




Article

# Assessment of Airport Pavement Condition Index (PCI) Using Machine Learning

Bertha Santos <sup>1,2,\*</sup> , André Studart <sup>1,2</sup>  and Pedro Almeida <sup>1,2</sup> 

<sup>1</sup> Department of Civil Engineering and Architecture, University of Beira Interior, 6200-358 Covilhã, Portugal; andre.studart@ubi.pt (A.S.); galmeida@ubi.pt (P.A.)

<sup>2</sup> GeoBioTec, University of Beira Interior, 6200-358 Covilhã, Portugal

\* Correspondence: bsantos@ubi.pt

## Abstract

Pavement condition assessment is a fundamental aspect of airport pavement management systems (APMS) for ensuring safe and efficient airport operations. However, conventional methods, which rely on extensive on-site inspections and complex calculations, are often time-consuming and resource-intensive. In response, Industry 4.0 has introduced machine learning (ML) as a powerful tool to streamline these processes. This study explores five ML algorithms (Linear Regression (LR), Decision Tree (DT), Random Forest (RF), Artificial Neural Network (ANN), and Support Vector Machine (SVM)) for predicting the Pavement Condition Index (PCI). Using basic alphanumeric distress data from three international airports, this study predicts both numerical PCI values (on a 0–100 scale) and categorical PCI values (3 and 7 condition classes). To address data imbalance, random oversampling (SMOTE—Synthetic Minority Oversampling Technique) and undersampling (RUS) were used. This study fills a critical knowledge gap by identifying the most effective algorithms for both numerical and categorical PCI determination, with a particular focus on validating class-based predictions using relatively small data samples. The results demonstrate that ML algorithms, particularly Random Forest, are highly effective at predicting both the numerical and the three-class PCI for the original database. However, accurate prediction of the seven-class PCI required the application of oversampling techniques, indicating that a larger, more balanced database is necessary for this detailed classification. Using 10-fold cross-validation, the successful models achieved excellent performance, yielding Kappa statistics between 0.88 and 0.93, an error rate of less than 7.17%, and an area under the ROC curve greater than 0.93. The approach not only significantly reduces the complexity and time required for PCI calculation, but it also makes the technology accessible, enabling resource-limited airports and smaller management entities to adopt advanced pavement management practices.

**Keywords:** airport pavement management system (APMS); pavement condition index (PCI); machine learning (ML); predictive modeling



Academic Editor: Mario Di Nardo

Received: 13 September 2025

Revised: 11 October 2025

Accepted: 17 October 2025

Published: 24 October 2025

**Citation:** Santos, B.; Studart, A.; Almeida, P. Assessment of Airport Pavement Condition Index (PCI) Using Machine Learning. *Appl. Syst. Innov.* **2025**, *8*, 162. <https://doi.org/10.3390/asi8060162>

**Copyright:** © 2025 by the authors.

Published by MDPI on behalf of the International Institute of Knowledge Innovation and Invention. 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

### 1.1. Framework

Airports play a vital role in society, connecting people around the world. The ever-growing world population, estimated to reach 9.7 billion by 2050 [1], requires infrastructures such as airports to withstand intense traffic and heavily loaded aircraft while ensuring the safety of users and operations. As a result, maintenance programs are in high demand to

ensure that airports remain operational and functional. However, budget shortfalls, non-preventive maintenance programs, and resource availability are often some of the factors that can limit the proper maintenance of airport pavements, especially the runway [2–4].

Several factors can impact the short- and long-term performance of airport pavements. These factors include subgrade bearing load capacity, the quality of pavement layer aggregates and binders, climate and drainage conditions, uneven distribution of landings and takeoffs across runway thresholds, and traffic growth rate [3,5–7]. All these ultimately influence the overall pavement design and performance. Although proper pavement design can retard the emergence and development of pathologies, wear will always occur, as pavement usage from repeated traffic load causes fatigue cracking at the bottom of the surfacing layer and permanent deformation at the surface of the subgrade layer, affecting the pavement's condition and support capacity. To effectively manage this inevitable decline and optimize maintenance strategies, modern technological solutions are required.

The current Industrial Revolution 4.0 technologies, such as Machine Learning (ML), which can be applied to various fields and process large amounts of data in record time, can be exploited in the targeted and timely definition of pavement maintenance programs.

This study addresses a critical knowledge gap by identifying the most effective algorithms for Pavement Condition Index (PCI) prediction. This study focuses on validating class-based predictions, which can be used to define specific maintenance and rehabilitation strategies, as well as their performance with relatively small data samples. To demonstrate the feasibility of using Machine Learning (ML) for PCI calculation, five algorithms are compared: Linear Regression (LR), Decision Tree (DT), Random Forest (RF), Artificial Neural Network (ANN), and Support Vector Machine (SVM). These algorithms can use basic alphanumeric distress data (type, severity, and density), which can be obtained through traditional inspections or other methods, to predict both numerical (on a 0–100 scale) and categorical PCIs (e.g., poor, fair, good, or a more granular seven-class scale). This approach significantly enhances time efficiency and reduces costs, making the technology accessible to smaller management entities and airports with limited financial, human, and technical resources. The software used is WEKA 3.8.6, with a database containing information from 261 pavement sample units from three international airports as input data.

The structure of this article is organized into four sections. Section 1 highlights the importance of pavement maintenance and the potential of using ML in this process, the main aspects of the pavement condition evaluation using PCI, and describes ML procedures and operation. Section 2 presents the study methodology. The main results from the applied ML algorithms to a case study are analyzed and discussed in Section 3, titled Case Study. Section 4 presents the conclusions of the study. Highlights, limitations, and directions for future work are also presented.

### 1.2. Pavement Condition Index (PCI)

The Pavement Condition Index (PCI), based on the ASTM D5340 [8], is the most widely adopted methodology for evaluating airport pavement conditions. This method involves visually identifying seventeen types of pavement distress, including alligator cracking, rutting, raveling, patching, and various forms of cracking and deformation [4,8,9].

For that, the pavement network is divided into branches, sections, and sample units, and the minimum number of sample units to be inspected within each section to provide a statistical estimate of the section's PCI (95% confidence) is determined. Each sample unit can be inspected using a manual, equipped vehicle, or unmanned aerial vehicle (UAV) approach to collect distress pavement data (type, severity, and density) [10,11]. A meticulous calculation is followed to produce the PCI value, which ranges from 0 to 100, and can be reclassified into three classes—Good (71–100), Fair (56–70), and Poor (0–55)—or

seven classes—Good (86–100), Satisfactory (71–85), Fair (56–70), Poor (41–55), Very Poor (26–40), Serious (11–25), and Failed (0–10) [8].

The use of the PCI for pavement condition assessment is considered vital, as it enables the development of appropriate and effective maintenance plans. As such, PCI evaluation is an essential component of an APMS, directly contributing to the extension of pavement life.

Table 1 shows the relationship between the PCI, the type of intervention required, the cost implications, and the improvement in pavement performance. Lower PCI values, implying later maintenance, lead to more expensive interventions, with rehabilitation costing up to five times higher than maintenance [6]. This tends to follow Pareto’s 20/80 rule, which states that 80% of problems could be avoided if the first 20% of causes were addressed.

**Table 1.** Cost and performance impact based on the intervention timing (adapted from [2,3,12–14]).

PCI Classes	PCI Values	Type of Intervention	Cost-Impact	Performance Improvement
Good	71–100	Preventive maintenance	Low	Considerable
Fair	56–70	Short-term preventive maintenance or reconstruction	High	High
Poor	0–55	Minimum level of serviceability	Very high	Pavements overdo
Good	86–100	Routine maintenance	Very low	Small
Satisfactory	71–85	Preventive maintenance	Low	Considerable
Fair	56–70	Short-term preventive maintenance or reconstruction (Rehabilitation trigger)	High	High
Poor	41–55	Reconstructive maintenance	Very high	Pavements overdo
Very Poor	26–40	Short-term reconstruction	Very high	Pavements overdo
Serious	11–25	Urgent reconstruction	Extremely high	Pavements overdo
Failed	0–10	Immediate and complete reconstruction	Extremely high	Pavements overdo

Although the PCI method is efficient, its application can be time-consuming due to the extensive calculations and the need to consult ‘Deduct Value’ curves for each selected pavement sample unit [15]. Consequently, there is an urgent need to optimize and automate this process, which can be achieved through the application of machine learning methodologies. This would reduce the overall cost and time needed for PCI calculations and make it accessible to airport pavement maintenance entities of all sizes.

### 1.3. Machine Learning: Concepts, Algorithms, and PCI Prediction

To overcome budget constraints while maintaining or increasing efficiency, the digital 4.0 revolution offers new technologies to enhance processes in pavement engineering [15,16]. In this context, ML, which leverages the potential of digital environments to process vast amounts of data at computational speed, emerges as a key technology that directly enables the creation of models to achieve these goals.

