Application of Artificial Intelligence techniques
Machine Learning for airport pavement condition index (PCI) assessment

André de Paula Pessoa Studart

Dissertação para obtenção do Grau de Mestre em
Engenharia Civil
(2º ciclo de estudos)

Orientadora: Prof. Doutora Bertha Maria Batista dos Santos
Co-orientador: Prof. Doutor Pedro Gabriel de Faria Lapa Barbosa de Almeida

junho de 2024

Declaração de Integridade

Eu, André de Paula Pessoa Studart, que abaixo assino, estudante com o número de inscrição (M13098) de/o Engenharia Civil da Universidade da Beira Interior declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o **Código de Integridades da Universidade da Beira Interior**.

Mais concretamente afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, que em particular atendi à exigida referenciação de frases, extratos, imagens e outras formas de trabalho intelectual, e assumindo assim na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 07/06/2024

André Studart

Dedicatory

To my blood Family and chosen family, my friends.

Acknowledgment

I would like to thank my family for all their support throughout my journey, my advisor and co-advisor for showing me the right path to follow, my friends and to UBI´s Civil Engineering and Architecture department (DECA) for providing all the necessary support to carry out this dissertation.

In addition, I would also like to thank Débora Lima, Adriana Domingos, Ianca Feitosa, Aeroporto e Segurança Aérea (ASA) - Cape Verde´s airport operator - for providing the necessary database used for ML´s application and evaluation.

Resumo

A avaliação da qualidade dos pavimentos é uma componente essencial dos Sistemas de Gestão da Conservação de Pavimentos Aeroportuários. Esta componente permite sustentar a definição de estratégias e programas de conservação que têm como objetivo assegurar o nível de serviço e a segurança de operações e utentes.

Nesse contexto, diversas metodologias existentes contribuem para avaliar o estado geral dos pavimentos. Contudo, tais metodologias muitas vezes demandam uma grande quantidade de tempo, desde a realização de inspeções *in-situ* ao procedimento de cálculo, que pode revelar-se complexo. Dessa forma, com a recente revolução provida pela indústria 4.0, o surgimento de novas tecnologias na área digital, mais especificamente da Inteligência Artificial (IA), faz com que o conceito de *Machine Learning* emerja como ferramenta de alto potencial para análise e processamento dos dados obtidos nas inspeções *in-situ,* contribuindo para otimizar as metodologias atuais de análise geral do estado dos pavimentos, modernizando a Engenharia de Infraestruturas de Transporte.

Portanto, a aplicação dessa técnica na avaliação de pavimentos aeroportuários contribui para a redução do tempo e da complexidade de cálculo do PCI – *Pavement Condition Index*. O grau de fiabilidade vai depender essencialmente do tamanho e das características da base de dados usada na modelação.

O presente trabalho tem como objetivo identificar os algoritmos de machine learning mais adequados à modelação do índice PCI a partir da informação recolhida sobre as degradações presentes à superfície dos pavimentos, de acordo com a norma ASTM D5340-23. Para atingir este objetivo é usada uma base de dados de degradações com informação respeitante a 261 unidades de amostra de pavimentos aeroportuários e considerada a variável PCI traduzida de 3 formas distintas – PCI numérico (0-100), PCI categórico com 3 classes (Bom, regular e ruim) e PCI categórico com 7 classes (Excelente, Bom, satisfatório, regular, ruim, muito ruim, desgastado e falha). Os algoritmos de regressão linear, *decision tree, random forest, artificial neural network e support vector machine* foram testados no software WEKA para o conjunto de dados disponíveis e para 3 modelos de aprendizado – conjunto de treinamento, validação cruzada com 10 dobras e divisão percentual de 80% para treino e 20% para teste.

Os resultados obtidos confirmam a capacidade dos modelos para traduzir o valor do PCI a partir da informação sobre a densidade e nível de gravidade das degradações superficiais, e por consequência a viabilidade da utilização de abordagens baseadas em Machine Learning no processo de cálculo do índice PCI, destacando-se o algoritmo *random forest* com validação cruzada com 10 dobras.

Palavras-chave

Pavimentos aeroportuários; Degradações de pavimentos; *Pavement Condition Index (PCI); Machine Learning*.

Abstract

The evaluation of pavement quality is an essential component of Airport Pavement Management Systems as it supports maintenance strategies aimed at ensuring service levels and the safety of operations and users.

In this context, diverse methodologies contribute to assess the overall condition of pavements. However, these methodologies often require a significant amount of time, from conducting *in-situ* inspections to performing complex calculations. Thus, with the recent 4.0 industry revolution, new digital technologies, namely Artificial Intelligence (AI), has led the concept of Machine Learning to emerge as a highly potential tool for analyzing and processing data obtained from *in-situ* inspections. This contributes to optimize current methodologies for pavement condition analysis, modernizing Transportation Infrastructure Engineering.

Therefore, the application of this technique in airport pavement evaluation helps reduce the time and complexity involved in calculating the Pavement Condition Index (PCI). The reliability degree mainly depends on the size and characteristics of the database used in the modeling.

The present work aims to identify the most suitable machine learning algorithms for modeling the PCI index considering surface pavement distress, based on ASTM D5340-23 standard. To achieve this objective, a pavement distress density and severity database of 261 airport runway sample units was used, considering the PCI in three distinct ways – numerical PCI (ranging from 0-100), categorical PCI with 3 classes (Good, Fair, and Poor), and categorical PCI with 7 classes (Excellent, Good, Satisfactory, Fair, Poor, Very Poor, Failed). Linear regression, decision tree, random forest, artificial neural network, and support vector machine algorithms were tested in WEKA software for three learning processes – use training set, 10-fold cross-validation, and 80% training and 20% testing split.

Found results confirmed the models´ capability to output the PCI value based on the density and severity level of pavement surface distress, indicating the feasibility of using Machine Learning-based algorithms for the PCI calculation process, highlighting the random forest algorithm with 10-fold cross-validation as the better performance model.

Keywords

Airports pavements; Pavement distress; *Pavement Condition Index (PCI); Machine Learning*.

Index

List of figures

List of tables

List of equations

List of acronyms

AI	Artificial Intelligence
PCI	Pavement Condition Index
GRP	Gabinete de Relações Públicas
UBI	Universidade da Beira Interior
DECA	Departamento de Engenharia Civil e Arquitetura
ASA	Aeroportos e Segurança Aérea
ML	Machine Learning
SLA	Self Learning Algorithm
ANN	Artificial Neural Network
RF	Random Forest
DT	Decision Tree
LR	Linear Regression
SVM	Support Vector Machine
ROS	Random Over Sampling
RUS	Random Under Sampling
CC	Correlation Coefficient
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
RAE	Relative Absolute Error
RRSE	Root Relative Squared Error
aPCI	Mean actual PCI value
pPCI	Mean predicted PCI value
CV10	Cross validation in 10 folds
IoT	Internet of Things
UAV	Unmanned Aerial Vehicle

1 Introduction

1.1 Framework

Considering the roads and airports pavements´ importance in connecting countries and cities, it is needed to understand how its design and performance are considered. Several factors can affect their short and long-term behavior. Subgrade support load capacity, pavement layers aggregate´s and binder´s quality, climate and drainage conditions, and growth rate are some of the factors that should be taken into account.

Based in the materials used in layers, different types of pavements can be considered. The ones based on bituminous material are the most common (asphalt pavements). Because of their use by traffic and exposure to weather conditions, the quality of the pavement degrades over time. According to Feitosa (2020), the pavements must withstand distresses to ensure the users´ safety, while providing friction and braking capacity for the airplanes. Budgeting, non-preventing maintenance programs and resources availability often represent some of the strands that can limit the correct maintenance of the pavement. Additionally, the overall condition of the pavement can vary from point to point, being hard to assess exactly the best timing of intervention. Consequently, the importance of pavement quality assessment is a must and will be the topic addressed in this master´s thesis, that will focus on airport pavements.

For the evaluation of pavement´s damage state, the *pavement condition index* (PCI) methodology, presented in D5340 standard (ASTM International, 2023), is the most commonly used. According to the standard, seventeen types of pathologies can be considered when evaluating the pavement condition: alligator cracking, bleeding, block cracking, corrugation, depression, jet blast erosion, joint reflection cracking, longitudinal and transverse cracking, oil spillage, patching and utility cut patching, polished aggregate, raveling, rutting, shoving, slippage cracking, swell and weathering (ASTM International, 2023). It consists of analyzing the pavement distress through a manual, in vehicle or UAV inspection proceeded by a meticulous calculation of pavement condition. Maintenance plans, and lifecycle expansion are some of the benefits of using PCI methodology to evaluate pavement condition. Although efficient, this methodology can take considerable time to be applied, as the required calculations to evaluate each section of the airport pavement represents a long time-consuming process (Osman et al., 2022). Hence, finding new possibilities to optimize this process is a pressing need, lowering overall PCI determination related costs.

Considering the current 4.0 industrial revolution, the digital world is rapidly evolving and providing new tools for humanity. Internet of Things (IoT), Artificial Intelligence (AI) and machine learning (ML) are some of the recently developed tools. The latter is mostly based on advanced statistics that can predict values based on learning-algorithms derived from data related to previous situations. Considering the wide range of possibilities and its high potential, the use of ML approaches for PCI calculation can be feasible, boosting the digital revolution also in the fiend of airport pavement maintenance.

Therefore, the present research aims to evaluate the application of ML algorithms for predicting PCI values from airport pavement distress data. Several algorithms will be tested based on existing distress data and PCI values of 3 international airports.

In addition, a comparison between real PCI data and PCI results obtained with an algorithm built from a self-enlarging database will also be conducted. The analysis and validation of the results will be performed based on the statistics provided by the software WEKA (Frank et al., 2017), analyzing the feasibility of using ML approaches to optimize PCI determination.

1.2 Objectives

The present work aims to evaluate the feasibility of using ML algorithms to calculate PCI of airport sample units, according to ASTM D5340-23. To achieve this, different types of ML algorithms – linear regression (LR), support vector machine (SVM), artificial neural network (ANN), decision tree (DT) and random forest (RF) - will be tested for PCI prediction models based on the existing pavement distress data and PCI values of 3 international airports. Differences between the algorithms and results will be described, exposing their reliability compared to traditional PCI calculation. In addition, larger comprehension on ML *modus operandi,* PCI procedure and pathologies within airport pavement environment are also addressed, supporting the understanding of how the digital revolution – namely ML – can contribute for optimizing current standardized pavement maintenance procedures.

1.3 Document structure

The present work is divided into seven chapters, the first one is entitled "Framework" and consists in presenting the topic to be developed, exposing the problematic, objectives and relevance of the research.

The second chapter "Airport pavements characteristics and condition" presents an overview of airport pavements structure and composition, main type of pavement surface distress and pavement maintenance plans definition, providing an overview of the standardized procedure for PCI calculation based on ASTM D5340-23.

In chapter three, "Machine learning and pavement evaluation", an overview on ML definition, the *modus operandi* and a state of art on ML applied to pavement condition assessment, exposing benefits and drawbacks associated to several ML algorithms are presented.

The "Methodology" proposed for PCI assessment using ML is presented in chapter four. The chosen software, data requirements, assumptions, algorithms and analysis options are presented, providing a clear view of the overall PCI assessment process.

The fifth chapter presents the “Case study”. This chapter aims to validate the feasibility of the proposed ML approach and identified adequate techniques to deal with specific datasets characteristics, such as dimensionality and sparsity.

Finally, the seventh chapter presents the main research conclusions, showcasing bullet points while indicating limitations and future research possibilities.

2 Airport pavements characteristics and condition

2.1 Characteristics

Airports pavements are crucial to ensure the airports proper functioning, as they must have specific properties, such as resisting intense weather and high loads (heavy-loaded airplanes), presenting friction and braking capacity, while also being properly cleaned, as hydrocarbons can be present in the surface. The design of flexible pavements has to consider variables inherent to airports, such as aircraft volumes and loads.

The basic structural model for road flexible pavement in Portugal has similarities with the American model, mainly composed by the following layers (Santos, 2009; U.S Department of Transportation, 2006):

- Wear or surface layer: high quality bituminous material, providing durability and impermeability, representing a high percentage of the total pavement cost.
- Regularization layer: Bituminous material to ensure compatibility and bond between wear layer and aggregate layers, serving also for transferring the load.
- Base: layer generally consisting of aggregate material, representing a high cost.
- Sub-base: layer consisting of intermediary resistance, having a lower cost compared to the base layer.
- Foundation: generally, consisting of *in-situ* material with lower range of load capacity support.

In airports, the layering is similar, relaying the difference on the quality of materials and capacity of load support, being the base and wear layer the most cost-impacting layers, needing to provide proper design-characteristics to attend the desired service life (Federal Aviation Administration, 2021; Domingos, 2017). Figure 2.1 exposes the typical flexible pavement structure.

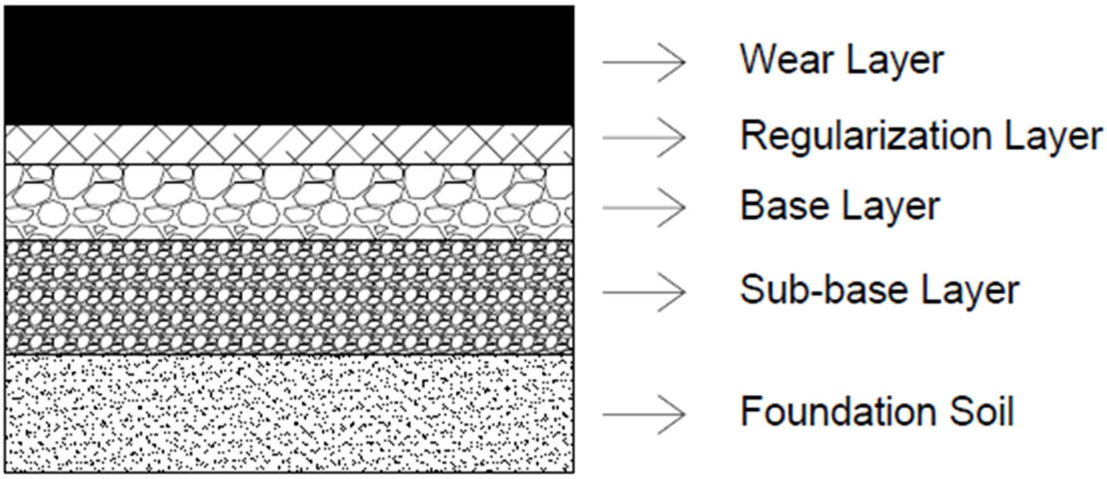


Figure 2.1 – Typical pavement structure (adapted from Federal Aviation Administration, 2021).

Besides the basic structure and traffic volume and load, other components must be taken in consideration while designing the pavement, such as the ground water level and environmental factors (temperature, moisture, and frost action) in order to ensure the pavement´s serviceability. Conceiving a pavement for all-type demand can be utopic and highly expensive. Thus, it is necessary to understand and identify the variables to prioritize during its design. Seasonal frost, for example, should be taken into consideration in areas that suffer from this type of weather, as the freeze-thaw cycle can cause the cracking of the pavement.

Overall, for layer design, the closer to the surface a layer is, the smaller the diameter of the aggregate– generally associated to fine sand -, while the opposite happens as the deeper the layer is – generally associated to gravels. This ensures an efficient load transfer between closer-to-surface layers to deeper layers, dissipating the high demands from the first layers to the subsequent layers. Grain-by grain contact is more uniform, transferring the loads successfully to deeper layers (Giwangkara, et al., 2020).

In addition, it is equally important to reiterate the quality control during the construction of the pavement, as contamination from the equipment and unproper compaction can impact the performance of the pavement throughout its lifecycle, compromising its support capacity (Federal Aviation Administration, 2021).

Although the proper design of the pavement can avoid the emergence and development of pathologies, weathering will always occur, as the use of the pavement and repeated traffic-load cause the soil to suffer settlements, affecting the pavements´ support layers. As a result of this use, pathologies can emerge. Thus, maintenance is an important variable to be also considered during the pavement´ lifecycle to ensure that the project´s characteristics persist over time. As an effect of the use, pathologies start to emerge, being necessary their evaluation for defining and proceeding to the pavement maintenance plan.

2.2 Pathologies

The pathologies associated to airport pavements emerge as an effect from its use or lack of proper design and this phenomena intensifies when pavements are not properly maintained. Pavement design is usually based on the mass of aircraft, generally higher than 5700 kg, aircraft landing gear and traffic load. Currently, air traffic has been growing, thus, elevating the pavement load-demands, which can approach their limit state. As a consequence, the pavement wear is augmented, shifting the attention from the construction of new pavements to the maintenance and rehabilitation of the existing ones (Irfan et al., 2015).

In addition, during the pavements´ lifecycle, their resilience modulus will vary, due to external events, such as earthworks, or continuous loads, changing their load support capacity over time (Basu et al., 2015), as shown on figure 2.2.

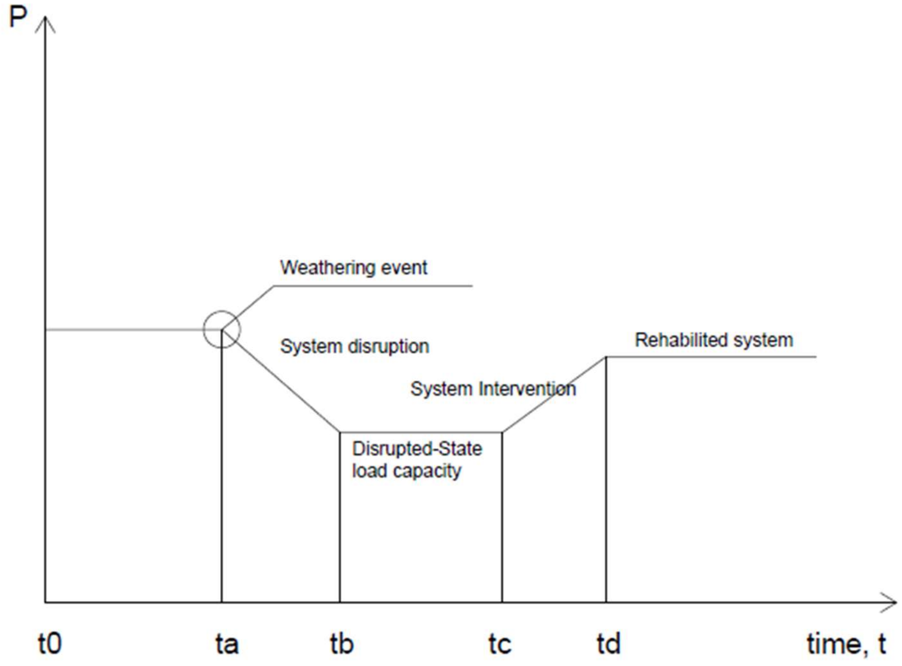


Figure 2.2 – Pavement lifecycle and intervention impact (adapted from Basu et al., 2015). Note: P means pavement performance.

Considering the necessity of intervention on airport runways, the use of life-cycle cost analysis is essential for a proper maintenance-budget operation. The correct type and timing for an intervention can be addressed after a field survey, analyzing the associated pathologies and overall pavement´ state. (Augeri, et al., 2019; Irfan et al., 2015). PCI determination can support maintenance plans definition and save costs if correctly used as a trigger for preventive maintenance, avoiding future significant and costly labors. Table 2.1 compares intervention timing between preventive maintenance, rehabilitation treatment and minimum serviceability.

Table 2.1 - Comparison between intervention timing (adapted from Federal Aviation Administration, 2021; Irfan et al., 2015).

