

**An optimization-based AI investigation for
crack detection in single crystal GaAs using
nanoindentation data**



**By
Talha Tariq**

**School of Chemical and Materials Engineering
National University of Sciences and Technology
2024**

**An optimization-based AI investigation for
crack detection in single crystal GaAs
using nanoindentation data**



Name: Talha Tariq

Reg. No.: Fall-2021-MSE16-00000363069

**This work is submitted as a partial fulfillment of the requirement for
the degree of**

MS in Materials & Surface Engineering

Supervisor Name: Dr. -Ing. Farhan Javaid

Co-Supervisor Name: Dr. Muhammad Nouman Aslam Khan

School of Chemical and Material Engineering (SCME)

National University of Sciences and Technology (NUST)

H-12 Islamabad, Pakistan

January, 2024



THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS thesis written by Mr. Talha Tariq, Registration No. 00000363069, of School of Chemical & Materials Engineering (SCME) has been vetted by undersigned, found complete in all respects as per NUST Statues/Regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.


Signature: 

Name of Supervisor: Dr. -Ing. Farhan Javaid

Date: 19.01.2024

Signature (HOD): 

Date: 19.1.24

Signature (Dean/Principal): 

Date: 19.1.2024

National University of Sciences & Technology (NUST)
MASTER'S THESIS WORK

Formulation of Guidance and Examination Committee (GEC)

Name: Talha Tariq NUST Reg No: 00000363069
 Department: Department of Materials Engineering Specialization: Master of Science in Materials & Surface Engineering
 Credit Hour Completed: 18.0 CGPA: 3.33

Course Work Completed

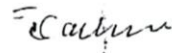


S/No:	Code:	Title:	Core/Elective:	CH:	Grade:
1.	MSE-811	Materials Thermodynamics	Compulsory	3.0	A
2.	MSE-821	Mechanical Behaviour Of Materials	Compulsory	3.0	B
3.	RM-898	Research Methodology	Additional	2.0	Q
4.	MSE-881	Manufacturing Processes	Elective	3.0	A
5.	MSE-812	Phase Transformation & Microstructures	Compulsory	3.0	C+
6.	MSE-851	Surface Engineering & Characterisation	Compulsory	3.0	B+
7.	MSE-854	Characterization Of Materials	Elective	3.0	B

Date 02 - Nov - 2022


Student's Signature: 

Thesis Committee

- Name: Farhan Javaid (Supervisor)
Department: Department of Materials Engineering
- Name: Usman Liaqat (Internal)
Department: Department of Materials Engineering
- Name: Muhammad Shahid (Internal)
Department: Department of Materials Engineering

Signature 
 Signature 
 Signature 

Date: 02 - Nov - 2022

Signature of Head of Department: 

APPROVAL

Date: 02 - Nov - 2022

Signature of Dean/Principal: 



National University of Sciences & Technology (NUST)

FORM TH-4

MASTER'S THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by

Regn No & Name: 00000363069 Talha Tariq

Title: An Optimization-based AI Investigation for crack detection in single crystal GaAs using nanoindentation data.

Presented on: 18 Jan 2024 at: 1430 hrs in SCME Seminar Hall

Be accepted in partial fulfillment of the requirements for the award of Masters of Science degree in Materials & Surface Engineering.

Guidance & Examination Committee Members

Name: Dr Usman Liaqat

Signature: [Signature]

Name: Dr Mohammad Shahid

Signature: [Signature]

Name: Dr M. Nouman Aslam Khan Co-Supervisor

Signature: [Signature]

Supervisor's Name: Dr Farhan Javaid

Signature: [Signature]

Dated: 18-JAN-2024

[Signature]
Head of Department

[Signature]
Dean/Principal

Date 18/1/24

Date 18.1.2024

School of Chemical & Materials Engineering (SCME)

Dedication

*To my very Supportive Parents and
My elder Brother*

Acknowledgments

I would like to begin by expressing my gratitude to the **Almighty Allah** for bestowing his blessings and guiding me through this effort.

My sincere gratitude goes out to my supervisor, **Dr. -Ing. Farhan Javaid**, and my co-supervisor, **Dr. Muhammad Nouman Aslam Khan**, for their prompt counsel, helpfulness, and practical mindset throughout this project. It is crucial to remember that my project has only been completed because of his relentless effort. There were numerous obstacles and challenges during my investigation, but they stayed approachable and pursued me to do the task more effectively.

