The Parameter Houlihan: a solution to high-throughput identifiability indeterminacy for brutally ill-posed problems

D. J. Albers,[1,2,*] M Levine,[3,†] L Mamykina,[2,‡] and G. Hripcsak[2,§]

[1] *Department of Pediatrics, Division of Informatics, University of Colorado, Aurora, CO 80045*
[2] *Department of Biomedical Informatics, Columbia University, 622 West 168th Street, PH-20, New York, NY 10032*
[3] *Department of computational and mathematical sciences, 1200 E California Blvd M/C 305-16 Pasadena, CA 91125*

One way to interject knowledge into clinically impactful forecasting is to use data assimilation, a nonlinear regression that projects data onto a mechanistic physiologic model, instead of a set of functions, such as neural networks. Such regressions have an advantage of being useful with particularly sparse, non-stationary clinical data. However, physiological models are often nonlinear and can have many parameters, leading to potential problems with parameter identifiability, or the ability to find a unique set of parameters that minimize forecasting error. The identifiability problems can be minimized or eliminated by reducing the number of parameters estimated, but reducing the number of estimated parameters also reduces the flexibility of the model and hence increases forecasting error. We propose a method, the parameter Houlihan, that combines traditional machine learning techniques with data assimilation, to select the right set of model parameters to minimize forecasting error while reducing identifiability problems. The method worked well: the data assimilation-based glucose forecasts and estimates for our cohort using the Houlihan-selected parameter sets generally also minimize forecasting errors compared to other parameter selection methods such as by-hand parameter selection. Nevertheless, the forecast with the lowest forecast error does not always accurately represent physiology, but further advancements of the algorithm provide a path for improving physiologic fidelity as well. Our hope is that this methodology represents a first step toward combining machine learning with data assimilation and provides a lower-threshold entry point for using data assimilation with clinical data by helping select the right parameters to estimate.

PACS numbers:
Keywords: data assimilation; identifiability; machine learning; inverse problems; physiology; Markov Chain Monte Carlo; glucose-insulin

I. INTRODUCTION

We want to use data and our understanding of the world to better manage health — we want evidence and understanding to guide clinical and personal health-related decisions. Of course at a high level this is generally what medicine is about: interventions are undertaken only when they are understood or predicted to improve an individual's health. However, traditionally this prediction is done in a non-personalized manner, meaning that interventions treat the "mean" person or patient. Personalized and precision medicine were conceptualized to relax this constraint by tailoring an intervention to a person. While genetics offers a path to personalizing treatment, we can also use data science machinery together with personal ([1]) and population-scale data to better personalize treatment ([2–4]). Specifically here, we want to leverage our knowledge encapsulated in mechanistic physiologic models and combine it with free living or clinical data to allow this knowledge and data to be used to make decisions related to health. In this context, computational problems related to personalized medicine can be broken into two broad categories: *forecasting*, where we make quantitative predictions about a patient's future state that can be used by clinicians and patients to take corrective action, and *phenotyping* ([5–9]), where we identify properties of macroscopic observables that can be used to classify patients into subgroups that can give clinicians and researchers actionable insight into commonly occurring treatment outcomes and biological phenomena.

The idea of using mechanistic models and data assimilation in biomedicine or healthcare is old, but what is new is attempting to integrate models with variable complexity with sparse, noisy free-living and clinically collected data.

*Electronic address: david.albers@ucdenver.edu
†Electronic address: mlevine@caltech.edu
‡Electronic address: om2196@cumc.columbia.edu
§Electronic address: hripcsak@columbia.edu

Many mathematical biology models were designed to have variable degrees of biological fidelity, fidelity that we do not necessarily want to eliminate or reduce, but fidelity that we generally need to constrain in the usual case where we cannot estimate all the parameters because of data limitations that always exist in practice. This problem poses a significant barrier to using data assimilation—enough of a barrier that often data assimilation is not even attempted because the models, given data are hopelessly poorly resolved. This paper poses a machine learning solution to this problem—by using machine learning to identify and rank-order which model parameters are the most necessary to estimate.

Returning to the more practical contexts of phenotyping and forecasting, both applications impose particular demands on certain aspects of computational machinery used to model data. The properties we focus on here are the selection of the model parameters to estimate and the ensuing *identifiability* of a model, or ability to uniquely solve for parameters that yield optimal solutions ([10–12]). Our goal is to strike a balance between identifiability and model fidelity in situations where a model is not fully identifiable if all or even sometimes when any model parameters are estimated, given the available data. The method we develop here can facilitate both forecasting and phenotyping studies, and we evaluate this method in the context of modeling glucose dynamics using mechanistic models, machine learning and data assimilation.

The Houlihan, or the Houlihan throw, is a lasso throw used for roping livestock, e.g., a horse. It is used often under difficult circumstances such as picking out, from a substantial distance, a single horse from among a crowd of horses standing close together. It is a particularly flexible technique that can be used in a variety of circumstances. In this spirit we intuitively define the Houlihan method(s) as a collection of methods that use for selecting the most productive model parameters to estimate; specifically, the collection of methods uses machine learning techniques applied to simulated model output under parameter variation subject to a set of features, e.g., the mean of a state.

II. BACKGROUND

The larger biomedical context of this work is the application of data science machinery used to personalize forecasts and phenotypes via a broadly defined regression. While there are many linear versions of regression that have been successfully applied to healthcare data ([13–16]), here we focus on a specific type of nonlinear regression—data assimilation—in an effort to take advantage of potentially important nonlinearities present in most biological systems. Nonlinear regression approaches such as deep learning and related methods ([9, 17–19]) have seen some success in a number of biomedical applications thanks to their ability to approximate arbitrary, non-linear functions. While the flexibility of universal approximator approaches ([20, 21]) is particularly useful when little is known about the system and data are plentiful, this approach does not always work well when data are sparse and non-stationary, leading to problems such as poor generalization to new or unobserved individuals, problems with quickly changing health conditions, and difficulties with fast, accurate prediction with very few, e.g., 20, data points. Unfortunately, many health data and healthcare situations fit one or more of these data pathologies ([22, 23]).

In order to exploit the complex yet rich quantity of available health data, it is natural to consider ways of constraining the search space for machine learning methods. One way to do this is to constrain the model search space in accordance with as much expert knowledge as possible. To achieve this here we turn to mechanistic models developed by mathematical biologists and physiologists [24], which are typically formulated as dynamical systems ([25–27]), e.g., $x_{t+1} = f(x_t, \theta)$, or differential equations ([28, 29]), e.g., $\frac{dx}{dt} = f(x, \theta, t)$, where x are the time-varying states of the system and θ are the physiologic parameters that govern the process. For example, in the case of phenotyping type 2 diabetes one way of constraining the search space of a regression is to regress the data onto a nonlinear physiologic model [1, 30] instead of regressing the data onto a universal approximator [20, 21] function space such as neural networks. The way this is done is using data assimilation.

Data assimilation (DA) is a collection of methods ([31–40]) concerned with performing the types of non-linear regressions we describe for dynamical systems, and centers itself around forecasting and inferring mechanistic states under available observations; it solves both forward and inverse problems ([41–43]). There have been many successful applications of mechanistic modeling and data assimilation in biomedicine ([1, 30, 42, 44–74]). However, mechanistic models that are typically developed in biological laboratory settings are often not designed to interface with health data collected in the process of delivering care or in free-living situations—in particular, the physiologic models often model macroscopic states that *are* observable from routinely collected data but are governed by a composition of *unobservable* mechanisms. While these models capture the dynamics we are interested in and constrain the regression to a smaller class of functions, their high-fidelity creates issues of identifiability and ill-posedness, problems for which this paper develops a practical, machine-learning-based work-around.

To understand how identifiability works for these machines, consider a trivial case of identifiability for the model $\frac{dx}{dt} = abx$. If we assume that a and b are unknown parameters, they cannot both be identifiable without another equation that could uniquely determine one of them. This topic and the the associated methods for handling this

situation are too old and wide ranging to give complete background ([10–12, 75]). We can, however, give a broad sketch of how identifiability has been traditionally approached. Identifiability analysis generally follows one of three pathways: *analytical methods*, e.g., showing algebraically that all parameters can be uniquely solved for ([76–78]); *numerical methods* ([10, 12, 79]); and *heuristic, knowledge-driven* sensitivity analyses where certain parameters are chosen based on computational experiments or knowledge of the system. In many complex, non-linear mechanistic physiologic models algebraic methods and linear computational methods are not tractable or applicable. In these situations nonlinear methods can be applied, but nonlinear methods usually have to be constructed for a particular situation ([80]), and, much like nonlinear optimization, generally do not have clearcut or simple resolutions ([75, 80]). *These problems pose a significant roadblock to parameter inference in the context of DA.* Nevertheless, there exist methods for working to remain within a traditional identifiability framework, e.g., [75] uses Bayesian inference to determine when parameters can be made identifiable.

The usual way of addressing these issues focuses on making sure the model is identifiable or finding ways of making it more identifiable ([10, 12, 75]). This work is often performed using substantial intuition about the important features encoded in the model, and parameterizing and grouping sub-processes. However, this creates silos of expertise and prevents wide-spread dissemination and evaluation of mechanistic models in potential application domains. Therefore, to progress toward understanding complex physiology via model refinement and selection, and to provide solutions in clinical situations that come with constraints of time-sensitive solutions, we must find a robust way of coping with brutally ill-posed problems and accept certain impurities and inaccuracies.

Here, we develop and evaluate a method for rank-ordering mechanistic parameters based on their "influence" on important dynamical features, in order to improve forecast accuracy and help determine which models most faithfully represent a given system. This provides a starting point from which to estimate parameters, prune the model, etc., that can be automated.

