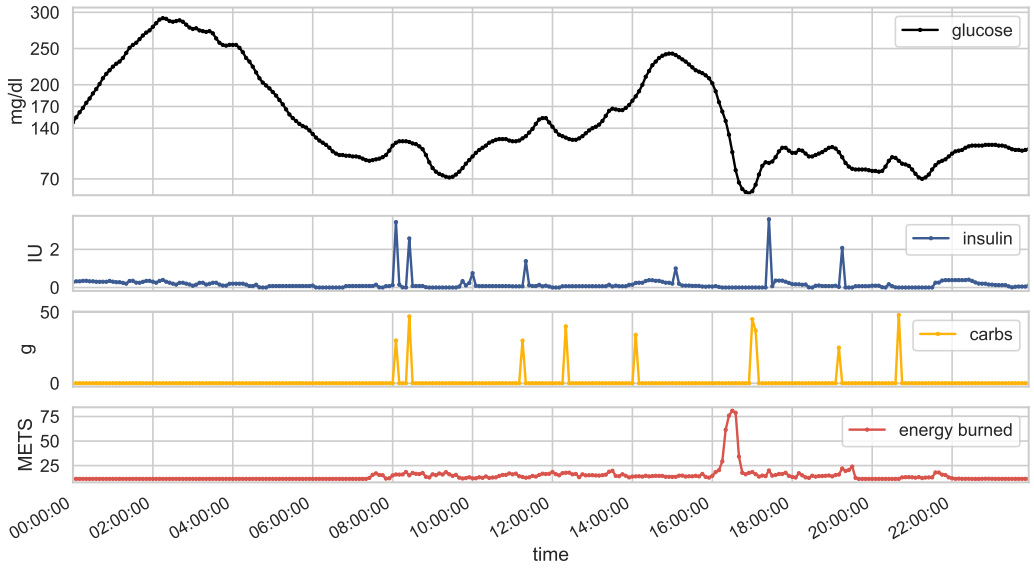


Figure 1: Data for a single day: blood glucose as measured by CGM, insulin delivered by pump, user carbohydrate log, and energy use captured every five minutes by HealthKit.

- estimated carbohydrate ingestion (grams), and
- estimated active energy burned (METs).

Figure 1 shows this data for a single day.

The management of T1D is highly personalized — carbohydrate intake and insulin dosage for one person may not be appropriate for another. While this work only studies the data of two individuals (separately), we analyze an individual's data streams over a long time window. We consider data collected over a 150 day period, using the first 120 days to train and validate a forecasting model, and the subsequent 30 days to measure prediction accuracy. At one sample every five minutes, each participant has recorded over 40,000 data points. A summary of the time period for each participant is in Table 1.

Collecting and processing sensitive health data demands high security standards and strict protocols to protect the privacy and interests of all study participants. The types of data collected — CGM, insulin pump, meal logs, and activity — pose a potential privacy concern to participants. Upon enrollment, all participants were made aware of these risks via an informed consent form. To reduce privacy risks, we restrict the focus of our analysis to a small number of temporal data streams relevant to the management of T1D. Personally identifying information was separated from study data; participants were assigned unique keys. Additionally, researchers accessed the data through a secure virtual private network (VPN). As this study is purely retrospective, there were no direct medical risks to participants throughout the course of the study.

Subject	Train			Test		
	start date	# days	# samples	start date	# days	# samples
1	March 24	120	34381	July 22	31	8819
2	February 25	120	34396	June 25	31	8804

Table 1: Cohort and data summary.

2.1. Data Extraction

The raw CGM and pump data falls on a potentially irregular grid. For simplicity, we interpolate CGM values to a fixed five minute grid, e.g. 6:00AM, 6:05AM, etc. For each five minute period, we compute total insulin units delivered, grams of carbohydrates ingested, and units of energy burned. For CGM gaps longer than five minutes in our observations, we linearly interpolate the values for the relevant five minute periods;[3] this method accounted for fewer than 2% of all observations.

3. Methods

Insulin and glucose obey complicated unobserved dynamics. In T1D, subcutaneous insulin takes time to absorb before reducing blood glucose. Analogously, ingested carbohydrates are absorbed over time before increasing blood glucose levels. Time-varying endogenous glucose production, motion, energy expenditure, and natural diurnal variation in insulin sensitivity further complicate dynamics.

Statistical methods can find patterns in glucose, insulin, and carbohydrate sequences that are predictive of future glucose values. However, these detected patterns may not be stable or structured enough to form reliable long term predictions.

On the other hand, physiological models of insulin-glucose dynamics can — in controlled settings — describe the evolution of blood glucose farther into the future. The T1D simulator we study, the UVA/Padova simulator, is a physiologically-grounded model that describes the interaction of glucose, insulin, and orally ingested carbohydrates within different subsystems of a T1D patient (Dalla Man et al., 2014). While the simulator is faithful to T1D physiology, it was not designed to be robust to the noise and missing observations commonly found in CGM, insulin pump, and meal logs. Additionally, the UVA/Padova simulator does not account for fluctuations in insulin sensitivity and meal absorption rates, limiting its application to short-range forecasts.

Here we present a unified model that that fuses two distinct components — the structured UVA/Padova simulator and a deep state-space model — that balances the useful inductive bias of the physiological simulator with the flexibility of a modern machine learning sequence model to describe complex dynamics in time series data.

3. A cubic spline interpolation was initially used, but discarded because it created glucose values outside of the observed range (and in some instances, negative values).

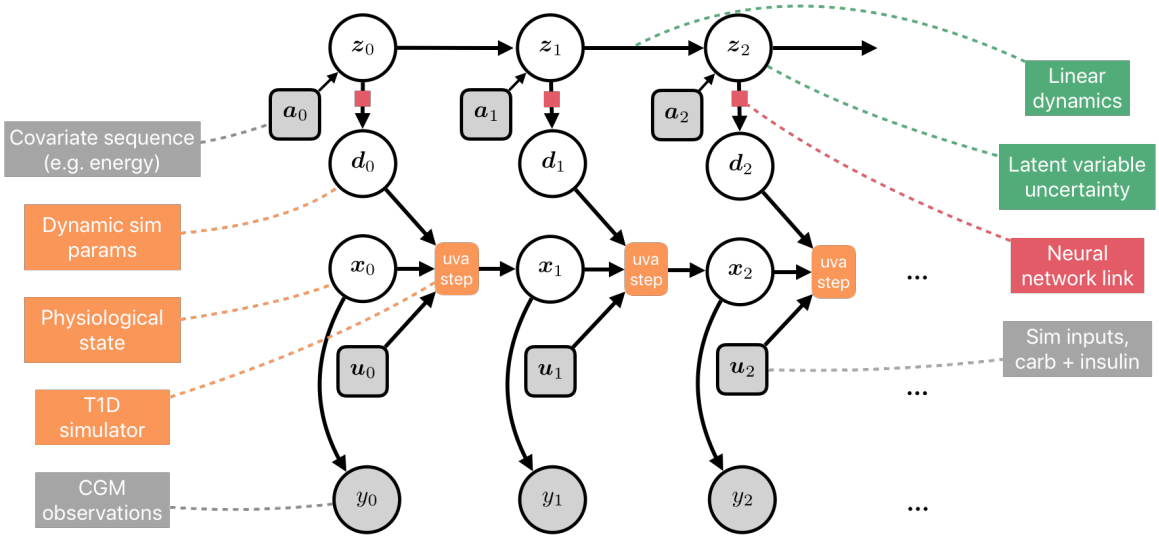


Figure 2: Graphical depiction of the `DTD-sim` model.

The UVA/Padova simulator parameters, which represent insulin/carbohydrate absorption rates and sensitivities, undergo physiologically plausible variation over time that we describe with a deep state-space model. The result is a parsimonious representation of blood glucose that describes short time scales with the UVA/Padova model (for which it was designed), and long-range temporal variation with the deep state-space model. We formalize this fused approach within a single probabilistic generative model, which we call the Deep T1D Simulator (`DTD-Sim`). First, we describe the full `DTD-Sim` generative model. We then unpack the individual model components in the following two subsections.