Type of intervention	**Cost-impact**	**Performance improvement**	**PCI Values**
Preventive maintenance (PM)	Low cost	Considerable	70-100
Rehabilitation treatment trigger (RTT)	High cost	High	55-70
Minimum level serviceability	Very high cost	Pavement overdo	0-55

Thus, the latter the maintenance starts, the more expensive it is, being up to five times more expensive to rehabilitate than to maintain a pavement (Domingos, 2017), stating a tendency such as the 20/80 rule theorized by Pareto, which indicates that 80% of the problems could be avoided if action were taken in the first 20% of the causes.

Nevertheless, due to lack of budget or necessary equipment, political bureaucracy, severe weather, low-quality construction procedures, low quality control or unforeseen variables, several pathologies can emerge, impacting the overall pavement´ performance. Table 2.2 synthetizes the main types of pathologies that can be found on the surface of airport pavements according to the ASTM D5340-23 (ASTM International, 2023; USACE, 2009).

Table 2.2 - Pathologies synthesis on airfields pavements adapted from (ASTM International, 2023; USACE, 2009).

Pathology/Distress type	Description	Origin	How to measure / Severity
Alligator cracking	Interconnected cracks as sharp angled pattern pieces on the surface.	Surface layer fatigue cause by high traffic loads.	Surface area with low, medium and high distress severity.
Bleeding	Excessive film of bituminous material on the surface, sticky surface.	Excessive amounts of bituminous material, separating from the mixture.	Surface area with low, medium and high distress severity.
Block cracking	Interconnected cracks on rectangular shape.	Shrinking of surface layer due to temperature variation.	Surface area with low, medium and high distress severity.
Corrugation	Series of closely spaced ridges and valleys.	High traffic loads on an unstable pavement or base surface.	Surface area with low, medium and high distress severity.
Depression	Surface drawdown regions.	Settlement of the layering aggregates.	Measured per depth with low, medium and high distress severity.
Jet Blast	Darkened areas due to bituminous burn.	Localized burnt areas.	Measured per area of the surface, without degree of severity.
Joint Reflection of pavement concrete cement (PCC)	Joint cracking on the pavement.	Movement of pavement concrete cement slab under surface layer.	Measured in linear length, with low, medium and high distress severity.
Long. & Trans. Cracking	Longitudinal or transversal to the pavement center line cracks.	Volumetric variation of the surface layer.	Measured in linear length with low, medium and high distress severity.
Oil Spillage	Deterioration or softening of the surface layer.	Spilling of oil or hydrocarbons from vehicles, planes or equipment.	Surface area with low, medium and high distress severity.
Patching	Punctual or large repairs done in the pavement.	Intervention due to traffic loads and settlement of the subsequent layers.	Surface area with low, medium and high distress severity.
Polished Aggregate	Lack of skid resistance due to polished aggregate.	Repeated traffic applications.	Surface area with low, medium and high distress severity.
Raveling / Weathering	Aggregates segregation.	Incorrect mixing or aggregates level.	Measured per area of the surface with low, medium and high distress severity.
Rutting	Surface depression in wheel path.	Settlement of the supporting layers.	Depth indicates the severity per area with low, medium and high distress.
Shoving from PCC	Pavement´ length increase due to swell and crack.	Opening of the joints due to the filling with uncompressible materials.	Surface area with low, medium and high distress severity.
Slippage Cracking	Cracks with angular definition, generally half-moon shaped.	Pavement deformation due to braking or wheel turning.	Surface area with low, medium and high distress severity.
Swell	Bulge in the pavement´s surface.	Volumetric gain due to frost action or soil swelling.	Surface area with low, medium and high distress severity.
Weathering	Separation of the asphalt binder and fine aggregate matrix from the pavement surface	Surface wear due to traffic load and abrasion	Surface area with low, medium and high distress severity.

Having a high-quality identification and analysis of the pathologies on airport pavements is important, as airports usually do not receive large interventions during functioning, as it would block their operation, focusing the necessary intervention on limited-time windows and pavement inspection on nondestructive tests (Xie et al., 2021).

High-quality assessment can be achieved by PCI determination. This index contemplates the pavements´ overall condition per area, associated to the quality conformity, based in a large pavement distress data collection and treatment. It is considered a reliable method, although presenting some degree of subjective analysis and a time and budget-consuming technique due to the necessarily procedure needed, opening a strand of possible optimization using new digital techniques, such as ML.

2.3 Pavement Condition Index (PCI)

Pavement condition index (PCI) represents the overall state of the pavement, exposing its weathering or integrity degree. It is obtained after a *"in-situ"* visual analysis of sectioned areas of the pavement under evaluation. Afterwards, a series of calculations is conducted to obtain the final index value for the pavement (ASTM International, 2023). PCI values can be classified into seven or three classes, based on values ranging from 0-100 scale, as figure 2.3 exposes.

Figure 2.3 - PCI categoric classification (adapted from ASTM International, 2023).

In order to correctly obtain a PCI value, tree main steps are considered: establishing the number of units to be inspected, the inspection procedure and, finally, PCI determination.

2.3.1. Establishing the number of units to be inspected

In order to establish the number of units to be inspected, the pavement network to be analyzed undergoes three main steps.

1. Pavement network division into branches.
2. Branches division into sections.
3. Section division into sample units.

Branches corresponds to a set of primary division of pavement network, where each branch represents a continuous part of the network. This procedure aids to assign an identifier to each branch, registering relevant information, such as location, length and functional class.

The branches are then divided into sections, representing smaller and manageable segments of the pavement. Sections can be determined by the pavement condition, allowing to identify each section inside the branch, recording pavement type, length and overall state. Then, sections are divided on sample units, consisting of small representative areas within each section for detailed PCI survey. Sample units defined for asphalt pavements typically have a standard size of 450 ± 180 m^2 (ASTM International, 2023). The minimum number of sample units (n) per section to obtain a statistically adequate confidence level (95%), based on assumed standard deviation, is calculated by equation 1:

$$n = \frac{N.s^2}{\left(\left(\frac{e^2}{4}\right)(N-1)+s^2\right)} \qquad \text{(Equation 1)}$$

where:

n = minimum number of sample units to be inspected in the section.
N = total number of samples units in the section.
e = acceptable error margin of ±5 PCI points.
s = standard deviation of the PCI from one sample to another.

The n value must be rounded to the next highest whole number. In addition, if obtaining a 95% confidence level is critical, the adequacy of the n value must be confirmed, calculating the actual standard deviation using the equation 2:

$$s = \sqrt{\sum_{i=1}^{n} \frac{(PCIi - PCIf)^2}{(n-1)}} \qquad \text{(Equation 2)}$$

where:

s = Standard deviation of the PCI from one sample to another.
n = minimum number of sample units to be inspected.
PCIi = PCI of sample unit i.
PCIf = mean PCI of inspected samples units.

If the revised number (s) is greater than the inspected amount of sample units, the additional sample units must be selected, randomly surveyed and evenly spaced to achieve the 95% confidence level. If not necessary, a lesser confidence level can be achieved based on the survey objective.

After having determined the number of sample units to be inspected, calculating the spacing interval (i) of the units by systematic random sampling and rounding to the next lowest whole number is necessary, following equation 3.

$$i = \frac{N}{n} \qquad \text{(Equation 3)}$$

where:

N = total number of samples units in the section.
n = minimum number of sample units to be inspected.

The first unit to be inspected is selected randomly from unit 1 to *i*, being the next one chosen by successive increments of *i* units.

2.3.2 Inspection procedure

The inspection consists in a manual or vehicle-assisted survey to analyze the pavement state. For asphalt surfaces, indicating the sample unit identification, orientation and recording the branch, section number and area are necessary. After initial inputs, the identification of pathologies according to table 2.2, and their degree of severity, on each sample unit is recorded.

Generally, a form is used to register the conducted inspection: type of pathology/distress, degree of severity, length or area affected and number of section that it was identified, as table 2.3 below exemplifies.

Table 2.3 - Example of PCI survey form (adapted from ASTM International, 2023).

AIRFIELD ASPHALT PAVEMENT
CONDITION SURVEY DATA SHEET
FOR SAMPLE UNIT

BRANCH________SECTION________SAMPLE UNIT________
SURVEYED BY_______DATE______SAMPLE AREA__________

SKETCH:

1. Alligator Cracking
2. Bleeding
3. Block Cracking
4. Corrugation
5. Depression
6. Jet Blast
7. Jt. Reflection (PCC)
8. Long. & Trans. Cracking
9. Oil Spillage
10. Patching
11. Polished Aggregate
12. Raveling / Weathering
13. Rutting
14. Shoving from PCC
15. Slippage Cracking
16. Swell

Distress Severity	QUANTITY									TOTAL	DENSITY %	DEDUCT VALUE
8 L	10	20	15							45	0.90	4.8
8 M	9									9	0.18	4.9
1 L	50									50	1.00	21.0
13 L	200	175								375	7.50	27.0
13 M	25									25	0.50	20.0
5 L	15									15	0.30	2.0
5 M	20									20	0.40	9.0
10 L	50									50	1.00	4.0

Dividing the total quantity of each distress and severity level by the sample unit area is necessary to find the density. Afterwards, the deduct value (DV) for each distress and severity level combination is extracted from curves in D5340 standard (ASTM International, 2023), depending on type of pavement, distress level and density.

2.3.3 PCI determination

2.3.3.1 CDV determination

DV per section is considered as the corrected deducted value (CDV) if none or only one individual DV is higher than five. Otherwise, the maximum CDV has to be determined through m value calculation, that correspond to a calculation based maximum allowable number of distresses (m) and using the highest deduct value (HDV) as follows:

$$m = 1 + \left(\frac{9}{95}\right).(100 - HDV) \leq 10 \qquad \text{(Equation 4)}$$

where:

m = maximum allowable number of distresses.
HDV = highest deduct value.

Afterwards, the m largest DVs values are inserted on line 1 and summed, as shown in table 2.4, including the fraction obtained by multiplying the last DV by the fractional portion of m. Values “q” represent the number of DV values higher than five. For the values higher than five, the total and q values are crossed in correction curves, as shown in figure 2.4, depending on the type of pavement, obtaining the CDV values. The line of the table is then duplicated with the same DVs, changing the smallest DV greater than 5 to 5. Such a procedure shall be done until “q” values are “1”.

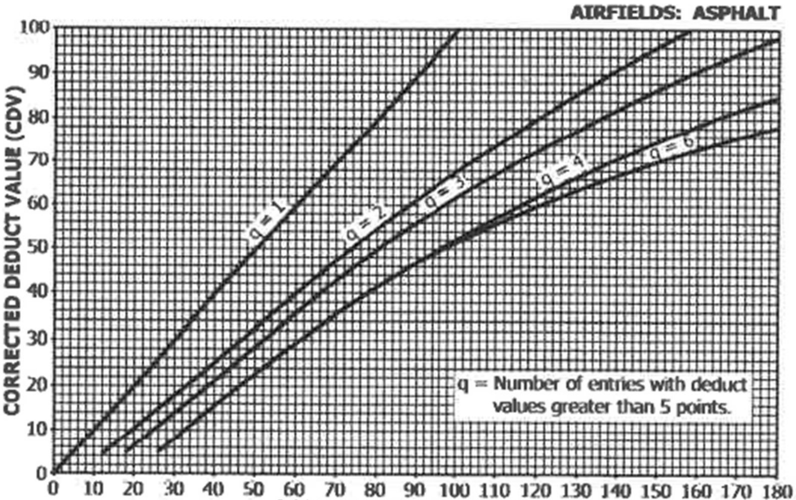


Figure 2.4 - Corrected DVs for flexible airfield pavement (ASTM International, 2023).

When the last line is filled the higher CDV value found (maxCDV) is used to calculate PCI (see equation 5).

Table 2.4 - Example of PCI survey (adapted from ASTM International, 2023).

#	Deduct Values										Total	q	CDV
1	27.0	21.0	20.0	9.0	4.9	4.8	4.0	1.8			92.5	4	50.0
2	27.0	21.0	20.0	5.0	4.9	4.8	4.0	1.8			38.5	3	56.0
3	27.0	21.0	5.0	5.0	4.9	4.8	4.0	1.8			73.5	2	51.0
4	27.0	5.0	5.0	5.0	4.9	4.8	4.0	1.8			57.5	1	57.5
5													

2.3.3.2 PCI calculation

The PCI value is calculated by subtracting the maxCDV from 100, as follows:

$$PCI = 100 - maxCDV \qquad \text{(Equation 5)}$$

where:

PCI = Pavement Condition Index.
maxCDV = Maximum corrected deduct value.

If two severities of one distress are present in the same sample unit, the global distress value has to be computed, as follows.

$$X2 = x1 + x2 \qquad \text{(Equation 6)}$$

where:

x1 = Distress percent of lower severity level.
x2 = Distress percent of higher severity level.

The value of PCI (x1, x2) should be higher than the PCI (0, X2) as the latter has more distress percentage of a higher severity level. If this is not the case, the PCI of the sample unit must be based on X2 and not x1 and x2.

If a sample unit presents tree severity levels of one distress, a PCI for different combinations must be calculated, as shown in table 2.5.

Table 2.5 - Table of combination for PCI determination when tree severity levels of a distress is reported (adapted from ASTM International, 2023).

	Distress severity level	PCI values
Start with:	l, m, h	PCI (l,m,h)
Set (l + m) = M	0, M, h	PCI (0, M, h)
Set (m + h) = H	L, 0, H	PCI (l, 0, H)
Set (l + h) = H	0, m, H	PCI (0, m, H)
Set (l + m + h) = H	0, 0, H	PCI (0, 0, H)

Note: l- low severity level, m- medium severity level, h – high severity level.

The PCI of the sample unit should be the one that provides the highest PCI value.

2.3.3.3 Determination of PCI for the section

The PCI value is a representative value for a unique combination of distress types and severity levels of the pavement. The PCI values obtained for the surveyed sample units, that are selected randomly within the section, are used to obtain the PCI of the section (PCIs), calculated as the area-weighted PCI of the randomly surveyed sample units ($\overline{\text{PCIr}}$), based on equation 7.

$$PCIs = \overline{PCIr} = \frac{\sum_{i=1}^{n}(PCIri.Ari)}{\sum_{i=1}^{n} Ari} \qquad \text{(Equation 7)}$$

where:

PCIs = PCI of the section.
$\overline{\text{PCIr}}$ = area-weighted PCI of the randomly surveyed sample units.
A_{ri} = Area of sample i.
n = number of random sample units surveyed

If additional sample units are surveyed, the area-weighted PCI of the surveyed additional units ($\overline{\text{PCIa}}$) is calculated, based on equation 8. The PCI of the section is then calculated by equation 9.

$$\overline{PCIa} = \frac{\sum_{i=1}^{m}(PCIai.Aai)}{\sum_{i=1}^{m} Aai} \qquad \text{(Equation 8)}$$

$$PCIs = \frac{\overline{PCIr}.(A - \sum_{i=1}^{m} Aai) + \overline{PCIa}\,(\sum_{i=1}^{m} Aai)}{A} \qquad \text{(Equation 9)}$$

where:

$\overline{PCIa}$ = Area-weighted PCI of additional samples units.
PCI_{ai} = PCI of additional sample i.
A_{ai} = Area of additional sample i.
A = Area of the section.
m = number of additional sample units inspected.
PCIs = area-weighted PCI of the pavement section.

Afterwards, with the overall PCI value for the pavement, the state of the pavement can be defined according to figure 2.3, indicating if maintenance or rehabilitation is required or not. If maintenance is necessary, budgeting and plan-of-action will be defined, as for airports pavements the time schedule for intervention has to been mostly during non-use periods and preferably considering fast interventions, in order not to block the airport´s operation. For deep rehabilitation intervention, runway closure periods, in most cases, have to be considered.

Therefore, the calculation of the PCI index involves the segmentation of the pavement in sample units and the evaluation of the pavement surface through a visual inspection that can be time-consuming, as it will be necessary to cover and analyze extensive areas.

New technologies have emerged to assist and optimize the pavement condition inspection process, where 3D laser scanners and other image capture devices installed in motor vehicles and drones have contributed to obtaining the necessary inspection information. AI and ML are indicated as high-tech tools that can drastically enhance the processing of this information, where a large volume of data, referring to visual inspections, can be treated much quickly and models can be derived for the calculation of the PCI.

Such an advance would make a considerable contribution to optimizing the calculation of this index. When associated with digital surveys, it would make the process more efficient, faster and more reliable.

3 Machine Learning and pavement evaluation

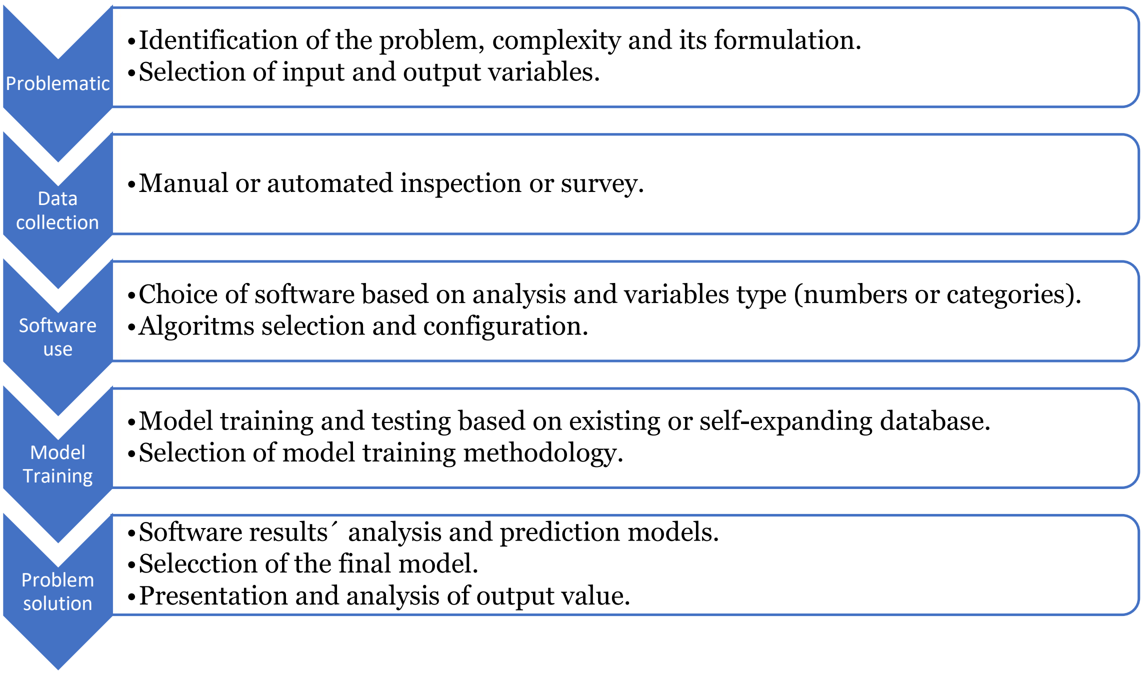
3.1 Overview

The current era of industrial revolution 4.0 exposes the digital world as a relevant and rapidly evolving area for the solution of complex problems. The rapid development of the information age, as seen by the internet of things (IoT), contributes to the expansion and development of all areas and sciences, transforming the way which information and data are processed.

With a great potential for the development of a digital environment due to the processing of a large volume of data at computerized speed, the concept of machine learning (ML) emerges as a tool that investigates and processes high quantities of data so that it can be used to create models for varied applications.

In this context, ML can be characterized as an advanced data analysis approach that is also based on statistics, in order to obtain patterns and correlations between the variables studied from input data. Its concept is also linked to AI, which is a deeper component that involves machine learning and the concept of deep learning, which is generally synthesized by an intelligence that can generate not only models, but also create new information itself.

