Landmark-based Emulation for Models with Misaligned Functional Response

Devin Francom¹, Bruno Sansó², Ana Kupresanin³ ¹Los Alamos National Laboratory ²University of California Santa Cruz ³Lawrence Livermore National Laboratory

November 1, 2019

Abstract

Many computer models output functional data, and in some cases, these functional data have similar, but misaligned, shape characteristics. We introduce a general approach for building emulators for computer models that output misaligned functional data when key values in the functional response (landmarks) can be easily identified. This approach has two main parts: modeling the aligned (using the landmarks) functional data, and modeling the functions that map the misaligned data to the aligned space (warping functions). As the warping functions are required to be monotonic, we give special attention to modeling monotonic functional response data. We discuss how our approach can be used for a variety of typical emulators, such as Gaussian processes, Bayesian multivariate adaptive regression splines, and Bayesian additive regression trees, and how sensitivity analysis can be performed. We demonstrate these approaches to build emulators for a high-energy-density physics computer model used to simulate inertial confinement fusion ignition experiments.

Keywords: Emulator, functional data analysis, registration, warping, computer experiment, monotonicity

technometrics tex template (do not remove)

1 Introduction

When a computer model is expensive to evaluate, a fast statistical surrogate model, or emulator, is often used if a large number of model evaluations is required. The emulator is built by modeling the input-output relationship between a relatively small (or easily obtainable) set of model runs. Assuming the emulator is sufficiently accurate, it can be used in place of the computer model for calculations of interest in uncertainty quantification, such as sensitivity analysis (Saltelli et al., 2008) and calibration (Kennedy and O'Hagan, 2001). The Gaussian process has been a popular emulator because it can be flexible, it can interpolate the training model runs, and its uncertainty away from training data is easy to quantify (Santner et al., 2003). However, other approaches, such as Bayesian additive regression trees (BART) and Bayesian multivariate adaptive regression splines (BMARS) have also proven useful for emulation in various cases (Pratola and Higdon, 2016; Chakraborty et al., 2013).

Computer models frequently have highly multivariate response. When highly multivariate data are indexed by a few variables, such as time and spatial location, functional data analysis methods can be useful for inferring quantities of interest, such as functional or non-functional parameters. A simple, but typical, example is when the output from the model corresponds to realizations of a curve are used to infer a mean curve. It is sometimes the case that curves have the same general shape, but that they are offset, stretched or contracted so that recognizable features of the curves occur at different points of the indexing variable. For instance, Figure 1 shows five curves that have exactly the same shape with a single peak, but where the location of each peak is different. Taking the pointwise average of the curves results in a curve with a single peak that is smaller than all of the peaks in the data, which is unlikely to be a reasonable representation of the average of the functional data. Functional data analysis methods have adopted techniques to align curves (and higher dimensional objects) so that inference methods are not biased by misalignement of data (see Chapter 7 of Ramsay and Silverman (2005) for details).

While these functional data analysis methodologies have primarily been motivated by inference, we are more interested in prediction. We would like to be able to predict functional data in its native, misaligned space, based on a number of non-functional covariates. Further, the applications we are interested in necessitate that this prediction be more flex-



Figure 1: Misaligned functional data are shown, along with their pointwise mean. Alignment before taking the pointwise mean would produce a curve that would likely be more informative. Similarly, modeling functional data using covariates is likely to be easier when the data are aligned.

ible than traditional linear models. These applications focus on building fast statistical surrogate models for expensive computer simulators used for understanding many domains of science and engineering. We will particularly focus on a high-energy-density physics computer model used to simulate inertial confinement fusion ignition experiments that take place on the National Ignition Facility (NIF) at Lawrence Livermore National Laboratory. The goal of these experiments is to achieve nuclear fusion by compressing and heating a fuel capsule using high-energy lasers. The simulator, called HYDRA, takes nine inputs describing the laser stimulus (shape, amount, and timing) given to the capsule as well as the density of the fuel in the capsule (Peterson et al., 2017). The simulator outputs many quantities, but we are particularly interested in the energy production rate over time. Figure 2 shows the log of the simulator output from five different combinations of the nine inputs. The goal is to use an ensemble of nearly 25000 such simulations to build an emulator that, given a new set of inputs, can produce accurate predictions and quantify prediction uncertainty. This problem is challenging because of the degree of misalignement of the functional responses, the large number of curves, and the non-trivial number of inputs.

Approaches to nonlinear modeling of misaligned functional responses based on covariates are limited. Hung et al. (2015) build a Gaussian process emulator, including the



Figure 2: Five curves output from five runs of the HYDRA simulator with different input settings.

functional response using a Kronecker covariance structure and handling the misalignment with missing data methods. Missing data methods, where the functional response is imputed for the entire range of possible functional responses, are unlikely to be useful here because of the large variability in the misalignment of the responses, as shown in Figure 2. Further, missing data methods will not help to align important features of the functional response. Without such alignment, even very flexible models may not perform well, as the parameter relationship that leads to the misalignment of features can be complex and difficult to disentangle from parameter influence on curve shape. Hence, other approaches center on registration, which is the process of aligning functional data. When only the aligned functions are of interest, the data can be registered as a preprocessing step, as in Bayarri et al. (2007). A model built from registered data would be limited to prediction of aligned curves only, unless a registration model was also built.