The `DTD-Sim` Model The generative model assumed for the `DTD-Sim` model is depicted graphically in Figure 2. The aim is to learn a non-linear mapping such that the time-varying parameters to the UVA/Padova simulator can be well-modeled with linear dynamics in some latent space. The `DTD-Sim` model is formally specified as:

$$\boldsymbol{z}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \qquad \text{initial latent state} \qquad (1)$$

$$\boldsymbol{z}_t \sim A\boldsymbol{z}_{t-1} + B\boldsymbol{a}_t + Q^{1/2}\boldsymbol{\epsilon}_t,\ \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, I) \qquad \text{latent temporal dynamics} \qquad (2)$$

$$\boldsymbol{d}_t = \texttt{NN}_\phi(\boldsymbol{z}_t) \qquad \text{dynamic simulator params.} \qquad (3)$$

$$\boldsymbol{x}_t = \texttt{UVA-step}(\boldsymbol{x}_{t-1}, \boldsymbol{d}_t, \boldsymbol{u}_t, \boldsymbol{s}, \Delta_t) \qquad \text{T1D simulator} \qquad (4)$$

$$y_t \sim \mathcal{N}(\texttt{CGM}(\boldsymbol{x}_t, \boldsymbol{s}), \sigma^2) \qquad \text{CGM observation} \qquad (5)$$

where $\boldsymbol{z}_t \in \mathbb{R}^D$, $\boldsymbol{d}_t \in \mathbb{R}^K$, $\boldsymbol{x}_t \in \mathbb{R}^{13}$, $\boldsymbol{u}_t \in \mathbb{R}^J$, and $\boldsymbol{s} \in \mathbb{R}^J$. The dimensionality of the latent space D can be tuned. The physiological state $\boldsymbol{x}_t$ size is fixed by the simulator definition. The simulator parameters chosen to be dynamic, K, and static, J, is a hyperparameter setting, which we fix for this work and describe in Appendix A.

`DTD-Sim` incorporates the following modeling components:

- linear dynamics of the latent state, $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_T$, parameterized by the dynamics matrix A, input matrix B, process covariance Q, and initial mean $\boldsymbol{\mu}_0$ and covariance $\boldsymbol{\Sigma}_0$,

- A non-linear mapping from the latent state $\boldsymbol{z}_t$ to the time-varying simulator parameters $\boldsymbol{d}_t$, modeled as a neural network parameterized by ϕ,
- `UVA-step` integrating the UVA/Padova ODEs, which evolves the physiological state $\boldsymbol{x}_t$ as a function of patient-specific dynamic and static parameters ($\boldsymbol{d}_t$ and $\boldsymbol{s}$, respectively) and insulin delivery and carbohydrates ingested ($\boldsymbol{u}_t$) over a period of time Δ_t, and
- the observed CGM value y_t at time t, modeled via a normal with mean $\texttt{CGM}(\boldsymbol{x}_t, \boldsymbol{s}) \triangleq \boldsymbol{x}^{(6)}/V_G$, where V_G is a parameter in $\boldsymbol{s}$.

The parameters to be estimated are $\boldsymbol{\theta} = \{A, B, Q, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{s}, \phi, \sigma\}$; we construct a variational *maximum a posteriori* estimate of $\boldsymbol{\theta}$. In the following two sub-sections we describe the T1D simulator and the proposed deep state-space model, both of which present challenges for performing parameter estimation. We address these challenges in Section 3.3.

3.1. T1D Simulator

The UVA/Padova T1D simulator represents the instantaneous state of various subsystems of the body, how it changes over time (i.e. dynamics) and how it is driven by inputs (e.g. insulin and ingested carbohydrates). We denote the instantaneous state at time t as $\boldsymbol{x}_t$ for times $t = 1, \ldots, T$.[4] How $\boldsymbol{x}_t$ changes over time is defined by instantaneous dynamics

$$\frac{d\boldsymbol{x}}{dt} = f^{(uva)}(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{p})\,, \tag{6}$$

where the dynamics are a function of the current state, time-varying inputs, and static parameters, respectively.[5] The evolution from state $\boldsymbol{x}_{t-1}$ to $\boldsymbol{x}_t$ involves integrating these dynamics over the time increment Δ_t, which in this work we assume is always one so that:

$$\boldsymbol{x}_{t+1} = \int_t^{t+1} f^{(uva)}(\boldsymbol{x}_{t'}, \boldsymbol{u}_t, \boldsymbol{p}) dt' \tag{7}$$

$$\triangleq \texttt{UVA-step}(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{p})\,. \tag{8}$$

Practically, this step integral can be computed using an ODE solver such as Euler's method or Runge-Kutta methods (Burden and Faires, 2015).

The model represented by $\boldsymbol{x}_t$ and $f^{(uva)}$ is a highly constrained yet complex system developed over a series of papers that spans two decades (Dalla Man et al., 2002, 2007, 2009, 2014) and rooted in the seminal work of Bergman et al. (1981). The components of the state vector $\boldsymbol{x}_t$ correspond to interpretable quantities. For instance, a set of components of $\boldsymbol{x}_t$ describe the *subcutaneous insulin delivery* sub-system, including a dimension that takes delivered insulin as an input. Similarly, other dimensions of $\boldsymbol{x}_t$ describe the *oral glucose*

4. In the exposition, we overload t to represent both index and time value, i.e. t should be thought of as integer values in $[0, T]$. We assume that the time horizon of interest has been rescaled so that the times of measurement are one unit apart.
5. Note that we have not included the *dynamic* parameters $\boldsymbol{d}_t$ in the previous section, as the original model only features static subject-specific parameters $\boldsymbol{p}$. How these dynamic parameters factor in will be described in Section 3.2.

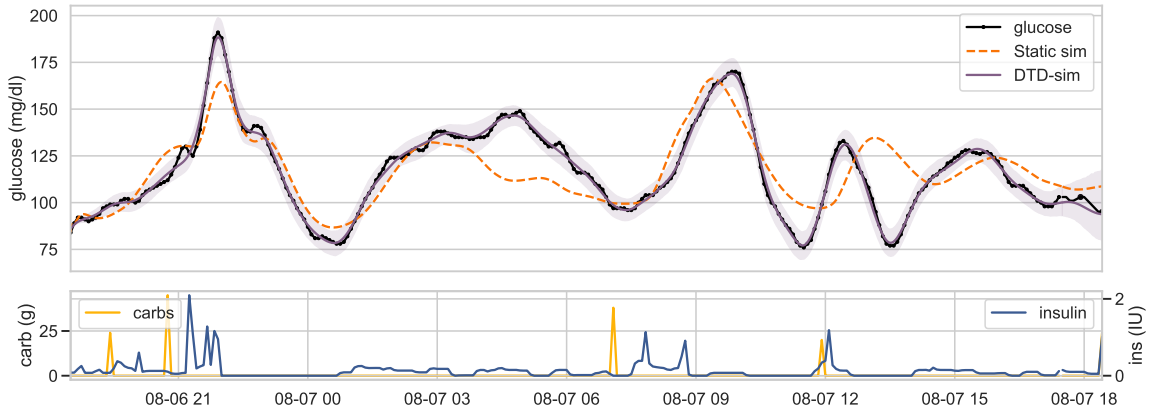


Figure 3: *Static* UVA/Padova lacks the capacity to describe real CGM data. *Top*: model fit comparison of *static* UVA/Padova simulator and *dynamic* `DTD-sim` model for a full day of CGM data. *Bottom*: corresponding insulin and carbohydrate data. Without varying parameters in time and accounting for data noise, the T1D simulator model does not describe observed data.

sub-system models the process by which ingested carbohydrates become measured glucose, including a dimension that is driven by grams of ingested carbohydrates as an input.

The nonlinear function $f^{(uva)}(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{p})$ describes the instantaneous change in state $\boldsymbol{x}_t$ over time. These dynamics are driven inputs into the system — grams of carbohydrates ingested and units of insulin delivered — represented by $\boldsymbol{u}_t$. Dynamics are also specified by subject-specific parameters $\boldsymbol{p}$, which describe (among other aspects) the rate of absorption of carbohydrates and insulin and the sensitivity of blood glucose to absorbed insulin concentration.

We implemented the T1D simulator described in Dalla Man et al. (2014) within the automatic differentiation framework JAX (Bradbury et al., 2018). We use an Euler integration step to solve the ODE at each time t. See Appendix A for details on all components of $\boldsymbol{x}_t$, simulator parameters, and details of the time-derivative function.

The UVA/Padova T1D simulator was designed to validate insulin delivery policies *in silico* over short periods of a few hours at a time — a task for which it is FDA-approved. It was not designed, however, to model continuously collected glucose monitor, insulin pump, and carbohydrate log data over the course of weeks and months. These in-the-wild data are riddled with sources of variability that frustrate the direct application of the UVA/Padova model, stemming from time-varying subject sensitivities, noisy measurements and movement. To illustrate this point, we compare the *static* UVA/Padova data fit to the `DTD-sim` data fit, depicted in Figure 3. The static model does not have the capacity to describe the variability present in noisy data. However, when parameters are allowed to smoothly vary in time, the simulator is able to describe the data over long periods of time.

3.2. Deep State Space Model

To augment the capacity of the T1D simulator model, we allow some of the originally static parameters $\boldsymbol{p}$ to vary in time. We separate the static and time-varying parameters into two vectors denoted $\boldsymbol{s}$ and $\boldsymbol{d}_t$, where $\boldsymbol{p}_t \triangleq (\boldsymbol{d}_t, \boldsymbol{s})$ represents the concatenation of static and dynamic parameters.