Since an algorithm is defined as a set of well-defined rules describing a computational problem-solving process [17,18], it becomes essential to understand and select the most suitable ML algorithm for the problem being analyzed. The choice of the algorithm is generally based on the type of input and output data being considered, namely, numerical, categorical, or an image. In the specific case of the PCI calculation, the input data are numerical variables representing quantities of distress (density) by distress type and severity level, and the output variable PCI can be considered both numerical and categorical (classes). It should be noted that smaller airports typically rely on traditional inspection methods (conducted on foot), and as a result, the collection of new data and the historical record of pavement surface distress are largely alphanumeric (without image data). Taking this into account, five classic ML algorithms were considered as adequate for building PCI prediction models: Linear Regression (LR), Decision Tree (DT), Random Forest (RF), Artificial Neural Network (ANN), and Support Vector Machine (SVM). These algorithms

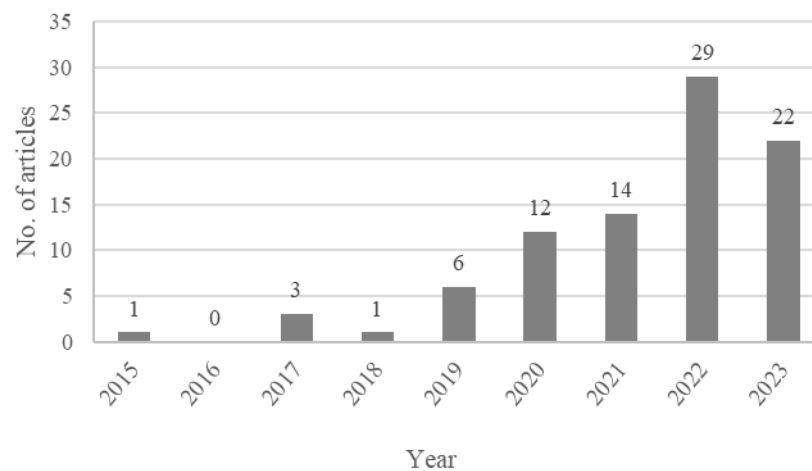
can be classified as opaque (black box) or transparent according to their mode of operation. In opaque algorithms (black box), such as the ANN and SVM, the operating mechanism is not available for examination; therefore, the algorithm and its equations are fully trusted. In transparent algorithms, such as LR, DT, and RF, the operating mechanism is clear and provides data that allow the interpretation of the algorithm's operation [19]. The ML algorithms mentioned use supervised learning techniques that produce a general pattern for the input data to provide the output value [17].

Linear Regression (LR) algorithms are simple algorithms used for numerical variables. It reduces the input values (independent variables) to an output expression that describes a behavior (dependent variable) [20]. Although simple and efficient, research suggests that their results are less efficient than opaque ML algorithms [19,21,22]. On the other hand, Rudin [23] points out that opaque algorithms are not necessary if LR is sufficiently efficient, suggesting that the added complexity and lack of visibility of the mechanism of opaque algorithms raise questions about the output values provided. Decision Trees (DT) are based on a tree of decisions, represented by nodes, edges, and leaves, where each decision, based on the input data, leads to the correct path to produce the output value. Although efficient, it can overfit values if the classes of the dependent variable are unbalanced, leading to the decision being made in the heavier statistically weighted class, which can propagate an error between the different levels of the structure, thus affecting the result [17,22]. Random Forest (RF) is a forest of decision trees where the algorithm decides which trees to consider for a given input dataset. It tends to provide greater visibility of output patterns. However, the added complexity can introduce errors into the decision trees, such as higher error propagation due to a greater number of trees, leading to errors between different decision levels [17,22]. The Artificial Neural Network (ANN) is an opaque algorithm inspired by the brain's structure. The input data run through a series of layers with interconnected nodes, where it is not possible to fully evaluate its mechanism due to the lack of transparency in how decisions are made. Nevertheless, it is usually associated with a successful correlation of diverse data [15,24]. Finally, the Support Vector Machine (SVM) is a complex algorithm based on the distance between vectors from different hyperplanes. The higher the 'margin', the greater the separation between datasets, avoiding overfitting or mixing problems. The high level of complexity makes the algorithm difficult to use. SVM is more commonly used for image classification [17,22].

ML algorithms are sensitive and can be affected by several factors, such as the size and balance of the database, the operating parameters chosen, and the type of data being considered. Several methods are available to minimize some of these effects. Random oversampling (ROS) and random undersampling (RUS) can be used to correct database imbalance, aiming to improve the discriminatory capabilities of the resulting models. According to Chawla et al. [25] and Johnson and Khoshgoftaar [26], applying ROS increases the training time due to the increased size of the training set and has also been shown to cause overfitting. The Synthetic Minority Oversampling Technique (SMOTE), which generates artificial minority samples by interpolating between existing minority samples and their nearest neighbors, is generally used to balance these trade-offs and improve discrimination [25]. Conversely, random undersampling (RUS) works by randomly removing samples from the majority class to achieve a more balanced distribution of data, reducing the overall data sample size.

ML has been applied to engineering for several purposes, such as control and systems engineering, including equipment automation and the digital threats area [27,28]. In pavement engineering, it is usually associated with predictive infrastructure maintenance [29,30] and represents a small portion of the ML use within civil engineering. Although the strand is being developed, ML for PCI prediction is still under development. Figure 1 illustrates

the co-occurrence keywords results obtained from a Scopus database search conducted on 8 June 2024 using the Boolean expression: ('PCI' OR 'pavement condition index') AND 'machine learning'.



**Figure 1.** Co-occurrence analysis of keywords related to machine learning and Pavement Condition Index (PCI).

Existing research on ML applied to pavement management ranges from deep learning applications and algorithm analysis to modeling and prediction. While a few studies have explored ML to calculate the PCI, research specifically predicting airport pavement conditions is scarce.

The single airport study [31] examined only two distress types (crack and utility crack) using a dataset of 32 sample units from a single runway, applying a Convolutional Neural Network (CNN) to image data. In contrast, several studies concerning road pavements [32–34] have successfully applied various ML algorithms to predict the PCI. Using alphanumeric and image data, these efforts (e.g., ANN [33,34], RF [32], and SVM [32]) relied on explanatory variables like surface distress data [32–34], the International Roughness Index (IRI) [32,33], Mean Profile Depth (MPD), and rut depth, resulting in consistently high accuracy rates, ranging from 75% to 99%.

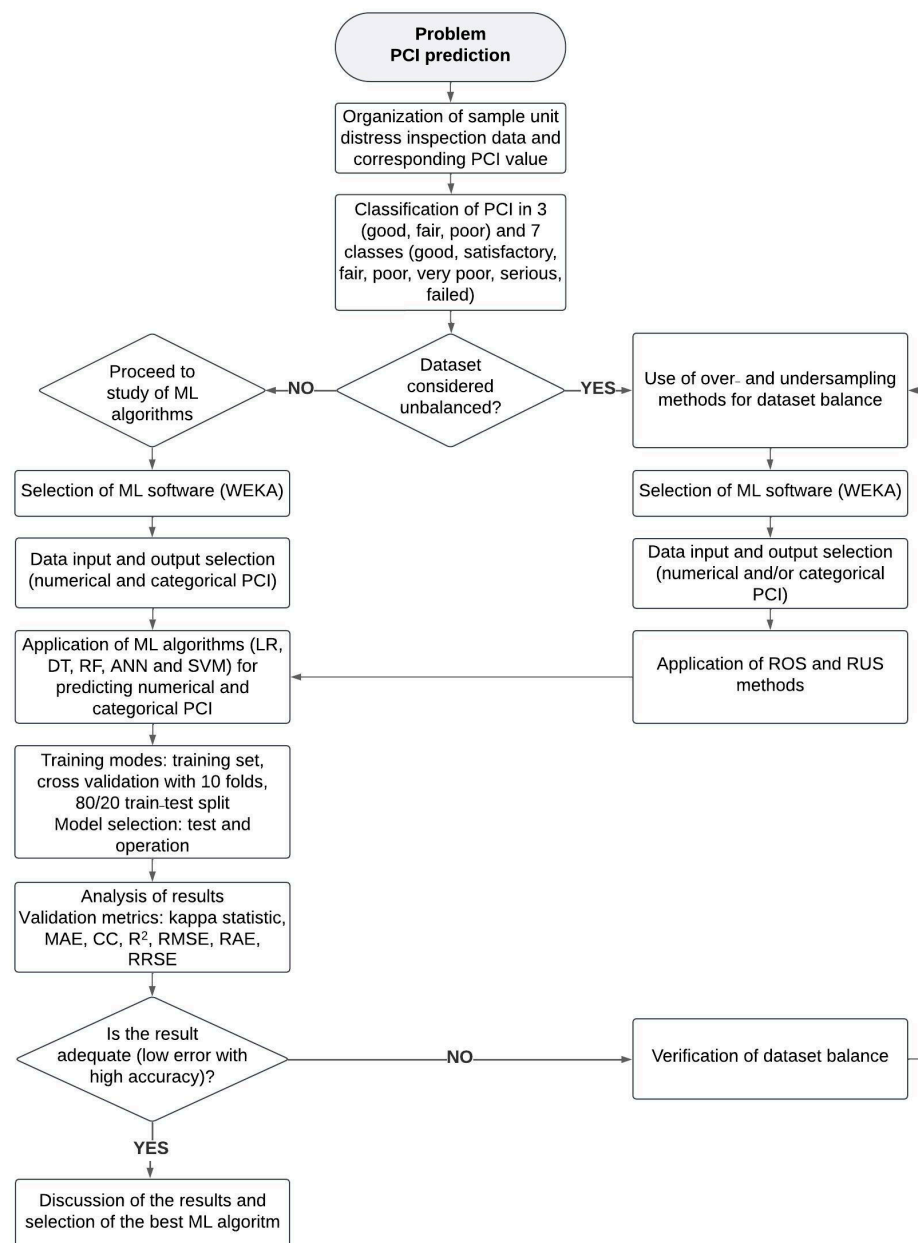
More recent studies on road pavements, such as those by Lin et al. [35] and Shaheen et al. [36], point toward a clear trend: the utilization of boosting algorithms, including Extreme Gradient Boosting (XGBoost), Gradient Boosting Decision Tree (GBDT), and Light Gradient Boosting Machine (LightGBM), alongside an expanded set of explanatory variables. These variables go beyond just pavement distress information or the IRI to include pavement age and external factors potentially influencing the PCI, such as traffic load, climate, and material properties. Critically, no studies were identified that comprehensively employed all pavement distress types and severity levels to predict PCI classes in road or airport environments. This lack of comprehensive research on PCI calculations based on ML algorithms, particularly their application to airport pavements, represents a considerable gap with high research potential.