I would like to acknowledge the GEC members, **Dr. Usman Liaqat** and **Dr. Muhammad Shahid**, for their insightful insights and constructive criticism, which considerably improved the quality of this work.

A sincere thanks is extended to the **School of Chemical and Materials Engineering (SCME)** for cultivating a scholarly atmosphere that promotes outstanding performance. The department's invaluable resources, mentorship, and collaborative environment have significantly influenced the trajectory of this research undertaking.

I would like to convey my deepest appreciation to our department head, **Dr. Khurram Yaqoob**, and principal, **Dr. Amir Azam Khan**, for their unwavering commitment to the advancement of education and their exemplary leadership. Their support and encouragement have been indispensable throughout this academic journey.

My sincere appreciation also extends to my family. Their steadfast motivation, assistance, and compassion served as fundamental components of my resilience. This accomplishment would not have been feasible without their unwavering dedication and kindness.

Abstract

This research introduces a novel approach to predict and analyze the presence of microcracks in gallium arsenide (GaAs) by employing ensemble learning tree (ELT), support vector machine (SVM), decision tree (DT), and gaussian process regression (GPR). The selection of the model is accomplished by assessing the value of the coefficient of determination (R^2) and the root mean square error (RMSE). The process of selecting the model involves evaluating the R^2 and RMSE values and the performance of these four models differs depending on the characteristics of the data. In the genetic algorithm (GA), the values of R^2 for GPR, SVM, DT, and ELT are 0.9979, 0.9780, 0.7744, and 0.9829, respectively, and the corresponding RMSE values are $6.25E^{-16}$, $2.61E^{-14}$, and $6.10E^{-4}$ and $3.26E^{-14}$. In contrast, particle swarm optimization (PSO) yields R^2 values of 0.9979, 0.9721, 0.7744, and 0.9581, with corresponding RMSE values of $7.31E^{-16}$, $2.10E^{-14}$, $5.10E^{-4}$, and $2.87E^{-6}$. Moreover, the GPR model demonstrates superior performance in dealing with complex variable target relationships as compared to other machine learning (ML) models in cracking detection. Additionally, the investigation of partial dependence plots (PDP) emphasizes the significance of load and displacement parameters for precision crack prediction. It helps in understanding complex model behaviors and revealing feature target relationships. Lastly, a user-friendly Graphical User Interface (GUI) has been meticulously designed to take advantage of the GA-based GPR model, enabling smooth computation of crack detection. The incorporation of these sophisticated methodologies not only improves prediction accuracy but also provides researchers with valuable tools to identify the anticipation of microcrack formation, contributing to the development of predictive modelling and materials science.

Keywords:

Nanoindentation, Machine Learning, Genetic Algorithm, Particle Swarm Optimization, Gallium Arsenide, Crack Prediction, Partial Dependence analysis.

Table of Contents

Dedication	i
Acknowledgments	ii
Abstract	iii
Nomenclature	ix
Chapter 1:	1
Introduction	1
1.1. Gallium Arsenide	1
1.2. Manufacturing Techniques	1
1.3. Comparative Analysis of GaAs Manufacturing Techniques	2
1.4. Segmenting Challenges	3
1.5. Challenges in Nanoindentation	4
1.6. Cutting-Edge AI Techniques	5
1.7. Problem Statement	5
1.8. Proposed Methodology	5
1.9. Objectives	6
Chapter 2:	7
Literature Review	7
2.1. Nanoindentation	7
2.1. Types of Nano-indenters	9
2.2. Indentation Methods	9
2.3. Pop-in Events in Load-Displacement Curve	10
2.4. Cracking in GaAs	11
2.5. Crack Observing Techniques	12
2.5.1. Ex-Situ Study	12
2.5.2. In-Situ Study	13
2.5.2.1. Cracking Sequence	13
2.6. Artificial Intelligence	15
2.6.1. Machine Learning	16
2.6.2. ML Models	17

