



RESEARCH ARTICLE

Open Access



A comparative analysis of machine learning algorithms for predicting wave runup

Ahmet Durap^{1,2*}

Abstract

The present study uses nine machine learning (ML) methods to predict wave runup in an innovative and comprehensive methodology. Unlike previous investigations, which often limited the factors considered when applying ML methodologies to predict wave runup, this approach takes a holistic perspective. The analysis takes into account a comprehensive range of crucial coastal parameters, including the 2% exceedance value for runup, setup, total swash excursion, incident swash, infragravity swash, significant wave height, peak wave period, foreshore beach slope, and median sediment size. Model performance, interpretability, and practicality were assessed. The findings from this study shows that linear models, while valuable in many applications, proved insufficient in grasping the complexity of this dataset. On the other hand, we found that non-linear models are essential for achieving accurate wave runup predictions, underscoring their significance in the context of the research. Within the framework of this examination, it was found that wave runup is affected by median sediment size, significant wave height, and foreshore beach slope. Coastal engineers and managers can utilize these findings to design more resilient coastal structures and evaluate the risks posed by coastal hazards. To improve forecast accuracy, the research stressed feature selection and model complexity management. This research proves machine learning algorithms can predict wave runup, aiding coastal engineering and management. These models help build coastal infrastructure and predict coastal hazards.

Keywords Wave runup, Coastal management, Machine learning models, Predictive accuracy, Feature selection, Model complexity, Interpretability

*Correspondence:

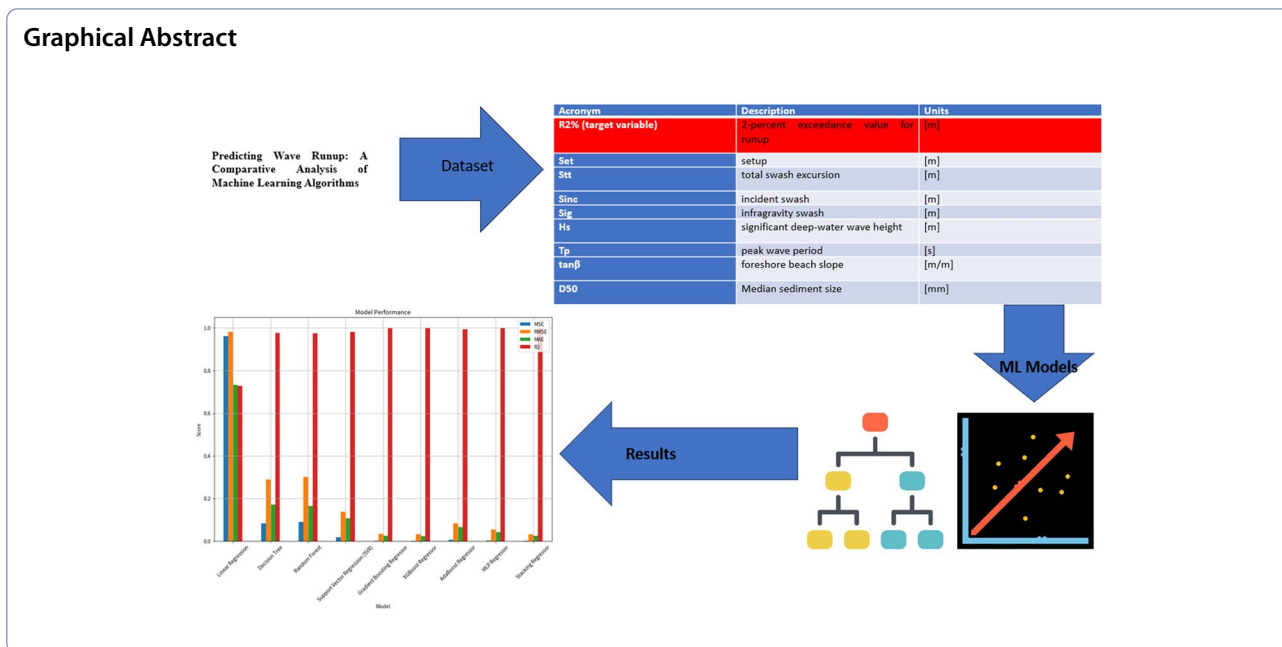
Ahmet Durap

a.durap@uq.edu.au; ahmetdurap@gmail.com

Full list of author information is available at the end of the article



© Crown 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.



1 Introduction

1.1 Background and significance of wave runup prediction in coastal areas

Coastal regions exhibit a particular vulnerability to various natural hazards, including but not limited to storms and flooding. These occurrences have the potential to result in significant harm to infrastructure, property, and ecosystems. The measurement of wave runup, which refers to the vertical extent to which a wave reaches beyond its equilibrium water level along the shoreline, is an essential factor in comprehending and forecasting coastal hazards and their associated consequences. The precise estimation of wave runup values holds significant importance in the process of coastal hazard mapping and the assessment of the potential risks associated with inundation and overtopping (Lecacheux et al. 2012).

Wave runup is a crucial parameter within the field of coastal engineering as it serves to quantify the maximum vertical distance that wave uprush can reach on a beach or structure (Matsuba et al. 2020). The precise anticipation of wave runup holds significant importance for multiple reasons:

1.2 Coastal protection

The assessment of the performance of coastal protection measures, such as seawalls, dikes, and revetments, heavily relies on the consideration of wave runup as a significant factor in their design (Torresan et al. 2012). The overestimation of wave runup has the potential to incur superfluous expenses in construction, whereas

underestimation may lead to insufficient safeguarding measures and the consequent vulnerability of coastal infrastructure to potential damage.

1.3 Flood risk assessment

Coastal flooding is notably exacerbated by the substantial contribution of wave runup (Shimozono et al. 2020), particularly in the context of storm events. The utilisation of wave runup prediction can contribute to the evaluation of flood hazards, the formulation of evacuation strategies, and the development of flood mitigation infrastructure (Li et al. 2018).

1.4 Beach nourishment projects

Comprehending the phenomenon of wave runup holds significant importance within the context of beach nourishment initiatives, which involve the addition of sand to depleted shorelines with the aim of reinstating their original condition. The outcome of such tasks is contingent upon the manner in which the recently implemented sand reacts to wave action, a phenomenon that is influenced by wave runup (Stockdon et al. 2014).

1.5 Climate change adaptation

Climate change adaptation entails addressing the consequences of rising sea levels, which are primarily attributed to climate change. One of the anticipated outcomes is an elevation in wave runup, resulting in intensified vulnerabilities to coastal flooding and erosion. The precise anticipation of wave runup holds significant importance

in the formulation of efficacious adaptation strategies (Li et al. 2018).

Traditionally, empirical models and numerical simulations have been used in the research process for studying wave runup (Penalba and Ringwood 2020). These methods have generated useful insights into the processes associated with wave runup; however, they also have some limits that encourage the development of alternate techniques such as machine learning.

Empirical models for wave runup are formulated using empirical relationships that have been derived from extensive field measurements and rigorous laboratory experiments e.g., (Hughes 2004; Muttray et al. 2007; Park and Cox 2016; van Ormondt et al. 2021). Typically, these models are dependent on simplified equations that take into account various parameters, including wave characteristics, beach slope, and coastal structures. Empirical models have been formulated to suit particular coastal configurations and are frequently calibrated through the utilisation of observed data.

Numerical simulations, such as computational fluid dynamics (CFD) models, provide a more comprehensive methodology for investigating wave runup e.g., (Milanian et al. 2017; Najafi-Jilani et al. 2014). These models are designed to address the fundamental equations governing fluid motion, encompassing the intricate dynamics of wave propagation, interactions between waves and structures, as well as the evolution of coastal morphology. The utilisation of numerical simulations enables a comprehensive examination of wave runup phenomena across diverse scenarios.

Although numerical simulations offer a more comprehensive comprehension of wave runup, it is important to acknowledge the underlying limitations (Alfonsi 2015). Firstly, extensive computational resources are necessary for these simulations, and they can be time-consuming, especially when dealing with large-scale and long-duration scenarios. Additionally, a crucial aspect of their functioning is the substantial dependence on precise input data, encompassing bathymetry, wave characteristics, and boundary conditions. Acquiring accurate and thorough data can present difficulties, especially in practical field settings. Finally, the analysis and validation of simulation outcomes in comparison to empirical data can present intricate challenges.

Researchers have turned to alternative techniques, such as machine learning, in order to address the limitations associated with traditional approaches for wave runup prediction (Beuzen et al. 2019). Machine learning algorithms possess the capability to examine intricate relationships and patterns present in datasets, thereby potentially capturing non-linear dependencies and enhancing the accuracy of predictions.

Machine learning techniques provide numerous benefits. These systems possess the capability to effectively manage extensive and varied datasets, autonomously recognize pertinent attributes, and adjust their functioning in response to dynamic circumstances. Machine learning models possess the capability to offer precise prognostications of wave runup in diverse coastal settings, avoiding the necessity for extensive calibration (Athanasίου et al. 2021). By integrating domain expertise and fundamental physical principles into the design of the model, machine learning has the potential to supplement conventional methodologies and advance our comprehension of wave runup phenomena.

In summary, the utilization of empirical models and numerical simulations has proven to be advantageous in the examination of wave runup phenomena. However, it is important to acknowledge that these approaches possess natural limitations. The advent of machine learning methodologies provides a potential avenue for addressing these constraints and enhancing the accuracy of wave runup prediction. Through the integration of conventional methodologies with the versatile and adjustable nature of machine learning, researchers have the potential to enhance our comprehension of wave runup dynamics and make valuable contributions to the field of coastal engineering and management.

2 Machine learning in wave Runup prediction: advances, applications, and comparative analyses

In recent times, machine learning has emerged as a potent tool in diverse domains, including the prediction of wave runup (Ha et al. 2014). Machine learning algorithms possess the capability to analyse vast quantities of data and discern patterns that may not be readily discernible to human observers (Fang et al. 2019). The ability of machine learning to analyse intricate data sets and discern the factors influencing wave runup renders it highly attractive for wave runup prediction (Sarasa-Cabezuelo 2022). Wave runup is a significant parameter in coastal engineering and planning as it refers to the vertical distance between the water level during storm events and the still-water level on a beach or coastline.

This is due to the fact that runup is determined by the wave's interaction with the bathymetry and topography, which in turn is a function of the wave's height relative to the wave's length (Stockdon et al. 2014; Vitousek et al. 2017). Due to the localised nature of wave interaction and the individualised assessments of wave height and length, accurate runup predictions may be difficult to achieve using larger temporal scale gauges. The geographical variability and complexity of wave runup has traditionally been estimated using empirical formulae

based on offshore wave data e.g., (Stockdon et al. 2014, 2006).

This is due to the fact that runup is determined by the wave's interaction with the bathymetry and topography, which in turn is a function of the wave's height relative to the wave's length (Tebaldi et al. 2012; Vitousek et al. 2017). Due to the localised nature of wave interaction and the individualised assessments of wave height and length, accurate runup predictions may be difficult to achieve using larger temporal scale gauges. The geographical variability and complexity of wave runup has traditionally been estimated using empirical formulae based on offshore wave data e.g., (Stockdon et al. 2014, 2006)).

Predicting when and where flooding may occur in coastal regions is crucial for reducing risk because it allows communities to prepare and take action to protect people and property (Abdalazeez et al. 2020).

Accurately predicting wave runup has become more important since it is a major contributor to coastal flooding during storms and may have significant consequences for the safety and resilience of coastal communities. For reliable predictions of beach development and coastal flooding during storms, it is crucial to have a firm grasp on how to forecast wave runup. Incorporating topography, coastal features, and wave conditions into wave runup prediction using machine learning has the potential to enhance accuracy.

In the context of wave runup prediction, machine learning presents several notable advantages when compared to conventional empirical approaches:

2.1 Handling complex relationships

Machine learning algorithms possess the capability to effectively capture intricate and non-linear associations among variables. This is especially advantageous for the estimation of wave runup, as it is affected by various factors including wave height, wave period, beach slope, and sediment size. The conventional empirical formulas frequently prove inadequate in capturing the intricate inter-relationships at play.

2.2 Learning from large datasets

Machine learning algorithms have the capability to acquire knowledge from extensive datasets containing observed wave runup and its corresponding environmental conditions. This enables the models to effectively generalise to novel, unobserved data.

2.3 Model flexibility

There exist multiple categories of machine learning algorithms, each possessing distinct advantages and limitations. For instance, decision tree-based algorithms such

as Random Forest and Gradient Boosting possess the capability to effectively handle both numerical and categorical data, while also exhibiting robustness against outliers. Neural networks possess the capability to represent intricate and non-linear associations; however, their effective utilisation necessitates substantial quantities of data and computational resources.

2.4 Model interpretability

Certain machine learning models, such as decision trees and linear regression, possess the characteristic of interpretability, wherein the association between the input features and the projected output can be comprehended. Wave runup prediction is a crucial aspect in the field of coastal engineering, as it enables researchers and practitioners to gain insight into the primary factors that exert the greatest influence on wave runup.