For the ML concept to be applicable, input data obtained through manual or automated processes is needed. These data will be treated and analyzed by advanced statistics, generating models that can improve as more input data is added to the dataset. Accordingly, algorithms have each a different methodology for treating the given problem, indicating the importance of understanding which algorithm would better fit the analyzed problem. Figure 3.1 exposes a simplified overview on ML problem-solving.



- Identification of the problem, complexity and its formulation.
- Selection of input and output variables.

- Manual or automated inspection or survey.

- Choice of software based on analysis and variables type (numbers or categories).
- Algoritms selection and configuration.

- Model training and testing based on existing or self-expanding database.
- Selection of model training methodology.

- Software results´ analysis and prediction models.
- Selecction of the final model.
- Presentation and analysis of output value.

Figure 3.1 - Simplified overview on ML-managed problem (adapted from Kufel et al., 2023; Reich, 1997; Taye, 2023).

As an example, the recommendation of specific products on social networks is based on machine learning algorithms that analyze how users use the devices and items they search for most and, therefore, constantly learn more about the users. In engineering, ML can also contribute significantly to predictive maintenance, where equipment failures can be identified through its behavior pattern (Carvalho et al., 2019), or in the infrastructure area, where data analysis can identify pathologies and, consequently, the need for maintenance (Barua & Zou, 2022).

The investigation of ML applied in civil engineering has increased exponentially in the recent years, as its potential can be applied for several uses and diverse fields. ML use in civil engineering open bibliographic research was conducted using “Scopus” database, searching for review articles, research articles and book chapters limited to engineering with co-concurrence of “Machine Learning” AND “engineering” words as article title, abstract or keywords. The result is presented in figure 3.2.

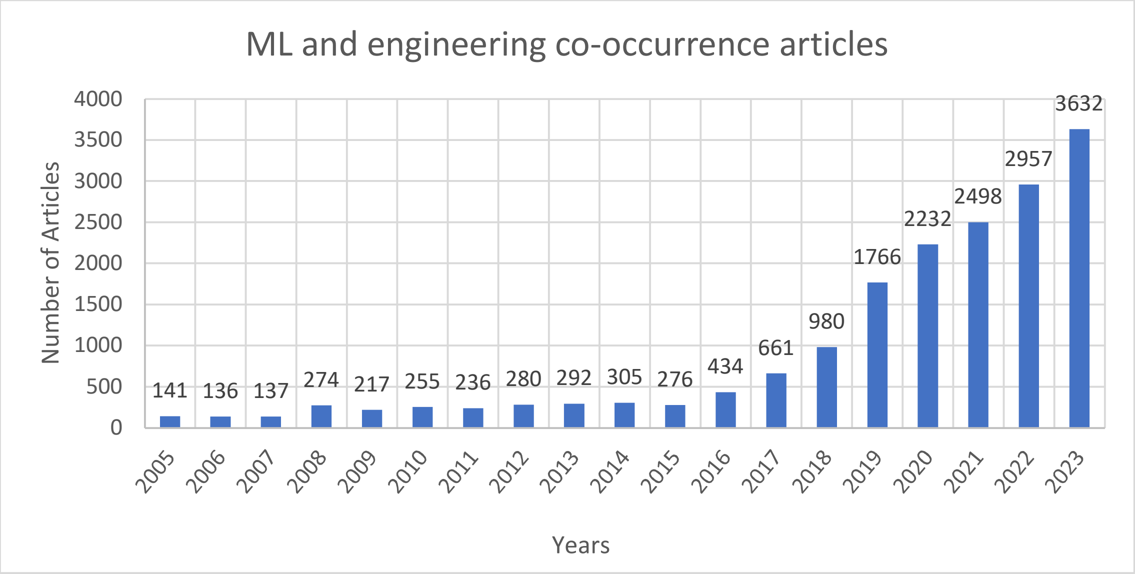


Figure 3.2 - ML and engineering co-occurrence keywords.

Although it has grown considerably, ML is a large concept that can extend between areas and fields, indicating this exponential growth could not have been uniformly present across fields of investigation. ML use for pavement analysis in civil engineering open bibliographic research was conducted using “Scopus” database, searching for the review articles, research articles and book chapters limited to engineering with co-concurrence of “Machine Learning” AND “pavement” words as article title, abstract or keywords, exposed in figure 3.3.

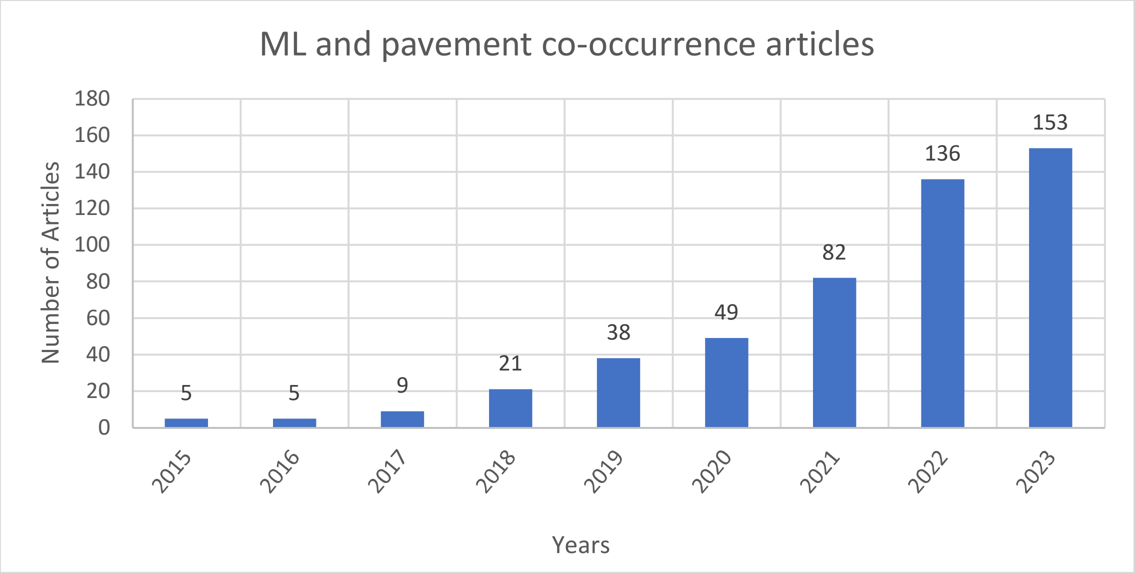


Figure 3.3 – ML and pavement co-occurrence keywords.

ML use for pavements analysis represents a small part of the ML use within civil engineering. Considering the problematic of PCI calculation, fewer papers are found within this topic, exposing that this field is not yet fully covered.

A third search for ML and PCI in civil engineering open bibliographic research was conducted using the same database, searching for review articles, research articles and book chapters limited to engineering with co-concurrence of “Machine Learning” AND “PCI” or “pavement condition index” words as article title, abstract or keywords, exposed in figure 3.4.

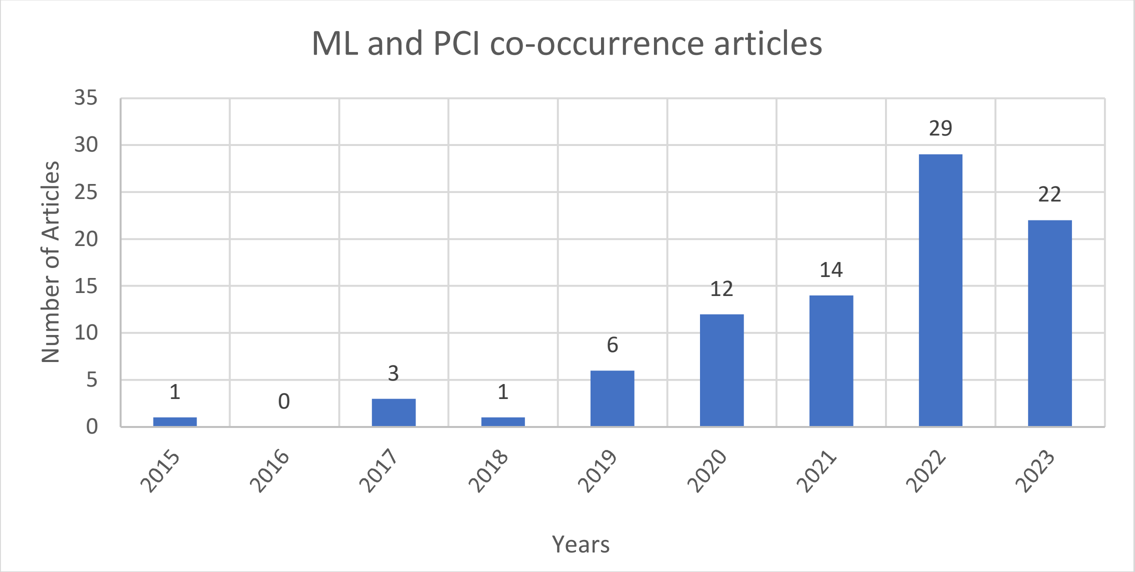


Figure 3.4 – ML and PCI co-occurrence keywords.

Covered topic by published research was correlated considering 99 published articles, conference papers, review papers and book chapter within “Scopus” database searching for the keywords “PCI” OR “pavement condition index” AND “machine learning”. The minimum number of keyword co-occurrences was set as 10, as more keyword occurrences would not fulfill correlation criteria due to few papers on the topic. Figure 3.5 indicates the mentioned correlations between machine learning and PCI.

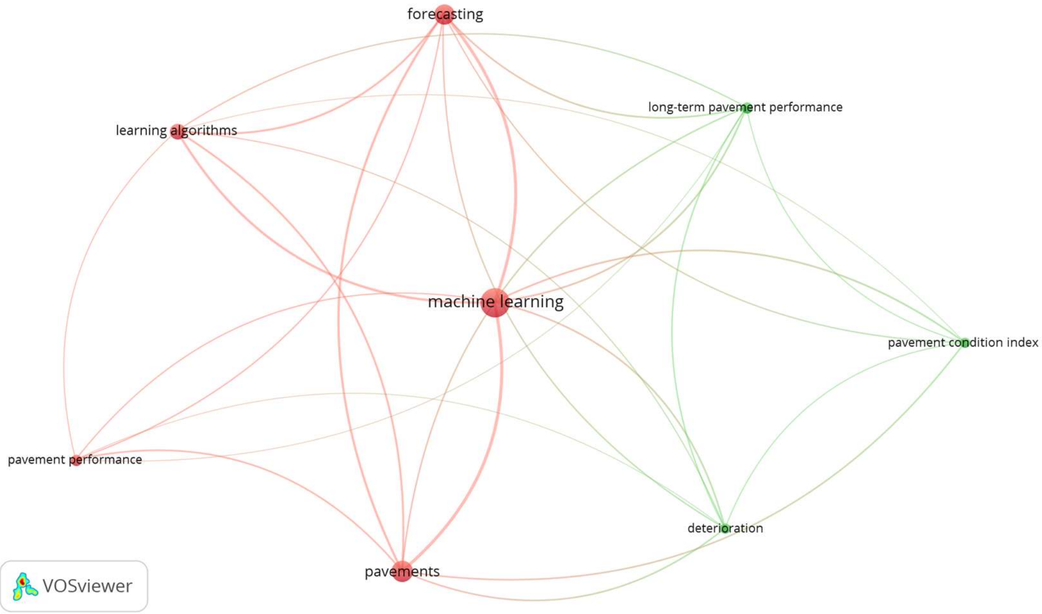


Figure 3.5 - ML and PCI words correlation (analysis performed with VOSviewer).

Results indicate ML´s stronger relationship when associated to pavements and forecasting, which involve areas other than PCI, such as material behavior, design and performance forecasting. The weaker correlation with PCI, although still relevant, suggests the application of ML to PCI calculation is not as developed, indicating a research gap.

Thus, found results indicate ML use for PCI calculation is growing year by year, although further work is necessary to fully cover and develop the area. Researches conducted by (Ali, M. Eskebi, & Sreg, 2022; Marcelino, et al., 2021; Pietersen, et al., 2022; Zhang, et al., 2021) indicate the feasibility of ML and algorithms models for PCI analysis, exposing a field with high development potential.

3.2 ML Algorithms

Several ML algorithms are available and each one of them can be used for specific type of problem-solving. The choice of the algorithm to be used in the analysis is generally based on the type of input data that will be considered, namely numerical or categorical.

Numerical (quantitative) data is usually associated to regression models, as these are used for predicting discrete or continuous data based on numerical input values, allowing establishing a functional mathematic relation between the variables (Maroco, 2014). Categorical classes differ from numerical data in the model presumption and variables definition, which can assume a non-numerical value (Maroco, 2014). Categoric are usually used to predict values into classes, mapping the nominal or ordinal data into a categorical classification, providing a clear classification and categorization (Goneppanavar et al., 2019). Hence, the desired output, numerical or categoric, depends on the nature of the problem and desired outcome, impacting the algorithm to be chosen.

An algorithm can be defined as a set of well-defined rules or instructions that describe a computational process for problem-solving applications. For a specific use, such as PCI calculation, an algorithm would involve a sequence of steps to be followed to process the input data, that is numerical in this specific case (quantities of distress), to obtain the final result, that is a prediction of the value of PCI.

ML can represent an advance on how to manage and treat data, improving the current models, there are several strands of ML than could be applicable, depending on the use, involving varied concepts and models. ML algorithms can be divided into several strands, mainly relying on the type of learning and algorithm to be used, which will depend on the desired application. Supervised learning is often considered more suitable for numerical data, as it is able to produce general patterns and hypotheses based on rules and logic structure (Singh, et al,. 2016). In addition, supervised learning generally uses a part of the dataset to train the algorithms and another part to validate data, not using them for calculating the output, as it could lead to overfitting problems. Supervised learning algorithms include Linear regression (LR), Artificial

Neural Network (ANN), Decision Trees (DT), Random Forests (RF) and Support Vector Machine (SVM), which are some of the most used strands, each having its benefits and disadvantages.

The operating models of ML' algorithms generally vary between opaque (black box), characterized by ANN and SVM, where it functions as a processing black box (opaque), and transparent (clear), characterized by LR and DT, where the final result follows a determined flow. Furthermore, there are several ways to operate ML according to the classification of data, type of training model, association between variables and others factors that directly impact its operation and quality of result (Sarker, 2021). Figure 3.6 demonstrates how ML works in its two aspects: opaque and transparent (clear).

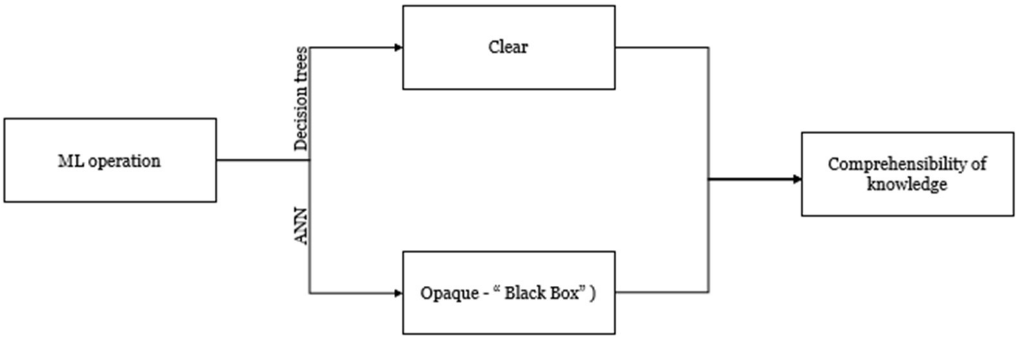


Figure 3.6 - Difference between operating models regarding ML (adapted from Reich, 1997).

Black Box operation is usually defined as a complex mechanism, as the considered variables and how they are related are not available for examination, not providing the user manners to investigate how the final output was obtained, purely based on the trust of its algorithms and equations (Rudin & Radin, 2019). Although no clear indication is given to examine this mechanism, attention to sample size while respecting the cross validation can aid the user to assess such mechanism and to avoid problems such as overfitting and underfitting, while acquiring knowledge regarding their operation.

On the other hand, clear operation relies on comprehensive knowledge, proactively receiving the input data, learning and storing knowledge that can be useful for problem solving (Reich, 1997). Hence, transparent (clear) mechanism is interpretable, such as linear regression and decision trees, enabling user-trust and more likely to be comply with possible regulations to come (Doshi-Velez & Kim, 2017), although their performance can sometimes not match black-box models (Rudin & Radin, 2019).

In addition, for regression problems, results can be assessed by the R^2 value, where the higher R^2 the better. Equally important, is testing Black Box algorithms in real-world scenarios to assess their reliability and avoid overfitting issues, where the output is reliable when under trained circumstances (Bellows et al., 2011).

Considering the numerical input database, linear regression, decision tree, random forest, artificial neural network and support vector machine algorithms are described, while their applicability for categoric PCI is also presented.

3.2.1 Linear Regression (LR)

Linear regression algorithms are typically associated to fast-learning and with a high explanatory power, as it searches for a direct correlation between one dependent variable and one or more independent variables, synthetized by the equation 9 (Kim et al., 2022).

$$y = a1x1 + \cdots + b \qquad \text{(Equation 10)}$$

where:

y = dependent variable (output value).
x1 = independent variable (xi for i variables considered).
a1 = weight of the independent variable.
b = bias or deviation.

Thus, LR identifies the weight of each variable and takes into consideration the deviation that minimizes the variability of the final output. In ML context, it repeats the calculation process to modify its parameters in each iteration to improve its reliability. In addition, gradient descent can also be used for determining the optimal learning rate, in order to stop the training of the algorithm and starting processing the output (Kim et al., 2022).

Although, it is important to reiterate that LR makes an assumption of linearity between the independent variables, which sometimes may not be so connected, being a rigid model and lacking more flexibility to adapt to not so related data (Schonlau & Zou, 2020).

3.2.2 Artificial Neural Network (ANN)

Artificial Neural Networks is inspired by the neural and brain structure, which connects the information from a network-like structure, generally built in multiple layers with interconnected nodes. This provides the possibility to correlate varied data, organizing in layers and adding weights to determine the correlation between the input layer and output layer (Osman et al., 2022; Tu, 1996).

Different types of organization and layers can be applied. Typical neural networks use: perceptron layered networks – simplest neural network with an input and output layer; layered networks – multiple layers of interconnected neurons; recurrent networks – neural networks with feedback loops; or gated recurrent unit – procedure with recursive tasks with the output dependent on calculations. Thus, they main difference rely on the complexity degree of the problem, impacting the type of connection between the layers (Kufel et al., 2023).

ANN procedure is based in collecting the input data, creating, and configuring the neural links with weight attribution to each variable, network validation and output analysis. Thus, the operation is based on a series of links to perform a regression task, following the concept of ML. In addition, ANN is often recommended for regression applications and generative modeling and can advance to more complex structures, defined by deep neural networks, although it requires a large database to improve its accuracy and can also be slower to train and expose a result sensitive to the chosen layers and parameters (Singh et al., 2016). In addition, ANN usually depends on a large database for properly operating while DT can be sometimes too rigid, overfitting the models, which could lead to satisfying results on the trained models but to unsatisfying ones in new situations (Barua & Zou, 2022; Tu, 1996). However expansion of the database by the pre-established database is a possibility (Taye, 2023). Figure 3.7 presents ANN structure overview.