## 2. Materials and Methods

This section presents the main steps of the proposed methodology for building sample unit PCI prediction models using the alphanumeric data and supervised ML algorithms that are available in WEKA 3.8.6 software [37]. Three types of PCI outcomes (the dependent variable) were considered for the analysis: the numerical PCI (0–100), the categorical PCI, divided into 3 classes (good, fair, and poor), and the categorical PCI, divided into 7 classes (good, satisfactory, fair, poor, very poor, serious, and failed).

The methodology aims to process data from pavement distress inspections carried out on airport runways to obtain the PCI. The independent (explanatory) variables considered were the density of the pavement surface affected by the 17 distresses considered in the ASTM D5340-23 standard [8], by level of severity (low, medium, and high), as a percentage of the sample unit area, totaling 51 (17 × 3) independent variables.

Figure 2 presents the flowchart of the decisions and processes involved in the analysis. First, the distress data were organized alongside their corresponding PCI value and categorical classification. Next, the database’s balance was checked to analyze any imbalance issues potentially impacting model training. Assuming a balanced database, the LR, DT, RF, SVM, and ANN algorithms were applied without any further data adjustment. Each model was evaluated under three distinct training schemes: a simple training set, 10-fold cross-validation (CV10), and an 80/20 train–test split. If unbalanced, the ROS (SMOTE) and RUS methods are considered to expand or reduce the database for over- or under-weighted classes.



**Figure 2.** Flowchart of the methodology for analyzing pavement surface distress data using ML to generate PCI sample unit prediction models [40].

Yang et al. [38] demonstrate the success of ROS in improving the results for small databases, while Hayaty et al. [39] highlight the potential of ROS to improve overall accuracy. RUS, on the other hand, limits the original database for balancing purposes, which can have a negative impact on already small databases.

The results were then evaluated by statistical analysis based on the correlation coefficient (CC), Kappa statistic, coefficient of determination ( $R^2$ ), mean absolute error (MAE), root mean square error (RMSE), relative absolute error (RAE), and root relative square error (RRSE) to assess overall efficiency and accuracy of the models. ROC area and confusion matrix were also considered for categoric data performance evaluation. Table 2 shows the criteria used to interpret the Kappa statistic correlation values, correlation coefficient, and ROC area.

While a simple comparison of the evaluation metrics provides an initial indication of the best-performing model, statistical confirmation is essential to validate its superiority. This comparative assessment of machine learning models was conducted using the WEKA Experimenter environment. Each model was rigorously evaluated through repeated k-fold cross-validation to obtain reliable performance estimates. For regression problems, the CC was used as the main evaluation metric, while the Kappa statistic was considered for classification problems. The Analyze module of WEKA was then utilized to compute the overall rankings and perform statistical significance testing. In particular, the corrected paired t-test (with the Nadeau and Bengio correction) was applied to account for dependencies arising from resampling, ensuring a robust comparison of model performance.

**Table 2.** Criteria for interpreting correlation coefficient, Kappa statistic, and ROC values (adapted from [41–43]).

<b>Correlation Coefficient</b>		
Range	Correlation	
±0.00–0.10	Negligible	
±0.10–0.39	Weak	
±0.40–0.69	Moderate	
±0.70–0.89	Strong	
±0.90–1.00	Very strong	
<b>Kappa statistic</b>		
Range	Correlation	
<0	Worse than expected	
0–0.20	None	
0.21–0.39	Minimal	
0.40–0.59	Weak	
0.60–0.79	Moderate	
0.80–0.90	Strong	
>0.90	Almost perfect	
<b>ROC area</b>		
Range	Model discriminative ability	
0.50	No discriminative ability	
0.51–0.70	Discrimination is weak	
0.71–0.80	Discrimination is acceptable	
0.81–0.90	Discrimination is good	
>0.90	Discrimination is exceptional	

Finally, the relevance of each independent variable for predicting the PCI for the best-performing model was determined using distinct statistical measures tailored to the output’s nature. For numerical output variables, ReliefFAtributeEval (in WEKA) was used to determine the attribute’s relevance by estimating its local discriminative power (the

ability to distinguish between nearest neighbors of different classes). For categorical output variables, attributes were evaluated via Information Gain, using InfoGainAttributeEval in WEKA.

### 3. Case Study

#### 3.1. Data Description

The database considered consists of distress and derived PCI information from 261 flexible pavement sample units from three similar international airport runways (two in Cape Verde and one in Peru) [14,44]. Density data related to the 17 surface pavement distress types, each with three severity levels (low, medium, and high), reported in ASTM D5340-12 [45], were considered to calculate the PCI of these sample units. Table 3 details the distresses present in the database, providing a count of entries for each distress type and severity level. The subsequent Figures 3 and 4 illustrate the distribution of the PCI when categorized into three and seven classes, respectively.



Figure 3. PCI distribution by 3 classes: Poor (0–55), Fair (56–70), and Good (71–100).

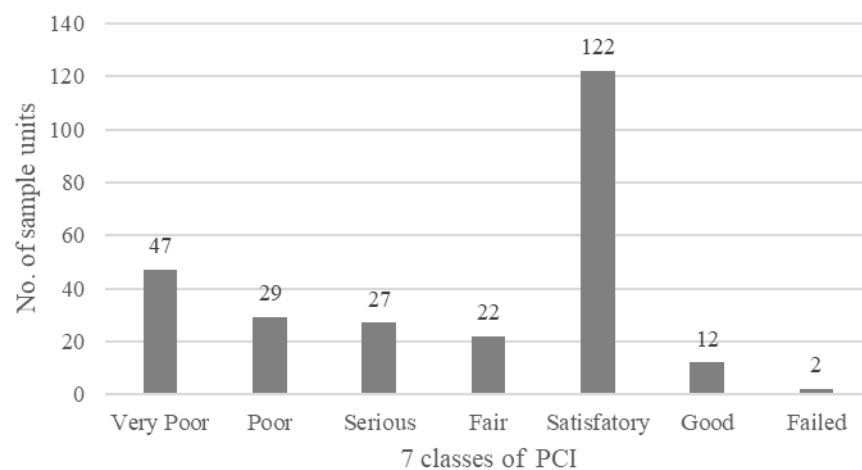


Figure 4. PCI distribution into 7 classes: Failed (0–10), Serious (11–25), Very Poor (26–40), Poor (41–55), Fair (56–70), Satisfactory (71–85), and Good (86–100).

**Table 3.** Distribution of distress entries by type and severity level in the database.

Distress and Severity Level	No. of Entries	Rank
Alligator Cracking (Low)	7	20
Alligator Cracking (Medium)	17	13
Alligator Cracking (High)	1	25
Depression (Low)	3	22
Depression (Medium)	8	17
Depression (High)	13	14
Jet-Blast Erosion (Low)	2	23
Join Reflection Cracking (Low)	117	2
Join Reflection Cracking (Medium)	104	3
Join Reflection Cracking (High)	1	25
Longitudinal and Transverse Cracking (Low)	138	1
Longitudinal and Transverse Cracking (Medium)	92	4
Longitudinal and Transverse Cracking (High)	8	17
Oil Spillage (Low)	1	25
Patching and Utility Cut Patching (Low)	12	15
Patching and Utility Cut Patching (Medium)	31	10
Patching and Utility Cut Patching (High)	81	5
Polished Aggregate (Low)	1	25
Raveling (Low)	19	12
Raveling (Medium)	46	9
Raveling (High)	59	7
Rutting (Low)	2	23
Rutting (Medium)	7	20
Swell (Low)	1	25
Swell (Medium)	8	17
Swell (High)	9	16
Weathering (Low)	20	11
Weathering (Medium)	54	8
Weathering (High)	71	6

Note: Shading identifies the top 15 distresses with the most entries in the database.

