

Wave overtopping predictions using an advanced machine learning technique

den Bieman, Joost P.; van Gent, Marcel R.A.; van den Boogaard, Henk F.P.

DOI

[10.1016/j.coastaleng.2020.103830](https://doi.org/10.1016/j.coastaleng.2020.103830)

Publication date

2021

Document Version

Accepted author manuscript

Published in

Coastal Engineering

Citation (APA)

den Bieman, J. P., van Gent, M. R. A., & van den Boogaard, H. F. P. (2021). Wave overtopping predictions using an advanced machine learning technique. *Coastal Engineering*, 166, 1-12. Article 103830. <https://doi.org/10.1016/j.coastaleng.2020.103830>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Wave overtopping predictions using an advanced machine learning technique

Joost P. den Bieman^{a,*}, Marcel R. A. van Gent^{a,b}, Henk F. P. van den Boogaard^a

^a*Deltares, Department of Coastal Structures and Waves (JdB & MvG) & Deltares Software Centre (HvdB), Boussinesqweg 1, 2629HV Delft, The Netherlands.*

^b*TU Delft, Department of Hydraulic Engineering, Stevinweg 1, 2628CN Delft, The Netherlands.*

Abstract

Coastal structures are often designed to a maximum allowable wave overtopping discharge, hence accurate prediction of the amount of wave overtopping is an important issue. Both empirical formulae and neural networks are among the commonly used prediction tools. In this work, a new model for the prediction of mean wave overtopping discharge is presented using the innovative machine learning technique XGBoost. The selection of features to train the model on is carefully substantiated, including the redefinition of existing features to obtain a better model performance. Confidence intervals are derived by tuning hyperparameters and applying bootstrap resampling. The quality of the model is tested against four new physical model data sets, and a thorough quantitative comparison with existing machine learning methods and empirical overtopping formulae is presented. The XGBoost model generally outperforms other methods for the test data sets with normally incident waves. All data-driven methods show less accuracy on oblique wave data, presumably because these conditions are underrepresented in the training data. The performance of the XGBoost model is significantly improved by adding a randomly selected part of the new oblique wave cases to the training data. In the end, this new model is shown to reduce errors on all data used in this work with a factor of up to 5 compared to existing overtopping prediction methods.

Keywords: Machine learning; Wave overtopping; Coastal structures; Physical

*Corresponding author

Email addresses: `joost.denbieman@deltares.nl` (Joost P. den Bieman),
`marcel.vangent@deltares.nl` (Marcel R. A. van Gent),
`henk.vandenboogaard@deltares.nl` (Henk F. P. van den Boogaard)

Preprint submitted to Coastal Engineering

December 11, 2020

1 **1. Introduction**

2 Wave overtopping has the potential to interfere with the function of a coastal
3 structure and cause structural damage or physical harm. To reduce these risks,
4 coastal structures are often designed to prevent exceeding a maximum allow-
5 able wave overtopping discharge. Therefore, estimates of the amount of wave
6 overtopping are important for the design of coastal structures.

7 Currently, different types of tools are available to predict the expected amount
8 of wave overtopping, given a certain configuration of a coastal structure. Firstly,
9 many empirical overtopping formulae have been derived from physical model
10 data. These form a relatively easy estimate of the mean wave overtopping dis-
11 charge, q [$\text{m}^3/\text{s}/\text{m}$]. A selection of those formulae are listed in TAW (2002) and
12 in the EurOtop manual (EurOtop, 2018). The so-called CLASH database (Steen-
13 dam et al., 2004) with wave overtopping data from measurements has been
14 used by Van Gent et al. (2007) as training data for a neural network (NN) to pre-
15 dict wave overtopping. Their ensemble of NNs outputs both the expected mean
16 wave overtopping discharge and an estimate for the corresponding uncertainty.
17 A similar approach was used while extending both the training data set and
18 adding predictions of wave transmission and reflection (Zanuttigh et al., 2016).
19 Recently, it was shown in Den Bieman et al. (2020) that the machine learning
20 method XGBoost (Chen & Guestrin, 2016) can be successfully applied as an al-
21 ternative to NN models. XGBoost is a relatively new method, finding success
22 in various practical applications from fault detection in wind turbines (Zhang
23 et al., 2018) to bridge damage estimation (Lim & Chi, 2019). Applying the method
24 to the prediction of wave overtopping significantly reduces the prediction er-
25 rors on the CLASH database compared to the NN by Van Gent et al. (2007),
26 see Den Bieman et al. (2020). In addition to empirical formulae and machine
27 learning methods, numerical models are capable of reproducing physical wave
28 overtopping models reasonably well. Hence, numerical modelling could also
29 be used to predict mean wave overtopping discharge, on the condition that ex-
30 tensive calibration and validation on physical model data has been carried out.

31 The exploratory work in Den Bieman et al. (2020) compares the existing NN
32 model by Van Gent et al. (2007) to an XGBoost model with a similar setup that
33 is trained on the same training data set. The XGBoost method is shown to out-
34 perform the NN, reducing errors by a factor of 2.8. In this paper, that work

35 is expanded upon in several ways to get to a state-of-the art XGBoost model
36 for the prediction of mean wave overtopping discharges. Firstly, the training
37 database is enlarged beyond the original CLASH database and the selection of
38 features for model training is carefully substantiated. Secondly, both the hy-
39 perparameter tuning and derivation of uncertainties is readdressed, as Den
40 Bieman et al. (2020) find surprisingly small confidence intervals. Thirdly, the
41 XGBoost model is validated on both the overtopping database and new physi-
42 cal model data previously unseen by the model. Finally, the model is compared
43 with predictions from two of the available neural network models (Van Gent
44 et al., 2007; Zanuttigh et al., 2016) and from two empirical overtopping formu-
45 lae (TAW, 2002; EurOtop, 2018).

46 This article is structured as follows. Section 2 contains the description of
47 the machine learning methods and the training and test data sets that have
48 been used. Section 3 expands upon feature engineering, hyperparameter tun-
49 ing, and uncertainty estimation. The model performance is quantified in Sec-
50 tion 4, using both the overtopping database and the test data sets. In Section 5,
51 a discussion of the results is presented. Finally, Section 6 contains conclusions
52 and recommendations.

53 **2. Method description**

54 In the following, the methods used in this paper are expanded upon: the
55 machine learning methods applied (Section 2.1), the data used to train them
56 (Section 2.2), the new test data sets (Section 2.3), and the other overtopping
57 prediction methods that are used for comparison (Section 2.4).

58 *2.1. XGBoost and gradient boosting decision trees*

59 XGBoost (Chen & Guestrin, 2016) is a Python (Van Rossum, 1995) imple-
60 mentation of a machine learning method of the type gradient boosting decision
61 trees (GBDT). These methods are based on decision trees that can solve either
62 classification problems (predicting a label) or regression problems (predicting
63 a quantity). These decision trees are therefore often called classification and re-
64 gression trees (CART). In this work regression trees are used for the prediction
65 of mean wave overtopping discharges at coastal structures.

66 Figure 1 shows an ensemble of three decision trees, which each consist of
67 decision and leaf nodes. In decision nodes (D_{ij}), a condition is defined based
68 on a feature from the training data. This combination of feature and condition
69 is often called a split. Node D_{11} for example could contain the condition: "Is

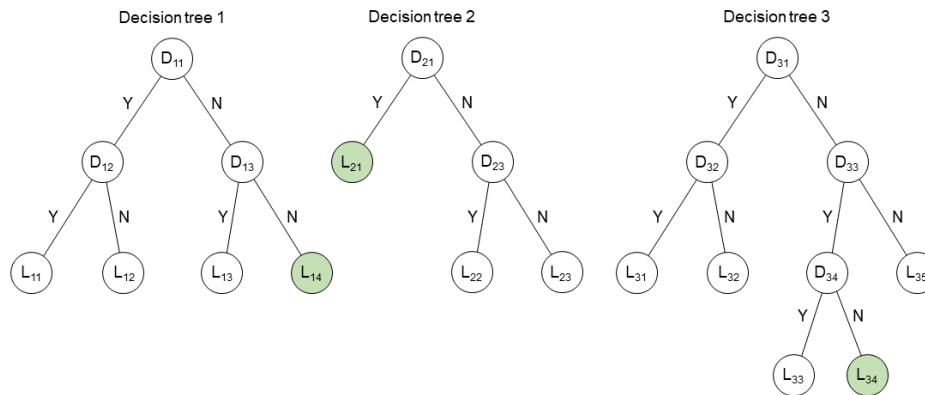


Figure 1: Schematic depiction of an ensemble of decision trees with decision (D_{ij}) and leaf nodes (L_{ij}). An example prediction for one combination of input parameters is shown in green. Source: Den Bieman et al. (2020)

70 the berm width larger than 0 m?". Two tree branches emerge from the node,
 71 one for each possible answer (Yes or No) to the question. These feed into either
 72 another decision node or a leaf node. Leaf nodes (L_{ij}) form the end points of
 73 the tree and contain the prediction. Leaves of regression trees predict values,
 74 whereas leaves of classification trees predict classes. The depth of a decision
 75 tree is defined as the number of subsequent decision nodes from start to leaf
 76 (i.e. decision trees 1 and 2 depicted in Figure 1 have a depth of 2, while decision
 77 tree 3 has a depth of 3).