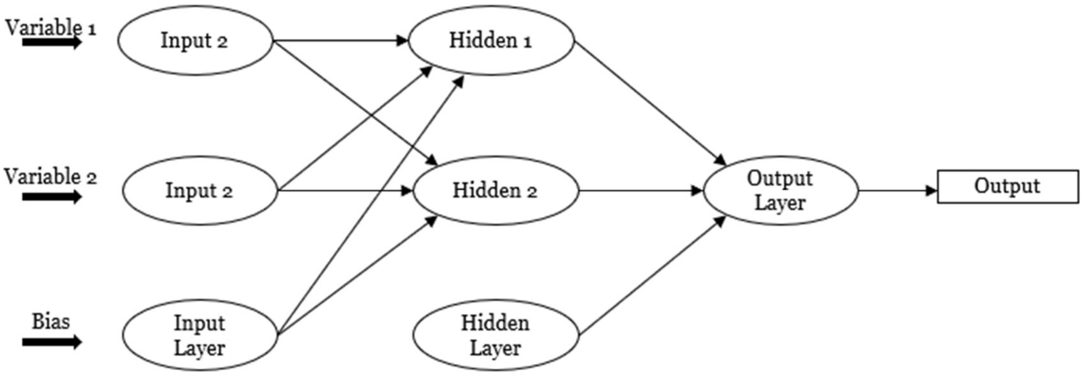


Figure 3.7 – Artificial neural network overview (adapted from Tu, 1996).

3.2.3 Decision Trees (DT)

Decision Trees can be considered easy algorithms to handle, as it makes decisions based on a hierarchical structure between nodes, edges, and leaves, being able to handle a variety of data with a robustness to missing values, providing an overall high generalization ability. Although efficient, main remarks are within the construction of the decision tree itself, where an error can propagate among the different levels of decision, impacting the overall decisions (Kotsiantis, 2007; Singh et al., 2016).

In addition, with a higher number of branches, equally higher is the probability of errors, as the several leaves may be too statistically small for representing a specific class and the nuances between then, creating the tendency to overfit the decisions in a single leaf (Kotsiantis, 2007; Singh et al., 2016).

Generally, DT simplify relationships between the input and target variables, handling missing values. It tends to overfit and underfit, indicating the importance of having a calibrated algorithm. The more complex the algorithm is, the less reliable it tends to be, being necessary to avoid several spreads within the tree. Thus, stopping rules are related to the minimum number of records of a leaf or in the node prior to splitting. When stopping rules do not work well, removing the size of the tree by removing nodes is also indicated by pruning (Song & Lu, 2015). Figure 3.8 presents decision tree structure overview.

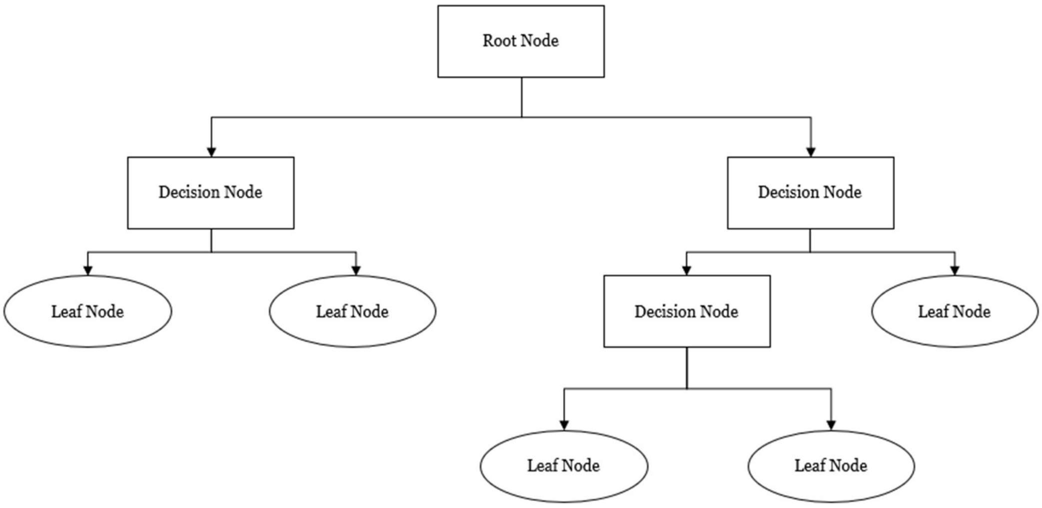


Figure 3.8 – Decision tree structure (adapted from Sarker, 2021).

3.2.4 Random Forest (RF)

Considering the DT tendency to overfit decisions, RF was developed to avoid the overfitting tendency by training several DT and composing an assembly of decisions. Although fast, robust and reliable, it requires a high number of DT for training, slowing down the overall processing time (Singh et al., 2016), although their capacity to overcome other models, such as SVM or LR, has been indicated for PCI calculation (Ali, et al., 2022).

Random forests can be defined as tree predictors where each tree is sampled according to the distribution of all trees in the forest. Hence, an information is converged through a forest of trees analysis, avoiding overfitting, achieving high accuracy for overall prediction analysis (Breiman, 2001).

Schonlau & Zou, 2020 indicates the superiority of RF over LR mechanism, as the latter considers a linearity between the variables, which are not always direct. Nonlinearities can be found among diverse database and several approaches, such as RF. RF can manage large datasets for different applications and contains logistic and regression tasks within its mechanism. In addition, a 50-50% ratio has been indicated for training and validation, although this amount is indicated for sufficiently large database and can considerably change depending on the database size and applicability (Schonlau & Zou, 2020). Figure 3.9 presents random forest structure overview.

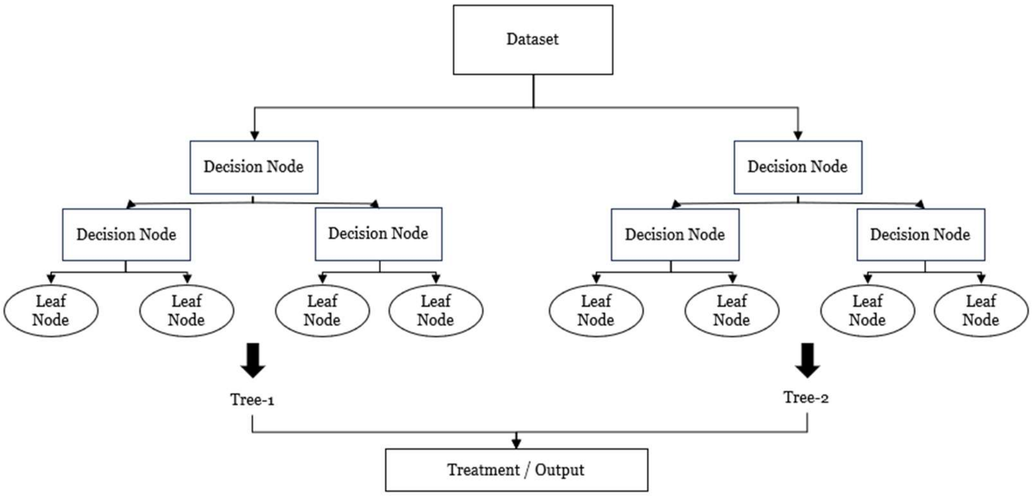


Figure 3.9 – Random Forest structure (adapted from Nabipour et al., 2019; Schonlau & Zou, 2020).

3.2.5 Support Vector Machine (SVM)

Support vector machine is a binary algorithm based on analyzing the input data as vectors, revolving the notion around a "margin", which can be defined as the hyperplane space that separates two different data sets, aiming to maximize the minimum distance between them and, thus, reducing the generalization error and achieving a high accuracy, although it can be equally slow to train and not easy to handle. It is able to handle linear and nonlinear data to find correlation between them, generally associated to bioinformatics and image classification areas (Kotsiantis, 2007; Singh et al., 2016).

3.2.6. Discussion

Considering the presented ML algorithms, their use and application vary depending on the problem complexity and input database, each presenting its advantages and disadvantages. Table 3.1 presents an overview on discussed ML algorithms.

Table 3.1 – Overview on ML algorithms.

ML algorithm	Advantages	Disadvantages	Application	References
LR	Clear (transparent) mechanism, handles numerical data, provides robustness with a linearity assumption.	Needs large database to provide accurate data regression. Limited to numerical data.	Simpler and numerical problems.	(Maroco, 2014; Sarker, 2021)
DT	Handles numerical or categorical data. Handles missing data with good prediction, provides robustness, transparency and uses small processing effort.	Tendency to overfit if a node is considerably higher on statistical value, propagating errors along the tree.	Numerical or categoric problems with a lower level of complexity.	(Kotsiantis, 2007; Singh et al., 2016; Song & Lu, 2015)
RF	Handles numerical or categorical data. Adaptable to regression problems.	Higher processing effort is necessary. Performance depends on trees construction.	Numerical or categoric problems with a certain degree of complexity.	(Sarker, 2021; Singh et al., 2016)
ANN	Handles numerical or categorical data. High accuracy and fast training mechanism with a high learning capacity.	Black-box mechanism, sensitive to missing and irrelevant data. Weights affect considerably.	Numerical or categoric problems with high degree of complexity.	(Kotsiantis, 2007; Tu, 1996)
SVM	Can handle numerical or categorical data. High accuracy and low overfitting problems. High precision for classes separation of data and simple training model.	Complexity based on vectors. Black-box mechanism and performance highly dependent on input parameters.	Numerical or categoric problems with varied degree of complexity.	(Carvalho et al., 2019; Kotsiantis, 2007; Singh et al., 2016)

For airport pavements´ maintenance, such algorithms could predict accurately PCI values that can be used for supporting maintenance planning. Pathologies patterns in pavements can be traced, providing a logic structure that would result in a reliable PCI value. The described algorithms are classified as supervised learning. Supervised models are based on mapping the inputs to provide outputs using a training set for the model calibration and validation, mostly focusing on regression or classification strands (Carvalho et al., 2019).

Other learning strands exist, such as unsupervised, semi-supervised and reinforcement, used to uncover patterns and relationships. These consider more complex input data, attached to other variables, while also requiring a large database, whereas the results are often more open and have a high-complexity nature. Reinforcement learning, for instance, represents a concept of reinforcement learning algorithm or self-learning algorithm (SLA) that involves dynamic programming where it starts mainly from a database obtained by simulated or real situations, receiving input values to predict the next situation for sequential decision-making (Barua & Zou, 2022). Although not the most recommended application for numerical and straight-forward data, reinforcement learning can also be related to supervised learning, generating a complex learning model. Figure 3.10 summarizes the main types of learnings and their algorithms.

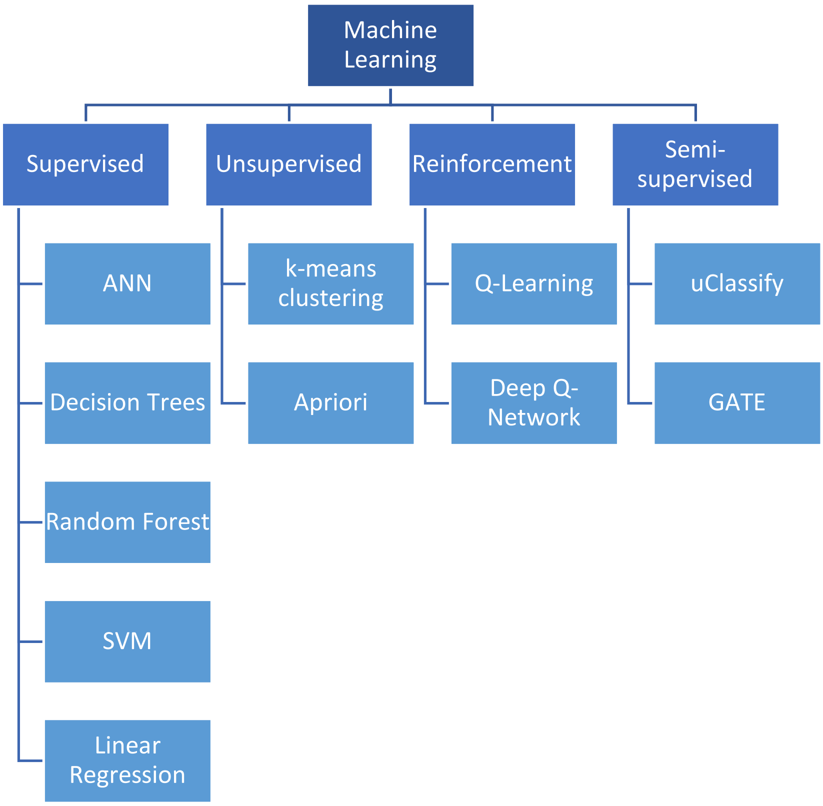


Figure 3.10 - Machine learning and main learning and algorithm types (adapted from Sarker, 2021; Taye, 2023).

Although the algorithms are different from each other, ML models are generally based on the following steps: initialization, variables selection, model training, operation, analysis and delivering. The behavior of a given model may improve as more iterations or calibrations are made to the model. There are several software programs that use ML concepts to generate models, such as WEKA, TensorFlow, PyTorch, Keras, Microsoft Azure or IBM SPSS Statistics, which can operate based on these ML algorithms.

Thus, the choice of the type of ML algorithm is primarily based on identifying the problem to be solved and the approach will act according to the following pillars (Reich, 1997):

1. Understanding the complexity in portraying and transforming input data into the final result.
2. Data preparation, handling missing values or standardizing the data.
3. Software to be chosen and its operation setup.
4. Database volume, with possibility to use self-expanding database algorithms.
5. Selection of model training mechanism – incremental or batch.
6. User interaction and use of the trained model for PCI calculations for new data.

3.3. Applications to pavement condition evaluation

The excessive time involved in calculating the PCI and the associated degree of subjective of the analysis makes this index somewhat complex to draw a pattern. However, considering the ongoing digitalization happening in several fields, ML emerges as an opportunity for automation in PCI calculation, where calculation time would be drastically reduced, transforming an essentially manual process into an automatic one. This strand will be analyzed in this study for airport pavements, where the results of the ML approach contribution will be compared to the conventional method, paving the way for simplifying the calculation of the PCI index, although only feasible if the input database is sufficiently large to provide enough accuracy for the ML models.

More recently, the digitalization of pavement inspection operations has been gaining ground, as less than 2% of the carried inspections in the United States are done manually, consisting in automated data collection vehicles, as it reduces the required time while ensuring the labors´ safety (Kheirati & Golroo, 2022). Thus, using automatized procedures for data collection is becoming present in civil engineering field, although the PCI calculation itself still presents a somewhat manual procedure, as not many studies indicate the investigation of ML for this use. Airport pavement studies tend to evaluate the aspects of the pavement itself, such as distress occurrence or roughness (Li et al., 2023; Noori & Sarkar, 2023), or the serviceability index (Irfan et al., 2015; Parsons & Murrell, 2023), using algorithms or standard procedures. In addition, airport´s pavements maintenance is still mostly based on corrective instead of predictive actions (Li et al., 2023) . Recent researches have also indicated ML feasibility for PCI calculation, contributing for the automation of the calculation process and change from corrective to predictive actions.

Li et al., 2023 used automated process for PCI survey and calculation based on ML, indicating the importance of not considering too many distresses as it could overlap the distresses while overfitting a model. Issa, Samaneh, & Ghanim, 2022 considered ANN as a successful model for PCI calculation, considering its model structure in hierarchically layers, connecting nodes between different types and levels of distresses, using 70% of the data for training, 15% for validating and 15% for testing and 20% for validating the model, providing accurate results. Although 70% of the training set is considerable used, Gholamy, et al., 2018 indicates 80% of the training set and 20% of the validating sets is empirically the best division for statistical problems, providing better accuracy overall.

Ali et al. (2023) also indicated ANN potential for PCI calculation, as it does not require linear correlation between the variables, possibly creating multilinear links between the input and output data, outperforming regression models, describing it as a high accuracy model. Although the efficiency of ANN has been stated, Ali, Esekbi, et al., 2022 tested SVM, LR and RF, indicating their feasibility for PCI calculation. SVM and RF were superior to conventional models, such as LR, although RF was more reliable and accurate than SVM, due to RF operating mechanism, which it is based on training each tree with a random sampling, managing overfitting problems better than ANN. Nabipour et al., 2019 also indicated SVM underperformance compared to other models.

In addition, deep learning can be considered as a further step, as it uses image scanning and learning, providing a fully automated processes where it can provide more consistent analysis. Although, this approach can be considerably complex to implement, as it requires specific methodology for both inspection and data process (Majidifard et al., 2020).

Thus, ongoing research demonstrate the ML algorithms´ efficiency in improving current calculation methods. Even considering that difference between the models is notable, ML techniques seem to be superior to conventional models, although important to reiterate that each case has a different database, different data and collection process and different data processing. This can make direct comparison between different cases and models, while the low number of works on this topic indicate the necessity of further scientific research to create patterns for comparison. In addition, statistical analysis must be conducted between the ML results to establish comparison between the models, identifying advantages and disadvantages of each.

3.4 Statistical analysis and models comparison

Several statistical parameters can aid establish the overall accuracy and efficiency of ML models, such as the correlation coefficient (CC). The correlation coefficient indicates the direct or inverse strength between different variables in a correlation analysis, ranging from -1 to +1, where a negative correlation indicates one variable increase while the other decreases and positive correlation indicates two variables react the same way, increasing or decreasing together. The higher the number, the stronger the impact (Keith, 2005; Schober & Schwarte, 2018). Table 3.2 summarizes correlation coefficient interpretation.

Table 3.2 - Correlation coefficient interpretation (adapted from Schober & Schwarte, 2018).

Correlation coefficient range	Interpretation
±0.00-0.10	Negligible correlation
±0.10-0.39	Weak correlation
±0.40-0.69	Moderate correlation
±0.70-0.89	Strong correlation
±0.90-1.00	Very strong correlation

As the results can be affected by chance and the correlation coefficient has no intrinsic interpretation, evaluating the variables and theory of correlation are necessary. Researches tend to also use the coefficient of determination (R^2), as it is the proportion of variance in one dependent variable, calculated by the square of the correlation coefficient (CC). Thus, the R^2 in percentage indicates that the X% variability can be explained by the relationship of the analyzed variables (Ozer, 1985; Schober & Schwarte, 2018).

In addition, root-mean-square error (RMSE) or mean-absolute error (MAE) are also frequent tools for statistical analysis. MAE value is calculated by the mean square difference of point-to-point of the standard value to the calculated value while RSME consider the square differences of the values, proceeding to the square root of the final value. RSMEs process penalizes higher errors, usually providing values higher than MAE due to the square differences of the values. Both approaches are derived from probabilistic laws and considered as standards, not existing a consensus on which one to use, although both of them are recommended for model evaluation and performance applications (Hodson, 2022).

Additionally, relative absolute error (RAE) parameter can be seen as a variation of MAE, being used for assessing precision accuracy in relation to the database, normalizing the absolute error by the mean of real database values. A lower RAE indicates a smaller absolute error in relation to the database scale. On the other hand, root relative squared error (RRSE) is the square root of mean square of all the errors, considering the real database values, comparing the prediction model to the dispersion of the original values, being indicated for numerical predictions (Christie & Neill, 2022).

Thus, MAE is a simpler absolute error without considering the database scale, while RMSE penalizes higher errors. RAE and RRSE are less used parameters and consider the database scale and variability based on the real data. Such parameters are widely used for ML algorithms evaluation and comparation, being MAE and RMSE standard procedures (Hodson, 2022).

For categorical variables, kappa statistic is usually used to correlate unbalanced classification systems, ranging from -1 to 1, being equivalent to the correlation coefficient, where its square value can be described as the accuracy degree. Although it can be directly interpreted, the context of analysis has to be considered in order to better evaluate the degree of correlation (Mchugh, 2012). Kappa´s value can be interpreted as presented in table 3.3.

Table 3.3 - Kappa statistic value (adapted from Mchugh, 2012).