The prospect of combining an aligned functional data model with a registration model is of interest to functional data analysis researchers more generally. For instance, Earls et al. (2017) develop a model that, for a functional dataset without covariates, uses Gaussian processes to simultaneously infer the aligned mean curve and the warping function, which is the function that maps the unregistered data to be registered (i.e., warps the misalgined data to an aligned space). Telesca (2015) also considers, for the purposes of inference and not prediction, a linear mixed model for both the aligned data and the warping functions. A good deal of attention has been given to modeling of warping functions for inference purposes (Gervini and Gasser, 2004; Tucker et al., 2013), but attention has not been given to modeling warping functions with covariates, especially in a nonlinear fashion. A warping function must be non-decreasing, and modeling with monotonicity constraints can be complicated. Further, joint modeling of the warped data and the warping functions (e.g., imagine amplitude and phase variability) can lead to identifiability issues, since overfitting of the warping functions (which represent phase variability) is detrimental to the modeling of the warped data (which represent amplitude variability).

While many methods for registration have been proposed, landmark registration is preferred when possible (Gervini and Gasser, 2004). Landmark registration consists of aligning functions according to where important functional traits (landmarks) occur. For instance, local maxima, minima, or inflection points may be considered landmarks. Landmarks correspond to important points along the warping functions, meaning that landmarks can be used to learn and constrain warping functions. In a sense, landmarks are the features that need to be aligned, where the values in between landmarks may be less important. The difficulty with landmark registration is that landmarks can sometimes be difficult to select. Hence, methods that do not depend on landmarks are more common. Recent work by Strait and Kurtek (2016) develops a method to estimate landmark placement with uncertainty. In the case of the HYDRA simulations, there are a few clear landmarks occurring in each curve. These are the start, end, global maximum, and point at which the curve changes direction between the start and global maximum. If we consider warping functions that are linear between the landmarks, Figure 3 shows what the aligned curves look like. For comparison, Figure 3 also shows what the aligned curves would look like with only the start and end points as landmarks. Variation near the excluded landmarks is likely to require the model for the aligned curves to be more complex, and justifies the inclusion of all four landmarks since they are easily identified. Figure 4 shows the (inverse) warping functions using four landmarks.



Figure 3: The five curves from Figure 2 warped in two different ways. The left panel shows landmark warping with two landmarks, while the right uses four landmarks. Dotted vertical lines indicate the location of landmarks

In order to model misaligned functional data using covariates, we propose using a nonlinear regression model for both the aligned functional response and the warping functions based on the covariates. Monotonicity of the warping function model is enforced simply by fitting the unconstrained model and rejection sampling the posterior predictive distribution. This misaligned functional data modeling approach is general, in that it can be applied for various choices of nonlinear regression model. We will demonstrate the approach with Bayesian multivariate adaptive regression splines (BMARS), Bayesian additive regression trees (BART), and local approximate Gaussian processes (laGP). These models have been found to work well for various large datasets (Francom et al., 2019; Gramacy et al., 2015; Pratola et al., 2014). The purpose of this paper is not necessarily to compare these models, but to show that they can each work under the proposed framework. We chose these particular methods because they are scalable to large datasets and they quantify their predictive uncertainty.

We summarize BMARS, BART, and laGP the modeling techniques in the following paragraphs. Assuming scalar model output, let \mathbf{x}_i be the *p*-dimensional inputs to the *i*th model run for $i = 1, \ldots, n_x$, and let y_i be the associated response.



Figure 4: Warping functions for the five curves in Figure 2 that are linear between landmarks.

BMARS The premise of multivariate adaptive regression splines (Friedman, 1991) is to learn learn how to combine covariates to build a set of basis functions that can predict the output of interest. The basis functions are tensor products of truncated polynomials (often with polynomial degree one), so that

$$y_i = a_0 + \sum_{m=1}^M a_m B_m(\mathbf{x}_i) + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2)$$
$$B_m(\mathbf{x}) = \prod_{j=1}^{J_m} \left[s_{jm} (x_{v_{jm}} - k_{jm}) \right]_+$$

where M is the number of basis functions, J_m is the degree of interaction in the m^{th} basis function (the elements of the tensor product must use unique variables, hence the product is an interaction between a collection of variables), v_{jm} is the variable used, k_{jm} is a knot, $s_{jm} \in \{-1, 1\}$ is the sign, a_m is the coefficient for the m^{th} basis function, and σ^2 is the error variance. The Bayesian approach to MARS (Denison et al., 2002) puts priors on the unknown parameters, which are all those listed in the previous sentence, and uses reversible jump MCMC to sample the posterior. We use the implementation of Francom and Sansó (In press), which uses the innovations of Francom et al. (2018) and Francom et al. (2019).