III. CONCEPTUAL CONSTRUCTION OF THE HOULIHAN APPROACH

A. Conventional operational use of data assimilation with ill-posed problems

The standard method of applying data assimilation (DA) or control in generic situations follows roughly the following steps: (i) select a model, (ii) work out identifiability, (iii) select a filter or inference method, (iv) find an optimal solution for states and parameters. This requires very careful experimental constructions, generally dense data streams, can be expensive, and requires relatively simple models, all situations that lie outside of what is possible in applications and even many basic science settings. The approach for applying DA in *operational*, complex, high-dimensional settings where accurate *real-time* forecasts are imperative is to: (i) select or develop a model, (ii) tune and fix parameters offline, often by hand or using a combination of by-hand and numeric tuning that allows the model to reconstruct or forecast states within some tolerance, (iii) select an inference scheme, and (iv) estimate states only and make a forecast. This is a tried and true method and is used in situations such as weather and climate forecasting ([39, 40, 81]). Neither of these approaches apply to biomedical situations that, by comparison, have a different set of constraints and problems, including: (i) the models are smaller, so they can be simulated faster and estimated faster, allowing for potentially many models to be used simultaneously; (ii) there are less data relative to the number of *unknown* parameters, so while parameter estimation is necessary [1] not all parameters can be estimated; (iii) models are not generated from first principles and their application to given individuals is potentially highly variable necessitating the use and potentially the averaging of many models; and (iv) tuning would have to be done for millions of people frequently, e.g., every patient in every ICU potentially every day, a process that is not likely to be practically possible. Because of these reasons, choosing which parameters to estimate is a significant barrier to the adoption and use of DA in biomedical situations.

B. Houlihan approach to ill-posedness

Here we are operating under a different situation from the more canonical DA application setting, one more heavily constrained by imperfections of data that will never disappear because the data are collected in the process of managing health instead of data collected in a controlled manner explicitly for the DA. In particular we assume: *(a)* we do not know the right model but we have some models we can try, *(b)* we do not know whether a given model is identifiable and that we do not have enough data to estimate all model parameters well anyway but that we have enough data to estimate at least one parameter, *(c)* for a given model, we don't know what parameters are the most useful to estimate, given that we cannot estimate all of them. Given this situation we develop a method for rank ordering which parameters to estimate, subject to features we want to capture, when we have no idea how to choose which

parameters to estimate or when we must choose parameters in a more high-throughput setting where we are using many models at once.

This solution involves stacking machine learning on top of DA: machine learning is applied to *simulated* model output to select the important parameters to estimate to best synchronize the model with the data, and then we use DA restricted to estimation of the parameters chosen by machine learning. In this way, the method will scale to a high-throughput setting and can be applied to many different models with high dimensional parameter spaces more easily. And while we know that this method may not lead to a unique solution in function, parameter, or initial condition space, the set of solutions will be reduced to a workable set of solutions that allow forward progress to be made.

Conceptually, we are proposing to: *(i)* assume a model, *(ii)* simulate the model under discrete parameter variation creating a grid in parameter space for which at every point we have simulated data from the model (i.e., the instance of one attractor of the model for a given set of initial conditions at the parameter grid point), *(iii)* select features, e.g., the mean, of the attractors that are important for estimating the physiologic system, and then *(iv)* use a machine learning algorithm to identify the parameters that have the greatest impact on the features. While for the authors the geometric intuition of this method originates from bifurcation theory—we will discuss this in a later section [82]—one useful way to think about the problem is in the inverse problems context. *As was the case for the bifurcation theory context, this discussion is allegorical; we are not proposing a formal inverse problems regularization framework.* From a high level, given data, Y, and a model, $\mathcal{F}$ with a state space x, the task is to find a *set* of parameter values, Θ_i, of which there may be many if the system is not identifiable, that minimize:

$$||Y - \mathcal{F}(x, \theta)||_Y^p \tag{1}$$

for some p, $p = 2$ being the commonly applied least squares minimization. The core of the identifiability issue is that, for complex models, and especially given sparse data, there may be many sets of parameters Θ_i that minimize the distance between the model and the data. In this case a goal might then be to balance the number of potential minimizing parameter sets, the number of Θ_i's, against the distance between the model and the data via an optimization algorithm, e.g.,

$$\min_{\Theta}(w_1(||Y - \mathcal{F}(x, \theta)||_Y^p) + w_2(\#\{\Theta_i\})) \tag{2}$$

where the w_i's are continuous functions. This framework, a formal regularization methodology, has many advantages, but can induce many complexities that increase rather than decrease the barrier to using data assimilation in more data-poor environments. Moreover, this relatively complex methodology may not be applicable in more high-throughput situations where, e.g., many models are used in a model averaging context. Therefore, motivated by the goal of an imperfect but practical solution, we postulate that if we carefully select the right parameters that maximize the parameter subspaces that can be explored relative to a set of desired features, we can often, effectively but imperfectly, solve the optimization problem. Effectively but not rigorously, we are *regularizing a priori*, by selecting and reducing the parameter set to be estimated before we go about estimating the parameters given data. Given the framework above, such a solution may be well handled by a tool from sparse machine learning such as lasso [83] because it uniquely rank-orders parameters by their predictive power, but it is easy to imagine using other methods. But, it is important to be clear that we are hypothesizing that the parameter subspaces that allow maximal exploration of dynamics relative to a given feature, e.g., the mean, will contain sets of parameters, Θ_i that also find relatively good minima of Eq. 1. In our evaluation we will see cases where this hypothesis fails, but we will also see that this hypothesis generally holds true in our data set, and this conclusion is the point of the paper.

In short, here we are assuming a problem is ill posed and a system that is likely not identifiable, and given this situation, we are trying to cope. Therefore, we are not really solving an identifiability task because we are not trying to find *the* best or most representative model that admits *unique* parameter estimates; rather we are solving a problem more akin to, but not literally, a regularization task. We are starting from a point where the problem is both brutally ill-posed and likely non-identifiable, and where investigating identifiability using analytic methods, or even many numeric-by-hand methods are intractable. In this case we are assuming there will be a few different parameter combinations that represent reasonable parameter estimates. In this situation each combination of parameter represents a hypothesis for how the system works. More importantly, the method we present here is a flexible entry-point for using data assimilation with a complex nonlinear model and data collected in an uncontrolled environment rather than directly solving an identifiability problem.

IV. DATA COHORT

We test and evaluate the Houlihan methodology in the context of modeling and forecasting blood glucose collected in a free-living setting — via a type 2 diabetes self-management moblie application. The dlood glucose and nutrition data

Data Summary				
Participant ID	P1	P2	P4	P5
Age	$40-50$	$40-50$	$40-50$	$40-50$
Disease Status	T2D	T2D	No Diabetes	No Diabetes
Medications	metformin	metformin	—	—
Total # glucose measurements	304	211	520	322
Total # meals recorded	124	76	370	184
Total # days measured	16	16	53	52
Mean measured glucose	113 ± 25	127 ± 32	92 ± 17	101 ± 16

TABLE I: Demographic information and summary statistics are reported for the four participants whose retrospectively collected data are included in the study.

used here were collected retrospectively from four participants, two with type 2 diabetes and two without diabetes, using custom-designed mobile applications for capturing self-monitoring data ([84]). These data are summarized in Table I. We acquired two types of data: 1) fingerstick blood glucose measurements taken at the discretion of each of the 4 participants (roughly 3-10 times per day) and 2) estimates of carbohydrate consumption over time (roughly 1-5 meals per day) determined by a certified dietitian's analysis of the daily meal logs (with photos and descriptions) reported by each participant.The data are documented more completely in ([1, 74]) and are available on PhysioNet upon request.

V. METHODS

A. Glucose-insulin physiologic model

The Houlihan method was conceived in the context of DA with a mechanistic model, and while it could be used in any nonlinear regression context, this paper will be restricted to the setting where we begin by projecting data onto a mechanistic dynamical system and then work to decide which parameters of that dynamical system we should estimate to represent the data. The mechanistic model is more formally either a dynamical system when time is discrete or a system of ODEs when time is continuous. Explicit versions of such systems form parameterized families of functions that are physically meaningful but generally do not satisfy nice function space properties such as completeness and are not universal approximators. The more general theory of dynamical systems can be found in many books ([25–27]), but here we will restrict our use of these details to an absolute minimum. We will assume that the systems we use have at least one invariant density; the invariant density is likely defined relative to a SRB-measure ([85–88]) rather than Lebesque measure, but the point is that for a given set of parameter values and initial conditions, the states have a probability density function associated with them denoted Λ. *This invariant density can potentially depend on both the parameters and the initial conditions for a set of parameters.*

As previously noted, we want to use DA to model the glucose-insulin system of a human being. We begin with a particular mechanistic glucose-insulin model, here the ultradian model that has been detailed in [1, 2, 4, 24, 89], and has 6 states and 21 parameters; its details can be found in the appendix A. The model has unknown identifiability properties, especially when only glucose is measured, but we have strong evidence that at least some of the model parameters and states are not identifiable ([74]). The Houlihan method rests on quantifying how the invariant densities of the *synthetic data sets* and their properties vary as parameters of the mechanistic model(s) vary. Specifically, the Houlihan method decides which parameters to estimate by varying the parameters of the ultradian model, observing how the invariant densities and their properties vary, and then using this information to select parameters to estimate by ranking ordering their importance using statistical inference or machine learning. The synthetic data used to select parameters to estimate will be generated by solving the ultradian model using an adaptive version of Runga-Kutta, ode23 in Matlab and will consist of 10^5 simulated data points.

