

Article

Adaboost Algorithm in Artificial Intelligence for Optimizing the IRI Prediction Accuracy of Asphalt Concrete Pavement

Changbai Wang *, Shuzhan Xu and Junxin Yang

School of Civil Engineering and Architecture, Anhui University of Science and Technology, Huainan 232001, China; shuzhanww@163.com (S.X.); 15856995257@163.com (J.Y.)

* Correspondence: chbwang@aust.edu.cn

Abstract: The international roughness index (IRI) for roads is a crucial pavement design criterion in the Mechanistic-Empirical Pavement Design Guide (MEPDG). However, studies have shown that the IRI transfer function in the MEPDG is simply a linear combination of road parameters, so it cannot provide accurate predictions. To solve this issue, this research developed an AdaBoost regression (ABR) model to improve the prediction ability of IRI and compared it with the linear regression (LR) in MEPDG. The development of the ABR model is based on the Python programming language, using the 4265 records from the Long-Term Pavement Performance (LTPP) that include the pavement thickness, service age, average annual daily truck traffic (AADTT), gator cracks, etc., which are reliable data that are preserved after years of monitoring. The results reveal that the ABR model is significantly better than the LR because the correlation coefficient (R^2) between the measured and predicted values in the testing set increased from 0.5118 to 0.9751, and the mean square error (MSE) decreased from 0.0245 to 0.0088. By analyzing the importance of variables, there are many additional crucial factors, such as raveling and bleeding, that affect IRI, which leads to the weak performance of the LR model.

Keywords: MEPDG; LTPP; IRI prediction; decision tree; AdaBoost regression



Citation: Wang, C.; Xu, S.; Yang, J. Adaboost Algorithm in Artificial Intelligence for Optimizing the IRI Prediction Accuracy of Asphalt Concrete Pavement. *Sensors* **2021**, *21*, 5682. <https://doi.org/10.3390/s21175682>

Received: 10 July 2021

Accepted: 20 August 2021

Published: 24 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

During the construction of asphalt concrete (AC) roads, the longitudinal profile of the pavement often changes due to the influence of the construction conditions. Moreover, in the long-term driving process, the outline shape will continue to change under heavy vehicle loads and harsh environments, visually manifesting as the pavement becomes rough. Vehicles running on uneven roads not only affect the safety and comfort of the passengers or drivers, but also increase operating costs (such as increasing fuel consumption, reducing driving speed, and extending the travel time) [1] and at the same time, will accelerate the destruction of the pavement structure, affecting the service life and maintenance cycle of the pavement [2]. Therefore, with the continuous improvement of highway service quality requirements and the establishment of pavement management systems, roughness has become one of the most critical indicators of current pavement performance. Accurately assessing the condition of a gradually distressed road surface will be a vital task of the Department of Transportation (DOT) to ensure the safe and durable operation of the national road traffic network [3,4].

To better express and quantify the roughness of the pavement surface, researchers have introduced a one-dimensional pavement longitudinal profile index called the international roughness index (IRI), which is widely used in the research on road and traffic engineering [5–9]. In the Mechanistic-Empirical Pavement Design Guide (MEPDG), IRI is also used as a robust design criterion that is critical to measuring the performance of AC pavement. Whether it is accurately estimated or not directly affects the reliability of pavement design. However, from the actual effect, the performance of the MEPDG model is not optimistic because its transfer function cannot predict the IRI of AC pavement well.

To solve this problem, researchers have been using data collected from sites in various states to build new models or to optimize transfer functions for improving the accuracy of IRI prediction, which is called local calibration [10–14].

The IRI transfer function in MEPDG combines road distress with pavement materials, structure, and traffic conditions linearly through field measured data to predict the IRI over time for AC pavement. However, factors affecting IRI are widespread, especially nonlinear factors, so the linear model is too simple to provide a convincing prediction result. Moreover, Thompson, Barenberg et al. pointed out that the statistics and practical structural model constructed in this way cannot explain the change of IRI well, and that further feeding into the transfer function will not produce an accurate prediction [15]. To improve prediction performance, we need to establish a new model that can cover various influencing factors (including nonlinearity) and build the relationship between IRI and multiple parameters.

On this topic, relevant researchers worldwide have conducted many studies and analyses, so this is not a new concept that being presented here [16–21]. The artificial intelligence (AI) algorithm can introduce nonlinear calculations, which are suitable for establishing AC road distress prediction models. For example, In 2003, Lin et al. found a three-layer ANN model by using the collected data for deep learning, in which the number of neurons of the input layer, hidden layer, and output layer was 14, 6, and 1, respectively, and analyzed the correlation between variables [18]. To explore the superiority of the ANN model compared to a single linear or nonlinear regression model, Chandra et al. designed three models. The results of their work showed that the IRI prediction performance of the ANN model is significantly better than the other two models regardless of whether it is used on the training or testing set [22]. However, the ANN model may not always perform well, especially in the face of deeper ANN structures and huge input data, where overfitting (the model performed well on the training set, but poor on the testing set) is more likely to occur, which leads to the decrease of the model's prediction ability [23,24].

Besides ANN, Freund and Schapire [25] proposed the AdaBoost model based on the boosting algorithm in 1997, which is one of the most widely used and researched algorithms. In particular, through 300 rounds of boosting tests Schapire [26] indicated that AdaBoost often did not find the overfitting phenomenon, apparently in direct contradiction of what was predicted by the Vapnik–Chervonenkis [27,28] (VC) theory, which ensures that the model has excellent and stable prediction performance. Thus, several recent works have proposed AdaBoost in order to compensate for pavement engineering distress. For example, In 2018, Wang, Yang, et al. used the AdaBoost algorithm to detect cracks on cement pavement and AC pavement, and the results were able to cover more than 83% of the crack length [29]. Hoang and Nguyen proposed an alternative method to automatically conduct periodic surveys of road conditions through image processing based on AdaBoost and achieved a high crack classification accuracy of about 90% [30]. Raveling is one of the most common forms of asphalt pavement distress on American highway pavements. Based on AdaBoost, after large-scale verification and refinement, the loose detection method developed by Tsai, Zhao, et al. had been successfully applied to the entire Georgia interstate highway system [31]. Compared to the application of AdaBoost in these aspects of road engineering, IRI predictions have higher similarities because the unevenness of AC pavement is caused by factors, including cracks, ruts, and raveling, etc., which contain some linear or nonlinear relationships. AdaBoost can explain the existence of these relationships through complex calculations, which provides the possibility of its application in IRI prediction modeling.

This study develops an AdaBoost-based IRI estimation model that utilizes the 4265 records from the Long-Term Pavement Performance (LTPP), including the pavement thickness, service age, average annual daily truck traffic (AADTT), gator crack, etc. Compared to the LR model from the MEPDG, the novelty of this model is that it comprehensively considers the input variables and introduces nonlinear factors. The results reveal that the AdaBoost regression (ABR) model has better fitting performance and has a lower error rate

than the LR model on both the training and testing sets. Therefore, the ABR model can better predict IRI, and it is also easy to interpret. By analyzing the importance of variables, there are many additional crucial factors, such as raveling and bleeding, that affect IRI, leading to the LR model's weak performance.