We use a state-space model to capture the temporal structure of these time-varying parameters. A state space model describes the evolution of a stochastic process in terms of transition dynamics. We use linear Gaussian dynamics to describe the evolution of $\boldsymbol{z}_t$

$$\boldsymbol{z}_t = A\boldsymbol{z}_{t-1} + B\boldsymbol{a}_t + Q^{1/2}\boldsymbol{\epsilon}_t\,, \tag{9}$$

where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, I)$, and $\boldsymbol{a}_t$ is a time-varying input sequence of covariates. Linear Gaussian dynamics admit computationally tractable inference routines, but assume the restrictive assumption that the latent variable has a multivariate normal distribution (Murphy, 2012). The parameters $\boldsymbol{p}_t = (\boldsymbol{d}_t, \boldsymbol{s})$ of the UVA/Padova simulator are a physiological representation of the underlying system and thus must satisfy complex constraints — to assume $\boldsymbol{d}_t$ evolves according to linear dynamics is oversimplified. To bridge this gap, we learn a neural network to link the latent variable $\boldsymbol{z}_t$ to the dynamic parameters $\boldsymbol{d}_t$ fed into `UVA-Step` at each time step. We denote the parameters of the neural network link function as ϕ, $\boldsymbol{d}_t = \texttt{NN}_\phi(\boldsymbol{z}_t)$. In this work, we use a multi-layer perceptron with two hidden layers of size 128 with rectified linear unit (relu) nonlinearities.

A state-space model is a natural fit to describe the periodic variation typical of a T1D subject. The well-documented "dawn phenomenon" indicates diurnal variation in endogenous glucose production (Porcellati et al., 2013). In fact, recent developments in T1D simulators have begun to incorporate time-variation in insulin sensitivity and endogenous glucose production (Visentin et al., 2018), albeit with rigidly defined variation over the course of a single day. Linear Gaussian state space models can describe periodic variation at multiple temporal resolutions and allow the data to dictate the temporal variability of the simulator parameters.

3.3. Model Fitting and Inference

The goal of model fitting and inference is to find a set of parameters $\hat{\boldsymbol{\theta}}$ that produces good forecasts. We use maximum-likelihood estimation to fit $\hat{\boldsymbol{\theta}}$, which involves maximizing the marginal log-likelihood of the data $\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}; \boldsymbol{\theta})$. Unfortunately, the introduction of the neural network link function and non-linearities in the `UVA-step` simulator do not allow for the closed form computation of the marginal likelihood, complicating inference.

To overcome this intractability, we use variational inference methods to optimize a lower bound of the log-marginal-likelihood (Jordan et al., 1999; Blei et al., 2017). The main issue prohibiting the computation of $\ln p(\boldsymbol{y}; \boldsymbol{\theta})$ is that the posterior of $\boldsymbol{z}_0, \ldots, \boldsymbol{z}_T$, $p(\boldsymbol{z}_{1:T}|\boldsymbol{y}; \boldsymbol{\theta})$, cannot be computed in closed form. To circumvent this, variational methods introduce a posterior approximation for the latent variables $\boldsymbol{z}$, $q_{\boldsymbol{\lambda}}(\boldsymbol{z}_1, ..., \boldsymbol{z}_T) = q_{\boldsymbol{\lambda}}(\boldsymbol{z}_{1:T})$, specified by

variational parameters $\boldsymbol{\lambda}$, resulting in the standard variational objective

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\lambda}) = \mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{z})} \left[\ln p(\boldsymbol{y} \,|\, \boldsymbol{z}; \boldsymbol{\theta})\right] - KL(q_{\boldsymbol{\lambda}}(\boldsymbol{z}) || p(\boldsymbol{z}; \boldsymbol{\theta})) \tag{10}$$

$$\leq \ln p(\boldsymbol{y} \,|\, \boldsymbol{\theta})\,. \tag{11}$$

The variational objective is an expectation over the approximate posterior for $\boldsymbol{z}_{1:T}$. Classic variational methods for graphical models (Ghahramani and Hinton, 2000; Beal et al., 2006; Blei and Lafferty, 2006; Wainwright and Jordan, 2008) rely on conditionally conjugate structure and closed-form updates to optimize the ELBO. Due to the non-linear structure in our generative model, we use Monte Carlo estimates of the gradient (Rezende et al., 2014) to maximize Eq. (10) over $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$. A prior $p(\boldsymbol{\theta})$ can also be incorporated into the variational objective.

Non-centered parameterization The most common form for the approximate posterior for $\boldsymbol{z}_{1:T}$ makes the *mean-field* assumption so that $q_{\boldsymbol{\lambda}}(\boldsymbol{z}_{1:T}) = \prod_{t=0}^{T} q_{\boldsymbol{\lambda}_t}(\boldsymbol{z}_t)$, breaking all temporal dependencies (Wainwright and Jordan, 2008). However, the prior over $\boldsymbol{z}_{1:T}$ is auto-correlated by design through Eq. (2) — we want the latent process to exhibit smooth dynamics, reflecting the belief that physiological parameters change slowly over time. While algorithmically simple, a mean-field approximation is not appropriate in this scenario.

Instead, we *reparameterize* $q_{\boldsymbol{\lambda}}(\boldsymbol{z}_{1:T})$ in terms of the exogenous noise variables $\boldsymbol{\epsilon}_{1:T}$. Using such an alternative parameterization can be algorithmically beneficial (Murray and Adams, 2010). Instead of approximating the posterior over $\boldsymbol{z}_{1:T}$, we can equivalently approximate the posterior over $\boldsymbol{\epsilon}_{1:T}$ and then deterministically transform this posterior (or its samples) to obtain an induced approximate posterior over $\boldsymbol{z}_{1:T}$. Because the $\boldsymbol{\epsilon}_t$ are a priori i.i.d. standard normal, their posterior can be more accurately modeled with a mean-field approximation. Additionally, the KL divergence term in Eq. (10) is simpler to compute.

We approximate the posterior of each $\boldsymbol{\epsilon}_t$ with a multivariate Gaussian with separate means and covariances (which we take to be diagonal for simplicity)

$$q_{\boldsymbol{\lambda}_t}(\boldsymbol{\epsilon}_t) = \mathcal{N}(\boldsymbol{m}_{\boldsymbol{\lambda}}^{(t)}, \mathrm{diag}(\boldsymbol{s}_{\boldsymbol{\lambda}}^{(t)}))\,, \tag{12}$$

where $\boldsymbol{\lambda}_t = (\boldsymbol{m}_{\boldsymbol{\lambda}}^{(t)}, \boldsymbol{s}_{\boldsymbol{\lambda}}^{(t)})$. Recall that $\boldsymbol{z}_t$ is obtained from $\boldsymbol{\epsilon}_t$ according to

$$\boldsymbol{z}_t = A\boldsymbol{z}_{t-1} + B\boldsymbol{a}_t + Q^{1/2}\boldsymbol{\epsilon}_t\,, \tag{13}$$

which shows that $\boldsymbol{z}_t$ depends on $\boldsymbol{z}_{t-1}$ and the induced posterior of $z_{1:T}$ will capture auto-correlation as desired. This dependence is inherited from the structure of the generative model — the correlation induced by the dynamics A. The variational parameters $\boldsymbol{m}_{\boldsymbol{\lambda}}^{(t)}$ and $\boldsymbol{s}_{\boldsymbol{\lambda}}^{(t)}$ will then alter this distribution to reflect the information learned from the data.

The variational objective when using the reparameterization in Eqs. (12) and (13) is nearly identical to the standard ELBO

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\lambda}) = \mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon})} \left[\ln p(\boldsymbol{y} \,|\, g(\boldsymbol{\epsilon}), \boldsymbol{\theta}) + \ln p(\boldsymbol{\epsilon} \,|\, \boldsymbol{\theta}) - \ln q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon})\right] \tag{14}$$

$$= \mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon})} \left[\ln p(\boldsymbol{y} \,|\, \boldsymbol{z}, \boldsymbol{\theta})\right] - KL\left(q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon}) \,||\, \mathcal{N}(0, I)\right)\,. \tag{15}$$

where the expectation is now over $q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon}_{1:T})$ and $g(\cdot)$ maps $\boldsymbol{\epsilon}$ to $\boldsymbol{z}$ as defined in Eq. (13). The KL divergence term is a simple analytic function. The expected log-likelihood term, however, is still intractable. Because we can easily sample from $q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon}_{1:T})$ we instead form a Monte Carlo estimate of the gradient of Eq. (15) and use stochastic gradient methods (Rezende et al., 2014). Using the whitened space is a simpler alternative to more advanced variational approximations for state space models (Archer et al., 2015; Bamler and Mandt, 2017) — incorporation of these techniques may benefit learning θ in our model setting.