B. Stochastic filtering and inverse problems methods

We use two previously documented data assimilation formulations, an unscented Kalman filter ([90–95]) (UKF) whose details can be found in [1] and a Metropolis-within-Gibbs Markov Chain Monte Carlo (MCMC) method whose details can be found in [74, 96]. As previously mentioned, these DA methods are used with the ultradian model ([89])

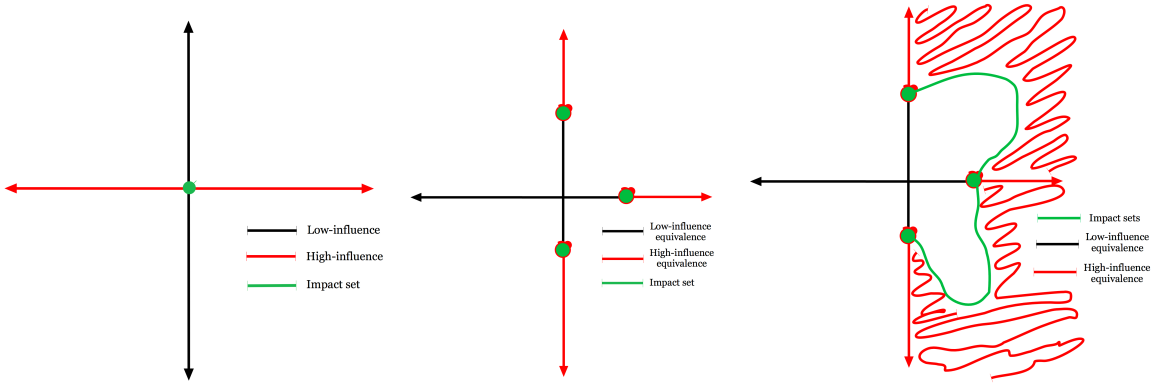

FIG. 1: Shown are three different Houlihan constructions: left shows equivalence class by coordinate—this is the construction we use in this paper; middle shows equivalence by subsets of coordinates but retains the non-joint parameter dependency assumption; right shows a fully joint equivalence where combinations of parameters can generate influence when individual parameters do not, similar to the notion of bifurcation sets.

for performing the DA tasks. We only use these methods over the course of evaluating the Houlihan methods; the exact implementation of the DA methods can be found in [1, 74].

C. Analytical construction and intuition for throwing the Houlihan around the right parameters

While the approach we are proposing is new, the *allegorical* geometric intuition motivating this approach comes from bifurcation theory and in particular the bifurcation sets defined in the 1970's ([97]) and the analytic geometry vision of bifurcation theory and singularities in parameter space [29]. Bifurcation sets are the low-dimensional sets or manifolds that denote transition/bifurcation surfaces between topologically equivalent invariant sets, *partitioning the parameter space into a set of equivalence classes.* It is this idea of partitioning the parameter space into equivalence classes that differently impact dynamical featuers we care about is they key motivational insight. In our context we want to partition the parameter space by influence on some feature or set of features, denoted the *feature-metric*, of the dynamics. Feature-metrics are calculated from the time-series of the simulated model (dynamical system), e.g., a mean. We do not want to be as rigid as requiring topological equivalence as was defined in the bifurcation sets framework, or necessarily strict classes, but we do want to partition the parameter space according to how parameters influence a dynamical feature we care about. The over-arching idea is that the subsets of parameter space that have the highest influence on the feature-metric are the parameters that will be the most useful to estimate to minimize Eq. 1. And, knowing the most useful parameters to estimate provides a systematic way of choosing the parameters to estimate until the system is either identifiable or identifiable enough to be serviceable; in practice serviceable might mean that the errors are within desired tolerances, that parameter estimates are unique, or that the parameter estimates have few enough equilibria or minima that they can be made useful. To make this more precise, begin with the following terms, which are functions of a parameter vector, p.
Feature metric: the feature of the dynamical system we wish to influence, denoted $g(p)$; feature metrics are estimated from the time-series of the simulated model output and vary with parameter variation.
Influence: the amount that a parameter influences the feature-metric, denoted $F_i(g(p))$ for the i-th parameter.
Influence equivalence: a rule that defines equivalence of influence, e.g, all parameters i such that $a_j \leq F_i(g(p)) < a_{j+1}$. This allows for us to introduce a partition over influence, $\{a\}_{j\in J}$ called an *impact set*, which represents the transitions or boundaries between influence equivalence classes.
Parameter influence sets: the sets of parameters with equivalent influence according to the influence equivalence rule.

Demonstrative example: Begin by defining the dynamical system f with state variables x_i and parameters p_i assuming at least four parameters. Next define the *feature-metric* as the mean of a single state variable x_*, μ_{x_*} (i.e. we are interested in how each parameter "influences" the state's mean). Set the *influence function* to be the absolute linear correlation, $|\beta_i|$, between the feature-metric, μ_x, and values of the parameter p_i. In this example, the influence function is a vector-valued function, with a scalar metric (linear correlation between parameter and the state's mean it induces) corresponding to each parameter. The influence per parameter defines a probability mass function (PMF) with support $[0, 1]$ with values $\frac{|\beta_i|}{\sum_j |\beta_j|}$. Finally, we define *influence equivalence* as membership in a given quartile of the PMF defined by the influence function. Note that the *impact set* is defined by the PMF quartiles, and the

influence sets are the parameters in respective quartiles of the PMF. Depending on the separation observed in the impact sets, we could ultimately choose to estimate parameters only from the upper equivalence class(es); i.e. the set of parameters with $|\beta_i|$ in the upper quartile. □

This example takes a narrow interpretation of the flexible construct we develop for identifying equivalence classes of parameter influence. However, even the above example allows for wild topological variation within a given equivalence class. For example, within a given equivalence class one would easily imagine there being many topologically distinct invariant sets due to *both* parameter variation and initial condition variation. Presumably there are other similar equivalence class violations such as ergodicity properties ([98, 99]), $k - LCE$ stability ([100, 101]), etc. These issues can all be addressed by defining the various properties, e.g., the influence function, differently, or more restrictively such that we end up with increasingly more restrictive constructions such as the original notion of bifurcation sets. This flexibility in equivalencies is the point of this construction: we can, depending on our goals, data, etc., have substantial flexibility in how we set up how to choose what parameters to estimate all while explicitly acknowledging *what we know we do not know we are preserving.* For example, if we define the feature-metric to be the mean, we know we are allowing the system to explore or have many different coexisting invariant densities as long as they have a mean that lies within a given equivalence class.

Visual example: Figure 1 shows three cases of the outcome of the Houlihan analysis. The left-most plot in fig. 1 shows the case where the rank-ordering of influence is on a by-coordinate basis; meaning, the equivalence classes were collections entire coordinates, here where each equivalence class has a single member. The middle plot in fig. 1 shows a case where the influence can be portions of different coordinates, but still there are is not joint dependence between variables. The right-most plot in fig. 1 demonstrates an example where the influence equivalence includes joint coordinate relationships. In this paper we will only address the first of these cases, leaving the more complex situations for later work.

D. Computational moving parts for throwing the Houlihan around the right parameters

The computational task of selecting parameters to estimate involves defining the equivalence-like classes, finding their boundaries, rank ordering the parameters by importance and has, broadly, five moving parts. *First*, select the feature-metric(s), $g(p)$, e.g., mean. *Second*, formulate the representation of the space of parameters and their variation, including (i) parameter grid resolution, (ii) parameter perturbation range, (iii) parameter variation type, e.g., joint versus individual by-parameter parameter variation. *Third*, choose an influence function that defines how to model the parameterized variation of the feature-metric variation with parameter variation. *Fourth*, choose a method for rank ordering these parameterizations by influence. Sometimes steps three and four can be done using a single method, e.g., linear regression with a L_1 regularization or by using lasso with cross validation, and sometimes it is done in two steps, e.g., linear regression with a threshold on the β's, partitioning the β's into equivalence classes. And *fifth*, decide which parameters to keep or which equivalence classes, or which impact sets are important.

a. Feature metrics We use two feature metrics, mean and standard deviation of the invariant density generated by mechanistic model with set parameter values and initial conditions.

b. Parameter grid We begin with the nominal parameters ([1, 24, 89]), and then vary them in intervals of $\log_2$ over 10 decades in both directions. For example, for parameter i the parameter grid point for the kth decade was set as $p_i(nominal)2^k$. We did not consider joint-variation of parameters, but varied parameters independently while holding all other parameters fixed at their nominal values.

1. *Parameter selection methods: Influence functions, impact sets, and ranking*

Given a feature metric as a covariate or input vector, e.g., the means of attractor densities for a set of parameter values, we use several methods for selecting the best set of parameters to estimate in a DA context. Some of these methods are stock—linear regression with lasso—some are standard practice—parameter selection using knowledge of the model—and some are modifications of existing methods—see PCA-lariat below. We will see that the method for selecting the parameters matters, although not as much as the feature metric, and it is clear that sophisticated machine learning methods could be useful in this context.

a. Covariates or input vectors All of the methods below take a covariate matrix as input. The covariates correspond to vectors: one dimension of the covariate matrix corresponds to a feature metric calculated at every point along the parameter variation, e.g., the mean of a simulated attractor at every point along a one-dimensional parameter curve.

b. By hand selection parameter selection — parameter selection using knowledge In our previous work we selected parameters to estimate by hand as they were tied to certain dynamical features, physiologic knowledge want to fit

something in particular to solve a problem, e.g., phenotyping. We selected E and V_p because they seemed to have an impact on the mean ([4]) and t_p because it was related to liver function; the results can be found in [1].

c. Automatic parameter selection using linear regression A basic method for determining influence is the linear dependence between the feature metric and parameter variation. In this setting we perform a linear regression between the feature metric and the parameters and we keep all β's for which $\beta_i > (\beta_1)(\kappa_{LR})$. Here we set $\kappa_{LR} = 20\%$ or 0.2, meaning that we keep all the parameters that have a regression coefficient that explains at least 20% of the regression coefficient with the highest influence.

d. Automatic parameter selection using Lasso and cross validation A natural way of reducing the number of parameters in a model is to select parameters that have a lot of power explaining the feature metric while simultaneously being non-redundant. One way of achieving this is to use lasso, or L_1 regularization to enforce a sparse representation of the parameter system ([83, 102–104]). We use the standard lasso formulation ([83]) with cross validation to determine the rank-ordering of parameters; the optimal value of λ, or the optimal number of parameters, is set using a cutoff of one standard error. Lasso automatically and uniquely rank orders parameters. We keep the parameters within one standard error of the minimum mean squared error (MSE) ensuring a sparse representation of the model.

