Received: 7 November 2021 | Revised: 21 December 2022 | Accepted: 26 January 2023

DOI: 10.1002/env.2796

RESEARCH ARTICLE

WILEY

Comparing emulation methods for a high-resolution storm surge model

Grant Hutchings[1,2] | **Bruno Sansó**[1] | **James Gattiker**[2] | **Devin Francom**[2] | **Donatella Pasqualini**[2]

[1]Department of Statistics, University of California Santa Cruz,
Santa Cruz, California, USA

[2]Statistical Sciences Group, Los Alamos National Laboratory,
Los Alamos, New Mexico, USA

Correspondence
Grant Hutchings, Department of Statistics, University of California Santa Cruz, Santa Cruz, CA, USA.
Email: grhutchi@ucsc.edu

Funding information
Los Alamos National Laboratory; University of California, Santa Cruz; National Science Foundation, Grant/Award Numbers: MMS2050012, DMS2153277

Abstract

Realistic simulations of complex systems are fundamental for climate and environmental studies. Large computer systems are often not sufficient to run sophisticated computational models for large numbers of different input settings. Statistical surrogate models, or emulators, are key tools enabling fast exploration of the simulator input space. Gaussian processes have become standard for computer simulator emulation. However, they require careful implementation to scale appropriately, motivating alternative methods more recently introduced. We present a comparison study of surrogates of the Sea, Lake, and Overland Surges from Hurricanes (SLOSH) simulator—the simulator of choice for government agencies—using four emulation approaches: BASS; BART; SEPIA; and RobustGaSP. SEPIA and RobustGaSP use Gaussian processes, BASS implements adaptive splines, and BART is based on ensembles of regression trees. We describe the four models and compare them in terms of computation time and predictive metrics. These surrogates use proven and distinct methodologies, are available through accessible software, and quantify prediction uncertainty. Our data cover millions of response values. We find that SEPIA and RobustGaSP provide exceptional predictive power, but cannot scale to emulate experiments as large as the one considered in this paper as effectively as BASS and BART.

KEYWORDS

BART, BASS, computer model emulation, Gaussian process, RobustGaSP, SLOSH, storm surge

1 | INTRODUCTION

The severity, location, and timing of an extreme environmental event interacts with critical infrastructures such as power and water systems in complex ways. The resulting impacts to those infrastructures have consequences to the populations and economies that depend on them. A particular extreme event may severely impact one decision maker while leaving another stakeholder relatively unharmed, and small changes in the properties of an event (e.g., the track of Hurricane Sandy) can also create significant changes to specific impacts and consequences.

Assessing the impacts of a hazard by analyzing a limited, predefined set of scenarios supported by computational simulations such as the Standard Project Hurricane (SPH) and the Probable Maximum Hurricane (PMH) methods (Graham & Nunn, 1959; Schwerdt et al., 1979) (e.g., few intense hurricanes or high values of sea level) may result in an incomplete or incorrect estimate of risk. Alternatively, statistical approaches have been developed and applied that aggregate the impacts and consequences into probability distributions enabling stakeholders to accommodate their assessed risk (Johnson et al., 2021; Pasqualini, 2017), such as the joint probability method (Ho et al., 1970; Myers, 1970; Myers, 1975), the joint probability with optimal sampling method (Irish et al., 2009; Resio, 1970; Resio et al., 2009; Resio et al., 2017; Toro et al., 2010; Yang et al., 2019), and the stochastic-deterministic track method (Emanuel et al., 2006; Vickery & Twisdale, 1995). However, model-supported approaches come with a drawback: running a large number of complex simulations, such as climate models, is computationally intense or even intractable. A alternative approach is to provide a resource that can be easily and quickly drawn on, and that incorporates observations of related events to infer a probability distribution over extreme events. With this distribution a historically-consistent ensemble of simulated events that are representative of possible outcomes can be generated. The use of a fast emulator of simulation models addresses the challenge of analyzing the domain of outcomes informed by a computationally expensive model. This paper provides analyses that support the choice of methodology for the efficient use of computation simulations through statistical emulation. The specific application domain we consider is the use of flood inundation models to support electrical grid impact, and generalizes to considerations in the use of alternative emulation approaches in a range of related application.

1.1 Background

Scientific analysis in environmetric application areas often relies on high resolution spatiotemporal physics-based simulations. For instance, various physics-based simulators of climate and weather systems generate spatially resolved simulations given a set of inputs (e.g., Borge et al., 2008; Petersen et al., 2019). Challenges associated with complex simulators of climate, weather and environmental systems include: (1) they require a supercomputer to run; (2) their run times and computational resources are significant; and (3) input parameters to the simulators have uncertainty associated with them which must be characterized in order to make reliable scientific conclusions (Fassò & Perri, 2002; Marrel et al., 2011).

Scientists deal with input parameter uncertainty in two ways: intrusively, meaning that they change the simulation model structure to allow for propagation of uncertainty distributions within the simulation (e.g., Venturi & Karniadakis, 2012); or non-intrusively, where the simulator is treated as a black box, and knowledge is gained by running it at different settings (Sacks et al., 1989). Non-intrusive approaches to dealing with parameter uncertainty are attractive because they do not require changes to the fundamental model structure represented in a simulator and its computer code, which may be infeasible. However, non-intrusive approaches require running the simulator at many different settings, which is computationally expensive. In non-intrusive settings, the approaches to dealing with the three challenges listed above almost always involve building a statistical surrogate, or emulator, for the simulator. After training an emulator, prediction at a new set of inputs is fast, making studies of model response and parameter uncertainty feasible. This is a dominant approach to treating parameter uncertainty in the field of computer experiments (Santner et al., 2018).

Our focus in this manuscript is the emulation of the Sea, Lake, and Overland Surges from Hurricanes (SLOSH) simulator (Jelesnianski et al., 1992). This simulator, developed by the National Weather Service, is used operationally for hurricane monitoring and response. After configuration of the domain of interest, SLOSH takes a hurricane track as input and outputs a spatiotemporal grid of storm surge, which is the water depth over normal levels (e.g., water depth above ground level for land spatial locations). Our particular application interest motivating this study is in flooding of electrical substations, so the key criteria is the maximum flood depth at each spatial location for a given hurricane, rather than the temporal aspect of the simulations. Our simulation area of interest is the Delaware Bay, which separates the south end of New Jersey from the North side of Delaware. Figure 1 shows one hurricane storm surge simulation from SLOSH for the Delaware Bay as well as electric power substations in the area. This paper does not describe that application in detail, instead focusing on the underlying emulation problem that is relevant to a range of applications where analysis relies on a complex computational model.

An early work in the literature of computer experiments (Sacks et al., 1989) showed how a Gaussian process (GP) could be used as a predictor, with quantified uncertainty, based on an ensemble of examples over the domain

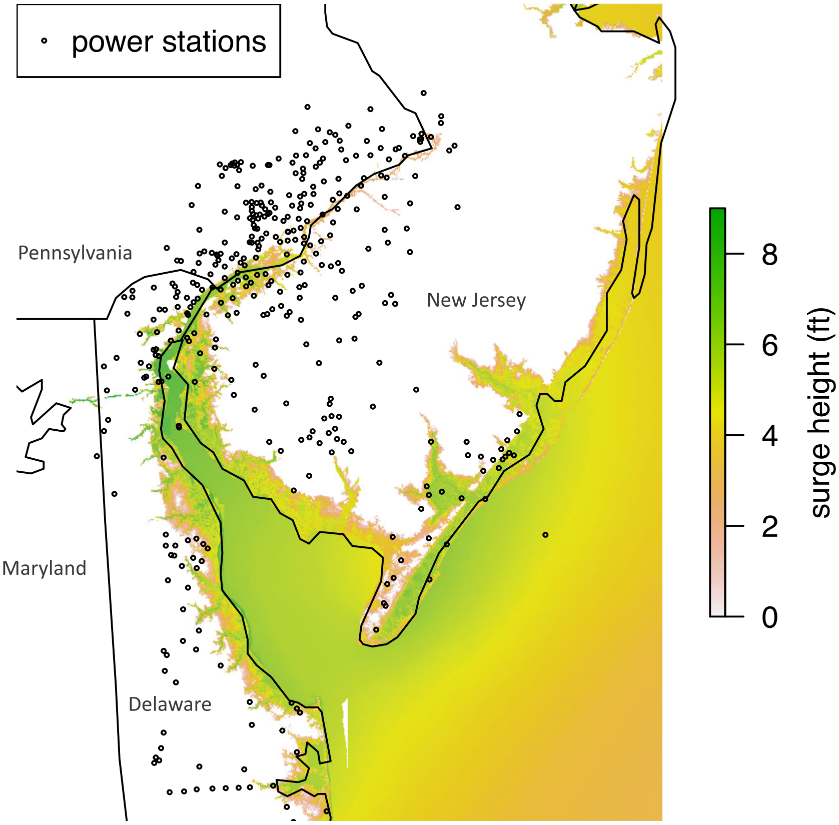


FIGURE 1 Surge output map from SLOSH. Also shown are electric power substations

of interest. GP's became and remain a common approach for emulating computer simulations, with a vast literature exploring their application and variations in specific approach. The main purpose of an emulator is to provide predictions at untried parameter settings with an estimate of the associated uncertainty, and to do it much faster than running the actual computer simulation (Salter & Williamson, 2016b). Kennedy and O'Hagan (2001) integrated the use of GP emulation for uncertainty analysis accounting for parameter uncertainty in addition to code uncertainty, describing the Bayesian calibration of computer models incorporating model inadequacy, residual variation and observation error. GP's are very desirable for their flexibility and accessible uncertainty quantification (UQ). They are however not always practical; GPs are limited by the computational bottleneck of covariance matrix inversion, which presents limits to the size of the ensemble dataset. Many recent methods such as LaGP (Gramacy & Apley, 2015), TGP (Gramacy & Lee, 2008), GPvecchia (Katzfuss & Guinness, 2021), and RobustGaSP (Gu et al., 2017) aim to tackle this scalability issue. More recently, competitive alternatives to GP's have been proposed such as BASS (Francom & Sansó, 2020) which implements an adaptive spline model for emulation, and BART (Chipman et al., 2010) which uses additive regression trees.

In the supplementary material of Francom et al. (2019), BASS and BART were shown to perform similarly well in out-of-sample RMSE for a synthetic data experiment. The comparison also includes TGP and a Random Forest (RF) model for three datasets of size $n = 50{,}500, 5000$. It was found that BASS and BART outperformed both TGP and the RF model for $n = 50$, and were beaten only by TGP for $n = 500$. At $n = 5000$ TGP was not included due to its significant computational burden and the RF model was again beaten by BASS and BART. These methods do not involve a step that scales as poorly as n^3, allowing larger datasets than Gaussian Process, and are well-known in practice to be significantly less computationally demanding for larger datasets (see (Francom et al., 2018; Francom et al., 2019; Pratola et al., 2014; Sparapani et al., 2021)). These methods also provide simple uncertainty quantification as they make inference by Markov-chain Monte Carlo (MCMC).

The analysis presented here compares four emulation methods on SLOSH simulated hurricane induced flooding in the Delaware Bay, described above. The goals of this study are to compare the accuracy of predictions, quality of uncertainty quantification, and the computational requirements of each method for a range of training set sizes. In doing so, we aim to understand how training set size effects predictions and run time. Additionally we will compare the model parameter importance assessment options given by each method. Investigating parameter importance for hurricane flooding models helps researchers understand which inputs to a simulation are most relevant to quantities of interest. Here we are interested in qualities of a hurricane or a particular area that are most influential in determining inland flooding. Some emulators allow for spatially resolved variable importance and variance-based assessment of importance (e.g., via the Sobol decomposition; Sobol, 2001), which both benefit analyses involving highly multivariate emulators.

The remainder of the paper is structured as follows: In Section 1.2 we give a brief overview of the four methods included in our study and explain why they were chosen; Section 2 is an overview of the simulations from SLOSH; Section 3 describes of each of the four emulator formulations in detail; Section 4 presents our comparison study, highlighting a variety of predictive metrics and scores; Section 5 gives an overview of the variable importance built into each package; and we conclude with a discussion of our findings and recommendations to the reader in Section 6.

1.2 | Emulation methods

The emulation methods we have chosen implement distinct statistical models, all of which have proven themselves a reasonable choice for computer model emulation. We give a brief overview of the methods here, with more detailed descriptions in Section 3. Two GP based models are considered, the first of which was introduced by Higdon et al. (2008) and is implemented in the Python package SEPIA (Gattiker, Klein, et al., 2020). For this model, a collection of independent GP's are fit to coefficients of an orthogonal basis representation of the simulation response data. The second GP based model is RobustGaSP (Gu et al., 2017) which implements a "many single" approach, fitting an independent GP to each spatial location in parallel. We also include two non-GP based models; BASS and BART. These models were chosen for their qualities discussed in Section 1.1. These four models cover some diverse modeling strategies, but in no way cover the full spectrum of emulation methodologies.

Emulator comparisons have been done in the past, often comparing on a host of test functions with relatively small amounts of data, or focusing on parameter calibration rather than strictly emulation (e.g., Erickson et al., 2018; Salter & Williamson, 2016a). The comparison here is motivated by the requirements of this application which poses particular problems that are relevant to spatial environmental modeling. What we present is a comparison which focuses only on a few models in greater detail, in an application driven big-data setting. This, to our knowledge, is not available in the literature.

SEPIA, the first of the four methods considered, was originally implemented at Los Alamos National Laboratory as the MATLAB code GPMSA (Gattiker, Higdon, & Williams, 2020) and was later refactored to Python. SEPIA makes use of a basis representation of multivariate simulation output, typically empirical orthogonal functions (EOF) (as in principle components analysis) of the data to fit a Gaussian process to each of the basis coefficients. This is a tried and true methodology for spatial modeling that has seen much success in the literature and in applications.

Our implementation of Bayesian Adaptive Spline Surfaces (BASS) makes use of the same basis representation, but takes a wholly different approach to modeling basis coefficients by using adaptive splines. BASS has been recently applied to large spatial data from computer experiments and has shown promising results (see, e.g., Francom et al., 2019).

The implementation considered in this work of Bayesian Additive Regression Trees (BART) once again makes use of this basis representation by fitting an independent BART model to each basis coefficient. The BART package does not inherently manage multivariate response through basis representation (as in SEPIA and BASS), and so we extend the functionality by explicitly supplying an EOF basis. The BART model fits the EOF weights and the predictions are expanded into the native spatial domain. This allows a more direct comparison to other methods. We have explored this implementation in the past (Francom et al., 2022). Treed models have seen success in the literature for their speed and flexibility, and BART has proven to be effective in settings similar to the one considered in this paper, such as a recent analysis of airborne particulate data over California (Zhang et al., 2020). Preliminary comparisons of BASS and BART in Francom et al. (2019) showed that both approaches can be highly accurate and efficient.

The fourth method considered in this work consists of Robust Gaussian Stochastic Process Emulation (RobustGaSP) which handles multivariate response by fitting a GP to each point in space, rather than reducing the modeling dimension through a linear projection as the other methods in this comparison. This is made computationally feasible by both parallel computation, and the assumption of shared range parameters for all GP's. RobustGaSP does not make use of MCMC for model fitting like the other three models. Instead parameters are fit using numerical optimization of marginal posterior distributions. These major model differences make this an interesting inclusion to our comparison study. RobustGaSP has also shown promising results on large scale computer model emulation of large volcanic flow simulations (Gu & Berger, 2016).

