

# Improvements in storm surge surrogate modeling for synthetic storm parameterization, node condition classification and implementation to small size databases

Aikaterini P. Kyprioti<sup>1</sup> · Alexandros A. Taflanidis<sup>1</sup>  Matthew Plumlee<sup>2</sup> · Taylor G. Asher<sup>3</sup>  Elaine Spiller<sup>4</sup> · Richard A. Luettich Jr<sup>5</sup> · Brian Blanton<sup>6</sup> · Tracy L. Kijewski-Correa<sup>7</sup> · Andrew Kennedy<sup>1</sup> · Lauren Schmied<sup>8</sup>

Received: 14 February 2021 / Accepted: 17 June 2021 / Published online: 14 July 2021

## Abstract

Surrogate models are becoming increasingly popular for storm surge predictions. Using existing databases of storm simulations, developed typically during regional flood studies, these models provide fast-to-compute, data-driven approximations quantifying the expected storm surge for any new storm (not included in the training database). This paper considers the development of such a surrogate model for Delaware Bay, using a database of 156 simulations driven by synthetic tropical cyclones and offering predictions for a grid that includes close to 300,000 computational nodes within the geographical domain of interest. Kriging (Gaussian Process regression) is adopted as the surrogate modeling technique, and various relevant advancements are established. The appropriate parameterization of the synthetic storm database is examined. For this, instead of the storm features at landfall, the features when the storm is at closest distance to some representative point of the domain of interest are investigated as an alternative parametrization, and are found to produce a better surrogate. For nodes that remained dry for some of the database storms, imputation of the surge using a weighted  $k$  nearest neighbor ( $k$ NN) interpolation is considered to fill in the missing data. The use of a secondary, classification surrogate model, combining logistic principal component analysis and Kriging, is examined to address instances for which the imputed surge leads to misclassification of the node condition. Finally, concerns related to overfitting for the surrogate model are discussed, stemming from the small size of the available database. These concerns extend to both the calibration of the surrogate model hyper-parameters, as well as to the validation approaches adopted. During this process, the benefits from the use of principal component analysis as a dimensionality reduction technique, and the appropriate transformation and scaling of the surge output are examined in detail.

**Keywords** Storm surge surrogate model · Kriging · Gaussian process · Storm parameterization · Overfitting · Binary classification · Dry node imputation

✉ Alexandros A. Taflanidis  
a.taflanidis@nd.edu

## Abbreviations

$n$	Total number of available synthetic storm simulations
$\cdot^h$	Characteristics pertaining to the $h$ th storm (superscript)
$\cdot_i$	Characteristics pertaining to the $i$ th node (subscript)
$\mathbf{z}$	Vector with peak surge over the geographic domain of interest
$n_z$	Total number of nodes considered (dimensionality of $\mathbf{z}$ )
$e_i$	Elevation of the $i$ th node
$\mathbf{x}$	Vector with parameters used to characterize storm input
$n_x$	Dimensionality of $\mathbf{x}$
$x_{lat}, x_{long}, \beta, \Delta P, R_{mw}, v_f$	Storm characteristics used as input
$z_i(\mathbf{x}^h) = z_i^h$	Peak surge for the $i$ th node location and the $h$ th storm
$\hat{z}_i^h$	Surge estimate for $z_i^h$ based on $k$ NN interpolation using neighboring nodes
$A_k^h[i]$	Set of $k$ closest nodes to the $i$ th node for the $h$ th storm
$d_{ij}$	Geo-distance between nodes $i$ and $j$
$w(d_{ij})$	Weight utilized for the weighted $k$ NN implementation
$A_{mc}$	Set of nodes that are misclassified as wet when it is known they are dry
$S_c, S_s$	Surrogate model for the classification of the node condition and surrogate model for surge prediction
$\mathbf{y}, n_y$	Output vector considered in the surrogate model development and its dimension
$\mathbf{X}$	Database considered in the surrogate model development
$\mathbf{X}_{-h}, \mathbf{X}_{-A_h}$	Database where the $h$ th storm is excluded, and database where the storm subset $A_h$ has been excluded
$A_h$	$k$ -fold storm subset including the $h$ th storm
$z^n$	Normalized surge by pressure deficit $\Delta P$
$z_i^t = g(z_i^n), c_i$	Transformed surge using function $g(\cdot)$ and constant to facilitate the transformation
$\mathbf{z}^{res}$	Surge residual after the subtraction of the surge corresponding to the retained principal components
$I_i(\mathbf{x})$	Classification of the $i$ th node condition (for storm input $\mathbf{x}$ ), corresponding to 1 if node is wet and to 0 if it is dry
$I_i^s(\mathbf{x} \mathbf{X}), I_i^c(\mathbf{x} \mathbf{X}), I_i^{cb}(\mathbf{x} \mathbf{X})$	Surrogate model predicted classification of the $i$ th node using the $S_s, S_c$ , or combined surrogate model formulations
$\tilde{\cdot}$	Surrogate model predictions
$\mathbf{Y}, \mathbf{F}, \mathbf{R}, \mathbf{B}_f, \mathbf{R}$	Matrix quantities for the surrogate model development
$\boldsymbol{\beta}^*, \boldsymbol{\alpha}^*, \mathbf{r}(\mathbf{x} \mathbf{X})$	Vector quantities for the surrogate model development
$\mathbf{f}(\mathbf{x}), n_b$	Basis function vector for surrogate model and its dimensionality
$\delta, \mathbf{I}_n$	Nugget parameter and identity matrix of dimension $n$
$R(\mathbf{x}^l, \mathbf{x}^m \mathbf{s})$	Correlation function for surrogate model
$\mathbf{s}$	Hyper-parameter vector for the surrogate model correlation function
$m_s$	Number of latent components retained in PCA process
$\mathbf{u}, u_j$	Vector of latent outputs and the notation for the $j$ th individual component
$\boldsymbol{\mu}$	Mean surge vector for all nodes across the different storm simulations

$\mathbf{P}, \mathbf{U}$	Projection matrix and matrix of latent components
$P(I_i(\mathbf{x}^h) = 1 \theta_i(\mathbf{x}^h))$	Bernoulli distribution approximation of probability of the $i$ th node being wet for the $h$ th storm
$\theta_i(\mathbf{x}^h)$	Natural parameter for the $i$ th node and the $h$ th storm of the logistic function
$n_p$	Dimensionality for the output vector that contains all the nodes that have remained dry in at least one of the storms within the database
$\boldsymbol{\theta}(\mathbf{x})$	Natural parameter vector for all nodes for storm input $\mathbf{x}$
$\mathbf{t}$	Vector of latent logistic principal components
$m_c$	Number of principal components retained in LPCA process
$\boldsymbol{\Theta}, \mathbf{T}, \mathbf{V}, \boldsymbol{\Delta}, \boldsymbol{\Delta}_\theta$	Quantities for the LPCA development
$\mathbf{p}^{res}$	Classification residual vector after the $S_c$ surrogate model development
$\tilde{p}_i^c(\mathbf{x} \mathbf{X})$	Probability of the $i$ th node being wet for storm input $\mathbf{x}$ given by surrogate model $S_c$
$MC_i^h, +MC_i^h, -MC_i^h$	Misclassification for the $i$ th node and the $h$ th storm, and false positive (node predicted wet when dry) or false negative (node predicted dry when wet) counterparts
$\overline{MC}, MC^h, MC_i$	Total misclassification across all nodes and storms, total misclassification across all nodes for the $h$ th storm, total misclassification across all storms for the $i$ th node
$NRMSE_i, \overline{NRMSE}$	Normalized root mean squared error for the $i$ th node across all storms, and its average value across all nodes
$NRMSE^h$	Normalized root mean squared error for the $h$ th storm
$SC_i^h, SC^h, SC_i, \overline{SC}$	Surge score for the $i$ th node and the $h$ th storm, and average values, respectively, across nodes (for specific storm), storms (for specific node) and both storms and nodes.
$JPM$	Input parametrization according to Joint Probability Method
$CARP$	Input parametrization with respect to the representative point

## 1 Introduction

Surrogate models (also referenced as emulators or metamodels) have emerged as a versatile technique for predicting storm surge caused by tropical cyclones (Jia et al. 2016; Bass and Bedient 2018; Zhang et al. 2018; Al Kajbaf and Bensi 2020; Contento et al. 2020; Kyriotti et al. 2020; Plumlee et al. 2021). These models correspond to data-driven approximations, developed using a database of surge predictions for a properly selected suite of storms. Their objective is to describe the input/output relationship of the expensive, high-fidelity hydrodynamic model used to develop the storm surge database. Once trained, the surrogate model can substitute for the high fidelity simulation model, maintaining, similar prediction accuracy at a dramatically lower computational cost. Different types of emulators have been examined in this context, ranging from surge response functions (Irish et al. 2009) and response surfaces (Taflanidis et al. 2012) to neural networks (Kim et al. 2015) and kriging (Jia and Taflanidis 2013). The flexibility of these emulators has been demonstrated in a number of implementations over different regions of interest (Irish et al. 2009; Jia et al. 2016; Al Kajbaf and Bensi 2020), for peak surge or surge time-series predictions (Jia et al.

2016), and even for addressing sea level rise implications (Contento et al. 2020; Kyprioti et al. 2020). Extensions have been recently considered for tide-, river- and atmosphere-driven water levels as well (Parker et al. 2019). The versatility of surrogate models, along with the associated computational efficiency of accommodating multiple predictions in a short amount of time with a minimum requirement on computational resources, has also promoted them as effective tools to support comprehensive regional flood risk assessment or emergency response management during landfalling hurricanes (Kijewski-Correa et al. 2020; Nadal-Caraballo et al. 2020; Plumlee et al. 2021).

In the majority of the aforementioned studies, especially the ones that adopted kriging as emulator, the number of surge simulations informing the surrogate model development was moderately large, whereas the surge predictions were commonly established for a small subset of save point locations within the geographical region of interest. Moreover, most of these studies focused on save point locations that were always wet across the storm simulations. Finally in those past studies, limited emphasis was placed on the appropriate parameterization of the storm features that serve as input for the surrogate model. This study investigates the development of a surrogate model for the peak storm surge considering the regional area around Delaware Bay, using a database of simulations established recently for a FEMA coastal flood insurance study (Blanton et al. 2011; Hanson et al. 2013; Vickery et al. 2013), and offers critical improvements on the aforementioned four topics. The database consists of 156 synthetic storms, and predictions extend across all nodes (approximately 300,000) of the underlying numerical model (computational grid nodes) within the geographical domain of interest.

Kriging, also referenced as Gaussian Process regression, is adopted as the surrogate model of choice here. Instead of using as emulator input, the storm features at landfall, which has been the common implementation so far (Jia et al. 2016; Al Kajbaf and Bensi 2020), the use of the storm features when the storm track is at closest distance to the domain of interest is investigated. For nodes that have remained dry in some of the storms simulations, an imputation process is considered to estimate the so-called pseudosurge (Shisler and Johnson 2020). In order to achieve computational efficiency in such a setting, a  $k$  nearest neighbor ( $k$ NN) (Dudani 1976) interpolation with distance-dependent weights is adopted here (making this approach similar to a kernel smoothing), with the interpolation characteristics calibrated using a cross-validation setting. This imputation may provide pseudosurge estimates that misclassify the node condition (node predicted wet when it is known to be dry). Instead of forcing the pseudosurge estimates to directly accommodate the correct classification (Taflanidis et al. 2013), and in an effort to facilitate higher accuracy predictions for such nodes, a secondary surrogate model is considered to classify the node condition (wet or dry). This secondary surrogate model couples logistic principal component analysis (LPCA) (Schein et al. 2003) with a kriging emulator on the resultant natural parameters of the logistic process. The optimal coupling between the storm surge surrogate model and the surrogate model for the node condition classification is investigated in detail. Finally, challenges associated with the potential overfitting of the surrogate model due to the small size of the available database are examined. The focus is on the impact of dimensionality reduction tools on overfitting phenomena when the training database is small, and on the development of mitigation techniques in such cases. This constitutes one of the core advances offered in this article. Though dimensionality reduction tools, like principal component analysis (PCA) have been previously considered for storm surge surrogate models (Jia and Taflanidis 2013; Kyprioti et al. 2020), the pitfalls of overfitting have not been explored in detail previously. To address this challenge, for both PCA and LPCA, the use of a small number of principal components, coupled with a

supplementary surrogate model for the residuals of the prediction estimates, is examined. Two other issues related to the small size of the database are also discussed: the proper implementation of cross-validation in order to estimate the surge surrogate model accuracy, and the formulation of the hyper-parameter calibration.

The remainder of the paper is organized as follows. In the next section, an overview of the database used in this study is presented and the alternative storm parametrization is introduced. Section 3 describes the  $k$ NN imputation process and reveals the challenges associated with the classification of problematic nodes. Section 4 discusses the surrogate model development, distinguishing between the storm surge surrogate model, the surrogate model for the node classification, and the overall coupled implementation. Section 5 presents in detail the case study application and reveals critical trends related to the advances offered in Sects. 2, 3, 4.

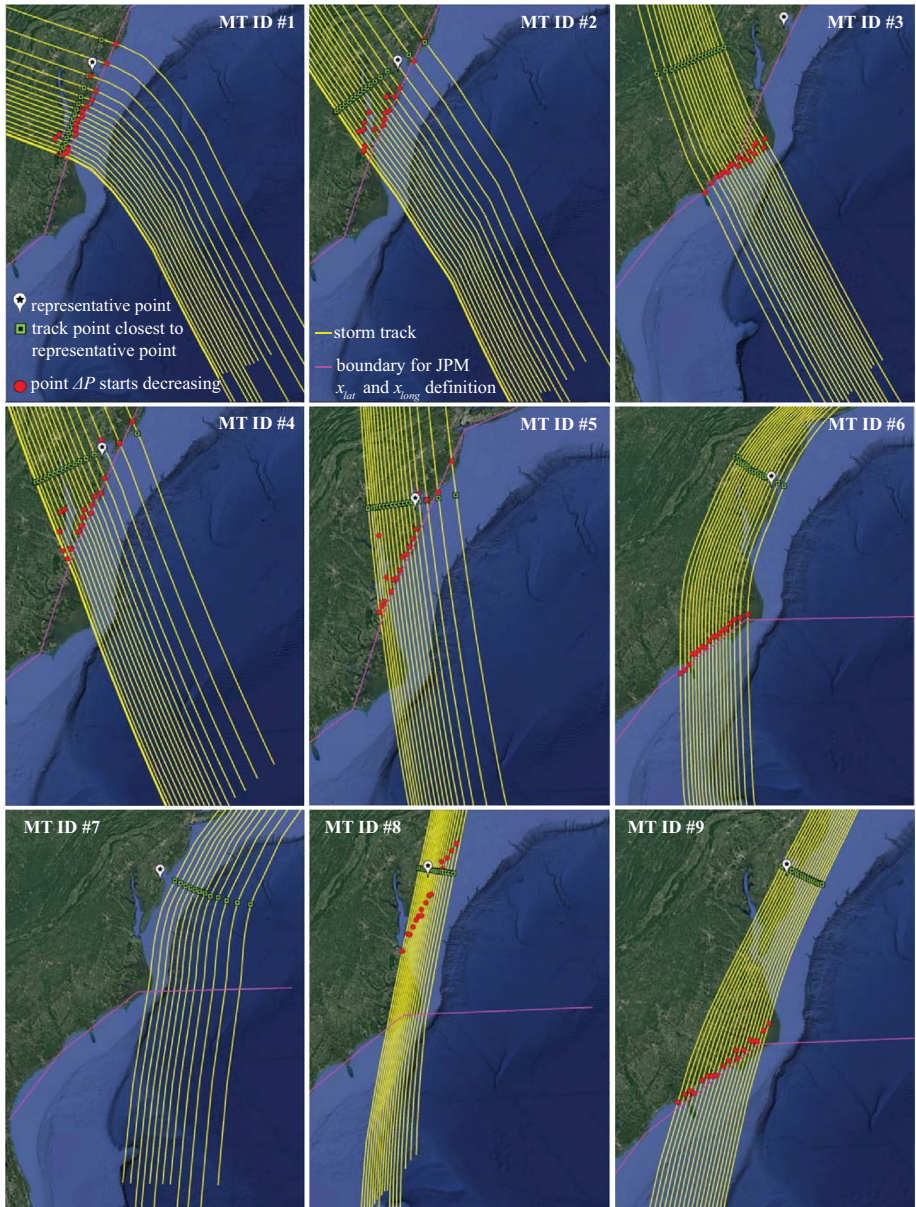
## 2 Database overview and storm input parametrization