e. Automatic parameter selection using elastic net approximation of ridge regression In addition to lasso regularization, we also use ridge regression, or L_2 regularization ([83, 104, 105]). We compute the ridge regression selected parameters using an elastic nets formulation with α set to 0.0001 where elastic nets formulation approachs L_2 regularization, and select the number of parameters using cross validation in the same way as is done in the lasso setting. We keep the parameters within one standard error of the minimum mean squared error (MSE) ensuring a sparse representation of the model.

f. Automatic parameter selection using PCA-lariat with a single metric To add diversity to the set of methods for selecting parameters beyond linear regression-based methods, we devised a principle component analysis (PCA) ([106–108]) based algorithm for computing an influence function, then implement a rank-ordering scheme for defining influence equivalence. The method we develop, *PCA-lariat*, follows seven steps. *First,* estimate the PCs for the feature-metric, $g(p)$, taking care to de-trend the summary. *Second,* estimate the percentage of the variance captured by the $i-th$ PC, $\sigma_{PC}(i)$. *Third,* identify the *important PCs*, or the PCs that explain variance above a threshold, κ_{PC}; we use 5%. *Fourth,* for each important PC, rank-order the contribution of each parameter or coordinate to the PC. *Fifth,* collect all the coordinates for all the important PCs that contribute proportionally to a given PC above a set threshold, κ_C, $PC_j(i) > \kappa_C$; we use 10%. *Sixth,* for the important parameters for the important PCs, estimate the contribution per parameter:

$$PCR(i) = \sum_j \sigma_{PC}(j) * PC_j(i). \tag{3}$$

And *seventh,* rank order the important parameters by PCR and select the parameters above a given threshold, κ_I; we use 0.1, or 10%.

g. Multi-directional parameter wrangling Combining models, or model averaging can be very useful for improving results ([1, 109–111]), especially when you either know you want to adjust to multiple feature-metrics, or you do not know what feature metrics are important. Here, we only consider using set operations over methods, and consider three cases. First, we take the union of: (number of rank-ordered parameters, feature-metric, influence function) using one parameter per influence function, two feature metrics, mean and standard deviation. Second and third, we take the union of: (number of rank ordered parameters, feature-metric, influence function) using one parameter per influence function and one feature metric, either mean or standard deviation.

E. Evaluation scheme

The evaluation of the Houlihan methods is done in four steps. *First*, we apply the Houlihan methods to the ultradian model to select parameters to estimate and compare the parameter selections as the method is perturbed. *Second*, we use both the UKF and the MCMC DA methods to estimate these Houlihan-selected parameters for the four people in our cohort and calculate the mean squared error (MSE) between the data and the model state estimates (MCMC methods) and forecasts (UKF methods). *Third*, we use both the UKF and the MCMC DA methods to estimate parameters for *both* parameters that were previously chosen by hand in previously published work ([1]) and parameters that the Houlihan methods determined were low-influence parameters and again calculate the MSE between the data and model state estimates and forecasts. *Fourth*, we compare the MSE for the variously selected parameter sets.

Rank-ordered parameters per selection method out of 21 possible parameters											
method	1	2	3	4	5	6	7	8	9	10	11
LASSO μ	a_1	C_1	V_p	t_p	R_m	C_3	—	—	—	—	—
LASSO σ	R_g	C_3	U_m	a_1	C_1	t_p	R_m	V_p	–	—	—
Linear regression μ	a_1	C_1	C_3	R_m	t_p	V_p	U_m	R_g	C_4	U_b	U_0
Linear regression σ	R_g	C_3	U_m	a_1	C_1	R_m	V_p	t_p	k_{decay}	—	—
Ridge regression μ	a_1	—	—	—	—	—	—	—	—	—	—
Ridget regression σ	R_g	—	—	—	—	—	—	—	—	—	—
PCA μ	a_1	C_1	C_3	R_m	t_p	V_p	U_m	—	—	—	—
PCA σ	R_g	C_3	U_m	a_1	C_1	—	—	—	—	—	—

TABLE II: The rank ordering choice of the four parameter selection methods for the feature-metrics mean, μ, and standard deviation, σ.

VI. RESULTS

The results come in two stages. *First*, we present the rank-ordered parameters selected by different methods in order to demonstrate: (i) which parameters the methods selected, (ii) that the methods selected some but not all parameters, (iii) how the parameter selection varied across methods, and (iv) the rank-ordering of parameters by method. *Second,* we evaluate the methods by using the parameters selected in each method to forecast glucose with the UKF and smooth glucose with MCMC; methods are compared via the MSE between measurements and predictions.

A. Parameter selections by method

Table II shows the rank-ordered parameters selected by each parameter selection method. The methods were sensitive to the feature metric; the mean and standard deviation-based methods did not select the same parameters as important.

For a given feature metric, all selection methods identified the same top two parameters — all methods ranked a_1 and C_1 as the top influencers of the mean, and ranked R_g and C_3 as the top influencers of the standard deviation. However, the entire influence sets differed substantially. This indicates that influence set structure, as defined (upper quartile of influence), is sensitive to choices of influence functions and influence equivalence definitions.

Interestingly, the equivalence classes of high and low parameter influence are preserved under perturbations to the influence function. Fig. 2 shows how the l_1, l_2 and PCA-based methods rank-order parameters according to how they influence the mean. While lasso is expected to preserve the ordering with different λ (it fits one-at-a-time), ridge regression also remains robust to variations in the regularization term, λ, adding parameters one at a time.

Most methods find only $5-6$ influential parameters out of 21, greatly reducing the dimension of the parameter space. In all cases, the methods gave an entry point for which parameters to begin estimating; the next question, then, is whether using the Houlihan approach helps to reduce forecasting errors and improve convergence of parameter estimates.

a. Redundancy and influence Our goal is to select parameters to estimate during forecasting and smoothing tasks. We aim to facilitate this goal by identifying small parameter sets that have significant, minimally redundant influence over important dynamical features. Accomplishing this can minimize problems in identifiability, multiple coexisting invariant sets, etc. Fig. 3 visualizes variation of the feature-metric, mean and standard deviation of the invariant density with parameter variation, as well as how the methods partitioned parameters into a high and low-influence equivalence class. It is clear that some variations in some parameters create large shifts in the mean and variance (e.g. a_1), whereas the mean and variance features are far less sensitive to other parameters, like E and t_d.

While the mean and standard deviation are not always influenced by the same parameters, the methods select parameters that have both high influence and relatively orthogonal influence; e.g., in the case of the mean the methods generally select a_1 and R_g first. The low influence parameters, by comparison, are not able to move the mean or standard deviation appreciably and are therefore not able to fully explore the space. Similarly, the low influence parameters are relatively redundant. Following this logic one might predict that estimating alpha and C_2 would lead to the most accurate model estimates while estimating E and t_d would lead to the least accurate model estimates.

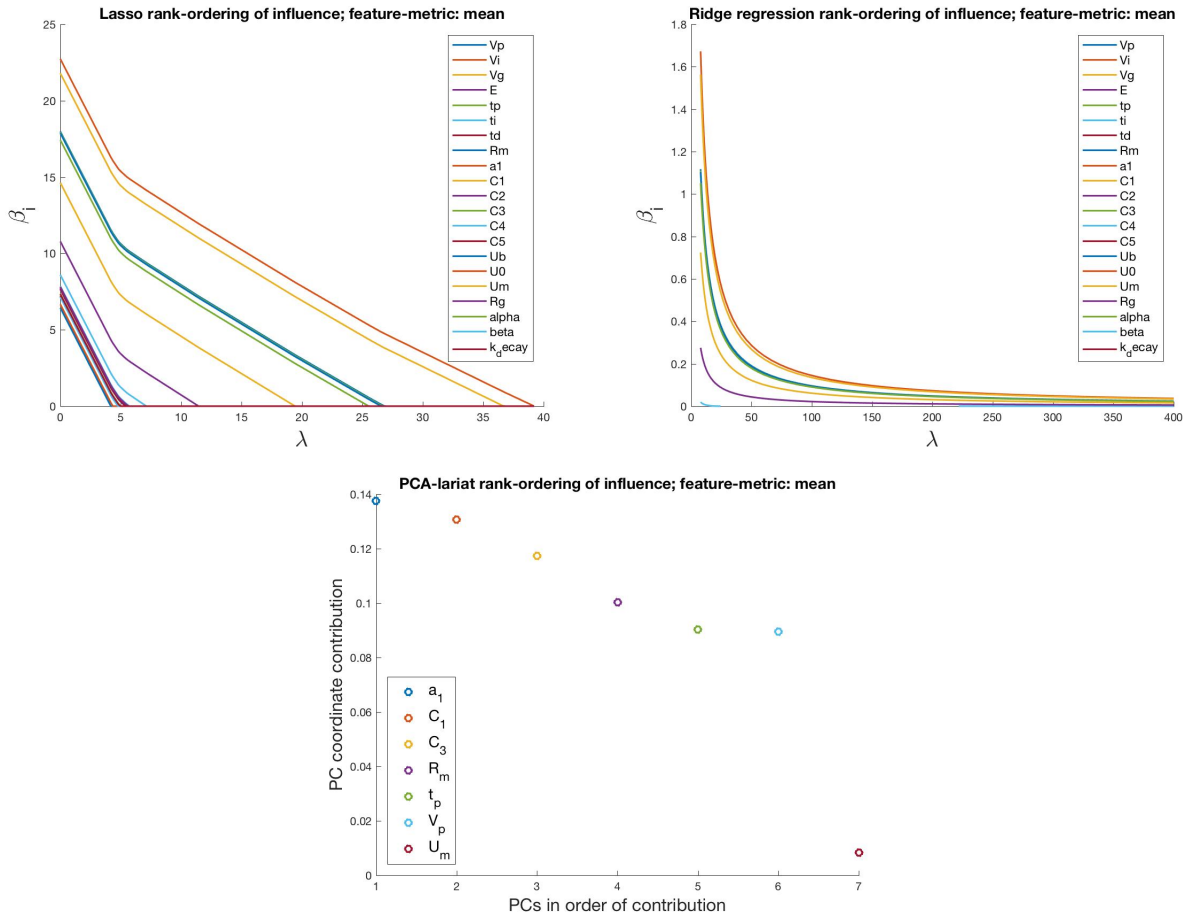

FIG. 2: The rank-ordered influence function with a feature-metric set to the mean for lasso, ridge regression, and PCA-lariat methods.

b. Comparison with by-hand selection In our previous work ([1]) we selected parameters to estimate by hand based on our desire to estimate certain parameters related to physiologic function, e.g., t_p, and because of their obvious influence on parameters, e.g., V_p as could be deduced from other previous work ([4]) to influence the mean state. The automated methods selected V_p and t_p as high influence parameters, but not E, a parameter the methods determined was a low-influence parameter.