Maintaining stable dynamics When fitting the dynamics matrix A, a practical consideration is ensuring stability in the latent time series. If the maximal eigenvalue of A is larger than one, $\boldsymbol{z}_t$ will diverge over time, leading to unstable forecasts. Ensuring the maximal eigenvalue of A is less than one is in tension with the desire to model long term structure in $\boldsymbol{z}_t$. If the maximal eigenvalue of A is less than one, the dynamics defined by A will dampen $\boldsymbol{z}$, and encourage the process to go to zero when unrolling into the future (i.e. forming long term forecasts).

To ensure that the eigenvalues of A are close to one in magnitude, we subtract a penalty term from Eq. (15) of the form $\text{trace}(A) - D$ — that is the average eigenvalue should be close to one. Our stochastic gradient updates may make A become unstable, so throughout optimization we project the iterates A back into the set of unit norm matrices by renormalizing eigenvalues that are greater than one. We found this projection to be an essential step to reliably learn model parameters $\boldsymbol{\theta}$ with stochastic gradients.

3.4. Forecasting

Given a variational approximation for the latent states up to time t, $q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon}_{1:t})$, constructing forecasts is a straightforward application of the `DTD-Sim` generative process. Given data $\boldsymbol{y}_{1:t}$ and the corresponding variational parameters $\boldsymbol{m}_{\boldsymbol{\lambda}}^{(1:t)}$ and $\boldsymbol{s}_{\boldsymbol{\lambda}}^{(1:t)}$, we can construct a forecast for a time horizon h by first sampling from the posterior distribution over latent variable dynamics

$$\boldsymbol{z}_0 \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Sigma}}_0) \tag{16}$$

$$\tilde{\boldsymbol{\epsilon}}_{1:t} \sim q_{\boldsymbol{\lambda}}(\boldsymbol{\epsilon}_{1:t}) \qquad \text{approx. posterior sample} \tag{17}$$

$$\boldsymbol{z}_{t+j} \sim \hat{A}\boldsymbol{z}_{t+j-1} + \hat{B}\boldsymbol{a}_{t+j} + \hat{Q}^{1/2}\tilde{\boldsymbol{\epsilon}}_{t+j},\ j = 1, \ldots, h \qquad \text{induced posterior} \tag{18}$$

where $\boldsymbol{\epsilon}_s \sim \mathcal{N}(0, I)$ for $s > t$ and $\hat{\boldsymbol{\mu}}_0$, $\hat{\boldsymbol{\Sigma}}_0$, $\hat{A}$, $\hat{B}$, and $\hat{Q}$ are plug-in estimates of the dynamics parameters from optimizing the variational objective. We then run the simulation forward

$$\hat{\boldsymbol{d}}_{1:t+h} = NN_{\hat{\phi}}(\boldsymbol{z}_1), \ldots, NN_{\hat{\phi}}(\boldsymbol{z}_{t+h}) \tag{19}$$

$$\hat{\boldsymbol{x}}_{1:t+h} = \texttt{UVA-solve}(\boldsymbol{d}_{1:t+h}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{x}}_0) \tag{20}$$

$$\hat{\boldsymbol{y}}_{1:t+h} \sim \mathcal{N}(\texttt{CGM}(\boldsymbol{x}_{1:t+h}, \hat{\boldsymbol{s}}), \hat{\sigma}^2) \tag{21}$$

where again $\hat{\boldsymbol{s}}$ and $\hat{\sigma}^2$ are plug-in estimates from maximizing Eq. (15). This procedure produces one posterior predictive sample of $\boldsymbol{y}_{t+1:t+h}$, which has a non-Gaussian marginal distribution due to the nonlinearities in the simulation component of the model. We use the plug-in Bayes estimate over samples of $\boldsymbol{z}$, $\mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{z})}\left[\boldsymbol{y}_{t+h}\right]$ for forecasts.

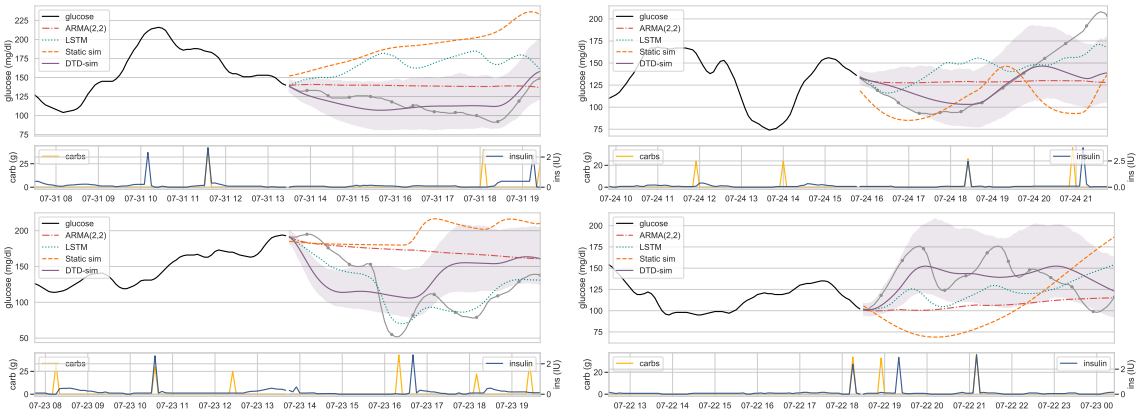


Figure 4: A sample of four forecasts from different days. Observed glucose is in solid black; future glucose is in solid grey with dots indicating every half hour. In solid purple we depict the proposed method `DTD-sim` forecasts and 95% posterior credible intervals. We also depict the point forecasts for the static UVA/Padova simulator, ARMA(2,2) and LSTM models.

4. Empirical Study

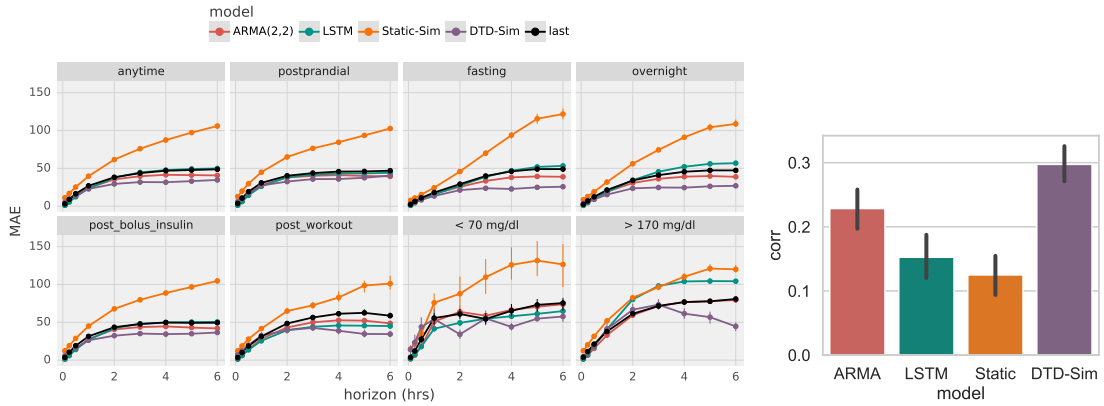
Here we describe the empirical study conducted on the cohort detailed in Section 2. We first measure the quality of forecasts produced by our `DTD-sim` model and compare it to a variety of baseline approaches. We then look at generated forecasts under counterfactual future meal and insulin schedules, examining how the different models are influenced by these inputs.

4.1. Forecasting glucose at varying time horizons

We measure the accuracy of forecasts at multiple time horizons h, up to six hours. For each forecast horizon h, we report the mean absolute error (MAE) between the forecast and the true glucose value. In Appendix B we report additional statistics, including root mean squared error, and mean absolute scaled error (MASE) (Hyndman and Koehler, 2006). We also consider predictions in different *contexts*. These contexts are defined by time of day, sleep, recent meals, recent bolus injections, or elevated/low glucose levels. Forecast accuracy is more important in some contexts — for example automated monitoring blood glucose during sleep is crucial for safely avoiding hypoglycemic episodes.

Baseline methods We compare our approach to a handful of baseline approaches for time series forecasting. These approaches include purely statistical approaches and a time-invariant simulator. Specifically, we compare our approach to the following baselines

- autoregressive moving average (ARMA) models,
- long short term memory (LSTM) neural networks,
- the static UVA/Padova simulator, and
- the last available glucose measurement.