2.6.2.1.	Decision Tree.....	17
2.6.2.2.	Support Vector Machine.....	19
2.6.2.3.	Gaussian Process Regression.....	20
2.6.2.4.	Ensemble Learning Tree.....	22
2.6.3.	Overview of Optimization Algorithm.....	22
2.6.3.1.	Genetic Algorithm.....	22
2.6.3.2.	Particle Swarm Optimization.....	24
Chapter 3:	26
Methodology	26
3.1.	Data Collection.....	26
3.1.1.	Overview of Python Plotting.....	27
3.1.1.1.	Box Plots.....	27
3.1.1.2.	Pair Plots.....	28
3.1.1.3.	Pearson Correlation.....	28
3.2.	Data Pre-processing.....	29
Chapter 4:	32
Results and Discussion.....	32
4.1.	Box plot, Pair plot, Correlation heatmap.....	32
4.2.	Performance Criterion of ML models.....	35
4.3.	Features Selection.....	37
4.4.	GA and PSO based hyperparameters tuning.....	38
4.5.	Models' performance.....	39
4.6.	Partial dependence plots.....	41
4.7.	Graphical representation between actual and predicted cracks.....	43
4.8.	Graphical user interface.....	44
Conclusions and Recommendations.....	46
Conclusions.....	46
Recommendations.....	46

List of Figures

Figure 1: Schematic Depiction of Breakage Mechanism [15].	3
Figure 2: SEM images of radial cracks around Berkovich indentation at different depths - (a) 20 μm and (b) 1 μm [21].	4
Figure 3: Schematic of the Workflow for Crack Prediction Utilizing Machine Learning.	6
Figure 4: Load-Displacement Curve during Nanoindentation [29].	7
Figure 5: (a) Nano-indenter G2000, (b) Cross-sectional View of G200 Nano-indenter [32].	8
Figure 6: (a) First Pop-in Occurrence within the Load-Displacement Curve, (b) Multiple Subsequent Pop-in Events.	10
Figure 7: Sequential micrographs illustrating the cracking process with a conical 60° indenter under varying loads, captured during in situ nanoindentation: (a) Loading at 150 mN, (b) Loading at 300 mN, (c) Unloading at 50 mN, (d) Full load at 500 mN, (e) Unloading complete with indenter withdrawal, and (f) Corresponding Load–Displacement Curve. The arrows (1) and (2) indicate radial cracks, while the arrow (3) and arrow (4) indicate slide lines and fragments, respectively [14].	14
Figure 8: Visualization encompassing key elements of AI. ML, DL, and ANN [51].	15
Figure 9: Representation of Decision Tree Structure.	18
Figure 10: Representation of SVM Structure.	19
Figure 11: Representation of GPR Structure.	20
Figure 12: Representation of ELT Structure.	21
Figure 13: Representation of workflow of GA.	23
Figure 14: Nature inspired Technique (PSO).	24
Figure 15: Workflow of PSO.	25
Figure 16: Data Classification Using Python Programming Language	26
Figure 17: Algorithmic Representation of Box Plots.	27
Figure 18: Algorithmic Representation of Pair Plots.	28
Figure 19: Algorithmic Representation of Box Plots.	29
Figure 20: Machine Learning Workflow.	30

Figure 21: (a) Load (mN) and Displacement (μm) Box Plot representation (b) dp , dh , dh/dp , dp/dh Box Plot representation.....32

Figure 22: Pair Plot representation, (a), (e), (I), shows the histogram. (b), (c), (d), (f), (g), and (h), indicate bivariate relationships.33

Figure 23: Pearson Correlation between inputs and output.34

Figure 24: PDP'S Demonstrating the Influence of Inputs on Microcrack Prediction.42

Figure 25: (a) DT-GA (b) ELT-GA (c) GPR-GA (d) SVM-GA predicted vs actual crack (e) SVM-GA (f) DT-PSO (g) ELT-PSO (h) GPR-PSO predicted crack against actual crack44

Figure 26: Screenshot of Graphical User Interface.....45

List of Tables

Table 1: GA and PSO parameters.	36
Table 2: GA and PSO based Features Selection.	37
Table 3: Parameter Ranges and Optimized Values in Selection Process.....	38
Table 4: Comparison of ML methods using PSO and GA.	40

Nomenclature

AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
DT	Decision Tree
SVM	Support Vector Machine
GPR	Gaussian Process Regression
EL	Ensembled Learning
ET	Ensembled Tree
GA	Genetic Algorithm
R^2	Coefficient of Determination
PSO	Particle Swarm Optimization
RMSE	Root Mean Square Error
GaAs	Gallium Arsenide
PDP	Partial Dependence Plot
MP	Manufacturing Process
GUI	Graphical User Interface
MBE	Metal Beam Epitaxy
LEC	Liquid Encapsulated Czochralski
VGf	Vertical Gradient Freeze