78 In practice, many classification or regression problems are far too complex
 79 to solve with a single decision tree. Hence, GBDT methods use a large amount
 80 of trees in an ensemble. The basic principle underlying an ensemble of deci-
 81 sion trees is that a combination of weak predictors can form a strong predictor.
 82 The prediction of the ensemble is the sum of the predictions of the individual
 83 trees (see the green leaf nodes in Figure 1 for example), taking the learning rate
 84 into account (see Section 3.1). Newly added trees seek to correct the prediction
 85 errors of the existing trees within the ensemble. In this way, the prediction er-
 86 ror is iteratively reduced. The total amount of trees in the ensemble can either
 87 be specified beforehand or determined on the fly based on the error reduc-
 88 tion (often referred to as "early stopping"). The latter is applied in this work
 89 and is further explained in Section 3.3. When determining the configuration of
 90 a tree, its splits need to be determined. First an objective function is defined
 91 that both rewards accurate predictions and penalizes tree complexity. The al-
 92 gorithm starts at a tree depth of 0 and iteratively adds levels of tree depth. For

93 every level, it finds the optimal condition and leaf values for the split per fea-
94 ture. Subsequently, the feature and split that result in the largest improvement
95 of the objective function is used in the decision node, growing the tree one level
96 deeper. The tree is grown up to the maximum tree depth. A more detailed de-
97 scription of the algorithm is given by Chen & Guestrin (2016).

98 The use of an ensemble of decision trees results in a flexible resolution, de-
99 pending on the local density of training data. This is especially useful given the
100 large density differences in overtopping databases. Note that, as a result, GBDT
101 methods are generally expected to be less suitable for extrapolation far beyond
102 the coverage of the training data.

103 *2.2. Training data set*

104 Currently, the available NN models are the model by Van Gent et al. (2007),
105 hereafter also referred to as "NN", and the model by Zanuttigh et al. (2016),
106 hereafter also referred to as "NNb". In this work, the XGB model performance is
107 compared to both NN, NNb, and empirical overtopping formulae (TAW, 2002;
108 EurOtop, 2018). The NN model is trained on a selection of entries from the
109 original CLASH database (Steendam et al., 2004). The NNb model by Zanuttigh
110 et al. (2016) uses an extended version of the CLASH database as training data
111 set. The extended database adds additional data on vertical walls (Oumeraci
112 et al., 2007), rubble mounds with cobs (Besley et al., 1993), reshaping berm
113 breakwaters (Lykke Andersen et al., 2008), smooth steep slopes (Victor & Troch,
114 2012), and smooth slopes with walls (Van Doorslaer et al., 2015). This addi-
115 tional data has been merged with the CLASH database into the database used
116 by Zanuttigh et al. (2016). This will be referred to as the "overtopping database"
117 in the rest of this paper. The overtopping database has been randomly split
118 80%/20% into two parts: a "training data set" (6943 records) used for training
119 the XGB model, and a "test data set" (1736 records) which is kept strictly sep-
120 arate and is only used to demonstrate the predictive quality of the final trained
121 model. Finally, the new data (from four new data sets described in Section 2.3)
122 is referred to as "additional test data sets" or "unseen data".

123 Not all available parameters from the overtopping database are used in model
124 training. Those parameters that are used to train a model are called features. In
125 Table 1 and Figure 2, the features used in the training of one or more models
126 (NN, NNb and/or XGB) are respectively listed and illustrated. This includes the
127 additions that follow from feature engineering, as described in Section 3.2. The
128 target variable used in model training is the \log_{10} of the mean wave overtopping
129 discharge q after Froude scaling.

130 As in Van Gent et al. (2007), Froude's similarity law is used to scale the over-
 131 topping database features to $H_{m0,toe} = 1$ m, which is indicated in the right-most
 132 column of Table 1. This scaled data is used in the model training detailed in
 133 Section 3. After being used for scaling, the feature $H_{m0,toe}$ is no longer used
 134 in model training. Similarly, the complexity (CF) and reliability factors (RF)
 135 are not directly used for model training. They serve strictly for the weight-
 136 ing of the training data records. Both factors take on integer values of 1 (low
 137 complexity, high reliability) through 4 (high complexity, low reliability). The
 138 weight factor (WF) is determined with the formula from Van Gent et al. (2007):
 139 $WF = (4 - RF) \cdot (4 - CF)$. This formula gives the highest WF to the most reliable
 140 and least complex training data. Very unreliable ($RF = 4$) or complex ($CF = 4$)
 141 are excluded from the training data. In the end this results in a total of 8679
 142 records in the overtopping database.

143 Additionally, Van Gent et al. (2007) state that measurements of very small
 144 mean wave overtopping discharges can be strongly affected by scale effects,
 145 and thus are less reliable. The practical application or relevance of discharges
 146 smaller than 0.001 l/m/s is also quite low. Therefore they suggest applying
 147 $WF = 1$ to all entries with $q < 10^{-6}$ m³/s/m (before Froude scaling) and dis-
 148 regarding their associated reliability and complexity factors. This suggestion is
 149 adopted and applies to 1060 of the 8679 records.

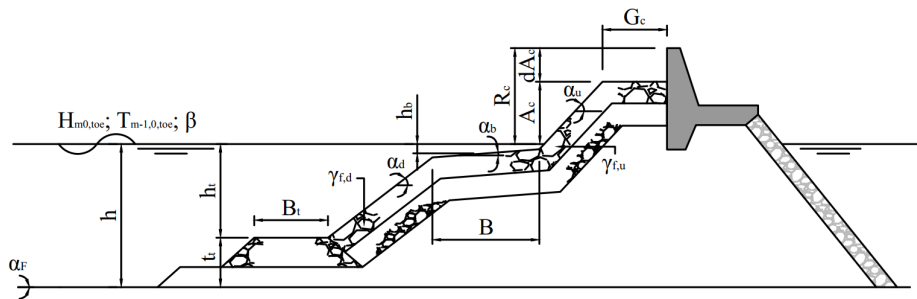


Figure 2: Feature definitions, adapted from Van Gent et al. (2007).

150 2.3. Additional test data sets

151 Next to the training data described in Section 2.2, several additional test
 152 data sets from recent physical model experiments are used to evaluate the model
 153 performance. These additional test data sets are as of yet unseen by any of the
 154 machine learning models, i.e. they are not part of the data the NN, NNb and

Table 1: Overview of features used in model training in the NN by Van Gent et al. (2007), the NNb by Zanuttigh et al. (2016) and the new XGB model.

Name	Symbol	Unit	NN	NNb	XGB	Fr scaled
Mean wave overtopping discharge	q	[m ³ /s/m]	✓	✓	✓	✓
Water depth, toe	h	[m]	✓	✓	✓	✓
Spectral significant wave height, toe	$H_{m0,toe}$	[m]	✓	✓	✓	-
Spectral wave period, toe	$T_{m-1,0,toe}$	[s]	✓	✓	✓	✓
Angle of wave attack	β	[°]	✓	✓	✓	-
Roughness factor of the structure	γ_f	[-]	✓	✓	-	-
Roughness factor of the lower slope	$\gamma_{f,d}$	[-]	-	-	-	-
Roughness factor of the upper slope	$\gamma_{f,u}$	[-]	-	-	✓	-
Ratio of roughness factors	$f_{\gamma_f} = \frac{\gamma_{f,d}}{\gamma_{f,u}}$	[-]	-	-	✓	-
Cotangent of the lower slope	$\cot \alpha_d$	[-]	✓	✓	✓	-
Cotangent of the upper slope	$\cot \alpha_u$	[-]	✓	-	✓	-
Cotangent of the average slope	$\cot \alpha_{incl}$	[-]	-	✓	-	-
Crest freeboard	R_c	[m]	✓	✓	✓	✓
Armour crest freeboard	A_c	[m]	✓	✓	-	✓
Difference between crest and armour crest freeboard	$dA_c = A_c - R_c$	[m]	-	-	✓	✓
Crest width	G_c	[m]	✓	✓	✓	✓
Width of the berm	B	[m]	✓	✓	✓	✓
Water depth above the berm	h_b	[m]	✓	✓	✓	✓
Tangent of berm slope	$\tan \alpha_B$	[-]	✓	-	✓	-
Water depth above the toe structure	h_t	[m]	✓	✓	-	✓
Thickness of the toe structure	$t_t = h - h_t$	[m]	-	-	✓	✓
Width of the toe structure	B_t	[m]	✓	✓	✓	✓
Element size structure	D	[m]	-	✓	-	✓
Cotangent of foreshore slope	$m = \cot \alpha_F$	[-]	-	✓	-	-
Tangent of foreshore slope	$\tan \alpha_F$	[-]	-	-	✓	-
Complexity factor	CF	[-]	✓	✓	✓	-
Reliability factor	RF	[-]	✓	✓	✓	-

155 XGB models are trained on. In Table 2, the relevant parameter ranges covered
 156 by these additional test data sets are listed. The individual data sets are de-
 157 scribed in more detail below. Part of these data sets contain situations that are
 158 underrepresented in the overtopping database; i.e. these data sets contain ele-
 159 ments in regions with little or no coverage, or may be situated in remote corners
 160 of the database. For such data records it can be difficult for data-driven models
 161 to obtain accurate and meaningful predictions.