BART Bayesian additive regression trees (Chipman et al., 2010) learns a set of trees that are added together to provide good prediction. Similar to the BMARS formulation above, we have

$$y_i = \sum_{m=1}^{M} g_m(\mathbf{x}_i; T_m, \boldsymbol{\theta}_m) + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2)$$

but now we have the m^{th} "basis" g_m as a tree model defined by splitting rules T_m and terminal node parameters $\boldsymbol{\theta}_m$. Tree priors, as well as terminal node parameter priors and a prior for σ^2 complete the model specification (for fixed M), and a Bayesian backfitting MCMC procedure is used to sample the posterior. We use the implementation of McCulloch et al. (2018).

laGP Local approximate Gaussian processes (Gramacy and Apley, 2015) fit GP models to neighborhoods of points around predictive locations. Hence, there is no general model fitting, but small models are fit independently for each prediction. This approach is augmented by creative ways of choosing neighborhoods, as merely using the nearest neighbors to fit a GP produces suboptimal predictions. A greedy search for the best set of "neighbors" of a given size coupled with active learning techniques typically yields accurate predictions quickly, even for very large datasets. While individual GP models are stationary, the predictions from laGP are nonstationary. While this approach does not quantify uncertainty in individual GP parameterizations, it does produce a GP and hence some gauge predictive uncertainty. We use the method of Gramacy and Haaland (2016) implemented in Gramacy et al. (2016).

In the remainder of the paper, we describe general methods for emulation with functional

response, including general methods for emulation with monotonic functional response (Section 2); we describe general approaches for emulating functional response models in combination with landmarks to build emulators for models with misaligned functional response (Section 3); we build and evaluate emulators for the HYDRA computer model, which has misaligned functional response, and perform a sensitivity analysis (Section 4); and finally we conclude (Section 5). The methods outlined in these sections could be used to build models and perform sensitivity analyses for various complex data generating processes that produce misalgined functional responses with clear landmarks. Such a contribution, to our knowledge, has not been made in the literature to date.

2 Emulation of Models With Functional Response

There are two popular approaches to building emulators for models with functional response. The first is to augment the computer model inputs with the functional variable, and build a model for a univariate output as a function of the larger collection of inputs. The second is to decompose the functional response onto a set of basis functions, and model the basis decomposition coefficients as a function of the computer model inputs. Below, we describe each of these approaches in more detail, after which we describe a simple approach to modeling monotone functional responses.

2.1 Augmentation Approach

The augmentation approach to modeling functional responses using covariates merely augments the covariates to include the variable that indexes the functional response. Thus, if the functional response for model run *i* given covariates \mathbf{x}_i is denoted $y_i(r)$, where *r* is the functional indexing variable, then the augmentation approach would model the univariate response y_{ij} as a function of covariates $(r_j, \mathbf{x}_i)'$ where r_j is the discretized variable *r*. When the functional responses for different model runs are given on the same discretized grid, or could be put on the same discretized grid (i.e., r_1, \ldots, r_{n_r}), this can be a popular choice for computational reasons. If the number of model runs is n_x and the number of covariates in \mathbf{x} is *p*, the scalar response model has p + 1 covariates and $n_x n_r$ "model runs," which can be difficult to handle if either n_x or n_r are moderately large. However, under a separable Gaussian process model, the computational cost is less than that necessary to build a GP with $n_x + n_r$ model runs because of the Kronecker structure of the covariance, though separability can be a strong assumption (Williams et al., 2006). Part of the draw of laGP and other approximate GP methods is that the scalability improvements allow the modeler to avoid the separability assumption.

Under the BMARS model with functional responses on a common grid, significant computational gains are achieved by recognizing that each basis function has a Kronecker structure, so that the matrix of basis functions is a Khatri-Rao product (Francom et al., 2018). While it is unclear whether there would be an opportunity for computational savings for BART with functional responses on a common grid, the development of BART has gone more in the direction of handling such datasets by improving scalability, as in Pratola et al. (2014). One of the issues with applying the augmentation approach to BART is that the functional response is often smooth, and the BART model does not provide smooth functional response predictions. Starling et al. (2018) develops a targeted smoothing approach where inputs that should have a smooth relationship with the response are not split on in trees, but are included in GP models at the leaves.

2.2 Basis Approach

Instead of augmenting the input space, the basis approach reduces the functional problem to that of considering several scalar response problems. This is achieved by decomposing, for each available combination of inputs, the functional responses onto a common set of basis functions and to model the basis coefficients using the covariates. Typically, the projection onto basis functions $K_1(r), \ldots, K_{n_r}(r)$ satisfies $y_i(r) = \sum_{l=1}^{n_r} m_{il} K_l(r)$, which we rewrite as

$$y_i(r) = \sum_{l=1}^{n_k} m_l(\mathbf{x}_i) K_l(r) + u_i(r)$$
(1)

where often the number of basis functions, n_k , is selected to be less than n_r in order to achieve dimension reduction. When $n_k < n_r$, the $u_i(r)$ term is included as a truncation error.

Higdon et al. (2008) uses GPs to model $m_l(\mathbf{x}_i)$, a principal component (or empirical orthogonal function) basis for K, and an iid Gaussian error for $u_i(r)$. Bayarri et al. (2007)

also uses GPs to model $m_l(\mathbf{x}_i)$, but uses a wavelet basis for K, and disregards $u_i(r)$. Francom et al. (2019) uses BMARS to model $m_l(\mathbf{x}_i)$, a principal component (or empirical orthogonal function) basis for K, and an empirical distribution for $u_i(r)$. They form the empirical truncation distribution by uniformly sampling from the observed truncations (with replacement) for each value of r. The number of elements to sample from for each r is the number of model runs (there is a truncation error for each model run). Thus, this approach is sensible for emulators built from a large number of model runs, since the truncation error is typically neither iid or Gaussian and it allows for the coefficient models to be fit completely in parallel.