Kappa statistic 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

3.5 Database imbalance

Statistic data and ML algorithms can be considerably affected by the database size and balance.

Database imbalance occurs when a class contains a considerably different number of cases that another class. In these circumstances, the ML algorithm can overfit the output in the class with the higher amounts, due to the increased probability, misclassing the values, affecting the PCI calculation and the final output. Some possible solutions data-level techniques, algorithm-level methods and hybrid approaches (Johnson & Khoshgoftaar, 2019), mostly translated as random under sampling (RUS) – limiting the size of the overweight class – or random over sampling (ROS) – increasing samples of underweighted classes by self-expanding database or new cases. Although data imbalance can affect output values, Japkowicz, 2000; Johnson & Khoshgoftaar, 2019; Krawczyk, 2016 indicate there is no large impact if the classes disproportions between the groups are well distributed without overlaps, not considerably impacting problem with low-complexity degree.

ROS methodology can impact the training time of complex algorithms, as new cases will be considered, while RUS methodology can reduce the overall limit, due to the database limitation, and increase the overall reliability, as ROS can generate new cases that do not consider the full length of the variables or problem´ complexity (Japkowicz, 2000).

In addition, ROS is known for its efficiency in self-enlarging a database randomly, affecting the training capacity and model calibration of algorithms. Yang, et al. (2024) indicate ROS and RUS methodology do not provide considerable impact on sufficiently large database, even if unbalanced. Conversely, on small datasets, ROS could provide substantial improvements. Hayaty et al. (2021) stated ROS´s successful approach for improving ML algorithms´ accuracy, based on the Synthetic Minority Over Sampling (SMOTE) technique (Chawla, et al., 2002), which replicates the data in the minority class based on the nearest sample data.

On the other hand, RUS methodology can negatively impact the overall classifier, as randomly underusing the available dataset could result in less accurate training models for the involved algorithm due to the non-use of essential data. Mohammed et al (2020) indicates ROS typically overperforms RUS for several algorithms, although it also creates the overfitting tendency as it creates similar examples present in the minority classes.

4 Case Study

4.1 Introduction and methodology

Effective airport pavements management ensures the pavements usability and users´ safety, while being cost-efficient. PCI calculation provides information about the overall state of the pavement, and its determination relay on information regarding the distresses and severity levels present on the pavement surface.

A pavement distress database for PCI calculation serves as a repository to provide information regarding surface conditions and structural integrity. Such information support compliance, reporting, performance monitoring, asset management, decision, and costs saving (Karim et al., 2016; Li et al., 2024).

Concerning the application of ML algorithms, the accuracy and efficiency of produced models depend heavily on the quality and quantity of the input data (database) to correctly train the models and to provide an accurate output value.

The present chapter will analyze supervised learning ML algorithms application for PCI calculation using WEKA software (Frank et al., 2017). LR, SVM, DF, DT and ANN will be applied to a set of real pavement distress data and the resulting models will be analyzed to expose their accuracy for PCI prediction. The algorithms will be run based on a pavement distress and PCI values database corresponding to 261 pavement sample units of 3 international airports runways (2 in Cape Verde and 1 in Peru) (Domingos, 2017; Lima, 2016). These 261 sample units were evaluated for 17 surface pavement distresses, considering three levels of severity (low, medium and high), that were used for the PCI calculation based on ASTM D5340-12 (ASTM International, 2012). Appendix I presents an extract of the database used in WEKA.

Three types of PCI output will be considered in the analyses: numerical PCI (0-100); 3 classes of PCI: good, fair and poor; and seven classes of PCI: good, satisfactory, fair, poor, very poor, serious, failed (see table 4.1).

Table 4.1 - PCI classes (adapted from ASTM International, 2023).

Seven classes		**Three classes**	
PCI values	PCI description	PCI values	PCI description
85.01-100	Good	70.01-100	Good
70.01-85	Satisfactory		
55.01-70	Fair	55.01-70	Fair
40.01-55	Poor	0-55	Poor
25.01-40	Very poor		
10.01-25	Serious		
0-10	Failed		

Results will be compared to assess the ML algorithms´ s capability to correctly predict PCI based on three test options: training set, cross validation and percentage split (80% for training and 20% for testing), totalizing 39 tests, as considered in table 4.2. Statistical analysis will be followed to evaluate the precision of the obtained models, presenting the final conclusions.

Table 4.2 - ML testing procedure (Part 1/2).

ML algorithm	**Test options**	**Output value**	**Algorithm Type (WEKA)**
Linear Regression (LR)	Training Set	Numerical PCI	Functions (Linear regression)
	Cross Validation (10)	Numerical PCI	Functions (Linear regression)
	80-20 training/testing set	Numerical PCI	Functions (Linear regression)
Decision Tree (DT)	Training Set	Numerical PCI	Trees (M5P)
		3 classes of PCI	Trees (J48)
		7 classes of PCI	Trees (J48)
	Cross Validation (10)	Numerical PCI	Trees (M5P)
		3 classes of PCI	Trees (J48)
		7 classes of PCI	Trees (J48)
	80-20 training/testing set	Numerical PCI	Trees (M5P)
		3 classes of PCI	Trees (J48)
		7 classes of PCI	Trees (J48)
Random Forest (RF)	Training Set	Numerical PCI	Trees (Random Forest)
		3 classes of PCI	Trees (Random Forest)
		7 classes of PCI	Trees (Random Forest)
	Cross Validation (10)	Numerical PCI	Trees (Random Forest)
		3 classes of PCI	Trees (Random Forest)
		7 classes of PCI	Trees (Random Forest)
	80-20 training/testing set	Numerical PCI	Trees (Random Forest)
		3 classes of PCI	Trees (Random Forest)
		7 classes of PCI	Trees (Random Forest)

Table 4.3 - ML testing procedure (Part 2/2).

ML algorithm	Test options	Output value	Algorithm Type (WEKA)
Artificial Neural Network (ANN)	Training Set	Numerical PCI	Functions (Multilayer Perception)
		3 classes of PCI	Functions (Multilayer Perception)
		7 classes of PCI	Functions (Multilayer Perception)
	Cross Validation (10)	Numerical PCI	Functions (Multilayer Perception)
		3 classes of PCI	Functions (Multilayer Perception)
		7 classes of PCI	Functions (Multilayer Perception)
	80-20 training/testing set	Numerical PCI	Functions (Multilayer Perception)
		3 classes of PCI	Functions (Multilayer Perception)
		7 classes of PCI	Functions (Multilayer Perception)
Support Vector Machine (SVM)	Training Set	Numerical PCI	Functions (SMOreg)
		3 classes of PCI	Functions (SMO)
		7 classes of PCI	Functions (SMO)
	Cross Validation (10)	Numerical PCI	Functions (SMOreg)
		3 classes of PCI	Functions (SMO)
		7 classes of PCI	Functions (SMO)
	80-20 training/testing set	Numerical PCI	Functions (SMOreg)
		3 classes of PCI	Functions (SMO)
		7 classes of PCI	Functions (SMO)

The dataset used in this study has been divided into 3 and 7 PCI classes, exposing the unbalance for both situations. Tables 4.3 and 4.4 show the PCI distribution by classes.

Table 4.4 - PCI 3 classes dataset distribution.

3-Classes sampling		
Poor	Fair	Good
105	22	134

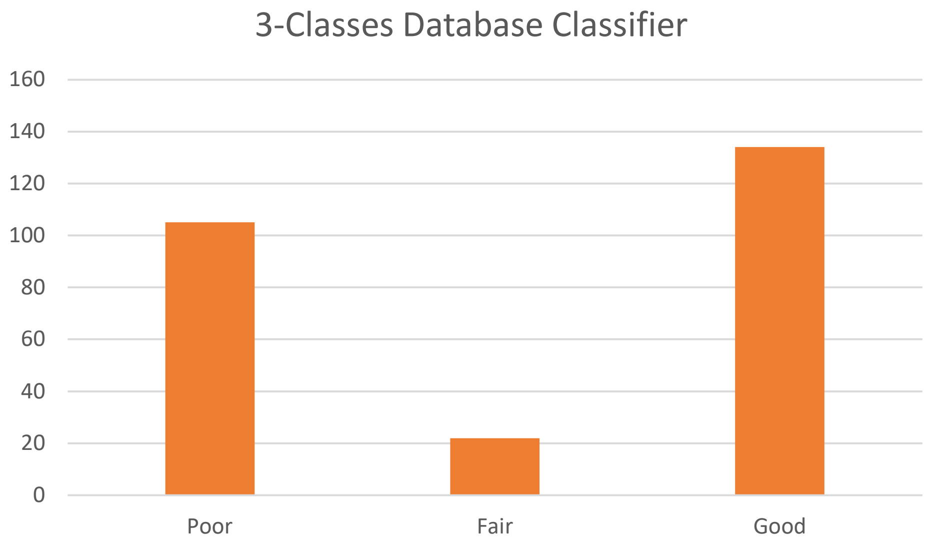
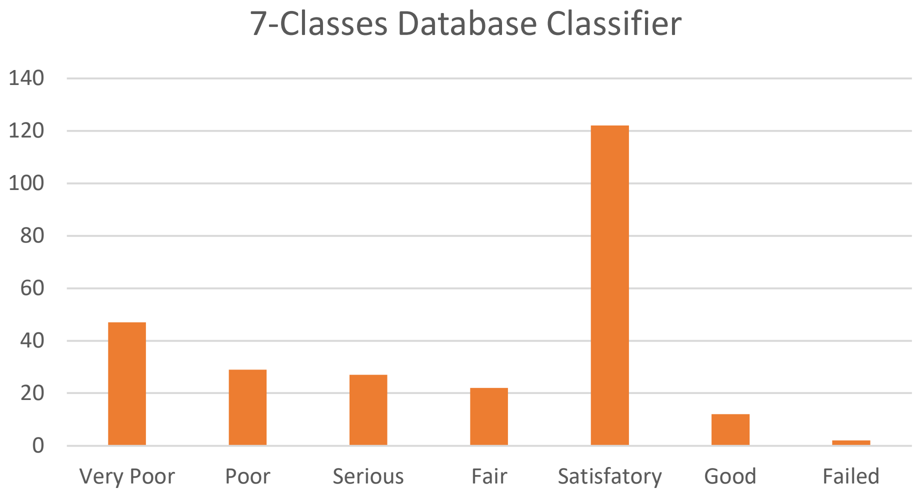


Table 4.5 - 7 Classes dataset balance.

7-Classes sampling						
Very poor	Poor	Serious	Fair	Satisfactory	Good	Failed
47	29	27	22	122	12	2



WEKA software (Frank et al., 2017) is imbued with a ROS tool, which can be used for self-expanding the database, using the database cases to randomly generate new ones. Considering the analyzed database imbalance, ROS and RUS tools will be considered for the cases with lowest accuracy and lower efficiency overall, to provide a comparison between imbalanced and modified databases (for variable balance), via ROS or RUS. Figure 4.1 shows the adopted methodology.

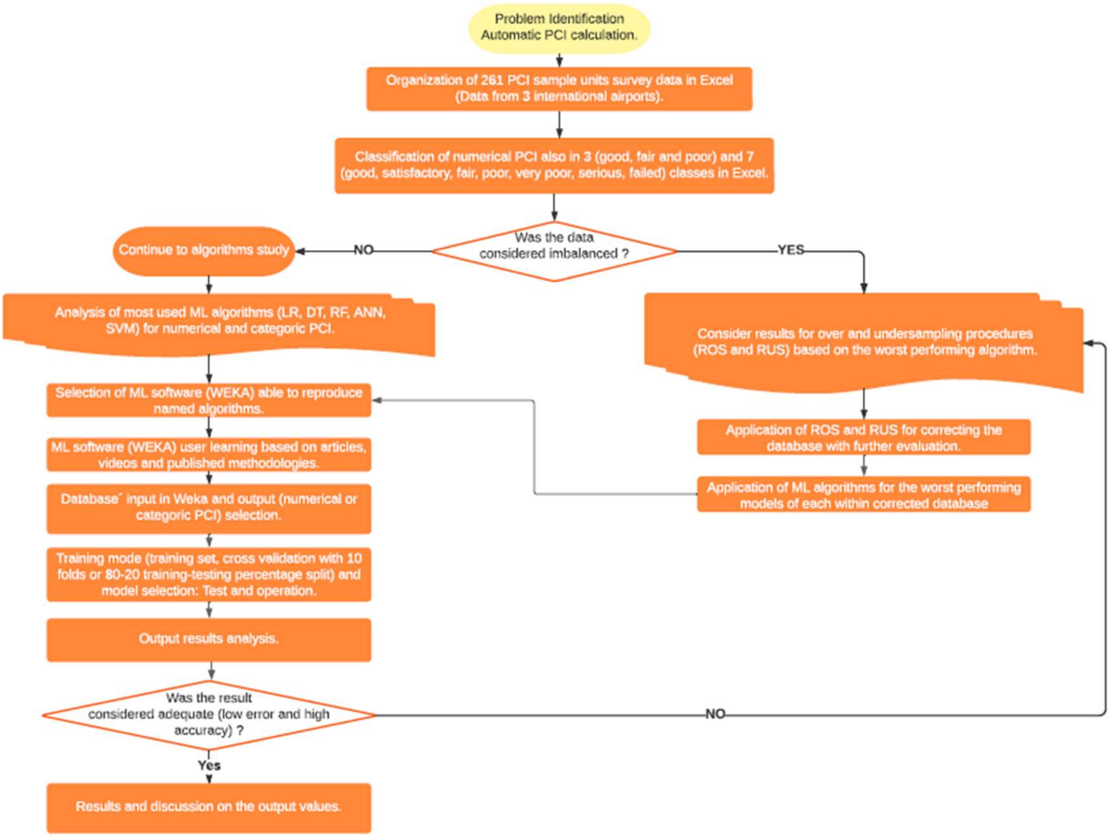


Figure 4.1 - Adopted methodology.

4.2 Algorithm tests

4.2.1 Linear Regression

Linear regression tests were done for numerical PCI output considering three test options: training set, cross validation with 10 folds and 80-20 training/testing percentage split. Using the training set is seen as a high-fidelity test option, although an overfit tendency appears (machine learning algorithm tends to be accurate within its database), not performing well for new cases (out of the database). Table 4.5 presents the results obtained for LR algorithm, exposing mean actual PCI (aPCI) and the mean predicted PCI (pPCI) values.

Table 4.6 - Linear Regression ML test results

Linear Regression: numerical PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
CC	0.91	0.69	0.88
R^2	0.82	0.48	0.77
MAE	7.21	9.80	8.41
RMSE	9.61	19.68	11.48
RAE (%)	34.85	47.19	39.83
RRSE (%)	42.03	85.75	49.74
aPCI	58.58	58.58	58.77
pPCI	58.58	59.35	59.67
Error (= aPCI-pPCI)	0.00	-0.76	-0.90

The 80-20 split seems to be the most adequate test option, as a strong correlation was found between the variables. Results indicate a relatively small MAE, which can be translated as the average size of found mistakes within a collection of predictions.

Although PCI point by point error ranged from +27 to -32, the mean PCI was highly accurate, with an error of -0.90, indicating the feasibility for using 80-20 training-testing percentage split for PCI calculation with LR.

4.2.2 Decision Tree

Decision trees tests were done using M5P trees classifier developed by Quinlan, 1992. The tests were conducted for numerical PCI (0-100), and for three and seven classes of PCI considering three test options: use training set, cross validation with 10 folds and 80-20 training-testing percentage split. Table 4.6 presents the results obtained for DT M5P algorithm, exposing mean actual PCI (aPCI) and mean predicted PCI (pPCI) values for numerical PCI.

Table 4.7 - Decision Trees tests results for numerical PCI (0-100).

Decision Tree: numerical PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
CC	0.93	0.91	0.87
R^2	0.86	0.82	0.75
MAE	5.55	6.22	7.66
RMSE	8.55	9.22	11.36
RAE (%)	26.82	29.97	36.29
RRSE (%)	37.38	40.16	49.21
aPCI	58.58	58.58	58.77
pPCI	58.54	58.83	58.65
Error (= aPCI-pPCI)	0.04	-0.24	0.12

The cross validation with 10 folds seems to be the most adequate test option, as a strong correlation was found between the variables and a small MAE was found. Although PCI point by point error ranged from +28 to -35, the mean PCI was highly accurate, with an error of -0.24, indicating the feasibility for using cross validation with 10 folds for numerical PCI calculation with DT.

Models were also tested with C4.5 model tree algorithm (Salzberg, 1994) for three and seven classes of PCI, as presented in table 4.7.

Table 4.8 - Decision Trees test results for 3 classes of PCI (good, fair and poor).

Decision Trees: 3 classes of PCI (n=261)			
Parameters	**Use training Set**	**Cross validation**	**80-20 split**
Kappa Statistic	0.89	0.80	0.76
MAE	0.06	0.09	0.12
RMSE	0.18	0.24	0.28
RAE (%)	17.47	24.65	32.34
RRSE (%)	41.84	57.11	65.86
Error (%) (Incorrectly Classified)	5.74	10.72	13.46

Similar behavior to numerical PCI occurs, where the use of the training set tends to overfit. Cross validation with 10 folds indicated a strong kappa statistic and relatively small error, correctly assessing 89% of the cases. The error size can be related to the unbalanced database, as previously mentioned. A higher database or balanced dataset could mitigate the error. Section 5.2.7 will expose the conducted tests applying ROS and RUS methods.

The results for 7 classes of PCI are presented in table 4.8.

Table 4.9 - Decision Trees tests results for 7 classes of PCI (good, satisfactory, fair, poor, very poor, serious, failed).

Decision Trees: 7 classes of PCI (n=261)			
Parameters	**Use training Set**	**Cross validation**	**80-20**
Kappa Statistic	0.81	0.56	0.53
MAE	0.05	0.09	0.10
RMSE	0.16	0.26	0.28
RAE (%)	25.83	46.78	50.13
RRSE (%)	50.95	82.53	89.58
Error (%) (Incorrectly Classified)	13.00	30.65	34.61

Found results expose that 80-20 training-testing percentage split and cross validation with 10 folds for 7 classes of PCI present a high amount of error and a weak correlation according to Kappa statistics. Although the use of the training set presents good results, the tendency of overfitting excludes it as a reliable option. Errors cause could be due to the unbalanced characteristic of the database, as previously stated, segregating smaller groups, not providing a sufficient amount for proper model training.

For airport pavement management, the division of PCI into 3 classes can indicate whether the overall pavement state is adequate or not for functioning, possibly attending the overall necessities. Nevertheless, self-expanding database is analyzed in section 4.2.7, providing results´ comparison.

4.2.3 Random Forest

Random forests tests were conducted using the algorithm proposed by Breiman (2001). RF ensure randomness for accurate classifiers and regressors, avoiding overfitting problems. The tests were performed for three and seven classes of PCI considering three test options: use training set, cross validation with 10 folds and 80-20 training-testing percentage split. Table 4.9 presents the obtained results, exposing mean actual PCI (aPCI) and mean predicted PCI (pPCI) values for numerical PCI.

Table 4.10 - Random Forest test results for numerical PCI.

Random Forest: numerical PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
CC	0.99	0.93	0.92
R^2	0.98	0.86	0.84
MAE	2.31	5.81	6.16
RMSE	3.47	8.73	9.15
RAE (%)	11.15	27.97	29.19
RRSE (%)	15.17	38.02	39.65
aPCI	58.58	58.58	58.77
pPCI	59.06	60.10	59.49
Error (= aPCI-pPCI)	-0.47	-1.51	-0.81

The results obtained indicate that all test options are reliable with a strong correlation index. Cross validation with 10 folds is considered the better option as it is reliable also for new cases, contrary to the training set that tends to overfit.