Additionally, we include a simple linear model on the coefficients of an orthogonal basis representation as a baseline to gauge the improvements provided by the models that allow greater complexity. We choose a linear model as a general formulation for an emulator of a scalar output, given a set of inputs $\boldsymbol{x}$ is

$$y(\boldsymbol{x}) = \sum_{k=1}^{K} g_k(\boldsymbol{x})\beta_k + \varepsilon(\boldsymbol{x})$$

for some collection of functions g_k. In the case of a GP, $\varepsilon(\boldsymbol{x})$ is correlated, according to the correlation function that specifies the GP, and g_k is known. In the case of BASS and BART $\varepsilon(\boldsymbol{x})$ is uncorrelated noise and g_k is inferred adaptively. A linear model corresponds to the case where $g_k(\boldsymbol{x}) = x_k$, and $\varepsilon(\boldsymbol{x})$ is uncorrelated, which are, arguably, the simplest possible non-trivial choices. Aside from the linear model, the models considered in this paper all show accurate predictions using quite different methodologies.

1.2.1 | Variable importance

Variable importance for computer models (often referred to as sensitivity analysis) consists of determining which inputs have the greatest (least) effect on the response. Keller et al. (2021) discuss the importance of sensitivity analysis for the identification of parameter uncertainties most relevant to decision making in climate hazard applications such as the case study presented in this paper. Validated emulators are useful for sensitivity analysis and variable importance calculations, as these operations typically require extensive evaluation of the response. Global sensitivity analysis consists of quantifying the percentage of the variability in the response due to each input, or combination of inputs, and is done through functional analysis of variance (ANOVA) which has previously been applied to computer experiments (Schonlau & Welch, 2006). More specifically, practitioners often use Sobol indices computed using draws from the emulator posterior predictive distribution (Sobol, 2001). An additional, very desirable property of Sobol indices is that different uncertainty distributions on the model inputs can be considered, and sensitivities can be compared across these distributional assumptions. This is very applicable to our case study as hurricane impacts are location specific, and there is no broad consensus on their spatial distributions (and the associated distributions in parameters). Wong and Keller (2017) use Sobol indices to identify those mechanisms which contribute most to uncertainty in projected flood risk for a levee ring in New Orleans, LA.

SEPIA and BASS include a closed-form technique for obtaining Sobol indices, facilitated by the underlying model form, BART and RobustGaSP do not. Methods for computing Sobol indices have been generalized in the R package "sensitivity" (Iooss et al., 2021), so in principle sensitivity indices is available through extensions of the packages. However, the Sobol analysis requires many predictions from the emulator at various input settings, compounded by distribution samples, which would entail considerable computation. RobustGaSP and BART provide their own form of variable importance natively.

BART offers a unique form of variable importance (and hence, sensitivity analysis) by keeping track of the number of times each input variable is included in the regression trees. The ratio of the number of trees containing a certain variable to the total number of trees is called the "percent usage" of that variable and has been used by the developers of BART as a variable importance measure (Chipman et al., 2010). For every posterior predictive sample, we calculate the percentage of trees containing each input variable. This gives a distribution of percentages over posterior draws. The drawback is that information is only available for individual models corresponding to a single EOF coefficient and we cannot simply aggregate over components to get sensitivity for the original response.

RobustGaSP determines if an input is believed to be inert, or contributes little to response variability. Inertness is decided through the estimated range parameters $\hat{\boldsymbol{\gamma}}$. This is really more of a variable selection technique introduced in (Linkletter et al., 2006), but can be considered a form of variable importance or sensitivity analysis. If γ_l is inert, $\hat{\gamma}_l \to \infty$ and has little effect on response variability (Gu, 2019). The JR prior we described in Section 3.4 is required for this to work. The key is that this prior, unlike the reference prior, makes sure the marginal posterior for $\boldsymbol{\gamma} > 0$ even if some

$\hat{\gamma}_l \to \infty$. To decide whether a $\hat{\gamma}_l$ is sufficiently large to consider the associated input inert, we consider the normalized inverse

$$\hat{P}_l = \frac{C_l \hat{\beta}_l}{\sum_{i=1}^{p_x} C_i \hat{\beta}_i} \tag{1}$$

where $\hat{\beta}_l = 1/\hat{\gamma}_l$ and C_l is a normalization constant to account for the different scales of the inputs (Gu, 2019). We can then set a threshold (default of 0.1) below which an input is determined to be inert. Table 2 shows the results for our RobustGaSP model trained on 500 storms. We see that none of the inputs are found to be inert.

Albeit far less informative from a sensitivity analysis point of view than a Sobol decomposition, this is valuable information which comes for free as a byproduct of the model fit as GP length-scale parameters must be estimated regardless of their usage in sensitivity analysis.

2 | SIMULATOR AND DATASET

The Sea, Lake, and Overland Surges from Hurricanes (SLOSH) simulator (Jelesnianski et al., 1992) is a computer code developed by the National Weather Service to estimate storm surge heights from hurricanes. Storm surge height is defined as the maximum water height due to a hurricane at any single location. Our data consists of an ensemble of 4000 runs from the SLOSH simulator, corresponding to 4000 simulated storms. Each storm in the ensemble is defined by a unique set of five input parameters:

1. Sea level rise in the year 2100 (lower: −20; upper: 350; units: cm). Sea level rise values are obtained from the open source LocalizeSL tool (Kopp et al., 2014). The tool is based on a probabilistic data fusion method that combines modern climate model projections, subject matter expert judgment where models are lacking, and historical tide gauge data. It provides annual projections and uncertainty of local sea level rise over the 21st century at tide gauge locations, expressed as probability distributions that evolve over time. It accounts for all major sources of sea level rise, including oceanographic changes in heat content, salt content and circulation patterns, glacier and ice sheet melting, the gravitational attraction of ice sheets on the nearby ocean, and transfer of water between the ocean and land reservoirs. It also accounts for local changes in coastal land elevation due to slow geological adjustment to the removal of glaciers from the last ice age, sediment compaction, plate tectonics, and groundwater withdrawal. Both sea level rise and land elevation changes combine to affect the local sea level, that is, the height of the sea relative to the coastline. Tides for this study were taken from the Lewes, DE tide gauge (NOAA, n.d.). Estimates for sea level rise are based on climate projections, Representative Concentration Pathways (RCP), that assume a socioeconomic scenario for future emissions of greenhouse gases and other atmospheric constituents. The analysis assumes that the world future fossil fuel emissions will be approximately represented by the RCP 6.0 scenario.
2. Heading of the eye of the storm when it made landfall (lower: 204.0349; upper: 384.0244; units: degrees, north is 0/360). The upper limit for heading is greater than 360° because there were a number of storms with a heading less than 60°, and a large gap between those and 204.0349°. To make the heading contiguous, we added 360° to all headings less than 60°.
3. Velocity of the eye of the storm when it made landfall (lower: 0; upper: 40; units: knots).
4. Minimum air pressure of the storm when it made landfall (lower: 930; upper: 980; units: millibars).
5. Latitude of the eye of the storm when it made landfall (lower: 38.32527° N; upper: 39.26811° N; units: degrees). Longitude is calculated as a function of latitude along the coastline; we roughly interpolate the coastline stretch of interest to be a one-to-one function with longitude as a function of latitude.

The radius of maximum wind, which is an input of SLOSH, is not an explicit input here; it is estimated as a function of a parametric model of minimum air pressure and latitude (Vickery & Wadhera, 2008).

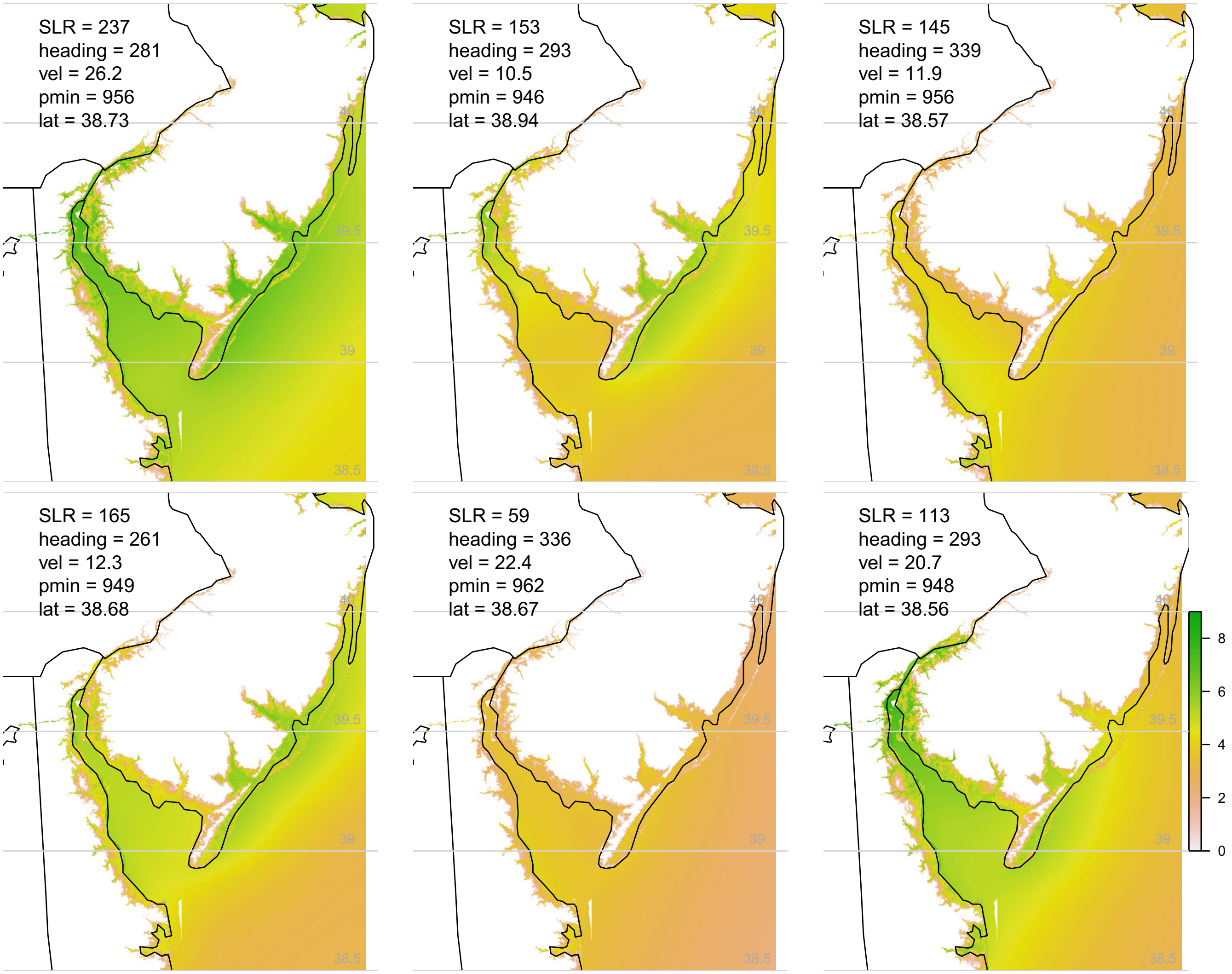


FIGURE 2 Surge output map from SLOSH using six different combinations of input settings

Input parameters for the ensemble are generated using a space-filling Latin hypercube design (Leary et al., 2003; Tang, 1993; Ye et al., 2000) over our five dimensional parameter space. Models are trained on subsets of this ensemble and tested on storms outside of the training sets. Further details regarding testing and training sets will be discussed in Section 4. Figure 2 shows six of the simulations from our ensemble.

Our interest lies in prediction of hurricane-induced flooding in the Delaware Bay. Under our model setup, one output from SLOSH is a 4520 × 5115 rectangular grid of storm surge heights for each of the 23,119,800 locations. The area of interest is between −76.11263° W and −70.99763° W longitude and 36.71718° N and 41.23718° N latitude with a spatial resolution of 0.001° in each direction. Such a large number of spatial locations presents a formidable computational challenge which is fortunately eased by the fact that the majority of the points on the grid are far enough inland that there is no flooding for any of the 4000 simulations. By modeling only cells which take non-zero values in at least one of the simulations we reduce the size of the field to around 3,500,000 locations.

Accurate prediction of flooding is important for a variety of reasons including displacement of residents, and property/infrastructure damage. One area of specific interest for this project is damage to electrical power stations which are shown as black dots in Figure 1. Power stations in this area are often fortified to handle four feet of flood water, with higher levels leading to catastrophic damage. We are therefore interested in an emulators' ability to accurately predict that a surge has reached four feet. This information is valuable for determining if an intervention in the form of a station shutdown is necessary due to an incoming storm.

The output from SLOSH is non-negative, but all of our models have the potential for negative predictions. A common strategy for modeling a non-negative response is to apply a log-transformation. We have chosen not to do this because of our interest in storm surge near four feet. The function $log(x)$ has a steep gradient for small x, but approaching $x = 4$ the slope is very low. This would bias our emulators to fit small values with relatively greater accuracy, which is not desirable for this application. We will discuss predictions around this threshold of four feet in more detail in Section 4.

3 | MODEL FORMULATION

The emulation problem considered in this paper presents the challenge of building emulators that are able to handle 4,000 runs from SLOSH, each with $n_y = 3.5 \times 10^6$ response values. One very common approach to reduce the dimension of a problem like this is to decompose the data into principal components (PCs; Ramsay & Silverman, 1997) using a singular value decomposition (SVD). The output vector $\boldsymbol{y}(\boldsymbol{x}) \in \mathbb{R}^{n_y}$ from one SLOSH run, corresponding to inputs $\boldsymbol{x} \in \mathbb{R}^p$ can be represented on a set of orthogonal basis functions as

$$\boldsymbol{y}(\boldsymbol{x}) = \sum_{j=1}^{\infty} w_j(\boldsymbol{x})\boldsymbol{b}_j \tag{2}$$

where $\boldsymbol{b}_j \in \mathbb{R}^{n_y}$ captures the spatial variation. By stacking the output obtained from each of the m storms in the training set, we obtain the matrix $\boldsymbol{Y} \in \mathbb{R}^{m \times n_y}$, which we center by subtracting the mean storm. $\boldsymbol{Y}_{ik}$ then corresponds to the standardized output from storm i at location k. We compute $SVD(\boldsymbol{Y}) = \boldsymbol{UDV}^T$ where $\boldsymbol{U}, \boldsymbol{V}$ are orthogonal matrices and $\boldsymbol{D}$ is a diagonal matrix of singular values. $\boldsymbol{V}^T$ and $\boldsymbol{UD}$ store the empirical $w_j(\boldsymbol{x})$ and $\boldsymbol{b}_j$ respectively. We choose to truncate the sum in Equation (2) at n_{pc} principal components, so that 99% of the variation in the data is captured by the basis representation. Our comparison involves repeated analysis with different sized training sets and the number of principal components required varies by training set. The smallest training set with only 50 storms requires just 14 principal components to capture 99% of the variation in the data. On the other hand, the largest training set with 3636 storms required 24 components. The power of this decomposition comes from the fact that, rather than fitting an emulator to all n_y response values, we only need to fit n_{pc} scalar response models to the coefficients $w_j(\boldsymbol{x})$, which results in drastic computational savings. We utilize the identical matrix decomposition when fitting BASS, BART, SEPIA, and the linear model. RobustGaSP does not make use of this representation, as discussed. In Subsections 3.1–3.3 we will suppress the subscript j for simplicity and refer to an arbitrary $w_j(\boldsymbol{x})$ as $w(\boldsymbol{x})$.