2.3 Monotonic Functional Response

Many computer models have monotone functional response, though the monotonicity constraint is typically not enforced in the emulator because it is difficult to do (for instance, the implosion model of Higdon et al. (2008)). In our case, our warping function model will need to have monotone functional response, since changing the order of the functional variable is not sensible. Below we discuss some possible approaches.

A natural way to incorporate monotonicity is in the prior, so that models that yield nonmonotone predictions (or at least the predictive mean) would be given no posterior weight. This requires the ability to check whether the model yields any non-monotone predictions, perhaps by checking if the derivative of the model with respect to the functional variable is positive. It is easy to calculate the derivative of the BMARS model with respect to the functional variable since $\frac{d}{dr} [s(r-t)]^{\alpha}_{+} = \alpha [s(r-t)]^{\alpha-1}_{+} 1(s(r-t) > 0)s$. The difficulty comes when checking that the derivative is positive for a given model (with any combination of covariates, not just the ones we have seen) since we have a linear combination of such functions where the coefficients in the linear combination are a function of the other inputs.

Another approach is to incorporate monotonicity into the modeling framework so that the response is monotone by construction. This can be done in a few ways. For instance, we could model the log derivative of the functional response, perhaps obtained by differencing (Earls et al., 2017). The problem with this is that when we transform (exponentiate and integrate) back to the original space, we get undesirable heteroskedasticity in the functional response predictions. This is because any error in the log derivative space (particularly a nugget term) is compounded when integrating (or summing, when differencing is used to approximate the derivative). Another way to build monotonicity into the modeling framework is to model the functional response using monotonic basis functions with positive coefficients. This is common in one dimension (Bornkamp and Ickstadt, 2009; Shively et al., 2009), but when we need to be able to include interactions between the functional variable and covariates, this becomes difficult, again, since we have coefficient functions rather than scalars. The way that Chipman et al. (2016) are able to build monotonicity into BART is similar to this, but their approach (ensuring marginal monotonicity of the functional response in each additive component, or "basis" function) works because BART is additive rather than a linear combination. Projecting onto a set of monotone basis functions and modeling the log of the coefficients is another approach. Projecting onto I-splines (Ramsay, 1998) can result in a similar problem to the case of modeling in the derivative space, where basis functions represent very local movements so that we get a similarly heteroskedastic result when combining them.

Yet another approach to the monotonicity problem is to fit the unconstrained model and do a correction after the fact. This has been a topic of study in other fields, like quantile regression and dose response modeling. For instance, Chernozhukov et al. (2009) show that, in a quantile regression setting, swapping quantiles that are out of order improves estimates. Lin and Dunson (2014) develop a projection of an unconstrained GP into a monotone space which amounts to averaging non-monotone quantities, and show that this has empirical Bayes properties.

A more appealing approach that involves correcting the unconstrained model is to use rejection sampling of the posterior predictive distribution, since in our case (when modeling warping functions) we need samples from the posterior predictive distribution to have monotone functional response. Holmes and Heard (2003) apply reasoning to MCMC sampling in the model space that is applicable here. The posterior predictive distribution we want to sample is $\pi(y_{new}|y, C)$ where C is the monotonicity constraint. If we sample $\pi(y_{new}|y)$ and reject samples that do not satisfy C, then the resulting samples are from $\pi(y_{new}|y, C)$. This approach is simple, easy to implement for a wide range of models (including both the augmentation approach and basis approach models from Sections 2.1 and 2.2 above), theoretically justified, and is efficient in many cases. Specifically, this approach is efficient when, given the unconstrained model, there are few, if any, predictions that are not monotone. In cases where all or almost all posterior predictive samples violate the constraint, this would not work. However, such a situation could be symptomatic of more significant problems, as the training data may have strong violations of the constraint or perhaps the model is overfitting.

While we generally opt to use Bayesian approaches in order to propagate uncertainty in a coherent way, this approach could be applied to non-Bayesian models that provide predictive distributions (like laGP and other non-Bayesian GP models). For these models, we apply the same reasoning and use rejection sampling of the predictive distribution.

3 Misaligned Functional Response Emulation using Landmarks

Now that we have in place the ability to build a model for a functional response in terms of covariates, as well as a monotone functional response in terms of covariates, we will use these tools to build our model for the misaligned functional response in terms of covariates.

Let $y_i(r)$ denote the output from the *i*th computer model run, for $i = 1, ..., n_x$. The output is a function of r, though r may have a different range for different model runs. We model $y_i(r)$ as the composition of two random functions,

$$y_i(r) = y_i(r^*) \circ w_i^{-1}(r),$$
 (2)

where $y_i(r^*)$ is the aligned version of $y_i(r)$, and $w_i(r^*)$ is the monotone warping function that maps the aligned functional variable to the original (misaligned) scale. To be more explicit, let T be the set of all possible values of r and let T^* be the set of possible values of r^* . Then $w_i: T^* \to T$, $y_i(r): T \to R$, and $y_i(r^*): T^* \to R$. Since function composition is associative, and $w_i^{-1} \circ w_i$ is the identity function, we can rewrite the Equation 2 as

$$y_i(r) \circ w_i(r^*) = y_i(r^*).$$

3.1 Two Independent Models Approach

One way to get a model for the misaligned data is to model the aligned data and the warping functions independently. Using Equation 2, these can be combined to model the misaligned data. Also, the aligned data model and the warping function model can each be built using the approaches of Section 2, since each now has aligned functional response.