2. IRI Transfer Function

2.1. Definitions

To better perform local calibration, that is, to adapt the IRI transfer function to the requirements of different regions, LTPP has been collecting research quality pavement performance data from in-service test sections across the U.S. and Canada. The calibration process is to choose different coefficients to obtain the equation that best represents the actual situation. From the point of view of mathematical relationships, it is to select appropriate coefficients to obtain specific IRI calculation equations. Equation (1) gives the transfer function calculated by the IRI in the MEPDG. When calibrating locally, we generally need to go through two processes. The first step is to determine the value of $C_{1,2,3,4}$, and the second step is linearly combining these coefficients with the road parameters by substituting them into the transfer function.

$$\begin{aligned} IRI &= IRI_0 + C_1(RD) + C_2(FC_{\text{Total}}) + C_3(TC) + C_4(SF) \\ SF &= Age^{1.5} \{ \ln[(Precip + 1)(FI + 1)p_{02}] \} + \{ \ln[(Precip + 1)(PI + 1)p_{200}] \} \end{aligned} \quad (1)$$

where IRI_0 is the initial IRI after construction, SF is the site factor, FC_{Total} is the area of fatigue cracking, TC is the length of transverse cracking, RD is the average rut depth, $C_{1,2,3,4}$ are the calibration factors, Age is the pavement age, PI is the percent plasticity index of the soil, FI is the average annual freezing index, $Precip$ is the average annual precipitation or rainfall, p_{02} is the percent passing the 0.02 mm sieve, and p_{200} is the percent passing the 0.0075 mm sieve.

2.2. IRI Local Calibration Efforts

Table 1 shows some fitting parameters of the IRI calibration results, where R^2 and MSE represent the correlation coefficient and the mean square error, respectively. R^2 is also called the coefficient of determination, which defines how many future samples may be expected by the regression model. Moreover, it also means that the variance ratio of the predicted response variable in the variable can be explored [32]. Its value ranges from 0 to 1.0, where 1.0 is the best. MSE is a measure that reflects the degree of difference between the estimated and the actual value. Equation (2) gives the specific calculation method of R^2 and MSE. From the records listed in Table 1, it can be seen that the results of these calibrations are not satisfactory. For regions such as New Mexico, Arkansas, Kansas, and Ontario (Canada), the R^2 is below 0.5, which means that these models cannot explain more than half of the sample data. Specifically in New Mexico, the MSE is as high as 0.12. Even if the R^2 in some regions exceeds 0.5, such as in Arizona and Iowa, these indicators are only based on training data. In practice, it cannot behave like this in the face of unknown samples.

$$\begin{aligned} R^2 &= 1 - \frac{\sum (y - \hat{y})^2}{\sum (y - \bar{y})^2} \\ MSE &= \frac{1}{n} \sum (y - \hat{y})^2 \end{aligned} \quad (2)$$

where y is the measured value, \hat{y} is the predicted value, and n is the total number of samples, R^2 is the coefficient of determination, and MSE is the mean square error.

Table 1. A summary of efforts in calibrating IRI prediction.

Literature	Region	R^2	MSE	Samples
AASHTO [33]	National (U.S.)	0.5118	0.0249	244
Schram, Abdelrahman [34]	National (U.S.)	0.621	0.0235	670
Tarefder, R.-R. [10]	New Mexico (U.S.)	0.326	0.12	85
Xiao and Wang [35]	Arkansas (U.S.)	0.476	0.063	193
Sufian [36]	Kansas (U.S.)	0.22	0.052	90
Souliman, M. [37]	Arizona (U.S.)	0.543	0.0355	178
Ceylan, K. [38]	Iowa (U.S.)	0.685	0.0196	65
Gulfam, Y. [13]	Ontario (Canada)	0.438	0.0775	90
Yuan, L. [39]	Ontario (Canada)	0.578	0.03	15

3. Overviews of the AdaBoost Method

3.1. Decision Tree

The decision tree is a practical machine learning classifier that requires supervised learning, which trains a given sample when the result is already known [40–42]. As shown in Figure 1, the decision tree mainly consists of the root node, decision node, and leaf node. The sample starts from the root node and is classified according to the rules of each layer until it can no longer be divided. When using a decision tree, you first need to construct the root node, which contains all of the training data. Second, an optimal feature needs to be selected, and the training data need to be divided into subsets according to this feature so that each subset can achieve the best classification under the current conditions. Next, the leaf nodes need to be constructed for the subsets that have been correctly classified, and these leaf nodes need to be assigned to the corresponding leaf nodes. If there are subsets that cannot be correctly classified, then new optional features need to be selected for them. These need to be continued to be divided, and corresponding tree nodes need to continue to be built until all of the training subsets are correctly classified or until there are not suitable features. Finally, each subset needs to be assigned to a leaf node and needs to have a clear class. Commonly used decision trees are ID3 [43,44], C4.5 [45,46] and the classification and regression tree (CART) [47–50]. ID3 calculates the information gain of all of the possible features and selects the maximum value as the feature of the node. Then, leaf nodes need to be constructed from the different features in order to recursively generate decision trees for the leaf nodes until there are no features left to be chosen. However, such mechanical recursive calculations will not continue until the tree generated is often very accurate in the classification of the training data but not as accurate on the test set. That is, overfitting will occur. C4.5 inherits the advantages of ID3 and has been improved. Pruning (cutting some subtrees or leaf nodes from the tree that has been generated and uses the root node or parent node as the new leaf node to simplify the classification tree model) is conducted during the tree construction process, and the classification rules that are generated are easy to understand and have a high accuracy rate. However, in constructing the tree, the data set needs to be scanned and sorted multiple times, which leads to the inefficiency of the algorithm. Regardless of ID3 or C4.5, the main solution is the classification problem, which is the processing of discrete values. CART has made significant improvements based on the C4.5 model and can handle continuous values in regression problems. In the pruning process, the generated tree is pruned with the verification data set, and the optimal subtree is selected, thereby effectively reducing the occurrence of overfitting. Moreover, CART is suitable for large-scale data sets, especially considering that the more complex the sample and the more variables that there are, the more significant its superiority is. Since CART can solve two types of problems: classification and regression problems, it is used as a weak classifier for IRI prediction.

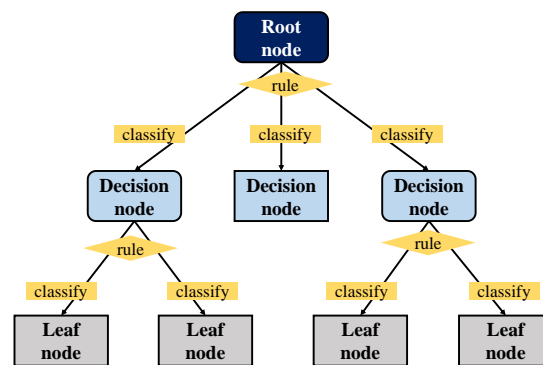


Figure 1. Decision tree.