Truncating the sum in Equation (2) at n_{pc} components leads to some level of reconstruction error. We accept this so long as the important features are captured. Figure 3 shows a handful of representative basis vectors computed using a training set of 1000 storms. These basis vectors give strong evidence that important coastline features are captured. Additionally, in our early exploration we considered fitting models with up to 50 basis vectors, but saw no improvement in scores over the n_{pc} chosen to capture 99% of the variability in the training data.

3.1 | Simulation enabled prediction and inference

SEPIA is a python code developed by Jim Gattiker, Natalie Klein, Grant Hutchings and Earl Lawrence at Los Alamos National Laboratory (Gattiker, Klein, et al., 2020) and implements the model described in Higdon et al. (2008), with extensions. Here we use the emulator component only, without SEPIA's full model calibration functionality. By utilizing the orthogonal basis representation described above, a Gaussian process is fit to each basis function coefficient $w(\boldsymbol{x})$.

$$w(\boldsymbol{x}) \sim GP(0, \boldsymbol{\Sigma}); \ \boldsymbol{\Sigma} = \sigma_n^2 \boldsymbol{I} + \sigma_p^2 \boldsymbol{C} \tag{3}$$

where $\boldsymbol{C}_{kl} = \exp\{-\frac{1}{2}\sum_{i=1}^{p}\beta_i(\boldsymbol{x}_{ki} - \boldsymbol{x}_{li})^2\}$ is the matrix obtained by applying the negative exponential squared (”Gaussian”) correlation function to each pair of inputs, which is parameterized by length scale $\boldsymbol{\beta}$. $\boldsymbol{\Sigma}$ incorporates process variance σ_p^2 and includes a noise process with variance σ_n^2. This is a Bayesian model with priors on $\boldsymbol{\beta}, \sigma_p^2, \sigma_n^2$. For a full model

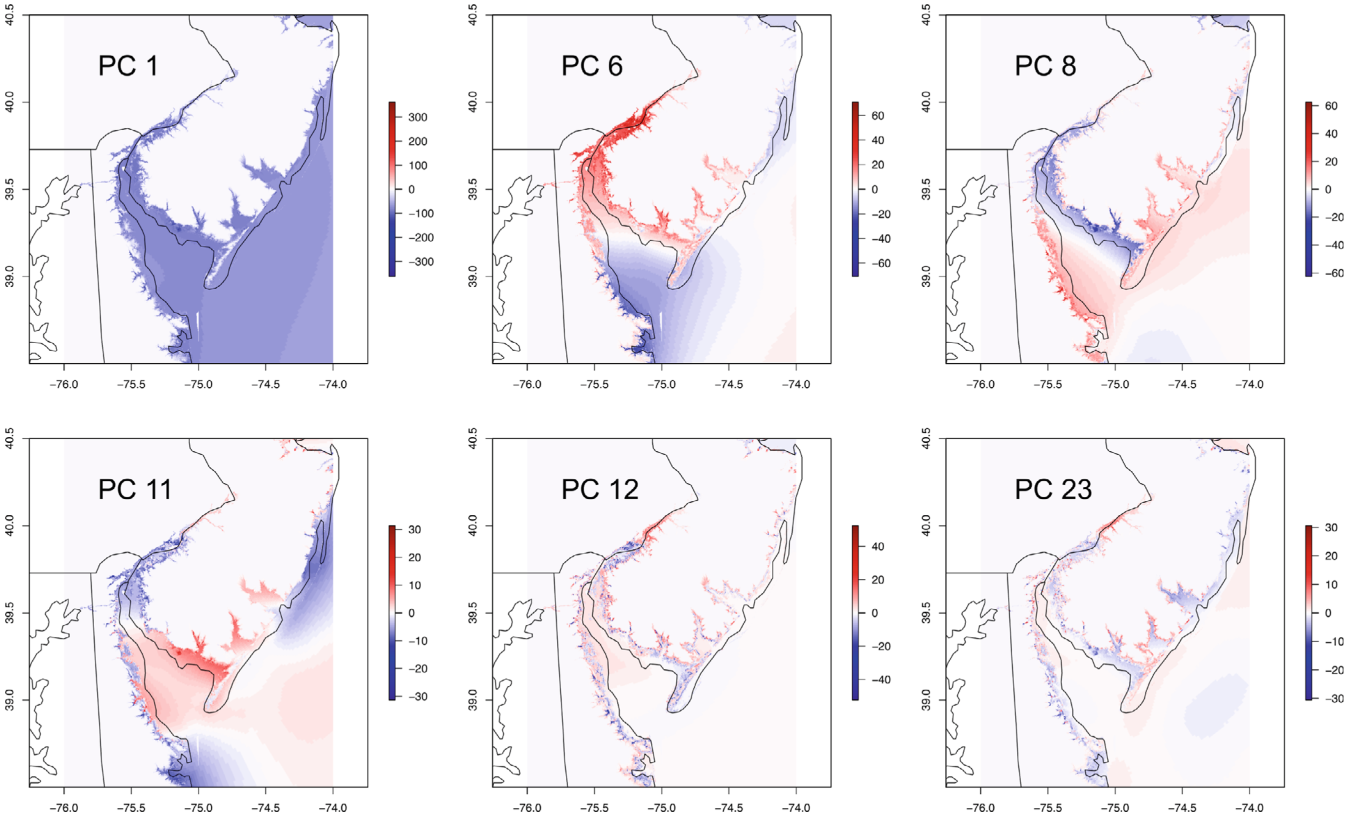


FIGURE 3 Spatial representation of basis vectors. PC 1 more or less captures the mean, while subsequent components capture coastline features

specification including discussion of priors, refer to Higdon et al. (2008). The resulting posterior distributions are explored via MCMC.

3.2 | Bayesian adaptive spline surfaces

BASS is an R package to fit Bayesian adaptive spline surfaces (Francom & Sansó, 2020). It implements a Bayesian version of multivariate adaptive regression splines (Friedman, 1991). Similar to the approach we took with SEPIA, we make use of a basis representation for the SLOSH output. BASS models each $w(\boldsymbol{x})$ as

$$w(\boldsymbol{x}) = a_0 + \sum_{m=1}^{M} a_m Z_m(\boldsymbol{x}) + \epsilon(\boldsymbol{x}), \quad \epsilon(\boldsymbol{x}) \sim N(0, \sigma^2) \tag{4}$$

where $a_0, a_1, \dots, a_M$ are constants and $Z_1, \dots, Z_M$ are basis functions learned from the data. The basis functions have the form

$$Z_m(\boldsymbol{x}) = \prod_{k=1}^{K_m} g_{km} \max\left(0, s_{km}(x_{v_{km}} - t_{km})\right)^{\alpha} \tag{5}$$

where $s_{km} \in \{-1, 1\}$ is the sign, $t_{km} \in [0, 1]$ is a knot, v_{km} selects a covariate, K_m is the degree of interaction and $g_{km} = [(s_{km} + 1)/2 - s_{km}t_{km}]^{\alpha}$ is a constant that makes the basis function have a maximum of one. The exponent α defines the degree of the polynomial splines. Note that variables can only be used once in each basis function.

To fit this model we need to estimate $\boldsymbol{\theta} = \{\sigma^2, M, \boldsymbol{a}, \boldsymbol{K}, \boldsymbol{s}, \boldsymbol{t}, \boldsymbol{v}\}$. This is done via a reversible jump MCMC (RJMCMC) algorithm (Green, 1995). For specifics on priors and the RJMCMC algorithm see Francom and Sansó (2020).

3.3 | Bayesian additive regression trees

BART is a treed model with strong predictive power for non-linear responses. A recent example is the use of BART for spatial modeling of ambient fine particulate matter pollution (PM_2.5) over California (Zhang et al., 2020). As detailed in Chipman et al. (2010), BART is a sum of trees model where scalar output $w(\boldsymbol{x})$ is approximated as

$$w(\boldsymbol{x}) = \sum_{i=1}^{I} g(\boldsymbol{x}|T_i, M_i) + \epsilon, \quad \epsilon \sim N(0, \sigma^2) \tag{6}$$

where each T_i is a regression tree that can incorporate one or more of the p inputs, corresponding to main and interaction effects. A tree T utilizing $\boldsymbol{x}_t \subseteq \boldsymbol{x}$ consists of a set of interior nodes with binary decision rules, and a set of leaf nodes containing parameter estimates. Let $M = \{\mu_1, \dots, \mu_b\}$ be the parameter estimates associated with the leaf nodes. The interior decision rules are binary splits of the predictor space, either $\boldsymbol{x}_t \in A$ or $\boldsymbol{x}_t \notin A$ where A is a subset of the range of $\boldsymbol{x}_t$. Then any fixed $\boldsymbol{x}_t^*$ is assigned a μ^* by the function $g(\boldsymbol{x}|T, M)$ based on the sequence of decision rules leading to a leaf node.

This additive structure endows BART with a high degree of flexibility when the number of trees is large. This does however come at the price of complexity. BART needs to estimate $\{(T_1, M_1), ..., (T_I, M_I), \sigma\}$ for I trees where T_i and M_i are not single parameters, but an entire tree structure fit with a set of decision rules, and a set of terminal nodes respectively. A backfitting MCMC algorithm is used for posterior sampling, which is designed to efficiently sample the many parameters in the additive tree structure. As a result, BART provides great flexibility with a relatively low computational cost. A key component of the model is a regularization prior which forces the effect from each tree to be small. This prevents individual tree effects from dominating the additive structure. Once posterior draws $(T_1^*, M_1^*), ..., (T_I^*, M_I^*)$ are available, predictions f^* can be obtained as

$$f^*(\cdot) = \sum_{i=1}^{I} g(\cdot|T_i^*, M_i^*) \tag{7}$$

(Sparapani et al., 2021).

3.4 | Robust Gaussian stochastic process emulation

RobustGaSP (Gu et al., 2017) is a GP-based method that avoids the use of the basis function representation that we have used for SEPIA, BASS and BART. Also, unlike the other three models the estimation procedure relies on marginal likelihood optimization rather than MCMC. This has its drawbacks when it comes to UQ as confidence bounds must be estimated using distributional assumptions. On the other hand it avoids the iterative sampling involved in MCMC, which incurs relatively large computational cost and memory footprint.

RobustGaSP is an implementation of a computationally feasible alternative to the Many Single (MS) emulation approach (Conti & O'Hagan, 2010; Lee et al., 2011; Lee et al., 2012). Individual emulators are fit to each coordinate of the output, which, in the context of our case study, consists of n_y independent Gaussian process emulators. Each emulator has its own mean function and variance, but they all share the same correlation parameters $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_p)$, which are estimated from the joint marginal likelihood of all emulators (Gu & Berger, 2016).

Let $i = 1, \dots, n_y$ index the locations so that $y_i(\boldsymbol{x})$ denotes the scalar response at location i with inputs $\boldsymbol{x}$. $y_i(\boldsymbol{x})$ is modeled with the Gaussian Process

$$y_i(\boldsymbol{x}) \sim GP(\mu_i(\boldsymbol{x}), \sigma_i^2 c(\boldsymbol{x}, \boldsymbol{x}')), ; \; i = 1, ..., k \tag{8}$$

where $\mu_i(\boldsymbol{x})$ is the location specific mean function, σ_i^2 the location specific variance, and $c(\boldsymbol{x}, \boldsymbol{x}')$, by default, is the product of p Matèrn 5/2 correlation functions, each with its own range parameter $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_p)$. Then for m runs of the simulator at inputs $\boldsymbol{x}_1, \dots, \boldsymbol{x}_m$ we have the multivariate likelihood

$$\left(y_i(\boldsymbol{x}_1), ..., y_i(\boldsymbol{x}_m)|\boldsymbol{\mu}_i, \sigma_i^2, \boldsymbol{\Sigma}\right) \sim \mathbf{MVN}\left((\mu_{i\boldsymbol{x}_1}, ..., \mu_{i\boldsymbol{x}_m}), \sigma_i^2 \boldsymbol{\Sigma}\right) \tag{9}$$

where $\mathbf{\Sigma}$ is the correlation matrix obtained by applying $c(\boldsymbol{x}, \boldsymbol{x}')$ to each pair of input vectors. The mean function is modelled using a linear regression, $\mu_i(\boldsymbol{x}) = \sum_{l=1}^{L} h_l(\boldsymbol{x})\theta_l$, with basis functions $\boldsymbol{h}_i(\boldsymbol{x}) = (h_{i1}(\boldsymbol{x}), \dots, h_{iL}(\boldsymbol{x}))$ and unknown regression parameters $\boldsymbol{\theta}_i = (\theta_{i1}, \dots, \theta_{iL})$. Here we use the default basis function, a single $\boldsymbol{h}(\boldsymbol{x}) = \mathbf{1}$ representing a constant mean. This is appropriate given that we centered our response to mean zero. An important aspect of this approach is the definition of the prior for the model parameters. This consists of the product of a standard objective prior is for the mean and variance parameters (Gu & Berger, 2016),

$$\pi^R(\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_{n_y}, \sigma_1^2, \dots, \sigma_{n_y}^2) \propto \frac{1}{\prod_{i=1}^{n_y} \sigma_i^2} \tag{10}$$

and a jointly robust (JR) prior applied to the correlation parameters $\boldsymbol{\gamma}$. This prior was introduced in Gu (2019) and is called jointly robust because is cannot be written as the product of marginal priors and its robust in marginal posterior mode estimation.

First consider reparameterizing to the inverse range parameters $\beta_j = 1/\gamma_j, j = 1, \dots, p$. Then the JR prior is defined as

$$\pi^{JR}(\beta_1, \dots, \beta_p) = C_0 \left(\sum_{l=1}^{p} C_l \beta_l \right)^{\alpha} \exp\left\{ -b \left(\sum_{l=1}^{p} C_l \beta_l \right) \right\}, \tag{11}$$

where $C_0 = \frac{(p-1)! b^{a+p} \prod_{l=1}^{p} C_l}{\Gamma(a+p)}$, $a > -(p+1)$, $b > 0$ and $C_l > 0$ are parameters. We use the default values for these parameters; $a = 0.2$, $b = n^{-1/p}(a+p)$. The default values for C_l are not clearly given in the documentation. As we will discuss in Section 5, this prior facilitates the form of variable importance provided by the package. The posterior distribution resulting from this model formulation is marginally optimized to obtain parameter estimates.

3.5 | Linear model

For a baseline comparison, we include a simple linear model on the EOF basis coefficients $w(\boldsymbol{x})$ with the form

$$w(\boldsymbol{x}) = \sum_{i=1}^{p} \beta_i x_i + \epsilon, \epsilon \sim N(0, \sigma^2) \tag{12}$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)$ are unknown regression coefficients which we determine using the function `lm` from base R (R Core Team, 2020).

3.6 | Model tuning

A potential drawback of this study is that model hyperparameters are not tuned using cross validation. However, we do not believe this is a significant limitation given the models considered. BART is a sum of trees model combined with a regularization prior forcing each tree to be a weak learner, contributing only a small amount to the overall fit. The relevant tuning of BART lies in the number of trees and the hyperparameters of this regularization prior. BART uses the data to empirically determine the hyperparameters. Sparapani et al. (2021), the authors of BART, recommend using these default priors when strong prior information is lacking. They show these default priors and number of trees to be “remarkably effective" over a range of examples. To summarize Sparapani et al. (2021), BART is designed to preform well as an untuned model in part because the prior parameters are estimated from the data. Green and Kern (2012) have also found the default setting in BART to work well for a variety of real and simulated data and find that there is often little to be gained from tuning the BART model via cross-validation.