In practice, the landmarks we identify are the only points for which we have realizations of the warping functions, and hence, the only points we can use to train the warping function model. The warping functions themselves will need to be evaluated at points between the landmarks. If we did not have landmarks, the warping functions could be learned based on how they improve the fit of the aligned data model, though it is difficult to identify whether variations in the functions ought to be modeled by warping or by changing the aligned data model. Hence, regularization is often necessary to enforce a certain level of smoothness of warping functions. Landmarks are beneficial since they are realizations of the warping functions, but the warping model fit to landmarks also ought to be smooth between landmarks. We go as far as to adopt linearity between landmarks as a simple approach that seems to work well in practice. Hence, we will not require our emulators to predict warping functions away from the landmarks, though we note that the augmentation approach to BMARS would use a linear prediction between the landmarks because of the form of the model.

In practice, we also have that r and r^* are discretized. Say that r is discretized into r_1, \ldots, r_{n_r} with corresponding function values y_1, \ldots, y_{n_r} so that $y_j = y(r_j)$, and the curve has associated landmarks L_1, \ldots, L_q . Then we get the warping function w by linearly interpolating the x-y pairs $(L_1^*, L_1), \ldots, (L_q^*, L_q)$, where L_1^*, \ldots, L_q^* are the (arbitrary) new locations of the landmarks after alignment. Given w we can get a discretized r^* by evaluating $r_j^* = w^{-1}(r_j)$. Hence, x-y pairs $(r_1, y_1), \ldots, (r_{n_r}, y_{n_r})$ are aligned as $(w^{-1}(r_1), y_1), \ldots, (w^{-1}(r_{n_r}), y_{n_r})$. The result can be put on an arbitrary grid r_1^*, \ldots, r_m^* by linear interpolation. We can go backwards (i.e., given aligned data, "unwarp" it back to its misaligned space) using the same approach replacing w^{-1} with w.

3.2 Single Model Approach

Treating the aligned data and the warping functions independently could be justified from a modularization approach (Liu et al., 2009), where the modeler thinks that the warping functions ought to only be influenced by the landmarks and that the aligned data model ought to only involve the data aligned by the true landmarks. However, to condition on all the data (landmarks and aligned data) would mean that the warping model ought to depend on the fit to the landmarks and the fit of the aligned model according to the alignment from the warping model. This becomes difficult to implement, especially for large datasets, and is conceptually difficult to justify since we have landmarks. If we did not have landmarks, this would be a more justifiable approach.

A different approach to jointly considering the aligned data and the warping functions is to model the aligned data and the warping functions with a single functional (or multivariate) response model. This is because the misaligned functional response could really be thought of as multivariate functional response, as for the i^{th} realization we get two functional responses $y_i(r^*)$ and $w_i(r^*)$. In practice, this is further simplified since r^* is discretized and $w_i(r^*)$ is completely determined by the landmarks L_1^i, \ldots, L_q^i . Hence, the i^{th} model response is $(y_i(r_1^*), \ldots, y_i(r_m^*), L_1^i, \ldots, L_q^i)$. With multivariate response like this, the basis approach of Section 2.2 with empirical orthogonal functions (EOFs) is reasonable since those basis functions need not be smooth, so a single basis function can encode the relationship between the aligned data and the warping functions. This allows for correlations between the position of the landmarks and the y-values of the functions to be captured in the basis.

It should be noted that with eigen-decomposition based bases like EOFs, care needs to be exercised so that the empirical basis functions correctly describe variation is disparate data types. The x and y units of the functional response of computer models are typically different, and often have vastly different scales. In our multivariate response $(y_i(r_1^*), \ldots, y_i(r_m^*), L_1^i, \ldots, L_q^i)$, the units of $y_i(r_1^*), \ldots, y_i(r_m^*)$ are very different from the units of L_1^i, \ldots, L_q^i . Hence, standardization is necessary. Also, if equal attention is to be given to the aligned data and the warping functions in the basis decomposition, the number of discretized values of those functions need to be comparable or some kind of weighting needs to be implemented. The weight of $y_i(r_1^*), \ldots, y_i(r_m^*)$ should be the same as the weight of L_1^i, \ldots, L_q^i in determining variance to be explained in basis functions. Given a careful basis decomposition, the methods of Section 2.2 can be used to fit a single model for both the aligned data and the warping functions, which can then be combined as described in the previous section to provide predictions in the misaligned space.

4 HYDRA Emulation

HYDRA is used by scientists seeking to simulate inertial confinement fusion ignition experiments at the National Ignition Facility at Lawrence Livermore National Laboratory (Marinak et al., 2001). We will focus on HYDRA's simulation of energy production rate over time as a function of nine inputs described in Peterson et al. (2017). Eight of the inputs describe a laser stimulus (shape, amount, and timing) given to a fuel capsule, and the ninth is the density of the fuel in the capsule. We utilize a pre-existing ensemble of simulations consisting of 24440 completed model runs, which is a combination of two latin hypercubes where one is more concentrated than the other. A small percentage of the runs failed because of errors during in-situ processing of the model output. We disregard the runs that failed.