B. Parameter selection method evaluation

To evaluate the effectiveness of the machine-selected parameters compared to low-influence parameters as characterized by their β_i's, and the by-hand-selected parameters we used in our previous work, we compare the mean squared error (MSE) between the data and the forecasts for the various parameter combinations as shown in table III. Fig 4 provides a visual summary of the results in table III—the plots are calculated directly from table III—for the MCMC smoothing setting, and demonstrates that all Houlihan-based parameter sets (of any size) noticeably out-performed the by-hand and low-influence parameter sets. Moreover, we see that most Houlihan-based methods achieve similar overall accuracy for parameter sets of cardinality ≤ 3. In addition, Houlihan-based methods that selected parameter sets with 4 or more parameters achieved the best performance, and there is a general trend of improved fit with more parameters—this contrasts sharply with the by-hand parameter selections, whose performance tapered with more than 3 parameters (probably due to unforeseen issues of identifiability).

In particular, lasso chose parameters with the lowest MSE between forecasts and measurements in 7 of 8 cases. In one case, taking the union over methods shared the same MSE with lasso. And, in one case, the lowest MSE was observed with a pair of low-influence parameters. In this case it was the parameter-pair combination, α with E, that mattered. This result implies that generally low influence parameters may, for some people, be physiologically

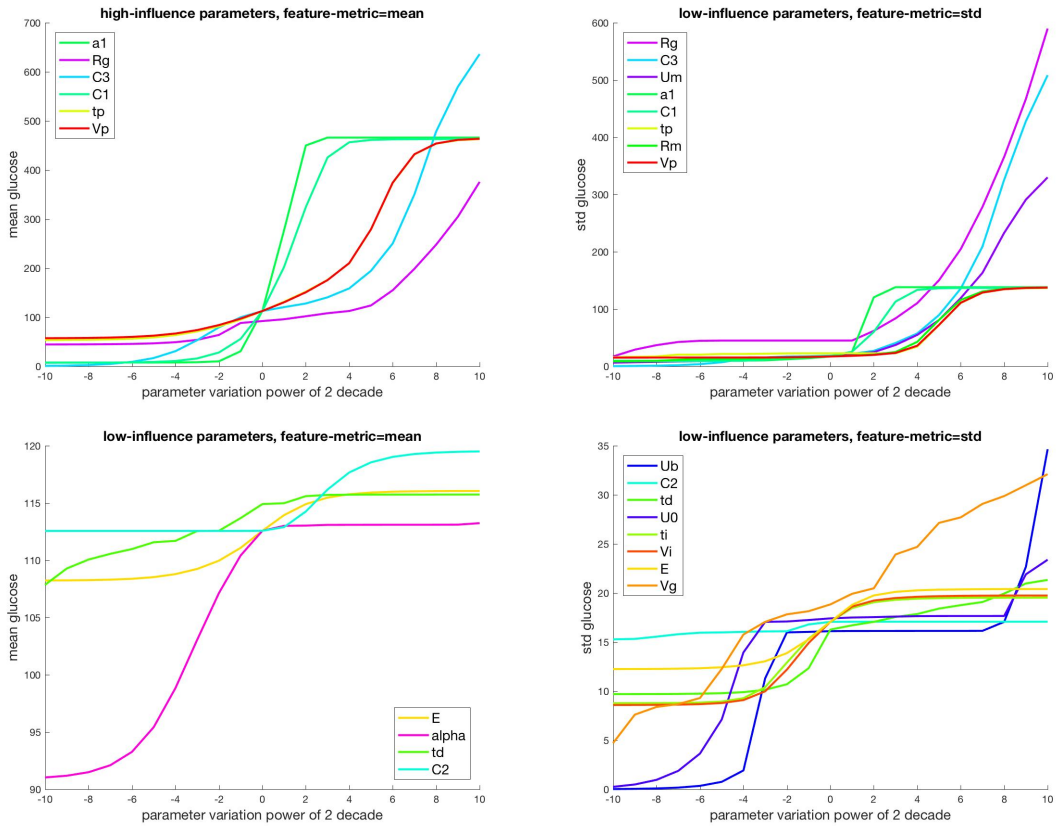

FIG. 3: The influence for two feature metrics, mean and standard deviation, versus parameter variation for high impact and low impact parameters.

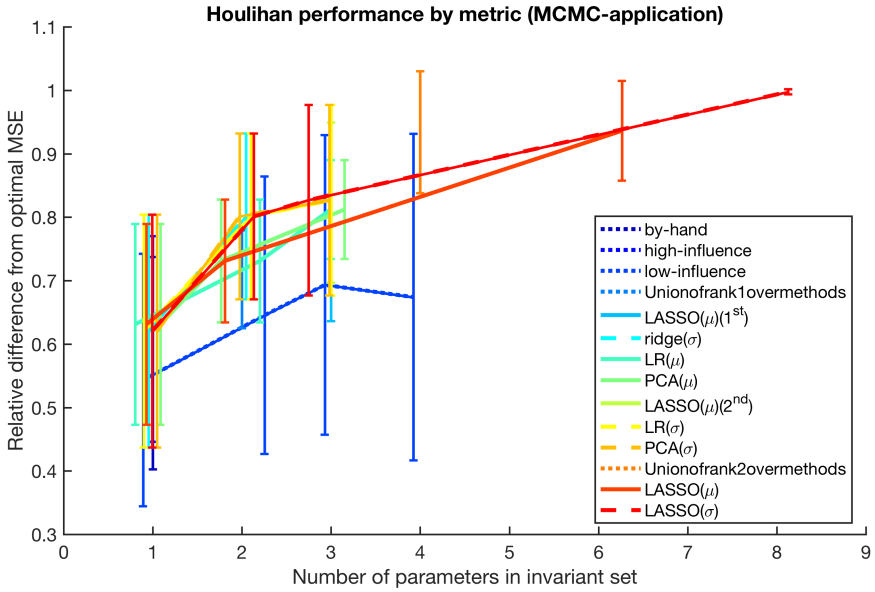

FIG. 4: The overall performance of each method in the smoothing setting. The vertical axis indicates the %-optimal MSE for a given method, averaged over the four patient data sets. Note that methods are labeled as blue to red, where the minimally-performing methods are blue and the maximally-performing methods are red. The plots are estimated directly from the information in Table III.

Rank-ordered parameters per selection method									
	MSE for MCMC				MSE for UKF				
parameter	P1	P2	P4	P5	P1	P2	P4	P5	method-feature-metric pairs
a_1	822	1140	338	296	809	1270	304	356	LASSO(μ), LR(μ), PCA(μ)
R_g	655	1180	475	288	672	1490	470	401	LASSO(σ), LR(σ), ridge(σ), PCA(σ)
t_p	807	1020	448	349	788	1050	407	420	by-hand, high-influence
V_p	820	1120	332	320	805	1300	313	362	by-hand, high-influence
E	681	1250	655	500	721	1380	704	724	by-hand, low-influence
α	501	1250	526	346	526	1580	528	394	low-influence
t_d	530	1080	730	674	NaN	1260	NaN	480	low-influence
Rank-ordered parameter pairs per selection method									
(a_1, C_1)	570	1080	285	276	698	1290	258	330	LASSO(μ), LR(μ), PCA(μ)
(R_g, C_3)	593	923	210	297	613	1260	215	385	LASSO(σ), LR(σ), ridge(σ), PCA(σ)
(a_1, R_g)	578	1130	292	296	614	1400	269	343	Union of rank 1 over methods
(α, E)	454	1174	518	345	483	1310	535	520	low-influence
(α, t_d)	432	993	525	347	NaN	1120	NaN	NaN	low-influence
(E, t_d)	462	1030	592	487	643	1190	NaN	490	low-influence
Rank-ordered parameter 3-tuple per selection method									
(a_1, C_1, V_p)	569	1060	284	276	663	1310	260	329	LASSO(μ) (1st)
(a_1, C_1, t_p)	518	864	247	275	NaN	NaN	234	294	LASSO(μ) (2nd)
(R_g, C_3, U_m)	590	922	190	294	618	1140	228	391	LASSO(σ), LR(σ), PCA(σ)
(a_1, C_1, C_3)	431	1020	261	274	1330	1110	251	340	LR(μ) PCA(μ)
(C_2, E, α)	442	1020	518	346	479	1250	535	515	low-influence
(t_d, C_2, α)	432	894	525	347	NaN	1190	NaN	NaN	low-influence
(t_d, E, α)	398	956	479	343	NaN	1120	NaN	520	low-influence
(t_d, E, C_2)	464	941	592	489	630	1190	NaN	NaN	low-influence
Rank-ordered parameter 4-tuple per selection method									
(a_1, R_g, C_1, C_3)	398	864	182	288	649	985	265	324	Union of rank 2 over methods
Full Houlihan for μ and σ									
$(a_1, C_1, V_p, t_p, R_m, C_3)$	414	862	217	229	661	NaN	236	291	Lasso(μ)
$(R_g, C_3, U_m, a_1, C_1, t_p, R_m, V_p)$	375	863	182	231	632	942	224	289	Lasso(σ)
Method with the lowest MSE									
	Lasso	Lasso	Lasso/Union	Lasso	low-influence	Lasso	Lasso	Lasso	

TABLE III: The mean squared error (MSE) between forecast/smoothed and measured glucose. The machine-based methods, almost always selected the parameter set that achieved the MSE minimum, but for some individuals, certain hand-chosen parameters matter.

important and explore particular pathophysiology necessary to synchronize to the individual. We also know that as the number of parameters increased to 3 $\geq$, some of the MCMC parameter estimates with the lowest MSE found multiple, competing equilibria, were not unique, and sometimes did not fully converge. For example, Fig. 5 shows parameter estimates of two different parameters—one that converges and one that does not—for two parameter sets for P1 with standard deviation as the feature-metric. When lasso-selected parameters are restricted to two parameters for P1, then both parameters, R_g and C_3 converge producing a MSE of 600; C_3 is shown in Fig. 5. In contrast, lasso restricted to the one standard error minimum selects eight parameters, has a lower MSE of 375 but at least one of the parameters, t_p, does not converge well as shown in Fig. 5. This means that as we increased the flexibility, we lowered the MSEs but possibly came at the expense of physiology or convergent parameter estimates.