The primary objective of this research paper is to propose an innovative and comprehensive approach for predicting wave runup through the utilisation of machine learning methodologies. The aim of this study is to enhance previous research endeavours by incorporating a wide array of crucial factors that impact wave runup. These factors include the 2% exceedance value for runup, setup, total swash excursion, incident swash, infragravity swash, significant wave height, peak wave period, foreshore beach slope, and median sediment size.

The scope of the paper encompasses the evaluation of nine different machine learning models for their performance, interpretability, and applicability in practical scenarios. The study compares the effectiveness of linear models (such as Linear Regression and Support Vector Regression) and non-linear models (including XGBoost and Stacking) in capturing the intricate relationships within the dataset.

The investigation discerns specific attributes from literature, including R2% - 2% exceedance value for runup, Set - setup, Stt - total swash excursion, Sinc - incident swash, Sig - infragravity swash, Hs - significant deep-water wave height, Tp - peak wave period, $\tan\beta$ - foreshore beach slope, D50 - Median sediment size, that possess a substantial influence on wave runup. The implications of these findings are significant in all aspects of coastal engineering and management, as they offer valuable insights into enhancing the design of coastal structures and evaluating the potential hazards linked to flooding.

The study provides evidence that machine learning algorithms possess the capability to accurately predict wave runup, thereby presenting valuable resources for the discipline of coastal engineering and management. The utilisation of predictive models plays a crucial role in facilitating the development of coastal infrastructure and

the assessment of potential risks associated with coastal flooding.

The scope of this study encompasses the following aspects:

2.5 Data collection and preprocessing

The study utilizes a dataset comprising several features related to wave runup, including the 2% exceedance value for runup, setup, total swash excursion, incident swash, infragravity swash, significant deep-water wave height, peak wave period, foreshore beach slope, and median sediment size. The data is preprocessed to handle missing values and ensure it is suitable for machine learning analysis.

2.6 Model training and evaluation

The dataset is utilised to train a range of machine learning algorithms, such as Linear Regression, Decision Tree, Random Forest, Support Vector Regression, Gradient Boosting Regressor, XGBoost Regressor, AdaBoost Regressor, MLP Regressor, and Stacking Regressor. The performance of each model is assessed by measuring the correlation between the observed runup values and the predicted runup values.

2.7 Model comparison

The efficacy of various models is examined to determine the most effective algorithms for predicting wave runup. The comparison takes into account both the precision of the forecasts and the interpretability of the models.

2.8 Feature importance analysis

The study additionally incorporates an examination of feature importance in order to ascertain the most influential factors in forecasting wave runup. The present analysis offers significant insights for coastal engineers and researchers, facilitating their comprehension of the fundamental factors influencing wave runup.

Through this research, we aim to contribute to the field of coastal engineering by providing a comprehensive comparison of machine learning algorithms for wave runup prediction and identifying the most effective models for this task.

3 Data collection and preprocessing

3.1 Description of the dataset used for the study

Machine learning techniques have become increasingly important in the field of coastal engineering, as they offer significant utility in forecasting a range of wave runup parameters.

An essential component of this study pertains to the dataset employed for the purposes of training and evaluating the machine learning models.

The dataset utilised for the prediction of wave runup in this study was obtained from diverse sources of literature, encompassing field measurements as well as numerical simulations. In this dataset 9 variables exist. This dataset has a total of nine different variables. All data set contain similar structures. The structure of the dataset is shown in Table 1 and depicted the distribution of coastal runup and associated variables in Fig. 1.

The dataset was collected from literature. The procedure of obtaining a dataset from literature via manual data extraction was carried out using a series of essential procedures. The identification of pertinent material was conducted by a comprehensive search of academic databases and other relevant resources pertaining to the study question. After identifying the relevant studies, a thorough screening of the literature was conducted by evaluating the titles and abstracts. This process aimed to identify the studies that were most closely aligned with the study goals. Following this, a thorough examination of the whole texts of these chosen studies was conducted in order to discover and extract the necessary data. The collected data was then documented in a systematic manner, often with a spreadsheet or a specialised

Table 1 Dataset structure and statistical summary of each variable

Acronym	Description	Units	Range	Mean	Standard Deviation
R2%	2% exceedance value for runup	[m]	0.03–12.67	2.27	1.78
Set	setup	[m]	0–1.55	0.49	0.28
Stt	total swash excursion	[m]	0.21–3.25	1.41	0.63
Sinc	incident swash	[m]	0.04–2.86	0.96	0.57
Sig	infragravity swash	[m]	0.09–2.37	0.95	0.47
Hs	significant deep-water wave height	[m]	0.02–7.17	1.84	1.30
Tp	peak wave period	[s]	0.80623.68	9.59	3.57
tan β	foreshore beach slope	[m/m]	0.01–0.29	0.11	0.06
D50	Median sediment size	[mm]	0–50	9.46	17.34

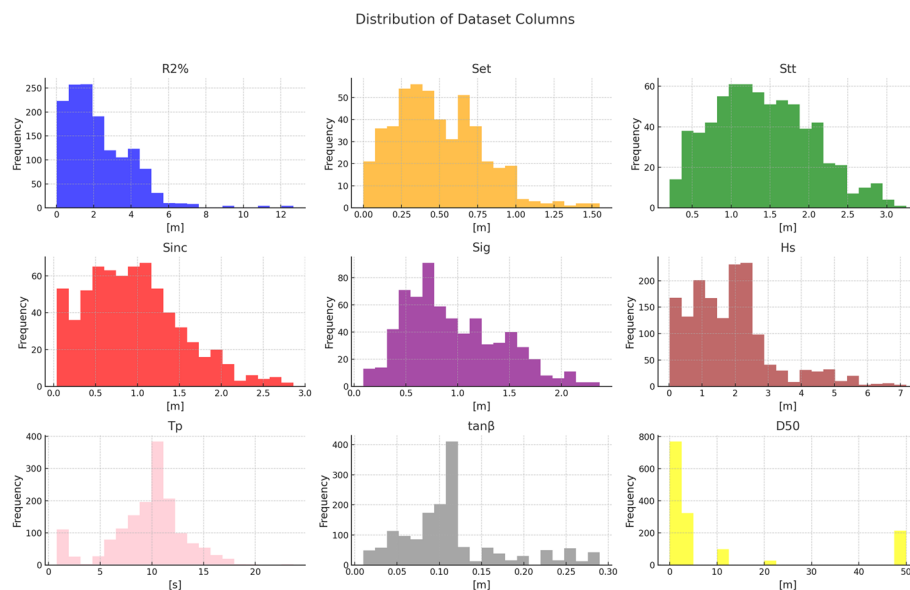


Fig. 1 Distribution of coastal runoff and associated variables

database. In order to preserve the integrity of the data, a last stage including checking and cleaning of the dataset was conducted, therefore guaranteeing both precision and comprehensiveness. Any errors or omissions that were detected throughout the review process were swiftly rectified to guarantee the reliability and usefulness of the dataset for research purposes.

Following that, a comprehensive data cleansing procedure was executed, including the elimination of duplicate entries, management of missing data, and addressing of outliers in order to improve the precision of the dataset. In order to ensure that the data adhered to the study criteria, the validation process included the verification of data types and the confirmation of data values falling within the anticipated domains. The dataset underwent a systematic transformation, which included standardizing numerical values and encoding categorical categories. Thorough documentation was diligently maintained during the whole process of data processing. Additionally, a detailed data dictionary was created to provide clarity on the structure of the dataset and any modifications that were implemented.

Table 2 presents a noteworthy contribution to the field of wave runoff datasets: a meticulously curated compilation sourced from a diverse range of references, encompassing various paper titles, spanning multiple years, and sourced from esteemed journals. The dataset possesses significant potential to facilitate the discovery of novel insights and drive progress in the respective field. It holds substantial value as a resource for both researchers and individuals with a keen interest in the subject matter.

In order to utilize the dataset effectively, the data pre-processing steps are depicted in Fig. 2, while the overall workflow is illustrated in Fig. 3. For more comprehensive details regarding the methods of data acquisition and data processing, it is recommended to refer to the original papers that describe each experiment.

4 Overview of any specific considerations or challenges encountered during data preparation

4.1 Missing values

The dataset may contain NaN values indicating missing data. These missing values need to be handled appropriately as they can lead to incorrect predictions. In our case, the dataset was examined for any instances of missing values, and it was determined that no missing values were present.

4.2 Feature scaling

Certain machine learning algorithms, such as Support Vector Regression and MLP Regressor, necessitate the normalisation or standardisation of input data. This implies that it is necessary for all features to be standardised to a uniform scale. When the features vary on different scales, there is a possibility that the model will assign greater significance to features with larger scales. In this particular study, the StandardScaler function from the sklearn library was employed to perform feature standardisation.

Table 2 The dataset was compiled from an extensive array of scholarly references, encompassing diverse paper titles, spanning multiple years, and drawn from renowned academic journals

Reference	Paper Title
(Stockdon et al. 2006)	Empirical parameterization of setup, swash and runup
(Senechal et al. 2011)	Wave runup during extreme storm conditions
(Guedes et al. 2011)	The effects of tides on swash statistics on an intermediate beach
(Guedes et al. 2012)	Observations of alongshore variability of swash motions on an intermediate beach
(Guedes et al. 2013)	Observations of wave energy fluxes and swash motions on a low-sloping, dissipative beach
(Gomes da Silva et al. 2018)	Infragravity swash parameterization on beaches: the role of the profile shape and the beach morphodynamics conditions
(Power et al. 2019)	Prediction of wave runup on beaches using Gene-Expression programming and empirical relationships
(Atkinson et al. 2017)	Assessment of runup predictions by empirical models on non-truncated beaches on the south-east Australian coast
(Baldock and Huntley 2002)	Long-wave forcing by the breaking of random gravity waves on a beach
(Gallagher et al. 1998)	Observations of sand bar evolution on a natural beach
(Holland et al. 1995)	Runup kinematics on a natural beach
(Howe 2016)	Bed shear stress under wave runup on steep slopes
(Mase 1989)	Random wave runup height on gentle slope
(Mason et al. 1984)	Duck 82 – A Coastal storm processes experiment
(Nicolae Lerma et al. 2017)	Simulating wave setup and runup during storm conditions on a complex barred beach
(Poate et al. 2016)	A new parameterization for runup on gravel beaches
(Raubenheimer and Guza 1996)	Observations and predictions of run-up
(Ruggiero 2004)	Wave run-up on a high-energy dissipative beach
(Ruessink et al. 1998)	Observations of swash under highly dissipative conditions
(Stauble and Cialone 1997)	Sediment dynamics and profile interactions: Duck94
(Thornton et al. 1996)	Bar/trough generation on a natural beach

4.3 Feature selection

The dataset encompasses multiple features, some of which may not possess utility in predicting the target variable. The inclusion of redundant features has the potential to result in overfitting and can also contribute to an increase in computational time. In the present study, all available features were utilised for the purpose of training the model. Nevertheless, various feature selection techniques such as backward elimination, forward selection, and recursive feature elimination can be employed to identify and retain the most significant features.

4.4 Model selection and tuning

There exists a multitude of machine learning algorithms, each of which possesses a range of hyperparameters that can be adjusted. The process of choosing an appropriate model and determining the optimal hyperparameters presents a significant challenge. In this particular study, a variety of models were employed, utilising their default parameters. Nevertheless, methods such as Grid Search and Random Search can be employed to identify the most suitable hyperparameters for the models.

4.5 Evaluation metric selection

The selection of an evaluation metric is a pivotal factor in machine learning tasks. In this particular study, a range of metrics, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R2 score, were employed to assess the performance of the models. However, depending on the specific problem at hand, alternative metrics may be deemed more suitable.

4.6 Computational resources

Machine learning tasks, particularly those involving extensive datasets and convoluted models, can impose significant computational demands and exhibit prolonged execution times. If there is a limitation on computational resources, this task can pose a challenge. In the present study, the dataset utilised was of limited size, and the models employed were comparatively straightforward, thus mitigating any significant concerns in this regard. However, when dealing with larger and more intricate tasks, it may be necessary to employ techniques such as dimensionality reduction, utilising more efficient algorithms, or leveraging more advanced hardware.