Chapter 1:

Introduction

1.1. Gallium Arsenide

Gallium arsenide (GaAs), a compound semiconductor with a zinc blende crystal structure, has a direct bandgap of approximately 1.42 electron volts [1]. Its distinct characteristics greatly influence a wide range of real-world applications [2]. Unlike silicon, which has an indirect bandgap and inefficient photon emission, GaAs has a direct bandgap that allows for efficient photon emission, which makes it perfect for optoelectronic devices [2]. In addition, GaAs has more electron mobility than silicon and performs better in high-frequency electronic circuits, which makes it a perfect material for complex communication systems [3]. Moreover, power amplifiers and other high-power components may benefit from its great thermal conductivity and effective heat dissipation [4]. The versatile nature of GaAs, as evidenced by its higher frequency and robust thermal characteristics, empowers it to assume a pivotal function in numerous specialized electronic and telecommunication fields, despite the challenges associated with integration and comparatively increased manufacturing expenses [5].

1.2. Manufacturing Techniques

GaAs semiconductor materials can be manufactured utilizing an extensive array of manufacturing processes (MP) and techniques. Each approach has its own distinct set of benefits and is chosen by the specific quality requirements of the end products. [6]. The most popular techniques are Liquid Encapsulated Czochralski (LEC), Vertical Gradient Freeze (VGF), Metal-Organic Chemical Vapor Deposition (MOCVD), and Metal Beam Epitaxy (MBE). The LEC method demands the controlled melting of high-purity gallium and arsenic elements in order to produce single crystal GaAs. A seed crystal is then dipped into the molten mixture and slowly withdrawn while being encapsulated to grow a single, high-quality GaAs crystal [7]. In VGF, a seed crystal is similarly pulled to the chamber's surface utilizing a regulated temperature gradient. This precise process results in the production of single crystal GaAs wafers [8]. In MBE, it utilizes heated sources to emit

atom beams that deposit gallium and arsenic atoms on a heated substrate, layer by layer [9]. On the other hand, MOCVD involves the introduction of arsenic and gaseous organometallic gallium compounds into a reactor compartment, where they transform into crystals and subsequently adhere to a heated substrate [10]. All strategies offer precise control over the growing environment in order to provide the necessary material properties.

1.3. Comparative Analysis of GaAs Manufacturing Techniques

This comparative analysis of GaAs is crucial for the advancement of electronic and optoelectronic device technology. Their quality, efficacy, and applicability are all determined by the manufacturing processes employed in the production of GaAs crystals. There are four well-known techniques for fabricating GaAs are evaluated: LEC, VGF, MBE, and, MOCVD. The unique characteristics of each method, focus on crystal quality, cost, impurity control, and applications, and provide decision-makers and researchers with the knowledge to choose the most suitable strategy for their GaAs-related projects. For example, LEC excels in applications requiring high purity and cost-effectiveness as a result of the liquid encapsulation process, which reduces impurity levels and provides a cost-effective solution [11]. Prioritizing extraordinary crystal quality and purity over growth rate, LEC becomes the preferred option when a moderate growth rate is acceptable [12]. On the contrary, when a quicker rate of growth is of the utmost importance, the VGF technique emerges as a feasible alternative, facilitating advanced crystal formation [8]. Despite its comparatively lower reputation for crystal quality in comparison to LEC, VGF is still capable of generating crystals of exceptional quality; thus, the selection between the two processes in the semiconductor industry is dependent upon particular growth parameters and application demands. In the semiconductor industry, MOCVD is extensively utilised to mass produce GaAs-based devices, regardless of its lower cost for large-scale GaAs production due to its greater growth rates and reduced apparatus expenditures [10]. MBE offers precise control and high purity but comes with higher equipment cost and slower growth rate, making it suitable for high-quality, low-volume applications [13].

1.4. Segmenting Challenges

A significant challenge that arises after the manufacturing of GaAs is the integration of multiple identical pattern circuits layered upon each other on a single wafer. This complexity necessitates the need for precise techniques to convert a single wafer into multiple chips [14]. A wide range of methodologies are available for performing the dicing operation; but the 'scribe and break' technique remains the most frequently employed, as displayed in Figure 1.