Similar to BART, BASS fitting is designed to be automatic, with very little (if any) hyperparameter tuning. Francom et al. (2019) found that BASS performs well in many cases without tuning. If a user chose to tune BASS, they could change the prior on the number of basis functions to alter the level of regularization, the prior on the residual variance, and the prior on degree of interaction that each basis function can attain. However, in a setting with a large number of training model runs (like ours) we are very unlikely to overfit, so that regularizing the number of basis functions is unnecessary and the residual variance can easily be learned. We allow for up to three way interactions in our basis functions (the default), but recognize that this could be increased and potentially tuned. In practice, however, the authors have found that higher order interactions are rare and are often sufficiently captured with combinations of lower order interaction effects. There are other options for tuning the MCMC algorithm, such as tempering and slightly modifying some RJMCMC proposals, but these are merely for MCMC mixing and are not actually changing the model (and hence do not need to be tuned if the MCMC has converged).

There is very little to be done for the Gaussian Process models in the way of tuning. The prior for RobustGaSP was designed specifically to provide robust parameter estimation. Tuning this model comes down to the choice of correlation function. RobustGaSP allows for the Màtern 3/2 or 5/2 as well as the powered exponential correlation function. In this massive data setting we are severely limited in our ability to explore different correlation functions due to runtime constraints. RobustGaSP is the slowest model considered. Tuning the correlation function might be feasible for $m = 50$ testing storms, but certainly not $m > 50$ as the model takes over 30 hours to run for $m = 500$. Similarly, without strong prior information on the GP parameters, tuning SEPIA is limited to the choice of correlation function. In this case we are limited by the fact that SEPIA only has the Gaussian correlation function implemented. Because of our inability explore the range of correlation functions available, we use the default Màtern 5/2 in RobustGaSP which affords some comparison with the Gaussian correlation used by SEPIA, although the models are different enough that strong conclusions about correlation function cannot be drawn from differences between the two.

We have alluded to another important reason for leaving models untuned which is the scale of the tuning problem for these data. Consider that tuning multiple hyperparameters for BASS and BART models would require a large cross-validation exercise for hundreds of models over the considered training sets and principal components. Tuning of the GP models becomes infeasible not because of the number of models involved, but in the time required for model fitting. We are limited in our ability to fit these models with moderately sized training sets even with state of the art supercomputing resources. Lastly, we have specific interest in the effect of training ensemble size on predictive capability. A comparison of these methods for a single fixed training set might make the hyperparameter tuning exercise feasible and could be a natural extension of this work.

4 | COMPARISON STUDY

This section presents assessment of the four different emulators on the basis of out-of-sample predictive accuracy, uncertainty quantification, and computational feasibility. Predictions and UQ are assessed using scores including root-mean-squared error (RMSE), energy score, interval score (Gneiting & Raftery, 2007), coverage probability, and some application specific scores. Section 4.1 compares the accuracy of the mean predictive surface, and Section 4.2 considers estimates to be used in uncertainty quantification. Our results indicate that in these metrics GP based emulators (SEPIA and RobustGaSP) may produce better mean predictions than BASS and BART with less training data, but as the training data set increases in size, the non-GP methods provide similar accuracy. In terms of uncertainty quantification, the choice of scoring rule may dictate choice of emulator, therefore strong conclusions cannot be drawn.

We would like to be able to train our models with as few storms as needed for accuracy, while minimizing computation time and leaving more examples in the model test set. To examine the impact of training set size (m) for each emulator we consider seven different training sets with 50, 100, 500, 1000, 1750, 2500, and 3636 storms. 3636 was chosen as the largest training set size because it is the largest number that permits a testing set size of approximately 10% of the training set (364 testing storms). The largest training set was sampled randomly once from the full 4000 storm ensemble, and subsequent training sets sampled randomly from this set of 3636. While randomly sub-sampling a space-filling design is not optimal, the same selection is used for each emulator, meaning that within training set size, the emulators are trained on the same data, affording fair comparison.

Our comparison study involves training each of the four emulators on each of the seven training sets, and computing all prediction metrics on the testing set. All models are tested on the same 364 storms. This allows estimation of the impact of training set size, and comparison of performance both within and between these training set sizes. Computation time is compared across training set sizes revealing the scaling properties of each algorithm. Our results underline an important and well known fact that GPs are excellent predictors, but become prohibitive with large data-sets. In fact, we were only able to fit SEPIA and RobustGaSP with a maximum of 1000 and 500 training storms respectively. We will discuss this further in Section 4.3.

BASS, BART, and SEPIA all make use of MCMC for parameter estimation. BASS and BART are quite fast affording 100,000 samples. We discard the first 50,000 samples to eliminate transient state (so-called "burn-in"). For SEPIA we are somewhat constrained by longer MCMC computation time, so we collect a more modest 25,000 samples and discard the first 10,000. SEPIA initializes it's MCMC with a step-size tuning routine that is quite effective at eliminating the transient phase. With step-size tuning, we found that MCMC appeared well converged by 10,000 samples, and this seems to be confirmed by computing the effective sample size. We describe our methods for determining model convergence in Section 4.4. Due to computation time constraints we could only collect 10,000 samples at $m = 1750$ and SEPIA appeared to not be converged at that point. For this reason we do not present results for SEPIA at $m > 1000$. Because of the size of the spatial field, we thin the remaining samples down to 50, driven by memory constraints on our computational resources. To fully appreciate the memory challenge, recall that our testing set is 364 storms. 1000 posterior samples predictive samples requires a double precision matrix of size $(364 \times 3{,}500{,}000 \times 1000)$, which requires 10 terabytes of storage. We are limited on our platforms to a more modest 500 gigabyte matrix resulting from the use of 50 samples. This is one of the many challenges involving an application dataset of this size. We appreciate that given the relatively small number of retained samples (50), there may be questions regarding how well the 50 samples represent the posterior distributions. We give consideration to MCMC convergence in Section 4.4, but results should be viewed with the understanding that issues due to small a sample set are potentially present in predictive metrics of accuracy and coverage. To have more confidence that the retained samples are representative, we would have to recommend reducing the computation by further reduction of the spatial data to make investigation tractable.

4.1 | Predictive accuracy

In this section we will assess out-of-sample predictive accuracy for each emulator. Predictive accuracy is assessed through scoring rules on the mean predicted storm surge map from each emulator. For all but RobustGaSP, predictions of the random basis coefficients are made using 50 posterior samples from the model and each predictive sample is multiplied through by the basis vectors to build a predicted storm surge map. The mean predicted storm surge map is composed of the mean surge prediction at each cell over the 50 samples. For RobustGaSP, the model is fit on native storm surge units and the predictive mean storm surge map is returned by the package. Our assessment considered RMSE, mean absolute error (MAE), and a domain specific loss (fragility) function designed specifically for infrastructure risk analysis which incorporates the sign of the residuals by penalizing under-prediction more severely than over-prediction. We found that MAE and our fragility function showed the same information as RMSE, so those results and a discussion are left in the supplementary material. We also considered mean relative absolute error (MRAE), but found it did not produce results useful in the context of the application. A discussion of relative errors can be found in the supplement. We also consider each emulator's ability to correctly predict storm-surge events greater than the four foot threshold of risk to electrical power substations.

4.1.1 | RMSE

Figure 4 shows boxplots of RMSE for each emulation method and for each training set size at which they were run. Samples in each correspond to the 364 test storms dataset.

As expected, RMSE is generally decreasing with training set size. The plots show diminishing returns, with a reasonable conclusion that a training set size greater than 1000 runs is unnecessary to achieve near best performance in RMSE. Additionally, the figures show that the GP based methods, SEPIA and RobustGaSP, tend to have the lowest RMSE for smaller training sets of $m < 1000$. For $m > 500$ SEPIA performs on par with BASS and BART. We do not know how RobustGaSP would perform with these larger training sets due to computational constraints.

We present RMSE, and will present additional scores averaged over the spatial domain. Spatial aggregation is useful to summarize results for different scores, emulators, and training sets, but provides no information of location specific performance for the different emulators. Some exploration of spatial behavior is presented in Section 4.1.3. In addition, scores such as RMSE lack information regarding over and under-prediction, or other application specific diagnostic. Thus, we consider both general error measures and examples of scores tailored to this application. In the following subsection, we will discuss our emulators' ability to correctly predict risk to electrical power stations, which is directly related to the sign of residuals.

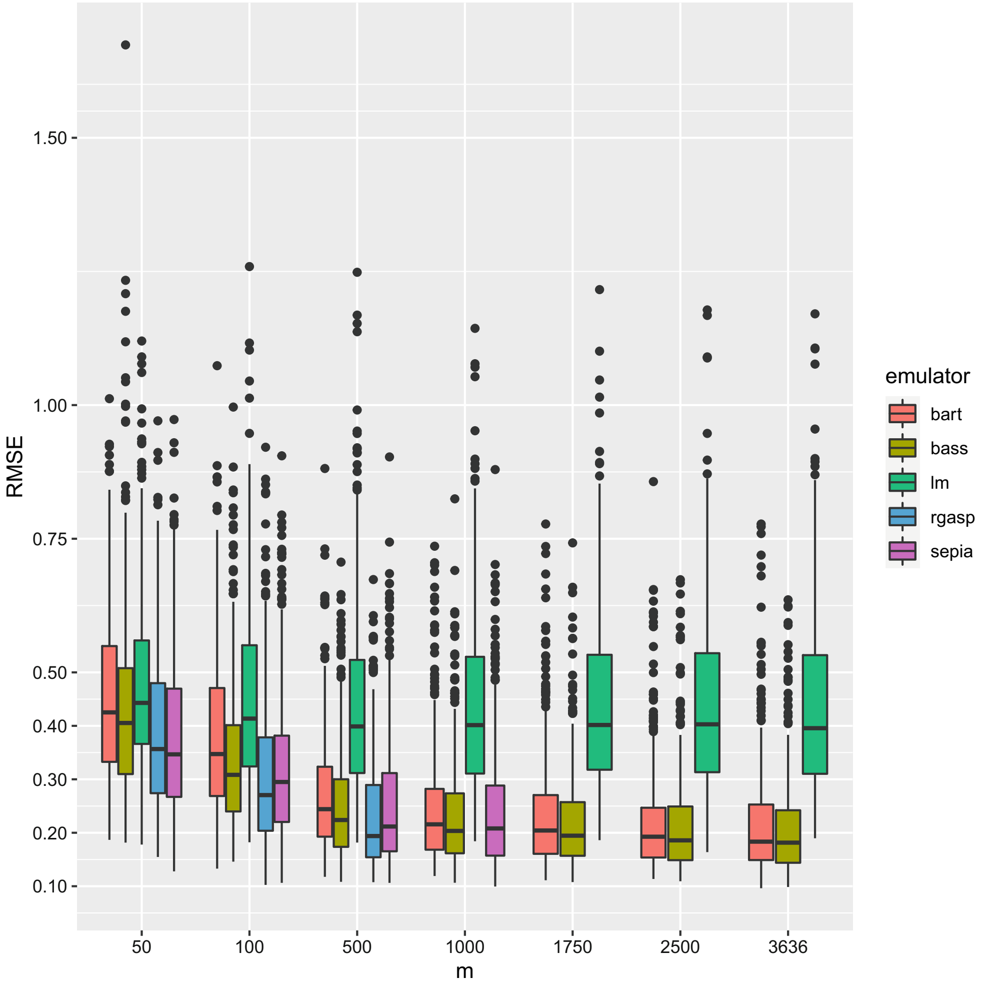


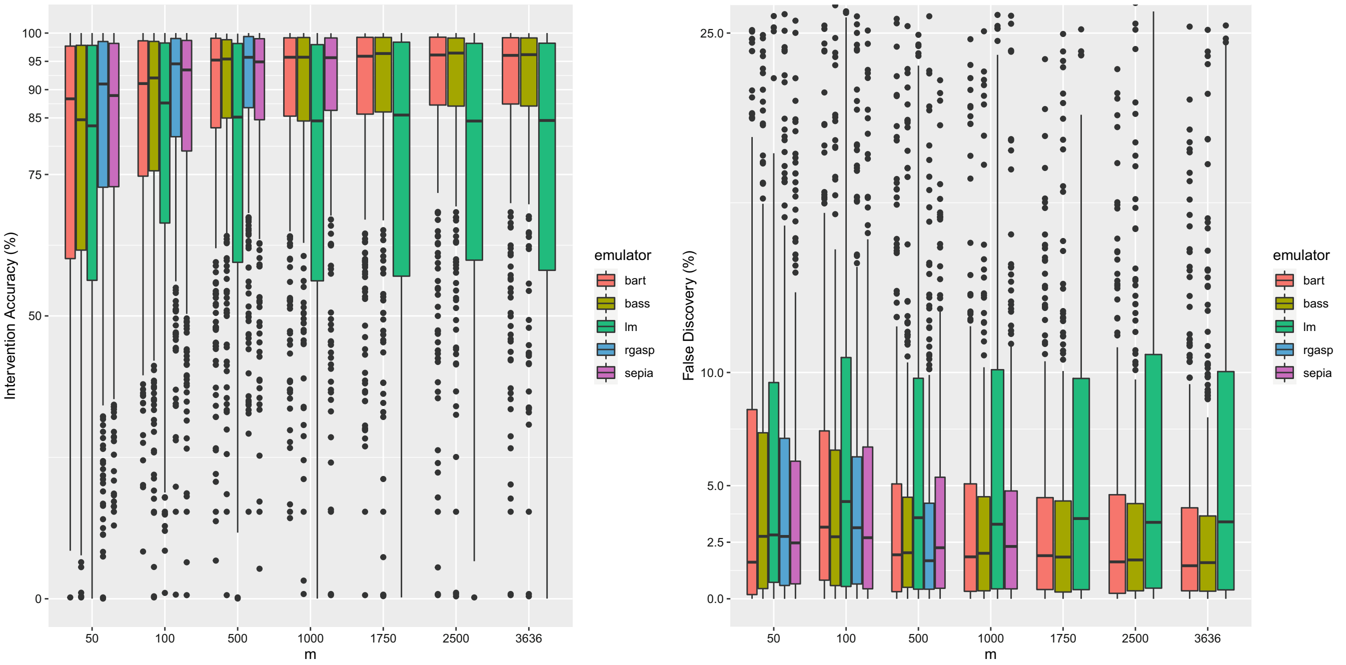
FIGURE 4 RMSE by training set size with distributions over test storms

4.1.2 | Power station risk

In Section 2, we noted that a flooding threshold of four feet is of special interest. This number has real implications in that many power stations are fortified to withstand this level of flood water.[1] Therefore, it is desirable for an emulator to correctly predict flood level above four feet, a domain-relevant criterion for evaluation. We will evaluate this with standard emulation methods, rather than creating an emulator to satisfy the application-specific loss. To assess these emulators with respect to this feature, we consider the percentage of predictions that correctly indicate that an intervention is needed, which we call the intervention accuracy. To compute this metric for the mean prediction, we consider all cells in which the true SLOSH output is greater than four feet, and determine the percentage of cells in which the prediction is also greater than four feet. Figure 5a shows boxplots of our results where distributions are over the 364 testing storms. We find that there is not a great deal of difference between the emulators especially for $m > 100$. For small m BASS seems to under-perform while BART and the GP based methods are best. BASS improves significantly at $m = 100$, slightly beating BART, but the GP methods are still superior. At $m = 500$ there is almost no difference between the methods (excluding lm), all of which have a median intervention accuracy of about 95%.