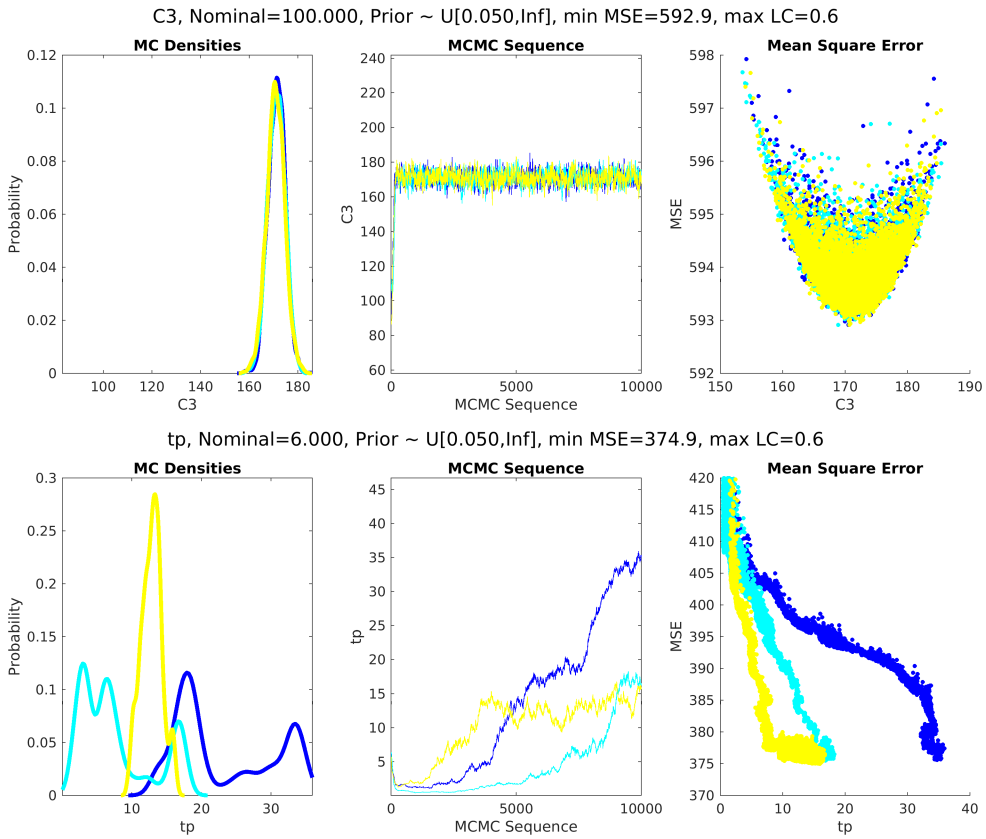

FIG. 5: The posterior densities, Markov chains, and MSE surfaces, for two parameters taken from two sets of parameters. The top set of plots shows C_3 estimates for P1 where lasso is allowed to select two parameters with standard deviation set as the feature metric; C_3 converges well. The bottom set of plots show t_p estimates for P1 for lasso-selected parameters at one standard error minimum—eight parameters are selected in this case—with standard deviation set as the feature metric; t_p does not converge to a unique minimum but has a lower MSE than cases where the parameters are uniquely identified..

VII. DISCUSSION

Summary Our most broad conclusion is that the machine-selected parameters work better than hand-selected parameters and that the Houlihan methods are a scalable method for selecting which parameters of a mechanistic model to estimate using DA methods. This means that stacking machine learning techniques on top of, or together with, DA is a helpful strategy, especially when models are complex and data are sparse, as in our glucose modeling example.

Houlihan methods We intuitively define the Houlihan method(s) as a collection of methods for selecting the most productive model parameters to estimate with machine learning techniques applied to simulated model output under parameter variation subject to a set of features, e.g., the mean of a state.

using machine learning to

Feature metric selection matters: For all methods, the feature metric (mean or standard deviation) was the first-order driver of differences in parameter rank orderings. This choice is highly problem-dependent. In some biomedical applications, sensitivity of the mean to parameter perturbation is not especially important for a good fit; e.g., there are physiologic systems where variation in the mean across people is small, but excursions, peaks, number of peaks, location of peaks, etc., may be a more important types of features to capture.

The cutoff matters: The cutoff for influence has a substantial impact on the ability to estimate parameters. For example, lasso-selected parameters usually minimized MSE, but the induced MSE and MCMC convergence were both sensitive to the influence cutoff. All the methods had this sensitivity, and estimating optimal cutoffs automatically would be beneficial.

The selection method sometimes matters: For the high-ranked parameter choices, the feature-metric was the primary difference between selected parameter sets. However, as the number of parameters included was increased, the methods diverged. We suspect that as the complexity of feature metrics and ranking methods increases, e.g.,

using nonlinear regressions, there will be more sensitivity of the parameter selections to the methods.

Physiology matters: We know from carefully considering the convergence properties of the MCMC chains that some of the lowest MSEs for the runs with three or more parameters didn't converge well. Meaning, as we increased the flexibility, we lowered the MSE but possibly at the expense of physiologic fidelity or convergent parameter estimates. For pure forecasting applications this may or may not matter, but when we want the parameters to be meaningful, we need the parameter estimates to converge, not necessarily to *a* unique set of parameters, but to distinctly different parameter estimates that can be treated as hypotheses. Another problem that can arise because of physiology is that different people with different physiology can be sensitive to different parameters. For example, the physiological feature that is important to personalize the model for a particular person may not be related to the properties captured by the feature metric, e.g., the mean, and in this circumstance parameters identified as low influence relative to the feature-metric will not be estimated. A potential example of this is P1, for whom estimating α and E achieved the lowest MSE despite E and α being low-influence parameters relative to both the mean and standard deviation.

Effective parameter space exploration: Abstractly, a mechanistic model is a parameterized family of functions whose parameters, depending on the model, have varying degrees of independence. From this perspective, the goal of the Houlihan methodology is to find a way to explore the maximal amount of the parameter space while minimizing the redundancy between parameters. The feature-metrics and the influence functions define which subsets of the parameter space are most useful to open for exploration, which in turn defines which dynamics can be explored. For example, focusing on variations of the mean may close off other dynamical features such as amplitude variations or any feature that is not uniquely defined by the variation of the mean. We do not yet have a good method for understanding how a feature-metric may influence other, potentially valuable explorations. We acknowledge that understanding and quantifying how limited feature-metrics influence the effective parameter space of a model is an important, unexplored problem.

Computational complexity: We consider only the case here where we vary any one single parameter while leaving all other parameters fixed at their nominal values; this means that the dimension of the input for regression used to select the most useful parameters scales linearly in the number of parameters. If we were to co-vary parameters, meaning if varied all parameters at once, depending on how one choose to partition the parameter space, the computational complexity would explode. In this way, the framework we present here does not solve the computational complexity problem of exploring parameter space. Instead, the results in this paper show that even by only considering feature-metric variation along one-dimensional subspaces of the full parameter space we can gain substantial insight into which parameters have the most impact on the features we are interested in approximating. Moreover, we can also see the limitations of this approach — we do observe synergy between parameters where combinations of some low-influence parameters for some people can end up having a high influence on the model fit.

Obvious extensions: In this paper, we stack machine learning on top of DA, which has many potential extensions. Feature-metrics could be generalized to be multi-dimensional both over states and over types of feature-metrics. Feature selection methods could be developed or employed to select feature metrics. The estimates of influence could be calculated to include jointly varying parameters—this would be computationally expensive and would require computational innovation in high-dimensional settings, cf the computational complexity discussion above. Moreover, this problem is not necessarily a simple extension because the parameter spaces of mechanistic models are not likely to form a basis for the model space, in contrast to the parameters of the space of polynomials which do form a basis. Of course this lack of a basis structure is part of the problem—parameters of mechanistic models and likely the physiology they represent are redundant, likely for biological reasons such as robustness. We use linear regression and PCA-based machine learning methods; it is likely that more sophisticated machine learning methods e.g., full elastic nets, support vector machines, deep learning, sparse machine learning (compressed sensing), Bayesian methods, model averaging and ensemble learning could all be used and would likely improve the parameter selections. Similarly, further stack of machine learning techniques on top of the Houlihan methods would likely be productive. For example, greedy, Gibbs-sampling-like rotation between sets of parameters that are identifiable and explore different subsets of the parameter space could minimize both model errors and identifiability issues. And finally, feature-metrics could be made substantially more sophisticated, insightful and tailored to circumstance or physiologic knowledge, such as preserving power in certain frequency bands. More sophisticated feature metrics could also be used to gain insight into potentially meaningful constraints on parameters for use in operational DA.