PCI tests considering 3 and 7 categorical classes were also performed based on the model proposed by Breiman (2001). Table 4.10 and 4.11 present the test results.

Table 4.11 - Random Forest test results for 3 classes of PCI (good, fair and poor).

Random Forest: 3 classes of PCI (n=261)			
Parameters	**Use training Set**	**Cross validation**	**80-20 split**
Kappa Statistic	1.00	0.88	0.86
MAE	0.03	0.10	0.10
RMSE	0.07	0.21	0.20
RAE (%)	9.48	25.58	26.09
RRSE (%)	18.31	48.15	44.70
Error (%) (Incorrectly Classified)	0.00	6.51	7.69

Similar behavior to numerical PCI occurs, where the use of the training set tends to overfit. Both cross validation with 10 folds and 80-20 training-testing percentage split presents a strong kappa statistic associated to relatively small error, correctly assessing over 90% of the cases. The two test options are seen as reliable and replicable to new cases.

Table 4.12 - Random Forest test results for 7 classes of PCI (good, satisfactory, fair, poor, very poor, serious, failed).

Random Forest: 7 classes of PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
Kappa Statistic	1	0.58	0.55
MAE	0.04	0.10	0.11
RMSE	0.09	0.23	0.24
RAE (%)	20.00	51.34	55.92
RRSE (%)	28.23	72.83	74.44
Error (%) (Incorrectly Classified)	0.00	29.11	30.76

Considering 7 classes for PCI, results indicate a high error for both cross validation and 80-20 training-testing percentage split, not providing reliable results, probably due to the size of the database and number of classes considered for PCI. When using the training set, a perfect result (0% error) is obtained, indicating that the larger training set is, a higher reliability it will achieve.

4.2.4 Artificial Neural Network

ANN tests were based on algorithm proposed by Frank et al., (2017), being defined as a classifier that uses backpropagation to learn a multi-layer entanglement, creating nodes and connection between the variables with weights. Hidden layers, momentum and weights were set by the algorithm in "auto build" mode.

ANN tests are slower compared to other algorithms, as the built structure and entanglement between the variables are more complex, taking time to construct the neural network and correctly assess their connection. Table 4.12 presents the tests results for ANN, exposing mean actual PCI (aPCI) and mean predicted PCI (pPCI) values for numerical PCI.

Table 4.13 - Artificial Neural Network tests results for numerical PCI (0-100).

ANN: numerical PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
CC	0.94	0.81	0.88
R^2	0.88	0.65	0.77
MAE	6.02	9.04	7.43
RMSE	9.22	15.91	11.39
RAE (%)	29.10	43.54	35.18
RRSE (%)	40.34	69.31	49.35
aPCI	58.58	58.58	58.77
pPCI	54.40	59.15	55.87
Error (= aPCI-pPCI)	4.18	-0.56	2.90

Results indicate cross validation would be the less reliable from the tested options. 80-20 training-testing percentage split is considered the better option due to its reliability and strong correlation index. In addition, although the training´s set tendency to overfit, it was seen a slight decrease in term of accurate output, probably due to the entanglement between the layers and variables inherent to ANN´s structure. Tests for 3 and 7 classes of PCI were also conducted based on the model proposed by Frank et al., (2017). Tables 4.13 and 4.14 present the obtained results.

Table 4.14 - Artificial Neural Network test results for 3 classes of PCI (good, fair and poor).

ANN: 3 classes of PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
Kappa Statistic	0.92	0.82	0.79
MAE	0.03	0.07	0.08
RMSE	0.15	0.23	0.23
RAE (%)	9.49	19.56	22.45
RRSE (%)	34.22	54.00	55.41
Error (%) (Incorrectly Classified)	4.21	9.96	11.53

Similar behavior to numerical PCI occurs, exposing a decrease in accuracy on the use of the training set option. Cross validation with 10 folds appears as the better option, providing a strong kappa statistic value, while 80-20 training-testing percentage split performed slightly worse.

For PCI divided in 7 classes, results indicate a high error for all test options, probably due to the size and division of the database. The number of cases per class are believed to be too low to provide accurate classification considering 7 groups.

Table 4.15 - Artificial Neural Network test results for 7 classes of PCI (good, satisfactory, fair, poor, very poor, serious, failed).

ANN: 7 classes of PCI (n=261)			
Parameters	**Use training Set**	**Cross validation**	**80-20 split**
Kappa Statistic	0.80	0.54	0.52
MAE	0.05	0.10	0.10
RMSE	0.17	0.26	0.26
RAE (%)	27.47	49.09	51.58
RRSE (%)	53.76	80.52	82.88
Error (%) (Incorrectly Classified)	14.17	32.56	34.61

4.2.5 Support Vector Machine

SVM tests were conducted using the algorithm proposed by Frank et al., (2017); Shevade, Keerthi, Bhattacharyya, & Murthy, (2000); Smola & Schölkopf, (2004), being defined as a capable tool for pattern recognition and regression problems, providing excellent generalization performance. Numerical PCI was calculated for each test option and results are presented in table 4.15.

Table 4.16 - SVM for numerical PCI (author´s bibliography).

SVM: numerical PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
CC	0.90	0.69	0.88
R^2	0.81	0.47	0.77
MAE	6.48	9.31	8.26
RMSE	10.02	20.20	11.38
RAE (%)	31.30	44.82	39.13
RRSE (%)	43.81	87.99	49.31
aPCI	58.58	58.58	58.77
pPCI	58.63	59.34	59.50
Error (= aPCI-pPCI)	-0.04	-0.75	-0.72

The results obtained indicate the feasibility of 80-20 training-testing percentage split, as it achieved a high correlation index close to the one obtained for the use of the training set. Contrary, the cross validation has not performed well. As previously indicated, although SVM´s considerable performance in 80-20 training-testing percentage split , DT and RF presents a better result for PCI calculation, corroborating the findings of Ali, Esekbi, et al., (2022).

PCI tests for 3 and 7 classes of PCI were also conducted, being the results presented in tables 4.16 and 4.17, respectively.

Table 4.17 - Support Vector Machine test results for 3 classes of PCI (good, fair and poor).

SVM: 3 classes of PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
Kappa Statistic	0.85	0.80	0.72
MAE	0.25	0.26	0.26
RMSE	0.31	0.33	0.34
RAE (%)	65.13	67.58	70.67
RRSE (%)	72.26	75.60	79.80
Error (%) (Incorrectly Classified)	8.43	11.11	15.38

Table 4.18 – Support Vector Machine for 7 classes of PCI (good, satisfactory, fair, poor, very poor, serious, failed).

SVM: 7 classes of PCI (n=261)			
Parameters	**Use training set**	**Cross validation**	**80-20 split**
Kappa Statistic	0.63	0.51	0.50
MAE	0.21	0.21	0.21
RMSE	0.31	0.32	0.32
RAE (%)	101.96	103.70	104.74
RRSE (%)	97.04	98.85	100.10
Error (%) (Incorrectly Classified)	25.67	33.33	34.61

The results found indicate that for 3 classes of PCI, the cross-validation test option is adequate presenting an acceptable error of 11%. Most probably this error can be lowered with a larger database to intensify the algorithm´s training phase and thus, its accuracy. Considering the PCI divided in 7 classes, the model performed worst as a high error is present in all test options. In addition, RAE and RRSE values were over 100%, indicating the models are performing worse than a simple prevision, while also exposing a variability higher than the real cases.

4.2.6 Overall analysis of results

The conducted tests provide an overview on how the algorithms behave for PCI calculation in the three considered options of PCI independent variable: numerical, 3 and 7 PCI classes. Table 4.18 presents the test options with better performance – with higher CC and lower error – for each algorithm to further discuss the results and algorithm´s application. The use of training test option was not considered, as it tends to overfit, obtaining high accuracy levels which can may not be true for new cases scenario, focusing mainly on cross validation with 10 folds (CV10) and 80-20 training-testing percentage split (80-20).

Table 4.19 – Models with best performance for each algorithm considering numerical PCI.

Numerical PCI (0-100)					
Parameters	**Algorithms**				
	LR	**DT**	**RF**	**ANN**	**SVM**
Test option	80-20	CV10	CV10	80-20	80-20
CC	0.88	0.91	0.93	0.88	0.88
R^2	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	59	58.58	58.58	58.77	58.77
pPCI	60	58.83	60.10	55.87	59.50
Error (= aPCI-pPCI)	-1.00	-0.24	-1.51	2.90	-0.72

Conducted tests have shown RF and DT as the most accurate algorithms for numerical PCI calculation, corroborating the results obtained by Ali, et al. (2022) and Nabipour et al. (2019), followed by ANN, SVM and LR. Such results indicate RF and DT´s feasibility in training the trees.

Regarding the test options, cross validation with 10 folds is considered the most accurate test option for DT and RF. Thus, the average of evaluation metrics within 10 folds can increase the reliability on the dataset while reducing the bias, possibly overcoming problems such as small and unbalance database and tendency to overfit, providing stability to the training model of the algorithm. In addition, ANN and SVM´s better result with 80-20 training-testing percentage split indicates the training data offers enough randomness that allows to obtain a solid model, stablishing correct connections between the variables and providing an accurate output value.

Considering the classification of the airport pavement, using PCI as a categorical variable can also provide a faster analysis on the overall pavement´s state. Table 4.19 and 4.20 present the models with best performance for each algorithm considering PCI divided in 3 and 7 classes.

Table 4.20 - Models with best performance for each algorithm considering 3 classes of PCI.

3 classes of PCI (good, fair and poor)					
Parameters	Algorithms				
	LR	DT	RF	ANN	SVM
Test option	-	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

Table 4.21 - Models with best performance for each algorithm considering 7 classes of PCI.

7 classes of PCI (good, satisfactory, fair, poor, very poor, serious, failed)					
Parameters	Algorithms				
	LR	DT	RF	ANN	SVM
Test option	-	Cross10	Cross10	Cross10	Cross10
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

Results indicate RF´s slight superiority to DT for categorical classification of PCI in 7 classes. The increase can be justified by RF´s training model based on individual trees (DT) to comply a forest of cross validation data, providing a sufficiently large statical analysis through individual trees´ random sampling. Thus, RF with cross validation was considered as the most efficient and accurate combination of ML model and test option for PCI calculation, with weak to moderate agreement with the database.

Also, ANN and SVM cross validation option results overperformed 80-20 training-testing percentage split outcome. Although the results were similar, it indicates the 80-20 training-testing percentage split floatability which can be sensitive to the randomness of data selection for the training model, while cross validation provides a more robust and stable overall model. In addition, the error parameter, although low, indicates the necessity of human intervention for confirming the output values before fully entrusting ML models to provide the PCI output. Equally important is to reiterate that no model was accurate enough to correctly classify PCI when considering 7 classes, probably due to a considerable low sampling for each class, deeply affecting the training model and output values for all classes.

For PCI divided in 7 classes, ROS and RUS methodology was be applied to verify the efficiency of such methods to increase the accuracy of the models.

4.2.7 Model results for over and under sampling databases

The used database, as indicated in tables 4.3 and 4.4, is considered unbalanced and results have shown that the consideration of 7 classes categoric PCI, as presented in ASTM 5340-23, did not perform accurately, as results differ largely, not providing an efficient predictive model.
Both ROS and RUS methodologies were tested, based on the SMOTE technique (Chawla et al., 2002) and random under sampling, respectively. Treated databases with ROS and RUS tools were represented in table 4.21 and figures 4.2 and 4.3 below.

Table 4.22 - Comparison between original and treated databases (over and under sampling).

Classes	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

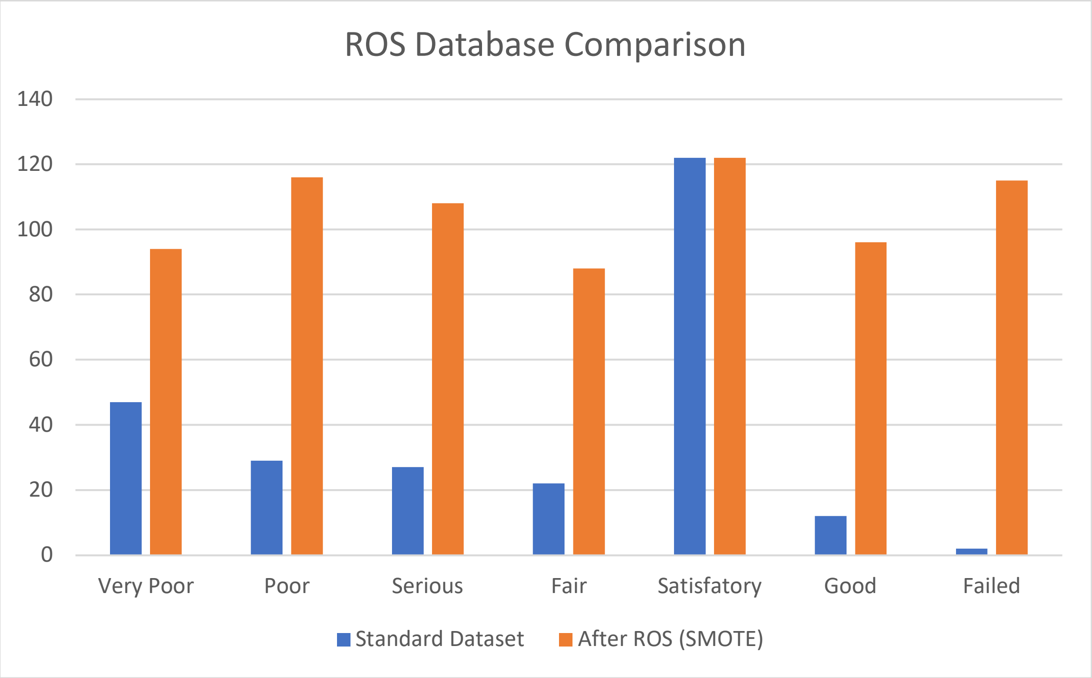


Figure 4.2 - 7 classes of PCI variable before (left) and after (right) ROS application.

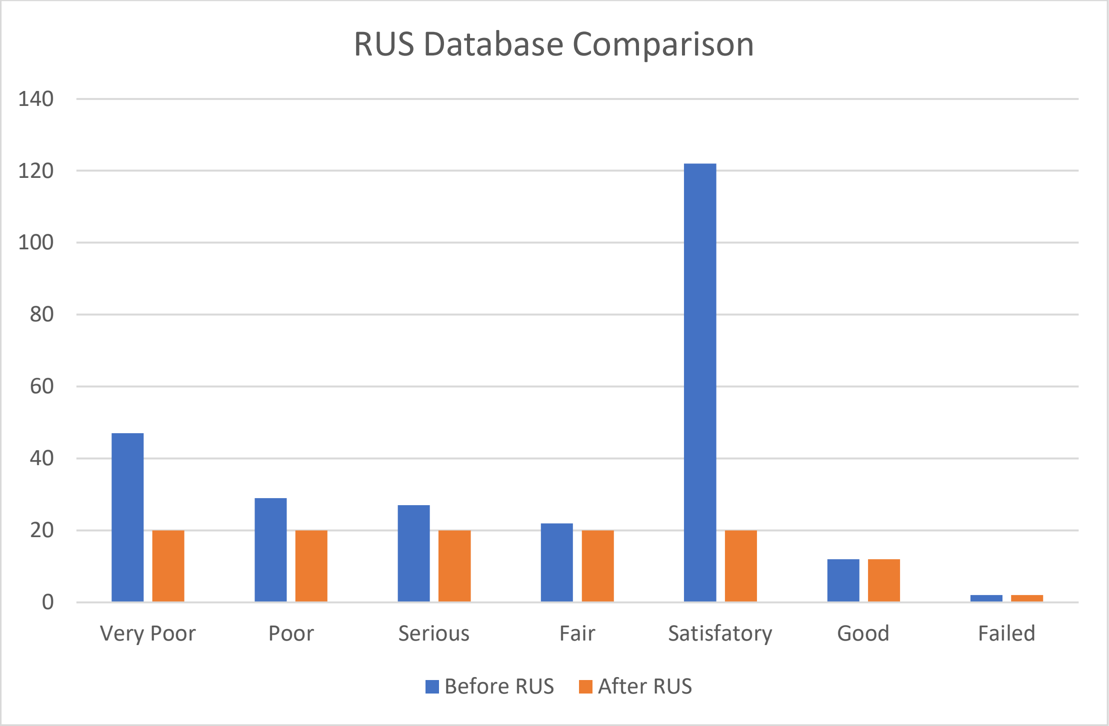


Figure 4.3 - 7 classes of PCI variable before (left) and after (right) RUS application.

The impacts of ROS and RUS method application in 8 built models obtained for 7 classes of PCI using cross validation with 10 folds are presented in table 5.25.

Table 4.23 – Model results for 7 classes of PCI considering original, RUS and ROS databases and cross validation option.

Parameters	DT			RF			ANN			SVM		
	Original database	RUS	ROS	Original database	RUS	ROS	Original database	RUS	ROS	Original database	RUS	ROS
Test option	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10	Cross10
Kappa Statistic	0.56	0.46	0.80	0.58	0.55	0.91	0.54	0.48	0.83	0.51	0.39	0.71
MAE	0.09	0.13	0.05	0.10	0.15	0.05	0.10	0.14	0.05	0.21	0.22	0.21
RMSE	0.2642	0.32	0.22	0.23	0.27	0.14	0.26	0.32	0.17	0.32	0.32	0.31
RAE (%)	46.78	56.86	19.61	51.34	62.23	22.55	49.09	57.16	20.72	103.70	90.79	85.61
RRSE (%)	82.53	94.03	62.63	72.83	78.60	39.64	80.52	92.68	50.29	98.85	93.13	88.52
Error (%) (Incorrectly Classified)	30.65	44.73	16.78	29.11	36.84	7.17	32.56	42.98	14.88	33.33	50.00	24.35

The results obtained support the feasibility of ROS methodology, considerably improving all algorithms, exposing the algorithms´ sensitivity to database size. RUS, in the other hand, has not shown an accurate improvement, most probably due to too small database, indicating that RUS should be essentially applicable when the minority class is sufficiently large to justify under sampling of the other classes, presenting overall a balanced and sufficiently large database. Both ROS and RUS methodologies have shown to be considerably fast for the used database, concluding all training and tests in a couple of seconds. Such results expose the potential of ML´ algorithms for optimizing the standard PCI calculation process.

In addition, RF with cross validation proved to be the best algorithm for PCI prediction, providing a strong kappa statistic and relatively small error. With ROS methodology, the PCI models considering 7 classes became feasible. As for the other algorithms, ROS has also considerably improved them, although to levels that imply a considerable error, indicating their use with attention, as a more complex analysis would need to be done to trust solely on the algorithm.

5 Conclusions and future work

The traditional methods for pavement condition index determination can be time-consuming and inefficient. The 4.0 industry revolution has provided tools that can optimize current standard process. Through the applications of machine learning algorithms, significant improvements in PCI calculation speed and accuracy can be achieved for both numerical and categorical PCI variable.

Airports pavement management teams could highly benefit from this approach, as it could lead to enhanced decision-making regarding the intervention and maintenance timing of the airports' pavements, contributing to safer and more cost-effective airports operation as PCI calculations could be done more frequently in a faster, cheaper and efficient way.

However, the approach of ML for airports pavements´ maintenance should be adopted with caution, as overfitting and correlation issues can arise, requiring robust validation procedures and performance monitoring by critical human analysis.