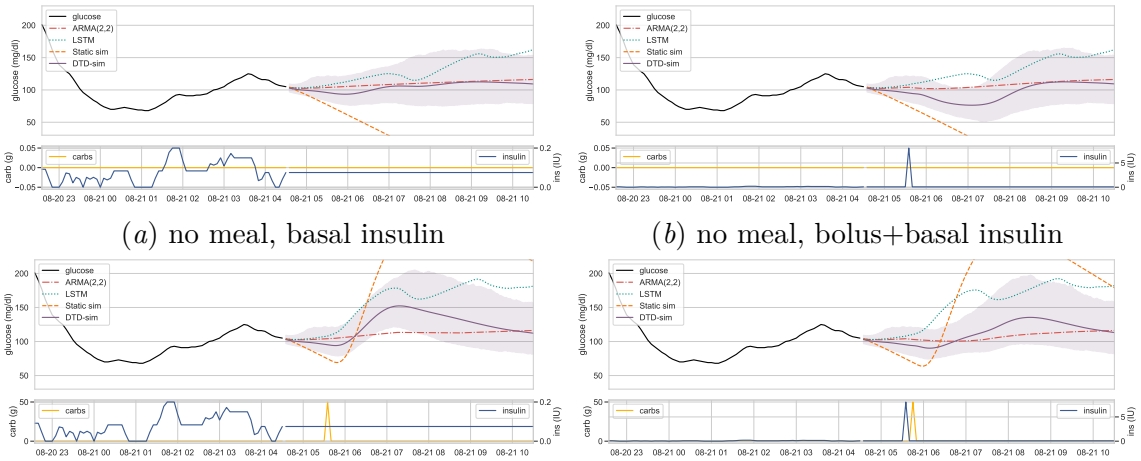
(*a*) MAE (participant 1) (*b*) Forecast correlations

Figure 5: Prediction results by context. The first context, anytime, is an average over the entire prediction window. We observe that the `DTD-Sim` model outperforms both the statistical and mechanistic baselines across all contexts at longer prediction horizons, where h is one to six hours.

For all models, we train on the first 90 days, validate using the next 30 days and test on the remaining 31 days (as described in Table 1). For the ARMA model, we grid search over both autoregressive order p and moving average order q using a validation set. The LSTM model includes a forget gate, and hidden states feed into a two layer perceptron with a ReLU nonlinearity; we search over latent state dimension size using a validation set. Linear time series methods have been used extensively for blood glucose forecasting, and we include the non-linear LSTM model as an additional strong benchmark (Montaser et al., 2017; Xie and Wang, 2018). The static UVA/Padova T1D simulator model is a baseline with fixed simulator parameters over time. Because this model cannot describe long periods of time, we re-train the static simulator for each forecast over a running window of data. Here, we use a moving window of 6 hours to tune model parameters before constructing a forecast.

We compare these baselines to the `DTD-Sim` model where we grid search over the latent state dimension D using a validation set. The results of the these quantitative experiments are depicted in Figure 5a. The `DTD-sim` model performs as well or better than the baselines a few hours after the baseline. We see a particularly large improvement at much longer horizons — for example the `DTD-Sim` model improves upon the baseline LSTM by 5% at one hour, 27% at two hours, and 35% at three hours and 36% at six hours. Further, we see improvements at long horizons across most contexts. The LSTM and ARMA(2,2) models form the best short term forecasts, at 5 to 30 minutes in the future.

The static simulator underperforms the other dynamical models, including the purely statistical models. This poor performance highlights the unrealistic constraints imposed by the simulator — that sensitivities and endogenous glucose production are fixed in time.



(*a*) no meal, basal insulin

(*b*) no meal, bolus+basal insulin

(*c*) full meal, basal insulin

(*d*) full meal, bolus+basal insulin

Figure 6: The hybrid model balances sensitivity to meals and bolus insulin doses. Depicted are four counterfactual scenarios — bolus/no-bolus insulin and meal/no meal, each one hour after the latest observation. The fully mechanistic static simulator model is overly sensitive to bolus insulin doses and full meals, quickly growing above 200 mg/dl and shrinking to 0 mg/dl. The AR and LSTM models are less sensitive to bolus doses and full meals. The LSTM model appears to be influenced by a large meal but not a bolus insulin dose. The `DTD-sim` model is both sensitive to bolus insulin and meals, but more stable than the static simulator.

For a qualitative comparison of model forecasts, we graphically depict a sample of forecast sequences in Figure 4. In these plots we can see some differing behaviors between the models, including the sensitivity to carbohydrate and insulin inputs (which we explore in more detail in the following section).

While model predictions may be off in terms of mean squared error, the general shape of the forecast can still match the true glucose value quite well. To quantify this and compare our model, we compute the empirical correlation between the forecast $\hat{y}_{t:t+h}$ and true glucose $y_{t:t+h}$ for $h = 6$ hours. We report the average forecast correlation over $N = 1{,}000$ randomly chosen test locations, plotted in Figure 5b. We observe that, on average, the `DTD-sim` model consistently produces forecast sequences that correlate more highly than the baseline approaches.

4.2. Counterfactual forecasts

There is a causal relationship between insulin and carbohydrate inputs and the resulting blood glucose level — increasing insulin dose should cause the glucose level to fall, and ingesting a large meal should cause the glucose level to rise. The T1D simulator encodes this inductive bias in the structure of the differential equation $f^{(uva)}(\cdot)$. The statistical

methods, on the other hand, need to discover this relation from data and could potentially learn a spurious relationship.

We compare the forecast behavior of the ARMA, LSTM, Static simulator and `DTD-Sim` models under different counterfactual scenarios. We construct a synthetic insulin delivery and meal schedule and generate forecasts given a fixed sequence of observed data. We consider two settings for each input — a 50 gram meal compared to no meal and a bolus dose of 8 insulin units compared to no bolus dose (with a constant basal delivery) resulting in four counterfactuals. We graphically compare these scenarios in Figure 6.

The static simulator is overly sensitive to insulin, going to zero within three to four hours when receiving both a bolus dose or just basal insulin. Similarly, the static simulator spikes to over three hundred after a meal (with and without the insulin bolus). The LSTM model appears to be sensitive to carbohydrate inputs, with predicted glucose increasing shortly after the meal. However, the LSTM model does not appear to be sensitive to an insulin bolus, forming similar forecasts in the bolus and no bolus scenarios. The `DTD-sim` model noticeably reacts to carbohydrate and insulin inputs, with glucose increasing shortly after receiving a meal and decreasing shortly after receiving an insulin bolus.

Additional synthetic meal and insulin schedule comparisons are depicted in Figure 8 at randomly selected test points. We observe a similar pattern of carbohydrate sensitivity and insulin insensitivity for the LSTM compared to the `DTD-sim` model.

5. Discussion and Related Work

Related work As machine learning methods become more pervasive in the sciences, a common goal is to endow ML algorithms with knowledge from applied domains; our work shares this goal with many other approaches.

For example, physics-guided neural networks have been applied in a geological setting (Karpatne et al., 2017). This approach introduces constraints on activations and outputs of the neural network model to enforce physical consistency. In the application of modeling lake temperatures, the physics-guided RNN leverages physical knowledge by pre-training on data simulated from a standard differential equation model of lake temperature (Jia et al., 2019). More broadly, simulation models in the physical sciences are an invaluable tool for understanding complex phenomena. Machine learning techniques have been used to aid inference in applying simulators to real data (Cranmer et al., 2020), develop new models (Carleo et al., 2019), constrain neural networks (Raissi et al., 2019), and model control problems (Long et al., 2018). With these lines of research we share the common goal of incorporating complex scientific knowledge into a model for real data. Our approach does not use physical laws or prior knowledge to restrict a flexible model, but rather embeds a physiological model into the data generating procedure. We introduce the flexibility needed to model real-world observational data by allowing physiological parameters to vary in time according to a sequence model. A more expansive study of hybrid statistical-and-physical techniques, such as pre-training on simulated data, may yield additional benefits and will be considered in future work.

Physiological simulators of insulin-glucose dynamics for T1D subjects have been developed over the past few decades (Cobelli et al., 2011). One line of research has matured into a FDA-approved simulator (Dalla Man et al., 2002; Kovatchev et al., 2009; Dalla Man et al., 2014). Further enhancements have been considered and tested in lab settings, including improvement of meal simulation (Dalla Man et al., 2007) and the incorporation of physical activity (Dalla Man et al., 2009). Physiological models of T1D have been applied to real world CGM and insulin pump data. Liu et al. (2019) demonstrate the utility of a simple physiological model fit using a deconvolution method of the glucose signal. This work, however, does not consider temporal-variation or patterns in subject-specific variables, such as insulin sensitivity.

The use of tractable latent dynamics with a neural network emission or link function is a common strategy for describing complex observations. Structured variational autoencoders (Johnson et al., 2016) and deep state-space models (Krishnan et al., 2017) both use variational inference to fit models with complex observations or complex dynamics (or both). Our approach, while conceptually similar, is distinct in that we model the latent dynamics that capture the variation in simulator *parameters* rather than the data itself. We model a low-dimensional phenomenon governed by complex, time-varying latent dynamics

A related line of modeling work incorporates differential equation solvers in probabilistic models (Chen et al., 2018; Rubanova et al., 2019). This framework uses neural networks to learn the functional form of the dynamics. Our goal is to instead make an existing ODE simulator more flexible, but still enjoy the inductive bias described by the simulator.