We also consider the rate at which each emulator falsely determines that an intervention is necessary. We call this the false discovery rate, which is important as shutting down a station unnecessarily could be costly. This score is calculated as the percentage of cells where the prediction is greater than four feet, but the true value is less than four feet. The results

[1] Different flood impact thresholds can be found in the literature. The four foot threshold is driven by our application context of US infrastructure planning, and is indicative of threshold-based evaluation of emulators.



(a) Intervention Accuracy

(b) False Discovery

FIGURE 5 Accuracy in predicting threshold storm-surge events

are shown in Figure 5b and we see that as we increase the training set size, distributions generally become narrower and the median approaches about 2%. Differences between emulators are not large, and performance is not improved much over a training set of size of 500–1000, where the interquartile range for all emulators aside from the linear model sit below about 5%. All emulators perform surprisingly well in the median at $m = 50$, especially BART. BART, the linear model, RobustGaSP, and SEPIA all perform better in the median at $m = 50$ compared to $m = 100$ while BASS is approximately the same. Tail performance improves with increasing m for all methods. The y-axis has been limited to the range [0, 25] so that differences between the emulators are more apparent. This does leave some outliers out of the plot. We do not find the structure of these outliers to be important, but a plot with the full y-axis scale is shown in the supplementary material.

4.1.3 | Hurricane severity

Another comparison of interest is an emulator's ability to model hurricanes of varied intensity. High storm surge has been seen from both low and high category hurricanes so we would hope to understand if certain emulators do better or worse by hurricane severity. To compare predictions across hurricane category, we split our testing set in two; hurricanes of category less than three and those of category three, four, and five. Figure 6 shows spatial residuals for emulators trained on $m = 1000$ storms ($m = 500$ for RobustGaSP) where cells are averaged over testing storms. Cells in the left panels are averaged over all 364 storms, the middle panels are averaged over only lower category storms, and the right panels only high category storms. The small area shown in Figure 6 was chosen for the high density of electrical power substations in this region (see Figure 1). We can see that residuals in native units of feet are mostly negative for low category storms indicating over-prediction and positive for higher category storms indicating under-prediction. When averaging over all storms, we tend to overestimate, which may be due to the ensemble dataset set having more storms of category 0–2. While there is a general trend of under-prediction of the high category storms, the GP based methods still seem to have some over-prediction inside the bay. Previous results indicate that the linear model is not competitive, so in the interest of brevity, we do not include it in this comparison.

The over(under)-estimation of low(high) category storms seen in Figure 6 indicates that models are not capturing tail behavior especially well. One possible way to improve tail prediction is a transformation of the response. Specifically in this case, the large number of zero surge values make domain transformation approaches, for example, log or logistic,

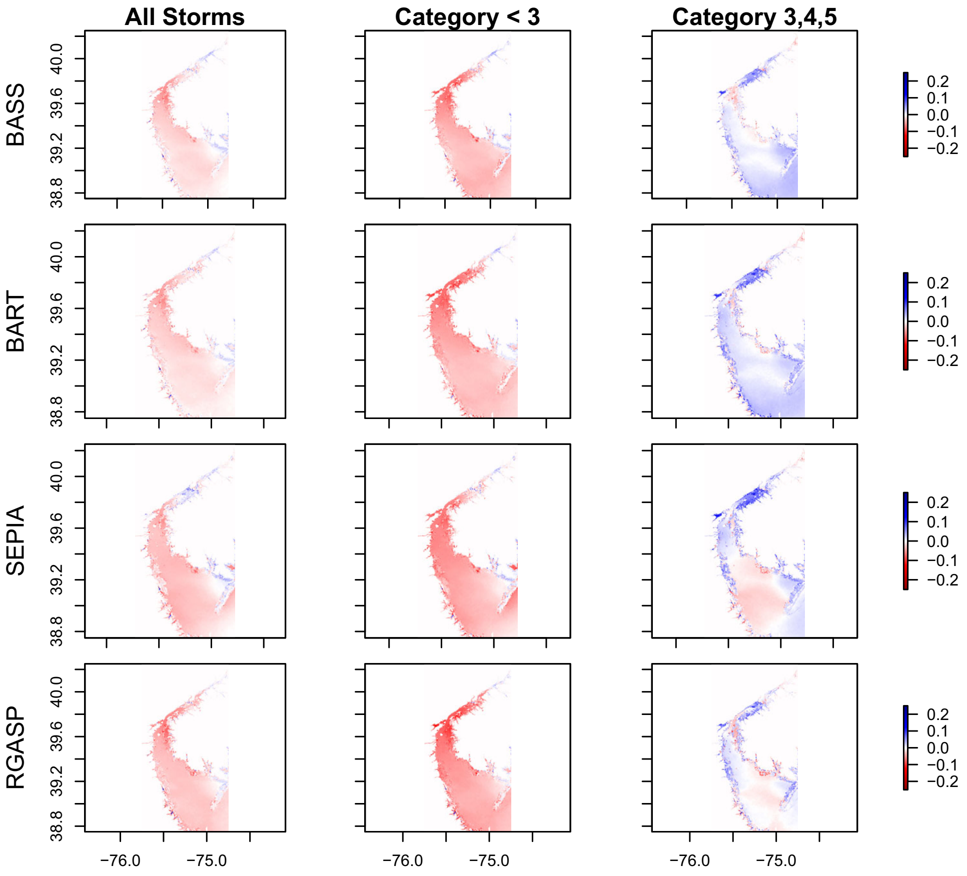


FIGURE 6 Spatial error maps by storm severity. Emulators trained on 1000 storms (500 for RobustGaSP)

challenging. Another possible way to improve tail prediction is to explore error structures beyond Gaussian such as a heavy-tailed t-distribution. All emulation methods considered use a Gaussian error structure, and it might be that a heavy tailed distribution is more suited for these data. There is a large literature on addressing prediction dispersion issues in statistical theory and practice, and this is a topic for detailed application beyond method intercomparison, particularly as here we see that all methods have similar outcomes. We have special interest in a flooding threshold of four feet and as such we are not especially focused on prediction near the tails. For different applications such as worst case flood analysis, a practitioner with special interest in predictive accuracy for extreme response values may want to explore response transformations, an emulator with a different error model, or take a different approach such as emulating surge quantiles.

4.1.4 | Comparing error structure

Here we compare the sensitivity of predictive accuracy to changes in input variables. In the interest of space, we present only a single score (RMSE) at a single representative training set size ($m = 1000$). Figure 7 shows input variables versus marginal RMSE, along with lowess fits. The linear model is excluded given its lack of competitive results in other scores.

Even though these methods use different modeling strategies, the RMSE structure is similar. In fact, they are within regression uncertainty in all cases. Furthermore, the RMSE values between the emulators are very highly correlated, as shown in Table 1.

Figure 7 shows small trends in the structure of the RMSE curves, such as decreasing RMSE with increasing minimum air pressure, and a convex RMSE curve for heading; however, these trends are relatively small compared to the spread of error. From an application point of view the underlying reasons for these trends are unclear, and a potential topic for future research. An in-depth analysis of these trends is outside the scope of a methods intercomparison, beyond the main point that all methods show similar behavior in this respect.

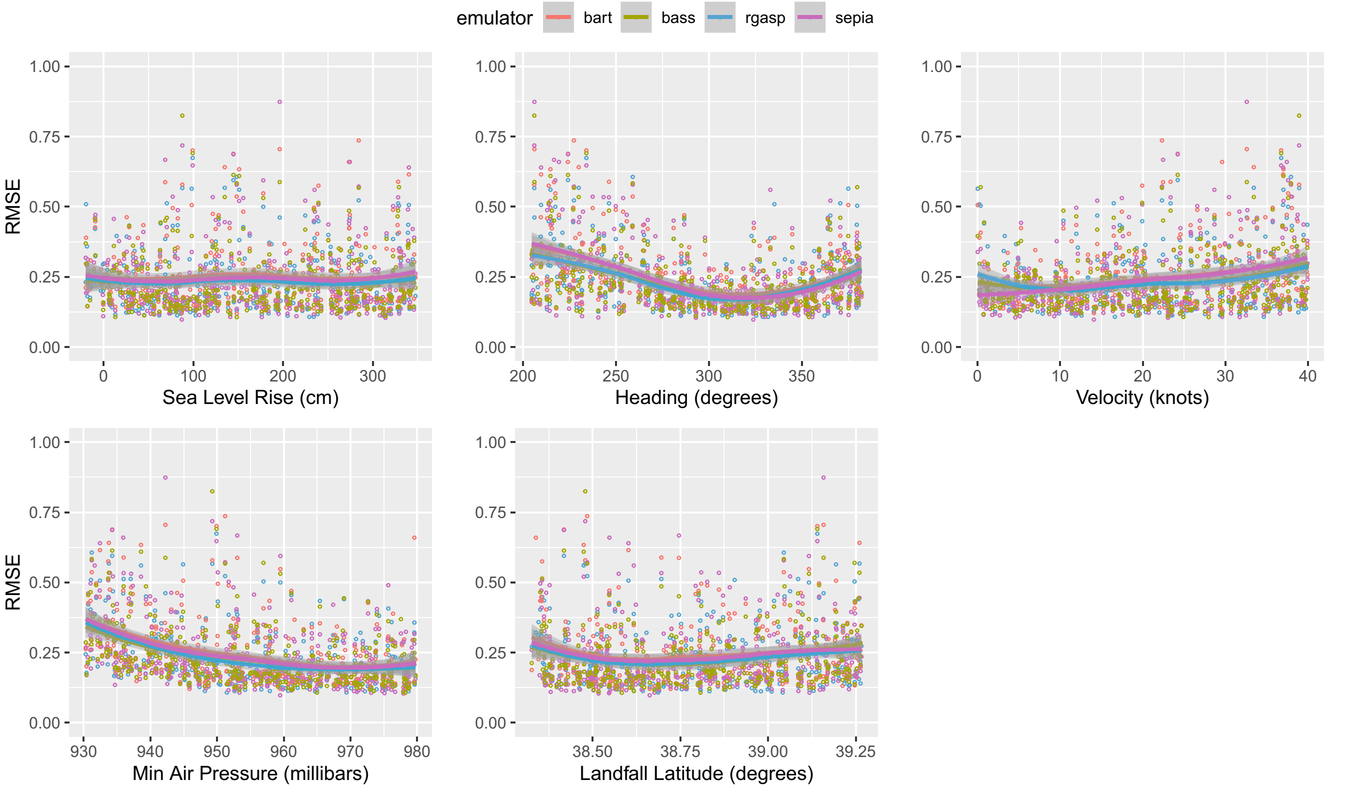


FIGURE 7 Changes in RMSE with respect to test storm inputs

TABLE 1 Correlations between test set RMSE for $m = 1000$ ($m = 500$ for rgasp)

	Bass	Bart	Sepia	Rgasp
Bass	1.00	0.89	0.86	0.91
Bart	0.89	1.00	0.87	0.86
Sepia	0.86	0.87	1.00	0.82
Rgasp	0.91	0.86	0.82	1.00

4.2 | Predictive uncertainty

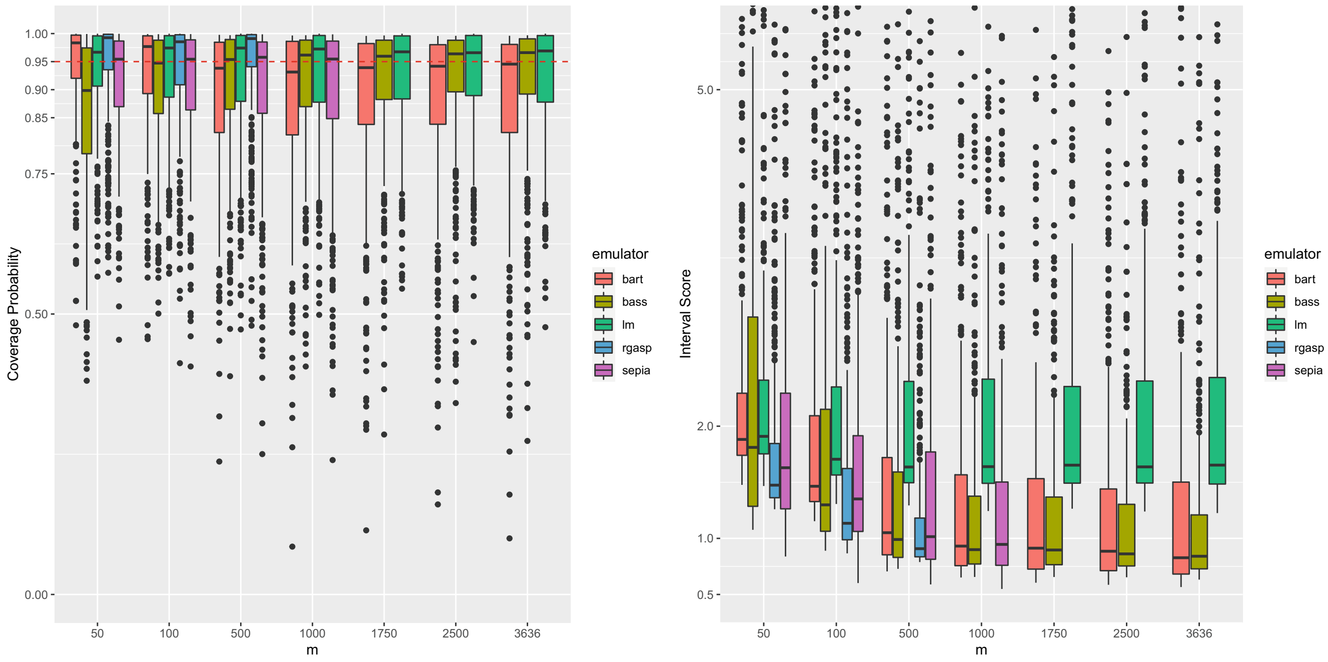
This section presents the results of predictive metrics which take uncertainty into consideration: *coverage probability*, *energy score*, and *interval score*.

4.2.1 | Coverage

In Figure 8a, we present coverage probability distributions for 95% intervals over the 364 testing storms. Using the dashed red line at 0.95, we can see that SEPIA consistently provides near 95% coverage at all training sets. BART tends to over-cover with small training sets and under-cover with larger training sets. BASS does the opposite. RobustGaSP and the linear model consistently over-cover. We extend our assessment of coverage by comparing the models using a score proposed in Gneiting and Raftery (2007), the interval score.

The interval score for confidence level α is defined as

$$S_\alpha^{int}(l, u; x) = (u - l) + \frac{2}{\alpha}(l - x)\mathbb{1}\{x < l\} + \frac{2}{\alpha}(x - u)\mathbb{1}\{x > u\}. \tag{13}$$



(a) Coverage probability, 95% interval.

(b) Interval scores for 95% confidence level.

FIGURE 8 Coverage metrics

Where l, u are the lower and upper bounds of the $1 - \alpha$ confidence interval, and x is the true data value. This is a negative oriented score that is minimized at the width of the interval. The score then increases proportional to α if the true data value is outside the interval. This score provides more insight than coverage probability by consciously favoring models with the smallest possible intervals that still contain the data. In Figure 8b, we present interval score distributions over the 364 testing storms where each storms score is an average over scores for each cell. For $m < 1000$ RobustGaSP provides the best interval score. It seems that its wider confidence bounds that lead to over-coverage are offset in the interval score. SEPIA performs consistently well over all m providing better or comparable performance to BASS and BART. It is difficult to clearly distinguish between BASS and BART. BASS has slightly better median performance for all m, but wider distributions at small m indicating poorer performance on some testing storms. At larger m their median performance is very similar, but BART then has wider distributions. The visual upper limit in Figure 8b has been chosen so that differences between the emulators are easier to see which results in some upper outliers not being shown. The structure of these outliers does not give insight, but for completeness a similar plot is left in the supplementary material without a limited y-axis.