The analysis revealed that the most frequent distresses in the database are as follows: Longitudinal and Transverse Cracking (Low and Medium), Joint Reflection Cracking (Low and Medium), Patching and Utility Cut Patching (Low, Medium, and High), Weathering (Low, Medium, and High), Raveling (Low, Medium, and High), Alligator Cracking (Medium), and Depression (High). These highly represented distresses are common across the existing literature on airport pavements.

However, as can be observed in Figures 1 and 2, the database exhibits a significant imbalance, specifically regarding the PCI classes. The least represented classes are the 'Fair' class for the three-class PCI and the 'Good', 'Fair', and 'Failed' classes for the more granular seven-class PCI.

To address this, the database must be expanded in the future to increase the total number of entries and ensure the maximum number of distresses listed in the ASTM 5340 standard [8]. Furthermore, augmenting the database with more cases will critically improve the representativeness of the PCI classes, especially for accurate seven-class classification. Despite the observed database imbalance, the LR, SVM, RF, DT, and ANN algorithms were initially applied to the original data. This step was taken to build baseline PCI prediction models and to gain insight into the capacity of each model to address this imbalance across the three tested output schemes (the numerical, three-, and seven-class PCI).

### 3.2. Algorithms

Table 4 presents the ML testing procedure adopted. The results for the chosen ML algorithms are compared to assess the ability of the ML algorithms to correctly predict the PCI based on the three following training options: training set, cross-validation with 10-folds (CV10), and 80/20 train-test split, totaling 39 models. The hyperparameters adopted in each model were set to the default values provided by the WEKA 3.8.6 software, as shown in Tables 5 and 6.

**Table 4.** ML testing procedure.

ML Algorithm	Training Options	PCI Output	Algorithm (WEKA)
Linear Regression (LR)	Training set CV10 80/20 split	Numerical	Linear regression
Decision Tree (DT)	Training set CV10 80/20 split	Numerical Categoric	M5P [46] J48 [47]
Random Forest (RF)	Training set CV10 80/20 split	Numerical Categoric	Random Forest [48]
Artificial Neural Network (ANN)	Training set CV10 80/20 split	Numerical Categoric	Multilayer Perception (MLP) [37]
Support Vector Machine (SVM)	Training set CV10 80/20 split	Numerical Categoric	SMOreg [49,50] SMO [49,50]

**Table 5.** Hyperparameter configuration for the five ML models predicting numerical PCI (0–100).

WEKA Hyperparameters	LR	DT (M5P)	RF	ANN (MLP)	SVM (SMOreg)
attributeSelectionMethod	M5	-	-	-	-
batchSize	100	100	-	100	100
debug	False	False	False	False	False
doNotCheckCapabilities	False	False	False	False	False
eliminateColinearAttributes	True	-	-	-	-
minimal	False	-	-	-	-
numDecimalPlaces	4	4	2	2	2
outputAdditionalStats	False	-	-	-	-
ridge	$1.0 \times 10^{-8}$	-	-	-	-
useQRDecomposition	False	-	-	-	-
buildRegressionTree	-	False	-	-	-
minNumInstances	-	4.0	-	-	-
saveInstances	-	False	-	-	-
unpruned	-	False	-	-	-
useUnsmoothed	-	False	-	-	-
bagSizePercent	-	-	100	-	-
breakTiesRandomly	-	-	False	-	-
calcOutOfBag	-	-	False	-	-
computeAttributeImportance	-	-	False	-	-
maxDeep	-	-	0	-	-
numExecutionSlots	-	-	1	-	-
numFeatures	-	-	0	-	-
outputOutOfBagComplexity Statistics	-	-	0	-	-
printClassifiers	-	-	False	-	-
seed	-	-	1	0	-
storeOutOfBagPredictions	-	-	False	-	-
GUI*	-	-	-	False	-
autoBuild	-	-	-	True	-
decay	-	-	-	False	-
hiddenLayers	-	-	-	a**	-
learningRate	-	-	-	0.3	-
momentun	-	-	-	0.2	-
nominalToBinaryFilter	-	-	-	True	-

Table 5. Cont.

WEKA Hyperparameters	LR	DT (M5P)	RF	ANN (MLP)	SVM (SMOreg)
normalizeAttributes	-	-	-	True	-
normalizeNumericClass	-	-	-	True	-
reset	-	-	-	True	-
resume	-	-	-	False	-
trainingTime	-	-	-	500 epochs	-
validationSetSize	-	-	-	0	-
validationThreshold	-	-	-	20	-
c***	-	-	-	-	1.0
filterType	-	-	-	-	Normalize training data
kernel	-	-	-	-	PolyKernel -E 1.0 -C 250007
regOptimizer	-	-	-	-	RegSMOImproved -T 0.001 -V -P $1.0 \times 10^{-12}$ -L 0.001

\* Graphical User Interface. \*\* The parameter 'a' (automatic) was used, which triggers WEKA's default heuristic to calculate the number of hidden neurons. This resulted in a single hidden layer with 26 nodes. \*\*\* The parameter 'c' is the regularization or cost parameter.

Table 6. Hyperparameter configuration for the four ML models predicting 3- and 7-class PCI.

WEKA Hyperparameters	DT (J48)	RF	ANN (MLP)	SVM (SMO)
batchSize	100	100	100	100
binarySplits	False	-	-	-
collapseTree	True	-	-	-
confidenceFactor	0.25	-	-	-
debug	False	False	False	-
doNotCheckCapabilities	False	False	False	-
doNotMakeSplitPointActualValue	False	-	-	-
minNumObj	2	-	-	-
numDecimalPlaces	2	2	2	2
numFolds	3	-	-	-
reducedErrorPruning	False	-	-	-
saveInstanceData	False	-	-	-
seed	1	1	0	-
subtreeRaising	True	-	-	-
unpruned	False	-	-	-
useLaplace	False	-	-	-
useMDLCorrection	True	-	-	-
bagSizePercent	-	100	-	-
breakTiesRandomly	-	False	-	-
calcOutOfBag	-	False	-	-
computeAttributeImportance	-	False	-	-
maxDepth	-	0	-	-
numExecutionSlots	-	1	-	-
numIterations	-	100	-	-
outputOutOfBagComplexityStatistics	-	False	-	-
printClassifiers	-	False	-	-
storeOutOfBagPredictions	-	False	-	-
GUI*	-	-	False	-
autoBuild	-	-	True	-
decay	-	-	False	-
hiddenLayers	-	-	a**	-
learningRate	-	-	0.3	-
momentum	-	-	0.2	-
nominalToBinaryFilter	-	-	True	-
normalizeAttributes	-	-	True	-
normalizeNumericClass	-	-	True	-
reset	-	-	True	-
resume	-	-	False	-
trainingTime	-	-	500 epochs	-
validationSetSize	-	-	0	-
validationThreshold	-	-	20	-
buildCalibrationModels	-	-	-	False
c***	-	-	-	1.0
calibrator	-	-	-	Logistic -R $1.0 \times 10^{-8}$
checksTurnedOff	-	-	-	-M -1 -num-decimal-places 4
epsilon	-	-	-	False
filterType	-	-	-	$1.0 \times 10^{-12}$
kernel	-	-	-	Normalize training data
numFolds	-	-	-	PolyKernel -E 1.0 -C 250007
randomSeed	-	-	-	-1
toleranceParameter	-	-	-	1
				0.001

\* Graphical User Interface. \*\* The parameter 'a' (automatic) was used, which triggers WEKA's default heuristic to calculate the number of hidden neurons. This resulted in a single hidden layer with 26 nodes. \*\*\* The parameter 'c' is the regularization or cost parameter.

3.3. Results and Discussion

3.3.1. Numerical PCI

Table 7 summarizes the best results obtained by each of the five algorithms considered for the numerical PCI prediction. Subsequently, Table 8 provides a comparative statistical assessment of these models, including the overall rankings and statistical significance testing. Finally, Table 9 presents the Top 15 Relief Attribute Evaluation ranking for the best-performing model (RF).

**Table 7.** Validation metrics of the best-performing models for each algorithm considering numerical PCI (0–100).

Validation Metric	Algorithms				
	LR	DT	RF	ANN	SVM
Training	80/20 split	CV10	CV10	80/20 split	80/20 split
CC	0.88	0.91	0.93	0.88	0.88
R <sup>2</sup>	0.77	0.82	0.86	0.77	0.77
MAE	8.41	6.22	5.81	7.43	8.26
RMSE	11.48	9.22	8.73	11.39	11.38
RAE (%)	39.83	29.97	27.97	35.18	39.13
RRSE (%)	49.74	40.16	38.02	49.35	49.31
aPCI	58.79	58.58	58.58	58.77	58.77
pPCI	59.68	58.83	60.10	55.87	59.50
Error (aPCI-pPCI)	−1.00	−0.24	−1.51	2.90	−0.72

Note: aPCI means actual calculated PCI, and pPCI means predicted PCI. Shading identifies the best-performing models.

**Table 8.** Ranking and pairwise statistical comparison of regression models for numerical PCI (correlation coefficient,  $p < 0.05$ ).