Table 2: Overview of the relevant parameter ranges covered by the additional test data sets and the examples shown in Figure 3.

Symbol	DS 1a	DS 1b	DS 2	DS 3	DS 4	Fig. 3
$R_c/H_{m0,toe}$	0.9 - 1.8	0.9 - 2.1	0.8 - 2.1	0.8 - 2.2	1.0 - 3.0	1 - 1.5
$dA_c/H_{m0,toe}$	0	0	-0.7 - 0	-0.8 - -0.2	0	-0.5 - 0
$B/H_{m0,toe}$	0 - 2.2	1.4 - 2.2	0	0	1.1 - 2.5	0.5
$h_b/H_{m0,toe}$	-0.3 - 0.3	-0.5 - 0.5	0	0	-0.42 - 0.42	0
$\cot \alpha_d$	3	3	2	2	3	4
$\cot \alpha_u$	3	3	2	2	3	2.5
$\gamma_{f,u}$	0.4 - 1.0	0.5 - 1.0	0.4	0.45	0.8	0.5
f_{γ_f}	1.0	0.5 - 2.0	1.0	1.0	1.0	1.0
$h/H_{m0,toe}$	4.3 - 6.7	4.3 - 6.5	4.1 - 10.2	3.5 - 11.6	2.7 - 6.7	1.5
$s_{m-1,0}$ [%]	2.7 - 4.9	1.3 - 4.2	1.3 - 4.2	1.4 - 4.8	1.7 - 4.2	2.4
β [°]	0	0	0	0 - 75	0 - 75	0
$\tan \alpha_F$	0	0	0	0	0	0

162 Data Set 1 (362 records) comes from physical model studies of the influ-
 163 ence of roughness on wave overtopping at dikes and rock structures (Chen
 164 et al., 2020a,b). These experiments feature different revetment types, includ-
 165 ing roughness differences between upper and lower slope. None of the exist-
 166 ing overtopping prediction methods properly take those roughness differences
 167 into account, except for the method proposed by Chen et al. (2020b). As a con-
 168 sequence, the prediction methods applied here are expected to be less accurate
 169 for the entries with roughness differences than they will be for entries with con-
 170 stant roughness. Hence, in the following the data set will be split into two parts:
 171 Data Set 1a (206 records) only contains data with constant roughness, while
 172 Data Set 1b (156 records) exclusively contains the data records with roughness
 173 differences between the upper and lower slopes.

174 Data Set 2 (51 records) contains physical model experiment data of a rock
175 structure with a crest wall (Jacobsen et al., 2018).

176 Data Set 3 (242 records) features physical model experiment data of wave
177 overtopping on rubble mound breakwaters with a crest wall under oblique wave
178 attack (Van Gent & Van der Werf, 2019).

179 Data Set 4 (177 records) consists of data from physical model experiments of
180 impermeable slopes with a berm under oblique wave attack (Van Gent, 2020).

181 In general, Data Sets 1a and 2 are expected to be within the range of the data
182 already present in the training data set. Data Set 1b features roughness differ-
183 ences between the lower and upper slope, which is relatively rare in the training
184 data (2.8% of all records). Data Sets 3 and 4 feature oblique wave attack which is
185 also underrepresented in the training data (10.9% of entries), especially when
186 combined with the presence of a crest wall (1.1% of entries) or a berm (1.0%
187 of entries). Note that the constructions used in Data Set 1-4 are not complex
188 and can be described exactly following the hydraulic structure definitions in
189 Figure 2. Hence $CF = 1$ for all above-mentioned additional test data sets.

190 *2.4. Other overtopping prediction methods*

191 The XGB model results and the performance on measurement data are com-
192 pared to those of other often used overtopping prediction methods. The meth-
193 ods compared to in this work are the empirical formulae from TAW (2002), the
194 second edition of the EurOtop manual (EurOtop, 2018), the NN by Van Gent
195 et al. (2007), and the NNb by Zanuttigh et al. (2016).

196 The TAW and EurOtop manuals contains a selection of empirical formulae
197 that predict mean wave overtopping discharge. Two versions of these formulae
198 are presented; a mean value approach that represents a best fit with data, and
199 a design and assessment approach which includes some conservatism. In this
200 work, the best fit with data is of importance, hence only results from the mean
201 value approach are considered.

202 Van Gent et al. (2007) made use of machine learning methods by applying
203 a NN to predict mean wave overtopping discharge. They use an ensemble of
204 NNs that gives both the expected discharge and the associated prediction un-
205 certainty as an output. Their NN is available through the NN-Overtopping web
206 application (Deltares). Zanuttigh et al. (2016) continues on the same concept
207 but makes use of a combination of a classifier model coupled to three separate
208 neural networks. They used slightly different features to describe the charac-
209 teristics of the hydraulic structure (see Table 1).

210 3. Model training and tuning

211 Training and tuning a machine learning model comprises of several differ-
212 ent steps. Section 3.1 describes the process of tuning the different hyperparam-
213 eters of the XGB model. In Section 3.2, several features from the overtopping
214 database are redefined to be more suitable for use in machine learning meth-
215 ods. Additionally, the process of coming to a final selection of features to train
216 the model on is explained. Section 3.3 deals with the derivation of confidence
217 intervals using bootstrap resampling.

218 3.1. Hyperparameter tuning

219 The term hyperparameter refers to the run control parameters of a machine
220 learning method. For XGB, these run control parameters govern the complexity
221 and architecture of individual decision trees. Without any restriction to com-
222 plexity, the model is expected to be overfitted on the training data, losing any
223 generic predictive skill.

224 The XGB hyperparameters that have been tuned are listed in Table 3 and
225 can be explained as follows. The maximum depth of a single decision tree
226 (*max_depth*) restricts the number of subsequent splits in decision nodes. The
227 values used in the tuning process are chosen to stay within the total number of
228 features in the training data set. Furthermore, when growing the tree each leaf
229 node must contain a minimum number of data points (*min_child_weight*).
230 In this case, leaf nodes with a single data point are not allowed and up to twelve
231 are required. Leaf node values are multiplied by a learning rate *learning_rate*
232 to obtain a slower convergence that reduces overfitting. Learning rates of < 0.1
233 are very common in machine learning, to which the chosen values in Table 3
234 adhere. Note that both the complexity regularization (*reg_lambda*) and the
235 subsampling (*subsample*) terms are not included in hyperparameter tuning.
236 The reason for excluding these parameters is that it leads to more realistic un-
237 certainty estimates, as is further described in Section 3.3. Both hyperparam-
238 eters are set to their default values, with mild regularization (*reg_lambda* = 1)
239 and no subsampling (*subsample* = 1).

240 The optimal hyperparameter values are listed in Table 3 (indicated in blue).
241 These optimal values are found with a K-fold cross-validation (with $K = 5$) com-
242 bined with a grid search. In the K-fold cross-validation, the total data set is split
243 into K parts (or folds). One of the folds is used for model validation, while the
244 rest is used for model training. This is repeated K times, with a different fold
245 used for validation each time. Hence, the choice of $K = 5$ uses 80% of the data

246 for training, the remaining 20% for validation, and is repeated 5 times with a
247 different part used for validation. The K-fold cross-validation is performed in a
248 grid search, which uses every combination of parameters listed in Table 3. Fi-
249 nally, the best performing hyperparameter set is selected. This hyperparameter
250 set prescribes rather shallow trees, with a reasonable amount of data points per
251 leaf and a rather large learning rate.

Table 3: XGB parameter combinations used in the K-fold cross-validation, optimal values in blue.