4.2.2 | Energy score

The energy score, a multivariate extension of the Continuous Rank Probability Score (CRPS) is proposed in Gneiting and Raftery (2007). This score takes into account not only the predictive accuracy of each sample from the posterior predictive distribution, but also the level of uncertainty in the distribution. For this reason, the CRPS and energy score have gained interest in recent literature as a model ranking mechanism (Heaton et al., 2018; Möller et al., 2013; Muniain and Ziel, 2020). With m draws from the posterior predictive distribution of Y, $\tilde{Y} = \{\tilde{Y}_1, \tilde{Y}_2, \ldots, \tilde{Y}_m\}$, we compute the energy score as

$$es(Y, \tilde{Y}) = \frac{1}{m}\sum_{j=1}^{m} ||\tilde{Y}_j - Y|| - \frac{1}{2m^2}\sum_{j=1}^{m}\sum_{k=1}^{m} ||\tilde{Y}_j - \tilde{Y}_k||, \tag{14}$$

where Y is the true response.

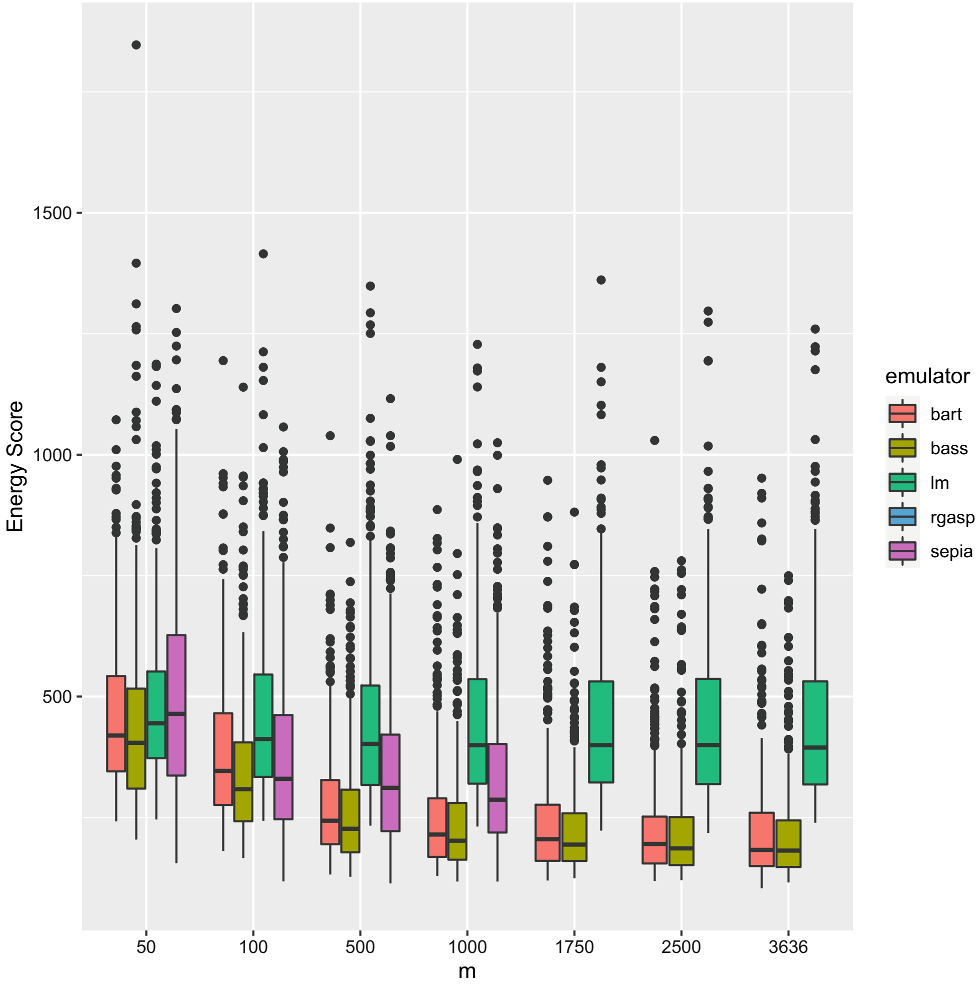


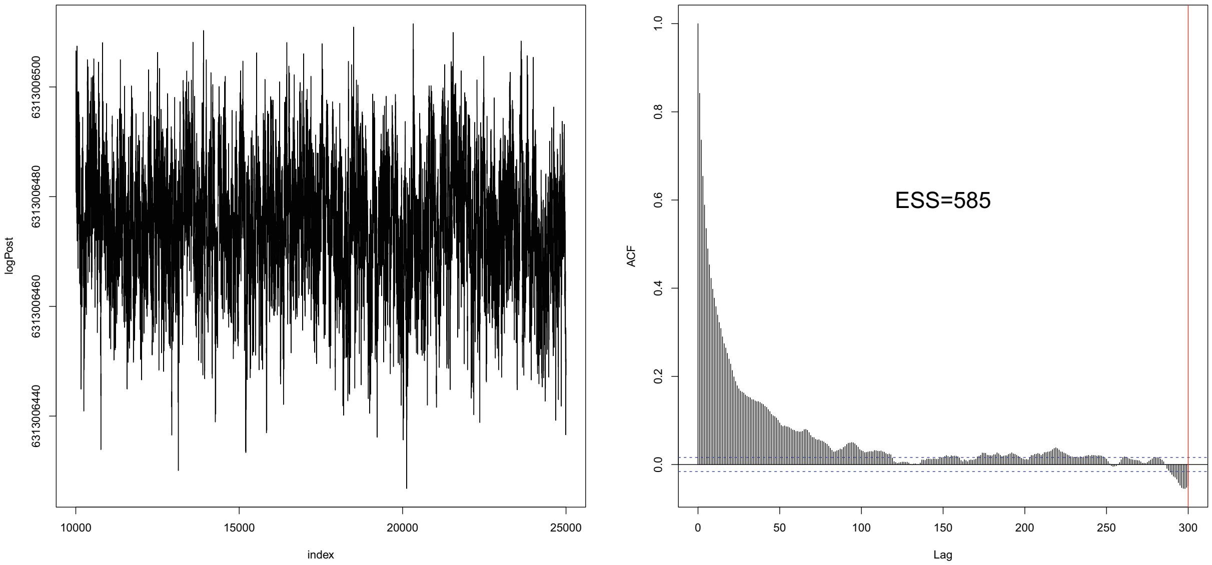
FIGURE 9 Energy score by training set size

Results from Figure 8 indicate that the GP methods provide good uncertainty quantification. They tend to have lower or equivalent Interval Scores at each m and SEPIA provides very consistent coverage. Interestingly, the energy score shown in Figure 9, which explicitly uses predictive samples rather than the just the mean and 95% intervals tends to favor the tree and spline based models over SEPIA. We therefore cannot make strong claims about which models might provide the best UQ. We do not present results for RobustGaSP here because it does not give samples from the posterior predictive distribution, only a mean and 95% prediction interval.

4.3 | Computational feasibility

Computation time is an important aspect of any comparison of emulators especially on large data sets where some methods are simply not feasible. All of the models were built on a Los Alamos National Laboratory compute cluster node with an AMD EPYC 7513 CPU @ 2.6 GHz. This is a 64 core CPU which allowed us to parallelize over the principal components where possible.

As expected, the baseline linear model is extremely fast and scales well, but it performs relatively poorly in predictive metrics. We can see that BASS remains relatively fast and scales well over the range of training set sizes, requiring about 2 min at $m = 1000$ and 4 min at $m = 3636$ for 100,000 MCMC samples. BART is noticeably slower but still feasible, requiring about 12 min for $m = 1000$ and 45 min at $m = 3636$ for 100,000 MCMC samples. By comparison, SEPIA took nearly 39 h for 25,000 samples at $m = 1000$ and RobustGaSP took over 35 h for $m = 500$. This is because both methods make use of Gaussian Process which is inherently $O(m^3)$ scaling. SEPIA makes use of optimized likelihood calculations and the



(a) Trace plot oflog posterior for SEPIA.

(b) ACF plot for SEPIA chain in panel (a). Red line at lag used for predictive samples.

FIGURE 10 MCMC diagnostics for SEPIA model trained on $m = 1000$ storms

scaling we see here is closer to $O(m^2)$ for a fixed number of MCMC samples, although that still becomes quickly becomes infeasible for large m. RobustGaSP is the slowest of the four emulators exhibiting approximately $O(m^3)$ scaling. This is perhaps not surprising given the scope of the optimization problem it is addressing on the native response space.

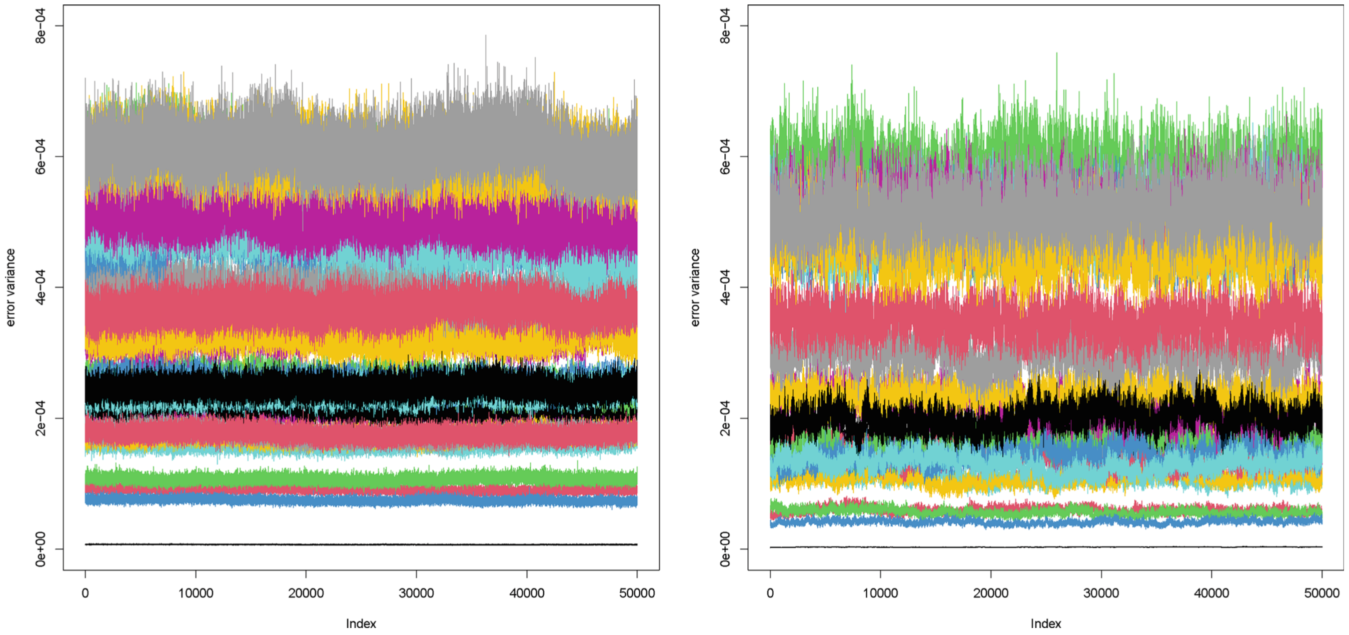
Parallel MCMC chain approaches may be able reduce execution time for SEPIA by a fixed factor admitting somewhat larger problems, but will not change the inherent scaling. In the performance analysis previously presented, there is evidence that SEPIA and RobustGaSP can be competitive in prediction with BASS and BART with smaller training sets and the largest training sets are not needed for near optimal prediction.

4.4 | Model convergence

To assess the convergence of our models we compute the effective sample size (ESS) of post burn-in samples as well as visually inspect the chains for signs of transient behavior. We use the R package Coda (Plummer et al., 2006), a package for MCMC diagnostics to calculate ESS and require that $ESS > 50$ for our post burn-in samples. In practice we almost always have $ESS >> 50$ for the 15,000 retained samples from SEPIA and 50,000 from BASS and BART for each principal component model.

To visually assess the convergence of SEPIA, we looked at the single log-posterior chain for each training set size as convergence of this chain is indicative of model convergence. We plot this chain for $m = 1000$ along with the ACF for the post burn-in samples in Figure 10 and see that the auto-correlation for the posterior samples used is small, and the $ESS = 585$.

For BASS and BART we look at the error variance chains (σ^2) for each of the n_{pc} models at each training set. BASS provides a routine for MCMC convergence assessment which plots the posterior samples for the number of basis functions and the error variance. In Figure 11, we show the post burn-in error variance chains for $m = 1000$ which show some auto-correlation, but no significant transient behavior. Samples of the error variance are also used by the authors of BART to assess model convergence in Sparapani et al. (2021), so we make a similar plot for the BART models. For these chains we have mean ESS over the n_{pc} models of 781 and 463 for BASS and BART respectively. Note that error variance is generally increasing from principal component models 1 to 24. The chains with smaller error variance are from models for principal components which account for the most variability in the data, and convergence of these models is more indicative of overall model convergence. For $m = 1000$ over 95% of the variability in the data is captured in the first 4 components. Each of the remaining 20 components account for $< 1\%$ of the variability.



(a) Trace plots of error variance chains for BASS models.

(b) Trace plots of error variance chains for BART models.

FIGURE 11 MCMC diagnostics for BASS and BART models trained on $m = 1000$ storms

5 | VARIABLE IMPORTANCE

In Section 1.2.1, we described the type of variable importance each model provides. Here we give representative results for $m = 1000$ ($m = 500$ for RobustGaSP). This section is not a direct comparison of like quantities, but rather a presentation and qualitative comparison of the different information available from the methods to the user.

Using the Sobol decomposition routine provided by the BASS package we show selected main effect Sobol indices colored by the square root of the explained variance in panels $(a) - (d)$ of Figure 12. The results from Figure 12 indicate that uncertainties in Sea Level Rise will contribute the most uncertainty to storm surge predictions, which may be important in a decision making framework. We also see that velocity is most important at the northern opening to the bay. Minimum pressure has an important effect inside the bay and along the coastline to the north.

Panel (e) shows the variance explained by the interaction between sea level rise and minimum pressure. The interaction effect is not as strong as the main effects in the water, but plays a stronger role at the flood boundary. Panel (f) shows the remaining variance unexplained by main effects and interactions of order 2. Our goal here is not to analyze these sensitivities in detail, but rather to demonstrate the information provided by the Sobol decomposition. These results were generated using uniform priors over the input parameter ranges. SEPIA also has built in functionality for computing Sobol indices which provides sensitivities for the original response, not just the EOF coefficients. Unfortunately, we found data of this size infeasible in the current implementation.

BART keeps track of how often each input variable is used in a decision rule for a regression tree. Figure 13a shows distributions over MCMC samples for the third principal component model and we notice that heading (theta), velocity (v), and latitude (lat) appear to be the most important inputs. This plot is more informative when combined with a visualization of the principal component as seen in Figure 13b. Now we can see that these inputs explain variability mostly near the northern coast between 39° N and 40° N. Combining information from these figures gives us an idea of the locations in space where certain inputs are having an important effect. We show PC3 rather than another PC simply because it shows interesting structure and provides a good example of the results that are available from BART.

Table 2 shows the estimated normalized inverse range parameters for our RobustGaSP model trained on 500 storms and we see that none of the inputs are found to be inert.