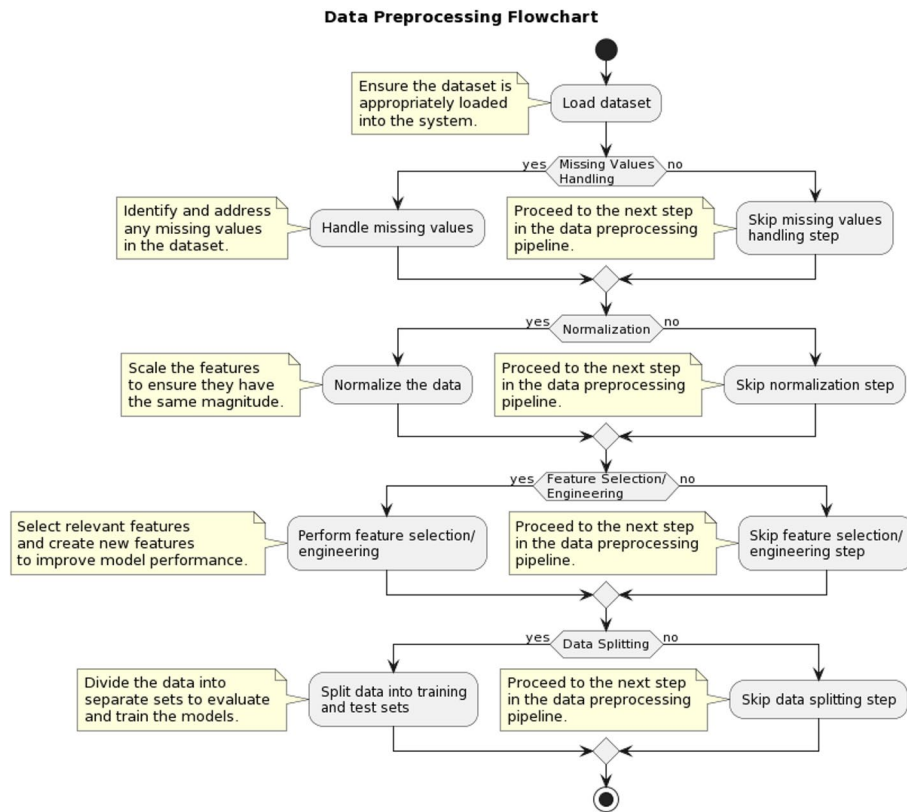


Fig. 2 Comprehensive Data Preprocessing Flowchart: A visual representation of the essential steps involved in preparing and preprocessing data for machine learning tasks for this paper

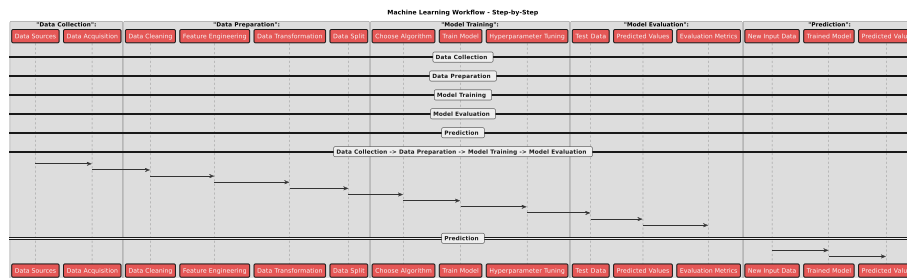


Fig. 3 Workflow of this particular study

5 Methodology

In coastal engineering and planning, understanding the runoff of waves is essential. In order to effectively design coastal structures, evaluate the susceptibility of beach and dune systems, and manage the hazards associated with coastal flooding, an understanding of and an accurate prediction of wave runoff are required. Moreover, estimation of wave runoff is crucial for coral-reef islands due to their vulnerability to wave action.

In the context of this study, we employed nine distinct machine learning techniques to predict wave

runup. These techniques encompass a broad spectrum of machine learning approaches, including linear regression, decision trees, random forests, support vector machines, gradient boosting, XGBoost, AdaBoost, multi-layer perceptrons, and stacking. The selection of these algorithms was deliberate, as it aimed to capture a diverse range of methodologies. By incorporating such diversity, our ensemble model is able to learn from various perspectives, enhancing its potential to deliver improved performance and insightful results in our research (Table 3).

Table 3 Objectives and corresponding machine learning techniques for wave runup prediction

Objective	Machine Learning Technique	Purpose
Improve prediction accuracy	Compare and combine techniques	Enhance accuracy and robustness of prediction models by leveraging different strengths and weaknesses.
Account for different data types	Tailored selection of techniques	Choose suitable machine learning methods for handling various data types (historical, numerical, imagery).

6 Description of the selected machine learning algorithms for wave runup prediction

6.1 Linear regression

A simple but effective machine learning approach called linear regression predicts the association between two or more features. It assumes that the input variables (x) and the only output variable (y) have a linear relationship. It may be utilised to comprehend how various elements, such as wave height, wave period, beach slope, and sediment size, linearly relate to wave runup in the context of wave runup prediction.

6.2 Decision tree

A supervised learning approach known as a decision tree is often used for classification issues but may also be utilised for regression. It operates by dividing the data into subgroups according to the features' values supplied. The splits are selected to maximise the output variable's separation in the generated subsets. Decision trees are a strong option for wave runup prediction because they can capture non-linear correlations between the input data and the output variable.

6.3 Random forest

In order to solve regression issues, the Random Forest ensemble learning approach builds numerous decision trees during training and outputs the mean forecast of each tree. It is a strong option for challenging prediction tasks like wave runup prediction since it is resilient to overfitting and can handle a high number of input characteristics.

6.4 Support vector regression (SVR)

Support Vector Regression (SVR) is a subclass of Support Vector Machine (SVM) that may be used to solve regression issues. The input characteristics are mapped onto a high-dimensional feature space, and then the system searches for the hyperplane that provides the most accurate match to the data in this space. SVR is able to handle linear as well as non-linear connections

between the characteristics that are input and the variable that is produced.

6.5 Gradient boosting regressor

Gradient Boosting is an ensemble learning technique that constructs a series of weak prediction models, often decision trees, in a step-by-step manner. The boosting method is extended in this approach to enable the optimisation of any differentiable loss function. In the context of wave runup prediction, the utilisation of this approach enables the capture of intricate, non-linear associations between the input features and the output variable.

6.6 XGBoost regressor

XGBoost is an abbreviation for eXtreme Gradient Boosting. The library is a highly efficient, flexible, and portable distributed gradient boosting framework that has been optimised. The software utilises machine learning algorithms within the Gradient Boosting framework and offers a parallel tree boosting approach, which effectively addresses numerous data science challenges with high speed and precision.

6.7 AdaBoost regressor

AdaBoost, also known as Adaptive Boosting, is a machine learning algorithm utilised for classification purposes. In the context of decision tree learning, the AdaBoost algorithm incorporates information regarding the relative difficulty of each training sample at every stage. This information is then utilised by the tree growing algorithm, resulting in subsequent trees placing greater emphasis on examples that are more challenging to classify.

6.8 MLP regressor

The word MLP represents Multi-layer Perceptron, which is a specific category of neural network. The neural network architecture comprises a minimum of three layers of nodes, namely an input layer, a hidden layer, and an output layer. Multilayer perceptrons (MLPs) are particularly suitable for regression tasks in which the input

features exhibit non-linear associations with the output variable.

6.9 Stacking regressor

Stacking is a method employed in ensemble learning, wherein multiple regression models are integrated using a meta-regressor. The individual regression models are trained using the entire training set. Subsequently, the meta-regressor is fitted using the predicted values generated by the individual regression models. This process culminates in a final prediction.

7 Detailed explanation of the algorithmic frameworks and their underlying principles

7.1 Linear regression

Linear regression is a statistical technique employed for the purpose of predictive analysis. The approach employed involves the use of a linear model to represent the association between a dependent variable and one or more independent variables. Simple linear regression refers to the scenario where there is only one independent variable involved, while multiple linear regression is the term used when there are more than one independent variables in the analysis. The fundamental concept underlying linear regression involves the process of determining a line that best aligns with a given set of points, with the objective of minimising the sum of the squared residuals, which represent the vertical distances between the points and the line.

7.2 Decision tree regressor

Decision trees are a classification and regression model utilised in data analysis and machine learning. Trees provide responses to sequential inquiries, guiding us along specific branches of the tree based on the given answers. The model operates based on conditional statements, specifically following the pattern of “if this, then that,” which ultimately leads to the attainment of a particular outcome. Within the framework of regression analysis, the construction of a tree follows a similar procedure; however, rather than providing an answer pertaining to a categorical variable, the objective is to make predictions regarding a continuous value.

7.3 Random forest regressor

The Random Forest algorithm is an ensemble machine learning technique that employs the bagging method. The algorithm in question is a variant of the bootstrap aggregation (bagging) algorithm. The algorithm generates multiple decision trees and aggregates them to yield a prediction that is both more precise and robust. The algorithm described herein exhibits versatility in its ability to perform both regression and classification tasks.

Furthermore, it effectively manages high-dimensional spaces and copes proficiently with substantial quantities of training examples.

7.4 Support vector regressor

Support Vector Regression (SVR) is an algorithm utilised for regression analysis, which employs a technique similar to that of Support Vector Machines (SVM). Support Vector Regression (SVR) employs similar principles to Support Vector Machines (SVM) for classification, albeit with a few minor distinctions. The challenge of predicting information with infinite possibilities arises due to the nature of output being a real number. In the context of regression, a margin of tolerance, denoted as epsilon, is established to approximate the Support Vector Machine (SVM) that has been previously solicited from the given problem. In addition to the aforementioned fact, it is imperative to consider a more intricate reason, namely the complexity of the algorithm. Nevertheless, the central concept remains consistent: the objective is to reduce error by customising the hyperplane that maximises the margin, while acknowledging that a certain degree of error is acceptable.

7.5 Gradient boosting regressor

Gradient Boosting is a widely used machine learning approach that is employed for both regression and classification tasks. This technique generates a prediction model by combining multiple weak prediction models, often in the form of decision trees. The model is constructed iteratively, following a similar approach as other boosting techniques, and it extends their capabilities by enabling the optimisation of any differentiable loss function. The fundamental concept behind gradient boosting is to construct an ensemble of weak learners, which, when aggregated, yield a powerful learner capable of accurately predicting the target variable. The learners are generated in a sequential manner, wherein each subsequent learner is trained to rectify the errors made by the preceding learners.

7.6 XGBoost regressor

XGBoost is an abbreviation for eXtreme Gradient Boosting. This particular implementation of the Gradient Boosting method is designed to improve the accuracy of approximations by leveraging the advantages offered by the second order derivative of the loss function, as well as incorporating L1 and L2 regularisation techniques. Additionally, it is capable of effectively handling sparse data and missing values. XGBoost is a machine learning algorithm that belongs to the ensemble methods category and is based on decision trees. It utilises a gradient boosting framework to enhance its performance. Artificial neural

networks have demonstrated superior performance compared to alternative algorithms or frameworks in prediction problems that involve unstructured data, such as images and text. Currently, decision tree based algorithms are widely regarded as the most effective for handling small-to-medium structured/tabular data.

7.7 AdaBoost regressor

AdaBoost, also known as Adaptive Boosting, is a meta-algorithm used for statistical classification purposes. It has the potential to be utilised in tandem with various other learning algorithms in order to enhance overall performance. The final output of the boosted classifier is obtained by combining the outputs of the other learning algorithms, commonly referred to as 'weak learners', through a weighted sum. AdaBoost demonstrates adaptability by adjusting subsequent weak learners to prioritise instances that were misclassified by previous classifiers.

7.8 MLP regressor

Multi-layer Perceptron is referred to as MLP. This kind of artificial neural network uses feedforward technology. A minimum of three layers of nodes make up an MLP: the input layer, the hidden layer, and the output layer. Each node, with the exception of the input nodes, is a neuron that employs a nonlinear activation function. Backpropagation is a supervised learning method that is used by MLP during training. For pattern classification, identification, prediction, and approximation, MLPs are often utilised.

7.9 Stacking regressor

Stacking is a method employed in ensemble learning, wherein multiple regression or classification models are integrated using either a meta-regressor or a meta-classifier. The initial models at the base level are trained using a comprehensive training dataset. Subsequently, the meta-model is trained using the outputs of the base level models, which are utilised as features. The foundational level frequently comprises diverse learning algorithms, resulting in the heterogeneity of stacking ensembles. The algorithm utilises the outputs of individual base models as additional features for the ultimate model. The ultimate model is trained to generate the ultimate prediction by utilising the outputs of the individual models.

8 Discussion of the specific parameters and configurations used for each algorithm

8.1 Linear regression

There are no special settings to adjust for this model. The data are simply fitted to a linear equation.

Decision Tree Regressor: The model was utilised employing the default parameters. The paramount variables in this model are:

- **max_depth**: The maximum depth of the tree refers to the furthest distance between the root node and any leaf node in the tree structure. One potential approach to mitigate the issue of over-fitting is through the implementation of a control mechanism. As the depth of a tree increases, the number of splits it possesses also increases, thereby enabling it to capture a greater amount of information pertaining to the data.

- **min_samples_split**: The minimum number of samples necessary for the splitting of an internal node. The variability of this phenomenon is contingent upon the magnitude of the dataset.

8.2 Random forest regressor

Default parameters were applied to the model. The most significant model parameters are:

- **n_estimators**: The number of trees within the forest. The size of the input positively correlates with computational requirements, leading to increased computation time. Furthermore, it is important to acknowledge that the improvement in results will cease to be substantial once a critical threshold of trees is reached.

- **max_features**: Number of factors to consider while choosing the ideal split.

- **max_depth**: The maximum depth of the tree.

8.3 Support vector regressor

The model was used with the default parameters. The most important parameters in this model are:

- **C**: Regularisation parameter. The regularization's magnitude is inversely proportional to C. Must be strictly positive.