### 2.1 FEMA Region 3 database

The synthetic storm surge simulations and associated storm parameters developed as part of the FEMA Region 3 Coastal Storm Surge Study (Hanson et al. 2013) are utilized as the training database for the surrogate model development, and in this section a brief description of the storms and their statistics are presented. Interested readers can find more details regarding the database in (Vickery et al. 2013). The storm population was defined based on regional hurricane characteristics using a joint probability method (JPM) optimal sampling (JPM-OS) approach, though the JPM-OS methodology adopted differed notably from other variants of the same name published in literature [e.g., (Toro et al. 2010)]. Three different sets of storms were considered: (1) storms making landfall in Virginia, Delaware and New Jersey, (2) storms making landfall in North Carolina, and (3) bypassing storms. Storm features are summarized in Table 1. The storm simulation population corresponds to nine different master tracks (MTs), each characterized by a different overall storm path and landfalling heading  $\beta$ . The total resulting 156 storms assigned to the nine different MTs, along with other details discussed later, are shown in Fig. 1. For the bypassing storms,

**Table 1** Summary details for the master tracks of FEMA Region 3 storm database

Master track classification	Number of storms	$\beta$ ( $^{\circ}$ )	$\Delta P$ (mbar)	$v_f$ (m/s)	Master Track (MT) ID#
Landfalling in Virginia, Delaware and New Jersey	18	-75	34, 51, 65	4.0, 7.2	1
	18	-45		4.7, 9.5	2
	18	-30		4.7, 9.5	4
	18	-10		4.8, 10.5	5
	18	15		5.0, 13	8
Landfalling in North Carolina	18	-35	38, 56, 75	4.6, 11.5	3
	18	0			6
	18	22			9
Bypassing	12	12	42, 67	4.0, 11.4	7



**Fig. 1** Tracks for all 156 synthetic storms in the database, distinguished into the nine different MTs, along with (a) reference point (white location pin) and the landfalling points (green rectangular) based on *CARP* parameterization, (b) linearized boundary (magenta lines) used for *JPM*  $x_{lar}/x_{long}$  selection (different for the landfalling and bypassing storms) and (c) for landfalling storms the point  $\Delta P$  starts decreasing (indicating significant reduction for  $\Delta P$ )

the heading  $\beta$  is defined when the storm approaches latitude  $35^\circ$  N. The MT identification number, assigned in ascending order based on  $\beta$ , is also shown in Table 1. For each MT, different values for the forward speed  $v_f$  and the central pressure  $\Delta P$  were considered at

landfall based on JPM-OS, and both these characteristics varied along each storm track (Vickery et al. 2013). The radius of maximum winds  $R_{mw}$  and *Holland-B* were determined based on the track definition and  $\Delta P$  characteristics using regression analysis results from Vickery and Wadhera (2008). *Holland-B* was restricted to its mean value only, and so it does not need to be considered as an additional input parameter for the surrogate model development (it is uniquely defined based on the remaining storm features), whereas three different values of  $R_{mw}$ , corresponding to mean and mean  $\pm$  one standard deviation from the corresponding regression, were considered. Each combination of  $\Delta P$ ,  $v_f$  and  $R_{mw}$  was assigned randomly to a unique landfalling/bypassing location, producing the total 156 storm tracks. The storm parameter values considered for each MT, resulting from the combination of three values of  $R_{mw}$ , two values of  $v_f$  and three values (for landfalling) or two values (for bypassing) of  $\Delta P$  at reference locations, are also provided in Table 1. Each storm track was used to compute the storm surge and wind wave response on a high-resolution ADCIRC (Luettich et al. 1992) grid for the region, having approximately 1.5 million computational nodes in the Region 3 area. Details for the computational model are included in (Blanton et al. 2011). A subset of the entire domain will be considered for the metamodel development, focusing on areas around Delaware Bay, constrained by latitude  $[38^\circ, 40^\circ]$  N and longitude  $[72^\circ, 75.7^\circ]$  W. Estimates are restricted to the peak-surge only.

Storm parameters are not constant along a storm's track. The same is therefore true for the storm forcing in the hydrodynamic model simulations. The parameters presented in Table 1 are those at some particular point (typically near landfall). Pre-landfall parameter variations are relatively small and gradual, whereas post-landfall changes in  $\Delta P$  (and  $R_{mw}$ ) are large and abrupt. This is consistent with basic tropical cyclone physics, which states that a storm will weaken and widen when it moves inland where the warm, moist air it feeds upon is cutoff. The coastal storm surge study (Hanson et al. 2013) adopted the post-landfall filling model of (Vickery 2005), corresponding to a generalized exponential decay model for central pressure, to define the post-landfall changes in  $\Delta P$  (and  $R_{mw}$ ). This storm behavior is highly important in this study because Delaware Bay is further to the north, and the local peak surges for many storms are likely to occur post-landfall. This means that the JPM parameter values may not be the most representative values defining the peak surge response. This issue is studied further in the next section.

## 2.2 Storm input parametrization

The surrogate model development requires an efficient parameterization of the synthetic storm database, to serve as the emulator's input. This parameterization includes features of storm intensity ( $\Delta P$ ), size ( $R_{mw}$ ) and translational speed ( $v_f$ ), as well as features for the storm track description corresponding to the storm heading ( $\beta$ ), and the latitude and longitude of the track ( $x_{lat}$ ,  $x_{long}$ ). As described in the previous section, storm parameters (and the corresponding water level response) vary in time, meaning that some reference features need to be chosen. The common implementation when developing a surrogate model (Resio et al. 2009; Kim et al. 2015; Zhang et al. 2018), is to utilize the characteristics at landfall, or at the crossing of a reference line for bypassing storms (reference landfall) (Nadal-Caraballo et al. 2015). For this study, this is equivalent to using the JPM optimal sampling parameters. But, given that landfall is far from the study area for many storms, this seems suboptimal. Typically, peak surge at a given location should occur when the shore-normal wind speed is the greatest, though this notion is confounded by the complex coastal geometry. A location undergoes peak wind speed when the storm is nearby (within

tens of kilometers), though this is further complicated by the rapid weakening post-landfall and the variability in wind direction. Defining this relationship exactly is difficult for any arbitrary storm. So instead, the storm characteristics at the moment (time stamp) the storm track is closest to a representative point (CARP) centered in the study area are adopted as the set of reference storm parameters. Both parameterizations for the storm input will be investigated and referenced herein as (a) *JPM* input and (b) *CARP* input.

For the *JPM* input parameterization, the heading  $\beta$ , intensity  $\Delta P$ , and speed  $v_f$  characteristics are the ones reported in Table 1, whereas the size  $R_{mw}$  is estimated using (Vickery and Wadhera 2008). For the reference landfall,  $x_{lat}$  and  $x_{long}$ , different approaches are taken depending on the storm heading when the storm track crosses latitude  $38^\circ\text{N}$ . If that heading is less than  $0^\circ$ , a simplified (piece-wise linear) US coast boundary is utilized to define landfall. The boundary simplification is chosen to avoid any ambiguous definition of landfall due to the existence of bays. For storms with heading greater than  $0^\circ$  when the storm track crosses latitude  $38^\circ\text{N}$ , a boundary consisting of two segments is utilized to define the reference landfall: this boundary follows the previous linearized US coast up to latitude  $35.2^\circ\text{N}$  and then extends horizontally across  $35.2^\circ$ . Both these boundaries are shown in Fig. 1. The distinction of storm groups based on heading when a storm crosses  $38^\circ$ , instead of the original MT classification shown in Table 1, guarantees that similar storms are represented by similar parameterization with respect to reference landfall. Note that similar criteria for the distinction of North Atlantic storms based on heading around  $38^\circ\text{N}$  has been adopted in another recent flood study (Nadal-Caraballo et al. 2015).

For the *CARP* parameterization, the geographic domain of interest, that of the Delaware bay, is represented by a point with coordinates  $39^\circ\text{N}$  and  $75.25^\circ\text{W}$ . As discussed above, the reference point for each storm is taken to correspond to the instance when the storm track is at closest distance to this representative point. This leads to the definition of  $x_{lat}$  and  $x_{long}$  inputs. The storm heading and intensity/size/speed parameters are calculated using a time window of 4 hrs before and after the reference point. For  $\Delta P$  and  $R_{mw}$ , the maximum value over this time-window is adopted, while for  $v_f$  and  $\beta$  the average value across the time-window is utilized. It should be noted that the exact choice for the representative point of the domain was determined to be of small importance as long as it lies inside the geographic domain of interest.

Figure 1 provides some details for the reference point locations for the two different parameterizations. For the *CARP* parameterization the points that are closest to the representative point of the geographic domain and the representative point itself are shown. For the *JPM* parameterization the linearized boundary used for the  $x_{lat}$ ,  $x_{long}$  definition is also shown in the figure and, for landfalling storms, the point along the storm track that the post-landfall filling starts (indicating a significant drop in  $\Delta P$ ) is marked. Note that the latter point is close (but not necessarily identical) to the farthest point along the storm track that the  $\Delta P$  *JPM*-defined parameter value is attained. Comparisons indicate that the alternative storm parameterization considered here leads to fundamentally different characteristics for the storms making landfall in North Carolina, for which not only is the reference point location significantly impacted, but also the strength of the storms is drastically reduced, since the reference point corresponds to a time substantially after the time of conventional landfall. Bypassing storms also have significant differences in terms of reference location, but their strength changes slightly along their track since these storms do not undergo post-landfall filling.



### 3 Dry node imputation

Inland nodes that have remained dry for some of the synthetic storms in the database provide incomplete information for the surrogate model development: for some storms the only information available is the fact that the node has remained dry, and for some other storms, when the node is wet, the exact surge is known. If the goal was to predict only the node condition, corresponding to a binary wet/dry classification, such incomplete information would not pose any challenge, but for predicting the actual surge value, an adjustment of the database is needed. This ultimately defines an imputation process for the missing data, corresponding to predictions for the dry nodes, with values inferred from the remaining database (Taflanidis et al. 2013; Shisler and Johnson 2020), and facilitates the development of a single surrogate model for the entire set of nodes (Jia et al. 2016). The term pseudosurge (Shisler and Johnson 2020) will be also used herein to distinguish the imputed surge value.

One potential imputation strategy is to estimate the pseudosurge for each node by explicitly optimizing the prediction accuracy of the surrogate model that is eventually established using the imputed surge values (Shisler and Johnson 2020). This approach may encounter challenges from overfitting the available data, especially for smaller size databases like the one examined here. It also has a substantial computational burden, since it needs to be separately applied to each of the nodes that need to be imputed, in a setting that usually a good portion of the database nodes (> 30%) has remained dry for some storms. An alternative strategy is to use geospatial interpolation for each storm, utilizing information from the wet nodes to infer the values for the neighboring dry ones (Taflanidis et al. 2013). Various implementations (Furrer et al. 2006; Cressie and Johannesson 2008) can be adopted in this geo-interpolation to accommodate the large dimensionality of the data (number of nodes in the database). Here a local interpolation is advocated to address this challenge, since it can additionally enforce the underlying hydraulic connectivity between the nodes: a node can provide useful information for a neighboring one, only if they both are connected within the ADCIRC grid. Here, a weighted  $k$  nearest neighbor interpolation ( $k$ NN) is specifically chosen.

#### 3.1 Weighted $k$ NN formulation

Let  $z_i^h$  denote the surge for the  $i$ th node and the  $h$ th storm, and  $\hat{z}_i^h$  its estimate based on neighboring nodes. The weighted  $k$ NN interpolation is expressed as:

$$\hat{z}_i^h = \frac{\sum_{j \in A_k^h[i]} w(d_{ij}) z_j^h}{\sum_{j \in A_k^h[i]} w(d_{ij})} \quad (1)$$

$$w(d_{ij}) = \begin{cases} e^{-\left(\frac{d_{ij}}{q}\right)^p} & \text{if } d_{ij} < d \\ 0 & \text{if } d_{ij} \geq d \end{cases}$$

where  $A_k^h[i]$  defines the set of  $k$  closest nodes to the  $i$ th node for the  $h$ th storm,  $d_{ij}$  is the geodistance between nodes  $i$  and  $j$ , and  $w(d_{ij})$  is a distance-dependent weight taken as a power exponential expression with parameters  $d$ ,  $q$ ,  $p$ . Only nodes with known surge values are included in set  $A_k^h[i]$ ; these may correspond to inundated nodes for the  $h$ th storm or nodes with already imputed values within the iterative formulation discussed next. A cutoff distance  $d$  is introduced in the definition of weights  $w(\cdot)$  to avoid far-away nodes influencing

the  $k$ NN interpolation, used to accommodate any irregular parts of the grid. For such parts of the grid, where distances between nodes might be large, some of the  $k$  nearest neighbors can be far away from node  $i$ , and in those cases the use of  $d$  prevents such neighbors from impacting the interpolated surge for node  $i$ . It should be pointed out that the formulation of Eq. (1) can be viewed as a kernel smoothing implementation with  $w(d_{ij})$  corresponding to the chosen kernel function. The set of  $[k d q p]$  parameters of Eq. (1) correspond to the hyper-parameters of the weighted  $k$ NN interpolation that need to be calibrated. This calibration is performed using information for the wet-nodes as discussed in “Appendix A”.

### 3.2 Iterative surge imputation using $k$ NN

Once the calibration is performed, the  $k$ NN interpolation scheme can be applied to impute the database for the dry nodes. As discussed earlier, accommodating hydraulic connectivity is very important in this imputation process. Unfortunately, this connectivity can be very localized and irregular due to coastal geometry. It is therefore difficult conceptually and computationally to perform efficiently the imputation at a very large number of points while taking this into consideration. Using relatively a few nearby points is desirable, yet hundreds or thousands of points may need to be imputed, which may not have any nearby wet points. Therefore, the  $k$ NN interpolation is done iteratively. At each iteration, imputation is done only on dry nodes that have at least  $k$  wet (imputed and genuine) neighbors, in a larger set of  $k_c$  nodes. If fewer than  $k$  wet neighbors are available, no value is imputed in the current iteration; in other words, dry nodes whose closest  $k_c$  neighbors are mostly dry (do not include at least  $k$  wet nodes) are left unadjusted in this step. This facilitates a gradual spatial imputation, promoting local connectivity. The value of  $k_c$  was chosen to be twice the value of  $k$  in this study.

Computational efficiency for both the  $k$ NN calibration, discussed in the previous section, and for the iterative  $k$ NN-based imputation for the dry nodes in each storm, is established by pre-calculating the distances between each node and its  $k_{max}$  closest neighbors. Value of  $k_{max}$  needs to be larger than the  $k_c$  value ultimately utilized, and since this is not a priori known, the use of an arbitrarily large value (for example 50) is suggested.

### 3.3 Correction of imputed surge values

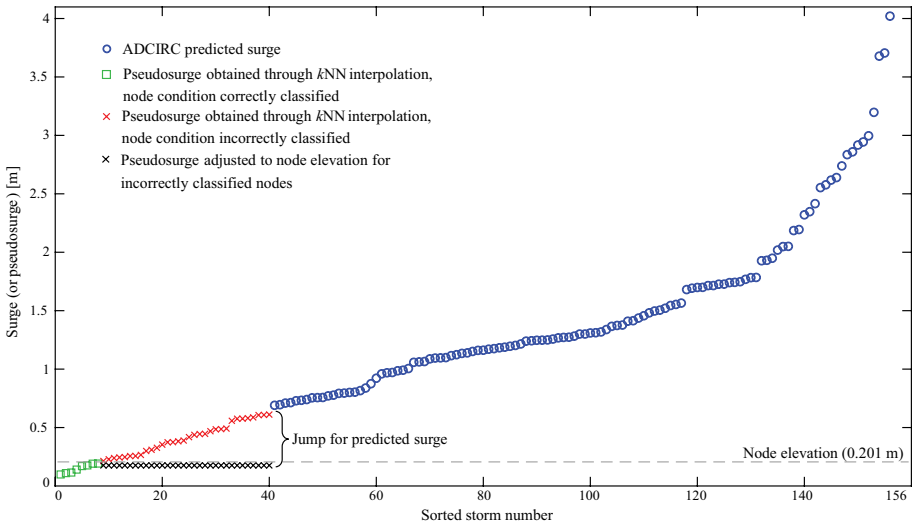
The imputation process may lead to the misclassification of some nodes, with imputed surge (pseudosurge)  $z_i^h$  projected higher than the node elevation (ideally all the dry nodes should be filled with surge values that are below their elevation so that they maintain their identity as “dry”). This ultimately yields a false classification of the node as wet based on the imputed surge. Two alternative paths are considered for addressing such misclassification instances:

- (1) Provide no correction for the misclassified nodes and address any false classification information later in the development of the surrogate model for the surge predictions (a remedy to directly address this will be discussed later). The corresponding database for such an implementation will be denoted herein as “*pseudo-s*”. The set of nodes for which the *pseudo-s* database includes erroneous information for their classification,

meaning classified as wet based on the pseudosurge estimate when known to be dry, will be denoted as  $A_{mc}$ .