VIII. CONCLUSION

We devised a methodology for rank-ordering parameters of a mechanistic model and using this rank-ordering to select an effective subset of parameters to estimate when projecting biomedical data onto the model via data assimilation. This methodology specifically targets parameter sets that avoid issues of model identifiability and parameter-estimation convergence problems, improving forecasting and phenotyping performance of data assimilation

methods that use mechanistic biological models. Using machine learning to select parameters to estimate worked: the machine-chosen parameters reduced the mean-squared error between estimates and forecasts and data in nearly all cases by factors as large as three. These results imply that combining mechanistic and non-mechanistic machine learning could be a particularly productive direction of future research and could greatly aid in our ability to use computational machinery to both help deepen our physiologic understanding and help clinicians achieve more positive outcomes in clinical settings.

Appendix A: Ultradian model

The model is comprised of a set of six ordinary differential equations; the model is non-autonomous because it has an external, time-dependent driver, consumed nutrition. The six dimensional state space made up of three physiologic variables and a three stage filter. The physiologic state variables are the glucose concentration G, the plasma insulin concentration I_p, and the interstitial insulin concentration I_i. The three stage filter (h_1, h_2, h_3) which reflects the response of the plasma insulin to glucose levels [89]. The model was designed to capture ultradian oscillations missing in previous models. The ordinary differential equations that define the model are [24]:

$$\frac{dI_p}{dt} = f_1(G) - E\big(\frac{I_p}{V_p} - \frac{I_i}{V_i}\big) - \frac{I_p}{t_p} \tag{A1}$$

$$\frac{dI_i}{dt} = E\big(\frac{I_p}{V_p} - \frac{I_i}{V_i}\big) - \frac{I_i}{t_i} \tag{A2}$$

$$\frac{dG}{dt} = f_4(h_3) + I_G(t) - f_2(G) - f_3(I_i)G \tag{A3}$$

$$\frac{dh_1}{dt} = \frac{1}{t_d}\big(I_p - h_1\big) \tag{A4}$$

$$\frac{dh_2}{dt} = \frac{1}{t_d}\big(h_1 - h_2\big) \tag{A5}$$

$$\frac{dh_3}{dt} = \frac{1}{t_d}\big(h_2 - h_3\big) \tag{A6}$$

The state variables include physiologic processes that have been parameterized, including: $f_1(G)$ represents the rate of insulin production; $f_2(G)$ represents insulin-independent glucose utilization; $f_3(I_i)G$ represents insulin-dependent glucose utilization; $f_4(h_3)$ represents delayed insulin-dependent glucose utilization. These functions are defined by:

$$f_1(G) = \frac{R_m}{1 + \exp(\frac{-G}{V_g c_1} + a_1)} \tag{A7}$$

$$f_2(G) = U_b(1 - \exp(\frac{-G}{C_2 V_g})) \tag{A8}$$

$$f_3(I_i) = \frac{1}{C_3 V_g}(U_0 + \frac{U_m - U_0}{1 + (\kappa I_i)^{-\beta}}) \tag{A9}$$

$$f_4(h_3) = \frac{R_g}{1 + \exp(\alpha(\frac{h_3}{C_5 V_p} - 1))} \tag{A10}$$

$$\kappa = \frac{1}{C_4}(\frac{1}{V_i} - \frac{1}{E t_i}) \tag{A11}$$

The nutritional driver of the model $I_G(t)$ is defined over N discrete nutrition events [4], where k is the decay constant and event j occurs at time t_j with carbohydrate quantity m_j

$$I_G(t) = \sum_{j=1}^{N} \frac{m_j k}{60} \exp(k(t_j - t)); N = \#\{t_j < t\} \tag{A12}$$

[1] D. Albers, M. Levine, B. Gluckman, H. Ginsberg, G. Hripcsak, and L. Mamykina, PloS Comp Bio **13**, e1005232 (2017).

TABLE IV: Full list of parameters for the ultradian glucose-insulin model [24]. Note that IIGU and IDGU denote insulin-independent glucose utilization and insulin-dependent glucose utilization, respectively.

Ultradian model parameters		
Name	Nominal Value	Meaning
V_p	3 l	plasma volume
V_i	11 l	interstitial volume
V_g	10 l	glucose space
E	0.2 l min^{-1}	exchange rate for insulin between remote and plasma compartments
t_p	6 min	time constant for plasma insulin degradation (via kidney and liver filtering)
t_i	100 min	time constant for remote insulin degradation (via muscle and adipose tissue)
t_d	12 min	delay between plasma insulin and glucose production
k	0.5 min^{-1}	rate of decayed appearance of ingested glucose
R_m	209 mU min^{-1}	linear constant affecting insulin secretion
a_1	6.6	exponential constant affecting insulin secretion
C_1	300 mg l^{-1}	exponential constant affecting insulin secretion
C_2	144 mg l^{-1}	exponential constant affecting IIGU
C_3	100 mg l^{-1}	linear constant affecting IDGU
C_4	80 mU l^{-1}	factor affecting IDG
C_5	26 mU l^{-1}	exponential constant affecting IDGU
U_b	72 mg min^{-1}	linear constant affecting IIGU
U_0	4 mg min^{-1}	linear constant affecting IDGU
U_m	94 mg min^{-1}	linear constant affecting IDGU
R_g	180 mg min^{-1}	linear constant affecting IDGU
α	7.5	exponential constant affecting IDGU
β	1.772	exponent affecting IDGU

[2] D. Albers, G. Hripcsak, and M. Schmidt, PLoS One **7**, e480058 (2012).
[3] Y. Xu, Y. Xu, and S. Saria, in *Proceedings of the 1st Machine Learning for Healthcare Conference*, edited by F. Doshi-Velez, J. Fackler, D. Kale, B. Wallace, and J. Weins (PMLR, Northeastern University, Boston, MA, USA, 2016), vol. 56 of *Proceedings of Machine Learning Research*, pp. 282–300, URL `http://proceedings.mlr.press/v56/Xu16.html`.
[4] D. Albers, N. Elhadad, E. Tabak, A. Perotte, and G. Hripcsak, PLoS One **6**, e96443 (2014).
[5] G. Hripcsak and D. J. Albers, Journal of the American Medical Informatics Association p. ocx110 (2017).
[6] J. Pathak, A. Kho, and J. Denny, J Am Med Inform Assoc. **20**, e206 (2013).
[7] R. Pivovarov, A. Perotte, E. Grave, J. Angiolillo, C. Wiggins, and N. Elhadad, Journal of Biomedical Informatics (2015).
[8] Y. Halpern, Y. Choi, S. Horng, and D. Sontag, JAMIA (2016).
[9] A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, P. J. Liu, X. Liu, M. Sun, P. Sundberg, H. Yee, et al., CoRR **abs/1801.07860** (2018), 1801.07860, URL `http://arxiv.org/abs/1801.07860`.
[10] D. T. Westwick and R. E. Kearney, *Identification of nonlinear physiological systems* (IEEE Engineering in Medicine and Biology, 2003).
[11] L. Ljung, *System Identification* (Prentice Hall, 1987).
[12] J. Schoukens, M. Vaes, and R. Pintelon, IEEE Control Systems pp. 38–69 (2016).
[13] M. Levine, D. Albers, and G. Hripcsak, in *Annual Symposium Proceedings* (AMIA, 2016).
[14] M. Levine, D. Albers, and G. Hripcsak (2018), arXiv:1801.08929.
[15] G. Hripcsak, D. Albers, and A. Perotte, JAMIA **18**, 109 (2011).
[16] G. Hripcsak and D. Albers, JAMIA **0**, 1 (2013).
[17] A. Perotte, R. Ranganath, J. Hirsch, D. Blei, and N. Elhadad, JAMIA **22**, 872 (2015).
[18] R. Ranganath, A. Perotte, N. Elhadad, and D. Blei, in *Proceedings of Machine Learning for Healthcare* (2016), vol. 56.
[19] T. Lasko, J. Denny, and M. Levy, PLOS One (2013).