3.2. AdaBoost

A single decision tree is called a weak learner because of its limited capabilities. Researchers imagine whether a strong learner can be obtained if multiple weak learners are combined. Schapire [51] proved this conjecture in 1990 and laid the foundation for the boosting algorithm, which combines multiple weak learners in series. As shown in Equation (3), each time it adds a new tree model to the whole, the general tree will be eliminated, and only the strongest tree will be added. In this way, with the accumulation of iterative calculations, the overall model performance will gradually improve. However, there is a problem here. After the first basic tree model is obtained, some samples on the data set are correctly classified, but some are wrong. The AdaBoost algorithm is a simple weak classification algorithm improvement process, which improves data classification ability through continuous training. The first weak classifier is obtained by learning the training samples, and the wrong samples are combined with the untrained data to form a new training sample. Furthermore, the second weak classifier is obtained by learning this sample. The wrong sample is combined with the untrained data to form another new training sample, which can be trained to obtain the third weak classifier. After repeating this process many times, we can finally obtain the improved robust classifier. To increase the number of correct classifications, the AdaBoost algorithm gives different weights to the samples [52]. The correctly classified samples are provided relatively low weights, and the wrong ones must be increased, which forces the model to pay more attention to the misclassified samples [53]. Figure 2 describes the overall calculation process of the AdaBoost algorithm. When training each basic tree model, the weight distribution of each sample in the data set needs to be adjusted. Since each training data will change, the training results will also be different, and finally, all of the results are summed [26].

$$F_n(x) = F_{n-1}(x) + \operatorname{argmin}_h \sum_{i=1}^n L(y_i, F_{n-1}(x_i) + h(x_i)) \quad (3)$$

where $F_n(x)$ is the overall model, $F_{n-1}(x)$ is the overall obtained in the previous round, y_i is the prediction result of the i -th tree, and $h(x_i)$ is the newly added tree.

3.3. Framework of the ABR Model

3.3.1. The Process of Building ABR Model

This section explains the overall establishment of the ABR model for estimating road IRI by using sample data including 21 (20 features and 1 target) columns of data such as the initial IRI (IRI_0), time of service, the total thickness of the pavement, AADTT, etc. The data used for training is the most original, and none of them have been modified by an algorithm in order to prevent damage to the model's practicality. As illustrated in Figure 3, the ABR framework consists of four parts: (1) randomly splitting training instances into training and testing subsets, (2) training an AdaBoost model on the training subsets, (3)

making predictions on the testing subset, and (4) comparing the testing predictions to the testing targets to assess accuracy.

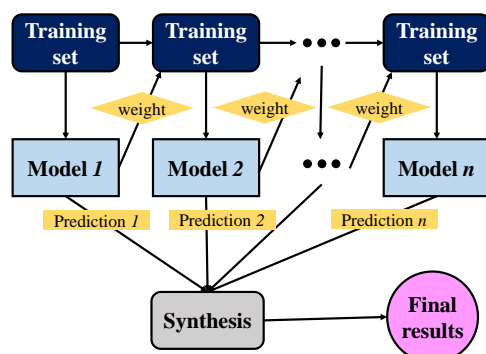


Figure 2. AdaBoost algorithm calculation process.

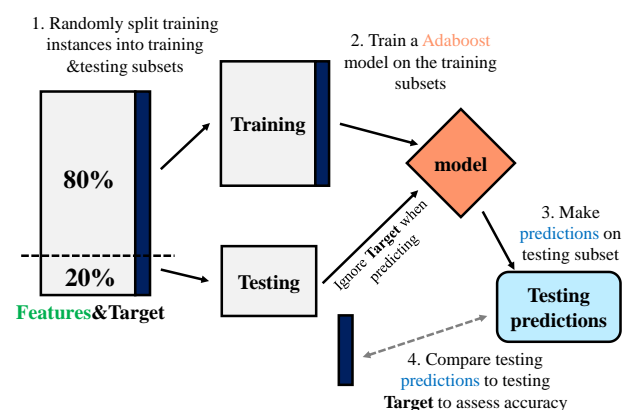


Figure 3. The AdaBoost model building process.

In the first step, the processed data are randomly divided into two data sets, employing one of them as the training set (80%) for model building. The other part is used as the testing set (20%) for model evaluation. The second step is to use the data in the training set for model training, during which cross-validation methods will be used for validation. In the third step, the build model is used to predict the samples on the testing set. In the last step, the difference between the the predicted value and the actual value to judge the performance of the model.

3.3.2. Real AdaBoost or Gentle AdaBoost

To solve the IRI prediction regression problem, as mentioned above, CART is selected as the weak learner. On this basis, the commonly used AdaBoost models are the Real and Gentle algorithms. Real AdaBoost [54,55] uses a logarithmic function to map to the real number domain after each weak classifier outputs the probability that the sample belongs to a specific class, and Gentle AdaBoost [56] performs a weighted regression based on the least-squares in each iteration [57]. The difference between the two algorithms is that Real is mainly used for classification problems, while Gentle is good at solving regression problems. IRI prediction is a regression problem, so Gentle is more suitable for modeling. To verify this judgment, we attempted to compare the MSE of the two models by establishing 2000 estimators (trees), and the results are shown in Figure 4. Regardless of whether it is used on the training set or the testing set, Gentle has a lower error rate, and the data fluctuation range is small. After about 200 calculation iterations, it can quickly converge, and the error curve tends to be stable.

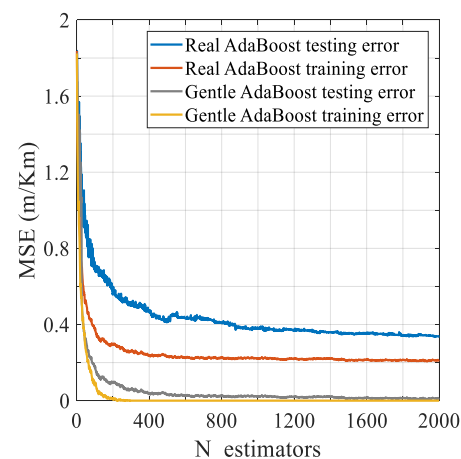


Figure 4. Real and Gentle error decay curve.

3.3.3. Loss Function

In the model, the predicted value of the regression equation and the actual value of the sample points are not one-to-one. There will be an error between each actual and predicted value, which is usually called an error term. We hope that the difference between the predicted formula and the actual value is as small as possible, so we define a way to measure the quality of the model, that is, the loss function (used to express the degree of difference between the prediction and the actual data). The loss function quantifies this error, and whether its selection is appropriate can affect subsequent optimization work. In the field of machine learning, common loss functions are linear, square, and exponential. We tried to establish 50 estimators to determine the appropriate loss function, and the results are shown in Figure 5, wherein the exponential score is better than the other two functions. After about 20 iterations in the training set, the curve tends to be stable.