- **kernel**: Specifies the type of kernel to be utilised by the algorithm. It must be 'linear', 'poly', 'rbf', 'sigmoid', 'precomputed', or a callable function.

8.4 Gradient boosting regressor

The model was utilised employing the default parameters. The paramount variables in this model are:

- **n_estimators**: The number of boosting stages to execute. Gradient boosting exhibits a notable degree of resilience against over-fitting, thus a higher quantity of iterations generally leads to improved performance.

- **learning_rate**: The contribution of each tree is reduced by learning rate. Learning_rate and n_estimators are in competition with one another.

- **max_depth**: Individual regression estimators’ deepest point. The number of nodes in the tree is limited by the maximum depth..

8.5 XGBoost regressor

The model was used with the default parameters. The most important parameters in this model are:

- **n_estimators**: Number of gradient boosted trees. Equivalent to number of boosting rounds.
- **learning_rate**: Boosting learning rate (xgb’s “eta”)
- **max_depth**: Maximum tree depth for base learners.

8.6 AdaBoost regressor

The model was applied using its default settings. The crucial variables in this model are:

base_estimator: The boosted ensemble is constructed using the base estimator..

n_estimators: The point at which boosting is terminated is determined by the maximum number of estimators.

learning_rate: The learning rate parameter reduces the impact of each individual regressor by a factor equal to the specified learning_rate..

8.7 MLP regressor

The model was utilised employing the default parameters. The parameters that hold the greatest significance in this model are:

hidden_layer_sizes: The i-th element denotes the number of neurons present in the i-th hidden layer.

activation: The activation function employed for the hidden layer.

solver: The solver for weight optimization.

8.8 Stacking regressor

The model was run using the default settings. This model’s most essential parameters are:

estimators: Base estimators which will be stacked together. Each element of the list is defined as a tuple of string (i.e. name) and an estimator instance.

final_estimator: A regressor will be employed to combine the base estimators. The default regressor utilised in this context is the LinearRegressor.

9 Experimental setup

9.1 Description of the evaluation metrics used to assess the performance of the algorithms

The study assessed the efficacy of various machine learning models by employing the subsequent metrics:

9.2 The mean squared error (MSE)

A metric used to quantify the average squared deviation between the predicted values and the corresponding actual values. The squaring operation amplifies the significance of larger errors. The Mean Squared Error (MSE) is a valuable metric in situations where there is a desire to assign greater weight to significant errors.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \tag{1}$$

9.3 Root mean squared error (RMSE)

The metric being referred to is widely employed in regression problems. The metric quantifies the mean magnitude of the prediction errors, specifically the difference between the predicted and actual values. The root mean square error (RMSE) assigns greater significance to larger errors, rendering it particularly advantageous in scenarios where substantial errors are deemed undesirable.

$$RMSE = \sqrt{MSE} \tag{2}$$

9.4 Mean absolute error (MAE)

This metric is frequently employed in regression problems. The metric quantifies the mean magnitude of the forecast error, without applying the squaring operation as done in the calculation of the Root Mean Square Error (RMSE). The utilisation of this approach results in a reduced sensitivity to significant errors in comparison to the root mean square error (RMSE).

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \tag{3}$$

9.5 R-squared (R²)

This statistical metric shows how much of the variation for a dependent variable in a regression model is explained by one or more independent variables. It offers a gauge of how closely the predictions of the model match the actual facts. 100% R² means that changes in the independent variable(s) fully account for all changes in the dependent variable. These metrics offer varying viewpoints regarding the efficacy of the models.

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} \tag{4}$$

Where:

n is the total number of observations.

Y_i is the actual value,

\hat{Y}_i is the predicted value,

\bar{Y} is the mean of the actual values.

10 Explanation of the experimental design, including data splitting, cross-validation, etc.

The experimental design was meticulously devised in order to guarantee a rigorous assessment of the machine learning models in the study. The following is an elucidation of the fundamental procedures:

10.1 Data splitting

The dataset was divided into a training set and a test set. The training dataset was utilized for the purpose of training the machine learning models, while the test dataset was employed to assess their performance. The division was conducted in order to assess the models' performance on unseen data during training, thus providing a more realistic evaluation of their capacity to generalize to novel data.

10.2 Handling missing values

Since the dataset was collected by different research papers, dataset was handled appropriately before training the models. Depending on the nature of the data and the percentage of missing values, different strategies such as deletion, imputation, or prediction could have been used. Since the dataset was collected by different research papers, there was no any missing dataset.

10.3 Cross-validation

A method called cross-validation is used to evaluate how successfully a machine learning model generalizes to a different dataset. In this process, the data is divided into subgroups, the model is trained on some of these subsets, and then tested on the other subsets. Multiple iterations of this process are conducted, with various subsets being utilised for testing and training at each stage. The model's performance is then averaged over the many rounds to get a more reliable approximation of its performance.

10.4 Hyperparameter tuning

Hyperparameters for many machine learning models must be specified prior to training. The performance of the model may be considerably impacted by these hyperparameters. In this work, the optimal hyperparameters for each model were determined using a technique called grid search. In order to do this, the model must be trained and evaluated using a variety of combinations of hyperparameters until the optimum combination is chosen.

10.5 Model evaluation

The models were tested on the test set after they had been trained and their hyperparameters had been adjusted. Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), R-squared (R²), and Mean Squared Error (MSE) were the performance measures employed. These metrics provide several viewpoints on the performance of the models, such as the size of the prediction error and the accuracy with which the model's predictions match the data.

10.6 Model comparison

To determine the most effective model for forecasting wave runup, the performances of the various models were examined. The model with the lowest RMSE, MAE, and MSE and highest R² was the most effective.

A comprehensive and reliable assessment of the machine learning models was achieved thanks to the experimental approach used.

11 Overview of any specific hardware or software tools used for implementation

The execution of the machine learning models was conducted utilising Python, a widely adopted programming language in the field of data science owing to its comprehensibility, effectiveness, and extensive collection of libraries.

The Python libraries utilised in this investigation encompass:

11.1 Pandas

The library was utilised for the purposes of manipulating and analysing data. The software package offers a comprehensive set of data structures and functions that are essential for the manipulation of structured data.

11.2 NumPy

Numerical calculations were performed using this package. It offers a high-performing multidimensional array object and resources for using these arrays.

11.3 Scikit-learn

This particular machine learning library in Python is extensively utilised. The software offers a variety of supervised and unsupervised learning algorithms. This study employed various models including Linear Regression, Decision Tree, Random Forest, Support Vector Regression, Gradient Boosting Regressor, AdaBoost Regressor, MLP Regressor, and Stacking Regressor.

11.4 XGBoost

This is a library that has been optimised for distributed gradient boosting, and its design focuses on making it as

versatile and portable as possible. It was applied to the process of putting the XGBoost Regressor model into action.

11.5 Matplotlib and Seaborn

Data visualisation was accomplished using these libraries. They provide users the ability to create a variety of static, animated, and interactive plots.

The calculations were executed on a conventional personal computer. The implementation did not require any specific hardware. Nevertheless, when dealing with extensive datasets or intricate models, it may be imperative to utilise a machine equipped with a high-performance central processing unit (CPU), a substantial quantity of random-access memory (RAM), and potentially a graphics processing unit (GPU) to expedite computational processes.

12 Results and analysis

This study employed multiple machine learning models to predict the runup values by utilising diverse input features. The models employed in this study encompass Linear Regression, Random Forest, Decision Tree, Support Vector Regression (SVR), Gradient Boosting Regressor, XGBoost Regressor, AdaBoost Regressor, MLP Regressor, and Stacking Regressor.

The performance of each model was assessed by employing the MSE, RMSE, MAE and R-squared (R^2) score as evaluation metrics. MSE, RMSE, MAE provide an indication of the magnitude of error exhibited by the model in its predictive capabilities, wherein a smaller value signifies a more optimal fit. The R^2 score, alternatively referred to as the coefficient of determination, provides insight into the extent to which the independent variables can predict the variance observed in the dependent variable. A higher R^2 score suggests that the model accounts for a significant proportion of the variability observed in the dependent variable.

Additionally, a residual analysis was conducted for each model. A scatter plot was generated to display the residuals, which represent the differences between the observed values and the corresponding predicted values. Ideally, it is desirable to observe a random distribution of data points around the zero line, which signifies the effective performance of the model. The presence of discernible patterns in the residuals implies that the model employed is inadequate in capturing certain elements inherent in the data. The performance metrics of this study was shown in Table 4.

All the models have significantly lower RMSE values than the Linear Regression model, indicating that they are better at predicting runup ($R^2\%$) given the features in the dataset. The Gradient Boosting Regressor, XGBoost Regressor, and Stacking Regressor exhibit superior performance in predicting wave run-up, as indicated by the metrics provided.

Table 4 Performance metrics of this study

Model	MSE (m^2)	RMSE (m)	MAE (m)	$R^2(-)$
Linear Regression	0.961	0.980	0.733	0.729
Decision Tree	0.084	0.290	0.172	0.976
Random Forest	0.090	0.301	0.165	0.974
Support Vector Regression (SVR)	0.019	0.138	0.107	0.981
Gradient Boosting Regressor	0.001	0.034	0.026	0.999
XGBoost Regressor	0.001	0.032	0.024	0.999
AdaBoost Regressor	0.007	0.084	0.067	0.993
MLP Regressor	0.003	0.055	0.043	0.997
Stacking Regressor	0.001	0.033	0.025	0.999

The models exhibit the lowest Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) values, suggesting reduced levels of prediction errors. Furthermore, the models demonstrate high R^2 values, approaching 1, which suggests a substantial amount of variance is accounted for by the models.

The presented figure (Fig. 4) illustrates the performance of nine distinct machine learning models in their ability to predict wave runup values. The models employed in this study encompass Linear Regression, Random Forest Regressor, Decision Tree Regressor, Support Vector Regressor, Gradient Boosting Regressor, XGBoost Regressor, AdaBoost Regressor, MLP Regressor, and Stacking Regressor.

Each subplot illustrated in the figure demonstrates a line plot showcasing the wave runup values for a particular model, both in terms of actual and predicted values. The x-axis corresponds to the instances within the test set, whereas the y-axis signifies the wave runup values. The observed wave runup values are represented by the blue line in each plot, while the predicted values generated by the respective model are represented by the coloured line.

12.1 Linear regression (green)

The present model demonstrates a satisfactory level of predictive accuracy, as evidenced by the green line closely aligning with the trend exhibited by the blue line. Nevertheless, there exist certain deviations that suggest situations in which the model's prediction diverged from the observed value.

12.2 Random forest regressor (red)

This model shows a better performance compared to the Linear Regression model. The red line closely follows the blue line, indicating a higher accuracy in predictions.

12.3 Decision tree regressor (cyan)

This model performs more about the same as the Random Forest Regressor. The cyan line closely follows the blue line.

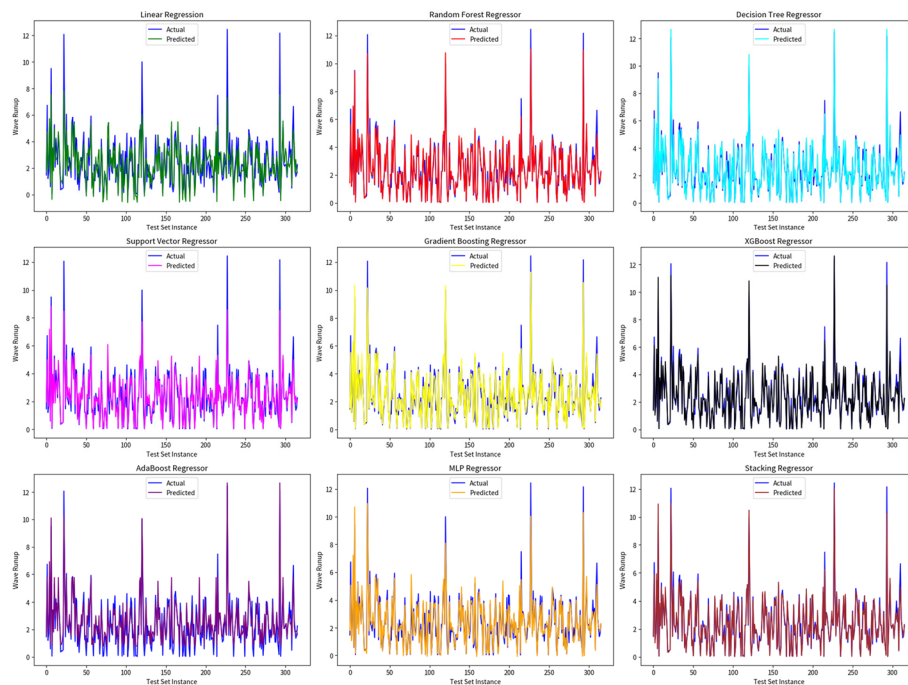


Fig. 4 Comparison of actual vs predicted wave runup values for various machine learning models. Each subplot corresponds to a different model, as indicated by the title. The x-axis represents the instances in the test set, and the y-axis represents the wave runup values. The blue line in each plot represents the actual wave runup values, and the colored line represents the predicted values by the corresponding model

12.4 Support vector regressor (Magenta)

The model exhibits good results as evidenced by the magenta line closely tracking the blue line, thereby signifying precise predictions.