Algorithm Ranking	Ranking Analysis		Comparison Analysis	
	> (Wins)	< (Losses)	CC (Mean ± sd)	Comparison vs. RF (Victories/Ties/Losses)
RF	2	0	0.93 ± 0.03	-
DT	0	0	0.90 ± 0.05	(0, 1, 0) slightly worse, significant difference
LR	0	0	0.80 ± 0.19	(0, 1, 0) worse
ANN	0	1	0.85 ± 0.08	(0, 0, 1) worse
SVM	0	1	0.80 ± 0.18	(0, 0, 1) worse

Note: Shading identifies the top-ranked model.

Considering Table 7, for the best-performing models obtained for each tested algorithm, the 80/20 train-test split and the 10-fold cross-validation approaches were considered appropriate and reliable training options. The RF model achieved the best overall performance, with the highest correlation coefficient (0.93) and coefficient of determination (0.86), as well as the lowest prediction errors (MAE = 5.81 and RMSE = 8.73). The DT also performed well, slightly below RF, showing comparable accuracy but higher error values. LR, ANN, and SVM presented similar and less accurate results, with lower correlation

values (0.88) and higher residual errors. The predicted PCI (pPCI) values obtained with RF were the closest to the actual PCI (aPCI), confirming its superior predictive reliability. The use of the ‘training set’ option was discarded, as it determines the same PCI values used to train the algorithm, leading to overfitting and no observable difference between aPCI and pPCI.

To confirm this result, an initial statistical comparison of the regression models was performed using the corrected paired *t*-test based on the correlation coefficient as the performance metric (Table 8). This preliminary ranking analysis confirmed the RF model as the best-performing algorithm, showing statistically significant improvements over two of the competing approaches, while the remaining models exhibited no significant difference among themselves. Consequently, the RF model was selected as the baseline for the subsequent detailed statistical evaluation. In this second analysis, conducted again using the corrected paired *t*-test at the 0.05 confidence level, the RF achieved the highest correlation coefficient ( $0.93 \pm 0.03$ ), outperforming all other algorithms with statistically significant differences. The DT model obtained a slightly lower correlation ( $0.90 \pm 0.05$ ), whereas LR, SVM, and ANN displayed lower correlations (0.88–0.85). Overall, the results confirm the superior predictive accuracy and robustness of the RF approach for the database under study.

**Table 9.** Top 15 Relief Attribute Evaluation ranking results for the RF model applied to the numerical PCI (0–100) original database with 10-fold cross-validation (CV10).

Rank	Attribute (Independent Variable and Severity Level)	Score
1	Raveling (Medium)	0.0756
2	Depression (High)	0.0513
3	Weathering (Medium)	0.0464
4	Rutting (High)	0.0409
5	Swell (High)	0.0398
6	Swell (Medium)	0.0333
7	Alligator Cracking (Medium)	0.0270
8	Raveling (High)	0.0262
9	Alligator Cracking (Low)	0.0236
10	Longitudinal and Transverse Cracking (High)	0.0226
11	Rutting (Medium)	0.0225
12	Weathering (Low)	0.0181
13	Patching and Utility Cut Patching (High)	0.0170
14	Raveling (Low)	0.0140
15	Depression (Medium)	0.0139

Finally, an analysis of the independent variables’ importance in a numerical PCI prediction for the RF model is detailed in Table 9. These findings indicate that three of the five most influential variables, namely Raveling (Medium), Depression (High), and Weathering (Medium), are also among the top 15 variables with the most database entries. However, the other two influential variables, Swell (High) and Rutting (High), have no database records. This lack of data for key variables represents a significant constraint on the model’s reliability, underscoring the necessity of including entries for them to properly evaluate their true impact on the PCI.

### 3.3.2. Three- and Seven-Class PCI

Tables 10–16 summarize the best results obtained by each of the four algorithms considered for the three- and seven-class PCI predictions. They also provide a comparative statistical assessment of these models, including the overall rankings and statistical significance testing, and the Top 15 Information Gain ranking for the best-performing model (RF) for the three-class PCI.

**Table 10.** Validation metrics of the best-performing models for each algorithm considering 3-class PCI (good, fair, and poor).

Validation Metric	Algorithms				
	LR	DT	RF	ANN	SVM
Training	-	CV10	CV10	CV10	CV10
Kappa statistic	-	0.80	0.88	0.82	0.80
MAE	-	0.09	0.10	0.07	0.26
RMSE	-	0.24	0.21	0.23	0.33
RAE (%)	-	24.65	25.58	19.56	67.58
RRSE (%)	-	57.11	48.15	54.00	75.60
Error (%) (Incorrectly classified)	-	10.72	6.51	9.96	11.11

Note: Shading identifies the best-performing model.

**Table 11.** Detailed accuracy for 3-class PCI using RF with 10-fold cross-validation.

Confusion Matrix			Precision	ROC Area	Class
a	b	c			
99	4	2	0.93	0.97	a
6	12	4	0.75	0.81	b
1	0	133	0.95	0.992	c
-	-	-	0.93	0.970	Avg.

Note: a = poor; b = fair; c = good; Avg. = average.

**Table 12.** Ranking and pairwise statistical comparison of regression models for 3-class PCI (Kappa statistic,  $p < 0.05$ ).

Algorithm Ranking	Ranking Analysis		Comparison Analysis	
	> (Wins)	< (Losses)	Kappa Statistic (Mean ± sd)	Comparison vs. RF (Victories/Ties/Losses)
RF	2	0	0.88 ± 0.09	-
ANN	0	0	0.82 ± 0.11	(0, 1, 0) slightly worse, significant difference
DT	0	1	0.83 ± 0.09	(0, 0, 1) worse
SVM	0	1	0.80 ± 0.11	(0, 0, 1) worse

Note: Shading identifies the top-ranked model.

**Table 13.** Top 15 Information Gain Attribute Evaluation ranking results for the RF model applied to the 3-class PCI original database with 10-fold cross-validation (CV10).

Rank	Attribute (Independent Variable and Severity Level)	Score
1	Join Reflection Cracking (Low)	0.6776
2	Longitudinal and Transverse Cracking (Low)	0.6032
3	Join Reflection Cracking (Medium)	0.5469
4	Patching and Utility Cut Patching (High)	0.4993
5	Weathering (High)	0.3800
6	Longitudinal and Transverse Cracking (Medium)	0.3542

Table 13. Cont.

Rank	Attribute (Independent Variable and Severity Level)	Score
7	Raveling (High)	0.3216
8	Patching and Utility Cut Patching (Medium)	0.1561
9	Weathering (Medium)	0.1492
10	Raveling (Medium)	0.1235
11	Alligator Cracking (Medium)	0.0905
12	Weathering (Low)	0.0520
13	Depression (High)	0.0520
14	Patching and Utility Cut Patching (Low)	0.0466
15	Swell (High)	0.0466

Table 14. Validation metrics of the best-performing models for each algorithm considering 7-class PCI (good, satisfactory, fair, poor, very poor, serious, and failed).

Validation Metric	Algorithms				
	LR	DT	RF	ANN	SVM
Training	-	CV10	CV10	CV10	CV10
Kappa statistic	-	0.56	0.58	0.54	0.51
MAE	-	0.09	0.10	0.10	0.21
RMSE	-	0.2642	0.23	0.26	0.32
RAE (%)	-	46.78	51.34	49.09	103.70
RRSE (%)	-	82.53	72.83	80.52	98.85
Error (%) (Incorrectly classified)	-	30.65	29.11	32.56	33.33

Note: Shading identifies the best-performing model.

Table 15. Detailed accuracy for 7-class PCI using RF with 10-fold cross-validation (CV10).

Confusion Matrix							Prec.	ROC Area	Class
a	b	c	d	e	f	g			
33	5	3	5	1	0	0	0.51	0.89	a
16	9	1	2	1	0	0	0.45	0.86	b
12	4	7	3	0	0	1	0.54	0.87	c
3	1	1	12	5	0	0	0.52	0.83	d
0	1	0	1	117	3	0	0.91	0.99	e
0	0	0	0	5	7	0	0.70	0.98	f
1	0	1	0	0	0	0	0.00	0.99	g
-	-	-	-	-	-	-	0.68	0.93	Avg.

Note: a = very poor; b = poor; c = serious; d = fair; e = satisfactory; f = good; g = failed; Avg. = average; Prec. = precision.

Considering Table 10, for the best-performing models obtained for each tested algorithm, the 10-fold cross-validation approach was identified as a reliable and consistent training option. Among the tested algorithms, the RF model achieved the best overall performance, presenting the highest Kappa statistic (0.88) and the lowest error rate (6.51%), confirming its superior classification capability. The DT and ANN models also yielded competitive results, with Kappa values of 0.80 and 0.82, respectively, but slightly higher misclassification rates. Conversely, the SVM displayed weaker performance, with a Kappa value of 0.80 and the highest error rate (11.11%).

**Table 16.** Ranking and pairwise statistical comparison of regression models for 7-class PCI (Kappa statistic,  $p < 0.05$ ).