Among the 39 built models – 3 linear regression, 9 decision tree, 9 random forest, 9 artificial neural network and 9 support vector machine – for the 3 types of outputs – numerical PCI, 3 and 7 classes of PCI – and 3 types of test options – use training set, cross validation in 10 folds and 80-20 percentage split – the following sequence was considered from worst to best performing algorithm: linear regression, support vector machine, artificial neural network, decision tree and random forest, based on the correlation coefficient, kappa statistic and associated error values.

In addition, use training set test option was considered not adequate as it showed a tendency to overfit within all tested models. 80-20 training-test percentage split showed a lower performance, probably due to the small database, and cross validation in 10 folds was considered the better option as it was able to correctly train the model, achieving a high accuracy for the calculated PCIs.

Thus, Random Forest with cross validation data was considered the most efficient and reliable algorithm for all outputs, as it provided a strong correlation coefficient (0.93) for numerical PCI with a MAE value of 5.81, and kappa statistic 0.88 with a MAE value of 0.10 for 3 classes PCI, indicating the model accuracy and overall efficiency. In addition, for 7 classes of PCI, 8 built models – 4 ROS and 4 RUS - for database balance were done, where ROS methodology indicated the enhancement of all algorithms´ performance, highlighting random forest´s kappa statistic enhancement from 0.58 to 0.91 – increase of 56.89% - with a low error value. On the other hand, RUS methodology exposed it is not recommended for small databases, even if providing a balanced database, as it does not provide enough sample units for accurately training the ML model, worsening the original result.

Algorithms´ sensitivity to the database size indicate the need of promoting a universal database, providing open access reliable data regarding PCI calculation and distresses evaluation of airports´ pavements, enabling more accurate and efficient models to be developed.

Therefore, a balanced approach that combines the strengths of ML algorithms with human expertise is essential, rigorously validating the data to ensure the reliability of the chosen model and accuracy of the PCI calculations. In addition, the integration of ML techniques for airport pavements' maintenance represents a strategic opportunity to optimize the overall pavement evaluation and performance process, extending their lifespan and ensuring the users´ safety.

In conclusion, the synergy between ML-driven PCI and maintenance plans could considerably contribute for and efficient airport pavement management, lowering the costs and enhancing their overall performance.

The main limitations of the work can be considered the database size and the large number of independent variables resulting from the consideration of 3 levels of severity for each pavement distress (51 independent variables), often without sufficient information to make them representative, which can affect the overall performance of tested models.

For future work, the analysis of confusion matrixes and ROC curves can provide further knowledge on the database impact in modeling process and on the overall algorithm´ accuracy. Also, as the analyzed database was considered as unbalanced, the possibility of not considering all the pavement distresses or levels of severity of the ASTM D5340-23 must be analyzed.

6 References

1. Ali, A., Esekbi, M., & Sreh, M. (2022). Predicting Pavement Condition Index Using Machine Learning Algorithms and Conventional Techniques. *Journal of Pure & Applied Sciences*, *21*(4), 304–309. https://doi.org/10.51984/jopas.v21i4.2267
2. Ali, A., Milad, A., Hussein, A., Md Yusoff, N., & Heneash, U. (2023). Predicting pavement condition index based on the utilization of machine learning techniques: A case study. *Journal of Road Engineering*, *3*(3), 266–278. https://doi.org/10.1016/j.jreng.2023.04.002
3. ASTM International. (2012). *Standard test method for airport pavement condition index surveys (D5340).*
4. ASTM International. (2023). *Standard test method for airport pavement condition index surveys (D5340).*
5. Augeri, M. G., Greco, S., & Nicolosi, V. (2019). Planning urban pavement maintenance by a new interactive multiobjective optimization approach. *European Transport Research Review*, *11*(1). https://doi.org/10.1186/s12544-019-0353-9
6. Barua, L., & Zou, B. (2022). Planning maintenance and rehabilitation activities for airport pavements: A combined supervised machine learning and reinforcement learning approach. *International Journal of Transportation Science and Technology*, *11*(2), 423–435. https://doi.org/10.1016/j.ijtst.2021.05.006
7. Basu, D., Misra, A., & Puppala, A. J. (2015). Sustainability and geotechnical engineering: perspectives and review. *Canadian Geotechnical Journal*, *52*(1), 96–113. https://doi.org/10.1139/cgj-2013-0120
8. Bellows, B., Bhandari, A., Ibrahim, M., & Sandhu, J. S. (2011). Peering into the Black Box. *Information Communication Technologies*, (January), 38–43. https://doi.org/10.4018/9781599049496.ch067
9. Breiman, L. (2001). Random Forests. *Machine Learning*, *45*, 5–32. https://doi.org/10.1023/A:1010933404324
10. Carvalho, T. P., Soares, F. A. A. M. N., Vita, R., Francisco, R. da P., Basto, J. P., & Alcalá, S. G. S. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, *137*(April), 106024. https://doi.org/10.1016/j.cie.2019.106024
11. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*(Sept. 28), 321–357. https://doi.org/10.1613/jair.953
12. Christie, D., & Neill, S. (2022). Measuring and Observing the Ocean Renewable Energy Resource. In *Comprehensive Renewable Energy* (pp. 149–175). Elsevier. https://doi.org/10.1016/B978-0-12-819727-1.00083-2
13. Domingos, A. (2017). *Determinação e Interpretação do Índice PCI de Pavimentos Aeroportuários*. Master´s dissertation. Retrieved from http://hdl.handle.net/10400.6/7893

14. Doshi-Velez, F., & Kim, B. (2017). *Towards A Rigorous Science of Interpretable Machine Learning*. (Ml), 1–13. Retrieved from http://arxiv.org/abs/1702.08608
15. Federal Aviation Administration. (2021). Advisory Circular No. 150/5320-6G, Airport Pavement Design and Evaluation. *US Department of Transportation Federal Aviation Administration*, 1–195.
16. Feitosa, I. T. (2020). *Validação de um método indireto de auscultação de degradações para avaliação da qualidade de pavimentos aeroportuários*. Master´s dissertation. University of Beira Interior. Retrieved from http://hdl.handle.net/10400.6/10620
17. Frank, E., Hall, M. A., & Witten, I. H. (2017). The WEKA workbench. *Data Mining* (pp. 553–571). Elsevier. https://doi.org/10.1016/B978-0-12-804291-5.00024-6
18. Gholamy, A., Kreinovich, V., & Kosheleva, O. (2018). Why 70/30 Or 80/20 Relation Between Training And Testing Sets : A Pedagogical Explanation. *Departmental Technical Reports (CS)*, *1209*, 1–6.
19. Giwangkara, G. G., Mohamed, A., Nor, H. M., Hafizah A., N., & Mudiyono, R. (2020). Analysis of Internal Friction Angle and Cohesion Value for Road Base Materials in a Specified Gradation. *Journal of Advanced Civil and Environmental Engineering*, *3*(2), 58. https://doi.org/10.30659/jacee.3.2.58-65
20. Goneppanavar, U., Ali, Z., Bhaskar, Sb., & Divatia, J. (2019). Types of data, methods of collection, handling and distribution. *Airway*, *2*(1), 36. https://doi.org/10.4103/ARWY.ARWY_11_19
21. Hayaty, M., Muthmainah, S., & Ghufran, S. M. (2021). Random and Synthetic Over-Sampling Approach to Resolve Data Imbalance in Classification. *International Journal of Artificial Intelligence Research*, *4*(2), 86. https://doi.org/10.29099/ijair.v4i2.152
22. Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development*, *15*(14), 5481–5487. https://doi.org/10.5194/gmd-15-5481-2022
23. Irfan, M., Khurshid, M. B., Iqbal, S., & Khan, A. (2015). Framework for airfield pavements management—an approach based on cost-effectiveness analysis. *European Transport Research Review*, *7*(2), 13. https://doi.org/10.1007/s12544-015-0165-5
24. Issa, A., Samaneh, H., & Ghanim, M. (2022). Predicting pavement condition index using artificial neural networks approach. *Ain Shams Engineering Journal*, *13*(1), 101490. https://doi.org/10.1016/j.asej.2021.04.033
25. Japkowicz, N. (2000). The Class Imbalance Problem: Significance and Strategies. *Proc. of the Int. Conf. on artificial intelligence, 50,* 111-117.
26. Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, *6*(1), 27. https://doi.org/10.1186/s40537-019-0192-5
27. Karim, F. M. A., Rubasi, K. A. H., & Saleh, A. A. (2016). The Road Pavement Condition Index (PCI) Evaluation and Maintenance: A Case Study of Yemen. *Organization, Technology and Management in Construction: An International Journal*, *8*(1), 1446–1455. https://doi.org/10.1515/otmcj-2016-0008
28. Keith, C. (2005). Applied Statistics - Lesson 5. Retrieved May 26, 2024, from

http://www.andrews.edu/~calkins/math/edrm611/edrm05.htm

29. Kheirati, A., & Golroo, A. (2022). Machine learning for developing a pavement condition index. *Automation in Construction*, *139*(December 2021), 104296. https://doi.org/10.1016/j.autcon.2022.104296
30. Kim, S.-J., Bae, S.-J., & Jang, M.-W. (2022). Linear Regression Machine Learning Algorithms for Estimating Reference Evapotranspiration Using Limited Climate Data. *Sustainability*, *14*(18), 11674. https://doi.org/10.3390/su141811674
31. Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, *31*, 249–268. Retrieved from http://link.springer.com/10.1007/s10751-016-1232-6
32. Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, *5*(4), 221–232. https://doi.org/10.1007/s13748-016-0094-0
33. Kufel, J., Bargieł-Łączek, K., Kocot, S., Koźlik, M., Bartnikowska, W., Janik, M., … Gruszczyńska, K. (2023). What Is Machine Learning, Artificial Neural Networks and Deep Learning?—Examples of Practical Applications in Medicine. *Diagnostics*, *13*(15), 2582. https://doi.org/10.3390/diagnostics13152582
34. Lima, D. (2016). *Sistema de Gestão de Pavimentos Aeroportuários para Cabo Verde*. Master´s dissertation. Retrieved from http://hdl.handle.net/10400.6/6335
35. Li, L., Liu, D., Teng, L., & Zhu, J. (2024). Development of a Relationship between Pavement Condition Index and Riding Quality Index on Rural Roads: A Case Study in China. *Mathematics*, *12*(3), 410. https://doi.org/10.3390/math12030410
36. Li, Y., Zou, Z., Zhang, J., & He, Y. (2023). Study on the evolution of airport asphalt pavement integrated distress based on association rule mining. *Construction and Building Materials*, *369*, 130565. https://doi.org/10.1016/j.conbuildmat.2023.130565
37. Majidifard, H., Adu-Gyamfi, Y., & Buttlar, W. G. (2020). Deep machine learning approach to develop a new asphalt pavement condition index. *Construction and Building Materials*, *247*, 118513. https://doi.org/10.1016/j.conbuildmat.2020.118513
38. Marcelino, P., de Lurdes Antunes, M., Fortunato, E., & Gomes, M. C. (2021). Machine learning approach for pavement performance prediction. *International Journal of Pavement Engineering*, *22*(3), 341–354. https://doi.org/10.1080/10298436.2019.1609673
39. Maroco, J. (2014). *Análise Estatística com o SPSS Statistics*.
40. Mchugh, M. L. (2012). Interrater reliability: the kappa statistic. *Biochem Med (Zagreb)*, *22*(3), 272–282.
41. Mohammed, R., Rawashdeh, J., & Abdullah, M. (2020). Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results. *2020 11th International Conference on Information and Communication Systems, ICICS 2020*, (May), 243–248. https://doi.org/10.1109/ICICS49469.2020.239556
42. Nabipour, N., Karballaeezadeh, N., Dineva, A., Mosavi, A., Mohammadzadeh S., D., & Shamshirband, S. (2019). Comparative analysis of machine learning models for prediction of remaining service life of flexible pavement. *Mathematics*, *7*(12).

https://doi.org/10.3390/MATH7121198

43. Noori, H., & Sarkar, R. (2023). Airport Pavement Distress Analysis. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*. https://doi.org/10.1007/s40996-023-01240-5
44. Osman, S. A., Almoshaogeh, M., Jamal, A., Alharbi, F., Al Mojil, A., & Dalhat, M. A. (2022). Intelligent Assessment of Pavement Condition Indices Using Artificial Neural Networks. *Sustainability*, *15*(1), 561. https://doi.org/10.3390/su15010561
45. Ozer, D. J. (1985). Correlation and the Coefficient of Determination. *Psychological Bulletin*, *97*(2), 307–315. https://doi.org/10.1037/0033-2909.97.2.307
46. Parsons, T. A., & Murrell, S. (2023). Verification of Airport Pavement Serviceability Level Index Components Proposed by the Federal Aviation Administration. *Transportation Research Record: Journal of the Transportation Research Board*, *2677*(11), 629–641. https://doi.org/10.1177/03611981231166388
47. Pietersen, R., Beauregard, M., & Einstein, H. (2022). Automated method for airfield pavement condition index evaluations. *Automation in Construction*, *141*, 104408. https://doi.org/10.1016/j.autcon.2022.104408
48. Quinlan, J. R. (1992). Learning with Continuous Classes. *Proceedings of Australian Joint Conference on Artificial Intelligence*, 343–348.
49. Reich, Y. (1997). Machine Learning Techniques for Civil Engineering Problems. *Computer-Aided Civil and Infrastructure Engineering*, *12*(4), 295–310. https://doi.org/10.1111/0885-9507.00065
50. Rudin, C., & Radin, J. (2019). Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From An Explainable AI Competition. *Harvard Data Science Review*, *1*(2), 1–10. https://doi.org/10.1162/99608f92.5a8a3a3d
51. Salzberg, S. L. (1994). C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Machine Learning*, *16*(3), 235–240. https://doi.org/10.1007/BF00993309
52. Santos, M. (2009). Dimensionamento de camadas de reforço de pavimentos rodoviários flexíveis. Master's dissertation. University of Aveiro.
53. Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, *2*(3), 160. https://doi.org/10.1007/s42979-021-00592-x
54. Schober, P., & Schwarte, L. A. (2018). Correlation coefficients: Appropriate use and interpretation. *Anesthesia and Analgesia*, *126*(5), 1763–1768. https://doi.org/10.1213/ANE.0000000000002864
55. Schonlau, M., & Zou, R. Y. (2020). The random forest algorithm for statistical learning. *Stata Journal*, *20*(1), 3–29. https://doi.org/10.1177/1536867X20909688
56. Shevade, S. K., Keerthi, S. S., Bhattacharyya, C., & Murthy, K. R. K. (2000). Improvements to the SMO algorithm for SVM regression. *IEEE Transactions on Neural Networks*, *11*(5), 1188–1193. https://doi.org/10.1109/72.870050
57. Singh, A., Thakur, N., & Sharma, A. (2016). A review of supervised machine learning algorithms. *Proceedings of the 10th INDIACom; 2016 3rd International Conference on*

Computing for Sustainable Global Development, INDIACom 2016, 1310–1315.

58. Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, *14*(3), 199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88
59. Song, Y. Y., & Lu, Y. (2015). Decision tree methods: applications for classification and prediction. *Shanghai Archives of Psychiatry*, *27*(2), 130–135. https://doi.org/10.11919/j.issn.1002-0829.215044
60. Taye, M. M. (2023). Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions. *Computers*, *12*(5), 91. https://doi.org/10.3390/computers12050091
61. Tu, J. V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*, *49*(11), 1225–1231. https://doi.org/10.1016/S0895-4356(96)00002-9
62. U.S Department of Transportation. (2006). Geotechnical Aspects of Pavements. In *NHI Course No.132040*. Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Geotechnical+Aspects+of+Pavements#1
63. USACE. (2009). *Asphalt Surface Airfields: PAVER™ Distress Identification Manual.*
64. Yang, C., Fridgeirsson, E. A., Kors, J. A., Reps, J. M., & Rijnbeek, P. R. (2024). Impact of random oversampling and random undersampling on the performance of prediction models developed using observational health data. *Journal of Big Data*, *11*(1). https://doi.org/10.1186/s40537-023-00857-7
65. Zhang, J. Guo, W. Song, B. Zhuo, Y. Zhang, Y. (2021). Performance Prediction of Asphalt Pavement Based on Random Forest. *Journal of Beijing University of Technology*, *47*(11), 1256–1263. https://doi.org/10.11936/bjutxb2020020010

7 Appendices

Appendix I. Database extraction.

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (numeric, 3 classes and 7 classes).

Appendix III. WEKA´s output of over sampling for the best 7 classes PCI variable.

Appendix I. Database extract

Sample Unit	Section naming	Section	PCI Value	PCI 3 Classes	PCI 7 Classes	1. Alligator Cracking			2.Bleeding			3.Block Cracking			4. Corrugation			5.Depression			6.JetBlast			7.JointReflection(PCC)		
						L	M	H	L	M	H	L	M	H	L	M	H	L	M	H	L	M	H	L	M	H
1	S002Av	1	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	S008Av	1	40	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	S014Av	1	45	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	S020Av	1	40	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	S026Av	1	52	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	S032Av	1	22	poor	Serious	14.68	16.21	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	S038Av	1	46	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8	S044Av	1	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
9	S050Av	1	37	poor	Very Poor	0.00	8.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	S056Av	1	46	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
11	S062Av	1	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
12	S068Av	1	15	poor	Serious	22.31	19.65	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
13	S074Av	1	54	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	S080Av	1	48	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
15	S001Bv	2	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
16	S003Bv	2	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
17	S009Bv	2	41	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
18	S017Bv	2	27	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	S025Bv	2	28	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	S033Bv	2	24	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	S041Bv	2	18	poor	Serious	10.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22	S049Bv	2	16	poor	Serious	0.00	2.63	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
23	S057Bv	2	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
24	S065Bv	2	59	fair	Fair	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
25	S073Bv	2	38	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
26	S081Bv	2	47	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
27	S089Bv	2	31	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
28	S097Bv	2	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
29	S105Bv	2	25	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
30	S002Cv	3	47	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
31	S008Cv	3	23	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
32	S014Cv	3	32	poor	Very Poor	0.00	18.55	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
33	S020Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
34	S026Cv	3	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
35	S032Cv	3	19	poor	Serious	0.00	6.23	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
36	S038Cv	3	35	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
37	S044Cv	3	53	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
38	S050Cv	3	32	poor	Very Poor	0.00	9.82	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
39	S056Cv	3	39	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
40	S062Cv	3	54	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
41	S068Cv	3	29	poor	Very Poor	0.00	14.79	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
42	S074Cv	3	24	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Appendix I. Database extract