Name	Parameter name	Values
Max. tree depth	<i>max_depth</i>	6; 7; 8; 10; 12; 14
Min. data points per leaf	<i>min_child_weight</i>	3; 5; 7; 10; 12
Learning rate	<i>learning_rate</i>	0.005; 0.0075; 0.01; 0.02; 0.05

252 3.2. Feature engineering and feature importance

253 Feature engineering is a common step in the process of improving machine
254 learning models. The parameters selected from the data to train a model on are
255 called features. Feature engineering as a term refers to deriving new features
256 that add to or replace existing parameters in the training data set. The intent of
257 feature engineering is to derive new features that provide a better description of
258 the dependencies and sensitivities of the target variable ($\log_{10}(q)$ in this case)
259 to the model input. Note that successful feature engineering depends heav-
260 ily on the characteristics of the data set in question. Hence, there is no single
261 approach that always leads to good results.

262 One often used approach in machine learning is to perform a feature im-
263 portance analysis. This type of analysis seeks to quantify the influence of in-
264 dividual features on the target variable, which is useful information in the de-
265 cision to in- or exclude features in model training. A permutation importance
266 analysis (Breiman, 2001; Fisher et al., 2018) is performed to gain insight into
267 the influence of each feature. In this method, the data is split into a test and a
268 training data set, the latter of which is used to train a single model. For one fea-
269 ture at a time, the test data set values are randomly scrambled. Subsequently,
270 the trained model is used to generate predictions for the test data set. For im-
271 portant features, the scrambling should have a large effect on the prediction
272 of q compared to the unscrambled test data set, whereas the influence will be
273 small for unimportant features. All features are scrambled one-by-one, with the

274 scrambling repeated 5 times to account for the effect of random sampling. The
275 ELI5 (ELI5) Python permutation importance implementation has been used in
276 this paper. In Table 4, the weight and standard deviation (σ) resulting from the
277 permutation analysis are listed for both a selection of features similar to the
278 NNb model (using Froude scaled features and replacing both A_c and h_t with
279 dA_c and t_t respectively) and the candidate features for the XGB model.

280 Using uncorrelated features is imperative to obtain an accurate representa-
281 tion of the importance of each feature. An accurate overview of which features
282 are (un)important to predict the mean wave overtopping discharge enables a
283 well-argued choice for the selection of features used in a machine learning
284 method. Simply using all features unnecessarily increases the computational
285 demand of model training, can promote overfitting, and can reduce the generic
286 applicability of the model. The results of the permutation importance analysis
287 listed in Table 4 support redefinition or removal of certain highly correlated fea-
288 tures, which is explained below.

289 Added information in the overtopping database allows for distinguishing
290 differences between the roughness of the upper ($\gamma_{f,u}$) and lower slope ($\gamma_{f,d}$)
291 of the structure. Since there are many entries in the database with the same
292 roughness on both slopes, $\gamma_{f,u}$ and $\gamma_{f,d}$ will be highly correlated in practice.
293 Since uncorrelated features are preferred, the XGB model uses the ratio be-
294 tween lower and upper slope roughness $f_{\gamma_f} = \frac{\gamma_{f,d}}{\gamma_{f,u}}$ instead of $\gamma_{f,d}$. Similarly, two
295 additional features are made uncorrelated. Firstly, the armour crest freeboard
296 (A_c) is made uncorrelated from the crest freeboard (R_c) by using the difference
297 between both as a feature: $dA_c = A_c - R_c$. Secondly, the water depth above the
298 toe structure (h_t) is replaced by the thickness of the toe structure ($t_t = h - h_t$)
299 to remove the correlation with the water depth (h).

300 In contrast to the NN, the NNb also uses the element size of the structure
301 (D) as a feature. Zanuttigh et al. (2016) indicate that a weighted average of the
302 element size in the wave run-up and run-down area is taken as the represen-
303 tative element size. Next to the mean wave overtopping discharge, the NNb
304 is also used to predict wave reflection and wave transmission coefficients for the
305 given structure, for which the element size is of significant importance. In the
306 context of wave overtopping however, D relates in large part to the roughness
307 of the profile, which is already represented by $\gamma_{f,u}$ and either $\gamma_{f,d}$ or f_{γ_f} . Ta-
308 ble 4 also illustrates that the importance of γ_f (left-hand side) is significantly
309 smaller than that of $\gamma_{f,u}$ (right-hand side), since the roughness in the NNb is
310 represented by both γ_f and D . Thus, since the main effects of the element

size are already present in the parameter(s) accounting for the roughness, there seems to be no benefit to including D as a feature in wave overtopping prediction models. Analogously, the importance of the berm width (B) is also diminished (left-hand side) when it is already implicitly included in the average slope ($\cot \alpha_{incl}$). Replacing the average slope with the upper slope ($\cot \alpha_u$) resolves the problem.

In addition to reducing the amount of highly correlated features, the cotangent of the foreshore slope m is replaced with the tangent of the foreshore slope, $\tan \alpha_F$. In this way, $\tan \alpha_F$ will be 0 for cases without a foreshore.

Table 4: Overview of permutation importance of XGB models with NNb-like feature set and overview of candidates. Features selected in the XGB model are indicated by (\checkmark).

Rank	NNb features		Overview of candidates	
	Feature	Weight $\pm \sigma$	Feature (selected)	Weight $\pm \sigma$
1	R_c	0.9178 ± 0.0405	R_c (\checkmark)	0.8899 ± 0.0279
2	γ_f	0.3093 ± 0.0124	$\gamma_{f,u}$ (\checkmark)	0.4610 ± 0.0197
3	$T_{m-1,0,toe}$	0.1702 ± 0.0041	G_c (\checkmark)	0.1922 ± 0.0081
4	G_c	0.1632 ± 0.0106	$T_{m-1,0,toe}$ (\checkmark)	0.1530 ± 0.0066
5	$\cot \alpha_{incl}$	0.0824 ± 0.0047	B (\checkmark)	0.0799 ± 0.0030
6	D	0.0810 ± 0.0045	$\cot \alpha_u$ (\checkmark)	0.0641 ± 0.0025
7	β	0.0460 ± 0.0021	β (\checkmark)	0.0433 ± 0.0021
8	h	0.0411 ± 0.0037	h (\checkmark)	0.0417 ± 0.0030
9	B	0.0378 ± 0.0025	$\tan \alpha_F$ (\checkmark)	0.0405 ± 0.0017
10	m	0.0364 ± 0.0024	$\cot \alpha_d$ (\checkmark)	0.0236 ± 0.0028
11	$\cot \alpha_d$	0.0194 ± 0.0019	dA_c (\checkmark)	0.0234 ± 0.0020
12	dA_c	0.0150 ± 0.0011	t_t (\checkmark)	0.0200 ± 0.0024
13	t_t	0.0150 ± 0.0027	h_b (\checkmark)	0.0154 ± 0.0016
14	h_b	0.0143 ± 0.0006	B_t (\checkmark)	0.0093 ± 0.0012
15	B_t	0.0063 ± 0.0019	f_{γ_f} (\checkmark)	0.0002 ± 0.0001
16	-	-	$\tan \alpha_B$ (-)	0.0000 ± 0.0001

The selection of features for the XGB model is indicated with check marks in Table 4. The berm slope ($\tan \alpha_B$) is not used to train the model, since it ranks as the least important feature and its importance in absolute terms is very small. Surprisingly, the newly introduced feature f_{γ_f} also ranks low on importance, while Chen et al. (2020b) show the importance of taking into account roughness differences. One of the likely causes for this discrepancy is that only 2.8%

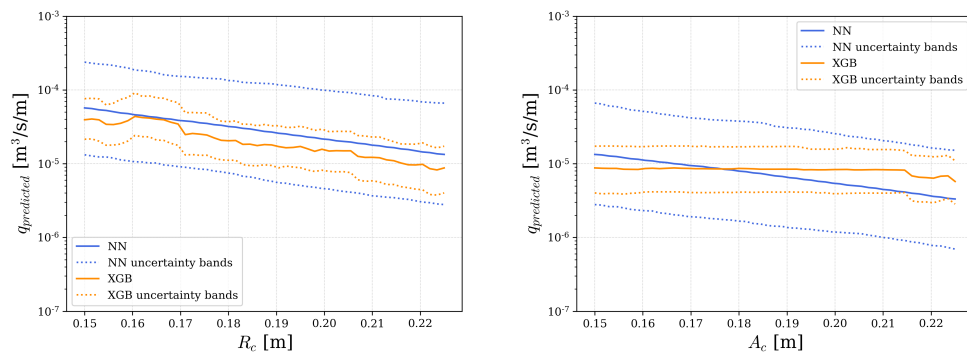
326 of the entries in the overtopping database has a difference between the rough-
327 ness on the upper and lower slopes, and conversely $f_{\gamma_f} = 1.0$ for 97.2% of the
328 data. This means that scrambling the feature in the permutation importance
329 analysis will give the same value for f_{γ_f} in many cases, and thus a seemingly
330 small importance is attributed to the feature. Hence, f_{γ_f} is still included as a
331 feature in model training.