We use the log of energy production rate since each simulation of energy production rate varies over many orders of magnitude, and we are interested in both small and large values. The simulations of log of energy production rate over time exibit some clear features. There is an initial ramp up, followed by a slowing, another ramp up which eventually peaks, after which we have a decrease. The landmarks that we align are (1) the start, (2) the point between the initial ramp up and slowing, (3) the peak, and (4) the end, demonstrated in Figure 5. All but the second landmark are trivial to identify. We identify the second landmark by finding the line between the first and third landmarks, transforming the log energy production rate so that this line is the new x-axis, and finding the new maximum.

4.1 Prediction performance

With landmarks, we can proceed with the approaches detailed above. We randomly sample 100 of the 24440 model runs to hold out from emulator training. We show results for six emulators: (1) BMARS using the augmentation approach, (1) BART using the augmen-



Figure 5: HYDRA simulation landmarks.

tation approach, (2) laGP using the augmentation approach, (3) BMARS for EOF basis coefficients, (4) BART for EOF basis coefficients, and (5) laGP for EOF basis coefficients. For each of the models in EOF space, 10 EOFs are used, and the truncation error is assumed to come from the empirical distribution of truncation errors as in Francom et al. (2019). Figure 6 shows the predictions for the augmentation approach emulators for each of three hold-out model runs. Figure 7 shows predictions for the EOF basis emulators. All of these emulators are generally able to predict well, as they can capture the general shape of the curves and their position in time (note the differences in both the x and y axes for each holdout model run).

Figure 6 and Figure 7 are meant to provide intuition about each emulator's predictions rather than comparing emulator performance. Figure 8 gives more information for comparison, including root mean squared prediction errors (aligned and warping models), 95% contour sizes, and empirical coverages for the 100 holdout model runs. The standing of each of the three holdout model runs shown in Figures 6 and 7 are also labeled in Figure 8. These demonstrate, for instance, that the relatively poor prediction of laGP-EOF in holdout 3 from Figure 7 is an outlier, and that laGP-EOF performs better in most of the other 100 holdout predictions. Figure 8 also demonstrates that (1) there is not much difference in aligned model prediction, though the EOF approaches seem to work slightly better here; (2) there is somewhat more spread in the RMSE of the laGP and laGP-EOF warping models; (3) likely because of (2), laGP has somewhat larger 95% contour areas and laGP-EOF has a large range of 95% contour areas; and (4) all emulators tend to have good



Figure 6: A comparison of emulator predictions for three holdout model runs. The columns show three model runs. The rows correspond to emulation techniques. Plots obtained using BMARS and BART approaches show posterior predictive samples, posterior predictive samples of the mean, and their corresponding average. In the plots using laGP approaches, GP means and samples are shown. In each plot, a 95% contour of the predictive samples is shown as well as the holdout functional response. Note the changes in both the x and y axes for each holdout model run.



Figure 7: A comparison of emulator predictions for three holdout model runs. The columns show three model runs. The rows correspond to emulation techniques. Plots obtained using BMARS and BART approaches show posterior predictive samples, posterior predictive samples of the mean, and their corresponding average. In the plots using laGP approaches, GP means and samples are shown. In each plot, a 95% contour of the predictive samples is shown as well as the holdout functional response. Note the changes in both the x and y axes for each holdout model run.

empirical coverage. The variance of the laGP predictions could be decreased by allowing larger numbers of neighbors to be used when building GP models.

Figures 9, 10, and 11 show more context for the predictions. Each figure shows 25 holdout model runs with emulator predictions, all with the same x and y axis limits. Figure 9 shows these predictions for the BMARS-EOF emulator, Figure 10 shows the predictions for the BART-EOF emulator, and Figure 11 shows the predictions for the laGP-EOF emulator. Each shows a good ability to follow the changes in curve shape as well as the changes in curve timing. The laGP-EOF emulator predictions show both the largest and smallest uncertainties, and could again be refined by allowing for larger neighborhoods, at some computational expense.

4.2 Sensitivity Analysis

We now demonstrate how a sensitivity analysis of the HYDRA model could be conducted with an emulator for misaligned functional response. The goal of sensitivity analysis is to understand how changing the inputs to a model will affect the output. Functional ANOVA techniques like the Sobol decomposition (Sobol', 2001) are often used to partition the output variance based on the inputs their interactions.

The Sobol decomposition is typically approximated with Monte Carlo methods. For expensive models, emulators are often used as surrogates for computer models when using Monte Carlo methods, as many evaluations of the model are required. While we could approximate the Sobol decomposition of any of the emulators we have introduced, the BMARS and BMARS-EOF emulators have the advantage that the Sobol decomposition can be performed analytically (Francom et al., 2018, 2019). Hence, we will illustrate the sensitivity analysis with the Sobol decomposition of the BMARS aligned data model and the BMARS warping model.