- (2) Implement a final correction step for the misclassified nodes: if  $\frac{z_i^h}{z_i}$  is estimated to be larger than the node elevation, the pseudosurge predictions are modified to correspond to the node elevation minus a threshold of 0.05 m. This adjusts the imputed value to provide the correct characterization of the node condition. The corresponding database that has sustained such a correction will be denoted herein as “*corrected pseudosurge*”.

Figure 2 illustrates the need of correcting the imputed surge, focusing here on a specific node. The surge is plotted in ascending order across the 156 storms and the node elevation (with a horizontal line) is also shown. The depicted blue circles correspond to surge predictions for the original database (node is wet), while the green squares and the red  $\times$  correspond to the  $k$ NN-based imputed surge for the storms for which the node was originally dry. Comparing these values to the node elevation, the following distinction can be made: the green squares correspond to correct classification where the imputed surge lies below the node elevation and therefore node is still predicted as dry, while the red  $\times$  to an erroneous classification: imputed surge is above the node elevation and therefore the node is predicted as wet. For the “*corrected pseudosurge*” database approach, the erroneous classified values corresponding to red  $\times$  would have been mapped below the node elevation, as shown in the figure with black  $\times$ . This evidently creates an abrupt “jump” in the *corrected pseudosurge* database for this specific node, which is expected to provide challenges later in the surrogate model development. This is the main reason that the *pseudo-s* database with no correction is also considered: though this database includes erroneous information for the nodes in set  $A_{mc}$  (red  $\times$  points), its use is expected to support a higher accuracy surrogate model, since for each node there exists continuous surge information. It should be pointed out that the case shown in Fig. 2 is carefully selected and it is an extreme one, chosen to



**Fig. 2** Ordered surge and pseudosurge values for a specific node (its elevation shown as a dashed grey line), with distinction between the correctly and incorrectly classified node condition cases. Adjustment of the incorrectly classified cases shows the jump created for the predicted surge if the *corrected pseudosurge* database is utilized

demonstrate the potential challenges associated with the pseudosurge correction. This large apparent jump should be also partially attributed to the small number of available storms in the investigated database.

## 4 Advances in surrogate model development

This section offers various advances in surge surrogate modeling, focusing mostly on two specific issues: (1) the classification of the wet/dry condition for nodes for which the pseudosurge database includes erroneous information, and (2) the various challenges that arise and are related to data overfitting. Finally, some additional advances are examined regarding functional transformations of the surge to accommodate higher surrogate model accuracy.

### 4.1 Surrogate modeling problem formulation

Let  $\mathbf{x}^h \in \mathbb{R}^{n_x}$  be the  $n_x$ -dimensional vector describing the input for the  $h$ th storm, established according to the storm parameterization approach discussed in Sect. 2.2, and  $n$  be the total number of storms in the available database. This database provides for each of the  $n$  storms the  $n_z$  dimensional surge vector  $\mathbf{z}^h \in \mathbb{R}^{n_z}$ , whose  $i$ th component  $z_i^h$  corresponds to the peak surge for the  $i$ th node and the  $h$ th storm. For nodes that have remained dry in the original database, the imputed surge (pseudosurge), calculated as discussed in Sect. 3, is utilized. The notation  $\mathbf{z}(\mathbf{x})$  will be also used to denote explicitly the relationship between input and output, with the surge for the  $i$ th node and the  $h$ th storm represented as  $z_i(\mathbf{x}^h) = z_i^h$ . The classification of the  $i$ th node condition for the  $h$ th storm is denoted by  $I_i(\mathbf{x}^h)$ , with  $I_i(\mathbf{x}^h) = 1$  corresponding to wet and  $I_i(\mathbf{x}^h) = 0$  to dry.

Two different surrogate models will be considered, one for the prediction of the surge  $z_i(\mathbf{x}^h)$  and one for the prediction of the node classification  $I_i(\mathbf{x}^h)$ . These will be referenced herein as  $S_s$  and  $S_c$ , respectively. The consideration of a separate model for the classification is necessary to accommodate the use of the *pseudo-s* database (without correction) and intends to address the erroneous information included in that database for set  $A_{mc}$ . This can ultimately facilitate greater accuracy for the metamodel used to predict the storm surge, since it restricts its scope: its only objective is the correct prediction of the pseudosurge and not necessarily the additional classification of the condition of each node (wet/dry). As discussed earlier, kriging is considered as the preferred surrogate modeling technique. Both metamodels  $S_s$  and  $S_c$  use  $\mathbf{x}$  as input, and try to predict the respective outputs. “Appendix B” reviews the essential characteristics for the metamodel development. The details and advancements for each of the two metamodels are separately discussed in the next two sections, while the integrated formulation is reviewed in Sect. 4.4. In the same section a schematic of the overall implementation is presented. Validation of the surrogate model formulation is examined in Sect. 4.5. The notation introduced in “Appendix B” is adopted herein, with lower case variables denoting characteristics for specific storms and upper case variables referring to characteristics across the entire database. For example,  $\mathbf{x}$  denotes the input vector for a specific storm, and  $\mathbf{X}$  the matrix having as columns the input for each of the  $n$  storms.

To accommodate the surrogate model implementation, some form of dimensionality reduction will be adopted for both  $S_s$  and  $S_c$ . Both this dimensionality reduction of the output and the surrogate model calibration encounter challenges related to overfitting as

it will be shown later on, since both of these tasks incorporate some form of explicit or implicit optimal selection of features (training) within the existing dataset. This overfitting can occur when the associated optimization fits too closely or exactly to that particular dataset, failing to predict reliably new data if they differ from the existing observations. This can be equivalently considered as extracting too much information from the existing dataset or developing models that have too many parameters, more than justified by the available data. For the database considered here, the danger of overfitting is significant since the number of storms in the database is relatively small (only  $n = 156$  storms), while the number of predicted outputs is large ( $n_z = 297,460$ ). This potential overfitting guides many of the advances discussed in the next two sections.

## 4.2 Surrogate model for surge predictions

### 4.2.1 Transformation of surge

The surrogate model for the surge predictions considers as input the imputed surge  $z_i(\mathbf{x}^h)$ . To improve the metamodel accuracy two different transformations are considered. The first one (Al Kajbaf and Bensi 2020) is a scaling by the central pressure, a transformation motivated by the physics of storm surge (Irish et al. 2009). This scaling is implemented here with respect to the mean sea level, as advocated in (Kyprioti et al. 2020), leading to the normalized surge:

$$z_i^n(\mathbf{x}^h) = \frac{z_i(\mathbf{x}^h) - s_i}{\Delta P^h} \quad (2)$$

where  $\Delta P^h$  is the central pressure deficit for the  $h$ th storm and  $s_i$  is any steric adjustment used in the coastal storm surge study. The second transformation is purely a functional one, and has as an objective to narrow the output variation across the database. Different invertible functions may be considered for this transformation leading to the transformed surge:

$$z_i^t = g(z_i^n) \quad (3)$$

with  $g$  representing the adopted function. Candidate choices for  $g(\cdot)$  include  $g(z_i^n) = \sqrt{z_i^n + c_i}$  or  $g(z_i^n) = \log(z_i^n + c_i)$ , denoted herein respectively as “ $g = \text{sqrt}$ ” or “ $g = \text{log}$ ”. Constant  $c_i$  is chosen equal to the minimum of  $z_i^n$  over the storm database, but not greater than 0, and is utilized to make the corresponding arguments positive across all storms.

### 4.2.2 Surge output dimensionality reduction and surrogate model formulation

In order to accommodate the large dimension of the database two alternative approaches can be adopted: the first is to calibrate one surrogate model to offer predictions across the entire database, corresponding to a parallel emulator implementation (Gu and Berger 2016), while the second one is to adopt principal component analysis (PCA) as a dimensionality reduction technique (Jia and Taflanidis 2013) and consider separate surrogate models for the individual latent components. Both formulations are examined in this study, and a new one is introduced.

PCA identifies a smaller number ( $m_s < n < n_z$ ) of latent outputs  $\mathbf{u} \in \mathbb{R}^{m_s}$  through a linear projection, with the exact selection of  $m_s$  based on the percentage (%) of the

variability of the original database that is desired to be explained by the retained components (Jolliffe 2002). Characteristics for the individual components will be denoted by subscript  $j$  herein, with  $u_j$  denoting the  $j$ th element of vector  $\mathbf{u}$ . Separate surrogate models can be then considered for each of the individual principal components  $\{u_j; j = 1, \dots, m_s\}$ , or for groups of them. This calibration of separate surrogate models for each of the latent outputs may offer increased overall accuracy compared to a parallel emulator implementation. Unfortunately, though, this combination of PCA and surrogate modeling for the selected principal components carries the potential of overfitting the available data. Typically, the first few principal components can be predicted well by the corresponding surrogate model, but the accuracy for higher components drastically reduces. In previous storm surge studies, this pattern has been manifested as a saturation of the surrogate model accuracy as the number of principal components increases (Jia and Taflanidis 2013). This ultimately means that the predictive capabilities of the surrogate model for higher principal components is lower. If such components become important for explaining unobserved data (when the metamodel is asked to make predictions for a new, unseen, storms), then the accuracy of the overall implementation (PCA and surrogate modeling) may decrease.

To address this challenge we introduce the following remedy: we consider only a small number of principal components  $m_s$ , and then complement these predictions with a surrogate model for the residuals. The reasoning is the following: the smaller number of significant latent components will still be accurately approximated by individual surrogate models (as it was before), while the surge residuals will be explained more accurately by a separate surrogate that will focus specifically on the entire residual, rather than the underlying components contributing to that residual.

The overall formulation is implemented through the following steps:

- (a) Perform PCA for the transformed output  $\{\mathbf{z}^t(\mathbf{x}^h); h = 1, \dots, n\}$  of the database and retain  $m_s$  principal components (Jia and Taflanidis 2013). Obtain the mean vector  $\boldsymbol{\mu} \in \mathbb{R}^{n_z}$ , whose  $i$ th component corresponds to the mean of  $\{z_i^t(\mathbf{x}^h); h = 1, \dots, n\}$ , the projection matrix  $\mathbf{P} \in \mathbb{R}^{n_z \times m_s}$ , and for each latent component,  $u_j$ , the PCA coefficients, corresponding to the output vector of responses over the database  $\mathbf{U}_j(\mathbf{X}) = [u_j(\mathbf{x}^1) \dots u_j(\mathbf{x}^n)]^T \in \mathbb{R}^n$  for the storm surge surrogate model.
- (b) Develop  $m_s$  separate surrogate models based on the procedure described in “Appendix B” for each of the principal components, setting  $\mathbf{Y}(\mathbf{X}) = \mathbf{U}_j(\mathbf{X})$  and  $n_y = 1$ . Note that instead of individual surrogate models for each component, a grouping of the components may be considered as an alternative implementation to facilitate higher computational efficiency.
- (c) Calculate the residual for the transformed output for each storm:

$$\mathbf{z}^{res}(\mathbf{x}^h | \mathbf{X}) = \mathbf{z}^t(\mathbf{x}^h) - \boldsymbol{\mu} - \mathbf{P}\tilde{\mathbf{u}}(\mathbf{x}^h | \mathbf{X}) \quad (4)$$

where  $\tilde{\mathbf{u}}(\mathbf{x}^h | \mathbf{X})$  is the predicted latent output vector, with  $j$ th component provided by Eq. (30) for the respective surrogate model ( $m_s$  such surrogate models exist).

- (d) Develop an additional surrogate model for the surge residual  $\mathbf{z}^{res}$  [calculated according to Eq. (4)] based on the procedure described in “Appendix B”, setting  $\mathbf{Y} = [\mathbf{z}^{res}(\mathbf{x}^1 | \mathbf{X}) \dots \mathbf{z}^{res}(\mathbf{x}^n | \mathbf{X})]^T \in \mathbb{R}^{n \times n_z}$  and  $n_y = n_z$ . This ultimately corresponds to a parallel emulator implementation on the surge residual (Gu and Berger 2016).

Selection of value of  $m_s$  should be based on a parametric sensitivity analysis, examining the metamodel accuracy for an increasing  $m_s$  value. This can be performed using the  $k$ -fold cross-validation setting discussed in Sect. 4.5.

### 4.2.3 Surge surrogate model predictions

The surrogate model predictions  $\tilde{\mathbf{z}}^t(\mathbf{x}|\mathbf{X})$  for a new storm input  $\mathbf{x}$  are obtained by combining the predictions  $\tilde{\mathbf{u}}(\mathbf{x}|\mathbf{X})$  for each of the  $m_s$  latent outputs and the predictions for the residual  $\tilde{\mathbf{z}}^{res}(\mathbf{x}|\mathbf{X})$ , all of them are obtained according to Eq. (30) for the corresponding, in each case, surrogate model formulation. This leads to:

$$\tilde{\mathbf{z}}^t(\mathbf{x}|\mathbf{X}) = \tilde{\mathbf{z}}^{res}(\mathbf{x}|\mathbf{X}) + \boldsymbol{\mu} + \mathbf{P}\tilde{\mathbf{u}}(\mathbf{x}|\mathbf{X}) \quad (5)$$

The inverse transformations of Eqs. (3) and (2), are applied to obtain the median predictions for  $\mathbf{z}$ , leading to:

$$\tilde{z}_i(\mathbf{x}|\mathbf{X}) = \Delta P^h \left[ g^{-1}(\tilde{z}_i^t(\mathbf{x}|\mathbf{X})) \right] + s_t \quad (6)$$

Note that as discussed in ‘‘Appendix B’’, a functional dependence on the database  $\mathbf{X}$  is utilized in our notation to accommodate the easier description of the cross-validation predictions made later in the manuscript.

The classification of the node condition (wet or dry) based on the  $S_s$  surrogate model is obtained by comparing the surge estimate to the node elevation. Denoting as  $I_i^s(\mathbf{x}|\mathbf{X})$  that classification for the  $i$ th node and for a storm with input  $\mathbf{x}$ , we have:

$$I_i^s(\mathbf{x}|\mathbf{X}) = \mathbb{I}[\tilde{z}_i(\mathbf{x}|\mathbf{X}) > e_i] \quad (7)$$

where  $e_i$  is the  $i$ th node elevation and  $\mathbb{I}[\cdot]$  is the indicator function, corresponding to one if the expression inside the brackets is true and to zero otherwise.

### 4.3 Surrogate model for the classification of the node condition

The surrogate model for the node wet/dry classification,  $S_c$ , considers the binary output  $I_i(\mathbf{x}^h)$  for near-shore nodes that have remained dry for some of the storms. Nodes that are inundated for all the storms are ignored in this classification problem, since they will be always predicted as inundated by a binary classifier. Additionally, the nodes considered for  $S_c$  can be further reduced to correspond only to the  $A_{mc}$  set, which as discussed earlier, is the set that includes erroneous information for the surrogate model  $S_s$  if the *pseudo-s* database (without correction) is utilized. This provides an output vector with dimension  $n_p$  composed of the nodes that remained dry in at least one storm within the database or of the nodes belonging in set  $A_{mc}$  ( $n_p < n_z$ ). Since the original output is categorical (binary), a transformation is required in order to accommodate the approximation with a continuous (kriging) metamodel. This transformation is implemented simultaneously with the dimensionality reduction, required to accommodate an implementation to large output databases, using logistic principal component analysis (LPCA) (Schein et al. 2003).