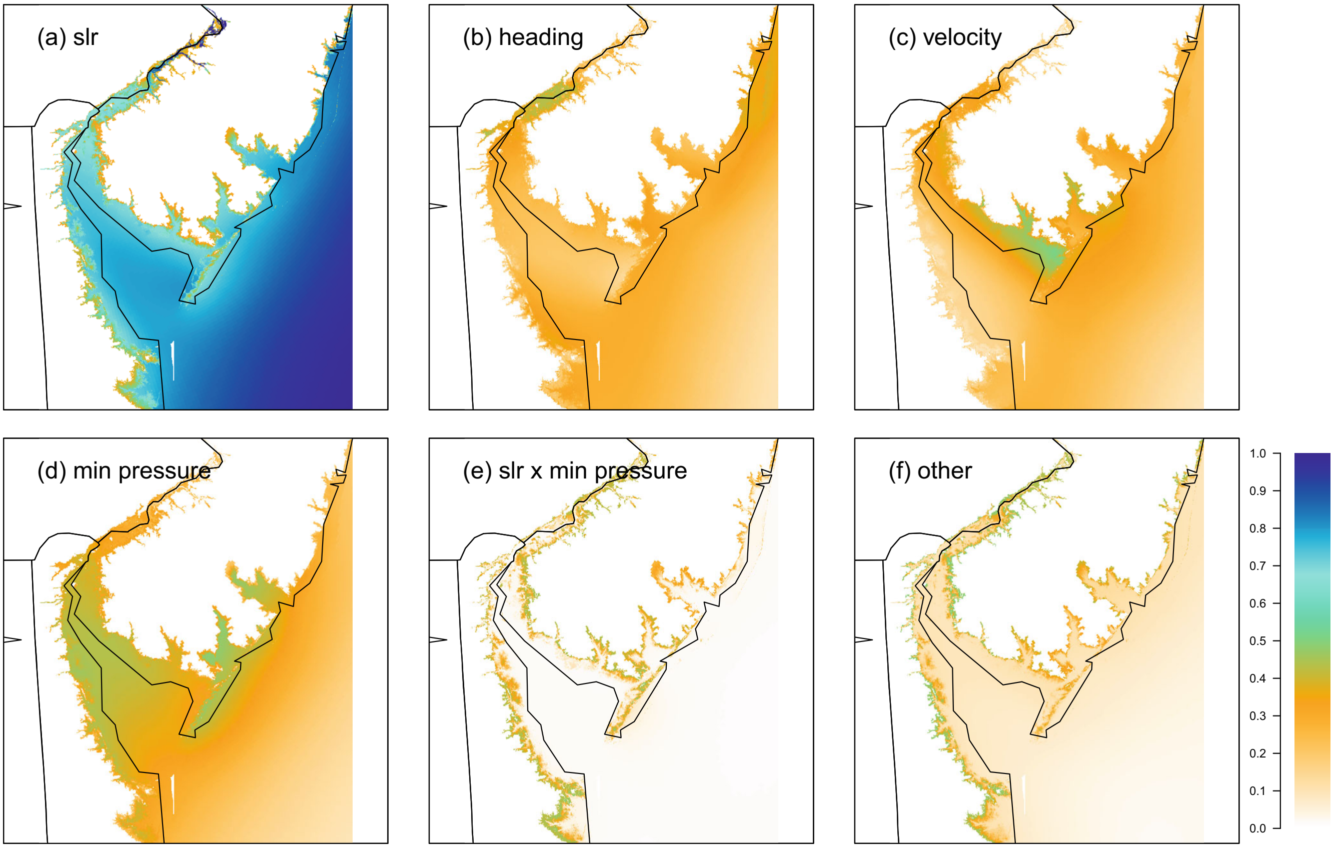
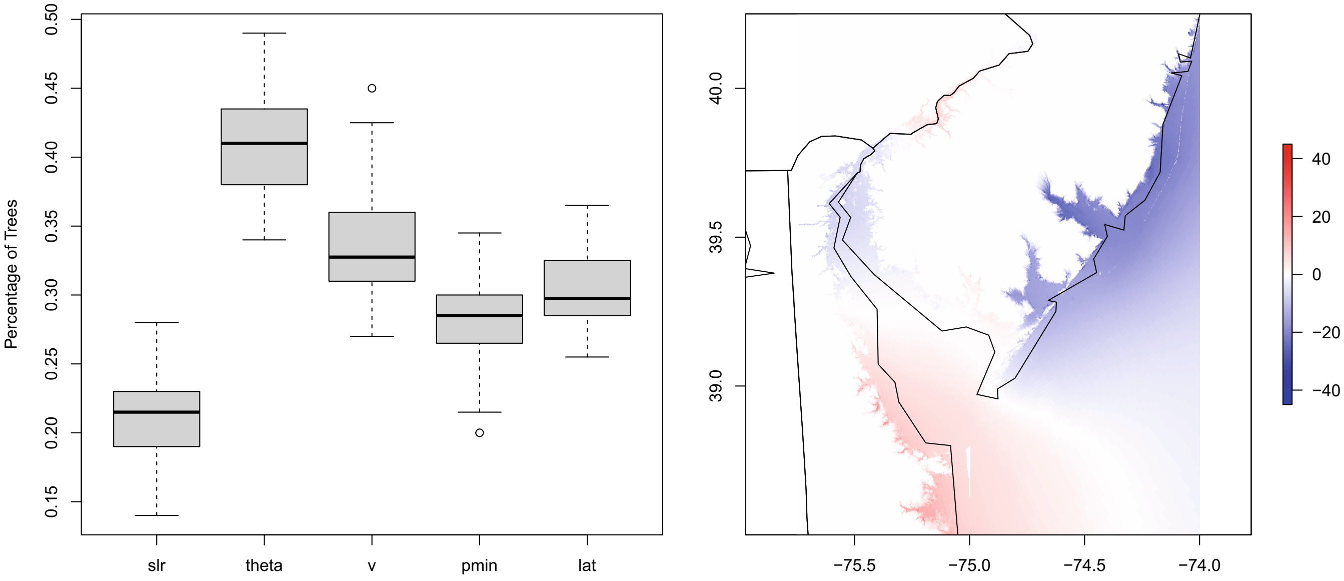


FIGURE 12 Bass Sobol indices, selected main and interaction effects. Color indicates the square root of the explained variance from each effect



(a) Percentage of regression trees containing each input for the Principal Component 3 model.

(b) Principal Component 3 basis spatial map

FIGURE 13 BART variable importance

TABLE 2 Estimated normalized inverse range parameters

Sea level rise	Heading	Velocity	Min pressure	Latitude
0.58	2.50	1.15	0.34	0.43

6 | DISCUSSION

We have presented an in-depth comparison of emulation methods applied to a SLOSH simulator application, and have documented valuable insights into emulator choice under a variety of conditions. An important aspect of simulation studies is ensemble size, whose magnitude may be constrained by computational limitations of simulation or emulation. Due to the relative speed of running the SLOSH simulator, we have a generous ensemble of 4000 runs which allowed for a training set size comparison. Studies using more complex storm surge simulators like ADCIRC (Luettich et al., 1992), which incorporates more sophisticated physics models, as well as any spatially-resolved model in increased resolution, may not have this luxury. We believe our training set size comparison can be a useful starting point for ensemble creation in future emulation studies. Predictive accuracy is always important, but correctness of quantified uncertainty is also critical to trust in analysis results. We have therefore compared these methods using a variety of metrics, focusing on different views of predictive accuracy and uncertainty and their potential trade-off. We have special interest in risk associated with electrical power substation damage. For this reason we define adaptable threshold based metrics with which to compare emulators.

While our study over training sets provides valuable insights, it also leads to a potential drawback which is that we do not tune models to these specific data using cross-validation. Section 1.2 discusses why tuning BART and BASS is likely not a worthwhile effort, and tuning SEPIA and RobustGaSP is infeasible given the large ensemble dataset of interest. Regardless, results must be viewed with the understanding that for each score, investing further effort in customizing models could change results. For a practitioner, hyperparameter tuning via hold-out testing may be useful if computational resources allow. SEPIA and RobustGaSP provides a built in cross validation routines, however these may be impractical for even moderate ensembles. BART also provides built-in hold-out testing and BASS can be easily adapted for the task. For GP based methods the choice of correlation function may be important. SEPIA only has the Gaussian correlation function available, but RobustGaSP has the powered exponential as well as the Màtern 3/2 and 5/2. We only consider the Màtern 5/2 here due to limits on model runtime, but a practitioner may want to perform a cross-validation exercise over the available correlation functions, which may result in improved prediction.

Figures 4 and 5a show that for our case study, GP based models produce the most accurate mean predictions and are best able to predict necessary power station interventions for small training sets. SEPIA also provides the most consistent coverage across all training sets and good performance in the interval score. This however comes at a significant computational cost as seen in Figure 14. Therefore, we recommend SEPIA when the size of the ensemble is relatively small with correspondingly tractable computational time. We do not believe the slight improvement in some scores associated with RobustGaSP is a worthwhile trade off given its computational requirements. In many applications efficiency is likely to be very important, and for these BASS would be preferred. BASS tends to slightly outperform BART in our predictive metrics such as RMSE, energy score, and interval score and it is relatively computationally tractable. Additionally BASS supplies intuitive variable importance analysis through Sobol indices, as demonstrated. Results from Section 5 point to sea level rise as the input which explains the most variability in storm surge. Minimum pressure also explains a significant amount of variability within the bay, while heading and velocity have more localized importance. Uncertainties around these parameters may therefore be important to decision makers.

In terms of application specific scores such as Intervention Accuracy SEPIA and RobustGaSP perform better for small m but by $m = 500$ there is no notable difference between models. We considered additional application-specific flooding and risk analysis related metrics which can be found in the supplementary materials. Specifically, we looked at predictions for the area and volume of catastrophic flooding, where area is defined as the number of land cells with greater than four feet of flood water, and volume is defined as the total water depth summed over all catastrophically flooded locations. We did not find our results to add a significant amount of information regarding the emulator methods directly to our already rich comparison. There is also a description in the supplementary materials of an asymmetric loss function that we created to penalize emulators more heavily for under-prediction. This is of interest as a tunable metric that can express

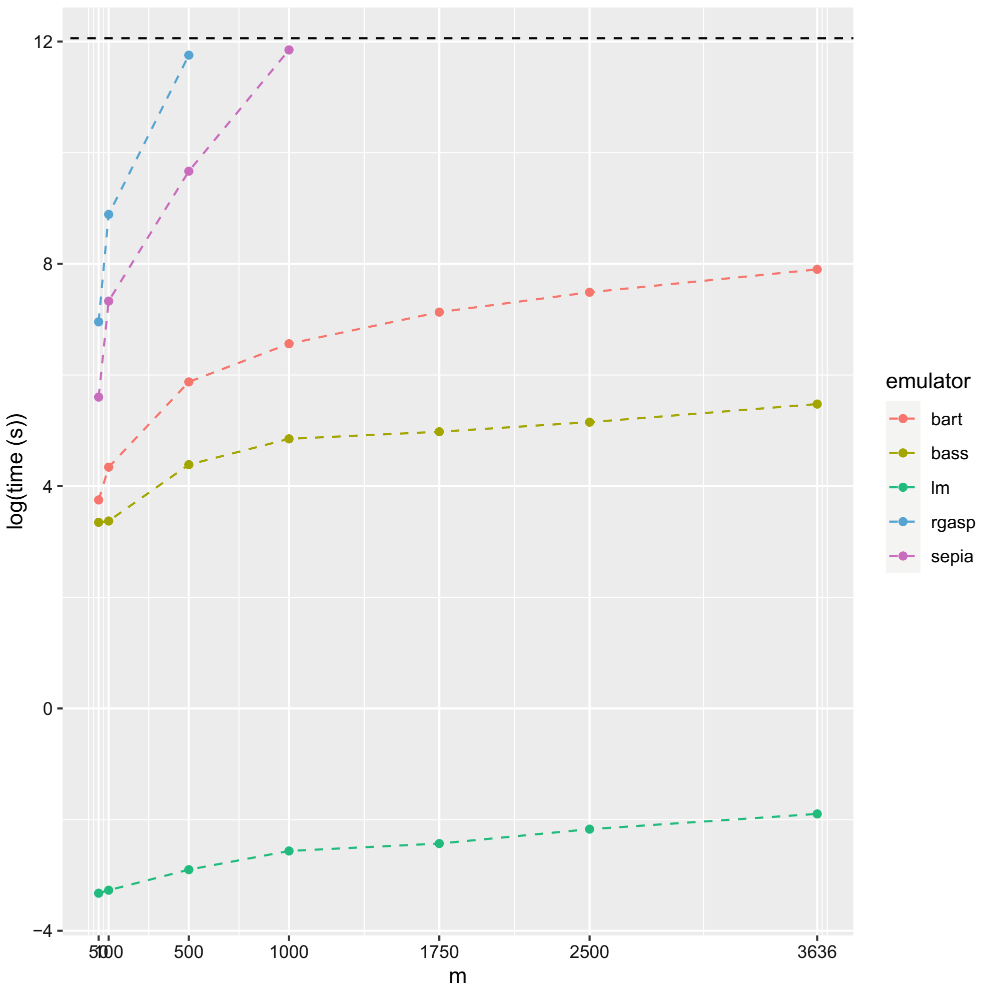


FIGURE 14 Model fit time

risk-aversion of decision-makers, especially surrounding the four foot threshold that results in power station damage. We applied this loss function to mean predictions and again found the results showed no significant difference for the purposes of comparative evaluation, when compared to RMSE.

In future work, we would like to confront some of the questions and limitations that arose during this study. One of these is the outliers seen in all scores. It is clear that some storms are performing very poorly for our predictive metrics, and although some were examined, a systematic characterization is not yet clear for emulation comparison. Initial exploration indicates that poorly predicted storms are often near the boundary of the parameter space for the training set, or even require extrapolation. Improved design for the training sets is likely to alleviate this issue in many cases. Section 4.1.4 indicates that outliers are fairly consistent across methods, but an extension of this work could study in more detail whether these outliers have particular features, for example a particular region of the parameter space. Another limitation that comes with data of this size is the storing of large matrices, which led us to use a relatively small number of posterior predictive samples. The question of these samples providing a representative set for the posterior distributions is difficult to answer. For RobustGaSP studying optimization over an increased number of initialized states to ensure global convergence may be useful, although some exploration of this did not indicate that our models converged to a local mode. As discussed, these analyses come with heavy computational burden and time that would likely not be available in a typical applied analysis. Lastly, in Section 4 we discussed the possibility of reducing the area of particular interest to the application context of power grid impacts, which would admit an effectively larger analysis within computational budget.

REFERENCES

Borge, R., Alexandrov, V., Del Vas, J. J., Lumbreras, J., & Rodríguez, E. (2008). A comprehensive sensitivity analysis of the wrf model for air quality applications over the Iberian Peninsula. *Atmospheric Environment*, *42*(37), 8560–8574. https://doi.org/10.1016/j.atmosenv.2008.08.032

Chipman, H. A., George, E. I., & McCulloch, R. E. (2010). Bart: Bayesian additive regression trees. *The Annals of Applied Statistics*, *4*(1), 266–298. https://doi.org/10.1214/09-AOAS285

Conti, S., & O'Hagan, A. (2010). Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference*, *140*(3), 640–651. https://doi.org/10.1016/j.jspi.2009.08.006

Emanuel, K., Ravela, S., Vivant, E., & Camille, R. (2006). A statistical deterministic approach to hurricane risk assessment. *Bulletin of the American Meteorological Society*, *87*(3), 299–314.