Figure 12 shows the functional sensitivity analysis of the BMARS aligned model. This indicates that most of the variance in the early to middle warped time period is due to variation in input three, which controls the density of the fuel in the capsule. There is a decrease in the total variance about 75% of the way through the warped time ($r^* = 0.75$), as many of the curves overlap around that point. Variation after that point is due to changes in the shape, timing, and amount of laser stimulus to the fuel capsule.



Figure 8: An evaluation of emulator predictive accuracy and uncertainty for five emulation approaches based on 100 held out model runs. For each of the 100 predictions, aligned and warping model RMSE are calculated as well as the area of the 95% contour and that contour's empirical coverage. The numbers on top of each boxplot show where the three holdout predictions shown in Figures 6 and 7 stand.

Figure 13 shows the functional sensitivity analysis of the BMARS warping model. This shows that input six explains most of the timing variation, with some contribution from input eight and a small contribution from input three for the initial timing.

5 Conclusion

We have presented approaches to modeling misalgined functional data in terms of covariates when landmarks can be identified. The approach is general, so that most nonlinear models could be used. The innovations of this work include (1) the general approach to modeling

BMARS-EOF prediction



Figure 9: Various holdout predictions using the BMARS-EOF emulator shown with the same axis limits.

misalgined functional responses by unitlizing landmarks and warping models, (2) a simple way to include a monotonicity constraint on the functional responses, (3) modeling aligned data and warping functions using independent models or a single model in EOF coefficient space, and (4) sensitivity analysis for misalgined functional response emulators.

We have demonstrated the effectiveness of our approach to emulate a computer model with nine inputs and misaligned functional response. Popular approaches to this problem may have emulated the aligned functional response (Bayarri et al., 2007) or some important feature or function of the output (Walters et al., 2018). Our approach demonstrates a successful way to emulate the full output of the computer model, in addition to a sensitivity analysis using the emulator.

This paper opens up the field of computer experiments for misaligned simulations. Future research in this area, such as incorporating landmark uncertainty, using these emulators for calibration, and applying the methodology to higher dimensional misaligned functional responses, could provide interesting solutions to problems that emulator users often face.

BART-EOF prediction



Figure 10: Various holdout predictions using the BART-EOF emulator shown with the same axis limits.

We are actively pursuing some of these areas.

References

- Bayarri, M., Berger, J., Cafeo, J., Garcia-Donato, G., Liu, F., Palomo, J., Parthasarathy, R., Paulo, R., Sacks, J., and Walsh, D. (2007), "Computer model validation with functional output," *The Annals of Statistics*, 1874–1906.
- Bornkamp, B., and Ickstadt, K. (2009), "Bayesian nonparametric estimation of continuous monotone functions with applications to dose–response analysis," *Biometrics*, 65, 198– 205.
- Chakraborty, A., Mallick, B. K., Mcclarren, R. G., Kuranz, C. C., Bingham, D., Grosskopf, M. J., Rutter, E. M., Stripling, H. F., and Drake, R. P. (2013), "Spline-based emulators for radiative shock experiments with measurement error," *Journal of the American Statistical Association*, 108, 411–428.
- Chernozhukov, V., Fernandez-Val, I., and Galichon, A. (2009), "Improving point and interval estimators of monotone functions by rearrangement," *Biometrika*, 96, 559–575.

Chipman, H. A., George, E. I., McCulloch, R. E., and Shively, T. S. (2016), "High-

IaGP-EOF prediction



Figure 11: Various holdout predictions using the laGP-EOF emulator shown with the same axis limits.

dimensional nonparametric monotone function estimation using BART," arXiv preprint arXiv:1612.01619.

- Chipman, H. A., George, E. I., McCulloch, R. E., et al. (2010), "BART: Bayesian additive regression trees," *The Annals of Applied Statistics*, 4, 266–298.
- Denison, D. G., Holmes, C. C., Mallick, B. K., and Smith, A. F. (2002), Bayesian Methods for Nonlinear Classification and Regression (Vol. 386), John Wiley & Sons.
- Earls, C., Hooker, G., et al. (2017), "Variational Bayes for Functional Data Registration, Smoothing, and Prediction," *Bayesian Analysis*, 12, 557–582.
- Francom, D., and Sansó, B. (In press), "BASS: An R Package for Fitting and Performing Sensitivity Analysis of Bayesian Adaptive Spline Surfaces," *Journal of Statistical Software*.
- Francom, D., Sansó, B., Bulaevskaya, V., Lucas, D., and Simpson, M. (2019), "Inferring Atmospheric Release Characteristics in a Large Computer Experiment using Bayesian Adaptive Splines," *Journal of the American Statistical Association*, 1–21.
- Francom, D., Sansó, B., Kupresanin, A., and Johannesson, G. (2018), "Sensitivity analysis and emulation for functional data using Bayesian adaptive splines," *Statistica Sinica*, 28, 791–816.



Figure 12: Functional pie chart of the Sobol decomposition and variance decomposition for the BMARS aligned model.