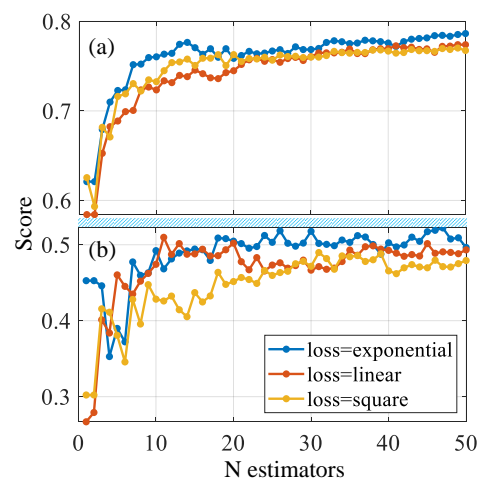


Figure 5. Different loss function score curves: (a) training set and (b) testing set.

4. Data Preparation

4.1. Data Acquisition from LTPP

Collecting comprehensive and accurately labeled data is necessary for model training. Hence, the data used comes from LTPP, which gathers research quality pavement performance data from in-service test sections across the United States and Canada. This study initially obtained more than 11,000 IRI records from 62 states in the United States and Canada. However, due to the lack of average annual daily truck traffic (AADTT) data, only 4265 samples with AADTT data were used in the end. Figure 6 depicts the source distribution of these data.

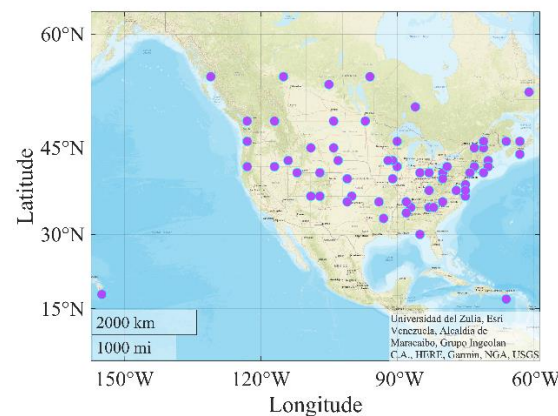


Figure 6. Distribution of data sources.

4.2. Predictor Variables Selection

The collected samples contain many variables, but not every variable can improve the model's performance, and unrelated variables reduce the accuracy and computational efficiency of the model. Equation (4) gives the calculation method for the variable importance. First, the accuracy of the out of bag (OOB) data set that is part of the data set is calculated, and the input variables are randomly arranged, and the OOB accuracy of the decision tree is recalculated to obtain the difference between the two arrangements [58]. To remove irrelevant variables, we selected relative importance to evaluate the importance of variables and eliminated variables with values less than zero. The steps are as follows: (1) calculate the importance values of all variables and sort them in descending order of relative importance values, (2) divide the variables evenly into n groups and keep the last set of variable rankings and values, (3) calculate the remaining variable importance values and sort them in descending order of relative importance value, (4) repeat step (3) until the calculation of these groups of variables is completed, and (5) repeat the simulation 100 times and take the mean value of relative importance of these 100 times as the variable importance value.

$$VI = \frac{1}{N} \sum_k \left(errOOB_k^l - errOOB \right) \quad (4)$$

where N is the number of trees, $errOOB_k^l$ is the error of predictor on the permuted sample, and $errOOB$ is the error of a single tree in the OOB sample.

The above process results show that the variable importance values of average wind velocity, average temperature, average cloud cover, and shortwave surface average are all less than zero, so they are all eliminated. Moreover, to further optimize the input variables, we introduced stepwise regression analysis to filter the variables. The basic idea of stepwise regression is to introduce variables, test them one by one, and eliminate variables that are no longer significant due to new variables. If neither significant variable is selected into the equation, and if all insignificant independent variables are excluded from the regression equation, the process ends. The specific process is shown in Figure 7. We can see that after the execution of this process, by comparing the OOB values of the variables before and after, the unimportant variables can be eliminated, and the minimum OOB can be obtained. After testing, we successfully eliminated the pumping length, pavement width, air voids, resilient modulus, aggregate percentage, and binder content of the four variables. Therefore, the model ultimately leaves 20 relevant variables. Table 2 describes the specific content of these variables in terms of structure, performance, traffic, and climate.

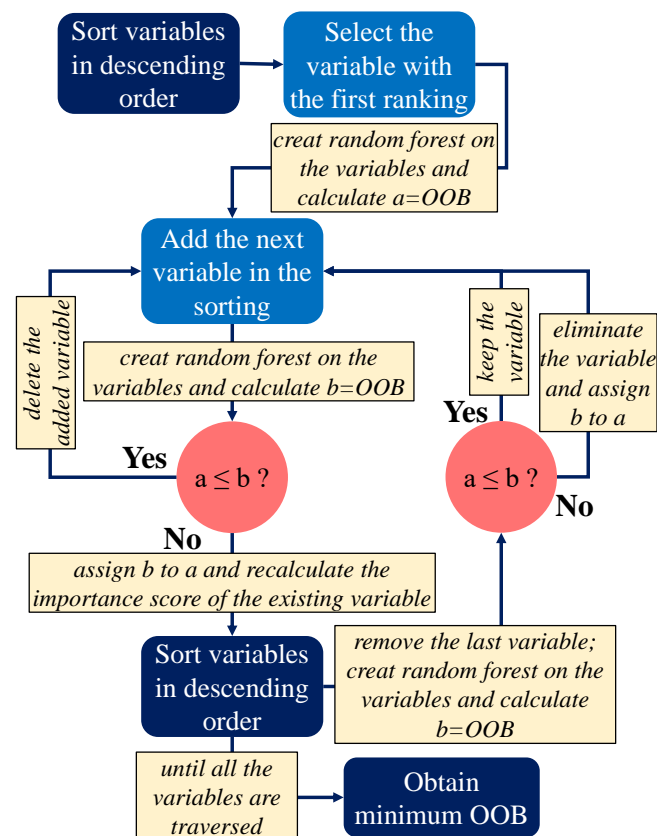


Figure 7. Stepwise regression.

Table 2. Explanation of input variables.

Category	Variable	Explanation	Unit
Structure	Thickness	The total thickness of pavement	mm
	Gator	Area of alligator (fatigue) cracking.	m ²
Performance	Block	Area of block cracking.	m ²
	Edge	Length of low severity edge cracking.	m
	Long-wp	Length longitudinal cracking in wheel path.	m
	Long-nwp	Length non-wheel path longitudinal cracking.	m
	Transverse	Length of transverse cracking.	m
	Patch	Area of patching	m ²
	Potholes	Area of potholes	m ²
	Shoving	Area of shoving, localized longitudinal displacement of the pavement surface.	m ²
	Bleeding	Presence of excess asphalt on the pavement surface which may create a shiny, glass-like reflective surface.	m ²
	Polish	Area of polished aggregate (binder worn away to expose coarse aggregate).	m ²
	Raveling	Wearing away of the pavement surface caused by the dislodging of aggregate particles and loss of asphalt binder.	m ²
	Rut	Depth of rut	mm
	IRI ₀	The first IRI when the road is put into use	m/Km
	Age	Road service time	\
Traffic	AADTT	Average annual daily truck traffic	\
	ESAL	The annual average of equivalent single axle load in the LTPP lane.	\

Table 2. Cont.