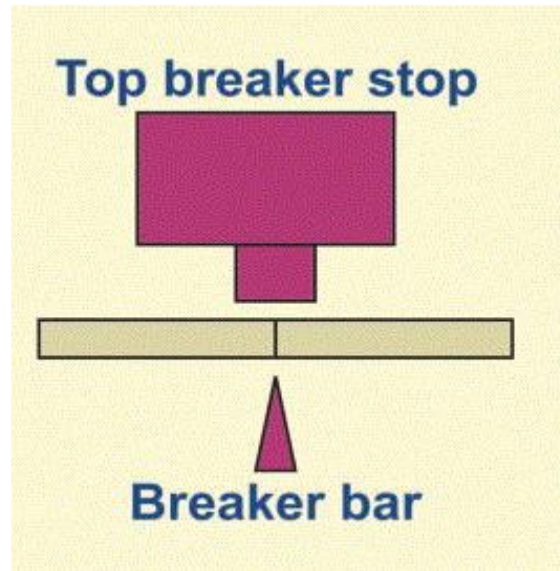


Figure 1: Schematic Depiction of Breakage Mechanism [15].

In order to commence this process, an initial cracking is generated on the wafer's surface through the scratching of its surface with a diamond instrument. Subsequently, the wafer is exposed to controlled stress or pressure. The separation is then completed by slicing the wafer along particular crystallographic planes [16]. However, this process can be cumbersome and may introduce stress concentrations in the material, potentially leading to defects and reduced chip quality [17]. To address these challenges, nanoindentation emerged as an effective method. In contrast to conventional dicing, nanoindentation improves the process of chip fabrication by generating indents at a precise distance to the thickness of the chip [18].

1.5. Challenges in Nanoindentation

The utilization of nanoindentation becomes increasingly significant in understanding mechanical properties as we proceed into the domain of the nanoscale, where materials show distinctive characteristics. However, the goal of perfection presents an array of enormous challenges. In nanoindentation, when an external force is applied through the indenter, achieving a critical stress, it results in the nucleation, movement, and interaction of new dislocations with pre-existing structural defects [19]. This process creates a larger stress field, leading to various types of microcracks, such as radial cracks, lateral cracks, and median cracks [20]. Among all of these, radial cracking is the most frequent type, which extends outward from the corners of the indentation and is more common with sharp indenters like Vickers or Berkovich tips as shown in the figure 2 [21].

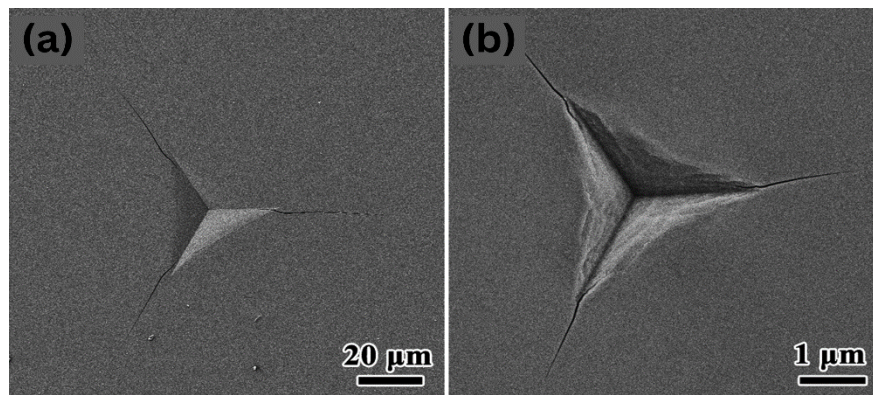


Figure 2: SEM images of radial cracks around Berkovich indentation at different depths - (a) 20 μm and (b) 1 μm [21].

In contrast, lateral and median cracks occur beneath the indenter and extend underneath the surface [22]. To overcome these challenges through the application of both ex-situ or in-situ techniques is not easily feasible due to several challenges. Ex-situ nanoindentation involves observing cracks after each indentation, is not a viable or feasible solution [23]. In contrast, in-situ nanoindentation is useful in extracting real-time analysis and observing cracks during indentation but due to a range of challenges such as instrumentation costs, setup complexity, environmental control, real-time monitoring, sample preparation, safety protocols, data interpretation, training, maintenance, compatibility, resource allocation, and risk mitigation, the utilization of this advanced technique is itself challenging [24].