Discussion and future work Accurately forecasting blood glucose can afford more time to adjust insulin dosage or meals, crucial to the management of T1D. To construct more accurate and physiologically plausible forecasts, we integrated a T1D simulator into a machine learning sequence model and applied it to real-world CGM, insulin, and meal log data. We view the `DTD-Sim` model as a first step in building a reliable glucose forecasting model that will enable better planning and management for T1D.

We envision many improvements to this model for long-term blood glucose forecasts. One obvious shortcoming of our approach is that we are not directly modeling the stochasticity of the input carbohydrates and insulin. While latent variables can account for some of this uncertainty, a direct model of noise in both the observation of meals and their overall mass could improve forecasts.

Another direction for improvement is to include additional input sequences as inputs to `DTD-Sim`. For example, movement, step count, heart rate, or other proxies for energy expenditure may inform the temporal variation in insulin sensitivities. Explicitly modeling seasonal variation at daily and monthly temporal resolutions may also improve forecast accuracy. Further, the functional form of $f^{(uva)}(\cdot)$ can likely be improved upon in a data-driven way. While we assume that $f^{(uva)}(\cdot)$ is fixed, we may want to use the simulator as a starting point and relax the functional form given more observations.

The expression of T1D varies from person to person. Our algorithm development has been limited to a small cohort. With an increased diversity in insulin-glucose observations, a joint model of many subjects through a hierarchy may help improve long term forecasts.

Finally, our hybrid statistical-and-physiological modeling approach may be suitable for adapting other biophysical ODEs that describe complex phenomena over time to model real-world data. Models of the cardiovascular system, biomechanics, or long term patient trajectories could be augmented in a way similar to our approach.

Acknowledgments

ACM thanks Ben Kreis, Nate Racklyeft, Jen Block, and Luke Winstrom for supporting this research project, and Leon Gatys for enriching methodological discussions and review of an early draft.

References

Roy M Anderson, B Anderson, and Robert M May. *Infectious diseases of humans: dynamics and control*. Oxford university press, 1992.

Evan Archer, Il Memming Park, Lars Buesing, John Cunningham, and Liam Paninski. Black box variational inference for state space models. *arXiv preprint arXiv:1511.07367*, 2015.

Robert Bamler and Stephan Mandt. Dynamic word embeddings. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 380–389, 2017.

Matthew J Beal, Zoubin Ghahramani, et al. Variational bayesian learning of directed graphical models with hidden variables. *Bayesian Analysis*, 1(4):793–831, 2006.

Richard N Bergman. Minimal model: perspective from 2005. *Hormone Research in Paediatrics*, 64(Suppl. 3):8–15, 2005.

Richard N Bergman, Lawrence S Phillips, and Claudio Cobelli. Physiologic evaluation of factors controlling glucose tolerance in man: measurement of insulin sensitivity and beta-cell glucose sensitivity from the response to intravenous glucose. *The Journal of clinical investigation*, 68(6):1456–1467, 1981.

David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120, 2006.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, and Skye Wanderman-Milne. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. Cengage Learning, Boston, tenth edition, 2015.

Giuseppe Carleo, Ignacio Cirac, Kyle Cranmer, Laurent Daudet, Maria Schuld, Naftali Tishby, Leslie Vogt-Maranto, and Lenka Zdeborová. Machine learning and the physical sciences. *Reviews of Modern Physics*, 91(4):045002, 2019.

Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in neural information processing systems*, pages 6571–6583, 2018.

Claudio Cobelli, Eric Renard, and Boris Kovatchev. Artificial pancreas: past, present, future. *Diabetes*, 60(11):2672–2682, 2011.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 2020.

C Dalla Man, Andrea Caumo, and Claudio Cobelli. The oral glucose minimal model: estimation of insulin sensitivity from a meal test. *IEEE Transactions on Biomedical Engineering*, 49(5):419–429, 2002.

Chiara Dalla Man, Robert A Rizza, and Claudio Cobelli. Meal simulation model of the glucose-insulin system. *IEEE Transactions on biomedical engineering*, 54(10):1740–1749, 2007.

Chiara Dalla Man, Marc D Breton, and Claudio Cobelli. Physical activity into the meal glucose—insulin model of type 1 diabetes: in silico studies, 2009.

Chiara Dalla Man, Francesco Micheletto, Dayu Lv, Marc Breton, Boris Kovatchev, and Claudio Cobelli. The uva/padova type 1 diabetes simulator: new features. *Journal of diabetes science and technology*, 8(1):26–34, 2014.

Zoubin Ghahramani and Geoffrey E Hinton. Variational learning for switching state-space models. *Neural computation*, 12(4):831–864, 2000.

Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688, 2006.

Mikko J Järvisalo, Maria Raitakari, Jyri O Toikka, Anne Putto-Laurila, Riikka Rontu, Seppo Laine, Terho Lehtimäki, Tapani Rönnemaa, Jorma Viikari, and Olli T Raitakari. Endothelial dysfunction and increased arterial intima-media thickness in children with type 1 diabetes. *Circulation*, 109(14):1750–1755, 2004.

Xiaowei Jia, Jared Willard, Anuj Karpatne, Jordan Read, Jacob Zwart, Michael Steinbach, and Vipin Kumar. Physics guided rnns for modeling dynamical systems: A case study in simulating lake temperature profiles. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 558–566. SIAM, 2019.

Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems*, pages 2946–2954, 2016.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

Anuj Karpatne, William Watkins, Jordan Read, and Vipin Kumar. Physics-guided neural networks (pgnn): An application in lake temperature modeling. *arXiv preprint arXiv:1710.11431*, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kiyoshi Kotani, Zbigniew R Struzik, Kiyoshi Takamasu, H Eugene Stanley, and Yoshiharu Yamamoto. Model for complex heart rate dynamics in health and diseases. *Physical Review E*, 72(4):041904, 2005.

Boris P Kovatchev, Marc Breton, Chiara Dalla Man, and Claudio Cobelli. In silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes, 2009.

Rahul G Krishnan, Uri Shalit, and David Sontag. Structured inference networks for nonlinear state space models. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

Chengyuan Liu, Josep Vehí, Parizad Avari, Monika Reddy, Nick Oliver, Pantelis Georgiou, and Pau Herrero. Long-term glucose forecasting using a physiological model and deconvolution of the continuous glucose monitoring signal. *Sensors*, 19(19):4338, 2019.

Yun Long, Xueyuan She, and Saibal Mukhopadhyay. Hybridnet: Integrating model-based and data-driven learning to predict evolution of dynamical systems. In *Conference on Robot Learning*, pages 551–560, 2018.

Hanna Dis Margeirsdottir, Knut Haakon Stensaeth, Jakob Roald Larsen, Cathrine Brunborg, and Knut Dahl-Jørgensen. Early signs of atherosclerosis in diabetic children on intensive insulin treatment: a population-based study. *Diabetes care*, 33(9):2043–2048, 2010.

Patrick E McSharry, Gari D Clifford, Lionel Tarassenko, and Leonard A Smith. A dynamical model for generating synthetic electrocardiogram signals. *IEEE transactions on biomedical engineering*, 50(3):289–294, 2003.

Kellee M Miller, Nicole C Foster, Roy W Beck, Richard M Bergenstal, Stephanie N DuBose, Linda A DiMeglio, David M Maahs, and William V Tamborlane. Current state of type 1 diabetes treatment in the us: updated data from the t1d exchange clinic registry. *Diabetes care*, 38(6):971–978, 2015.

Eslam Montaser, José-Luis Díez, and Jorge Bondia. Stochastic seasonal models for glucose prediction in the artificial pancreas. *Journal of diabetes science and technology*, 11(6): 1124–1131, 2017.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Iain Murray and Ryan P Adams. Slice sampling covariance hyperparameters of latent gaussian models. In *Advances in neural information processing systems*, pages 1732–1740, 2010.

Francesca Porcellati, Paola Lucidi, Geremia B Bolli, and Carmine G Fanelli. Thirty years of research on the dawn phenomenon: lessons to optimize blood glucose control in diabetes. *Diabetes care*, 36(12):3860–3862, 2013.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Araz Rawshani, Naveed Sattar, Stefan Franzén, Aidin Rawshani, Andrew T Hattersley, Ann-Marie Svensson, Björn Eliasson, and Soffia Gudbjörnsdottir. Excess mortality and cardiovascular disease in young adults with type 1 diabetes in relation to age at onset: a nationwide, register-based cohort study. *The Lancet*, 392(10146):477–486, 2018.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pages 1278–1286, 2014.

Yulia Rubanova, Tian Qi Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, pages 5321–5331, 2019.