### 4.3.1 Dimensionality reduction using logistic principal component analysis

LPCA is based on a multivariate generalization of the Bernoulli distribution, using the natural parameters (log-odds)  $\boldsymbol{\theta}$  and the canonical link function (logistic function). If  $\theta_i(\mathbf{x}^h)$  is the natural parameter for the  $i$ th node and the  $h$ th storm, then the probability of a node being wet is given by the logistic function:

$$P(I_i(\mathbf{x}^h) = 1 | \theta_i(\mathbf{x}^h)) = \frac{1}{1 + e^{-\theta_i(\mathbf{x}^h)}} \quad (9)$$

where  $P(\cdot)$  stands for probability. For facilitating the dimensionality reduction, a compact representation is assumed for the log-odds matrix  $\boldsymbol{\Theta} = [\boldsymbol{\theta}(\mathbf{x}^1) \dots \boldsymbol{\theta}(\mathbf{x}^n)]^T \in \mathbb{R}^{n \times n_p}$ , where  $\boldsymbol{\theta}$  is the log-odds vector for all considered nodes. Considering a total of  $m_c < n$  number of principal components, the compact representation is (Schein et al. 2003):

$$\boldsymbol{\Theta} = \mathbf{T}\mathbf{V}^T + \boldsymbol{\Delta} \quad (10)$$

where  $\mathbf{T}$  is the  $n \times m_c$  matrix of coefficients,  $\mathbf{V}$  is the  $n_p \times m_c$  matrix of projection vectors, and  $\boldsymbol{\Delta}$  is a matrix with each row corresponding to the same bias vector  $\boldsymbol{\Delta}_\theta^T$  ( $1 \times n_p$  vector). The unknown matrices  $\mathbf{T}$ ,  $\mathbf{V}$  and vector  $\boldsymbol{\Delta}_\theta$  can be obtained by maximizing (locally) the likelihood of the original binary observations given the representation of Eq. (9) for the natural parameters of the Bernoulli distribution (Schein et al. 2003). Unlike PCA, the results for all components of LPCA depend on the exact value adopted for  $m_c$ . The LPCA implementation ultimately provides the latent space of logistic principal components  $\mathbf{t}$  forming the matrix  $\mathbf{T}$ , as well as the projection matrix  $\mathbf{V}$ , and the bias vector  $\boldsymbol{\Delta}_\theta$ , that are used for the transformation from  $\mathbf{t}$  to  $\boldsymbol{\theta}$ . Denoting explicitly the dependence of  $\mathbf{t}$  and  $\boldsymbol{\theta}$  to  $\mathbf{x}$ , this transformation is:

$$\boldsymbol{\theta}(\mathbf{x}) = \mathbf{V}\mathbf{t}(\mathbf{x}) + \boldsymbol{\Delta}_\theta \quad (11)$$

The average misclassification error for the compact LPCA representation within the original database is given by:

$$\begin{aligned} \overline{MC}_{LPCA} &= \frac{1}{n} \sum_{h=1}^n \frac{1}{n_p} \sum_{i=1}^{n_p} |I_i(\mathbf{x}^h) - P(I_i(\mathbf{x}^h) = 1 | \theta_i(\mathbf{x}^h))| \\ &= \frac{1}{n} \sum_{h=1}^n \frac{1}{n_p} \sum_{i=1}^{n_p} \left| I_i(\mathbf{x}^h) - \frac{1}{1 + e^{-\theta_i(\mathbf{x}^h)}} \right| \end{aligned} \quad (12)$$

Even though this training error decreases as the number of principal components increases, LPCA is unfortunately known to be very prone to overfitting (Lee et al. 2010). Numerous approaches have been proposed to address this shortcoming (Lee et al. 2010; Song et al. 2020), with most of them sharing a common characteristic: the importance of selecting a small value for  $m_c$ . This challenge of overfitting the data is even greater for the implementation discussed here, since LPCA will be coupled with a surrogate model, forcing the selection of  $m_c$  to focus on the overall combined implementation (LPCA and surrogate model), instead of solely trying to address the LPCA overfitting.



### 4.3.2 Classification surrogate model formulation

The overfitting of the coupled LPCA-surrogate model implementation is addressed adopting an identical approach to the one developed for the surge surrogate model  $S_s$ , explicitly examining the sensitivity with respect to the selection of  $m_c$ . Similar to  $S_s$ , a surrogate model for the residual of the classification of the node condition may be considered to accommodate the use of smaller values for  $m_c$ . The steps for the  $S_c$  surrogate model development are:

- (a) For a specific  $m_c$  value perform LPCA and obtain the bias vector  $\Delta_\theta^T$ , the projection matrix  $\mathbf{V}$  and matrix  $\mathbf{T}$ . Note that the  $h$ th row of  $\mathbf{T}$  corresponds to the latent output for the  $h$ th storm  $\mathbf{t}(\mathbf{x}^h)$ , and the entire matrix corresponds to the observations for the latent components that will be used for the classification surrogate model.
- (b) Develop  $m_c$  different surrogate models based on the procedure described in ‘‘Appendix B’’ for each of the LPCA components  $t_j$ , setting  $\mathbf{Y}(\mathbf{X}) = \mathbf{T}_j(\mathbf{X})$  and  $n_y = 1$ , where  $\mathbf{T}_j(\mathbf{X})$  is the  $j$ th column of  $\mathbf{T}$ . Similar to the  $S_s$  implementation, instead of developing individual surrogate models for each component, a grouping of the components may be considered to facilitate higher computational efficiency.
- (c) Calculate the residual for node condition classification. First, the surrogate model predictions for the natural parameter vector are obtained as:

$$\tilde{\boldsymbol{\theta}}(\mathbf{x}^h|\mathbf{X}) = \mathbf{V}\tilde{\mathbf{t}}(\mathbf{x}^h|\mathbf{X}) + \Delta_\theta^T \quad (13)$$

where  $\tilde{\mathbf{t}}(\mathbf{x}^h|\mathbf{X})$  is the predicted latent output vector, with its  $j$ th component provided by Eq. (30) for the respective surrogate model ( $j=1, \dots, m_c$  such surrogate models exist). For the  $i$ th node for the  $h$ th storm the classification residual is then defined as:

$$p_i^{res}(\mathbf{x}^h|\mathbf{X}) = I_i(\mathbf{x}^h) - P(I_i(\mathbf{x}^h) = 1|\tilde{\boldsymbol{\theta}}_i(\mathbf{x}^h|\mathbf{X})) = I_i(\mathbf{x}^h) - \frac{1}{1 + e^{-\tilde{\boldsymbol{\theta}}_i(\mathbf{x}^h|\mathbf{X})}} \quad (14)$$

- (d) Develop an additional surrogate model for the residual  $\mathbf{p}^{res}$  [with each component calculated according to Eq. (13)] based on the procedure described in ‘‘Appendix B’’, setting  $\mathbf{Y} = [\mathbf{p}^{res}(\mathbf{x}^1|\mathbf{X}) \dots \mathbf{p}^{res}(\mathbf{x}^n|\mathbf{X})]^T \in \mathbb{R}^{n \times n_p}$  and  $n_y = n_p$ .

Selection of  $m_c$  should be based on parametric sensitivity analysis, as for  $m_s$ , examining the metamodel accuracy for increasing number of  $m_c$ . A  $k$ -fold cross-validation setting will be discussed in detail in Sect. 4.5 to accommodate this analysis.

### 4.3.3 Classification surrogate model predictions

The  $S_c$  surrogate model implementation combines, ultimately, separate surrogate models for the  $m_c$  LPCA components of the natural parameters of the logistic function representing the classification condition for each node, and, if deemed necessary, a surrogate model for the residual probability of each node being wet. The probability of the  $i$ th node being wet for storm input  $\mathbf{x}$  according to the  $S_c$  model is ultimately given by:

$$\tilde{p}_i^c(\mathbf{x}|\mathbf{X}) = P(I_i(\mathbf{x}) = 1|\tilde{\boldsymbol{\theta}}_i(\mathbf{x}|\mathbf{X})) + \tilde{p}_i^{res}(\mathbf{x}|\mathbf{X}) = \frac{1}{1 + e^{-\tilde{\boldsymbol{\theta}}_i(\mathbf{x}|\mathbf{X})}} + \tilde{p}_i^{res}(\mathbf{x}|\mathbf{X}) \quad (15)$$

where  $\tilde{p}_i^{res}(\mathbf{x}|\mathbf{X})$  corresponds to the mean predictions given by Eq. (30) in “Appendix B” for the surrogate model established for  $\mathbf{p}^{res}$  and  $\tilde{\theta}_i(\mathbf{x})$  is given by Eq. (10) with  $\mathbf{t}(\mathbf{x})$  replaced by  $\tilde{\mathbf{t}}(\mathbf{x}|\mathbf{X})$ . The classification prediction for node  $i$  according to surrogate model  $S_c$  can then be established by comparing value  $\tilde{p}_i^c$  to threshold 0.5. Denoting that prediction as  $I_i^c(\mathbf{x}^h|\mathbf{X})$ , we have:

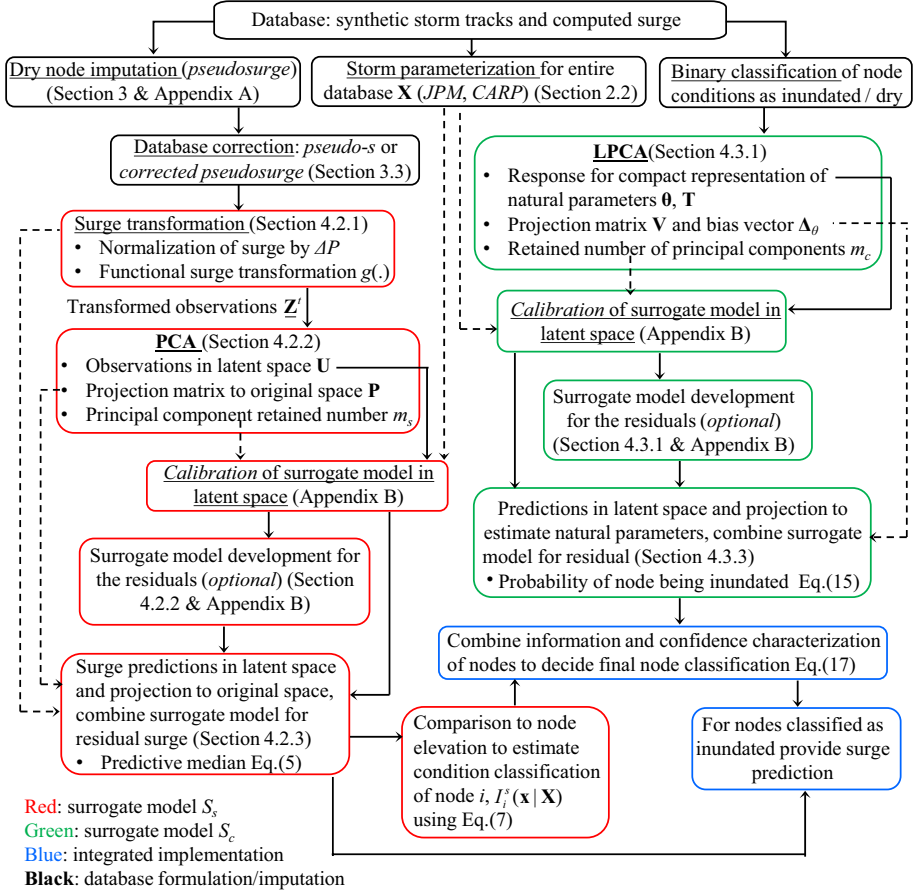
$$I_i^c(\mathbf{x}|\mathbf{X}) = \mathbb{I}[\tilde{p}_i^c(\mathbf{x}|\mathbf{X}) > 0.5] \quad (16)$$

#### 4.4 Coupled surrogate model implementation

For classifying the node condition either the  $S_s$  [Eq. (7)] or  $S_c$  [Eq. (15)] surrogate models can be used, while the surge predictions can be provided only through  $S_s$  [Eq. (5)]. An integrated formulation is proposed for the classification, coupling predictions from both  $S_s$  and  $S_c$  surrogate models. For this coupling different strategies can be adopted, for example combining the predictions from both  $S_c$  and  $S_s$  for all nodes. Model  $S_s$  is expected, though, to support higher accuracy classification predictions than model  $S_c$ , at least when pseudosurge is correctly approximated. The justification for this expectation is the fact that the binary classification presents higher challenges for any surrogate modeling technique, especially when dealing with small database sizes. Therefore, the  $S_s$  classification predictions enjoy higher trustworthiness. To leverage the strengths of each of the surrogate models,  $S_c$  and  $S_s$ , the three following groups of nodes are distinguished:

- (1)  $G^1$  group, containing nodes that remained inundated for the entire database (always wet). For these nodes, only surrogate model  $S_s$  may be utilized to predict their condition.
- (2)  $G^2$  group, containing nodes that were misclassified for at least one storm during the  $k$ NN imputation. If the *pseudo-s* database (without correction) is utilized for the formulation of  $S_s$  surrogate model, incorrect classification information is included on purpose in the database. Predictions will tend to be misclassified as false positives (nodes that are dry will be characterized as wet), since the database is biased that way. This is also evident by the comparison discussed in Fig. 2 (focus on the depicted red x markers). If  $S_s$  predicts nodes as dry, then its predictions should be trusted. If, on the other hand, node is predicted as wet, then due to the propensity of  $S_s$  for false positive misclassification, the  $S_c$  model should be used instead. If the *corrected pseudosurge* database is utilized, then the  $G^2$  group does not exist.
- (3)  $G^3$  group, containing all the remaining nodes. For these nodes, the classification can be performed either using  $S_s$  or  $S_c$ . Since the classification using  $S_s$  is trusted more than the  $S_c$  classification, then predictions for  $G^3$  are formulated exactly as the  $G^1$  predictions, ignoring the  $S_c$  surrogate model.

Based on the expectation that  $S_s$  will benefit from higher surrogate model accuracy, the recommendation is to rely only on this metamodel and utilize  $S_c$  only as a safeguard against the false positive misclassification propensity in group  $G^2$ . The final node condition classification, by coupling the two different surrogate models, is denoted as  $I_i^{cb}(\mathbf{x}|\mathbf{X})$  and for the  $i$ th node is given by:



**Fig. 3** The complete workflow for the database parameterization/imputation, the developments of the  $S_s$  and  $S_c$  surrogate models, and their coupling to get the final surge estimates. Solid arrows indicate the flow of computations and information, while dashed arrows show the secondary information utilized in the different steps

$$I_i^{cb}(\mathbf{x}|\mathbf{X}) = \begin{cases} G^1 : I_i^s(\mathbf{x}|\mathbf{X}) = \mathbb{I}[\tilde{z}_i(\mathbf{x}|\mathbf{X}) > e_i] \\ G^2 : \begin{cases} I_i^s(\mathbf{x}|\mathbf{X}) = \mathbb{I}[\tilde{z}_i(\mathbf{x}|\mathbf{X}) > e_i] & \text{if } \tilde{z}_i(\mathbf{x}|\mathbf{X}) < e_i \\ I_i^c(\mathbf{x}|\mathbf{X}) = \mathbb{I}[\tilde{p}_i^c(\mathbf{x}|\mathbf{X}) > 0.5] & \text{else} \end{cases} \\ G^3 : I_i^s(\mathbf{x}|\mathbf{X}) = \mathbb{I}[\tilde{z}_i(\mathbf{x}|\mathbf{X}) > e_i] \end{cases} \quad (17)$$

For any node that is classified as inundated,  $I_i^{cb}(\mathbf{x}^h|\mathbf{X}) = 1$ , the surge is estimated by Eqs. (5) and (6).

The workflow for the overall implementation, including the database parameterization, the surge imputation process, the development of  $S_s$  and  $S_c$ , and the combination of their predictions is reviewed in Fig. 3. It is important to stress that the surrogate model computational workflow starts with the available synthetic storm database, which includes both the synthetic storms themselves, as well as the storm surge computation (using some

appropriate numerical model) for these storms. The established surrogate model is contingent upon this information, and more specifically upon the details of the numerical model for computing the storm surge, and the characteristics of the synthetic storm tracks, especially the adopted variability along the storm track evolution.

## 4.5 Surrogate model validation

Validation is important for obtaining a confidence metric for the surrogate model prediction accuracy as well as for selecting the number of retained components for the PCA ( $m_s$ ) and LPCA ( $m_c$ ) implementations. Cross-validation (CV) is adopted for this purpose, while different statistical measures are utilized to quantify the surrogate model accuracy.

### 4.5.1 Cross-validation formulation

Cross-validation is implemented through the following steps: the storm database is partitioned to different groups; each group is sequentially removed from the database and the remaining storms are used to make predictions for the removed ones; accuracy statistics are estimated comparing these predictions to the actual storm output. The simplest implementation is the leave-one-out cross-validation (LOOCV), established by removing sequentially a single storm at a time from the original database  $\mathbf{X}$ . An alternative implementation is the  $k$ -fold cross-validation which partitions the database into  $k$  equal (or almost equal) size subsets to define the groups of storms that will be again sequentially removed.

LOOCV is typically implemented without repeating the PCA/LPCA or the hyperparameter calibration for each reduced database, since the opposite choice would substantially increase the computational complexity, requiring a total of  $n$  repetitions of the entire surrogate model calibration. This further allows the use of closed form solutions (Dubrule 1983) to obtain the leave-one-out (LOO) predictions without the need to explicitly remove each of the storms from the database. Following the notation of ‘‘Appendix B’’, the LOO predictions for output component  $y_j$  for storm  $\mathbf{x}^h$  are given by:

$$\tilde{y}_j(\mathbf{x}^h|\mathbf{X}_{-h}) = y_j(\mathbf{x}^h) - \sum_{p=1}^n \frac{[\mathbf{B}_f(\mathbf{X})]_{hp}}{[\mathbf{B}_f(\mathbf{X})]_{hh}} y_j(\mathbf{x}^p) \quad (18)$$

$$\mathbf{B}_f(\mathbf{X}) = \begin{bmatrix} \mathbf{R}(\mathbf{X}) & \mathbf{F}(\mathbf{X}) \\ \mathbf{F}(\mathbf{X})^T & \mathbf{0} \end{bmatrix}^{-1}$$

where  $\mathbf{X}_{-h}$  denotes the remaining database after the  $h$ th storm is removed and notation  $[\cdot]_{pq}$  is used to represent the entry on the  $p$ th row and  $q$ th column in a matrix. Unfortunately, such a LOO cross-validation implementation cannot explore in depth any challenges associated with overfitting, since it does not repeat the PCA or LPCA and the associated hyperparameter calibration after the removal of each storm. This challenge can be addressed using a  $k$ -fold CV and specifically by repeating both the PCA or LPCA and the hyperparameter calibration, since in this case, depending on the number of groups that will be selected, the re-calibration is not as expensive as in a LOOCV setting. If  $A_h$  is the subset containing the  $h$ th storm, the estimate  $\tilde{y}_j(\mathbf{x}^h|\mathbf{X}_{-h})$  is calculated directly from Eq. (30) by replacing in all relevant cases (on the right hand side of this equation)  $\mathbf{X}$  with  $\mathbf{X}_{-A_h}$ , representing the database  $\mathbf{X}$  after the removal of all the storms belonging in  $A_h$  subset.

