5th August 2019

"Ensemble models from machine learning: an example of wave runup and coastal dune erosion"

Response to Reviewers

We thank both Reviewers for their time and effort in providing constructive feedback on our manuscript. Their comments have led to a much-improved manuscript with greater clarity in the description and purpose of the work presented. Below is our point-by-point response to the comments made and details of where the related changes have been made in the revised manuscript, in which they have also been highlighted. For clarity, Reviewer comments have been separated into key points which are addressed individually.

REVIEWER 2

Comment 1:

"The authors make the bold (and most likely correct) statement that the development of a perfect deterministic parameterization of wave runup using only the typical inputs of beach slope, wave height, and wave period is improbable. They then go on to develop a GP runup model that has higher skill than the most typical deterministic runup model used today (Stockdon et al., 2006). However, to build this new model they still use the same three easily obtainable inputs. While perfectly reasonable for this paper's demonstration purposes, I am left wondering whether or not GP could be used to build an even better runup model if other input forcing dimensions were included? Figure 4 appears to have some structure in it, with low values of R2 overpredicted and high values underpredicted. Can we learn something from this? Even a few suggestions and/or speculations from the authors would be welcome about machine learning directions for developing even better runup models."

Author Response to Comment 1:

We expect that the performance of the runup predictor could potentially be improved using additional inputs in the future. We feel that useful inputs to include in the next iteration of the runup model would be bar morphology (i.e., presence/absence of an offshore bar) and wave spectra. Unfortunately, this data is not as easily available as Hs, Tp and beach slope, but deserves to be collected. This speculation and direction for future work has been included in Lines 637 – 638:

> *"Future work is focused on using more data and additional inputs, such as offshore bar morphology and wave spectra, to improve the GP runup predictor developed here, testing it at different locations and integrating it into a real-time coastal erosion forecasting system."*

Comment 2:

"In developing the input Hs and Tp time series for both the creation of the runup model and for the ultimate test against the dune erosion event, it is mentioned that SWAN is used to transform all conditions into the nearshore before being linear back shoaled. Did the authors really run 100s to 1000s of individual SWAN simulations? This effort seems like it must have had a high computational cost? Since the paper emphasizes the efficiency of the GP runup model some more detail of this step in the process is warranted. Have the authors considered developing simple look up tables, or better yet, a GP model of SWAN to simplify this stage of the process?"

Author Response to Comment 2:

The Reviewer makes the good point that running 1000s of SWAN simulations to calculate nearshore wave conditions would be impractical. Here, as the Reviewer suggests, we used a pre-calculated look-up table for Narrabeen Beach to transform the offshore wave conditions, which is computationally cheap. This has now been clarified at Lines 304 – 306:

> *"…offshore wave data were first transformed to a nearshore equivalent (10 m water depth) using a pre-calculated look-up table generated with the SWAN spectral wave model based on a 10 m resolution grid…"*

Comment 3:

"The decision to use MDA for developing the training data seems sound. However, a list, or discussion of other possible space filling algorithms might be useful for readers embarking on their own GP applications."

Author Response to Comment 3:

A short discussion on alternative data selection methodologies has now been added at Lines 264–267:

> *"While alternative data-splitting routines are available, including simple random sampling, stratified random sampling, self-organizing maps and k-means clustering (Camus et al., 2011), the MDA routine used in this study was found in preliminary testing (not presented) to produce the best GP performance with the least computational expense."*

Comment 4:

"I commend the authors for their relatively parsimonious and clear explanation of GP theory in section 2.1. However, I suspect that this treatment will still be an occasionally opaque to some readers (including this reviewer). My only suggestion here is to continue to work on describing machine learning approaches such as GP in as clear of terms as possible. This paper does this as well as I have seen."

Author Response to Comment 4:

We certainly agree with the Reviewer that the clear communication of machine learning methods in general is critical to the proper implementation and interpretation of these methods to coastal problems. We very much appreciate the Reviewer's positive feedback on our attempt to do that in this manuscript.

Comment 5:

"Line 81-82: Maybe add to this growing body of literature by including: Parker, K., P. Ruggiero, K. Serafin, and D. Hill. 2019. "Emulation as an Approach for Rapid Estuarine Modeling." Coastal Engineering 150: 79–93. https://doi.org/10.1016/j.coastaleng.2019.03.004."

Author Response to Comment 5:

We thank the Reviewer for providing this excellent paper which became available during the review process of the current paper and which we have read with interest. It very nicely supports the (presently limited) body of literature around Gaussian processes in coastal applications and we have now cited it in the text at Line 82:

> *"Recent work has specifically used Gaussian processes to model coastal processes such as large scale coastline erosion (Kupilik et al., 2018) and estuarine hydrodynamics (Parker et al., 2019)."*

Comment 6:
"Line 235-236: I thank the author for identifying which toolkit they used in developing the runup GP model. However, it might be helpful for a broad group of readers if the authors listed other potential toolkits that could also have been used – say for example in Matlab, or R?"
Author Response to Comment 6:
A comment on alternative languages/software for developing Gaussian Processes has now been added at Lines 234–236:

> *"For the Reader unfamiliar with the Python programming language, alternative programs for developing Gaussian Processes include Matlab (Rasmussen and Nickisch, 2010) and R (Dancik and Dorman, 2008; MacDonald et al., 2015)."*

Comment 7:
"Line 544-546: The statement about 10,000 samples taking less than one second on a standard desktop computer is repetitive at this point."
Author Response to Comment 7:
This statement has now been removed from old Lines 544–546.