Erickson, C. B., Ankenman, B. E., & Sanchez, S. M. (2018). Comparison of Gaussian process modeling software. *European Journal of Operational Research*, *266*(1), 179–192. https://doi.org/10.1016/j.ejor.2017.10.002

Fassò, A., & Perri, P. F. (2002). Sensitivity analysis. *Encyclopedia of Environmetrics*, *4*, 1968–1982.

Francom, D., & Sansó, B. (2020). BASS: An R package for fitting and performing sensitivity analysis of Bayesian adaptive spline surfaces. *Journal of Statistical Software*, *94*(8), 1–36. https://doi.org/10.18637/jss.v094.i08

Francom, D., Sansó, B., Bulaevskaya, V., Lucas, D., & Simpson, M. (2019). Inferring atmospheric release characteristics in a large computer experiment using Bayesian adaptive splines. *Journal of the American Statistical Association*, *114*(528), 1450–1465. https://doi.org/10.1080/01621459.2018.1562933

Francom, D., Sansó, B., & Kupresanin, A. (2022). Landmark-warped emulators for models with misaligned functional response. *SIAM/ASA Journal on Uncertainty Quantification*, *10*(1), 125–150.

Francom, D., Sansó, B., Kupresanin, A., & Johannesson, G. (2018). Sensitivity analysis and emulation for functional data using bayesian adaptive splines. *Statistica Sinica*, *28*(2), 791–816.

Friedman, J. (1991). Multivariate adaptive regression splines (with discussion). *Annals of Statistics*, *19*, 1–67.

Gattiker, J., Higdon, D., & Williams, B. (2020). GPMSA. https://github.com/lanl/GPMSA.

Gattiker, J., Klein, N., Hutchings, G., & Lawrence, E. (2020). SEPIA: v1.1. https://doi.org/10.5281/zenodo.4048801.

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*(477), 359–378. https://doi.org/10.1198/016214506000001437

Graham, H. E., & Nunn, D. E. (1959). *Meteorological considerations pertinent to standard project hurricane, Atlantic and Gulf coasts of the United States. Technical Report National Hurricane Research Project* (Vol. *33*). *Weather Bureau, U.S. Department of Commerce*.

Gramacy, R. B., & Apley, D. W. (2015). Local Gaussian process approximation for large computer experiments. *Journal of Computational and Graphical Statistics*, *24*(2), 561–578. https://doi.org/10.1080/10618600.2014.914442

Gramacy, R. B., & Lee, H. K. H. (2008). Bayesian treed Gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, *103*(483), 1119–1130. URL http://www.jstor.org/stable/27640148

Green, D. P., & Kern, H. L. (2012). Modeling heterogeneous treatment effects in survey experiments with Bayesian additive regression trees. *Public Opinion Quarterly*, *76*(3), 491–511.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, *82*(4), 711–732. https://doi.org/10.1093/biomet/82.4.711

Gu, M. (2019). Jointly robust prior for Gaussian stochastic process in emulation, calibration and variable selection. *Bayesian Analysis*, *14*(3), 857–885. https://doi.org/10.1214/18-BA1133

Gu, M., & Berger, J. O. (2016). Parallel partial Gaussian process emulation for computer models with massive output. *Annals of Applied Statistics*, *10*(3), 1317–1347. https://doi.org/10.1214/16-AOAS934

Gu, M., Wang, X., & Berger, J. (2017). Robust gaussian stochastic process emulation. *Annals of Statistics*, *46*, 8. https://doi.org/10.1214/17-AOS1648

Heaton, M., Datta, A., Finley, A., Furrer, R., Guinness, J., Guhaniyogi, R., Gerber, F., Gramacy, R., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D., Sun, F., & Zammit-Mangion, A. (2018). A case study competition among methods for analyzing large spatial data. *Journal of Agricultural, Biological and Environmental Statistics*, *24*, 12. https://doi.org/10.1007/s13253-018-00348-w

Higdon, D., Gattiker, J., Williams, B., & Rightley, M. (2008). Computer model calibration using high-dimensional output. *Journal of the American Statistical Association*, *103*, 570–583. https://doi.org/10.1198/016214507000000888

Ho, F., Schwerdt, R., & Goodyear, H. (1970). *Some climatological characteristics of hurricanes and tropical storms, gulf and east coasts of the United States* (Vol. *15*). *NOAA Technical Report National Weather Service, U.S. Department of Commerce*.

Iooss, B., Da Veiga, S., Janon, A., & Pujol, G. (2021). Sensitivity: Global sensitivity analysis of model outputs. URL https://cran.r-project.org/web/packages/sensitivity/index.html. R package version 1.24.0

Irish, J. L., Resio, D. T., & Cialone, M. (2009). A surge response function approach to coastal hazard assessment: Part 2, quantification of spatial attributes of response functions. *Journal of Natural Hazards*, *51*, 183–205. https://doi.org/10.1007/s11069-009-9381-4

Jelesnianski, C. P., Chen, J., & Shaffer, W. A. (1992). *Slosh: Sea, lake, and overland surges from hurricanes. Technical report*. United States, National Weather Service URL https://repository.library.noaa.gov/view/noaa/7235. NOAA technical report NWS; 48

Johnson, C. A., Flage, R., & Guikema, S. D. (2021). Feasibility study of PRA for critical infrastructure risk analysis. *Reliability Engineering and System Safety*, *212*, 107643. https://doi.org/10.1016/j.ress.2021.107643

Katzfuss, M., & Guinness, J. (2021). A general framework for vecchia approximations of Gaussian processes. *Statistical Science*, *36*(1), 124–141. https://doi.org/10.1214/19-STS755

Keller, K., Helgeson, C., & Srikrishnan, V. (2021). Climate risk management. *Annual Review of Earth and Planetary Sciences*, *49*(1), 95–116.

Kennedy, M., & O'Hagan, A. (2001). Bayesian calibration of computer models (with discussion). *Journal of the Royal Statistical Society Series B*, *63*, 425–464. https://doi.org/10.1111/1467-9868.00294

Kopp, R., Horton, R., Little, C., Mitrovica, J., Oppenheimer, M., Rasmussen, D., Strauss, B., & Tebaldi, C. (2014). Probabilistic 21st and 22nd century sea level projections at a global network of tide-gauge sites. *Earth's Future*, *2*(8), 383–406. https://doi.org/10.1002/2014EF000239

Leary, S., Bhaskar, A., & Keane, A. (2003). Optimal orthogonal-array-based latin hypercubes. *Journal of Applied Statistics*, *30*(5), 585–598. https://doi.org/10.1080/0266476032000053691

Lee, L., Carslaw, K., Pringle, K., & Mann, G. (2012). Mapping the uncertainty in global ccn using emulation. *Atmospheric Chemistry and Physics*, *12*, 9739–9751. https://doi.org/10.5194/acp-12-9739-2012

Lee, L. A., Carslaw, K. S., Pringle, K. J., Mann, G. W., & Spracklen, D. V. (2011). Emulation of a complex global aerosol model to quantify sensitivity to uncertain parameters. *Atmospheric Chemistry and Physics*, *11*(23), 12253–12273. https://doi.org/10.5194/acp-11-12253-2011

Linkletter, C., Bingham, D., Hengartner, N., Higdon, D., & Ye, K. Q. (2006). Variable selection for Gaussian process models in computer experiments. *Technometrics*, *48*, 478–490. https://doi.org/10.1198/004017006000000228

Luettich, R. A., Westerink, J. J., & Scheffner, N. W. (1992). Adcirc: An advanced three-dimensional circulation model for shelves, coasts, and estuaries. Report 1, Theory and methodology of ADCIRC-2DD1 and ADCIRC-3DL.

Marrel, A., Iooss, B., Jullien, M., Laurent, B., & Volkova, E. (2011). Global sensitivity analysis for models with spatially dependent outputs. *Environmetrics*, *22*(3), 383–397. https://doi.org/10.1002/env.1071

Möller, A., Lenkoski, A., & Thorarinsdottir, T. L. (2013). Multivariate probabilistic forecasting using ensemble Bayesian model averaging and copulas. *Quarterly Journal of the Royal Meteorological Society*, *139*(673), 982–991. https://doi.org/10.1002/qj.2009

Muniain, P., & Ziel, F. (2020). Probabilistic forecasting in day-ahead electricity markets: Simulating peak and off-peak prices. *International Journal of Forecasting*, *36*(4), 1193–1210. https://doi.org/10.1016/j.ijforecast.2019.11.006

Myers, V. A. (1970). *Joint probability method of tide frequency analysis applied to Atlantic city and long beach island, n.j. U.S. Department of Commerce Environmental Science Services Administration Weather Bureau*. https://doi.org/10.7282/T3ZK5DVQ

Myers, V. A. (1975). *Storm tide frequencies on the south Carolina coast* (Vol. *16*). *NOAA Technical Report National Weather Service, U.S. Department of Commerce* URL https://www.weather.gov/media/owp/oh/hdsc/docs/TR16.pdf

NOAA Tides and currents, lewes, de - station id: 8557380. URL \protect\LY1\textbracelefthttps://tidesandcurrents.noaa.gov/stationhome.html?id=8557380}

Pasqualini, D. (2017). *Resilient grid operational strategies. Technical Report LA-UR-17-21753*. U.S. Department of Energy Office of Scientific and Technical Information https://www.osti.gov/biblio/1345917

Petersen, M. R., Asay-Davis, X. S., Berres, A. S., Chen, Q., Feige, N., Hoffman, M. J., Jacobsen, D. W., Jones, P. W., Maltrud, M. E., Price, S. F., Ringler, T. D., Streletz, G. J., Turner, A. K., Van Roekel, L. P., Veneziani, M., Wolfe, J. D., Wolfram, P. J., & Woodring, J. L. (2019). An evaluation of the ocean and sea ice climate of e3sm using mpas and interannual core-ii forcing. *Journal of Advances in Modeling Earth Systems*, *11*(5), 1438–1458. https://doi.org/10.1029/2018MS001373

Plummer, M., Best, N., Cowles, K., & Vines, K. (2006). Coda: Convergence diagnosis and output analysis for mcmc. *R News*, *6*(1), 7–11 URL https://journal.r-project.org/archive/

Pratola, M. T., Chipman, H. A., Gattiker, J. R., Higdon, D. M., McCulloch, R., & Rust, W. N. (2014). Parallel bayesian additive regression trees. *Journal of Computational and Graphical Statistics*, *23*(3), 830–852. https://doi.org/10.1080/10618600.2013.841584

R Core Team. (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing URL https://www.R-project.org/

Ramsay, J., & Silverman, B. W. (1997). *Functional Data Analysis*. Springer.

Resio, D. (1970). White paper on estimating hurricane inundation probabilities. Technical Report. *Coastal and Hydraulics Laboratory (U.S.) Engineer Research and Development Center (U.S.)* URL https://hdl.handle.net/11681/22643

Resio, D. T., Asher, T., & Irish, J. L. (2017). The effects of natural structure on estimated tropical cyclone surge extremes. *Journal of Natural Hazards*, *88*(3), 1609–1637 URL https://link.springer.com/article/10.1007/s11069-017-2935-y

Resio, D. T., Irish, J. L., & Cialone, M. (2009). A surge response function approach to coastal hazard assessment: Part 1, basic concepts. *Journal of Natural Hazards*, *51*, 163–182. https://doi.org/10.1007/s11069-009-9379-y

Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, *4*(4), 409–423. https://doi.org/10.1214/ss/1177012413

Salter, J. M., & Williamson, D. (2016a). A comparison of statistical emulation methodologies for multi-wave calibration of environmental models. *Environmetrics*, *27*(8), 507–523. https://doi.org/10.1002/env.2405

Salter, J. M., & Williamson, D. (2016b). A comparison of statistical emulation methodologies for multi-wave calibration of environmental models. *Environmetrics*, *27*(8), 507–523. https://doi.org/10.1002/env.2405

Santner, T. J., Williams, B. J., Notz, W., & Williams, B. J. (2018). *The design and analysis of computer experiments* (2nd ed.). Springer.

Schonlau, M., & Welch, W. J. (2006. ISBN 978-0-387-28014-1.). *Screening the input variables to a computer model via analysis of variance and visualization* (pp. 308–327). Springer. https://doi.org/10.1007/0-387-28014-6_14

Schwerdt, R. W., Ho, F. P., & Watkins, R. R. (1979). *Meteorological criteria for standard project hurricane and probable maximum hurricane wind fields, gulf and east coasts of the United States* (Vol. *33*). *NOAA Technical Report National Weather Service, Department of Commerce* URL https://www.weather.gov/media/owp/oh/hdsc/docs/TR23.pdf

Sobol, I. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, *55*(1), 271–280. https://doi.org/10.1016/S0378-4754(00)00270-6

Sparapani, R., Spanbauer, C., & McCulloch, R. (2021). Nonparametric machine learning and efficient computation with Bayesian additive regression trees: The BART R package. *Journal of Statistical Software*, *97*(1), 1–66. https://doi.org/10.18637/jss.v097.i01

Tang, B. (1993). Orthogonal array-based latin hypercubes. *Journal of the American Statistical Association*, *88*(424), 1392–1397. https://doi.org/10.2307/2291282

Toro, G., Niedoroda, A., Reed, D., & Divoky, C. (2010). Quadrature-based approach for the efficient evaluation of surge hazard. *Ocean Engineering*, *37*(1), 114–124. https://doi.org/10.1016/j.oceaneng.2009.09.005

Venturi, D., & Karniadakis, G. E. (2012). Differential constraints for the probability density function of stochastic solutions to the wave equation. *International Journal for Uncertainty Quantification*, *2*(3), 195–213. https://doi.org/10.1615/Int.J.UncertaintyQuantification.2011003485

Vickery, P., & Wadhera, D. (2008). Statistical models of Holland pressure profile parameter and radius to maximum winds of hurricanes from flight-level pressure and H×Wind data. *Journal of Applied Meteorology and Climatology*, *47*(10), 2497–2517. https://doi.org/10.1175/2008JAMC1837.1

Vickery, P. J., & Twisdale, L. A. (1995). Wind field and filling models for hurricane wind-speed predictions. *Journal of Structural Engineering*, *121*(11), 1700–1709. https://doi.org/10.1061/(ASCE)0733-9445(1995)121:11(1700)

Wong, T., & Keller, K. (2017). Deep uncertainty surrounding coastal flood risk projections: A case study for new orleans. *Earth's Future*, *5*, 9. https://doi.org/10.1002/2017EF000607

Yang, K., Paramygin, V., & Sheng, P. (2019). An objective and efficient method for estimating probabilistic coastal inundation hazards. *Journal of Natural Hazards*, *99*(2), 1105–1130 URL https://link.springer.com/article/10.1007/s11069-019-03807-w

Ye, K. Q., Li, W., & Sudjianto, A. (2000). Algorithmic construction of optimal symmetric latin hypercube designs. *Journal of Statistical Planning and Inference*, *90*(1), 145–159. https://doi.org/10.1016/S0378-3758(00)00105-1

Zhang, T., Geng, G., Liu, Y., & Chang, H. H. (2020). Application of Bayesian additive regression trees for estimating daily concentrations of ±2.5 components. *Atmosphere*, *11*(11), 1233. https://doi.org/10.3390/atmos11111233

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Hutchings, G., Sansó, B., Gattiker, J., Francom, D., & Pasqualini, D. (2023). Comparing emulation methods for a high-resolution storm surge model. *Environmetrics*, *34*(3), e2796. https://doi.org/10.1002/env.2796