Additionally, the projection (**P** PCA and **V** for LPCA) and observation (**U** for PCA and **T** for LPCA) matrices, are updated to correspond only to the retained database  $\mathbf{X}_{-A_h}$ .

The CV setting ultimately provides for the  $h$ th storm predictions  $\tilde{z}_i(\mathbf{x}^h|\mathbf{X}_{-A_h})$  from the  $S_s$  surrogate model, and  $\tilde{p}_i^c(\mathbf{x}^h|\mathbf{X}_{-A_h})$  from the  $S_c$  surrogate model, established by updating the surrogate model predictions in Eqs. (5) and (6) (for  $S_s$ ) or in Eqs. (14) and (12) (for  $S_c$ ) to correspond to the database  $\mathbf{X}_{-A_h}$ , as discussed above, including an update for the PCA and LPCA characteristics. Using these predictions the node classification can be also obtained:  $I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h})$  according to  $S_s$  utilizing Eq. (7) [using  $\tilde{z}_i(\mathbf{x}^h|\mathbf{X}_{-A_h})$ ];  $I_i^c(\mathbf{x}^h|\mathbf{X}_{-A_h})$  according to  $S_c$  utilizing Eq. (15) [using  $\tilde{p}_i^c(\mathbf{x}^h|\mathbf{X}_{-A_h})$ ]; and the combined  $I_i^{sb}(\mathbf{x}^h|\mathbf{X}_{-A_h})$  according to Eq. (16) [using  $I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h})$  and  $I_i^c(\mathbf{x}^h|\mathbf{X}_{-A_h})$ ]. For LOOCV  $\mathbf{X}_{-A_h}$  is replaced with  $\mathbf{X}_{-h}$ .

An alternative to CV is a test-sample approach, established by removing a single, large sample set of storms from  $\mathbf{X}$ , and utilizing the remaining storms to develop a surrogate model and predict the removes ones. The proposed  $k$ -fold CV offers a more comprehensive implementation of this test-sample setting since it repeats this process multiple times, for different test-sample sets.

#### 4.5.2 Validation metrics

A range of error metrics are considered for assessing the surrogate model accuracy, all estimated based on the difference between the actual output and the CV-estimated output. For the node condition classification, the adopted metric corresponds to the node misclassification percentage. For the  $S_s$  surrogate model, the total misclassification indicator for the  $i$ th node and the  $h$ th storm is given by:

$$MC_i^h = \left| I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h}) - I_i(\mathbf{x}^h) \right| \quad (19)$$

We can further distinguish between the false positive, i.e., node predicted wet when dry, and false negative, i.e., node predicted dry when wet, indicators, given, respectively, by:

$$\begin{aligned} +MC_i^h &= \max(0, I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h}) - I_i(\mathbf{x}^h)) \\ -MC_i^h &= \max(0, I_i(\mathbf{x}^h) - I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h})) \end{aligned} \quad (20)$$

where  $\max(a,b)$  is the function that provides the maximum between the two arguments  $a$  or  $b$ . Averaged statistics per storm, node, or across the entire database can be then obtained. For the total misclassification these are denoted, respectively, as  $MC^h$ ,  $MC_i$  and  $\overline{MC}$ , and are given by:

$$MC^h = \frac{1}{n_z} \sum_{i=1}^{n_z} MC_i^h; \quad MC_i = \frac{1}{n} \sum_{h=1}^n MC_i^h; \quad \overline{MC} = \frac{1}{nn_z} \sum_{h=1}^n \sum_{i=1}^{n_z} MC_i^h \quad (21)$$

The misclassification per node,  $MC_i$ , is the standard validation metric utilized in surrogate model studies, and it reflects the accuracy per output (node in this case) across the entire database. Since multiple outputs are considered, the global metamodel accuracy is characterized by averaging the results for the individual nodes (outputs), and is expressed by  $\overline{MC}$ . The misclassification per storm  $MC^h$ , provides, furthermore, a global accuracy measure for individual storms, expressing the average error across all the nodes, and it can be used to compare the surrogate model performance for individual simulations (or sets of them) within the database. Similar expressions as in Eq. (20) hold for the false positive or

false negative misclassification definitions, with the only adjustment that instead of averaging by  $n_z$ , the number of nodes that were dry (for false positive misclassification) or wet (for false negative misclassification) is utilized. For the  $S_c$  surrogate model the same statistics are established by replacing  $I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h})$  with  $I_i^c(\mathbf{x}^h|\mathbf{X}_{-A_h})$  and by restricting the comparison only across the  $n_p$  nodes utilized for  $S_c$ , whereas for the combined implementation by replacing  $I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h})$  with  $I_i^{cb}(\mathbf{x}^h|\mathbf{X}_{-A_h})$ .

For the surge predictions, both normalized and unnormalized statistics should be utilized. Unnormalized statistics reflect the absolute error, while normalized ones express the relative error, incorporating the response magnitude when assessing the size of the error. Each of them has its utility in identifying key trends in the surrogate model performance. For normalized statistics, the normalized root mean squared error (NRMSE) is adopted here as one of the popular choices, though it should be pointed out that other alternatives like the coefficient of determination or the correlation coefficient yield identical trends in the case study examined later. The NRMSE is unit less (as a normalized error metric), with values close to 0 indicating a better performance. For the  $i$ th node it is expressed by:

$$NRMSE_i = \frac{\sqrt{\sum_{h=1}^n (z_i(\mathbf{x}^h) - \tilde{z}_i(\mathbf{x}^h|\mathbf{X}_{-A_h}))^2}}{\max_{h=1, \dots, n}(z_i(\mathbf{x}^h)) - \min_{h=1, \dots, n}(z_i(\mathbf{x}^h))} \quad (22)$$

and reflects the accuracy across the entire database for a specific surrogate model output. Similar to the misclassification metric, the overall metamodel accuracy is quantified by the average error statistics across all output locations, given by:

$$\overline{NRMSE} = \frac{1}{n_z} \sum_{i=1}^{n_z} NRMSE_i \quad (23)$$

For expressing accuracy per storm, the NRMSE for the  $h$ th storm across all  $n_z$  locations can be utilized, given by:

$$NRMSE^h = \frac{\sqrt{\sum_{i=1}^{n_z} (z_i(\mathbf{x}^h) - \tilde{z}_i(\mathbf{x}^h|\mathbf{X}_{-A_h}))^2}}{\max_{i=1, \dots, n_z}(z_i(\mathbf{x}^h)) - \min_{i=1, \dots, n_z}(z_i(\mathbf{x}^h))} \quad (24)$$

As explained earlier such a validation measure, that focuses on the normalized error per storm, can facilitate the comparison of error trends for specific simulations, or (as will be utilized in the case study implementation) for groups of them.

Common candidates for unnormalized statistics include measures like the absolute mean error or the mean squared error. An alternative option, and the one chose here, is the surge score (Shisler and Johnson 2020; Plumlee et al. 2021) which for the  $i$ th node and the  $h$ th storm is described as:

$$SC_i^h = \begin{cases} |\tilde{z}_i(\mathbf{x}^h|\mathbf{X}_{-A_h}) - z_i(\mathbf{x}^h)| & \text{if } I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h}) = 1 \ \& \ I_i(\mathbf{x}^h) = 1 \\ \tilde{z}_i(\mathbf{x}^h|\mathbf{X}_{-A_h}) - e_i & \text{if } I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h}) = 1 \ \& \ I_i(\mathbf{x}^h) = 0 \\ z_i(\mathbf{x}^h) - e_i & \text{if } I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h}) = 0 \ \& \ I_i(\mathbf{x}^h) = 1 \\ 0 & \text{if } I_i^s(\mathbf{x}^h|\mathbf{X}_{-A_h}) = 0 \ \& \ I_i(\mathbf{x}^h) = 0 \end{cases} \quad (25)$$

where recall that the elevation of the node is denoted as  $e_i$ . This surge score shares the units of surge (unnormalized) and provides a penalty function for the discrepancy between the predicted and actual surge, further incorporating the node classification: if node is predicted wet and is actually wet, then the absolute value of the predicted surge discrepancy is

used as penalty function; if node is predicted wet, but it is dry then the difference between predicted surge and node elevation is used as penalty; if node is predicted dry, but it is wet then the difference between actual surge and node elevation is used as penalty; if node is predicted dry and is dry then the penalty is zero. Averaged statistics per storm, node and for the total database can be then obtained, respectively, as:

$$SC^h = \frac{1}{n_z} \sum_{i=1}^{n_z} SC_i^h; SC_i = \frac{1}{n} \sum_{h=1}^n SC_i^h; \overline{SC} = \frac{1}{nm_z} \sum_{h=1}^n \sum_{i=1}^{n_z} SC_i^h \quad (26)$$

For the combined  $S_s$  and  $S_c$  surrogate model implementation, the surge score statistics are established by replacing  $I_i^s(\mathbf{x}^h | \mathbf{X}_{-A_h})$  with  $I_i^{cb}(\mathbf{x}^h | \mathbf{X}_{-A_h})$  for the node condition classification. Evidently the surge predictions themselves  $\tilde{z}_i(\mathbf{x}^h | \mathbf{X}_{-A_h})$  stem directly from the  $S_s$  surrogate model.

## 5 Case study implementation

Details regarding the synthetic storm database were already summarized in Sect. 2. The weighted  $k$ NN interpolation described in Sect. 3 is first implemented to impute the database. The calibration set (set  $A_w^l$  discussed in ‘‘Appendix A’’) is based on nodes with depth less than 5 m, so that the interpolation hyper-parameters are selected by examining the accuracy on near-shore and overland nodes. The optimal number of  $k$  is estimated as 6, with corresponding accuracy (mean absolute error) 1.9 mm, which is considered satisfactory. As mentioned earlier, for the surrogate model development, a subset of the entire domain is considered, constrained by latitude [38° 40°] N and longitude [72° 75.7°] W (Delaware Bay), with a depth of less than 30 m. The total number of nodes in this domain is 297,460 with 158,489 of them being dry for at least one storm within the database. The  $A_{mc}$  set consists of 75,578 nodes that were misclassified at least once when using  $k$ NN to perform the surge imputation.

For the surrogate model formulation, the  $n_x=6$ -dimensional input is chosen as  $\mathbf{x}=[x_{lat} \ x_{long} \ \beta \ \Delta P \ R_{mw} \ v_f]$ . For the correlation function (detailed in ‘‘Appendix B’’), an adjusted power exponential function is considered:

$$R(\mathbf{x}^l, \mathbf{x}^m | \mathbf{s}) = \exp\left[-\sum_{j=1}^2 s_j |\mathbf{x}_j^l - \mathbf{x}_j^m|^{s_{n_x+1}} + s_j |\mathbf{x}_j^l - \mathbf{x}_j^m|^{s_{n_x+2}} + \sum_{j=4}^{n_x} s_j |\mathbf{x}_j^l - \mathbf{x}_j^m|^{s_{n_x+3}}\right]; \mathbf{s} = [s_1 \ \dots \ s_{n_x+3}] \quad (27)$$

This function is using different exponents for the three defined storm input groups: the landfall location, the heading at landfall and the remaining (strength/intensity/translational speed) three inputs. For the remaining implementation characteristics, different variants are examined. These variants explore aspects that are the focus of this paper: the storm parameterization, the potential overfitting due to the small database size, and the impact of the correction of the imputed surge. The list of these variants, and the subsection they were first discussed in this paper (in parenthesis), is the following:

- (1) The use of different number of principal components for the development of  $S_s$  ( $m_s$  value in Sect. 4.2.2) and  $S_c$  ( $m_c$  value in Sect. 4.3.1) surrogate models. For the  $S_s$  meta-model, an implementation without PCA is also considered. This case will be denoted as  $m_s=0$ .

- (2) The potential of considering, or not, a secondary surrogate model for the residual of the PCA or LPCA predictions for  $S_s$  (step (d) for the surrogate model formulation in Sect. 4.2.2) and  $S_c$  metamodels (step (d) for the surrogate model formulation in Sect. 4.3.2), respectively. When necessary these will be denoted as *Res* or *NoRes*, respectively. In all instances the characteristics of the secondary surrogate model, such as the selection of basis function and the type of calibration for the hyper-parameters, are identical to the characteristics of the primary surrogate model.
- (3) The use of MLE or LOOCV approaches for the hyper-parameter calibration of the surrogate models (discussed in “Appendix B”). These will be denoted, respectively, as MLE and CV (cross-validation). For LOOCV the formulation in (Zhang et al. 2018) is adopted.
- (4) The use of the JPM-based storm parameterization, or of the alternative one using storm features at the moment the storm track, is closest to the representative point of the domain of interest (Sect. 2.2). These will be denoted herein as *JPM* and *CARP*, respectively.
- (5) The use of the *pseudo-s* or the *corrected pseudosurge* database for the  $S_s$  metamodel (Sect. 3.3). The first case consists of the  $S_s$  surrogate model with the *pseudo-s* database; the second case uses the  $S_s$  surrogate model with the *corrected pseudosurge* database, and the third case uses the  $S_s$  surrogate model with the *pseudo-s* database in combination with  $S_c$  for the set  $A_{mc}$  containing the misclassified nodes (Sect. 4.4) These choices result ultimately in the three variants shown in Table 2.

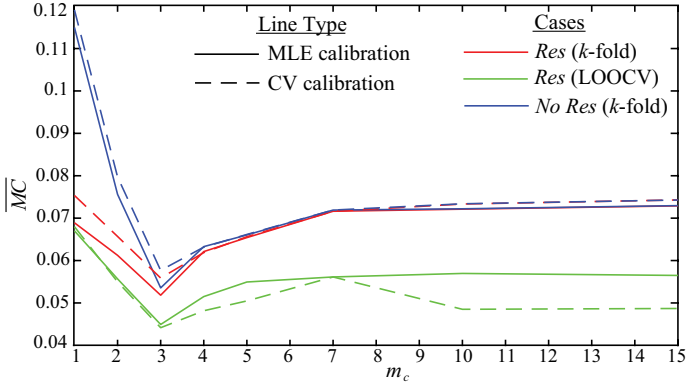
Additionally, in order to explore how other choices related to the surrogate model implementation may impact its accuracy and to compare that impact to the influence of the variants (1–5) discussed above, the following three variations will be also considered:

- (6) The use or not of the normalization by  $\Delta P$  for the  $S_s$  metamodel (Sect. 4.2.1). These will be denoted herein as  $\Delta P$  and  $N\Delta P$ , respectively.
- (7) The use or not of the functional transformation  $g = \text{sqrt}$  (Sect. 4.2.1). These will be denoted herein as *Tr* and *NTr*, respectively. The main transformation that will be discussed is  $g = \text{sqrt}$ , although some partial results for transformation  $g = \text{log}$ , which was found to considerably underperform, will be also reported. The latter case will be denoted as *TrLog*.
- (8) The use of linear basis functions  $\mathbf{f}(\mathbf{x}) = [1 \ x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6]$  or of a simpler constant basis  $\mathbf{f}(\mathbf{x}) = [1]$  (“Appendix B”). These will be denoted herein as *LB* and *NB*, respectively.