Algorithm Ranking	Ranking Analysis		Comparison Analysis	
	> (Wins)	< (Losses)	Kappa Statistic (Mean ± sd)	Comparison vs. RF (Victories/Ties/Losses)
RF	1	0	0.59 ± 0.10	-
DT	0	0	0.53 ± 0.11	(0, 1, 0) slightly worse, significant difference
ANN	0	0	0.54 ± 0.10	(0, 1, 0) slightly worse, significant difference
SVM	0	1	0.52 ± 0.08	(0, 0, 1) worse

Note: Shading identifies the top-ranked model.

The confusion matrix presented in Table 11 further supports the robustness of the RF model. The RF achieved high precision across all three PCI classes (0.93, 0.75, and 0.95 for good, fair, and poor, respectively), with an overall average precision of 0.93 and a ROC area of 0.97, indicating strong discrimination ability between classes and excellent generalization performance.

To confirm these findings, a statistical comparison of the models was also performed using the corrected paired *t*-test based on the Kappa statistic, as shown in Table 12. This ranking analysis reaffirmed the RF as the top-performing algorithm, achieving the highest mean Kappa statistic (0.88 ± 0.09). Compared with RF (first-ranked model), the ANN model presented a slightly lower correlation (0.82 ± 0.11) with a statistically significant difference, while DT (0.83 ± 0.09) and SVM (0.80 ± 0.11) also performed worse, though without significant improvement over one of the other models. Overall, the results confirm the superior predictive accuracy, consistency, and robustness of the Random Forest model for the three-class PCI classification.

Table 13 presents the importance analysis of the independent variables used in the RF model for three-class PCI prediction. The analysis reveals that all five of the most influential variables, namely Joint Reflection Cracking (Low, Medium), Longitudinal and Transverse Cracking (Low), Patching and Utility Cut Patching (High), and Weathering (High), are among the top 15 variables with the highest number of database entries.

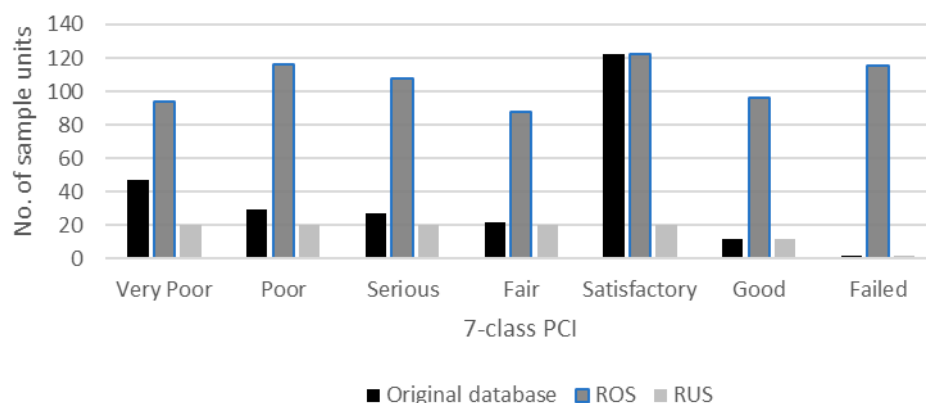
For the seven-class PCI prediction, the results proved to be inefficient and lacked the robustness required for reliable replicability, particularly for the classes ‘poor’, ‘serious’, and ‘failed’. This weak performance can be attributed to the small size and highly unbalanced distribution of the database across the seven PCI classes, which hindered effective model calibration. To address this limitation, the SMOTE and RUS techniques were applied to balance the database.

The application of the SMOTE technique generated additional synthetic cases for the underrepresented classes, increasing the total number of cases from 261 to 739. In contrast, RUS reduced the number of cases in the overrepresented classes, resulting in a smaller database of 114 cases. Because SMOTE appends synthetic cases at the end of the database, the Randomize function was subsequently applied to ensure an unbiased and randomized distribution of instances.

Table 17 and Figure 5 present a comparison between the original PCI class distribution and the balanced distributions obtained after applying SMOTE and RUS.

**Table 17.** Number of cases by PCI class for the original and treated databases (over and under sampling).

PCI Classes	No. of Sample Units		
	Original Database	Database After ROS (SMOTE)	Database After RUS
Very Poor	47	94	20
Poor	29	116	20
Serious	27	108	20
Fair	22	88	20
Satisfactory	122	122	20
Good	12	96	12
Failed	2	115	2
Total cases	261	739	114



**Figure 5.** Comparison between the distribution of the original and treated databases (over and under sampling).

A clear impact can be seen on the distribution of cases in the database by PCI class, with ROS (SMOTE) being the most efficient, as it was able to correct underweighted classes by creating new cases based on the original 261 sample units. On the other hand, although balanced, the reduction in cases by the RUS method can significantly impact the model results, as not enough cases are considered for consistent seven-class PCI modeling. Tables 18–21 present a comprehensive comparison of the best-performing models derived from the original and treated databases. These tables include validation metrics, detailed accuracy results for the best algorithm, the overall model rankings, and statistical significance testing results.

**Table 18.** DT and RF model results for 7-class PCI considering original, RUS, and ROS (SMOTE) databases, and 10-fold cross-validation (CV10).

Parameters	DT			RF		
	Original	RUS	ROS (SMOTE)	Original	RUS	ROS (SMOTE)
Training	CV10	CV10	CV10	CV10	CV10	CV10
Kappa statistic	0.56	0.46	0.80	0.58	0.55	0.91
MAE	0.09	0.13	0.05	0.10	0.15	0.05
RMSE	0.26	0.32	0.22	0.23	0.27	0.14
RAE (%)	46.78	56.86	19.61	51.34	62.23	22.55
RRSE (%)	82.53	94.03	62.63	72.83	78.60	39.64
Error (%) (Incorrectly Classified)	30.65	44.73	16.78	29.11	36.84	7.17

Note: Shading identifies the best-performing model.

**Table 19.** ANN and SVM model results for 7-class PCI considering original, RUS, and ROS (SMOTE) databases, and 10-fold cross-validation (10CV).

Parameters	ANN			SVM		
	Original	RUS	ROS (SMOTE)	Original	RUS	ROS (SMOTE)
Training	CV10	CV10	CV10	CV10	CV10	CV10
Kappa statistic	0.54	0.48	0.83	0.51	0.39	0.71
MAE	0.10	0.14	0.05	0.21	0.22	0.21
RMSE	0.26	0.32	0.17	0.32	0.32	0.31
RAE (%)	49.09	57.16	20.72	103.70	90.79	85.61
RRSE (%)	80.52	92.68	50.29	98.85	93.13	88.52
Error (%) (Incorrectly Classified)	32.56	42.98	14.88	33.33	50.00	24.35

**Table 20.** Detailed accuracy for ROS (SMOTE) 7-class PCI using RF with 10-fold cross-validation (CV10).

Confusion Matrix							Prec.	ROC Area	Class
a	b	c	d	e	f	g			
78	8	3	4	1	0	0	0.84	0.98	a
9	103	0	3	1	0	0	0.88	0.99	b
4	3	100	1	0	0	0	0.95	0.99	c
2	2	1	80	3	0	0	0.90	0.99	d
0	1	0	1	115	5	0	0.96	0.99	e
0	0	0	0	0	96	0	0.95	1.00	f
0	0	1	0	0	0	104	1.00	1.00	g
-	-	-	-	-	-	-	0.93	0.99	Avg.

Note: a = very poor; b = poor; c = serious; d = fair; e = satisfactory; f = good; g = failed; Avg. = average; Prec. = Precision.

**Table 21.** Ranking and pairwise statistical comparison of regression models for ROS (SMOTE) 7-class PCI (Kappa statistic,  $p < 0.05$ ).

Algorithm Ranking	Ranking Analysis		Comparison Analysis	
	> (Wins)	< (Losses)	Kappa Statistic (Mean ± sd)	Comparison vs. RF (Victories/Ties/Losses)
RF	3	0	0.92 ± 0.03	-
DT	1	1	0.84 ± 0.05	(0, 0, 1) worse
ANN	1	1	0.83 ± 0.05	(0, 0, 1) worse
SVM	0	3	0.70 ± 0.05	(0, 0, 1) worse

Note: Shading identifies the top-ranked model.

Table 22 summarizes the Top 15 Information Gain attribute ranking for the best-performing seven-class PCI model, which utilizes the RF and ROS (SMOTE) techniques.

Considering Tables 18 and 19, the 10-fold cross-validation approach was adopted as a reliable training option for the seven-class PCI prediction models across all four algorithms and three databases (Original, RUS, and ROS).

**Table 22.** Top 15 Information Gain Attribute evaluation ranking results for the RF model applied to the 7-class PCI database with ROS (SMOTE) oversampling and 10-fold cross-validation (CV10).