332 3.3. *Bootstrap resampling and confidence intervals*

333 For the sake of consistency and comparability of the results, the bootstrap
334 resampling method (Efron & Tibshirani, 1993) - proposed by Van Gent et al.
335 (2007) and also used by Zanuttigh et al. (2016) - is similarly applied to the XGB
336 model to obtain estimates of prediction errors. Note that there might be other
337 suitable methods for the estimation of predictions errors, but these are not ex-
338 plored in this work. The bootstrap resampling method can be summarized as
339 follows. Firstly, 500 bootstrap resamples are generated from all data available
340 for model training. A resample is a randomized selection from the overtopping
341 database, where individual entries can be selected more than once. When that
342 is the case, the weight factor for that entry is adjusted accordingly within the
343 resample. Subsequently, a model is trained for each resample. The resample
344 is used as a training data set. The training makes use of an "early stopping"
345 algorithm. This algorithm keeps adding new trees to the model, until either
346 the maximum number of trees (set to 100.000) is reached or if the last 1000
347 consecutively added trees do not improve the model prediction for the entries
348 not selected in the bootstrap resample. In the latter case, the model training
349 is stopped and the best model is selected as the training result. In this way,
350 500 models are trained with 500 different but overlapping data sets and no in-
351 dividual model is trained on all available data from the overtopping database.
352 Finally, for each prediction all 500 models are used, from which the median
353 value serves as model prediction and the associated error can be estimated.

354 The use of specific anti-overfitting parameters in XGB tends to generalize
355 the model fits to such degree that the variation in the model predictions - and
356 thus the estimated confidence intervals - is greatly reduced. Hence, in the hy-
357 perparameter tuning for this work (Section 3.1), subsampling is not applied and
358 a milder tree complexity regularization is used. In Figure 3, predictions of the
359 current XGB model and their associated 90% confidence interval are compared
360 to the NN model for singular variations along both crest and armour crest free-
361 board, with constant values for other features. The parameter ranges for these
362 laboratory scale examples are listed in Table 2. As can be seen in Figure 3a and

363 Figure 3b, the updated newly tuned hyperparameter settings lead to seemingly
 364 realistic uncertainty bands. The prediction uncertainty bands are expected to
 365 be smaller than those of the NN model, since the XGB model performance is
 366 generally better (see Section 4). Note that the NN model generally leads to a
 367 smoother trend than the XGB model. This is an inherent feature of the decision
 368 tree based machine learning method applied here. The many splits in an ensemble
 369 of decision trees inadvertently introduce some amount of discontinuity
 370 to the predicted overtopping discharge over a given parameter range. Hence,
 371 it is not related to the approach used to derive uncertainty estimates. Additionally,
 372 further analysis with 1000 resamples shows no significant differences,
 373 which suggests that the 90% interval can be adequately determined from 500
 374 resamples.



(a) Model responses to a variation in R_c , with a constant $A_c/H_{m0,toe} = 1.0$.

(b) Model responses to a variation in A_c , with a constant $R_c/H_{m0,toe} = 1.5$.

Figure 3: Examples of predictions and 90% uncertainty bands for NN (blue) and XGB (orange) models for a changing crest and armour crest freeboard (parameter ranges listed in Table 2).

375 4. Model validation

376 Validation of the XGB model is performed in several steps. Firstly, in Sec-
 377 tion 4.1 its performance is analyzed on the overtopping database and its pre-
 378 dictive skill is demonstrated using the test data set. Subsequently, in Section 4.2
 379 the generic applicability of the model is analyzed by applying the model to the
 380 challenging conditions of the additional test data, which was not used in any
 381 way in the model training. Finally, the model validation after retraining with the
 382 expanded training data set (indicated by "XGBr") is considered in Section 4.3.

383 4.1. Validation on the overtopping database

384 The performance of the XGB model is evaluated on the test data set (see
 385 Section 2.2). Additionally, the prediction errors are compared to those of the
 386 other existing tools to predict overtopping discharges. This primarily concerns
 387 the original NN model (Van Gent et al., 2007), but also includes the NNb model
 388 (Zanuttigh et al., 2016) and the empirical overtopping formulae (TAW, 2002;
 389 EurOtop, 2018) for wave overtopping prediction. The weighted root-mean-
 390 square-error (RMSE) is used as an error criterion. It is defined by Equation 1
 391 and listed for all methods in Table 5.

$$RMSE = \sqrt{\frac{1}{\sum_{n=1}^N (WF_n)} \frac{1}{N} \sum_{n=1}^N (WF_n \cdot (\log_{10}(q_{predicted,n}) - \log_{10}(q_{measured,n}))^2)} \quad (1)$$

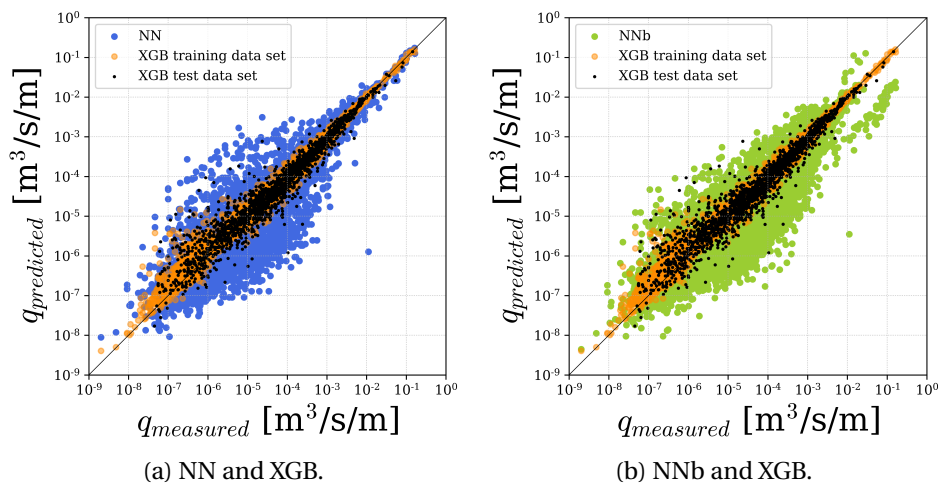


Figure 4: Predictions by the NN (blue) and NNb (green) models for the overtopping database, and by the XGB model for the training data set (orange) and the test data set (black).

392 The predictions of the different machine learning methods for the overtop-
 393 ping database are shown in the scatter plots of Figure 4. The larger RMSEs of
 394 the NN and NNb models translate into a large scatter around the diagonal, with
 395 prediction errors of up to a factor 100. The scatter in the XGB model predic-
 396 tions is visibly much smaller, with differences largely within a factor 10. This
 397 is reflected in Table 5, where a significantly smaller test data set RMSE is listed for

398 XGB (0.284) than for NN (0.478) and NNb (0.580). The same holds for the train-
 399 ing data set and the entire overtopping database. Note that the entire overtop-
 400 ping database was used as a training data set for the NNb model, while a smaller
 401 subset thereof was available to train the NN model.

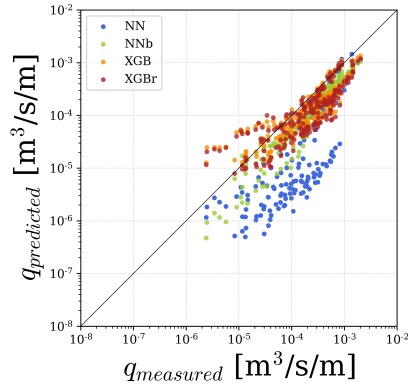
402 4.2. Validation on additional test data sets

403 In Section 4.1 the capability of the XGB model to predict the contents of
 404 the test data set is shown. The good performance on the test data set shows
 405 that the XGB model has predictive capabilities on previously unseen data that
 406 is fairly similar to the training data. It does not, however, show the predictive
 407 capability of the model for conditions that are meaningfully different from the
 408 training data set (in this case the overtopping database). As mentioned in Sec-
 409 tion 2.3, conditions that are not in the overtopping database but are very similar
 410 to it, should be predicted reasonably well. The real challenge lies in conditions
 411 that are not well represented in or covered by the overtopping database. For
 412 instance, conditions with oblique wave attack are relatively sparse in the over-
 413 topping database. Hence, there is relatively little data available for the model
 414 to correctly learn and predict wave overtopping discharges under oblique wave
 415 attack.

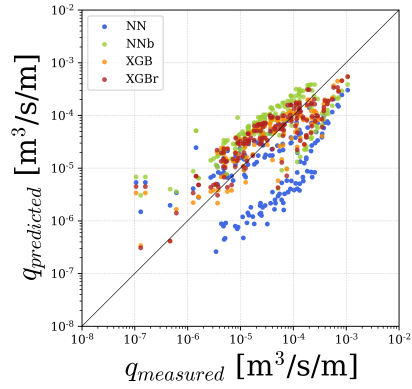
Table 5: RMSE for all overtopping prediction tools on the different data sets. "Unseen Data" includes both the Test data set and the (unseen parts of) Data Set 1-4.