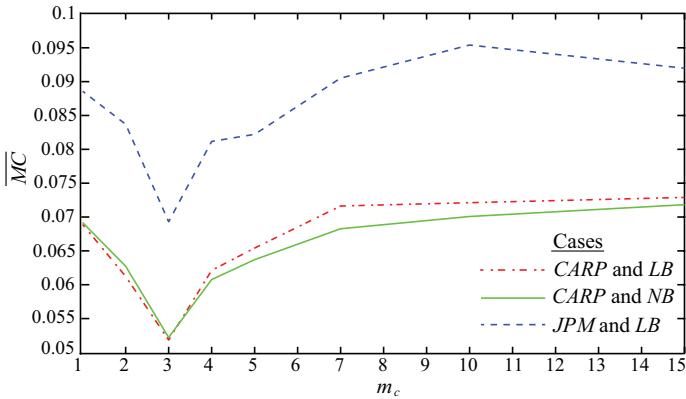
**Table 2** Summary of the variants related to the combination of  $S_s$  and  $S_c$  surrogate models and to the training database used for  $S_s$

Surrogate model implementation	Database used for $S_s$	Reference name
$S_s$ only	<i>pseudo-s</i>	$S_s$ only
$S_s$ only	<i>corrected pseudosurge</i>	$S_s$ corrected database
Combination of $S_s$ and $S_c$ for node classification, and $S_s$ for surge	<i>pseudo-s</i>	$S_s$ and $S_c$ combination





**Fig. 4** Total misclassification  $\overline{MC}$  for the  $S_c$  surrogate model for different number of components ( $m_c$ ). Different metamodel variants (use of additional metamodel on the residuals, different hyper-parameter calibration approach) and different validation approaches are shown. Across all the variants in this figure, the basis function selection is *LB* and the storm input is *CARP*



**Fig. 5** Total misclassification  $\overline{MC}$  for the  $S_c$  surrogate model for different number of components ( $m_c$ ). Metamodel variants related to storm parameterization and basis function selection are examined. Across all the variants in this figure, the validation is *k*-fold, the residual is included in the  $S_s$  formulation (*Res*) and the hyper-parameter calibration is MLE

Two different validation implementations are considered, LOOCV without repeating the PCA (for  $S_s$ ) or LPCA ( $S_c$ ) and the hyper-parameter calibration, and *k*-fold cross-validation (Sect. 4.5). These will be denoted as LOOCV and *k*-fold, respectively, and, when appropriate, they will be reported in parenthesis in the results to allow an easier distinction among the examined surrogate model variants. 19 different folds were used for the *k*-fold validation implementation, obtained by removing one storm from each of the nine MTs. This specialized *k*-fold implementation was selected to facilitate a consistent pattern of removing storms across the different *k*-folds. The number of removed storms ranged from 8 to 9 among the 19 MT-based folds. It should be stressed that, as discussed in Sect. 4.5, *k*-fold corresponds to the proper cross-validation implementation, with PCA (or LPCA) and hyper-parameter calibration repeated after the removal of each set of storms, and as such, it will be considered as the reference in all comparisons that will be established. Since

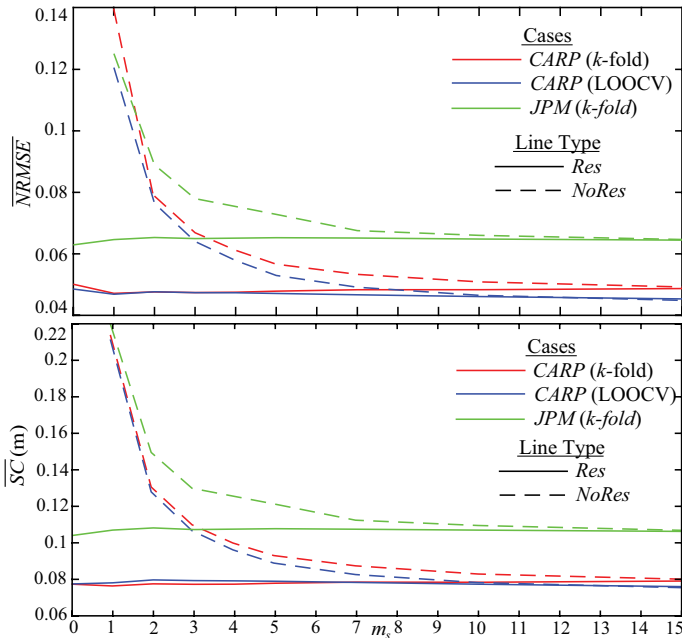
$k$ -fold has, though, substantial computational burden, LOOCV is explored as a more efficient alternative, provided that it does not encounter overfitting challenges associated with the small database size.

## 5.1 Results for the node classification surrogate model

The comparison first focuses on the  $S_c$  metamodel, utilizing the average misclassification,  $\overline{MC}$  (Eqs. 18–20), as a validation metric, examining the accuracy for an increasing number of latent components. For reference, if the implementation of (Song et al. 2020) was used to address the LPCA overfitting, the optimal number of principal components  $m_c$  would have been equal to 12. As it was stressed earlier, this number needs to be adjusted appropriately while considering the coupling with the surrogate model error, something that is examined in detail next.

The results are presented in Figs. 4 and 5. Figure 4 investigates different metamodel calibration (CV or MLE) and validation (LOOCV and  $k$ -fold) settings, as well as the use of a surrogate model for the LPCA prediction residual. Figure 5 further explores the basis function and storm parameterization selections.

Looking at the results, it is evident that the consideration of the node condition classification residual ( $Res$ ) improves the final surrogate model accuracy for smaller values of  $m_c$ , but becomes insignificant after  $m_c > 5$ . For the optimal value of  $m_c$  there seems to be some,



**Fig. 6** Total normalized root mean squared error  $\overline{NRMSE}$  (top) and surge score  $\overline{SC}$  (bottom) validation metrics for different variants of the  $S_s$  surrogate model across different number of retained components  $m_s$ . Metamodel cases correspond to either *CARP* or *JPM* for the storm parametrization and are integrating (solid lines) or not (dashed lines) a metamodel for the PCA surge residuals. Validation statistics shown for both LOOCV and  $k$ -fold implementations. Across all the variants in this figure the selected basis function is *LB*, the *pseudo-s* database is used, normalization by  $\Delta P$  and transformation using  $g(\cdot) = \sqrt{\text{Tr}}$  are also utilized

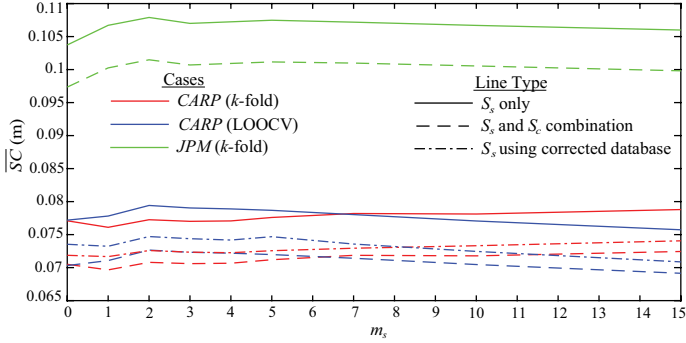
very marginal, improvement. This means that the surrogate model for the residual could be in this case ignored, providing some considerable computational benefits. Another important trend is that the LOOCV validation over-predicts the metamodel accuracy when compared to the reference ( $k$ -fold) results. This demonstrates, as stressed earlier, the fact that a proper validation needs to consider the potential overfitting, and repeat the LPCA and surrogate model calibration for the retained storm set. Though this approach has a larger computational burden, the alternative simplified formulation (LOOCV) faces challenges for the small size database examined here. Finally, the CV calibration appears to provide worse performance when the metamodel accuracy is properly estimated (using  $k$ -fold validation), verifying the anticipated challenges related to overfitting at the hyper-parameter calibration stage. It is important to stress that the results of Fig. 4 show that the use of LOOCV as a validation approach could lead to the identification of a wrong value for  $m_c$  for the case that CV calibration is utilized (observe the downward trend for larger values of  $m_c$  after 10). So not only there is an over-prediction of the accuracy statistics when LOOCV validation is used, but more importantly, erroneously identified trends related to overfitting features of the problem could promote suboptimal choices. This should be attributed to the compound effect of overfitting both at the calibration and validation stages, and to the fact that the same approach (leave-one-out cross-validation) is implemented in both stages.

The results in Fig. 5 show that the selection of basis function has a negligible effect ( $NB$  and  $LB$  behave similarly), while the use of the storm features when the storm is closest to the representative point of the domain of interest ( $CARP$  over  $JPM$ ) strongly influences the recorded performance. Apart from the  $m_c$  selection, the storm parameterization is the only variant that appears to have a strong influence on the metamodel predictive capabilities. This is a very important result and stresses the significance of the appropriate selection of the storm input compared to other surrogate model characteristics.

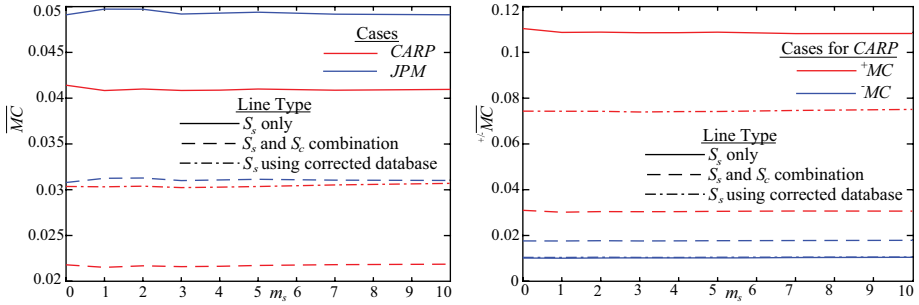
## 5.2 Results for the surge surrogate model

Moving on to the results for the  $S_s$  metamodel, emphasis is initially placed (Figs. 6–8) on the aspects that constitute the main topics of this paper: the storm parameterization, the number of principal components retained  $m_s$ , the use or not of a surrogate model for the surge residuals, the selection of the database, and the combination or not with the  $S_c$  surrogate model predictions. For reference, in order for PCA to explain 99.99% of the variability of the original database, a total of  $m_s = 123$  latent components would have been retained as the appropriate ones to inform the subsequent surrogate model development.

Figure 6 shows the results for the average normalized RMSE,  $\overline{NRMSE}$  (Eqs. 21–22), and surge score,  $\overline{SC}$  (Eqs. 24–25), error metrics across the different number of retained components  $m_s$  for the  $S_s$  surrogate model, examining the impact of the storm parameterization ( $CARP$  compared to  $JPM$ ), of the integration or not with a surrogate model for the PCA residuals ( $Res$  or  $NoRes$ ), and of the appropriate validation process (LOOCV or  $k$ -fold). The results make evident that, similar to the  $S_c$  case,  $CARP$  facilitates a better performance and that the LOOCV validation is potentially identifying some erroneous trends. The latter is evident by the continuous (though admittedly small) improvement in performance for larger values of  $m_s$ , something that is not aligned with the estimates offered through the  $k$ -fold implementation. This stresses, similar to the  $S_c$  surrogate model implementation, the importance of a validation setting that examines the influence of the PCA itself on the observed accuracy trends in order to be able to identify any overfitting issues. Results in both subplots indicate that when the residual is not incorporated in the



**Fig. 7** Total surge score  $\overline{SC}$  for the  $S_s$  surrogate model using the *pseudo-s* database and combining it or not with  $S_c$ , and  $SC$  for  $S_s$  metamodel that uses the *corrected pseudosurge* database. For  $S_s$ , results for different number of retained components  $m_s$  are shown, for the same variants examined in Fig. 6 with only one exception: cases that include a separate model for the PCA residuals are only considered here (*Res*)



**Fig. 8** Total, misclassification  $\overline{MC}$  (left) and its decomposition (right) to false positives  $^+\overline{MC}$  and false negatives  $^-\overline{MC}$  for the implementation corresponding to *CARP* storm parameterization. Same variants as in Fig. 6 are presented with only one difference: validation statistics shown only for the *k*-fold implementation

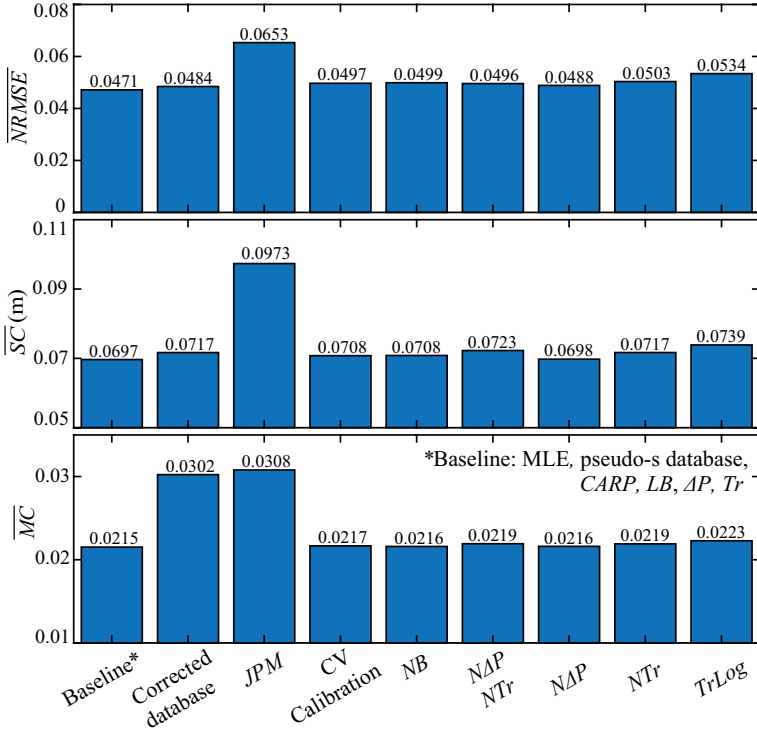
metamodel formulation (*NoRes*), an increase of  $m_s$  leads to an increase in accuracy until a plateau is reached, a trend consistent with past studies for the number of retained PCA components (Jia and Taflanidis 2013). When, though, a surrogate model for the residual is incorporated (*Res*), then independently of the value of  $m_s$ , similar performance is observed for lower values of  $m_s$ , while for substantially larger values of  $m_s$  a small, gradual deterioration of the performance (looking at both *NRMSE* and *SC*) is reported. Values of  $m_s$  in range of 1–3 seem to offer similar level of optimality, with some slight preference toward  $m_s=1$ . Similar level of accuracy can be established for the *NoRes* implementation utilizing a larger value of  $m_s$ . Note that implementation without PCA ( $m_s=0$ ) established in this case yields also a very good accuracy.

These discussions show that, contrary to the  $S_c$  surrogate model, for the  $S_s$  there are alternative formulations that can promote similar degree of predictive accuracy: *Res* with small  $m_s$  or *NoRes* with larger  $m_s$ . This different trend compared to the  $S_c$  should be attributed to the fact that, as discussed in Sect. 4, unlike *LPCA*, *PCA* as a dimensionality reduction technique suffers to a smaller degree from overfitting concerns, and the overfitting observed here should be primarily attributed to the coupling with a surrogate model. For

choosing the preferred formulation, given the aforementioned alternative choices, computational efficiency should be also considered beyond the demonstrated accuracy benefits. Looking at both the surrogate model predictions for the latent output (Eq. 30) and the transformation from that latent space to the original space through the projection matrix  $\mathbf{P}$  (Eq. 5), and considering that  $n_z \gg n \gg m_s$  (number of nodes  $\gg$  number of storms in database  $\gg$  number of retained latent components), the computational burden and memory requirements are proportional to  $m_s$  for the *NoRes* implementation and to  $n + m_s$  for the *Res* implementation, with the limiting case equal to  $n$  if no PCA is utilized ( $m_s = 0$ ). Therefore, the use of a larger  $m_s$  and the *NoRes* may be computationally optimal.

For the remainder of the results that will be presented, the implementation that considers the residual (*Res*) metamodel is adopted that appears to be more robust regarding the  $m_s$  selection with trends from Fig. 6 found to be consistent across all variants (not reported due to space limitations). Also unless otherwise specified, the validation statistics discussed correspond to  $k$ -fold. We will now further examine the variants presented in Table 2, related to the combination of the  $S_c$  and  $S_s$  metamodels and to the prediction performance when the *corrected pseudosurge* database is utilized for the  $S_s$  metamodel development. Note that in the latter case, there is no combination with the  $S_c$  metamodel, since the envisioned implementation, as discussed in Sect. 4, relies on predictions that come only from the  $S_s$  surrogate model. The  $S_c$  metamodel corresponds in all cases to the best variant identified earlier in this section:  $m_c = 3$ . Figure 7 shows the results for the  $\overline{SC}$  validation metric, and Fig. 8 for the misclassification  $\overline{MC}$  (left) and its decomposition (right subplot) to false positives  $^+ \overline{MC}$  (predicted wet when known to be dry) and false negatives  $^- \overline{MC}$  (predicted dry when known to be wet). The results indicate that the combination of  $S_s$  with  $S_c$  provides an improvement in the prediction accuracy, especially for the false positive misclassification rate; the  $S_s$  metamodel suffers from large values of  $^+ \overline{MC}$ , a pattern that is true even when the *corrected pseudosurge* database is utilized. These false positive values are immediately reduced when the proposed combination with  $S_c$  is implemented, which ultimately translates to better overall misclassification predictions (Fig. 8) and better surge score predictions (Fig. 7). Improvement in accuracy is significant when compared against the use of only  $S_s$  for the imputed surge without correction ( $S_s$  only case). These results indicate a preference for the combined metamodel implementation of  $S_s$  with  $S_c$ , with the use of metamodel  $S_s$  based on the *corrected pseudosurge* database also being a viable option, though at a reduced overall accuracy. Comparison between LOOCV and  $k$ -fold validation implementations depicted in Fig. 7 further stresses the potential problems that may arise for any validation that does not examine in detail the impact of PCA: as  $m_s$  increases, an erroneous trend of continuously improved performance is identified. As discussed earlier, this should be attributed to some (admittedly small) overfitting induced by the combined PCA and surrogate model implementation.