Tajinder P Singh, Harvey Groehn, and Andris Kazmers. Vascular function and carotid intimal-medial thickness in children with insulin-dependent diabetes mellitus. *Journal of the American College of Cardiology*, 41(4):661–665, 2003.

Janet K Snell-Bergeon and R Paul Wadwa. Hypoglycemia, diabetes, and cardiovascular disease. *Diabetes technology & therapeutics*, 14(S1):S–51, 2012.

Natalia A Trayanova. Whole-heart modeling: applications to cardiac electrophysiology and electromechanics. *Circulation research*, 108(1):113–128, 2011.

Roberto Visentin, Enrique Campos-Náñez, Michele Schiavon, Dayu Lv, Martina Vettoretti, Marc Breton, Boris P Kovatchev, Chiara Dalla Man, and Claudio Cobelli. The uva/padova type 1 diabetes simulator goes from single meal to single day. *Journal of diabetes science and technology*, 12(2):273–281, 2018.

Martin J Wainwright and Michael Irwin Jordan. *Graphical models, exponential families, and variational inference*. Now Publishers Inc, 2008.

Jinyu Xie and Qian Wang. Benchmark machine learning approaches with classical time series approaches on the blood glucose level prediction challenge. 2018.

Appendix A. UVA/Padova T1D Simulator

As noted in the main text, the UVA/Padova T1D simulator model represents the instantaneous state of various subsystems of the body as a state vector and dynamics function

$$\frac{d\boldsymbol{x}}{dt} = f^{(uva)}(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{s}) \equiv \dot{\boldsymbol{x}} \tag{22}$$

where the components of $\boldsymbol{x}_t \in \mathbb{R}^{13}$ represents the instantaneous physiological state of the body at time t, $\boldsymbol{u}_t \in \mathbb{R}^2$ denote the insulin units and carbohydrate mass delivered at time t, and $\boldsymbol{s}$ are subject-specific simulator parameters that represent endogenous glucose production rates and insulin sensitivity.

In the UVA/Padova model, the state is a thirteen-dimensional vector representing various subsystems. The oral glucose subsystem contains components

$$\boldsymbol{x}^{(1)} = Q_{sto1} \qquad \text{first stomach compartment} \tag{23}$$
$$\boldsymbol{x}^{(2)} = Q_{sto2} \qquad \text{second stomach compartment} \tag{24}$$
$$\boldsymbol{x}^{(3)} = Q_{gut} \qquad \text{first stomach compartment} \tag{25}$$

The glucose subsystem describes two compartment glucose kinetics

$$\boldsymbol{x}^{(4)} = G_p \qquad \text{plasma glucose} \tag{26}$$
$$\boldsymbol{x}^{(5)} = G_t \qquad \text{tissue glucose} \tag{27}$$
$$\boldsymbol{x}^{(6)} = G_s \qquad \text{subcutaneous glucose (CGM)} \tag{28}$$

The insulin subsystem describes insulin kinetics, including the absorption into active insulin

$$\boldsymbol{x}^{(7)} = I_p \qquad \text{plasma insulin} \tag{29}$$
$$\boldsymbol{x}^{(8)} = I_l \qquad \text{liver insulin} \tag{30}$$
$$\boldsymbol{x}^{(9)} = X_L \tag{31}$$
$$\boldsymbol{x}^{(10)} = X \qquad \text{active insulin} \tag{32}$$
$$\boldsymbol{x}^{(11)} = \tilde{I} \tag{33}$$

Finally, the subcutaneous insulin subsystem describes the absorption kinetics of delivered insulin

$$\boldsymbol{x}^{(12)} = I_{sc1} \qquad \text{subcutaneous compartment one} \tag{34}$$
$$\boldsymbol{x}^{(13)} = I_{sc2} \qquad \text{subcutaneous compartment two} \tag{35}$$

Given the definition of the $\boldsymbol{x}_t$ state components, we define the dynamics of each subsystem (along with useful intermediate quantities).

The oral glucose subsystem evolves as

$$\dot{Q}_{sto1}(t) = -k_{gri} \cdot Q_{sto1}(t) + D \cdot \delta(t) \tag{36}$$
$$\dot{Q}_{sto2}(t) = -k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) + k_{gri} \cdot Q_{sto1}(t) \tag{37}$$
$$\dot{Q}_{gut}(t) = -k_{abs} \cdot Q_{gut}(t) + k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) \tag{38}$$
$$Q_{sto}(t) \triangleq Q_{sto1}(t) + Q_{sto2}(t) \tag{39}$$
$$Ra(t) \triangleq \frac{f \cdot k_{abs} \cdot Q_{gut}(t)}{BW} \qquad \text{glucose rate of appearance} \tag{40}$$

Glucose kinetics are

$$\dot{G}_p(t) = EGP(t) + Ra(t) - U_{ii}(t) - E(t) - k_1 \cdot G_p(t) + k_2 \cdot G_t(t) \tag{41}$$
$$\dot{G}_t(t) = -U_{id}(t) + k_1 \cdot G_p(t) - k_2 \cdot G_t(t) \tag{42}$$
$$\dot{G}_s(t) = -\frac{1}{T_s} \cdot G_s(t) + \frac{1}{T_s} \cdot G(t) \tag{43}$$
$$G(t) \triangleq G_p(t)/V_G \tag{44}$$

where the endogenous glucose production and insulin-based utilization are defined

$$EGP(t) = k_{p1} - k_{p2} \cdot G_p(t) - k_{p3} \cdot X_L(t) \tag{45}$$
$$U_id(t) = \frac{(V_{m0} + V_{mx} \cdot X(t) \cdot (1 + r_1 \cdot risk)) \cdot G_t(t)}{K_{m0} + G_t(t)} \tag{46}$$

Insulin kinetics are defined

$$\dot{I}_p(t) = -(m_2 + m_4) \cdot I_p(t) + m_1 \cdot I_l(t) + R_{ai}(t) \tag{47}$$
$$\dot{I}_l(t) = -(m_1 + m_3) \cdot I_l(t) + m_2 \cdot I_p(t) \tag{48}$$
$$\dot{X}_L(t) = -k_i \cdot \Big(X_L(t) - \tilde{I}(t)\Big) \tag{49}$$
$$\dot{\tilde{I}}(t) = -k_i \cdot \Big(\tilde{I}(t) - I(t)\Big) \tag{50}$$
$$\dot{X}(t) = -p_{2U} \cdot X(t) + p_{2U}\,(I(t) - I_b) \tag{51}$$
$$I(t) = I_p(t)/V_I \tag{52}$$

The subject-specific simulator parameters $\boldsymbol{s}$ include

$$\boldsymbol{s} = (k_{min}, k_{max}, k_{abs}, f, b, d, V_G, k_{1:2}, V_I, m_{1:4}, k_{p1:3}, k_i, \tag{53}$$
$$F_{snc}, V_{m0}, K_{m0}, I_b, k_{e1:2}, k_{a1:2}, k_d, k_{sc}, BW) \tag{54}$$

We succinctly express these equations as $f^{(uva)}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{s}) : \mathbb{R}^{13} \times \mathbb{R}^2 \times \mathbb{R}^{29} \to \mathbb{R}^{13}$.

A.1. Time-varying simulator parameters

We augment the existing simulator model by allowing some components of the parameter vector $\boldsymbol{s}$ to vary over time.

Appendix B. Empirical Study Supplement

Here we include additional plots to support the empirical study. The mean absolute scaled error (MASE) is defined as

$$MAE_0(h) = \frac{1}{T-h}\sum_{t=h}^{T}|Y_t - Y_{t-h}| \tag{55}$$

$$q_t(h) = \frac{Y_t - \hat{Y}_{t|t-h}}{MAE(h)} \tag{56}$$

$$MASE(h) = mean(|q_t(h)|)\,, \tag{57}$$

where $MAE_0(h)$ is the error of a simple baseline — the forecast value is equal to the latest observation. Intuitively, MASE measures the MAE ratio between this simple baseline and the predicted model — a value of 1 indicates no improvement over the baseline. See Hyndman and Koehler (2006) for more details.