Data Set (size)	TAW	EurOtop	NN	NNb	XGB	XGBr
Training data set (6943)	1.089	1.313	0.490	0.566	0.098	0.097
Test data set (1736)	0.995	1.207	0.478	0.580	0.284	0.285
Overtopping db (8679)	1.071	1.292	0.488	0.569	0.154	0.154
Data Set 1a (206)	0.631	0.696	0.958	0.394	0.349	0.403
Data Set 1b (156)	1.069	1.203	0.860	0.594	0.408	0.419
Data Set 2 (51)	1.047	1.266	0.714	0.575	0.448	0.356
Data Set 3 (242)	1.033	1.097	1.112	0.921	1.127	0.622*
Data Set 4 (177)	1.791	1.792	1.472	1.732	1.743	0.972*
Unseen data	1.170	1.232	1.104	1.005	0.723	0.411
All data	1.086	1.283	0.602	0.636	0.409	0.248

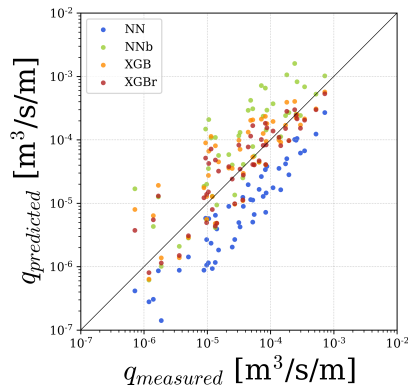
*Data points used in model training have been excluded from RMSE de-
 termination.



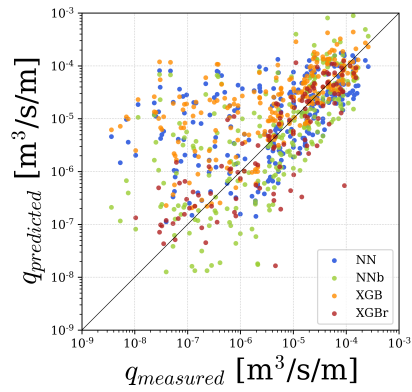
(a) DS1a Constant slope roughness.



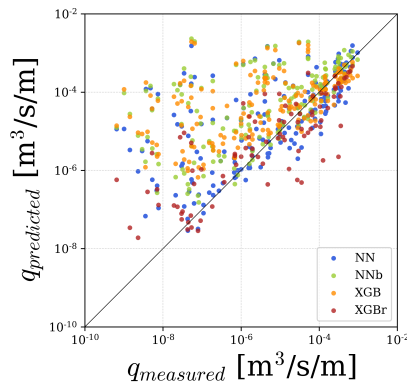
(b) DS1b Composite slope roughness.



(c) DS2 Crest walls.



(d) DS3 Oblique waves and crest walls.



(e) DS4 Oblique waves and berms.

Figure 5: Predictions by the NN (blue), NNb (green), XGB (orange) and XGBr (red) models for all additional test data sets.

416 In Figure 5, the predictions by different machine learning models for the
417 additional test data sets mentioned in Section 2.3 are shown. The RMSEs for all
418 test data sets are listed in Table 5. The table lists the errors of both the different
419 machine learning models and the TAW (2002) and EurOtop (2018) empirical
420 formulae. Note that Data Sets 1-4 use $RF = 1$ and $CF = 1$ for the purpose of
421 weighting in the RMSE calculation. Additionally, the RMSEs are based on all
422 data with $q > 0$ [$\text{m}^3/\text{s}/\text{m}$], including very small discharges.

423 The predictions of the different machine learning models are shown in Fig-
424 ure 5a for Data Set 1a and Figure 5b for Data Set 1b. For the constant roughness
425 cases in Data Set 1a, the predictions of the NN (blue) are in general significantly
426 underestimating the overtopping discharge. In fact, the errors of the NN show
427 a systematic behaviour in the sense that the predictions are rather constantly a
428 factor of about 100 smaller than the measurements. Both the NNb (green) and
429 XGB (orange) models also have a slight tendency towards underestimation, but
430 less severe. This is reflected by the RMSE values in Table 5. The NNb (0.394)
431 and XGB (0.349) models are fairly accurate, where the NN shows a much larger
432 RMSE (0.958) for Data Set 1a due to the systematic differences.

433 Data Set 1b, featuring slopes with roughness differences, shows distinct clus-
434 ters of points grouped in lines for both NN and NNb models. These distinct
435 lines are formed by the different revetment types. This pattern suggests that
436 the influence of roughness and roughness differences on the measured trends
437 is not completely captured by the models. The XGB results exhibit a more ran-
438 dom scatter along the diagonal, with a slight tendency towards overestima-
439 tion. These observations are supported by the RMSE values, which compared
440 to those for Data Set 1a, show some improvement of the NN model (0.860), a
441 significantly worse performance for the NNb model (0.594), and a comparable
442 performance for the XGB model (0.408). Presumably, the ability to recognize
443 roughness differences through the f_{γ_f} parameter explains the relatively high
444 accuracy of the XGB model for Data Set 1b.

445 Figure 5c shows the predictions for Data Set 2. The NN predictions are sys-
446 tematically underestimating the q . Conversely, the NNb and - to a lesser extent
447 - the XGB model tend to slightly overestimate overtopping. This is mirrored in
448 the RMSE values, where the XGB model is the most accurate (see Table 5).

449 All three models show a large amount of scatter for Data Set 3 (see Fig-
450 ure 5d), with errors of up to three orders of magnitude towards overestimat-
451 ing q . Notably, errors are significantly larger for small measured overtopping
452 discharges ($q < 10^{-6} \text{m}^3/\text{s}/\text{m}$). Further analysis shows that the errors also in-
453 crease with increasing angle of wave attack, β . The lower accuracy shown by

454 all models (see Table 5) is likely the result of the small amount of training data
455 containing $\beta > 0^\circ$.

456 In Figure 5e, a pattern similar to Data Set 3 emerges for Data Set 4. All
457 predictive models give very poor predictions, with a tendency towards overes-
458 timation of q up to four orders of magnitude. Again, the RMSE increases with
459 both increasing β and decreasing q .

460 4.3. Retrained XGB model

461 The combination of the lower accuracy on the additional test data sets con-
462 taining oblique wave attack - Data Set 3 and 4 - and the fact that entries with
463 $\beta > 0^\circ$ are underrepresented in the training data suggests that expanding the
464 training data with oblique wave data could improve the performance of data-
465 driven models. To that end, the training data set for the XGB model is ex-
466 panded by adding a random selection of half of Data Set 3 and 4. Subsequently,
467 the model is retrained, again following the bootstrap resampling approach de-
468 tailed in Section 3.3. The predictions of the retrained XGB model, indicated by
469 "XGBr", are shown in red in Figure 5 and the associated errors are again listed
470 in Table 5. Note that the predictions and RMSE shown are only based on the
471 parts of Data Set 3 and 4 not used for model training.

472 By changing the training data set and retraining a data-driven model, its
473 predictions change. The XGBr model shows significantly decreased errors for
474 (the unseen parts of) Data Set 3 and 4, as expected. Additionally, the RMSE for
475 Data Set 2 decreases as well, potentially because a part of the data added to
476 the training data set also includes crest walls. The errors for Data Sets 1a and
477 1b slightly increase however. The reason will be that adding data to the train-
478 ing data set causes data similar to Data Set 1a and 1b to become relatively less
479 important in model training. In general, the fact that the XGB models perform
480 well on unseen data (both the test data set and Data Set 1-4) strongly suggests
481 that the models are not overfitted and can be generically applied.

482 Comparison between the different machine learning methods shows that
483 the XGB errors are generally smaller than NN and NNb for both the test and
484 training data sets and the parts of the unseen data that are well represented in
485 the training data (Data Set 1a, 1b and 2). Unseen data in data sparse regions
486 of the training data set (Data Set 3 and 4 containing oblique waves) results in
487 significantly larger errors. Expanding the training data set with oblique wave
488 data and retraining the model (XGBr) results in significantly smaller errors for
489 Data Set 3 and 4, at the cost of a small increase of the errors for Data Set 1a and
490 1b.

491 The results of the TAW (2002) and EurOtop (2018) empirical overtopping
492 formulae are also included in Table 5. Note that here all data has been used,
493 also data points that are not strictly within the validity range of the empirical
494 expressions. This is done both to compare their performance on the same data
495 set as the machine learning methods and because these empirical expressions
496 are often applied outside their range of validity. TAW (2002) results in an RMSE
497 of around 1 for most tests, with a higher accuracy for Data Set 1a and a lower ac-
498 curacy for Data Set 4. A similar trend emerges from the EurOtop (2018) results,
499 although the errors are slightly higher than for TAW. In general, the machine
500 learning methods (both NN and XGB variants) perform better than the empiri-
501 cal overtopping formulae.