Figure 9 examines the different variants for the surrogate model with respect to the characteristics that were not exhaustively examined in detail in the previous figures. Validation is performed in a  $k$ -fold fashion, with the surrogate model implementation corresponding to *Res* for the optimal  $m_s$  in each case and, when appropriate, to the combination of  $S_s$  with  $S_c$  (apart from the case that  $S_s$  uses the corrected database), since all these selections were already shown to be the recommended ones. The baseline implementation corresponds to the use of the *pseudo-s* database (for  $S_s$ ), MLE optimization for the hyper-parameter calibration, *CARP* storm parameterization, linear basis function (*LB*), the normalization by  $\Delta P$  and the use of transformation  $g = \text{sqrt}(Tr)$ . The rest of the variants modify one or two of these selections and are denoted by the already established terminology. Only the modified choices are explicitly denoted for each of the examined metamodel variants in Fig. 9.



**Fig. 9** Statistics for  $\overline{NRMSE}$ ,  $\overline{SC}$  and  $\overline{MC}$  error metrics for different variants of the surrogate model implementation. Results correspond to  $k$ -fold validation, and in all instances the optimal number of components for PCA/LPCA implementation is used

Some of the previously examined variants are repeated here to facilitate an easier comparison of their optimal selections. The results indicate that most of the variations have minimal impact on the prediction accuracy. Only the use of the logarithm as transformation for  $g(\cdot)$ , or the adoption of an implementation without both the scaling by  $\Delta P$  and the transformation with  $g(\cdot)$ , provide a notably worse performance. The storm parameterization is by far the most impactful choice, with the transformation of  $g = \text{sqrt}$  and the scaling by  $\Delta P$  also offering some benefits. Comparison of the performance for  $\overline{NRMSE}$  between the two different database implementations also shows that use of the *corrected pseudosurge* database does not substantially impact the quality of the established metamodel. Note that the comparison of these two implementations for the  $\overline{SC}$  and  $\overline{MC}$  statistics is not entirely consistent, since the predictions for one of the metamodels (the one corresponding to the *pseudo-s* database without corrections), are in this case combined with the  $S_c$  metamodel predictions.

### 5.3 Examining error trends across nodes and storms

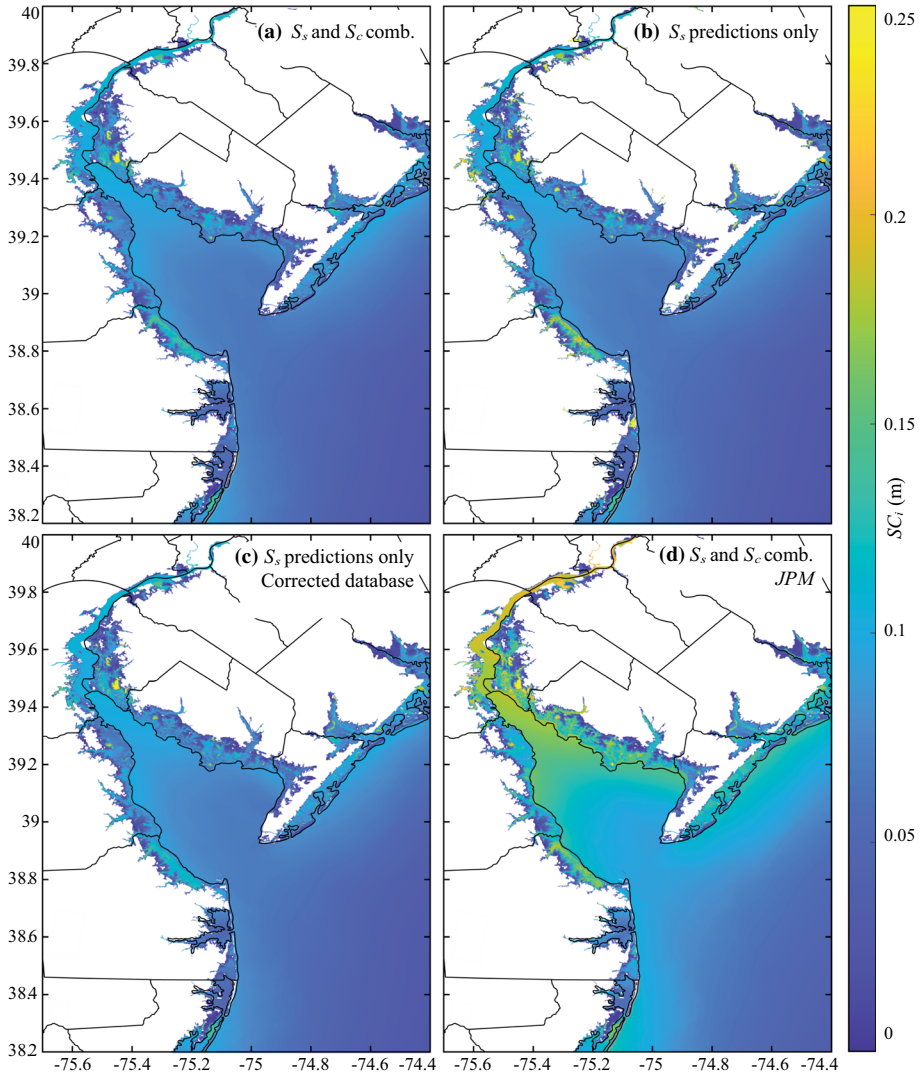
Discussions in the previous sections focused on the average error statistics across all nodes and storms. This section examines a decomposition of this error to different nodes and

storms. In all instances, validation metrics are calculated using  $k$ -fold. The baseline surrogate model examined in this section (identical to the one considered in Fig. 9) corresponds to: *Res* implementation with optimal selection for  $m_s$  and  $m_c$ , *CARP* storm parameterization, use of *pseudo-s* database for the  $S_s$  and its combination with  $S_c$ , adoption of MLE for the hyper-parameter calibration, *LB* for basis function, normalization by  $\Delta P$  and the use of transformation  $g = \text{sqrt}(Tr)$ . All other variants considered here will modify one of these choices.

Figure 10 shows the spatial distribution of the surge score per node  $SC_i$  (Eq. 25). Results are presented for the baseline surrogate model [part (a)], as well as for three additional variants: the same surrogate model but relying on predictions only from  $S_s$  [part (b)], the  $S_s$  surrogate model utilizing the *corrected pseudosurge* database [part (c)] and the surrogate model (combining  $S_s$  and  $S_c$ ) utilizing the *JPM* storm parameterization [part (d)]. The results in all subplots show that the storm surge errors are larger, as expected, for near-shore and overland nodes. Comparison between parts (a) and (b) shows that the combination of  $S_s$  and  $S_c$  offers the greatest improvements in specific near-shore regions, indicating that the benefits stemming from this combination are, ultimately, related to the database characteristics. Looking at parts (a) and (c), an improvement in certain near-shore regions is observed by using the metamodel that relies on the pseudosurge database. Finally, comparing parts (a) and (d) shows that the use of *JPM* storm input parameterization provides significant deterioration of the predictive metamodel capabilities for a substantial part of the domain, extending well beyond just the near-shore points.

Error statistics for each storm are examined next, utilizing the  $SC^h$  (Eq. 25) and  $NRMSE^h$  (Eq. 23) metrics. In order to better examine the influence of the storm input parameterization, statistics per MT (master track) group (as defined in Sect. 2) are presented in Fig. 11 for  $NRMSE^h$  (top row) and  $SC^h$  (bottom row). Results for both the baseline surrogate model with *CARP* storm parameterization (white boxplots) and the surrogate model using the *JPM* storm parameterization (shaded boxplots) are shown. Note that the normalized statistics (top row) also allow a comparison of the behavior between the MTs, since for the surge score (bottom row) the reported values are also influenced by how large the surge actually is for the storms belonging in the specific MT group. Both statistics, though, facilitate a comparison between the two different storm parameterizations across the different storm groups. It is evident that the *CARP* offers better median behavior and smaller dispersion across all MTs, especially for master tracks 6 and 9, and secondarily also for master track 3. These are the master tracks that were identified earlier to have fundamentally different surrogate model input definition for the two storm parameterization approaches (Fig. 1). Even for the remaining tracks, though, the *CARP* definition has an overall consistent positive influence. Observations validate our intuition for suggesting the *CARP* storm parameterization: selecting as surrogate model input the features of the storms when their track is further away from the geographic domain of interest provides erroneous information to the emulator, and does not confer the strength of the storm to the experienced surge.

Comparing across the different MTs, bigger challenges are identified for master tracks 1, 3 and 7 based on normalized accuracy measures. As can be seen in Fig. 1, these tracks have a combination of landfall and heading characteristics that fall near the boundary of the input domain for the given storm database (compare the landfall and heading for these cases to other MTs). As such, the lower accuracy for these storms is somewhat anticipated.

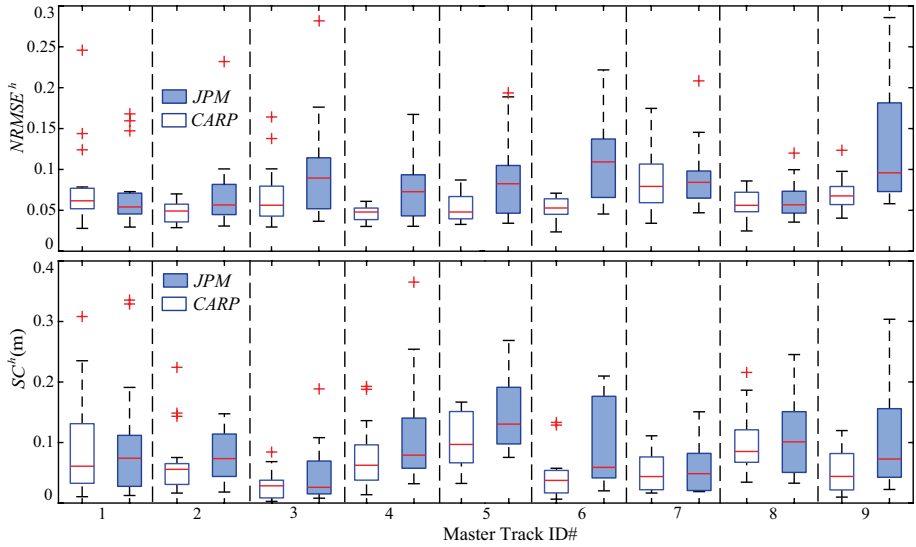


**Fig. 10** Spatial distribution of surge score per node  $SC_i$  for: (a) the baseline surrogate model combining  $S_s$  and  $S_c$ , (b) the same model but relying only on  $S_s$  predictions, (c) the surrogate model utilizing the corrected pseudosurge database, and (d) the surrogate model (combining  $S_s$  and  $S_c$ ) utilizing the JPM storm input parameterization. Black lines indicate the coastline and county boundaries. For the baseline surrogate model, the following selections are made: MLE optimization, *pseudo-s* database, *CARP*, *LB*,  $\Delta P$  and *Tr*

## 6 Conclusions

This paper offered various advancements in storm surge surrogate modeling. Kriging (Gaussian Process regression) was adopted as the surrogate model technique, and all advancements were demonstrated within a specific case study, examining 156 synthetic storms for Delaware Bay. Surge predictions across the entire geographic domain of interest,





**Fig. 11** Storm error statistics,  $NRMSE^h$  (top row) and  $SC^h$  (bottom row) for the nine different MTs for the baseline surrogate model with *CARP* storm parameterization and the surrogate model using *JPM* storm parameterization. Results are presented as box-plots, with boxes indicating 25% and 75% sample quantile, red line the median, crosses outliers and dashed lines the last non-outlier sample

including close to 300,000 nodes, were considered. The advancements can be grouped into five different themes:

- For the appropriate parameterization of the synthetic storm database, it was shown that substantial benefits are achieved in this case study by defining the surrogate model input for the storm features when the storm track is closest to a representative point of the geographic domain of interest. The traditional, alternative parameterization using the storm features at landfall can be problematic when that landfall location is further away from the geographic domain of interest. Though trends will depend on application-specific topics, such as how large the geographic domain of interest is, how far tracks are from the point of interest, and more nuanced characteristics of the storm structure, this result stresses the importance of examining alternative input definitions for the metamodel development.
- For nodes that remained dry for some of the database storms, imputation of the surge using a weighted  $k$  nearest neighbor ( $kNN$ ) interpolation can provide complete information, the so-called pseudosurge, for the development of the storm surge surrogate model. An optimization of the interpolation scheme hyper-parameters can be efficiently performed using a cross-validation setting on the near-shore wet nodes. For problematic nodes that this imputation process falsely identified as wet (imputed surge larger than node elevation), two alternatives were examined. Either correcting the imputed surge by assigning a value lower than the node elevation, or considering a secondary surrogate model for the classification of the node. The combination of logistic principal component analysis (LPCA) and Kriging was proposed here for the node classification surrogate model. It was shown that a better accuracy for the node classification, and for the overall surrogate model performance, can be accomplished by coupling the

surge and classification metamodels considering two key facts: (i) the binary condition classification surrogate model is expected to have lower accuracy overall, and (ii) the surge surrogate model has a propensity for false positive predictions for the problematic nodes. This means that the classification surrogate model should be used only for problematic nodes for which the surge surrogate model classifies nodes as wet. In all other cases, the classification based on the surge surrogate model predictions should be utilized.

- For small size databases, like the one considered in this study, the potential overfitting introduced by any selections that rely on some optimization process across the storm database needs to be carefully examined. The number of principal components for PCA and LPCA should be chosen considering the total error when these techniques are coupled with a surrogate model. A parametric analysis should be performed, measuring the surrogate model accuracy for an increasing number of retained principal components. Validation schemes that explicitly examine the impact of this selection should be adopted for this purpose. The same remark applies for the schemes of the surrogate model hyper-parameter optimization.
- It was shown that the use of a surrogate model for the residuals of the principal component analysis can decrease the number of principal components that need to be used in order to establish the same predictive accuracy. The overall computational benefits from using a reduced number of principal components in such a setting should be examined in each specific application and compared against the offered improvement in accuracy. In the case study examined here, for the node classification surrogate model, an optimal number of principal components was clearly identified, whereas the use of a surrogate model for the classification of the node condition residuals was not recommended. For the surge surrogate model, selecting a smaller number of principal components and combining it with an emulator for the surge residuals offered a slightly improved accuracy, but increased also the associated computational burden, compared to the implementation using a larger number of components without a supplemental emulator for the surge residual. It is important to note that the computational burden considered in this study focused on the surrogate model mean predictions only. Considerations about addressing the variability in these predictions were not examined.
- Scaling of surge by either  $\Delta P$  or by an appropriate functional transformation can offer an improvement in the prediction accuracy and should be explored. The exact degree of improvement or preference for a specific transformation cannot be a priori known.

## Appendix A: weighted $k$ nearest neighbor ( $k$ NN) calibration

The calibration of the weighted  $k$ NN interpolation is performed by examining its cross-validation accuracy for the always wet nodes. To formalize the implementation, denote by  $A_w^f$  the set of  $n_w^f$  always wet nodes within the database, and by  $A_w^t$  a subset of that set, with  $n_w^t$  nodes that the calibration is based upon.  $A_w^t$  may be chosen identical to  $A_w^f$ , though it should be further restricted to nodes corresponding to smaller depths, so that the calibration is based on predictions for near-shore nodes only. The surge for the  $i$ th node in  $A_w^t$  is predicted using Eq. (1) by considering its neighbors that belong in set  $A_w^f$ , excluding the  $i$ th node. This ultimately corresponds to a leave-one-out  $k$ NN prediction of the surge. As

an accuracy measure for the corresponding predictions, the average mean absolute error across all wet nodes and storms is used, given by:

$$AME_w = \frac{1}{nn_w^t} \sum_{h=1}^n \sum_{i \in A_w^t} |z_i^h - \hat{z}_i^h| \quad (28)$$

The calibration is finally expressed through the optimization of the hyper-parameters:

$$\begin{aligned} [k, d, q, p]^* &= \arg \min(AME_w) \\ k &\in \mathbb{N}, 1 \leq k \leq k_{max} \\ 0 < d &\leq d_{max}, 0 < q \leq q_{max}, p_{min} \leq p \leq p_{max} \end{aligned} \quad (29)$$

Calibration of Eq. (29) corresponds to a non-convex optimization with integer variables that is solved through a genetic algorithm (GA) implementation. To facilitate the non-convex optimization, additional box-bounded constraints are incorporated in the other hyper-parameters within the optimization, whereas  $k$  is constrained to be smaller than the  $k_{max}$  value discussed in Sect. 3.