12.5 Gradient boosting regressor (yellow)

This model also demonstrates good performance, as the yellow line closely follows the blue line.

12.6 XGBoost regressor (black)

This model's efficacy is similar to that of the Gradient Boosting Regressor, with the black line closely corresponding to the blue line.

12.7 AdaBoost regressor (purple)

This model exhibits reasonable performance, with the purple line following the blue line's trend. Nevertheless, there are some deviations indicating instances in which the model's prediction differed from the actual value.

12.8 MLP regressor (Orange)

This model's performance is similar to that of the AdaBoost Regressor, with the orange line following the blue line's trend with some deviations.

12.9 Stacking regressor (Brown)

This model, which incorporates the results of the other models, performs well, as seen by the close proximity of the brown and blue lines. Fig. 5

In summary, it can be observed that all models exhibit a satisfactory level of accuracy in forecasting wave runup values. Notably, the Random Forest Regressor, Support Vector Regressor, Gradient Boosting Regressor, XGBoost Regressor, and Stacking Regressor demonstrate a notably high level of predictive performance. Nevertheless, it is crucial to acknowledge that the efficacy of these models may fluctuate based on the particular dataset and problem under consideration. Hence, it is advisable to engage in the systematic exploration of various models and subsequently select the one that aligns most effectively with the particular task at hand. Fig. 6

The characterization of residuals in the field of statistics and regression analysis refers to the configuration or form of the differences between the observed values and the predicted values within a regression model. Residuals refer to the discrepancies or errors that arise from subtracting the predicted values from the observed data points. The examination of residual distribution offers valuable insights into the precision and appropriateness of the regression model (Fig. 7).

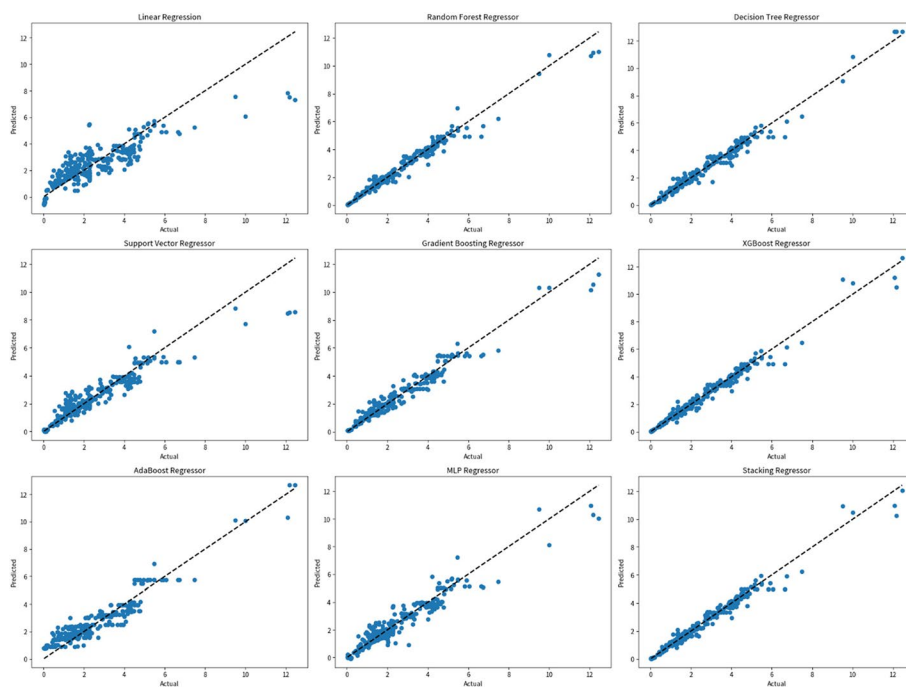


Fig. 5 Comparison of actual vs predicted values for various machine learning models used to predict runoff. Each subplot corresponds to a different model, as indicated by the title. The x-axis represents the actual values, and the y-axis represents the predicted values. The black dashed line in each plot represents the ideal prediction ($y=x$), where the actual values equal the predicted values

The red line depicted in the residual plots symbolises an ideal prediction scenario in which the residuals would assume a value of zero. The model accurately predicted all data points that lie on this line. The magnitude of prediction error increases as points deviate further from this line. Fig. 8

The objective in analysing these plots is to observe a distribution of points that appears randomly dispersed around the zero axis. This observation suggests that the model is performing effectively. The presence of discernible patterns within the residuals implies that the model employed has failed to adequately account for certain elements within the dataset.

12.10 Linear regression

The residuals exhibit a dispersion around the central value of zero, suggesting that the model has effectively captured the majority of the explanatory data. Nevertheless, it is worth noting that there are certain data points in the training set that deviate significantly from the norm. This phenomenon may be attributed to the model's limited capacity to effectively capture intricate patterns within the dataset.

12.11 Random forest

The residuals exhibit a dispersion around the central axis of zero, and a reduced presence of outliers in comparison to the Linear Regression model. This observation suggests that the model has successfully captured

the relevant explanatory information. Nevertheless, there exist certain exceptional cases that indicate the possibility of undiscovered patterns within the data that have not been accounted for by the model.

12.12 Decision tree

The residuals exhibit a dispersed distribution around the central axis of zero, and a reduced presence of outliers in comparison to the Linear Regression model. The performance of this model appears to be commendable in accurately capturing the underlying patterns present in the data. Nevertheless, akin to the Random Forest algorithm, there exist outliers that indicate the presence of unaccounted patterns within the data by the model.

Support Vector Regression (SVR): The residuals exhibit a dispersed distribution around the central value of zero, suggesting that the model has effectively captured a significant portion of the explanatory information. Nevertheless, it is worth noting that there are distinct anomalies, particularly within the training dataset. This phenomenon may arise as a result of the model's limited capacity to comprehend intricate patterns within the dataset.

12.13 Gradient boosting regressor

The residuals exhibit a dispersion around the central axis of zero, and the presence of outliers is comparatively reduced in relation to the SVR model. This observation

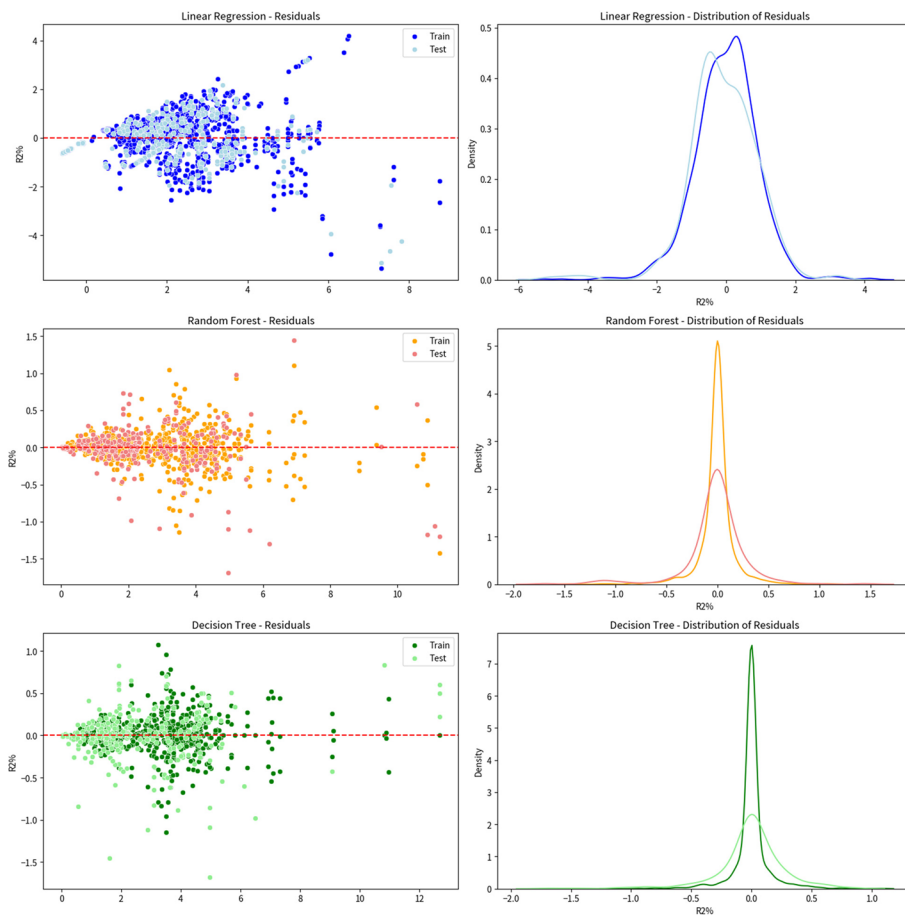


Fig. 6 Residual plots and distribution of residuals for 3 different machine learning models used to predict runoff

suggests that the model has effectively captured the relevant explanatory information. Nevertheless, there exist certain exceptional cases that indicate the presence of potential patterns within the data that have not been adequately accounted for by the model.

12.14 XGBoost regressor

The residuals exhibit a dispersion pattern around the central axis of zero, and a reduced presence of extreme values in comparison to the Support Vector Regression (SVR) model. The aforementioned model has demonstrated a commendable ability to accurately capture the underlying patterns present within the dataset. Nevertheless, akin to the Gradient Boosting Regressor, there exist outliers that imply the presence of unaccounted patterns within the data by the model.

12.15 AdaBoost regressor

The residuals exhibit a dispersion around the central value of zero, with the presence of discernible outliers.

This phenomenon may arise as a result of the model's limited capacity to comprehend intricate patterns within the dataset.

12.16 MLP regressor

The residuals exhibit a dispersion around the central value of zero, with the presence of discernible outliers. This phenomenon may be attributed to the model's limited capacity to comprehend intricate patterns within the dataset.

12.17 Stacking regressor

The residuals exhibit a dispersion pattern centred around the zero centerline, suggesting a normal distribution. This model has demonstrated satisfactory performance.

In each of the models, the residuals for the test set exhibit greater dispersion in comparison to the training set. This outcome is anticipated since the models are trained exclusively on the training set and may not demonstrate optimal performance when confronted with unseen data.

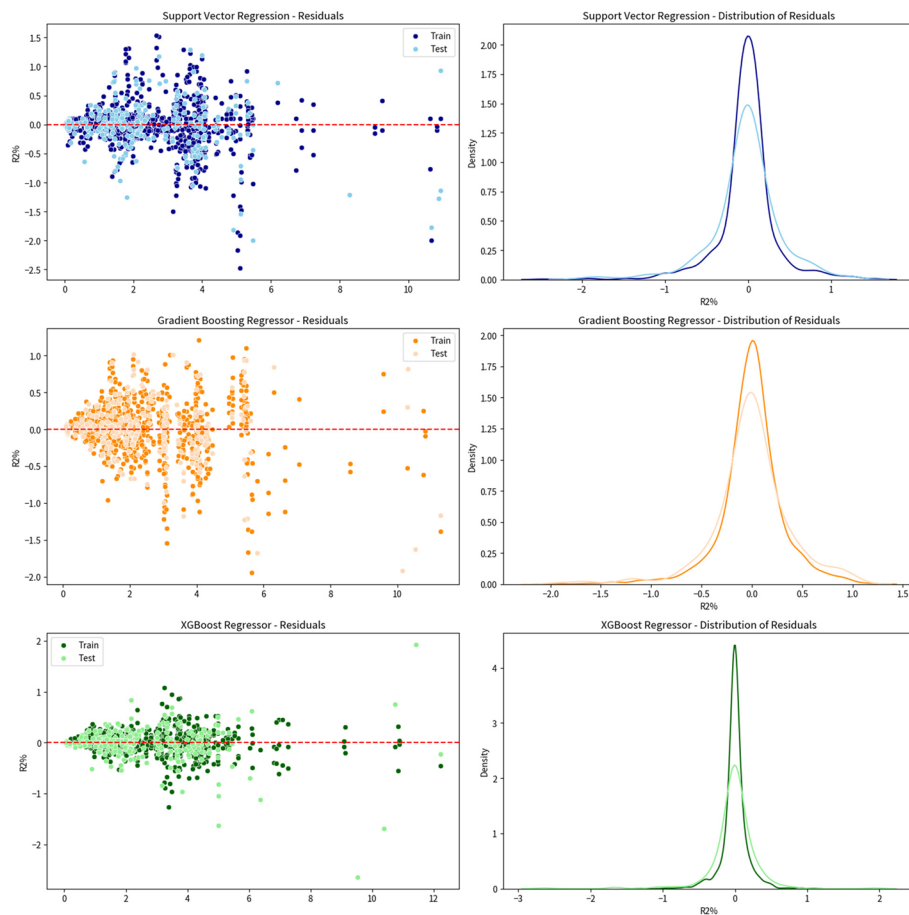


Fig. 7 Residual plots and distribution of residuals for 3 different machine learning models used to predict runoff

The calculation of feature significance is applicable to tree-based models like as Decision Tree, Random Forest, Gradient Boosting, XGBoost, and AdaBoost, as well as linear models like Linear Regression. On the other hand, it is not possible to compute it for models such as the Support Vector Regressor, the MLP Regressor, or the Stacking Regressor. These models operate within a high-dimensional or modified feature space, where the applied transformations to the input data pose challenges in directly determining the significance of individual characteristics.