Sample Unit	Section naming	Section	PCI Value	PCI 3 Classes	PCI 7 Classes	8.Long.& Trans. Cracking			9.OilSpillage			10.Patching			11.PolishedAggregate			12.Raveling			13. Rutting		
						L	M	H	L	M	H	L	M	H	L	M	H	L	M	H	L	M	H
1	S002Av	1	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.60	0.00	0.00	0.00	0.00	10.24	0.00	0.00	0.00	0.00
2	S008Av	1	40	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.42	0.84	0.00	0.00	0.00	0.00	0.00	1.03	0.00	0.00	0.00
3	S014Av	1	45	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	4.12	0.00	1.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	S020Av	1	40	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.40	0.00	0.00	0.00	0.00	0.00	6.85	0.00	0.00	0.00
5	S026Av	1	52	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.64	0.00	0.00	0.00	0.00	0.00	2.06	0.00	0.00	0.00
6	S032Av	1	22	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	S038Av	1	46	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.40	1.18	0.00	0.00	0.00	0.00	0.00	14.13	0.00	0.00	0.00
8	S044Av	1	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	1.50	0.00	0.00	0.00	0.00	1.89	0.00	0.00	0.00	0.00
9	S050Av	1	37	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.86	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	S056Av	1	46	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	1.00	0.00	0.00	0.00	0.00	12.71	0.00	0.00	0.00	0.00
11	S062Av	1	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.22	1.00	0.00	0.00	0.00	0.00	0.00	6.16	0.00	0.00	0.00
12	S068Av	1	15	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.52	0.00	0.00	0.00	0.00	31.51	0.00	0.00	0.00	0.00
13	S074Av	1	54	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.22	2.22	0.00	0.00	0.00	0.00	0.00	2.03	0.00	0.00	0.00
14	S080Av	1	48	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.20	2.12	0.00	0.00	0.00	0.00	0.00	15.51	0.00	0.00	0.00
15	S001Bv	2	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.40	0.40	0.00	0.00	0.00	0.00	1.15	27.73	0.00	0.00	0.00
16	S003Bv	2	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.70	0.00	0.00	0.00	0.00	0.00	34.03	0.00	0.00	0.00
17	S009Bv	2	41	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.40	1.08	0.00	0.00	0.00	0.00	3.00	18.60	0.00	0.00	0.00
18	S017Bv	2	27	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	17.28	0.00	0.00	0.00
19	S025Bv	2	28	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	18.88	0.56	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	S033Bv	2	24	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	34.29	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	S041Bv	2	18	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.20	0.40	0.00	0.00	0.00	0.00	10.06	0.00	0.00	0.00	0.00
22	S049Bv	2	16	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	0.00	18.34	0.00	0.00	0.00
23	S057Bv	2	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	54.18	0.00	0.00	0.00
24	S065Bv	2	59	fair	Fair	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.32	0.00	0.00	0.00	0.00	0.00	8.34	0.00	0.00	0.00
25	S073Bv	2	38	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	49.66	0.00	0.00	0.00
26	S081Bv	2	47	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	22.11	0.00	0.00	0.00
27	S089Bv	2	31	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.60	0.00	0.00	0.00	0.00	0.00	27.40	0.00	0.00	0.00
28	S097Bv	2	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	19.06	0.60	0.00	0.00	0.00	0.00	6.58	0.00	0.00	0.00	0.00
29	S105Bv	2	25	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	15.12	0.00	0.00	0.00	0.00	0.00	0.00	9.83	0.00	0.00	0.00
30	S002Cv	3	47	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.96	0.00	0.00	0.00	0.00	0.00	47.62	0.00	0.00	0.00
31	S008Cv	3	23	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.04	0.00	0.00	0.00	0.00	0.00	39.98	0.00	0.00	0.00
32	S014Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	7.85	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
33	S020Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.40	0.00	0.00	0.00	0.00	0.00	0.00	0.68	0.00	0.00	0.00
34	S026Cv	3	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.28	0.00	0.00	0.00	0.00	0.00	9.06	0.00	0.00	0.00
35	S032Cv	3	19	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
36	S038Cv	3	35	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
37	S044Cv	3	53	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.84	0.42	0.00	0.00	0.00	0.00	0.00	12.44	0.00	0.00	0.00
38	S050Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.42	0.50	0.00	0.00	0.00	0.00	0.00	11.05	0.00	0.00	0.00
39	S056Cv	3	39	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.40	1.00	0.00	0.00	0.00	0.00	0.00	22.03	0.00	0.00	0.00
40	S062Cv	3	54	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.84	0.68	0.00	0.00	0.00	0.00	0.00	23.36	0.00	0.00	0.00
41	S068Cv	3	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.46	0.52	0.00	0.00	0.00	0.00	19.14	0.00	0.00	0.00	0.00
42	S074Cv	3	24	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.20	1.18	0.00	0.00	0.00	0.00	0.00	40.20	0.00	0.00	0.00

Appendix I. Database extract

Sample Unit	Section naming	Section	PCI Value	PCI 3 Classes	PCI 7 Classes	14.ShovingFromPCC			15.SlippageCracking			16.Swell			17.Weathering		
						L	M	H	L	M	H	L	M	H	L	M	H
1	S002Av	1	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	13.20	33.87
2	S008Av	1	40	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.68	0.00
3	S014Av	1	45	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	S020Av	1	40	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	88.60
5	S026Av	1	52	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	33.80	0.00	25.60
6	S032Av	1	22	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20.40	0.00	0.00
7	S038Av	1	46	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	42.94
8	S044Av	1	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	31.20
9	S050Av	1	37	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	S056Av	1	46	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	23.33
11	S062Av	1	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20.22
12	S068Av	1	15	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.02
13	S074Av	1	54	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	51.65
14	S080Av	1	48	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	8.10	23.93
15	S001Bv	2	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.80	22.80
16	S003Bv	2	26	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	35.13
17	S009Bv	2	41	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	9.01	18.88
18	S017Bv	2	27	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	31.87
19	S025Bv	2	28	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	22.62
20	S033Bv	2	24	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	55.62
21	S041Bv	2	18	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.24	3.11
22	S049Bv	2	16	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	33.45
23	S057Bv	2	30	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	22.73
24	S065Bv	2	59	fair	Fair	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	12.94	8.71
25	S073Bv	2	38	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	41.36
26	S081Bv	2	47	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.78	24.80
27	S089Bv	2	31	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	25.40
28	S097Bv	2	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.29	21.61
29	S105Bv	2	25	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	14.00	23.40
30	S002Cv	3	47	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
31	S008Cv	3	23	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	16.05
32	S014Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	5.61
33	S020Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	31.40
34	S026Cv	3	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	15.11
35	S032Cv	3	19	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	18.63
36	S038Cv	3	35	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	48.11
37	S044Cv	3	53	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
38	S050Cv	3	32	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	16.15
39	S056Cv	3	39	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.80
40	S062Cv	3	54	poor	Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.95	0.00	0.00
41	S068Cv	3	29	poor	Very Poor	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	15.36	0.00
42	S074Cv	3	24	poor	Serious	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	19.02

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (numeric)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	Numerical PCI

Sample Unit	Instance	Actual PCI	Predicted PCI	Error (= aPCI-pPCI)
1	1	25.00	31.08	-6.08
2	2	72.00	75.24	-3.24
3	3	34.00	32.76	1.25
4	4	73.00	66.69	6.31
5	5	66.00	66.72	-0.72
6	6	46.00	36.05	9.95
7	7	29.00	39.06	-10.06
8	8	74.50	75.05	-0.55
9	9	74.50	76.33	-1.83
10	10	40.00	39.79	0.21
11	11	80.50	79.85	0.65
12	12	27.00	34.51	-7.51
13	13	38.00	29.91	8.09
14	14	75.50	75.05	0.45
15	15	77.00	75.40	1.60
16	16	74.00	75.09	-1.09
17	17	74.50	75.59	-1.09
18	18	81.00	79.52	1.48
19	19	67.00	63.58	3.42
20	20	75.00	79.34	-4.34
21	21	54.00	44.62	9.38
22	22	81.00	80.78	0.22
23	23	57.00	60.21	-3.21
24	24	33.00	40.74	-7.74
25	25	73.00	62.49	10.51
26	26	59.00	36.37	22.63
27	27	59.00	41.59	17.41
28	1	83.00	87.03	-4.03
29	2	35.00	43.13	-8.13
30	3	87.00	87.95	-0.95
31	4	29.00	37.96	-8.96
32	5	67.00	60.76	6.24
33	6	76.00	75.93	0.07
34	7	29.00	35.84	-6.84
35	8	30.00	40.08	-10.08
36	9	74.50	74.89	-0.39
37	10	27.00	40.63	-13.63
38	11	47.00	60.04	-13.04
39	12	44.00	49.08	-5.08
40	13	46.00	44.11	1.89
41	14	73.00	77.05	-4.05
42	15	28.00	39.81	-11.81
43	16	26.00	44.93	-18.93
44	17	75.00	75.07	-0.07
45	18	32.00	34.20	-2.20
46	19	35.00	47.61	-12.61
47	20	29.00	47.72	-18.72
48	21	75.50	75.47	0.03
49	22	78.00	74.46	3.54
50	23	40.00	47.26	-7.26
51	24	75.50	75.57	-0.07
52	25	88.00	74.26	13.74
53	26	79.50	78.62	0.88
54	1	74.50	75.61	-1.11
55	2	76.00	75.90	0.10
56	3	75.00	75.44	-0.44
57	4	48.00	33.73	14.27
58	5	76.80	76.61	0.19
59	6	75.70	79.99	-4.29
60	7	75.00	76.78	-1.78

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (numeric)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	Numerical PCI

Sample Unit	Instance	Actual PCI	Predicted PCI	Error (= aPCI-pPCI)
61	8	76.50	75.75	0.75
62	9	20.00	37.44	-17.44
63	10	75.00	76.24	-1.24
64	11	41.00	33.70	7.30
65	12	74.50	75.25	-0.75
66	13	77.50	73.98	3.52
67	14	52.00	40.21	11.79
68	15	64.00	62.03	1.97
69	16	18.00	32.05	-14.05
70	17	35.00	33.68	1.32
71	18	76.00	79.84	-3.84
72	19	85.50	84.94	0.56
73	20	75.00	75.97	-0.97
74	21	81.00	77.89	3.11
75	22	38.00	35.59	2.41
76	23	74.50	74.95	-0.44
77	24	81.00	79.89	1.11
78	25	53.00	38.24	14.76
79	26	85.00	84.85	0.15
80	1	75.50	75.31	0.19
81	2	75.00	75.38	-0.38
82	3	88.00	76.81	11.19
83	4	53.00	50.65	2.36
84	5	86.00	83.65	2.35
85	6	35.00	40.40	-5.40
86	7	62.00	61.15	0.85
87	8	78.00	73.90	4.10
88	9	15.00	27.25	-12.25
89	10	54.00	38.36	15.64
90	11	75.00	78.72	-3.72
91	12	74.80	75.22	-0.42
92	13	62.00	58.17	3.83
93	14	78.00	76.25	1.75
94	15	75.70	75.70	0.00
95	16	74.50	75.20	-0.70
96	17	23.00	30.18	-7.18
97	18	32.00	37.63	-5.63
98	19	75.00	75.82	-0.82
99	20	23.00	46.55	-23.55
100	21	74.00	74.60	-0.60
101	22	52.00	54.20	-2.20
102	23	79.50	79.80	-0.30
103	24	24.00	48.52	-24.52
104	25	74.50	75.01	-0.50
105	26	30.00	41.49	-11.49
106	1	78.50	75.73	2.77
107	2	8.00	28.91	-20.91
108	3	76.00	75.04	0.97
109	4	75.00	76.59	-1.59
110	5	71.00	74.94	-3.94
111	6	72.00	77.34	-5.34
112	7	85.00	83.30	1.70
113	8	48.00	35.73	12.27
114	9	43.00	39.78	3.22
115	10	75.00	76.61	-1.61
116	11	79.50	79.41	0.09
117	12	60.00	41.12	18.88
118	13	76.00	80.51	-4.51
119	14	73.00	74.98	-1.98
120	15	68.00	75.73	-7.73

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (numeric)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	Numerical PCI

Sample Unit	Instance	Actual PCI	Predicted PCI	Error (= aPCI-pPCI)
121	16	80.00	78.33	1.67
122	17	80.00	78.68	1.32
123	18	85.00	83.71	1.29
124	19	27.00	31.34	-4.34
125	20	90.00	86.10	3.91
126	21	25.00	36.62	-11.62
127	22	24.00	35.94	-11.94
128	23	80.00	80.75	-0.75
129	24	74.50	74.57	-0.07
130	25	74.50	75.42	-0.92
131	26	22.00	28.20	-6.20
132	1	76.00	75.95	0.05
133	2	77.00	76.24	0.76
134	3	80.00	77.85	2.15
135	4	87.00	73.83	13.17
136	5	91.00	86.63	4.37
137	6	22.00	37.59	-15.59
138	7	79.50	77.61	1.89
139	8	84.00	83.45	0.55
140	9	41.00	46.12	-5.12
141	10	30.00	31.26	-1.26
142	11	24.00	39.56	-15.56
143	12	80.00	79.19	0.81
144	13	24.00	35.72	-11.72
145	14	76.00	74.98	1.03
146	15	43.00	50.23	-7.23
147	16	82.00	75.52	6.48
148	17	32.00	29.90	2.10
149	18	76.00	77.55	-1.55
150	19	44.00	46.27	-2.27
151	20	17.00	31.11	-14.11
152	21	80.00	77.64	2.36
153	22	52.00	55.14	-3.14
154	23	76.00	73.56	2.44
155	24	33.00	43.38	-10.38
156	25	74.50	75.22	-0.72
157	26	67.00	63.45	3.56
158	1	75.00	75.91	-0.91
159	2	80.00	79.64	0.36
160	3	11.00	32.71	-21.71
161	4	31.00	33.01	-2.01
162	5	78.00	78.98	-0.98
163	6	6.00	29.77	-23.77
164	7	36.00	37.68	-1.68
165	8	17.00	25.62	-8.62
166	9	74.50	74.75	-0.25
167	10	80.00	79.49	0.51
168	11	56.00	40.71	15.30
169	12	24.00	42.17	-18.17
170	13	86.00	69.50	16.50
171	14	75.00	74.98	0.02
172	15	75.00	75.27	-0.27
173	16	62.00	56.13	5.87
174	17	63.00	31.38	31.63
175	18	47.00	32.60	14.40
176	19	75.00	75.12	-0.12
177	20	31.00	46.03	-15.03
178	21	30.00	31.15	-1.15
179	22	86.00	85.48	0.53
180	23	81.00	79.38	1.62

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (numeric)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	Numerical PCI

Sample Unit	Instance	Actual PCI	Predicted PCI	Error (= aPCI-pPCI)
181	24	74.50	75.92	-1.42
182	25	79.00	78.19	0.81
183	26	67.00	64.53	2.47
184	1	27.00	36.80	-9.80
185	2	62.00	66.21	-4.21
186	3	16.00	36.87	-20.87
187	4	15.00	28.63	-13.63
188	5	26.00	40.84	-14.84
189	6	47.00	53.73	-6.73
190	7	20.00	38.94	-18.94
191	8	26.00	31.69	-5.69
192	9	32.00	45.59	-13.59
193	10	45.00	45.40	-0.40
194	11	31.00	29.42	1.58
195	12	26.00	39.17	-13.17
196	13	81.00	80.17	0.83
197	14	70.00	74.69	-4.69
198	15	73.00	76.82	-3.82
199	16	71.00	73.81	-2.81
200	17	74.50	75.78	-1.28
201	18	76.00	75.63	0.37
202	19	80.00	79.11	0.89
203	20	53.00	40.24	12.76
204	21	63.00	64.07	-1.07
205	22	40.00	38.00	2.00
206	23	76.50	77.27	-0.77
207	24	30.00	50.18	-20.18
208	25	80.50	79.53	0.97
209	26	35.00	57.07	-22.07
210	1	64.00	61.14	2.86
211	2	47.00	29.61	17.39
212	3	75.00	75.10	-0.10
213	4	32.00	31.77	0.23
214	5	75.00	75.20	-0.20
215	6	74.50	75.91	-1.41
216	7	76.00	75.83	0.17
217	8	81.00	78.60	2.41
218	9	22.00	34.56	-12.56
219	10	75.00	75.37	-0.37
220	11	18.00	33.81	-15.81
221	12	68.00	61.52	6.49
222	13	39.00	31.86	7.14
223	14	74.80	75.47	-0.67
224	15	32.00	38.22	-6.22
225	16	79.50	78.18	1.32
226	17	25.00	31.12	-6.12
227	18	83.00	71.54	11.46
228	19	19.00	30.63	-11.63
229	20	37.00	45.82	-8.82
230	21	26.00	32.31	-6.31
231	22	89.00	83.75	5.25
232	23	38.00	37.86	0.14
233	24	95.00	73.97	21.03
234	25	76.00	75.01	0.99
235	26	76.00	77.56	-1.56
236	1	78.00	76.88	1.13
237	2	71.00	75.10	-4.10
238	3	17.00	37.39	-20.39
239	4	73.00	73.21	-0.21
240	5	78.00	72.29	5.71

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (numeric)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	Numerical PCI

Sample Unit	**Instance**	**Actual PCI**	**Predicted PCI**	**Error (= aPCI-pPCI)**
241	6	52.00	40.92	11.08
242	7	36.00	37.89	-1.89
243	8	45.00	48.13	-3.13
244	9	25.00	53.91	-28.91
245	10	28.00	32.06	-4.06
246	11	46.00	55.85	-9.85
247	12	20.00	25.72	-5.72
248	13	85.00	83.45	1.55
249	14	81.00	76.77	4.24
250	15	75.00	74.67	0.33
251	16	75.00	75.10	-0.10
252	17	74.50	74.85	-0.35
253	18	60.00	56.95	3.05
254	19	46.00	32.45	13.55
255	20	49.00	66.56	-17.56
256	21	46.00	30.90	15.10
257	22	76.00	75.44	0.56
258	23	75.00	75.03	-0.03
259	24	85.00	84.24	0.76
260	25	80.70	78.30	2.40
261	26	82.50	82.75	-0.25

Parameters	**Output**	**Output (%)**
Correlation Coefficient	0.93	-
Determination coefficient	0.86	-
Mean absolute error	5.81	-
Root mean squared error	8.73	
Relative absolute error (%)	-	27.96
Root relative squared error (%)	-	38.02
Total Number of Instances	261	

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (3 classes)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	3 Classes PCI

Parameters	**Output**	**Output (%)**
Correctly Classified Instances	244	93.49
Incorrectly Classified Instances	17	6.51
Kappa statistic	0.88	-
Mean absolute error	0.10	-
Root mean squared error	0.21	-
Relative absolute error (%)	-	25.58
Root relative squared error (%)	-	48.16
Total Number of Instances	261	-

Confusion Matrix

a	b	c	classified as		
99	4	2	a	=	poor
6	12	4	b	=	fair
1	0	133	c	=	good

Appendix II. WEKA´s outputs of the best models of each algorithm and type of PCI variable (7 classes)

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds
Output:	7 Classes PCI

Parameters	**Output**	**Output (%)**
Correctly Classified Instances	185	70.88
Incorrectly Classified Instances	76	29.12
Kappa statistic	0.58	-
Mean absolute error	0.10	-
Root mean squared error	0.23	-
Relative absolute error (%)	-	51.34
Root relative squared error (%)	-	72.83
Total Number of Instances	261	-

Confusion Matrix

a	b	c	d	e	f	g	classified as		
33	5	3	5	1	0	0	a	=	Very Poor
16	9	1	2	1	0	0	b	=	Poor
12	4	7	3	0	0	1	c	=	Serious
3	1	1	12	5	0	0	d	=	Fair
0	1	0	1	117	3	0	e	=	Satisfactory
0	0	0	0	5	7	0	f	=	Good
1	0	1	0	0	0	0	g	=	Failed

Appendix III. WEKA´s output of over sampling for the best 7 classes PCI variable.

Algorithm:	Random Forest
Training Model:	Cross Validation with 10 Folds

Parameters	**Output**	**Output (%)**	**Confusion Matrix**									
Correctly Classified Instances	686	92.83	a	b	c	d	e	f	g	Classified as		
Incorrectly Classified Instances	53	7.17	78	8	3	4	1	0	0	a	=	Very Poor
Kappa statistic	0.92	-	9	103	0	3	1	0	0	b	=	Poor
Mean absolute error	0.05	-	4	3	100	1	0	0	0	c	=	Serious
Root mean squared error	0.14	-	2	2	1	80	3	0	0	d	=	Fair
Relative absolute error (%)	-	22.55	0	1	0	1	115	5	0	e	=	Satisfatory
Root relative squared error (%)	-	39.64	0	0	0	0	0	96	0	f	=	good
Total Number of Instances	739	-	0	0	1	0	0	0	114	g	=	Failed