Appendix C. Inference Algorithm Details

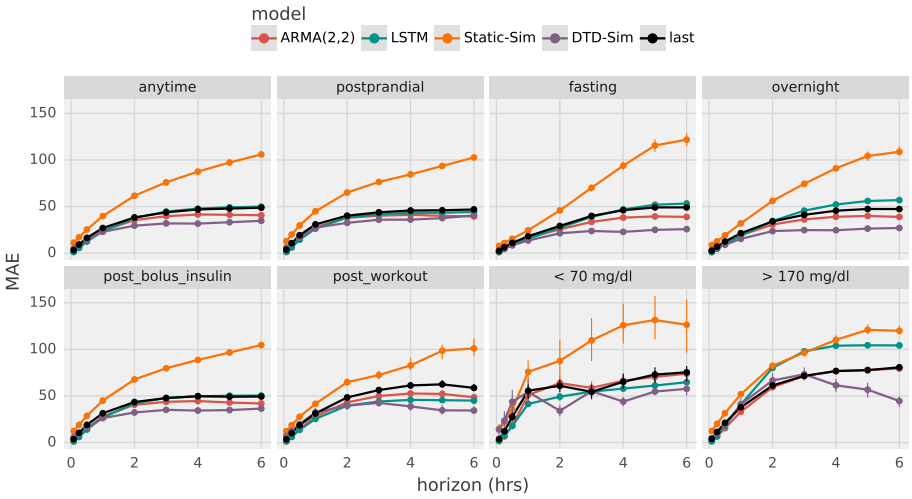
In this appendix we provide a step-by-step overview of our proposed variational inference algorithm for `DTD-Sim`. Throughout, we use the notation $\mathbf{z}_{1:T}$ to indicate the set of all $\mathbf{z}_t$, $t = 1, \ldots, T$, and similarly for other variables.

Each iteration of the algorithm proceeds by sampling the latent trajectories $\mathbf{z}_{1:T}$ and transforming them into the parameters of the UVA-Padova simulator. The interpretable state-space parameter at time t, $\mathbf{x}_t$, is then determined by integrating the UVA-Padova system forward one time-step. Then, $\mathbf{x}_t$ is used to compute the expected log-likelihood of the observed data which we differentiate through in order to update the parameters. The detailed steps of the algorithm are as follows:

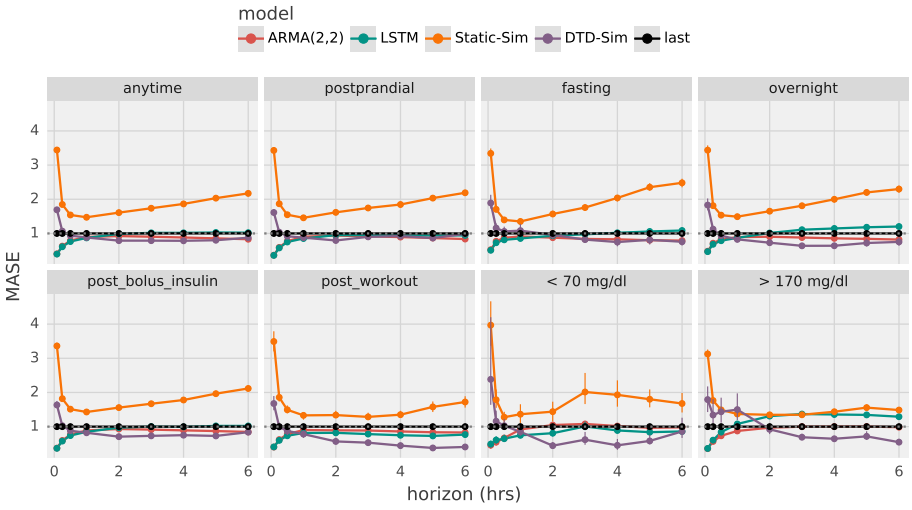
1. Sample $\tilde{\epsilon_\mathbf{t}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t = 1, \ldots, T$.
2. Sample $\mathbf{z}_0 \sim \mathcal{N}(\mu_0, \mathbf{\Sigma}_0)$.
3. For $t = 1, \ldots, T$
   (a) Construct $\mathbf{z}_t = A\mathbf{z}_t + B\mathbf{u}_t + Q^{1/2}\tilde{\epsilon}_t$.
   (b) Compute $\mathbf{d}_t = \mathrm{NN}_\phi(\mathbf{z}_t)$.
   (c) Compute $\mathbf{x}_t = \texttt{UVA-step}(\mathbf{x}_{t-1}, \mathbf{d}_t, \mathbf{u}_t, \mathbf{s}, \Delta_t)$ using a numerical integration scheme.
4. Compute $\nabla_{\lambda,\theta} \ln p(\mathbf{y}_{1:T}|\mathbf{z}_{1:T}; \theta)$, where

$$\ln p(\mathbf{y}_{1:T}|\boldsymbol{z}_{1:T}; \boldsymbol{\theta}) = \prod_{i=1}^{T} \ln p(y_t|\boldsymbol{z}_t; \boldsymbol{\theta}). \tag{58}$$

5. Update λ and θ according to this gradients.



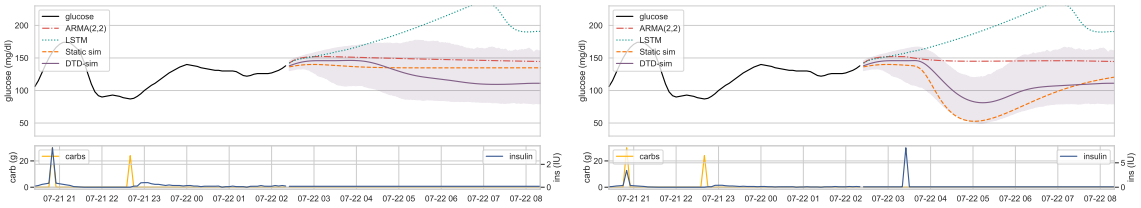
(*a*) RMSE (participant 1)



(*b*) MASE (participant 1)

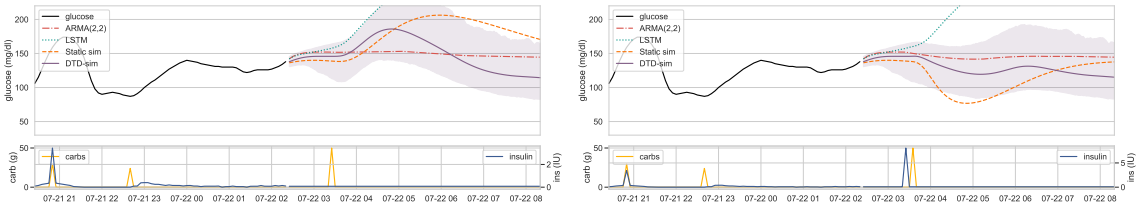
Figure 7: Prediction results by context. The first context, anytime, is an average over the entire prediction window. We observe that the `DTD-Sim` model outperforms both the statistical and mechanistic baselines across all contexts at longer prediction horizons, where h is one to six hours.

We used Euler's method to perform the `UVA-step` implemented with Jax in order to propagate derivatives. To optimize the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$ we used `adam` (Kingma and Ba, 2014) with a step size of 1e-4. A single Monte Carlo sample was found to adequately estimate the stochastic gradient for each update, which incorporates all of the training observations.



(*a*) no meal, basal insulin

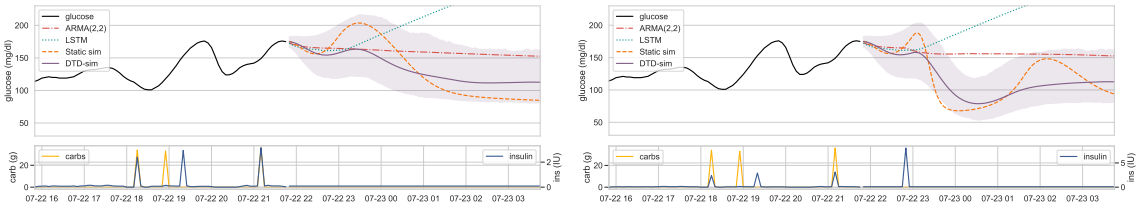
(*b*) no meal, bolus+basal insulin



(*c*) full meal, basal insulin

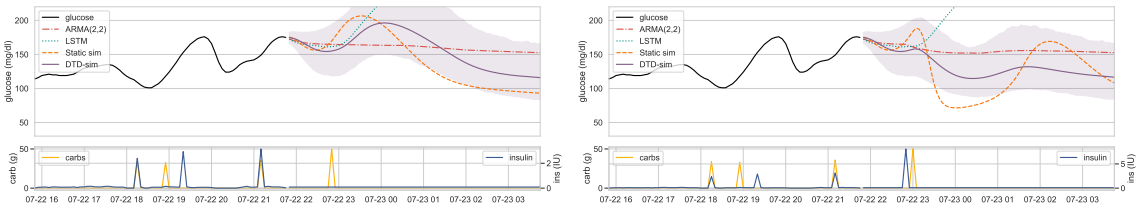
(*d*) full meal, bolus+basal insulin

Figure 8: Additional synthetic meal comparisons.



(*a*) no meal, basal insulin

(*b*) no meal, bolus+basal insulin



(*c*) full meal, basal insulin

(*d*) full meal, bolus+basal insulin

Figure 9: Additional synthetic meal comparisons.

We stop iterating the algorithm when the loss stops improving for 500 iterations — this typically occurred after 10-15,000 iterations.