- Friedman, J. H. (1991), "Multivariate adaptive regression splines," The Annals of Statistics, 1–67.
- Gervini, D., and Gasser, T. (2004), "Self-modelling warping functions," Journal of the Royal Statistical Society: Series B (Statistical Methodology), 66, 959–971.
- Gramacy, R. B., and Apley, D. W. (2015), "Local Gaussian process approximation for large computer experiments," Journal of Computational and Graphical Statistics, 24, 561–578.
- Gramacy, R. B., Bingham, D., Holloway, J. P., Grosskopf, M. J., Kuranz, C. C., Rutter, E., Trantham, M., Drake, R. P., et al. (2015), "Calibrating a large computer experiment simulating radiative shock hydrodynamics," *The Annals of Applied Statistics*, 9, 1141– 1168.
- Gramacy, R. B., and Haaland, B. (2016), "Speeding up neighborhood search in local Gaussian process prediction," *Technometrics*, 58, 294–303.
- Gramacy, R. B., et al. (2016), "laGP: large-scale spatial modeling via local approximate Gaussian processes in R," *Journal of Statistical Software*, 72, 1–46.
- Higdon, D., Gattiker, J., Williams, B., and Rightley, M. (2008), "Computer model calibration using high-dimensional output," *Journal of the American Statistical Association*, 103.
- Holmes, C., and Heard, N. (2003), "Generalized monotonic regression using random change points," *Statistics in Medicine*, 22, 623–638.
- Hung, Y., Joseph, V. R., and Melkote, S. N. (2015), "Analysis of Computer Experiments With Functional Response," *Technometrics*, 57, 35–44.



Figure 13: Functional pie chart of the Sobol decomposition and variance decomposition for the BMARS warping model.

- Kennedy, M. C., and O'Hagan, A. (2001), "Bayesian calibration of computer models," Journal of the Royal Statistical Society. Series B, Statistical Methodology, 425–464.
- Lin, L., and Dunson, D. B. (2014), "Bayesian monotone regression using Gaussian process projection," *Biometrika*, 101, 303–317.
- Liu, F., Bayarri, M., Berger, J., et al. (2009), "Modularization in Bayesian analysis, with emphasis on analysis of computer models," *Bayesian Analysis*, 4, 119–150.
- Marinak, M., Kerbel, G., Gentile, N., Jones, O., Munro, D., Pollaine, S., Dittrich, T., and Haan, S. (2001), "Three-dimensional HYDRA simulations of National Ignition Facility targets," *Physics of Plasmas*, 8, 2275–2280.
- McCulloch, R., Sparapani, R., Gramacy, R., Spanbauer, C., and Pratola, M. (2018), BART: Bayesian Additive Regression Trees, r package version 1.6.
- Peterson, J., Humbird, K., Field, J., Brandon, S., Langer, S., Nora, R., Spears, B., and Springer, P. (2017), "Zonal flow generation in inertial confinement fusion implosions," *Physics of Plasmas*, 24, 032702.
- Pratola, M. T., Chipman, H. A., Gattiker, J. R., Higdon, D. M., McCulloch, R., and Rust, W. N. (2014), "Parallel Bayesian additive regression trees," *Journal of Computational* and Graphical Statistics, 23, 830–852.
- Pratola, M. T., and Higdon, D. M. (2016), "Bayesian additive regression tree calibration of complex high-dimensional computer models," *Technometrics*, 58, 166–179.
- Ramsay, J., and Silverman, B. (2005), *Functional Data Analysis*, Springer Series in Statistics, Springer.

- Ramsay, J. O. (1998), "Estimating smooth monotone functions," Journal of the Royal Statistical Society: Series B (Statistical Methodology), 60, 365–375.
- Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008), *Global Sensitivity Analysis: The Primer*, John Wiley & Sons.
- Santner, T. J., Williams, B. J., Notz, W., and Williams, B. J. (2003), The design and analysis of computer experiments (Vol. 1), Springer.
- Shively, T. S., Sager, T. W., and Walker, S. G. (2009), "A Bayesian approach to nonparametric monotone function estimation," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71, 159–175.
- Sobol', I. M. (2001), "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and computers in simulation*, 55, 271–280.
- Starling, J. E., Murray, J. S., Carvalho, C. M., Bukowski, R., and Scott, J. G. (2018), "Functional response regression with funBART: an analysis of patient-specific stillbirth risk," arXiv preprint arXiv:1805.07656.
- Strait, J., and Kurtek, S. (2016), "Bayesian Model-Based Automatic Landmark Detection for Planar Curves," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 86–94.
- Telesca, D. (2015), "Bayesian analysis of curves shape variation through registration and regression," in *Nonparametric Bayesian Inference in Biostatistics*, Springer, pp. 287–310.
- Tucker, J. D., Wu, W., and Srivastava, A. (2013), "Generative models for functional data using phase and amplitude separation," *Computational Statistics & Data Analysis*, 61, 50–66.
- Walters, D. J., Biswas, A., Lawrence, E. C., Francom, D. C., Luscher, D. J., Fredenburg, D. A., Moran, K. R., Sweeney, C. M., Sandberg, R. L., Ahrens, J. P., et al. (2018), "Bayesian calibration of strength parameters using hydrocode simulations of symmetric impact shock experiments of Al-5083," *Journal of Applied Physics*, 124, 205105.
- Williams, B., Higdon, D., Gattiker, J., Moore, L., McKay, M., Keller-McNulty, S., et al. (2006), "Combining experimental data and computer simulations, with an application to flyer plate experiments," *Bayesian Analysis*, 1, 765–792.