502 Finally, in the scatter plots of Figure 6 predictions by the different prediction
503 methods are shown for the combination of the test data set and the additional
504 test data sets (i.e. all data that has not been used to train the machine learning
505 methods). Both the TAW (Figure 6a) and EurOtop (Figure 6b) empirical formu-
506 lae show a large amount of scatter, with many outliers severely underestimating
507 the amount of overtopping. Note that these outliers are a mix of both very re-
508 liable ($RF = 1$) and reasonably reliable ($RF = 2$ or 3) data. The NN (Figure 6c)
509 and NNb (Figure 6d) models show less scatter, more or less symmetrically dis-
510 tributed around the diagonal. The XGB (Figure 6e) and XGBr (Figure 6f) models
511 exhibit a very limited amount of scatter. These observations are reflected in
512 the RMSE values in Table 5. In general, the XGB methods result in the small-
513 est RMSE on all data. They are followed by both the NN and NNb models, that
514 perform reasonably similar to each other. Lastly, the empirical formulae result
515 in the largest errors, with TAW (2002) having a higher accuracy than EurOtop
516 (2018).

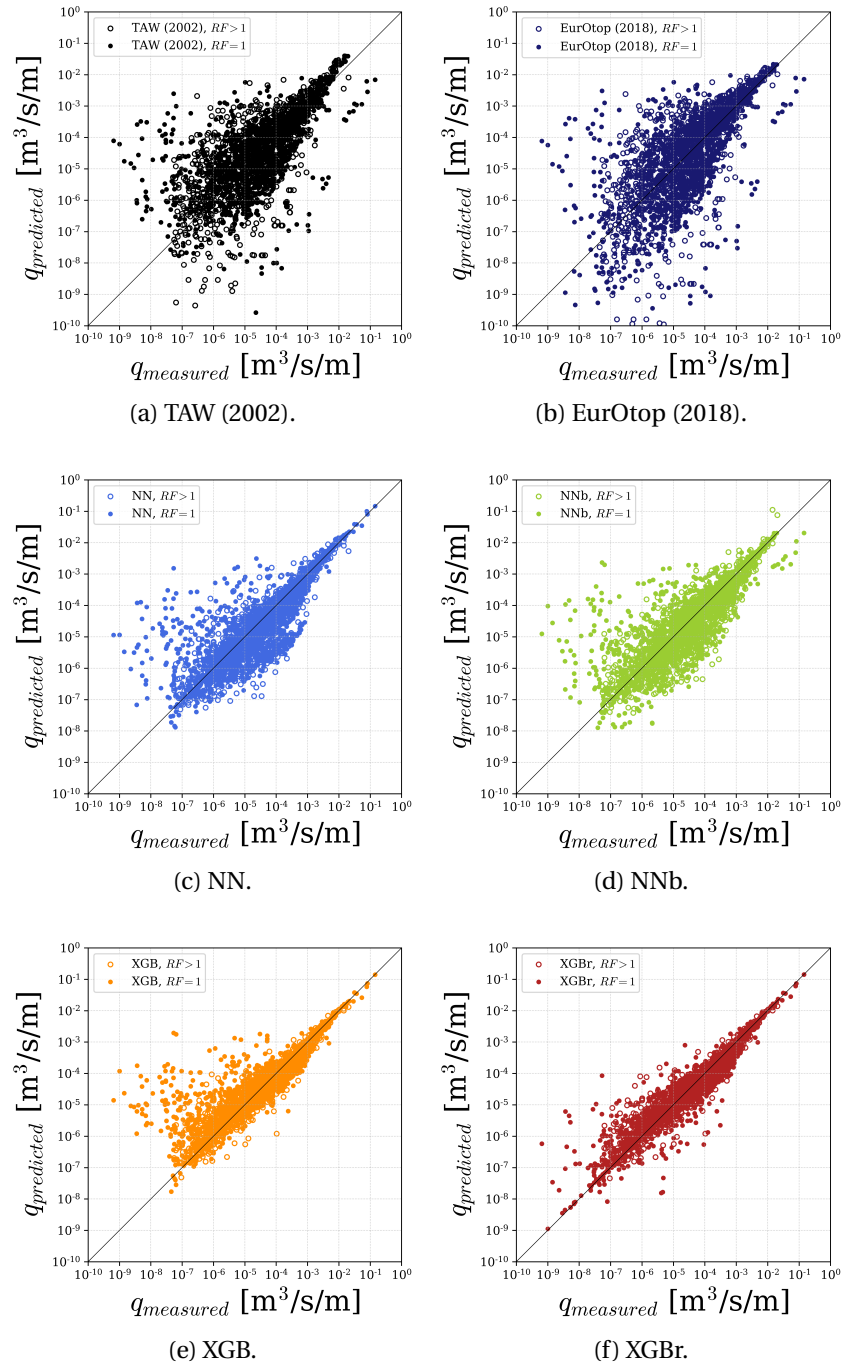


Figure 6: Overview of predictions for the test data set and additional test data sets using different prediction methods. For XGBr, only Data Set 3 and 4 data points not used in model training are shown.

517 **5. Discussion**

518 Section 4.3 shows that expanding the training data set with new data can
519 greatly improve the overall performance of data-driven methods. This is espe-
520 cially true when newly added data covers parameter combinations that are cur-
521 rently not covered by, or underrepresented in the training data. Here this is the
522 case for oblique wave attack combined with either a berm or a crest wall. Con-
523 tinuous expansion of training data and retraining and revalidation of models
524 is recommended for data-driven methods. Another advantage of adding data
525 from recent physical models is the relatively high reliability of recent data, e.g.
526 due to more advanced reflection compensation techniques and second-order
527 wave generation that are often lacking in older data.

528 In Section 2.2, a strict split was imposed between training and test data sets
529 to convincingly demonstrate the predictive quality of the trained model. The
530 NN by Van Gent et al. (2007), however, does not use a separate test data set.
531 Instead, all data is used in the model training process. Due to the application of
532 bootstrap resampling (as described in Section 3.3) the overall model is based on
533 500 individual models where no single model is trained on the entire training
534 data set. For the sake of completeness, an XGB model is trained using the same
535 method (see Figure 7). As a consequence of its construction, this model shows
536 small RMSEs for both the overtopping database (0.100) and all data (0.092).

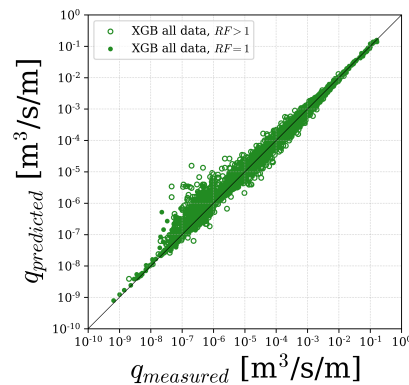


Figure 7: Overview of predictions for both the overtopping database and the additional test data sets by the XGB model trained on a bootstrap resampling of all data, following the same method as used in Van Gent et al. (2007).

537 In the model validation effort presented in this work, multiple data-driven
538 overtopping prediction methods - including the new XGB model - are com-

539 pared to empirical overtopping formulae. Overall, the data-driven methods
540 (especially the XGB models) perform better than the empirical formulae on
541 both the overtopping database and the additional test data sets (unseen data)
542 examined in this paper. This suggests that data-driven methods should become
543 increasingly important as a tool for engineering and design of coastal struc-
544 tures, at least alongside, if not instead of, the existing empirical formulae. If, for
545 design purposes, some conservatism is desirable, this could be derived from
546 the confidence intervals that are given together with the predictions.

547 **6. Conclusions and recommendations**

548 In this work, the application of XGBoost to the prediction of mean wave
549 overtopping is further refined and compared to other available prediction meth-
550 ods. The selection of features on which to train the model is expanded upon in
551 detail, with significant improvements compared to existing literature. A com-
552 bination of bootstrap resampling of the overtopping database and suitable se-
553 lection of model hyperparameters results in realistic confidence intervals. All
554 considered prediction methods are extensively validated on the training and
555 test data sets. The XGBoost model outperforms other prediction methods on
556 both test and training data sets from the overtopping database. All data-driven
557 methods show less accuracy on the oblique wave data present in the addi-
558 tional test data sets, presumably because these cases are underrepresented in
559 the overtopping database. Adding a randomly selected part of the new oblique
560 wave data to the training data greatly improves the quality of the XGBoost model.