## Appendix B: Review of surrogate model formulation

This appendix reviews the kriging surrogate model formulation. This formulation is common for the different implementations examined in Sects. 4.2 and 4.3. Input for all cases is the storm parameterization,  $\mathbf{x} \in \mathbb{R}^{n_x}$ , whereas the output definition depends on the specific implementation. This output will be denoted here as an  $n_y$ -dimensional vector  $\mathbf{y}(\mathbf{x}) \in \mathbb{R}^{n_y}$  and may correspond to the individual PCA components, or the surge residuals for the  $S_s$  surrogate model (Sect. 4.2), or to the LPCA natural parameters, or to the condition classification residuals for the  $S_c$  surrogate model (Sect. 4.3). The respective input and output matrices for the metamodel development will be denoted as  $\mathbf{X} = [\mathbf{x}^1 \dots \mathbf{x}^n]^T \in \mathbb{R}^{n \times n_x}$  and  $\mathbf{Y}(\mathbf{X}) = [\mathbf{y}(\mathbf{x}^1) \dots \mathbf{y}(\mathbf{x}^n)]^T \in \mathbb{R}^{n \times n_y}$ , respectively.

Kriging approximates the true response as a realization of a stationary stochastic Gaussian process (GP) that can also include a linear regression term (Sacks 1989), characterized by an  $n_b$ -dimensional basis vector, denoted as  $\mathbf{f}(\mathbf{x})$ , multiplied with coefficient vector  $\boldsymbol{\beta}$ . Typical choices for  $\mathbf{f}(\mathbf{x})$  is either a constant basis or some low order polynomial. The fundamental building block of Kriging is the GP correlation function  $R(\mathbf{x}^l, \mathbf{x}^m | \mathbf{s})$ , with  $\mathbf{s}$  denoting the hyper-parameter vector that needs to be calibrated. Specifics on the selection of the correlation function and the basis function will be discussed in the case study implementation. Let  $\mathbf{F}(\mathbf{X}) = [\mathbf{f}(\mathbf{x}^1) \dots \mathbf{f}(\mathbf{x}^n)]^T$  denote the  $n \times n_b$  basis matrix over database  $\mathbf{X}$ ,  $\mathbf{r}(\mathbf{x} | \mathbf{X}) = [R(\mathbf{x}, \mathbf{x}^1 | \mathbf{s}) \dots R(\mathbf{x}, \mathbf{x}^n | \mathbf{s})]^T$  the  $n$ -dimensional correlation vector between  $\mathbf{x}$  and each of the elements of  $\mathbf{X}$ , and  $\mathbf{R}(\mathbf{X})$  the  $n \times n$  correlation matrix over database  $\mathbf{X}$  with the  $lm$ -element defined as  $R(\mathbf{x}^l, \mathbf{x}^m | \mathbf{s})$ ,  $l, m = 1, \dots, n$ . To improve the surrogate model's numerical stability or even its accuracy when fitting noisy data (Sacks 1989; Gramacy and Lee 2012; Bostanabad et al. 2018), a nugget is incorporated in the formulation of the correlation function  $\mathbf{R}(\mathbf{X}) = \mathbf{R}(\mathbf{X}) + \delta \mathbf{I}_n$ , with  $\delta$  being the nugget value and  $\mathbf{I}_n$  an identity matrix of dimension  $n$ . The kriging prediction (given as row vector), corresponding to the GP predictive mean, is finally given by (Sacks 1989):

$$\begin{aligned}\tilde{\mathbf{y}}(\mathbf{x}|\mathbf{X})^T &= \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta}^*(\mathbf{X}) + \mathbf{r}(\mathbf{x}|\mathbf{X})^T \mathbf{R}(\mathbf{X})^{-1}(\mathbf{Y}(\mathbf{X}) - \mathbf{F}(\mathbf{X})\boldsymbol{\beta}^*) \\ &= \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta}^*(\mathbf{X}) + \mathbf{r}(\mathbf{x}|\mathbf{X})^T \boldsymbol{\alpha}^*(\mathbf{X})\end{aligned}\quad (30)$$

where  $\boldsymbol{\beta}^*(\mathbf{X}) \in \mathbb{R}^{n_b \times n_y}$  corresponds to the weighted least squares solution:

$$\boldsymbol{\beta}^*(\mathbf{X}) = (\mathbf{F}(\mathbf{X})^T \mathbf{R}(\mathbf{X})^{-1} \mathbf{F}(\mathbf{X}))^{-1} \mathbf{F}(\mathbf{X})^T \mathbf{R}(\mathbf{X})^{-1} \mathbf{Y}(\mathbf{X}) \quad (31)$$

and  $\boldsymbol{\alpha}^*(\mathbf{X}) \in \mathbb{R}^{n_b \times n_y}$  is defined as  $\boldsymbol{\alpha}^*(\mathbf{X}) = \mathbf{R}(\mathbf{X})^{-1}(\mathbf{Y}(\mathbf{X}) - \mathbf{F}(\mathbf{X})\boldsymbol{\beta}^*)$ . Kriging predictions can be efficiently facilitated (Lophaven et al. 2002) by keeping in memory both matrices  $\boldsymbol{\alpha}^*(\mathbf{X})$  and  $\boldsymbol{\beta}^*(\mathbf{X})$ , and for each new input  $\mathbf{x}$  multiply them by vectors  $\mathbf{f}(\mathbf{x})$  and  $\mathbf{r}(\mathbf{x}|\mathbf{X})$ . Note that the dependence on database  $\mathbf{X}$  is explicitly denoted herein, to accommodate the cross-validation discussions within the manuscript. For quantities, like  $\tilde{\mathbf{y}}(\cdot)$  and  $\mathbf{r}(\cdot)$  that are a function of  $\mathbf{x}$ , this dependence is expressed through the conditioning on  $\mathbf{X}$ , denoted as “ $|\mathbf{X}$ ”.

The quality of the surrogate model predictions is dictated by its calibration, corresponding to an optimization of the surrogate model hyper-parameters  $[\mathbf{s}, \delta]$ . This is typically done using maximum likelihood estimation (MLE) (Sacks 1989; Lophaven et al. 2002). An alternative implementation is to use cross-validation (CV) techniques (Sundararajan and Keerthi 2001) in order to identify the optimal hyper-parameters. In the latter case leave-one-out cross-validation (LOOCV) is preferred, since analytic expressions exist for estimating the leave-one-out (LOO) error with no need to sequentially remove each storm from the database (Dubrule 1983; Schobi et al. 2015). These analytic expressions greatly improve the computational efficiency for the LOOCV hyper-parameter calibration compared to alternative CV formulations.

**Acknowledgements** This work was supported by the U.S. Department of Homeland Security Coastal Resilience Center (CRC) under Grant Award Number 2015-ST-061-ND0001-01. Specific funding was provided to the Coastal Resilience Center by the Federal Emergency Management Agency via the DHS Basic Ordering Agreement HSHQDC-16-A-B0011. The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S Department of Homeland Security or the Federal Emergency Management Agency.

**Funding** This work was supported by the U.S. Department of Homeland Security Coastal Resilience Center (CRC) under Grant Award Number 2015-ST-061-ND0001-01. Specific funding was provided to the Coastal Resilience Center by the Federal Emergency Management Agency via the DHS Basic Ordering Agreement HSHQDC-16-A-B0011.

**Availability of data and material** Plan is to make the database of synthetic storms used in this study available through the National Science Foundation (NSF) NHERI (Natural Hazards Engineering Research Infrastructure) Computational Modeling and Simulation Center (SimCenter) <https://simcenter.designsafe-ci.org/>. Discussions and data transfer are currently ongoing.

**Code availability** All necessary equations are reported in the manuscript. Authors will not make complete codes publicly available but are happy to provide assistance upon reasonable request.

**Competing interests** The authors report no conflict of interest.

## References



- Al Kajbaf A, Bensi M (2020) Application of surrogate models in estimation of storm surge: A comparative assessment. *Applied Soft Computing*:106184
- Bass B, Bédient P (2018) Surrogate modeling of joint flood risk across coastal watersheds. *J Hydrol* 558:159–173

- Blanton B, Stillwell L, Roberts H, Atkinson J, Zou S, Forte M, Hanson J, Luettich RA, Jr. (2011) Coastal Storm Surge Analysis: Computational System, Report 2: Intermediate Submission No. 1.2. Engineer Research and Development Center Coastal and Hydraulics Laboratory Vicksburg MS
- Bostanabad R, Kearney T, Tao S, Apley DW, Chen W (2018) Leveraging the nugget parameter for efficient Gaussian process modeling. *Int J Numer Meth Eng* 114(5):501–516
- Contento A, Xu H, Gardoni P (2020) Probabilistic formulation for storm surge predictions. *Structure and Infrastructure Engineering*:1–20
- Cressie N, Johannesson G (2008) Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society: Series B (statistical Methodology)* 70(1):209–226
- Dubrule O (1983) Cross validation of kriging in a unique neighborhood. *J Int Assoc Math Geol* 15(6):687–699
- Dudani SA (1976) The distance-weighted k-nearest-neighbor rule. *IEEE Trans Syst Man Cybern* 4:325–327
- Furrer R, Genton MG, Nychka D (2006) Covariance tapering for interpolation of large spatial datasets. *J Comput Graph Stat* 15(3):502–523
- Gramacy RB, Lee HK (2012) Cases for the nugget in modeling computer experiments. *Stat Comput* 22(3):713–722
- Gu M, Berger JO (2016) Parallel partial Gaussian process emulation for computer models with massive output. *The Annals of Applied Statistics* 10(3):1317–1347
- Hanson JL, Forte MF, Blanton B, Gravens M, Vickery P (2013) Coastal Storm Surge Analysis: Storm Surge Results. Report 5: Intermediate Submission No. 3. Engineer Research and Development Center Vicksburg MS Coastal and Hydraulics Lab,
- Irish JL, Resio DT, Cialone MA (2009) A surge response function approach to coastal hazard assessment. Part 2: Quantification of spatial attributes of response functions. *Natural hazards* 51 (1):183–205
- Jia G, Taflanidis AA (2013) Kriging metamodeling for approximation of high-dimensional wave and surge responses in real-time storm/hurricane risk assessment. *CMAME* 261–262:24–38
- Jia G, Taflanidis AA, Nadal-Caraballo NC, Melby JA, Kennedy AB, Smith JM (2016) Surrogate modeling for peak or time-dependent storm surge prediction over an extended coastal region using an existing database of synthetic storms. *Nat Hazards* 81(2):909–938
- Jolliffe IT (2002) *Principal component analysis*. Springer series in statistics, 2nd edn. Springer, New York
- Kijewski-Correa T, Taflanidis A, Vardeman C, Sweet J, Zhang J, Snaiki R, Wu T, Silver Z, Kennedy A (2020) Geospatial environments for hurricane risk assessment: applications to situational awareness and resilience planning in New Jersey. *Frontiers in Built Environment* 6:162
- Kim S-W, Melby JA, Nadal-Caraballo NC, Ratcliff J (2015) A time-dependent surrogate model for storm surge prediction based on an artificial neural network using high-fidelity synthetic hurricane modeling. *Nat Hazards* 76(1):565–585
- Kyprioti AP, Taflanidis AA, Nadal-Caraballo NC, Campbell MO (2020) Incorporation of sea level rise in storm surge surrogate modeling. *Natural Hazards*:1–33
- Lee S, Huang JZ, Hu J (2010) Sparse logistic principal components analysis for binary data. *The Annals of Applied Statistics* 4(3):1579
- Lophaven SN, Nielsen HB, Sondergaard J (2002) DACE-A MATLAB Kriging Toolbox. Technical University of Denmark,
- Luettich RA, Jr. , Westerink JJ, Scheffner NW (1992) ADCIRC: An Advanced Three-Dimensional Circulation Model for Shelves, Coasts, and Estuaries. Report 1. Theory and Methodology of ADCIRC-2DDI and ADCIRC-3DL. Coastal engineering research center vicksburg MS,
- Nadal-Caraballo NC, Gonzalez V, Campbell MO, Torres MJ, Melby JA, Taflanidis AA (2020) Coastal Hazards System: A Probabilistic Coastal Hazard Analysis Framework. *Journal of Coastal Research Global Coastal Issues of 2020 (Special Issue No. 95)*: (in press)
- Nadal-Caraballo NC, Melby JA, Gonzalez VM, Cox AT (2015) North Atlantic Coast Comprehensive Study – Coastal Storm Hazards from Virginia to Maine, ERDC/CHL TR-15-5. Vicksburg, MS. U.S. Army Engineer Research and Development Center,
- Parker K, Ruggiero P, Serafin KA, Hill DF (2019) Emulation as an approach for rapid estuarine modeling. *Coast Eng* 150:79–93
- Plumlee M, Asher TG, Chang W, Bilskie MV (2021) High-fidelity hurricane surge forecasting using emulation and sequential experiments. *Annals of Applied Statistics* in press
- Resio DT, Irish J, Cialone M (2009) A surge response function approach to coastal hazard assessment–part 1: basic concepts. *Nat Hazards* 51(1):163–182
- Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. *Stat Sci* 4(4):409–435
- Schein AI, Saul LK, Ungar LH (2003) A generalized linear model for principal component analysis of binary data. In: *Aistats*. p 10

- Schobi R, Sudret B, Wiart J (2015) Polynomial-chaos-based Kriging. *International Journal for Uncertainty Quantification* 5 (2)
- Shisler MP, Johnson DR (2020) Comparison of Methods for Imputing Non-Wetting Storm Surge to Improve Hazard Characterization. *Water* 12(5):1420
- Song Y, Westerhuis JA, Smilde AK (2020) Logistic principal component analysis via non-convex singular value thresholding. *Chemometrics and Intelligent Laboratory Systems* 204:104089
- Sundararajan S, Keerthi SS (2001) Predictive approaches for choosing hyperparameters in Gaussian processes. *Neural Comput* 13(5):1103–1118
- Taflanidis AA, Jia G, Kennedy AB, Smith JM (2013) Implementation/optimization of moving least squares response surfaces for approximation of hurricane/storm surge and wave responses. *Nat Hazards* 66(2):955–983
- Taflanidis AA, Kennedy AB, Westerink JJ, Smith J, Cheung KF, Hope M, Tanaka S (2012) Rapid assessment of wave and surge risk during landfalling hurricanes: probabilistic approach. *J Waterw Port Coast Ocean Eng* 139(3):171–182
- Toro GR, Resio DT, Divoky D, Niedoroda AW, Reed C (2010) Efficient joint-probability methods for hurricane surge frequency analysis. *Ocean Eng* 37(1):125–134
- Vickery P, Wadhwa D, Cox A, Cardone V, Hanson JL, Blanton B (2013) Coastal Storm Surge Analysis: Storm Forcing. Report 3. Intermediate Submission No. 1.3. Army Engineer Waterways Experiment Station Kitty Hawk NC Field Research Facility,
- Vickery PJ (2005) Simple empirical models for estimating the increase in the central pressure of tropical cyclones after landfall along the coastline of the United States. *JApMe* 44(12):1807–1826
- Vickery PJ, Wadhwa D (2008) Statistical models of Holland pressure profile parameter and radius to maximum winds of hurricanes from flight-level pressure and H\* Wind data. *J Appl Meteorol Climatol* 47(10):2497–2517
- Zhang J, Taflanidis AA, Nadal-Caraballo NC, Melby JA, Diop F (2018) Advances in surrogate modeling for storm surge prediction: storm selection and addressing characteristics related to climate change. *Nat Hazards* 94(3):1225–1253

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Authors and Affiliations

Aikaterini P. Kyprioti<sup>1</sup> · Alexandros A. Taflanidis<sup>1</sup>  · Matthew Plumlee<sup>2</sup> · Taylor G. Asher<sup>3</sup>  · Elaine Spiller<sup>4</sup> · Richard A. Luettich Jr<sup>5</sup> · Brian Blanton<sup>6</sup> · Tracy L. Kijewski-Correa<sup>7</sup> · Andrew Kennedy<sup>1</sup> · Lauren Schmied<sup>8</sup>

<sup>1</sup> Department of Civil and Environmental Engineering and Earth Sciences, University of Notre Dame, 156 Fitzpatrick Hall, Notre Dame, IN, USA

<sup>2</sup> Industrial Engineering and Management Sciences, Northwestern University, Evanston, USA

<sup>3</sup> Department of Marine Sciences, University of North Carolina, Chapel Hill, USA

<sup>4</sup> Mathematical and Statistical Sciences, Marquette University, Milwaukee, USA

<sup>5</sup> Institute of Marine Sciences, University of North Carolina, Chapel Hill, USA

<sup>6</sup> Renaissance Computing Institute, University of North Carolina, Chapel Hill, USA

<sup>7</sup> Department of Civil and Environmental Engineering and Earth Sciences and Keough School for Global Affairs, University of Notre Dame, Notre Dame, USA

<sup>8</sup> Engineering Resources Branch, FEMA, Washington, USA