[20] K. Hornik, M. Stinchocombe, and H. White, Neural Networks **2**, 359 (1989).
[21] K. Hornik, M. Stinchocombe, and H. White, Neural Networks **3**, 551 (1990).
[22] G. Hripcsak and D. Albers, JAMIA **10**, 1 (2012).
[23] R. Pivovarov, D. Albers, J. Sepulveda, and N. Elhadad, Journal of Biomedical Informatics (2014).
[24] J. Keener and J. Sneyd, *Mathematical physiology II: Systems physiology* (Springer, 2008).
[25] M. Brin and G. Stuck, *Introduction to Dynamical Systems* (Cambridge University Press, 2004).
[26] J. Guckenheimer and P. Holmes, *Nonlinear Oscillaions, Dynamical Systems, and Bifurcations of Vector Fields* (Springer-Verlag, New York, 1983).
[27] D. K. Arrowsmith and C. M. Place, *An introduction to dynamical systems* (Cambridge University Press, 1990).
[28] V. I. Arnold, *Ordinary differential equations* (Springer-Verlag, 1992).
[29] V. Arnold, *Geometric methods in the theory of ordinary differential equations.*, Grundlehren de mathematischen Wissenschaften (Springer-Verlag, 1983), 2nd ed.
[30] D. Albers, L. Levine, B. Gluckman, G. Hripcsak, L. Mamykina, and A. Stuart (2018), in revision.
[31] A. Jazwinski, *Stochastic processes and Filtering Theory* (Dover, 1998).
[32] A. Lorenc, Q. J. R. Meterol. Soc. **112**, 1177 (1988).
[33] K. Law, A. Stuart, and K. Zygalakis, *Data assimilation* (Springer, 2015).
[34] M. Ash, M. Bocquet, and M. Nodet, *Data assimilation: methods, algorithms and applications* (SIAM, 2016).
[35] S. Reich and C. Cotter, *Probabilistic forecasting and Bayesian data assimilation* (Cambridge University Press, 2015).
[36] A. Haug, *Baysian estimation and tracking* (Wiley, 2012).
[37] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: particle filters for tracking and applications* (Artech house, 2004).
[38] J. Candy, *Bayesian signal processing: classical, modern, and particle filtering methods* (Wiley, 2009).
[39] G. Evensen, *Data assimilation, the ensemble kalman filter* (Springer, 2009).
[40] G. Evensen, Ocean Dynamics (2003).
[41] A. Stuart, Acta Numerica **19**, 451 (2010).
[42] S. Zenker, J. Rubin, and G. Clermont, PLoS Comput Biol **3** (2007).
[43] H. T. Banks, S. Hu, , and W. C. Thompson, *Modeling and inverse problems in the presence of uncertainty* (CRC Press, 2014).
[44] Y. Hirata, N. Bruchovsky, and K. Aihara, Journal of Theoretical Biology, **264**, 517 (2010).
[45] Y. Hirata, M. de Bernardo, N. Bruchovsky, and K. Aihara, CHAOS **20**, 0451251 (2010).
[46] S. Schiff, *Neural control engineering: The emerging intersection between control theory and neuroscience* (MIT Press, 2011).
[47] V. Dukic, H. Lopes, and N. Polson, Journal of the American Statistical Association **107**, 1410 (2012).
[48] H. Miao, X. Xia, A. Perelson, and H. Wu, SIAM Review **53**, 3 (2011).
[49] F. Chee and T. Fernando, *Closed-loop control of blood glucose* (Springer, 2007).
[50] P. B. P, M. Orsini, and MMBenedetti, Artif Cells Blood Substit Immobil Biotechnol. **2**, 127 (2003).
[51] P. Fabietti, V. Canonico, M. Orsini-Federici, E. Sarti, and M. Massi-Benedetti, Diabetes Technol Ther. **4**, 327 (2007).
[52] B. Kovatchev, M. Breton, C. Man, and C. Cobelli, J Diabetes Sci Technol **3**, 44 (2009).
[53] M. A. F. A. B. Coutinho, R. S. Azevedo, M. N. Burattini, L. F. Lopenz, and E. Massad, Phys. Rev. E **67**, 051907 (2003).
[54] M. Mirowski, P. Reid, M. Mower, L. Watkins, V. Gott, J. Schauble, A. Langer, M. Heilman, S. Kolenik, R. Fischell, et al., New England Journal of Medicine **303**, 322 (1980).
[55] H. Thabit, M. Tauschman, J. Allen, L. Leelarathna, S. Hartnell, M. Wilinska, C. Acerini, S. Dellweg, C. Benesch, L. Heinemann, et al., New England Journal of Medicine (2015).
[56] L. Leelarathna, S. W. English, H. Thabit, K. Caldwell, J. M. Allen, K. Kumareswaran, M. E. Wilinska, M. Nodale, J. Mangat, M. L. Evans, et al., Crit Care **17**, R159 (2013).
[57] C. Cobelli, C. Man, G. Sparacino, L. Magni, G. De Nicolao, and B. Kovatchev, IEEE Reviews in Biomedical Engineering **2**, 54 (2009).
[58] H. Thabit, M. Tauschman, J. M. Allen, L. Leelarathna, S. Hartnell, M. E. Wilinska, C. L. Acerini, S. Dellweg, C. Benesch, L. Heinemann, et al., N. Engl. J. Med. (2015).
[59] L. Glass and M. Courtemanche, *Cardiac Arrhythmias and Device Therapy: Results and Perspectives for the New Century* (Futura, 2000), chap. Control of atrial fibrillation: A theoretical perspective, pp. 87–94.
[60] D. Christini and L. Glass, Chaos **12**, 732 (2002).
[61] K. Hall, D. J. Christini, M. Tremblay, J. J. Collins, L. Glass, and J. Billette, Phys. Rev. Lett. **78**, 4938 (1997).
[62] M. Mackey and L. Glass, Science **197**, 287 (1977).
[63] R. S. Parker, F. J. D. III, J. F. Ward, and N. A. Peppas, AIChE Journa **46**, 2537 (20xx).
[64] R. S. Parker, F. J. D. III, , and N. A. Peppas, IEEE Engineering in Medicine and Biology **20**, 65 (2001).
[65] S. Donnet and A. Samson, Advanced Drug Delivery Reviews (2013).
[66] P. Bonate, AAPS (2005).
[67] M. Sadean and P. Glass, Curr Opin Anaesthesiol (2009).
[68] N. R. Kristensen, H. Madsen, , and S. H. Ingwersen, XXXX **32** (2005).
[69] M. Sedigh-Sarvestan, D. Albers, and B. Gluckman, in *34th Annual International IEEE EMBS Conference* (2012).
[70] J. F. Selgrade, L. A. Harris, and R. D. Pasteur, Journal of Theoretical Biology **260**, 572 (2009).
[71] M. Sedigh-Sarvestani, S. J. Schiff, and B. J. Gluckman, PLoS Comput Biol **8** (2012).
[72] J. Lin, J. G. Chase, G. M. Shaw, C. V. Doran, C. E. Hann, M. B. Robertson, P. M. Browne, T. Lotz, G. C. Wake, , et al.,

in *34th Annual International IEEE EMBS Conference* (2004).
[73] J. Lin, N. N. Razak, C. G. Pretty, A. L. Compte, P. Docherty, J. D. Parente, G. M. Shaw, C. E. Hann, , and G. Chase, Computer Methods and Programs in Biomedicine **102**, 192 (2011).
[74] M. Levine, G. Hripcsak, L. Mamykina, A. Stuart, and D. Albers (2017), arXiv:1709.00163.
[75] E. Hines, T. Middendorf, and R. Aldrich, J Gen. Physiol **143**, 401 (2014).
[76] J. Simens, M. Cree-Green, B. Bergman, K. Nadeau, and C. D. Behn, in *Association of Women in Mathematics Annual Symposium* (2017).
[77] M. Eisenberg and M. Hayashi, Math Biosciences **256**, 116 (2014).
[78] O. Walch and M. Eisenberg, Neurocomputing **199**, 137 (2016).
[79] A. Brouwer, R. Meza, and M. Eisenberg, Risk analysis (2018).
[80] O.-T. Chis, J. Banga, and E. Balsa-Canto, PLoS ONE **6**, e27755 (2011).
[81] T. M. Hamill, XXXX (2006).
[82] Y. Kuznetzov, *Bifurcation Theory* (Springer-Verlag, 1998), 2nd ed.
[83] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical learning with sparsity* (CRC, 2015).
[84] L. Mamykina, M. Levine, P. Davidson, A. Smaldone, N. Elhadad, and D. Albers, J Am Med Inform Asso (2015).
[85] L.-S. Young, J. Stat. Phys. **108**, 733 (2002).
[86] D. Ruelle, Amer. J. Math **98**, 619 (1976).
[87] Y. Sinai, Russian Math. Surveys **27**, 21 (1972).
[88] R. Bowen, *Equilibrium states and teh ergodic theory of Anosov diffeomorphisms*, vol. 470 of *Lect. Notes in Math.* (Springer-Verlag, Berlin, 1975).
[89] J. Sturis, K. S. Polonsky, E. Mosekilde, and E. V. Cauter, Am J Physiol Endocrinol Metab **260**, E801 (1991).
[90] S. Julier and J. Uhlmann, Proc. IEEE **92**, 401 (2004).
[91] S. Julier, J. Uhlmann, and H. Durrant-Whyte, in *American Control Conference*, IEEE (IEEE, 1995).
[92] E. Want and R. Merwe, in *Adaptive Systems for Signal Processing, Communications, and Control Symposium*, IEEE (Wiley, 2000), pp. 153–158.
[93] J. Gove and D. Hollinger, J Geophys Res **111**, DO8S07 (2006).
[94] E. Want, R. Merwe, and A. Nelson, in *NEURAL INFORMATION PROCESSING SYSTEMS* (MIT Press, 2000), pp. 666–672.
[95] E. A. Wan and R. V. D. Merwe, in *Kalman Filtering and Neural Networks* (Wiley, 2001), pp. 221–280.
[96] S. L. Cotter, G. O. Roberts, A. M. Stuart, and D. White, Statist. Sci. **28**, 424 (2013).
[97] J. Sotomayor, in *Dynamical Systems* (1973), pp. 549–560.
[98] C. Pugh and M. Shub, Trans. Amer. Math. Soc. **312**, 1 (1989).
[99] K. Burns and A. Wilkinson, Annals of Mathematics (2010).
[100] D. J. Albers and J. C. Sprott, Nonlinearity **19**, 1801 (2006).
[101] D. J. Albers, J. C. Sprott, and J. P. Crutchfield, Phys. Rev. E **74**, 057201 (2006).
[102] R. Tibshirani, Journal of the Royal Statistical Society, Series B **58**, 267 (1996).
[103] L. Breiman, Technometrics **37**, 373 (1995).
[104] H. Zou and T. Hastie, Journal of the Royal Statistical Society, Series B pp. 301–320 (2005).
[105] M. Jaggi, *An Equivalence between the Lasso and Support Vector Machines.* (Chapman and Hall/CRC, 2014).
[106] K. Pearson, Philosophical Magazine **2**, 559 (1901).
[107] H. Hotelling, J. Ed. Psych. (1935).
[108] I. Jolliffe, *Principal Component Analysis* (Springer, 2010).
[109] J. Hoeting, D. Madigan, A. Raftery, and C. Volinsky, Statistical Science **14**, 382 (1999).
[110] A. Raftery, T. Gneiting, F. Balabdaoui, and M. Polakowski, Monthly Weather Review **133**, 1155 (2005).
[111] G. Claeskens and N. Hjort, *Model selection and model averaging* (Cambridge Univesity Press, 2008).