Category	Variable	Explanation	Unit
Climatic	Freeze	Annual average freeze index	\
	Precipitation	Annual average precipitation	mm

4.3. Data Set Allocation

However, no matter how good the model is in all aspects of the training process, our ultimate goal is to achieve accurate prediction results in practice. Usually, the error that we produce when the model is applied to reality is called the generalization error, so we need to reduce the generalization error to improve the environmental adaptability of the model. Nevertheless, it is not realistic to understand the model's generalization ability directly by using the generalization error as a signal. This requires frequent interaction between the model and the reality, which increases the difficulty and cost of modeling. A better way is to split the data into two parts: the training set and the testing set. The data in the training set are used for model training. Then, the error of the trained model on the testing set is calculated as an approximation of the generalization error, so we only need to reduce the model's error on the testing set when optimizing the model. In this study, we randomly divided the data into two parts, where 3412 (80%) samples are used for model training, and the remaining 853 (20%) are used for testing.

5. ABR Model Construction

The transfer function given in MEPDG considers 10 factors as input variables when predicting IRI. On the contrary, this ABR model also considers other essential factors for comprehensive construction, listed in Table 1.

As mentioned above, this model training uses 4265 IRI samples from LTPP, and each sample contains 20 variables and 1 measured value of IRI. To achieve better prediction results, selecting appropriate hyperparameters, including CART parameters and ABR frame parameters, is necessary. The parameters that CART needs to determine mainly include the Max depth, Min samples split, and Min samples leaf nodes. The frame parameters of ABR include base estimator, loss, N estimators, and learning rate. Considering the limited number of samples, it would be very wasteful to set the verification set separately. To solve this problem, we used four-fold cross-validation, as shown in Figure 8. When verifying a certain result, the whole process needs to be divided into four steps. In the first step, use the first three iterations as the training set and the last iteration as the validation set to achieve a result. By analogy, each time uses a different three from the training set, and the rest are used as the validation set. After completing the four steps using this method, four results are obtained, each of which corresponds to each small part, and the combination contains all of the data in the original training set. The final four results are then averaged to obtain the final model evaluation result. Cross-validation can better evaluate the model and make the results more accurate.

In theory, to find a suitable combination of hyperparameters, all possible values should be listed in turn, but this will consume a lot of time for the traversal and reduce efficiency. In response to this problem, a common strategy is a random search, which randomly tests all of the possible values when optimizing hyperparameters to find a position that is roughly close enough to the optimal solution. Although its results are not as accurate as a comprehensive search, it dramatically improves the iterative efficiency, especially when facing large-scale data sets. When considering the hyperparameters of CART, we limit Max depth to between 4 and 16 and delineate the range of Min samples split from 2 to 10 because Max depth and Min samples split that are will also reduce modeling efficiency and are not conducive to learning the characteristics of the sample. For the choice of Min samples leaf, we considered a larger value in the early stage of modeling, but it caused a severe overfitting problem, so it was finally determined to be between 2 and 5. Moreover, a too large learning rate (step size) will cause the model to miss the minimum value when

calculating the loss, so the learning rate of this model is set to be low (0.001). After defining the range of these hyperparameters, the random search method (in the Python language, using the RandomizedSearchCV function under the random forest class) is then used to determine the specific value. Based on the fast convergence characteristics of the AdaBoost algorithm, the model’s error stabilizes after 80 iterations. Table 3 lists the names and specific values of these parameters.

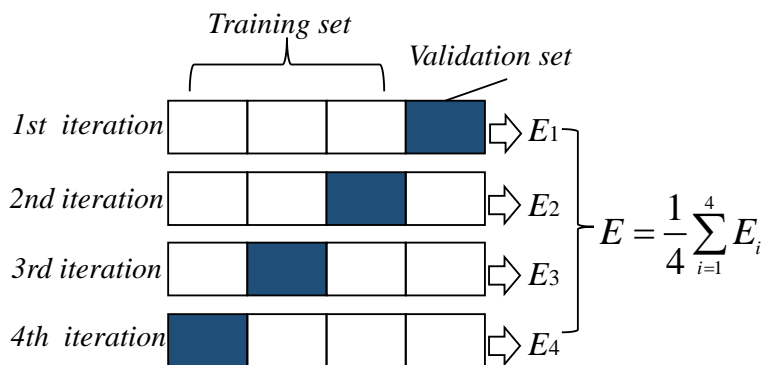


Figure 8. Cross-validation.

Table 3. Selection of model parameters.

	Parameter	Explanation	Value
CART	Max depth	maximum depth of decision tree	12
	Min samples split	minimum number of samples required for subdividing internal nodes	8
	Min samples leaf	minimum number of samples for leaf nodes	3
ABR frame	Base estimator	weak regression learner	CART
	Loss	loss function, there are three choices of linear, square and exponential	exponential
	N estimators	maximum number of iterations of the weak learner	80
	Learning rate	the step size of the update parameter, too small will slow down the iteration speed	0.001

6. Model Result and Analysis

6.1. Model Performance Evaluation

After going through the building steps of the above model, we need to evaluate the model’s quality. This time, the selected metrics are the R^2 and MSE. To better evaluate the ABR model, the training results will be compared with the LR model in the MEPDG, and the results are shown in Figure 9. The LR had an R^2 of 0.5588 and a MSE of 0.087 in the training set, versus 0.5118 and 0.0249, respectively, in the testing set. As the R^2 and MSE shown, this simple ABR model achieved a much better predictive performance than LR. Compared to the LR, the training R^2 of the ABR is improved by more than 78%, while the MSE decreased by 98%. In contrast to the LR, the testing R^2 of the ABR is also improved by 90%, while its MSE is reduced by 65%. Table 4 lists the more detailed results of the two models. The results demonstrate that a simple ABR can be more predictive and capable of handling the IRI than a LR model.