1.6. Cutting-Edge AI Techniques

In an age consumed by novelty, the advancements in artificial intelligence methodologies shed light on our path, providing an opportunity for advancement as well as a paradigm change in how we understand, adapt, and thrive in the digital era. Therefore, a machine learning (ML)-based model is needed to provide a suitable solution that could eliminate the need for both ex-situ and in-situ nanoindentation [25]. Consequently, the primary aim of this study is to utilize nanoindentation data actively for the purpose of making predictions about crack formation, effectively addressing this gap in the present research.

1.7. Problem Statement

In the segmentation procedure for slicing GaAs wafers, nanoindentation has been implemented to generate indentations at a distance proportional to the thickness of the chip. However, a significant challenge has emerged in the form of a cracking phenomenon that has been observed throughout this procedure. In order to overcome this problem, an AI-based ML approach helps to predict the cracking phenomenon based on the nanoindentation data.

1.8. Proposed Methodology

An approach is to develop a rigorous methodology that is centered on a comprehensive and methodical strategy that is specifically designed to tackle the challenges in order to identify the microcracks initiation. This section presents an in-depth overview of the complete procedure, including data extraction, the development of a machine learning model, the optimization of hyperparameters that follow, and the creation of a graphical user interface. Several ML models, such as Ensemble Learning Tree (ELT), Gaussian Process Regression (GPR), Support Vector Machine (SVM), and Decision Tree (DT), are developed for the prediction of crack formation based on nanoindentation data. Figure 3 depicts an overview of using nanoindentation data for training models for crack prediction. Basically, the use of load (mN) and displacement (μm) data as input parameters for these ML models. The application of data-driven modelling techniques led to the building of optimized models for process validation and prediction. In the subsequent step, the formulation of particle swarm optimization (PSO) and genetic algorithm (GA)-based ML models optimize the parameters that are effective for predicting

and evaluating cracking phenomena using nanoindentation data [26]. This integrated approach, which combines the use of ML models with GA and PSO tuning, has significant potential for experimental crack prediction research employing ML models. Finally, a graphic user interface will be developed using MATLAB, integrating both optimal features and ML algorithms to enable the prediction of nanoindentation-based crack occurrence. This study offers ease to researchers in terms of both reduced time consumption and efficient resource utilization through a trained ML model based on nanoindentation data.

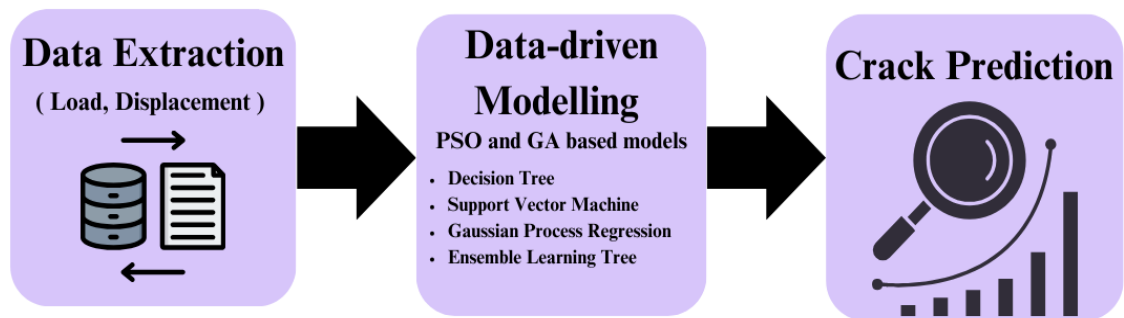


Figure 3: Schematic of the Workflow for Crack Prediction Utilizing Machine Learning.

1.9. Objectives

- Data collection from the relevant literature.
- Development of Machine Learning models.
- Optimization of ML models' hyperparameters using PSO and GA.