Expanding on the insights gained from Table 5, the analysis continues with, showcasing a heatmap of correlation coefficients for the Wave Runup dataset.

Based on the analysis of the heatmap, a consistent pattern results wherein the three most significant parameters in the dataset exhibit a strong alignment with the feature importance rankings derived from the XGBoost Regressor, Gradient Boosting Regressor, and Random Forests. However, they provide different types of information.

The feature importance method evaluates the individual significance of each feature in predicting the target variable. The features are ranked according to their respective contributions to the predictive performance of the model. XGBoost Regressor, Gradient Boosting Regressor, and Random Forests algorithms all offer feature importance metrics. These measures can aid in determining the most influential features in the model's predictions.

The correlation heatmap technique is employed to examine the pairwise correlations between the features and the target variable. A correlation heatmap is a graphical representation that utilises color-coded cells to depict correlations, thereby offering a concise summary of the associations between variables. The utilisation of a heatmap facilitates the identification of features that exhibit a robust linear correlation with the target variable.

Both methodologies possess fundamental advantages and are capable of yielding valuable insights from the dataset. The assessment of feature importance aids in comprehending the relative significance of features

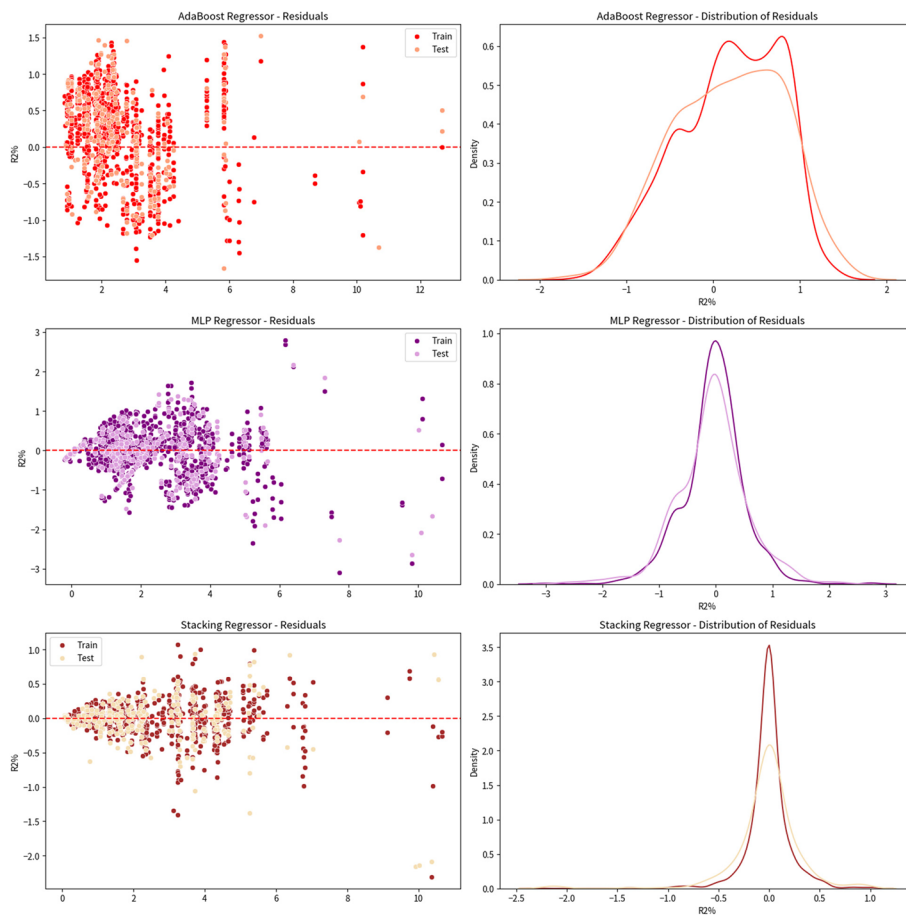


Fig. 8 Residual plots and distribution of residuals for 3 different machine learning models used to predict runup

Table 5 Feature importance

Model	Set	Stt	Sinc	Sig	Hs
Decision Tree	0.0172	0.0688	0.0723	0.0822	0.4474
Random Forest	0.0168	0.0598	0.0709	0.0716	0.4686
Gradient Boosting Regressor	0.0243	0.0728	0.0716	0.0692	0.4445
XGBoost Regressor	0.0211	0.0754	0.0697	0.0720	0.4464
AdaBoost Regressor	0.0142	0.0689	0.0701	0.0706	0.4721

within a model, whereas a correlation heatmap facilitates the identification of linear associations between features and the target variable. Depending on the specific objectives and characteristics of the data, researchers have the option to employ one or both of these methodologies in order to enhance their comprehension of the regression problem at hand. Fig. 9

The interrelationships between the characteristics of the wave runup dataset are shown graphically in this heatmap. The correlation coefficient between two variables is shown in each cell as the combination of the

colour intensity and the number. A correlation coefficient is a statistical metric used to determine how strongly two variables move in tandem. The numbers might go from negative one to positive one.

When the value of the correlation coefficient is -1.0 , it indicates a perfect negative correlation, and when it is 1.0 , it indicates a perfect positive correlation. There is no linear link between the two variables' movements if the correlation is 0.0 .

The heatmap displays correlation coefficients, with darker colours indicating higher absolute values and thus a stronger relationship.

Based on the analysis of the heatmap, it is evident that:

- The variables R2% and Stt (total swash excursion) exhibit a highly significant positive correlation, with a coefficient of 0.905 . As the magnitude of the swash excursion increases, there is a tendency for the 2% exceedance value of runup to also increase.
- The variables R2% and Set (setup) exhibit a significant positive correlation with a coefficient of 0.873 . There is a positive correlation between higher setup values and higher 2% exceedance values for runup.

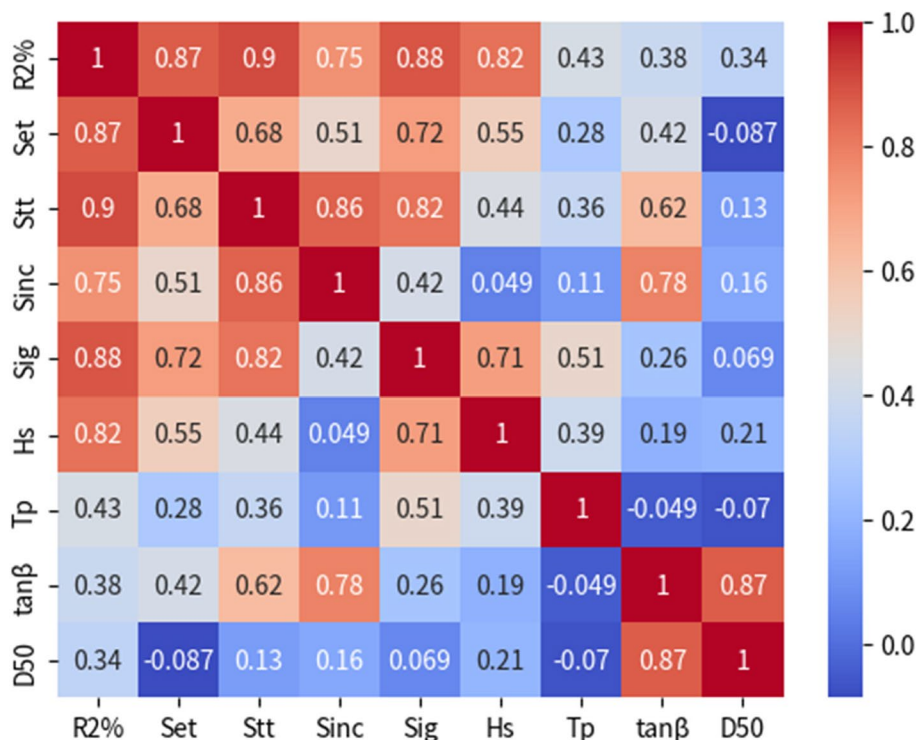


Fig. 9 Heatmap of Correlation Coefficients for Wave Runup Dataset

- The analysis reveals a robust positive correlation of 0.884 between R2% and Sig (infragravity swash). As the magnitude of the infragravity swash intensifies, there is a corresponding tendency for the 2% exceedance value of runup to increase.

- The variables R2% and Hs (significant deep-water wave height) exhibit a robust positive correlation with a coefficient of 0.822. There is a positive correlation between elevated deep-water wave heights and increased 2% exceedance values for runup.

- The relationship between R2% and Sinc (incident swash) exhibits a statistically significant positive correlation with a coefficient of 0.745, indicating a moderate to strong association. As the magnitude of the incident swash increases, there is a tendency for the 2% exceedance value for runup to also increase, albeit not as significantly as observed with other variables.

- The variables R2% and Tp (peak wave period) exhibit a moderate positive correlation with a coefficient of 0.427. There exists a moderate correlation between higher peak wave periods and higher 2% exceedance values for runup, although the strength of this relationship is relatively weaker.

- The correlation coefficient between R2% and tanβ (foreshore beach slope) indicates a moderate positive relationship, with a value of 0.382. As the slope of the foreshore beach increases, there is a tendency for the 2% exceedance

value of runup to also increase, although the correlation between these two variables is not particularly robust.

- The correlation between R2% and D50 (Median sediment size) is moderately positive, with a coefficient of 0.343. There is a tendency for the 2% exceedance value for runup to increase as the median sediment size increases, although the strength of this relationship is not particularly robust.

13 Discussion

13.1 Interpretation of the findings and their implications for wave runup prediction

The study yielded several important findings with implications for wave runup prediction:

13.2 Performance of machine learning models

The XGBoost and Stacking models exhibited superior performance on the given dataset, as evidenced by their success in achieving of the lowest Root Mean Squared Error (RMSE) values. This implies that ensemble methods, which combine predictions from multiple models, exhibit notable efficacy for this particular task. These models is likely to accurately reflect the complex structure of relationships between the data’s properties.

13.3 Importance of non-linear models

The notable efficacy exhibited by models such as XGBoost, Stacking, and Random Forest, which possess

the ability to capture non-linear relationships, suggests that the connection between the features and the target variable (wave runup) is likely to be non-linear. This observation holds significant implications for the future development of wave runup modelling.

13.4 Limitations of linear models

The linear regression and support vector regression models, specifically those employing a linear kernel, exhibited comparatively greater root mean square error (RMSE) values. This indicates that these models were less successful in accurately representing the intricate nature of the dataset when compared to the non-linear models. This finding underscores the significance of taking into account non-linear models when addressing this task.

13.5 Implications for wave runup prediction

The findings indicate that the utilisation of machine learning models holds promise in accurately forecasting wave runup, thus carrying significant implications for the field of coastal engineering and management. Accurate prediction of wave runup holds significant potential in facilitating the design of coastal structures and evaluating the risks associated with coastal hazards (Doğan and Durap 2017; Durap et al. 2023; Durap and Balas 2022).

13.6 Implications for coastal management

Accurate prediction of wave runup holds significant importance in the field of coastal management. Wave runup has the potential to induce adverse effects such as beach erosion and flooding, thereby exerting significant influence on coastal infrastructure and ecosystems. The results of this study indicate that the utilisation of machine learning models holds significant potential in the accurate prediction of wave runup, thereby providing valuable insights for informing decisions related to coastal management.

13.7 Future research directions

There is definitely space for improvement, even if the research showed the promise of machine learning for wave runup prediction. Future studies should examine further machine learning models, make use of bigger and more varied datasets, and look at additional variables that could be crucial for forecasting wave runup. The utilisation of feature engineering techniques to generate novel and informative features, coupled with the tuning of model hyperparameters, can be employed to enhance the performance of the model. Furthermore, it is worth considering the exploration of alternative machine learning models and deep learning approaches.

The research concluded by demonstrating that wave runup may be accurately predicted by machine learning

methods, with implications for coastal engineering and management. But further investigation is required to improve the precision of these forecasts and examine the possibility of additional machine learning methods.

14 Comparison of the algorithms' performance and suitability for practical applications

The present study involved the evaluation of multiple machine learning models with the aim of assessing their predictive capabilities in determining wave runup, a critical parameter in the field of coastal engineering. The following analysis presents a comparative assessment of their performance and applicability in practical contexts:

14.1 Linear regression

In contrast to its straightforwardness and efficiency, the efficacy of Linear Regression on this particular dataset was comparatively suboptimal when compared to the alternative models. The assumption of a linear relationship between features and the target variable may not be applicable in various real-world scenarios, including the present one. Hence, although it can function as a reliable reference model, its suitability for practical scenarios pertaining to wave runup prediction may be limited.