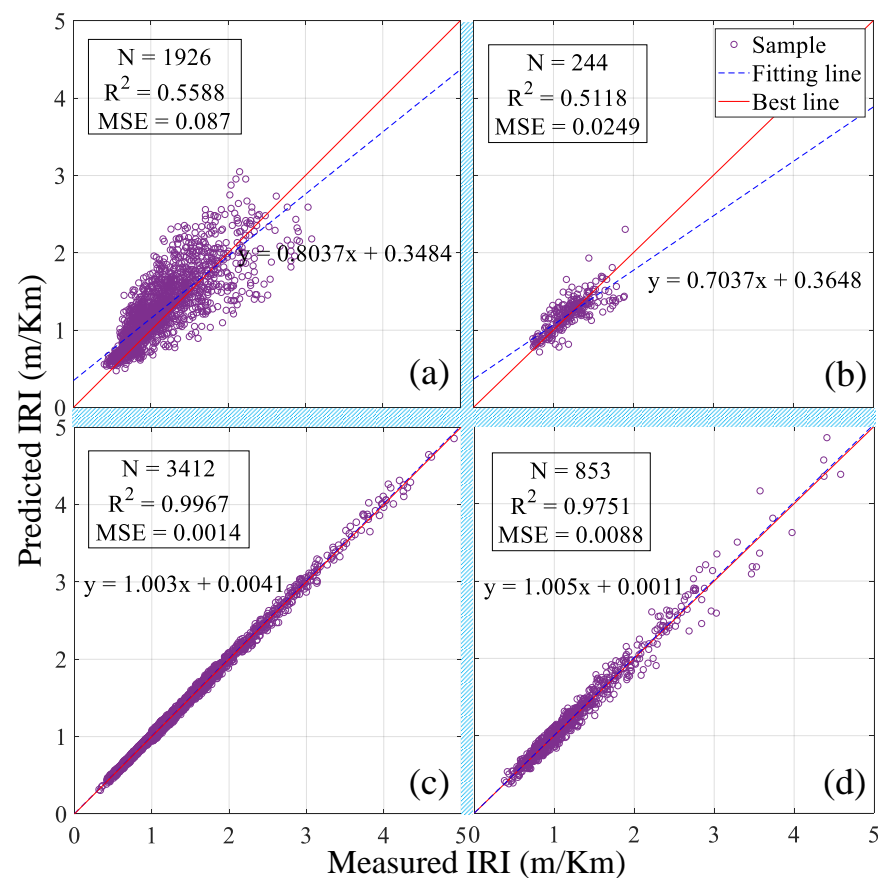


Figure 9. IRI predicted and measured values: (a) LR model on the training set, (b) LR model on the testing set, (c) ABR model on the training set, and (d) ABR model on the testing set.

Table 4. Performance comparison of LR and ABR.

Model		R^2	MSE	Fitting Equation
LR	Training set	0.5588	0.0870	$y = 0.8037x + 0.3484$
	Testing set	0.5118	0.0249	$y = 0.7037x + 0.3648$
ABR	Training set	0.9967	0.0014	$y = 1.0030x + 0.0041$
	Testing set	0.9751	0.0088	$y = 1.0050x + 0.0011$

6.2. Model Interpretation

Numerical analysis results show that ABR can accurately predict the IRI of asphalt concrete pavement, and it performs well in the test set so that the model performs well in preventing overfitting. However, the nature of it cannot be revealed well from the numerical results. We need to implement the actual interpretation of this model. To this end, we use the importance analysis method in the random forest module, which can help us understand the relationship between the input variables and the actual IRI to assume part of the interpretation of the model [59,60]. It can be seen from the descending order graph of variable importance in Figure 10 that IRI_0 is the most important factor affecting IRI, which is reflected in the IRI transfer function of MEPDG. The influence of the transverse cracks and AADTT is also very significant because the formation of transverse cracks will increase the vertical vibration of the driving vehicle, and the number of repeated actions of the vehicle will cause the permanent deformation of the road surface. Moreover, the traffic (ESAL), temperature (freeze index), and service age were also highly correlated with the IRI. Of course, some variables are easily overlooked, such as patches, which will cause a drop on the pavement to become uneven.

Further, Figure 11 draws a matrix scatter plot of the correlation between variables. Because there are too many variables to display, we have selected the most representative variables from each category for analysis. It can be seen that IRI_0 has a solid linear correlation with IRI, which is consistent with the conclusions we have previously obtained. In addition, other variables do not obey such an apparent linear relationship, indicating that nonlinear factors are widespread. This is also the key to the difference between ABR and MLR and why ABR is highly predictive.

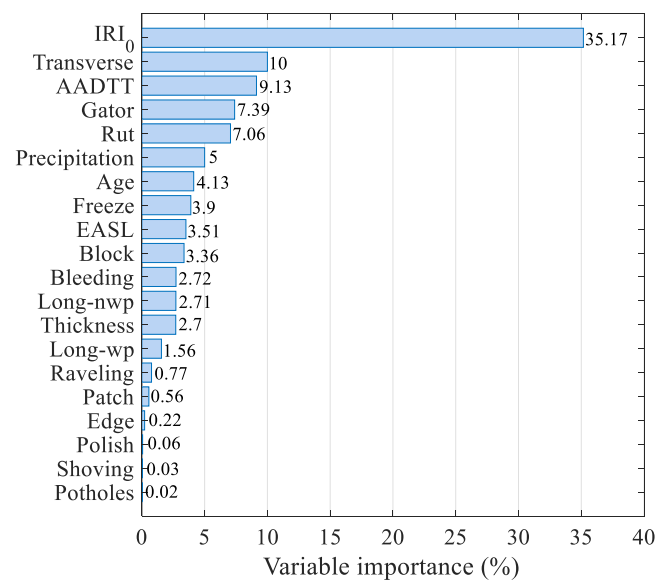


Figure 10. Importance of input variables.

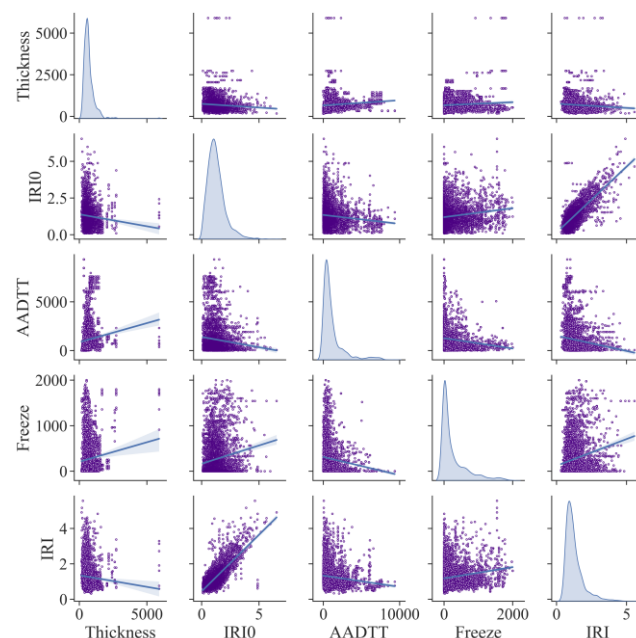


Figure 11. Variable correlation analysis matrix diagram.

7. Conclusions

In this paper, the authors have examined and analyzed an ABR model for IRI estimation in AC pavements, aiming to optimize its configuration and some influencing factors for maximizing the resulting quality and the estimation reliability. Further, several aspects of adjusting the accuracy of the ABR have been considered, including weak learners and

loss functions, etc., to arrive at recommendations for proper model optimization. The considered model data comes from the LTPP large-scale pavement information database and includes some parameters that affect the performance of the pavement structure, such as structure, climate, traffic, and performance variables. After the iterative calculation of 4265 samples from the LTPP and the analysis of the correlation between the variables, the following conclusions can be drawn:

- The R^2 and MSE of the ABR model on the testing set are 0.9571 and 0.0088, respectively. Hence, the ABR model has high accuracy and predictability, so the model's overfitting is well controlled;
- Compared to the LR, the testing R^2 of the ABR is improved by 90%, while its MSE is reduced by 65%. The results demonstrate that a simple ABR can be more predictive and capable than a LR model;
- By analyzing the importance of the variables to the model, we can see the degree of influence of each variable. IRI_0 is the largest, followed by transverse and AADTT. Moreover, there are many additional crucial factors such as raveling and bleeding that affect IRI, leading to the LR model's weak performance;
- One of the reasons for the low predictive ability of the LR model in MEPDG is that it does not consider nonlinear influencing factors, which can be well improved by an ABR model for future pavement design.