Chapter 2:

Literature Review

2.1. Nanoindentation

As a result of the fact that materials exhibit distinct behavior at the nanoscale, so, nanoindentation is an essential technique for determining the material's mechanical properties. [27]. It is an indentation test in which the dimension of the penetration is quantified in nanometers. Machines can precisely record small loads and movements in the form of load vs displacement data as depicted in the figure 4, and determine the modulus and other mechanical properties of thin films and coatings such as, hardness, yield stress, complex modulus, strain rate sensitivity, and fracture toughness [28].

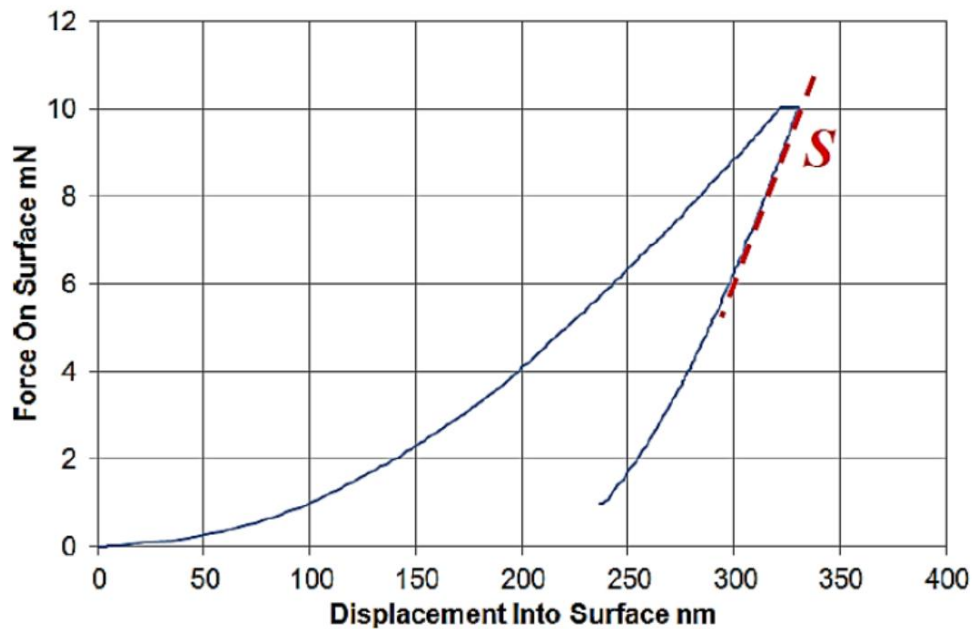


Figure 4: Load-Displacement Curve during Nanoindentation [29].

This is important in fields like microelectronics, where the performance and durability of thin layers are very important to understand [30]. It is not only limited to these properties, but the viscoelasticity, creeping, phase transition, dislocation movement, strain hardening effect, and residual stress, can also be determined [31]. It studies the effects of surface

treatments such as ion implantation and chemical doping. Moreover, it has a wide range of applications in the biomedical industry, including bone and dental materials' analysis [28]. The tool and its cross-sectional representation of the G200 Nano-indenter, are depicted in Figure 5.