14.2 Decision tree

The performance of the Decision Tree model was superior to that of Linear Regression, although it was surpassed by certain other models. Although Decision Trees possess the advantage of being easily interpretable, they are susceptible to overfitting the training data, resulting in inadequate generalisation to unfamiliar data. Hence, it is imperative to note that these options may not be optimal for practical implementations in the absence of appropriate calibration and verification processes.

14.3 Random forest

The Random Forest model exhibited strong performance on the given dataset, thereby showcasing the efficacy of ensemble methods. Random Forests exhibit a reduced susceptibility to overfitting compared to individual Decision Trees, rendering them capable of effectively managing intricate relationships within the dataset. Nevertheless, the training and prediction processes of these models may exhibit sluggishness, particularly when confronted with extensive datasets. This characteristic could potentially impede their efficacy in practical scenarios that necessitate prompt and real-time predictions.

14.4 Support vector regression (SVR)

The support vector regression (SVR) model exhibited the poorest performance on the given dataset. Although Support Vector Regression (SVR) has the

capability to handle non-linear relationships by utilising appropriate kernels, its performance tends to be sluggish when dealing with large datasets. Additionally, the task of selecting the optimal kernel and parameter values for SVR can pose a significant challenge. Hence, it may not be the most appropriate for practical applications pertaining to the prediction of wave runup.

14.5 Gradient boosting

The performance of the Gradient Boosting model on this dataset was satisfactory. Similar to the Random Forest algorithm, this ensemble method is capable of effectively addressing intricate relationships within the dataset. Nevertheless, the training process of this model may be characterised by a slow pace and a susceptibility to overfitting if appropriate regularisation techniques are not implemented. Hence, meticulous calibration and verification would be imperative for pragmatic implementations.

14.6 XGBoost

The XGBoost model exhibited superior performance on the given dataset, alongside the Stacking model. XGBoost is widely recognised for its exceptional performance across various machine learning tasks, owing to its built-in regularisation capabilities that effectively mitigate the risk of overfitting. However, similar to Gradient Boosting, the training process of this method can be time-consuming and necessitates meticulous parameter tuning. Notwithstanding these challenges, the commendable performance of the subject in question renders it a favourable option for pragmatic implementations.

14.7 AdaBoost regressors

The performance of the AdaBoost model was superior to that of Linear Regression and Support Vector Regression (SVR), although it was surpassed by certain alternative models. Although AdaBoost is relatively straightforward to implement and often yields favourable results, it is susceptible to the influence of noisy data and outliers. Hence, it may not be the most appropriate for practical applications in the absence of adequate data preprocessing and model tuning.

14.8 MLP regressor

The MLP model exhibited superior performance in comparison to Linear Regression and SVR, although it was surpassed by certain alternative models. Although Multilayer Perceptrons (MLPs) have the capability to handle intricate and non-linear relationships, their effectiveness is contingent upon meticulous parameter tuning and they often exhibit slow training speeds. Hence, it is plausible that these models may not be optimally suited for

practical applications in the absence of adequate computational resources and expertise in model calibration.

14.9 Stacking regressor

The Stacking model exhibited superior performance on the given dataset, alongside the XGBoost model. Stacking, as an ensemble technique, effectively leverages the collective capabilities of multiple models. Nevertheless, the training process of this model tends to be sluggish, necessitating meticulous selection of both the initial and ultimate models. Additionally, its interpretability falls short when compared to more straightforward models. Notwithstanding these challenges, the commendable performance of the subject in question renders it a favourable option for pragmatic implementations.

In summary, when considering practical applications related to wave runup prediction, it is evident that ensemble methods such as XGBoost and Stacking exhibit notable suitability owing to their exceptional performance. Nevertheless, the selection of a model is contingent upon additional considerations, including the computational resources at hand, the requirement for interpretability of the model, and the level of tolerance for both model training and prediction time.

Considering the presented results, it appears that the machine learning models, especially the ensemble ones, have performed well in predicting wave runup. However, we suggested that The XGBoost and Stacking models exhibited superior performance in the manuscript which are all ensemble learning methods.

15 Identification of key factors influencing the predictive accuracy of the algorithms

The study identified several crucial factors that exerted an influence on the predictive accuracy of the algorithms:

15.1 Quality and quantity of data

The critical factor in the training and testing of models is the quality and quantity of the data utilised. The utilisation of high-quality data containing a substantial number of instances has the potential to enhance the learning capabilities of models and improve their ability to generalise when presented with novel data. The present study utilised a comprehensive dataset that encompassed a diverse range of features pertaining to wave and beach characteristics.

15.2 Feature selection

The predictive accuracy of the models was significantly influenced by the choice of input features utilised for wave runup prediction. The present study identified the median sediment size, significant deep-water wave

height, and foreshore beach slope as the primary factors of significance. Enhancing the predictive accuracy of models can be achieved by incorporating relevant characteristics that have a substantial influence on the output.

15.3 Model complexity

The predictive accuracy of the model can also be influenced by its complexity. The utilisation of more sophisticated models, such as XGBoost and Stacking, facilitated the identification and comprehension of intricate patterns within the dataset, ultimately resulting in the attainment of the highest level of predictive accuracy. Nevertheless, it is important to note that excessively intricate models can also suffer from overfitting, a phenomenon wherein the model becomes too closely tailored to the training data, resulting in subpar performance when applied to new, unseen data.

15.4 Hyperparameter tuning

The performance of algorithms can be significantly influenced by their settings or hyperparameters. In this particular instance, default hyperparameters were employed for all models. The performance of the models could potentially be enhanced through hyperparameter tuning.

15.5 Model assumptions

The predictive accuracy of the model can also be influenced by the assumptions it makes about the underlying data. An instance of concern is that Linear Regression assumes a linear correlation between the independent variables and the dependent variable, which might not be valid in the given dataset, thereby resulting in suboptimal predictive accuracy.

15.6 Model interpretability

While not directly influencing predictive accuracy, the interpretability of the models is an important factor to consider. More interpretable models can provide insights into the relationships between the input features and the output, which can be valuable in understanding and improving the predictive accuracy of the models.

15.7 Training and validation strategy

The choice of training and validation strategy can have an impact on the predictive accuracy of the models. In this scenario, a basic train-test split methodology was employed. Alternative approaches, such as cross-validation, have the potential to yield a more reliable assessment of the performance of the models.

The graph (Fig. 10) illustrates the level of interpretability associated with each model, the interpretability score ranges from 1 to 5, with 5 representing the highest level of interpretability. Linear regression and decision trees are generally considered to be more interpretable models, as they provide clear insights into the relationship between the input variables and the target variable. On the other hand, multilayer perceptron (MLP) and stacking models tend to be less interpretable, as they involve complex computations and interactions between variables that are not easily understandable. The preference for Linear Regression and Decision Tree models stems from their simplicity and the ease with which their decision-making processes can be comprehended. In contrast, MLP and Stacking models are characterised by their intricate nature, as they integrate multiple models or layers, thereby rendering their decision-making process more challenging to comprehend.

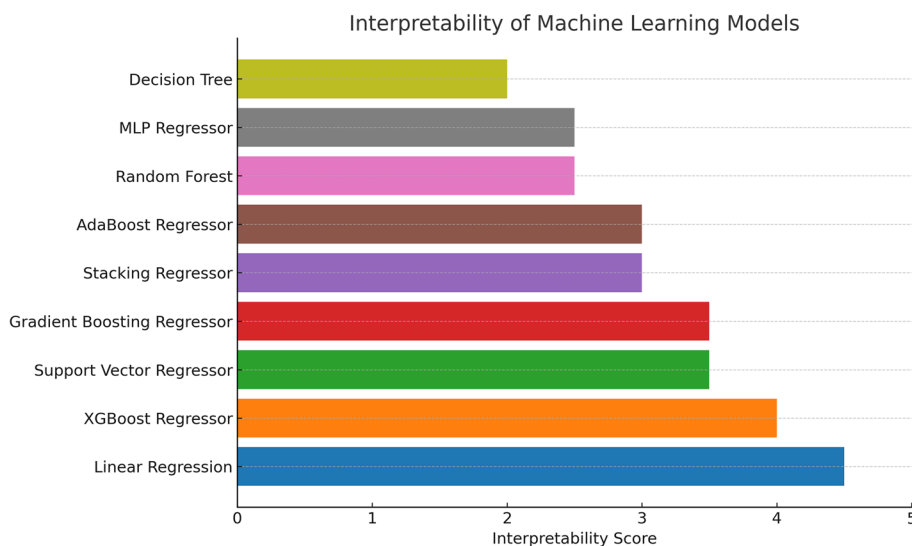


Fig. 10 Interpretability of Machine Learning Models for this particular dataset

In summary, the efficacy of machine learning models in forecasting wave runup was found to be impacted by various factors, encompassing the adequacy and abundance of data, the selection of relevant features, the complexity of the model, the fine-tuning of parameters, and the interpretability of the model.

16 Conclusion and future work

The research conducted in this study focused on the utilisation of different machine learning algorithms to predict wave runup. The dataset employed in this analysis consisted of multiple features, including the 2% exceedance value for runup, setup, total swash excursion, incident swash, infra-gravity swash, significant deep-water wave height, peak wave period, foreshore beach slope, and median sediment size. The models employed in this study encompassed a range of machine learning algorithms, namely Linear Regression, Decision Tree, Random Forest, Support Vector Regression, Gradient Boosting, XGBoost, AdaBoost, MLP, and Stacking.

The models exhibited varying levels of performance, with XGBoost and Stacking models demonstrating superior performance as indicated by the RMSE, MAE, and R-squared metrics. This implies that ensemble methods, which amalgamate predictions from multiple models, exhibit notable efficacy for this undertaking. Complex linkages and interactions between characteristics in the dataset may be captured by these models.

However, the RMSE values for Linear Regression and Support Vector Regression (with a linear kernel) were rather high, indicating that these models did not adequately represent the complexity of the dataset. Because of this, it's even more crucial to think about using non-linear models for this task.

The findings have significant implications for coastal engineering and management, suggesting that machine learning methods may be used to efficiently predict wave runup. The capacity to reliably forecast wave runup is useful for assessing coastal flooding threats and improving the design of coastal infrastructure.

The research also emphasised the significance of employing non-linear models, including XGBoost, Stacking, and Random Forest. The robust performance exhibited by the data implies that there is a probable non-linear association between the features and the target variable, specifically wave runup. This knowledge is essential for next wave runup prediction modelling attempts.

In contrast, linear models such as Linear Regression and Support Vector Regression with a linear kernel exhibited relatively high RMSE values, suggesting their limited ability to comprehend the complexities present in the dataset. This highlights the importance of incorporating non-linear models in order to achieve precise wave runup prediction.

The investigation also examined the significance of various characteristics in forecasting wave runup. The most crucial features identified in the study were the median sediment size, significant deep-water wave height, and foreshore beach slope. This observation has the potential to inform and direct future endeavours in data collection and model development. Subsequent investigations may delve into additional features that hold potential relevance, alongside exploring novel techniques for enhancing feature engineering and selection processes. These endeavours aim to enhance the predictive accuracy of the models.

Enhancing model performance by capturing complicated connections in the data is possible via feature engineering techniques including generating new features from existing ones. Using hyperparameter tuning strategies like grid search and random search, the models' performance might be further enhanced. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are two examples of machine learning models and deep learning methodologies that may provide light on the modelling of temporal relationships in the data. SHAP (SHapley Additive exPlanations) is a method that may be used to better explain the decision-making process of complicated models like XGBoost and Stacking. Optimising code, utilising more efficient methods, or using hardware accelerators like GPUs might increase the training and prediction performance of the models for realistic applications demanding real-time predictions.

While certain models, like Decision Trees, are easily interpretable, others, like neural networks, are typically seen as "black boxes" and need to be made more interpretable. These models may be interpreted, and their predictions gained knowledge from using methods like SHAP (SHapley Additive exPlanations). The interpretability of machine learning models is an area that might benefit from more study.

With more variables, machine learning models' complexity increases. The performance of the model can be enhanced through the incorporation of informative and relevant variables to effect wave runup. However, if the variables are irrelevant or introduce noise, the model may suffer from overfitting. In the context of wave runup analysis, this study demonstrates exceptional performance in machine learning models compared to existing literature due to the inclusion of a more extensive range of relevant variables for wave runup.

The complexity of the model rises as the number of variables grows. The model is enhanced if the new variables are informative and useful. However, overfitting may occur if the extra variables are irrelevant or introduce noise into the model. Since dataset include relevant variables in terms of wave runup, all machine learning

models show high performance compared to literature as they do not include as much as variable this study.

The findings of this study demonstrate the efficacy of machine learning algorithms in predicting wave runup, thereby offering significant implications for the fields of coastal engineering and management. However, it is imperative to conduct additional research in order to further enhance these models and investigate the possibilities of alternative machine learning methodologies.

Finally, these models might be applied to real-world challenges in coastal engineering, such managing beach erosion, developing coastal infrastructure, and assessing flood risk, in the future. Creating accessible software to run the models and guide the choices of coastal engineers would be a good first step in this direction.