Rank	Attribute (Independent Variable and Severity Level)	Score
1	Raveling (High)	0.9552
2	Patching and Utility Cut Patching (High)	0.7793
3	Alligator Cracking (Medium)	0.7650
4	Rutting (High)	0.6651
5	Join Reflection Cracking (Low)	0.6624
6	Raveling (Medium)	0.6523
7	Swell (High)	0.6038
8	Jet-Blast Erosion (Low)	0.5986
9	Swell (Medium)	0.5969
10	Join Reflection Cracking (Medium)	0.5708
11	Weathering (High)	0.5112
12	Longitudinal and Transverse Cracking (Medium)	0.4897
13	Longitudinal and Transverse Cracking (Low)	0.4814
14	Weathering (Medium)	0.3445
15	Patching and Utility Cut Patching (Medium)	0.3333

Among the tested configurations, the RF model trained with ROS (SMOTE) achieved the best overall performance. It presented the highest Kappa statistic (0.91) and the lowest error rate (7.17%), clearly confirming its superior classification capability. The other algorithms' best-performing configurations were competitive but inferior. The superior performance of ROS (SMOTE) across all models suggests that data balancing was crucial for accurate seven-class prediction.

The confusion matrix presented in Table 20 further supports the robustness of the best-performing model: RF with ROS (SMOTE). This model achieved high precision across all seven PCI classes, with a minimum of 0.84 for class 'a' (very poor). With an overall average precision of 0.93 and a ROC area of 0.99, the RF model demonstrates excellent generalization and strong discrimination ability across all seven classes.

Table 21 presents the statistical comparison of the four algorithms using the Kappa statistic. This ranking analysis reaffirms the RF model as the top performer, achieving the highest mean Kappa statistic ( $0.92 \pm 0.03$ ). The RF model won all its statistical comparisons against the other three algorithms (three wins, zero losses). Conversely, the DT ( $0.84 \pm 0.05$ ), ANN ( $0.83 \pm 0.05$ ), and SVM ( $0.70 \pm 0.05$ ) models all performed worse than the RF model.

Finally, Table 22 summarizes the Top 15 Information Gain attribute ranking for the best-performing seven-class PCI model (RF with ROS (SMOTE)). The analysis highlights the most crucial factors in differentiating the seven pavement classes. The top five most influential variables are Raveling (High), Patching and Utility Cut Patching (High), Alligator Cracking (Medium), Rutting (High), and Join Reflection Cracking (Low). Similarly, in the numerical PCI analysis, the Rutting (High) distress was identified as important despite having no database entries. This result requires a more thorough evaluation of the variable's true influence.

#### 4. Conclusions

Airport pavement management is based on standard procedures and practices. Although efficient, they can be time-consuming and costly. Newer technologies, namely machine learning, have emerged from the 4.0 industry revolution and could significantly contribute to the digitalization of civil engineering transport infrastructure processes, thereby reducing overall costs and increasing operational efficiency.

PCI prediction by ML was found to be accurate and showed a good degree of replicability. Based on the analyses performed, the following order was considered from worst- to best-performing algorithms: Linear Regression, Support Vector Machine, Artificial Neural Network, Decision Tree, and Random Forest. Considering the 39 models built and the models obtained by applying the ROS (SMOTE) and RUS methods to the original database, the following points can be considered:

- The 'training set' option on the chosen software is not recommended as it reveals overfitting tendencies due to the output calculation being based on the same trained database.
- Random Forest with 10-fold cross-validation was the most efficient and reliable algorithm for both numeric and categorical PCI, with Kappa statistic values between 0.88 and 0.93, error <7.17%, and a ROC area >0.93. Statistical comparison and ranking analysis consistently identified the RF model as the best-performing algorithm. This was evidenced by the highest mean correlation coefficient (0.93) for the numerical PCI, and superior Kappa statistics for both the three-class (0.88) and seven-class (0.92) classifications.
- The success of ROS (SMOTE) in the seven-class PCI prediction was marked by a significant increase in the Kappa statistical value (from 0.58 to 0.91). Conversely, RUS proved unsuccessful, likely due to the resulting reduction of an already limited database. However, the limited number of initial instances in certain classes (e.g., 'failed', 'good', and 'fair') might have constrained the SMOTE technique's effectiveness in generating meaningful synthetic cases.
- The database must contain sufficient information in both quantity and distress variety, as insufficient data will negatively impact the model's overall performance. However, database imbalance can also significantly impact the training of ML algorithms, even when the total number of cases is adequate.
- The analysis of the independent variables' influence showed that, consistently across the best-performing models, Joint Reflection Cracking, Raveling, Weathering, and Patching and Utility Cut Patching were the most critical distresses for PCI prediction within the tested database. These results are consistent with the findings presented in the literature (e.g., [10,31,51]).
- Despite the limitations imposed by a relatively small database size, limited geographical scope, and airport size, the demonstrated benefits and time savings of using ML algorithms for PCI prediction are promising. These findings underscore the need to organize larger and more diverse databases. Once such complete databases are achieved, the utilization of ML approaches will be especially valuable for resource-limited airports or management entities. By allowing the incorporation of AI-based approaches with low computational demands, this strategy significantly enhances efficiency and improves decision-making in airport pavement maintenance management, particularly for crucial assets like runways.

Based on the promising results of the ML-based approach proposed in this study, a concerted, global effort to create a collaborative open-access database for airport pavement information is recommended. The structure and management of this resource should mirror the successful model of the Long-Term Pavement Performance (LTPP) database for road pavements [52] to ensure data standardization and long-term utility. Implementing this universal database is the essential next step to rapidly advance research and maintenance efficiency within the airport pavement management community.

In conclusion, the prediction of the PCI value from pavement distress data using ML provides a crucial advantage by overcoming the limitations of conventional manual calculation. The traditional approach is highly labor-intensive and vulnerable to inconsistencies arising from human interpretation. Consequently, integrating ML into airport pavement

management systems represents a strategic opportunity to optimize pavement evaluation, maintenance, and performance, ultimately ensuring longer pavement life and greater user safety.

Future research will focus on three main developments: exploring the use of larger databases to improve model performance; investigating the possibility of building models that are not constrained by all pavement distresses or severity levels defined by the ASTM D5340 standard [8] (by identifying the most contributing attributes); and applying advanced boosting-based machine learning algorithms such as Gradient Boosting Machine (GBM), CatBoost, and XGBoost to develop and optimize predictive models.

**Author Contributions:** Conceptualization, A.S. and B.S.; methodology, B.S., A.S. and P.A.; validation, A.S. and B.S.; formal analysis, A.S. and B.S.; investigation, A.S., B.S. and P.A.; writing—original draft preparation, A.S. and B.S.; writing—review and editing, B.S.; supervision, B.S. and P.A.; funding acquisition, B.S. and P.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the GeoBioTec Research Unit through the strategic project UIDB/04035/2025, funded by Fundação para a Ciência e a Tecnologia, IP/MCTES, through national funds (PIDDAC).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Restrictions apply to the data availability. The data were obtained from airport management entities and can be made available by the corresponding author upon their authorization.

**Acknowledgments:** The authors acknowledge the University of Beira Interior and GEOBIOTEC—GeoBioSciences, GeoTechnologies and GeoEngineering (UID/GEO/04035/2025) for supporting the performed study; and Cabo Verde Airports.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Ritchie, H.; Rodés-Guirao, L.; Mathieu, E.; Gerber, M.; Ortiz-Ospina, E.; Hasell, J. Population Growth. Available online: <https://ourworldindata.org/population-growth> (accessed on 28 July 2024).
2. Federal Aviation Administration. *Federal Aviation Administration Advisory Circular No. 150/5320-6G, Airport Pavement Design and Evaluation*; US Department of Transportation Federal Aviation Administration: Washington, DC, USA, 2021; pp. 1–195.
3. Irfan, M.; Khurshid, M.; Iqbal, S.; Khan, A. Framework for Airfield Pavements Management—An Approach Based on Cost-Effectiveness Analysis. *Eur. Transp. Res. Rev.* **2015**, *7*, 13. [CrossRef]
4. National Academies of Sciences, Engineering, and Medicine. *Common Airport Pavement Maintenance Practices*; The National Academies Press: Washington, DC, USA, 2011. [CrossRef]
5. Basu, D.; Misra, A.; Puppala, A. Sustainability and Geotechnical Engineering: Perspectives and Review. *Can. Geotech. J.* **2015**, *52*, 96–113. [CrossRef]
6. Augeri, M.; Greco, S.; Nicolosi, V. Planning Urban Pavement Maintenance by a New Interactive Multi-Objective Optimization Approach. *Eur. Transp. Res. Rev.* **2019**, *11*, 17. [CrossRef]
7. Slabej, M.; Grinč, M.; Kováč, M.; Decký, M.; Šedivý, Š. Non-Invasive Diagnostic Methods for Investigating the Quality of Zilina Airport's Runway. *Contrib. Geophys. Geod.* **2015**, *45*, 237–254. [CrossRef]
8. *ASTM D5340-23*; ASTM International Standard Test Method for Airport Pavement Condition Index Surveys. *Advancing Standards Transforming Markets*; West Conshohocken, PA, USA, 2023.
9. U.S. Army Corps of Engineers, Engineer Research and Development Center (ERDC), Construction Engineering Research Laboratory (CERL). *Asphalt Surfaced Roads & Parking Lots-Paver™ Distress Identification Manual*; U.S. Army Corps of Engineers: Champaign, IL, USA, 2009.
10. Santos, B.; Almeida, P.; Feitosa, I.; Lima, D. Validation of an Indirect Data Collection Method to Assess Airport Pavement Condition. *Case Stud. Constr. Mater.* **2020**, *13*, e00419. [CrossRef]