In addition, the numerical results prove that the IRI_0 (initial IRI) is the most important influencing factor, which is consistent with the transfer function in MEPDG so that the ABR model can be explained by practice. The results and discussion are helpful in optimizing the IRI prediction model to evaluate the performance of the pavement design structure. It is recommended to conduct further research work, hoping to collect more accurate road performance data through innovative high-performance testing equipment and expand the data set (in order to make the data input to the model better and more comprehensive) to further develop the potential of the model.

Author Contributions: Data curation, C.W. and S.X.; formal analysis, S.X.; investigation, C.W.; writing—original draft, S.X.; writing—review and editing, J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Islam, S.; Buttlar, W.G.; Aldunate, R.G.; Vavrik, W.R. Use of cellphone application to measure pavement roughness. In *T&DI Congress 2014: Planes, Trains, and Automobiles*; American Society of Civil Engineers: Reston, VA, USA, 2014.
2. Jeong, J.H.; Jo, H.; Ditzler, G. Convolutional neural networks for pavement roughness assessment using calibration-free vehicle dynamics. *Comput.-Aided Civil Infrastruct. Eng.* **2020**, *35*, 1209–1229. [[CrossRef](#)]
3. Chen, S.; Saeed, T.U.; Alqadhi, S.D.; Labi, S. Safety impacts of pavement surface roughness at two-lane and multi-lane highways: Accounting for heterogeneity and seemingly unrelated correlation across crash severities. *Transp. A Transp. Sci.* **2019**, *15*, 18–33. [[CrossRef](#)]
4. Islam, S.; Buttlar, W.G.; Aldunate, R.G.; Vavrik, W.R. Measurement of pavement roughness using android-based smartphone application. *Transp. Res. Record.* **2014**, *2457*, 30–38. [[CrossRef](#)]
5. Douangphachanh, V.; Oneyama, H. A study on the use of smartphones for road roughness condition estimation. *J. East. Asia Soc. Transp. Studies.* **2013**, *10*, 1551–1564.
6. González, A.; O'Brien, E.J.; Li, Y.-Y.; Cashell, K. The use of vehicle acceleration measurements to estimate road roughness. *Veh. Syst. Dyn.* **2008**, *46*, 483–499. [[CrossRef](#)]

7. Fujino, Y.; Kitagawa, K.; Furukawa, T.; Ishii, H. Development of vehicle intelligent monitoring system (VIMS). In *Smart Structures and Materials 2005: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*; International Society for Optics and Photonics: San Diego, CA, USA, 2005.
8. Nagayama, T.; Miyajima, A.; Kimura, S.; Shimada, Y.; Fujino, Y. Road condition evaluation using the vibration response of ordinary vehicles and synchronously recorded movies. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2013*; International Society for Optics and Photonics: San Diego, CA, USA, 2013.
9. Zhao, B.; Nagayama, T.; Toyoda, M.; Makhata, N.; Takahashi, M.; Ieiri, M. Vehicle model calibration in the frequency domain and its application to large-scale IRI estimation. *J. Disaster Res.* **2017**, *12*, 446–455. [[CrossRef](#)]
10. Tarefder, R.; Rodriguez-Ruiz, J.I. Local calibration of MEPDG for flexible pavements in New Mexico. *J. Transp. Eng.* **2013**, *139*, 981–991. [[CrossRef](#)]
11. El-Badawy, S.; Bayomy, F.; Awed, A. Performance of MEPDG dynamic modulus predictive models for asphalt concrete mixtures: Local calibration for Idaho. *J. Mater. Civ. Eng.* **2012**, *24*, 1412–1421. [[CrossRef](#)]
12. Kim, Y.R.; Jadoun, F.M.; Hou, T.; Muthadi, N. Local Calibration of the MEPDG for Flexible Pavement Design. Available online: <https://connect.ncdot.gov/projects/research/RNAProjDocs/2007-07FinalReport.pdf> (accessed on 19 August 2021).
13. Jannat, G.E.; Yuan, X.-X.; Shehata, M. Development of regression equations for local calibration of rutting and IRI as predicted by the MEPDG models for flexible pavements using Ontario’s long-term PMS data. *Int. J. Pavement Eng.* **2016**, *17*, 166–175. [[CrossRef](#)]
14. Hoegh, K.; Khazanovich, L.; Jensen, M. Local Calibration of Mechanistic–Empirical Pavement Design Guide Rutting Model: Minnesota Road Research Project Test Sections. *Transp. Res. Record* **2010**, *2180*, 130–141. [[CrossRef](#)]
15. Thompson, M.; Barenberg, E.; Carpenter, S.; Darter, M.; Dempsey, B.; Ioannides, A. *Calibrated Mechanistic Structural Analysis Procedures for Pavements. Volume I-Final Report; Volume II-Appendices*; Technical Report for the National Cooperative Highway Research Program (NCHRP): Urbana, IL, USA, 1990.
16. Ahmed, A.; Labi, S.; Li, Z.; Shields, T. Aggregate and disaggregate statistical evaluation of the performance-based effectiveness of long-term pavement performance specific pavement study-5 (LTPP SPS-5) flexible pavement rehabilitation treatments. *Struct. Infrastruct. Eng.* **2013**, *9*, 172–187. [[CrossRef](#)]
17. Choi, J.H.; Adams, T.M.; Bahia, H.U. Pavement Roughness Modeling Using Back-Propagation Neural Networks. *Comput.-Aided Civ. Infrastruct. Eng.* **2004**, *19*, 295–303. [[CrossRef](#)]
18. Lin, J.-D.; Yau, J.-T.; Hsiao, L.-H. Correlation analysis between international roughness index (IRI) and pavement distress by neural network. In Proceedings of the 82nd Annual Meeting of the Transportation Research Board, Washington, DC, USA, 12–16 January 2003.
19. Meegoda, J.N.; Gao, S. Roughness progression model for asphalt pavements using long-term pavement performance data. *J. Transp. Eng.* **2014**, *140*, 04014037. [[CrossRef](#)]
20. Nitsche, P.; Stütz, R.; Kammer, M.; Maurer, P. Comparison of machine learning methods for evaluating pavement roughness based on vehicle response. *J. Comput. Civ. Eng.* **2014**, *28*, 04014015. [[CrossRef](#)]
21. Paterson, W. A transferable causal model for predicting roughness progression in flexible pavements. *Transp. Res. Record* **1989**, *1215*, 70–84.
22. Chandra, S.; Sekhar, C.R.; Bharti, A.K.; Kangadurai, B. Relationship between pavement roughness and distress parameters for Indian highways. *J. Transp. Eng.* **2013**, *139*, 467–475. [[CrossRef](#)]
23. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, UK, 2016.
24. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006.
25. Freund, Y.; Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [[CrossRef](#)]
26. Schapire, R.E. Explaining adaboost. In *Empirical Inference*; Springer: Berlin/Heidelberg, Germany, 2013.
27. Vapnik, V.N.; Chervonenkis, A.Y. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of Complexity*; Springer: Berlin/Heidelberg, Germany, 2015.
28. Vapnik, V. *Estimation of Dependences Based on Empirical Data*; Springer Science & Business Media: New York, NY, USA, 2006.
29. Wang, S.; Yang, F.; Cheng, Y.; Yang, Y.; Wang, Y. *Adaboost-based Crack Detection Method for Pavement*. IOP Conference Series: Earth and Environmental Science; IOP Publishing: Bristol, Britain, 2018.
30. Hoang, N.-D.; Nguyen, Q.-L. Fast local laplacian-based steerable and sobel filters integrated with adaptive boosting classification tree for automatic recognition of asphalt pavement cracks. *Adv. Civ. Eng.* **2018**, *2018*, 1–17. [[CrossRef](#)]
31. Tsai, Y.-C.; Zhao, Y.; Pop-Stefanov, B.; Chatterjee, A. Automatically detect and classify asphalt pavement raveling severity using 3D technology and machine learning. *Int. J. Pavement Res. Technol.* **2021**, *14*, 487–495. [[CrossRef](#)]
32. Haroon, D. *Python Machine Learning Case Studies*; Springer: Berlin/Heidelberg, Germany, 2017.
33. Transportation Officials. *Mechanistic-Empirical Pavement Design Guide: A Manual of Practice*; AASHTO: Washington, DC, USA, 2008.
34. Schram, S.; Abdelrahman, M. Improving prediction accuracy in mechanistic–empirical pavement design guide. *Transp. Res. Record* **2006**, *1947*, 59–68. [[CrossRef](#)]
35. Xiao, D.; Wang, K. *Calibration of the MEPDG for Flexible Pavement Design in Arkansas*; SAGE: Washington, DC, USA, 2011.