561 Similar to the lack of oblique wave data, the overtopping database contains
562 many more white spots. For further research, it is recommended to identify
563 these white spots and add data that falls within them. Hence, the white spots
564 in the overtopping database can also be used to identify which data is useful to
565 generate in new physical model experiments. At the same time, or as an alter-
566 native to physical model data, it is recommended to explore the possibility of
567 adding numerical model data to the training data set. Numerical models prove
568 to be a relatively efficient way of generating large amounts of data to address
569 white spots in the training data set. Note however that this requires numerical
570 models that are extensively validated and calibrated on physical model data, in
571 order to obtain reliable numerical data.

572 **References**

- 573 Besley, P., Reeves, M., & Allsop, N. W. H. (1993). *Random wave physical model*
574 *tests: overtopping and reflection performance*. Technical Report HR Walling-
575 ford. Report IT 384.
- 576 Breiman, L. (2001). Random forests. *Machine learning*, 45, 5–32.
- 577 Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting sys-
578 tem. *CoRR*, *abs/1603.02754*. URL: <http://arxiv.org/abs/1603.02754>.
579 arXiv:1603.02754.
- 580 Chen, W., Marconi, A., Van Gent, M. R. A., Warmink, J. J., & Hulscher, S. J. M. H.
581 (2020a). Experimental study on the influence of berms and roughness on
582 wave overtopping at rock-armoured dikes. *Journal of Marine Science and*
583 *Engineering*, 8. doi:10.3390/jmse8060446.
- 584 Chen, W., Van Gent, M. R. A., Warmink, J. J., & Hulscher, S. J. M. H.
585 (2020b). The influence of a berm and roughness on the wave overtopping at
586 dikes. *Coastal Engineering*, 156, 103613. doi:[https://doi.org/10.1016/](https://doi.org/10.1016/j.coastaleng.2019.103613)
587 [j.coastaleng.2019.103613](https://doi.org/10.1016/j.coastaleng.2019.103613).
- 588 Deltares (). Overtopping neural network webtool. URL: [https://www.](https://www.deltares.nl/en/software/overtopping-neural-network/)
589 [deltares.nl/en/software/overtopping-neural-network/](https://www.deltares.nl/en/software/overtopping-neural-network/) (accessed
590 on 8 April 2020).
- 591 Den Bieman, J. P., Wilms, J. M., Van den Boogaard, H. F. P., & Van Gent, M. R. A.
592 (2020). Prediction of mean wave overtopping discharge using gradient boost-
593 ing decision trees. *Water*, 12. doi:<https://doi.org/10.3390/w12061703>.
- 594 Efron, B., & Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. London:
595 Chapman & Hall.
- 596 ELI5 (). ELI5 python package. URL: [https://github.com/TeamHG-Memex/](https://github.com/TeamHG-Memex/eli5)
597 [eli5](https://github.com/TeamHG-Memex/eli5).
- 598 EurOtop (2018). *Manual on wave overtopping of sea defences and related struc-*
599 *tures*. J. W. van der Meer, N. W. H. Allsop, T. Bruce, J. de Rouck, A. Kortenhaus,
600 T. Pullen, H. Schüttrumpf, P. Troch, B. Zanuttigh (Eds.), [www.overtopping-](http://www.overtopping-manual.com)
601 [manual.com](http://www.overtopping-manual.com).

-
- 602 Fisher, A., Rudin, C., & Dominici, F. (2018). All models are wrong, but many are
603 useful: Learning a variable's importance by studying an entire class of predic-
604 tion models simultaneously. arXiv:<http://arxiv.org/abs/1801.01489>.
- 605 Jacobsen, N. G., Van Gent, M. R. A., Capel, A., & Borsboom, M. (2018). Nu-
606 merical prediction of integrated wave loads on crest walls on top of rubble
607 mound structures. *Coastal Engineering*, 142, 110 – 124. URL: [http://](http://www.sciencedirect.com/science/article/pii/S037838391730220X)
608 www.sciencedirect.com/science/article/pii/S037838391730220X.
609 doi:<https://doi.org/10.1016/j.coastaleng.2018.10.004>.
- 610 Lim, S., & Chi, S. (2019). Xgboost application on bridge management sys-
611 tems for proactive damage estimation. *Advanced Engineering Informatics*,
612 41, 100922. doi:<https://doi.org/10.1016/j.aei.2019.100922>.
- 613 Lykke Andersen, T., Skals, K. T., & Burcharth, H. F. (2008). Comparison of ho-
614 mogenous and multi-layered berm breakwaters with respect to overtopping
615 and front slope stability. In *Proc. 31th ICCE*. ASCE. doi:[https://doi.org/](https://doi.org/10.1142/9789814277426_0273)
616 [10.1142/9789814277426_0273](https://doi.org/10.1142/9789814277426_0273).
- 617 Oumeraci, H., Kortenhaus, A., & Burg, S. (2007). *Investigations of wave load-*
618 *ing and overtopping of an innovative mobile flood defence system: Analysis*
619 *of model tests and design formulae*. Technical Report Technische Universität
620 Braunschweig, Leichtweiß-Institut für Wasserbau, Abteilung Hydromechanik
621 und Küsteningenieurwesen. Report Nr. 949.
- 622 Steendam, G. J., Van der Meer, J. W., Verhaeghe, H., Besley, P., Franco, L., & Van
623 Gent, M. R. A. (2004). The international database on wave overtopping. In
624 *Proc. 29th ICCE, Vol.4* (pp. 4301–4313). World Scientific. doi:[https://doi.](https://doi.org/10.1142/9789812701916_0347)
625 [org/10.1142/9789812701916_0347](https://doi.org/10.1142/9789812701916_0347).
- 626 TAW (2002). *Wave run-up and wave overtopping at dikes*. Technical Report
627 Technical Advisory Committee for Flood Defence in the Netherlands (TAW).
628 Delft.
- 629 Van Doorslaer, K., De Rouck, J., Audenaert, S., & Duquet, V. (2015). Crest mod-
630 ifications to reduce wave overtopping of non-breaking waves over a smooth
631 dike slope. *Coastal Engineering*, 101, 69–88. doi:[https://doi.org/10.](https://doi.org/10.1016/j.coastaleng.2015.02.004)
632 [1016/j.coastaleng.2015.02.004](https://doi.org/10.1016/j.coastaleng.2015.02.004).

-
- 633 Van Gent, M. R. A. (2020). Influence of oblique wave attack on wave overtopping
634 at smooth and rough dikes with a berm. *Coastal Engineering*, 160, 103734.
635 doi:<https://doi.org/10.1016/j.coastaleng.2020.103734>.
- 636 Van Gent, M. R. A., Van den Boogaard, H. F. P., Pozueta, B., & Medina, J. R.
637 (2007). Neural network modelling of wave overtopping at coastal struc-
638 tures. *Coastal Engineering*, 54, 586–593. doi:[https://doi.org/10.1016/](https://doi.org/10.1016/j.coastaleng.2006.12.001)
639 [j.coastaleng.2006.12.001](https://doi.org/10.1016/j.coastaleng.2006.12.001).
- 640 Van Gent, M. R. A., & Van der Werf, I. M. (2019). Influence of oblique wave
641 attack on wave overtopping and forces on rubble mound breakwater crest
642 walls. *Coastal Engineering*, 151, 78–96. doi:[https://doi.org/10.1016/j.](https://doi.org/10.1016/j.coastaleng.2019.04.001)
643 [coastaleng.2019.04.001](https://doi.org/10.1016/j.coastaleng.2019.04.001).
- 644 Van Rossum, G. (1995). *Python tutorial*. Technical Report CS-R9526 Centrum
645 voor Wiskunde en Informatica (CWI).
- 646 Victor, L., & Troch, P. (2012). Wave overtopping at smooth impermeable steep
647 slopes with low crest freeboards. *Journal of Waterway, Port, Coastal, and*
648 *Ocean Engineering*, 138, 372–385. doi:[https://doi.org/10.1061/\(ASCE\)](https://doi.org/10.1061/(ASCE)WW.1943-5460.0000141)
649 [WW.1943-5460.0000141](https://doi.org/10.1061/(ASCE)WW.1943-5460.0000141).
- 650 Zanuttigh, B., Formentin, S. M., & Van der Meer, J. W. (2016). Prediction of
651 extreme and tolerable wave overtopping discharges through an advanced
652 neural network. *Ocean Engineering*, 127, 7–22. doi:[https://doi.org/10.](https://doi.org/10.1016/j.oceaneng.2016.09.032)
653 [1016/j.oceaneng.2016.09.032](https://doi.org/10.1016/j.oceaneng.2016.09.032).
- 654 Zhang, D., Qian, L., Mao, B., Huang, C., Huang, B., & Si, Y. (2018). A data-
655 driven design for fault detection of wind turbines using random forests and
656 xgboost. *IEEE Access*, 6, 21020–21031. doi:[https://doi.org/10.1109/](https://doi.org/10.1109/ACCESS.2018.2818678)
657 [ACCESS.2018.2818678](https://doi.org/10.1109/ACCESS.2018.2818678).