Having summarized the significant advancements made in the field of coastal engineering through this study, it is now essential to highlight the specific contributions and findings, which can be summarized as follows:

16.1 Application of machine learning to wave runup prediction

This study provides a significant contribution to the field by focusing on the practical application of various machine learning models in predicting wave runup, an essential parameter in the field of coastal engineering.

16.2 Comparison of machine learning models

This research paper provides a comprehensive assessment of nine different machine learning models, analysing their predictive precision, interpretability, and suitability for real-world applications. This resource possesses substantial significance as a point of reference for researchers and practitioners working within the relevant field.

16.3 Insights into factors influencing model performance

This research provides significant insights into the key factors that influence the predictive precision of models. The quality of data, the process of feature selection, the complexity of the model, the tuning of hyperparameters, the assumptions made by the model, and the strategy employed for training and validation are all factors that need to be considered. This can serve as a theoretical framework for future investigations in these particular academic settings.

16.4 Implications for coastal engineering and management

The study findings have substantial implications for the field of coastal engineering and management. The precise anticipation of wave runup possesses considerable potential in facilitating the development of coastal structures and the assessment of risks linked to coastal hazards such as flooding and erosion.

Acknowledgements

I would like to extend my sincere appreciation to the researchers whose experiments served as the foundation for this study. The combination of data gathered from both beaches and laboratories has allowed for a comprehensive analysis and a deeper understanding of the subject matter. The contributions of these researchers have been duly recognized, and proper credit has been given through meticulous citation, as presented in Table 2.

Author's contributions

Ahmet Durap.

Funding

Authors state no funding involved.

Availability of data and materials

This paper utilized experiments carried out both in beaches and labs by researchers. Papers and authors are cited as in shown in Table 2.

Declarations

Competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author details

¹Division of Civil Engineering, The University of Queensland, Brisbane 4072, Australia. ²Division of Coastal Sciences and Engineering, Civil Engineering Department, Civil Engineering Faculty, Istanbul Technical University, Istanbul 34467, Turkey.

Received: 3 August 2023 Revised: 31 October 2023 Accepted: 10 November 2023

Published online: 18 December 2023

References

- Abdalazeez A, Didenkulova I, Dutykh D, Labart C (2020) Extreme inundation statistics on a Composite Beach. *Water (Basel)* 12:1573. <https://doi.org/10.3390/w12061573>
- Alfonsi G (2015) Numerical simulations of wave-induced flow fields around large-diameter surface-piercing vertical circular cylinder. *Computation* 3:386–426. <https://doi.org/10.3390/computation3030386>
- Athanasiou P, van Dongeren A, Giardino A, Voudoukas M, Antolinez JAA, Ranasinghe R (2021) A clustering approach for predicting dune Morphodynamic response to storms using typological coastal profiles: a case study at the Dutch coast. *Front Mar Sci* 8. <https://doi.org/10.3389/fmars.2021.747754>
- Atkinson AL, Power HE, Moura T, Hammond T, Callaghan DP, Baldock TE (2017) Assessment of runup predictions by empirical models on non-truncated beaches on the south-east Australian coast. *Coast Eng* 119:15–31. <https://doi.org/10.1016/j.coastaleng.2016.10.001>
- Baldock TE, Huntley DA (2002) Long-wave forcing by the breaking of random gravity waves on a beach. *Proceedings of the Royal Society of London. Series A: Mathematical Phys Eng Sci* 458:2177–2201. <https://doi.org/10.1098/rspa.2002.0962>
- Beuzen T, Goldstein EB, Splinter KD (2019) Ensemble models from machine learning: an example of wave runup and coastal dune erosion. *Nat Hazards Earth Syst Sci* 19:2295–2309. <https://doi.org/10.5194/nhess-19-2295-2019>
- Doğan Y, Durap A (2017) Summarizing data sets for data mining by using statistical methods in coastal engineering. *World Acad Sci Eng Technol Int J Comput Inform Eng* 11:643–648
- Durap A, Balas CE (2022) Risk assessment of submarine pipelines: a case study in Turkey. *Ocean Eng* 261:112079. <https://doi.org/10.1016/j.oceaneng.2022.112079>
- Durap A, Balas CE, Çökçör Ş, Balas EA (2023) An integrated Bayesian risk model for coastal flow slides using 3-D hydrodynamic transport and Monte Carlo simulation. *J Mar Sci Eng* 11:943. <https://doi.org/10.3390/jmse11050943>

- Fang Y, Cui Y, Ao X (2019) Deep learning for automatic recognition of magnetic type in sunspot groups. *Adv Astron* 2019:1–10. <https://doi.org/10.1155/2019/9196234>
- Gallagher EL, Elgar S, Guza RT (1998) Observations of sand bar evolution on a natural beach. *J Geophys Res Oceans* 103:3203–3215. <https://doi.org/10.1029/97JC02765>
- Gomes da Silva P, Medina R, González M, Garnier R (2018) Infragravity swash parameterization on beaches: the role of the profile shape and the morphodynamic beach state. *Coast Eng* 136:41–55. <https://doi.org/10.1016/j.coastaleng.2018.02.002>
- Guedes RMC, Bryan KR, Coco G (2012) Observations of alongshore variability of swash motions on an intermediate beach. *Cont Shelf Res* 48:61–74. <https://doi.org/10.1016/j.csr.2012.08.022>
- Guedes RMC, Bryan KR, Coco G (2013) Observations of wave energy fluxes and swash motions on a low-sloping, dissipative beach. *J Geophys Res Oceans* 118:3651–3669. <https://doi.org/10.1002/jgrc.20267>
- Guedes RMC, Bryan KR, Coco G, Holman RA (2011) The effects of tides on swash statistics on an intermediate beach. *J Geophys Res* 116:C04008. <https://doi.org/10.1029/2010JC006660>
- Ha T, Yoo J, Han S, Cho Y-S (2014) Numerical study on tsunami Hazard mitigation using a submerged breakwater. *Sci World J* 2014:1–11. <https://doi.org/10.1155/2014/863202>
- Holland KT, Raubenheimer B, Guza RT, Holman RA (1995) Runup kinematics on a natural beach. *J Geophys Res* 100:4985. <https://doi.org/10.1029/94JC02664>
- Howe D (2016) Bed shear stress under wave runup on steep slopes (Phd). University of South Wales, Sydney
- Hughes SA (2004) Estimation of wave run-up on smooth, impermeable slopes using the wave momentum flux parameter. *Coast Eng* 51:1085–1104. <https://doi.org/10.1016/j.coastaleng.2004.07.026>
- Lecacheux S, Pedreros R, Le Cozannet G, Thiébot J, De La Torre Y, Bulteau T (2012) A method to characterize the different extreme waves for islands exposed to various wave regimes: a case study devoted to Reunion Island. *Nat Hazards Earth Syst Sci* 12:2425–2437. <https://doi.org/10.5194/nhess-12-2425-2012>
- Li N, Yamazaki Y, Roerber V, Cheung KF, Chock G (2018) Probabilistic mapping of storm-induced coastal inundation for climate change adaptation. *Coast Eng* 133:126–141. <https://doi.org/10.1016/j.coastaleng.2017.12.013>
- Mase H (1989) Random wave Runup height on gentle slope. *J Waterw Port Coast Ocean Eng* 115:649–661. [https://doi.org/10.1061/\(ASCE\)0733-950X\(1989\)115:5\(649\)](https://doi.org/10.1061/(ASCE)0733-950X(1989)115:5(649))
- Mason C, Sallenger AH, Holman RA, Birkemeier WA (1984) DUCK82 - a coastal storm processes experiment. *Coastal Eng Proceed* 1:128. <https://doi.org/10.9753/icce.v19.128>
- Matsuba Y, Shimozono T, Sato S (2020) Wave-breaking modulation by infragravity waves during an extreme typhoon. *PLoS One* 15:e0231242. <https://doi.org/10.1371/journal.pone.0231242>
- Milanian F, Niri MZ, Najafi-Jilani A (2017) Effect of hydraulic and structural parameters on the wave run-up over the berm breakwaters. *Int J Naval Architect Ocean Eng* 9:282–291. <https://doi.org/10.1016/j.jnnaoe.2016.10.001>
- Muttray M, Oumeraci H, ten Oever E (2007) WAVE REFLECTION AND WAVE RUN-UP AT RUBBLE MOUND BREAKWATERS, in: coastal engineering 2006. World scientific publishing company:4314–4324. https://doi.org/10.1142/9789812709554_0362
- Najafi-Jilani A, Niri MZ, Naderi N (2014) Simulating three dimensional wave run-up over breakwaters covered by antifer units. *Int J Naval Architect Ocean Eng* 6:297–306. <https://doi.org/10.2478/IJNAOE-2013-0180>
- Nicolae Lerma A, Pedreros R, Robinet A, Sénéchal N (2017) Simulating wave setup and runup during storm conditions on a complex barred beach. *Coast Eng* 123:29–41. <https://doi.org/10.1016/j.coastaleng.2017.01.011>
- Park H, Cox DT (2016) Empirical wave run-up formula for wave, storm surge and berm width. *Coast Eng* 115:67–78. <https://doi.org/10.1016/j.coastaleng.2015.10.006>
- Penalba M, Ringwood JV (2020) Systematic complexity reduction of wave-to-wire models for wave energy system design. *Ocean Eng* 217:107651. <https://doi.org/10.1016/j.oceaneng.2020.107651>
- Poate TG, McCall RT, Masselink G (2016) A new parameterisation for runup on gravel beaches. *Coast Eng* 117:176–190. <https://doi.org/10.1016/j.coastaleng.2016.08.003>
- Power HE, Gharabaghi B, Bonakdari H, Robertson B, Atkinson AL, Baldock TE (2019) Prediction of wave runup on beaches using gene-expression programming and empirical relationships. *Coast Eng* 144:47–61. <https://doi.org/10.1016/j.coastaleng.2018.10.006>
- Raubenheimer B, Guza RT (1996) Observations and predictions of run-up. *J Geophys Res Oceans* 101:25575–25587. <https://doi.org/10.1029/96JC02432>
- Ruessink BG, Kleinhans MG, van den Beukel PGL (1998) Observations of swash under highly dissipative conditions. *J Geophys Res Oceans* 103:3111–3118. <https://doi.org/10.1029/97JC02791>
- Ruggiero P (2004) Wave run-up on a high-energy dissipative beach. *J Geophys Res* 109:C06025. <https://doi.org/10.1029/2003JC002160>
- Sarasa-Cabezuelo A (2022) Prediction of rainfall in Australia using machine learning. *Information* 13:163. <https://doi.org/10.3390/info13040163>
- Senechal N, Coco G, Bryan KR, Holman RA (2011) Wave runup during extreme storm conditions. *J Geophys Res Oceans* 116:2010JC006819. <https://doi.org/10.1029/2010JC006819>
- Shimozono T, Tajima Y, Kumagai K, Arikawa T, Oda Y, Shigihara Y, Mori N, Suzuki T (2020) Coastal impacts of super typhoon Hagibis on greater Tokyo and Shizuoka areas, Japan. *Coast Eng J* 62:129–145. <https://doi.org/10.1080/21664250.2020.1744212>
- Stauble DK, Cialone MA (1997) Sediment dynamics and profile interactions: DUCK94. In: Coastal engineering 1996. American Society of Civil Engineers, New York NY, pp 3921–3934. <https://doi.org/10.1061/9780784402429.303>
- Stockdon HF, Holman RA, Howd PA, Sallenger AH (2006) Empirical parameterization of setup, swash, and runup. *Coast Eng* 53:573–588. <https://doi.org/10.1016/j.coastaleng.2005.12.005>
- Stockdon HF, Thompson DM, Plant NG, Long JW (2014) Evaluation of wave runup predictions from numerical and parametric models. *Coast Eng* 92:1–11. <https://doi.org/10.1016/j.coastaleng.2014.06.004>
- Tebaldi C, Strauss BH, Zervas CE (2012) Modelling Sea level rise impacts on storm surges along US coasts. *Environ Res Lett* 7:014032. <https://doi.org/10.1088/1748-9326/7/1/014032>
- Thornton EB, Humiston RT, Birkemeier W (1996) Bar/trough generation on a natural beach. *J Geophys Res Oceans* 101:12097–12110. <https://doi.org/10.1029/96JC00209>
- Torresan S, Critto A, Rizzi J, Marcomini A (2012) Assessment of coastal vulnerability to climate change hazards at the regional scale: the case study of the North Adriatic Sea. *Nat Hazards Earth Syst Sci* 12:2347–2368. <https://doi.org/10.5194/nhess-12-2347-2012>
- van Ormondt M, Roelvink D, van Dongeren A (2021) A model-derived empirical formulation for wave run-up on naturally sloping beaches. *J Mar Sci Eng* 9:1185. <https://doi.org/10.3390/jmse9111185>
- Vitousek S, Barnard PL, Fletcher CH, Frazer N, Erikson L, Storlazzi CD (2017) Doubling of coastal flooding frequency within decades due to sea-level rise. *Sci Rep* 7:1399. <https://doi.org/10.1038/s41598-017-01362-7>