36. Sufian, A.A. Local Calibration of the Mechanistic Empirical Pavement Design Guide for Kansas. Ph.D. Thesis, Kansas State University, Manhattan, Kansas, 2016.
37. Souliman, M.; Mamlouk, M.; El-Basyouny, M.; Zapata, C. *Calibration of the AASHTO MEPDG for Flexible Pavement for Arizona Conditions*; Transportation Research Board: Washington, DC, USA, 2010.
38. Ceylan, H.; Kim, S.; Gopalakrishnan, K.; Ma, D. *Iowa Calibration of MEPDG Performance Prediction Models*; Iowa State University: Ames, IA, USA, 2013.
39. Yuan, X.-X.; Lee, W.; Li, N. Ontario's Local Calibration of the MEPDG Distress and Performance Models for Flexible Roads: A Summary. In Proceedings of the TAC 2017: Investing in Transportation: Building Canada's Economy—2017 Conference and Exhibition of the Transportation Association of Canada, Ottawa, ON, Canada, 13–14 November 2017.
40. Myles, A.J.; Feudale, R.N.; Liu, Y.; Woody, N.A.; Brown, S.D. An introduction to decision tree modeling. *J. Chemom. A J. Chemom. Society* **2004**, *18*, 275–285. [[CrossRef](#)]
41. Song, Y.-Y.; Ying, L. Decision tree methods: Applications for classification and prediction. *Shanghai Arch. Psychiatry* **2015**, *27*, 130.
42. Quinlan, J.R. Learning decision tree classifiers. *ACM Comput. Surv.* **1996**, *28*, 71–72. [[CrossRef](#)]
43. Hssina, B.; Merbouha, A.; Ezzikouri, H.; Erritali, M. A comparative study of decision tree ID3 and C4. 5. *Int. J. Adv. Comput. Sci. Appl.* **2014**, *4*, 13–19.
44. Jin, C.; De-Lin, L.; Fen-Xiang, M. An improved ID3 decision tree algorithm. In Proceedings of the 2009 4th International Conference on Computer Science & Education, Nanning, China, 25–28 July 2009.
45. Quinlan, J.R. *C4. 5: Programs for Machine Learning*; Elsevier: Amsterdam, The Netherlands, 2014.
46. Salzberg, S.L. *C4. 5: Programs for Machine Learning* by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc. Available online: <http://server3.eca.ir/isi/forum/Programs%20for%20Machine%20Learning.pdf> (accessed on 19 August 2021).
47. Loh, W.Y. Classification and regression trees. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2011**, *1*, 14–23. [[CrossRef](#)]
48. Moisen, G. Classification and regression trees. In *Encyclopedia of Ecology*; Elsevier: Oxford, UK; pp. 582–588.
49. Speybroeck, N. Classification and regression trees. *Int. J. Public Health* **2012**, *57*, 243–246. [[CrossRef](#)]
50. Timofeev, R. Classification and regression trees (CART) theory and applications. Master's Thesis, Humboldt University, Berlin, Germany, 2004.
51. Schapire, R.E. The strength of weak learnability. *Mach. Learning* **1990**, *5*, 197–227. [[CrossRef](#)]
52. Freund, Y.; Schapire, R.; Abe, N. A short introduction to boosting. *J.-Jpn. Soc. Artif. Intell.* **1999**, *14*, 1612.
53. Ying, C.; Qi-Guang, M.; Jia-Chen, L.; Lin, G. Advance and prospects of AdaBoost algorithm. *Acta Autom. Sinica* **2013**, *39*, 745–758.
54. Wu, B.; Ai, H.; Huang, C.; Lao, S. Fast rotation invariant multi-view face detection based on real adaboost. In Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, Seoul, Korea, 17–19 May 2004.
55. Wu, S.; Nagahashi, H. Parameterized adaboost: Introducing a parameter to speed up the training of real adaboost. *IEEE Signal Process. Lett.* **2014**, *21*, 687–691. [[CrossRef](#)]
56. Ho, W.T.; Lim, H.W.; Tay, Y.H. Two-stage license plate detection using gentle Adaboost and SIFT-SVM. In Proceedings of the 2009 First Asian Conference on Intelligent Information and Database Systems, Dong Hoi, Vietnam, 1–3 April 2009.
57. Shahraki, A.; Abbasi, M.; Haugen, Ø. Boosting algorithms for network intrusion detection: A comparative evaluation of Real AdaBoost, Gentle AdaBoost and Modest AdaBoost. *Eng. Appl. Artif. Intell.* **2020**, *94*, 103770. [[CrossRef](#)]
58. Zhang, Y.-M.; Wang, H.; Mao, J.-X.; Xu, Z.-D.; Zhang, Y.-F. Probabilistic Framework with Bayesian Optimization for Predicting Typhoon-Induced Dynamic Responses of a Long-Span Bridge. *J. Struct. Eng.* **2021**, *147*, 04020297. [[CrossRef](#)]
59. Strobl, C.; Boulesteix, A.-L.; Kneib, T.; Augustin, T.; Zeileis, A. Conditional variable importance for random forests. *BMC Bioinform.* **2008**, *9*, 1–11. [[CrossRef](#)] [[PubMed](#)]
60. Van der Laan, M.J. Statistical inference for variable importance. *Int. J. Biostat.* **2006**, *2*. [[CrossRef](#)]