11. Santos, B.; Gavinhos, P.; Almeida, P.; Nery, D. Use of Unmanned Aerial Vehicles (UAVs) for Transport Pavement Inspection. In Proceedings of the 5th International Conference on Transportation Geotechnics (ICTG) 2024, Volume 1, Lecture Notes in Civil Engineering, Sydney, Australia, 20–22 November 2024; Volume 402, pp. 1–9. [\[CrossRef\]](#)
12. Walker, D. *PASER Asphalt Roads Pavement Surface Evaluation and Rating PASER Manual Asphalt Roads*; Madison: Lakewood, OH, USA, 2002.
13. Shahin, M. *Pavement Management for Airports, Roads, and Parking Lots*, 2nd ed.; Springer: New York, NY, USA, 2005; ISBN 0387234640.
14. Lima, D. Airport Pavement Management System for Cape Verde. Master Thesis, University of Beira Interior, Covilhã, Portugal, 2016. (In Portuguese).
15. Osman, S.; Almoshaogeh, M.; Jamal, A.; Alharbi, F.; Al Mojil, A.; Dalhat, M. Intelligent Assessment of Pavement Condition Indices Using Artificial Neural Networks. *Sustainability* **2022**, *15*, 561. [\[CrossRef\]](#)
16. Carvalho, T.; Soares, F.; Vita, R.; Francisco, R.; Basto, J.; Alcalá, S. A Systematic Literature Review of Machine Learning Methods Applied to Predictive Maintenance. *Comput. Ind. Eng.* **2019**, *137*, 106024. [\[CrossRef\]](#)
17. Singh, A.; Thakur, N.; Sharma, A. A Review of Supervised Machine Learning Algorithms. In Proceedings of the 10th INDIA-Com, 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 16–18 March 2016; pp. 1310–1315.
18. Maroco, J. *Análise Estatística Com o SPSS Statistics*; ReportNumber: Pêro Pinheiro, Portugal, 2014; ISBN 978-989-96763-4-3.
19. Sarker, I. Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Comput. Sci.* **2021**, *2*, 160. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Kim, S.-J.; Bae, S.-J.; Jang, M.-W. Linear Regression Machine Learning Algorithms for Estimating Reference Evapotranspiration Using Limited Climate Data. *Sustainability* **2022**, *14*, 11674. [\[CrossRef\]](#)
21. Schonlau, M.; Zou, R. The Random Forest Algorithm for Statistical Learning. *Stata J.* **2020**, *20*, 3–29. [\[CrossRef\]](#)
22. Kotsiantis, S. Supervised Machine Learning: A Review of Classification Techniques. *Informatika* **2007**, *31*, 249–268.
23. Rudin, C.; Radin, J. Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson from an Explainable AI Competition. *Harv. Data Sci. Rev.* **2019**, *1*, 1–10. [\[CrossRef\]](#)
24. Tu, J. Advantages and Disadvantages of Using Artificial Neural Networks versus Logistic Regression for Predicting Medical Outcomes. *J. Clin. Epidemiol.* **1996**, *49*, 1225–1231. [\[CrossRef\]](#)
25. Chawla, N.; Bowyer, K.; Hall, L.; Kegelmeyer, W. SMOTE: Synthetic Minority Over-Sampling Technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [\[CrossRef\]](#)
26. Johnson, J.; Khoshgoftaar, T. Survey on Deep Learning with Class Imbalance. *J. Big Data* **2019**, *6*, 27. [\[CrossRef\]](#)
27. Paracha, A.; Arshad, J.; Farah, M.; Ismail, K. Machine Learning Security and Privacy: A Review of Threats and Countermeasures. *EURASIP J. Inf. Secur.* **2024**, *2024*, 10. [\[CrossRef\]](#)
28. Xu, Z.; Saleh, J. Machine Learning for Reliability Engineering and Safety Applications: Review of Current Status and Future Opportunities. *Reliab. Eng. Syst. Saf.* **2021**, *211*, 107530. [\[CrossRef\]](#)
29. Rebolledo, J.; León, R.; Camapum de Carvalho, J. Performance Evaluation of Rigid Inclusion Foundations in the Reduction of Settlements. *Soils Rocks* **2019**, *42*, 265–279. [\[CrossRef\]](#)
30. Barua, L.; Zou, B. Planning Maintenance and Rehabilitation Activities for Airport Pavements: A Combined Supervised Machine Learning and Reinforcement Learning Approach. *Int. J. Transp. Sci. Technol.* **2022**, *11*, 423–435. [\[CrossRef\]](#)
31. Pietersen, R.; Beauregard, M.; Einstein, H. Automated Method for Airfield Pavement Condition Index Evaluations. *Autom. Constr.* **2022**, *141*, 104408. [\[CrossRef\]](#)
32. Ali, A.; Esekbi, M.; Sreh, M. Predicting Pavement Condition Index Using Machine Learning Algorithms and Conventional Techniques. *J. Pure Appl. Sci.* **2022**, *21*, 304–309. [\[CrossRef\]](#)
33. Kheirati, A.; Golroo, A. Machine Learning for Developing a Pavement Condition Index. *Autom. Constr.* **2022**, *139*, 104296. [\[CrossRef\]](#)
34. Issa, A.; Samaneh, H.; Ghanim, M. Predicting Pavement Condition Index Using Artificial Neural Networks Approach. *Ain Shams Eng. J.* **2022**, *13*, 101490. [\[CrossRef\]](#)
35. Lin, L.; Li, S.; Wang, K.; Guo, B.; Yang, H.; Zhong, W.; Liao, P.; Wang, P. A New FCM-XGBoost System for Predicting Pavement Condition Index. *Expert. Syst. Appl.* **2024**, *249*, 123696. [\[CrossRef\]](#)
36. Shaheen, M.; Elsayed, R.A.; Ghazoly, H.; Bekheet, W.A. Explainable and Economical AI-Based Approach for PCI Assessment. *Int. J. Pavement Eng.* **2025**, *26*, 2531195. [\[CrossRef\]](#)
37. Frank, E.; Hall, M.; Witten, I. *The WEKA Workbench*. In *Data Mining*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 553–571.
38. Yang, C.; Fridgeirsson, E.; Kors, J.; Reys, J.; Rijnbeek, P. Impact of Random Oversampling and Random Undersampling on the Performance of Prediction Models Developed Using Observational Health Data. *J. Big Data* **2024**, *11*, 7. [\[CrossRef\]](#)
39. Hayaty, M.; Muthmainah, S.; Ghufuran, S. Random and Synthetic Over-Sampling Approach to Resolve Data Imbalance in Classification. *Int. J. Artif. Intell. Res.* **2021**, *4*, 86. [\[CrossRef\]](#)

40. Studart, A. Application of Artificial Intelligence Techniques. Machine Learning for Airport Pavement Condition Index (PCI) Assessment. Master Thesis, University of Beira Interior, Covilhã, Portugal, 2024.
41. Schober, P.; Schwarte, L. Correlation Coefficients: Appropriate Use and Interpretation. *Anesth. Analg.* **2018**, *126*, 1763–1768. [[CrossRef](#)] [[PubMed](#)]
42. Mchugh, M. Interrater Reliability: The Kappa Statistic. *Biochem. Med.* **2012**, *22*, 272–282. [[CrossRef](#)]
43. Hosmer, D.; Lemeshow, S.; Sturdivant, R. *Applied Logistic Regression*; Wiley Series in Probability and Statistics; Wiley: Hoboken, NJ, USA, 2013; ISBN 9780470582473.
44. Domingos, A. Determination and Interpretation of the PCI for Airport Pavements. Master Thesis, University of Beira Interior, Covilhã, Portugal, 2017. (In Portuguese).
45. *ASTM D5340-12*; ASTM International Standard Test Method for Airport Pavement Condition Index Surveys. *Advancing Standards Transforming Markets*: West Conshohocken, PA, USA, 2012.
46. Quinlan, J. Learning with Continuous Classes. In Proceedings of the 5th Australian Joint Conference on Artificial Intelligence, Hobart, Tasmania, 16–18 November 1992; pp. 343–348.
47. Salzberg, S. C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Mach. Learn.* **1994**, *16*, 235–240. [[CrossRef](#)]
48. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
49. Shevade, S.; Keerthi, S.; Bhattacharyya, C.; Murthy, K. Improvements to the SMO Algorithm for SVM Regression. *IEEE Trans. Neural Netw.* **2000**, *11*, 1188–1193. [[CrossRef](#)]
50. Smola, A.; Schölkopf, B. A Tutorial on Support Vector Regression. *Stat. Comput.* **2004**, *14*, 199–222. [[CrossRef](#)]
51. Feitosa, I.; Santos, B.; Gama, J.; Almeida, P.G. Statistical Analysis of an In-Vehicle Image-Based Data Collection Method for Assessing Airport Pavement Condition. *Case Stud. Constr. Mater.* **2025**, *22*, e04792. [[CrossRef](#)]
52. Federal Highway Administration. *Long-Term Pavement Performance Information Management System User Guide (Publication No. FHWA-HRT-21-038)*; U.S. Department of Transportation: Washington, DC, USA, 2021.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.