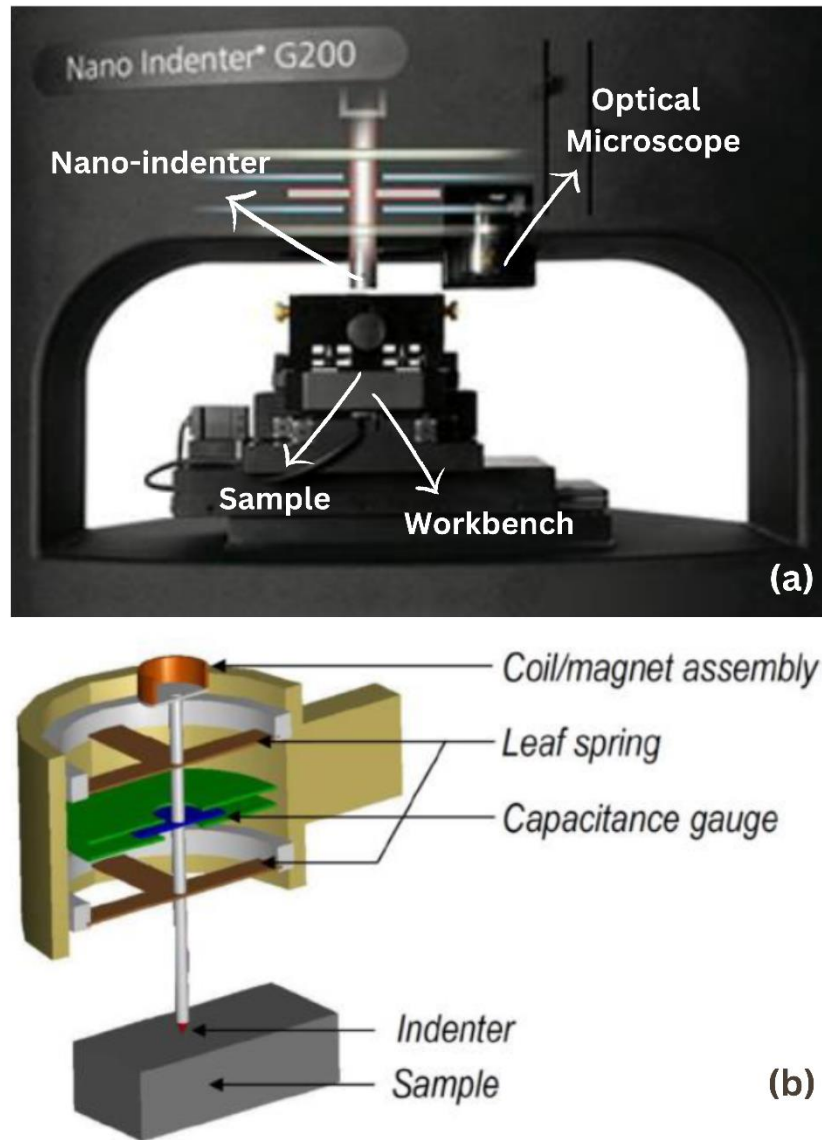


Figure 5: (a) Nano-indenter G2000, (b) Cross-sectional View of G200 Nano-indenter [32].

2.1. Types of Nano-indenters

Nanoindentation uses a variety of indenter types, primarily wedge-, spherical-, and pyramid-shaped indenters. Typical pyramidal indenters include the cubic-angled indenter, rectangle pyramidal Vickers indenter, triangle pyramidal Berkovich indenter, and Knoop indenter [33]. Each indenter has its own specific applications and limitations. As an illustration, the Berkovich indenter is a suitable instrument for quantifying hardness and elastic modulus due to its well-defined geometry and three-sided pyramidal shape [34]. The conical indenter is advantageous in particular situations involving materials that exhibit non-linear properties, owing to its conical shape. The spherical indenter, which is constructed in the shape of sphero-cones to facilitate mounting, exhibits negligible initial contact stress [34]. As a result, they are appropriate for evaluating flexible materials and simulating contact damage that may occur in use. Nevertheless, the attainment of submicron-scale spherical diamond indenters' superior quality poses a significant obstacle to wider implementation. In addition, cube-edge indenters have a pyramidal shape with a minutely cleaved corner and are used for the micro-indentation of the soft materials [35].

2.2. Indentation Methods

The nano-indenter is a nanoindentation device that operates in two modes: continuous stiffness measurement (CSM) and quasi-static loading [36]. The quasi-static loading method gives significant insight into the behavior of materials under gradual constant force application for determining hardness and elastic modulus under static conditions. When substances are subjected to continuous, long-lasting stress, this mode becomes critical. The CSM mode, developed by MTS Systems Corporation, offers an alternative approach to dynamic nanoindentation. The process involves applying a low-amplitude signal to the indenter while it is advancing through the material. This methodology enables simultaneous assessment of hardness and elastic modulus across multiple depths within one single indentation [33]. When examining gradient differences in mechanical properties across the depth of thin films, coatings, or layered materials, this feature is particularly useful. Moreover, when analyzing the viscoelastic features of materials that show both elastic and viscous properties when deformed, the CSM mode is especially beneficial. Researchers are able to gain in-depth information into the behavior of these materials in real-world scenarios by analyzing the way in which they respond to varying

stresses over time, due to the dynamic nature of CSM. The CSM mode may also adjust the strain rate indefinitely, allowing for consistent evaluation of materials under comparable strain rate settings and research into strain rate sensitivity in sectors such as aerospace and automotive. In addition, the CSM mode simplifies the correction of the indenter's area function, which is crucial for precisely identifying material qualities [37]. Overall, the nano-indenter provides extensive material characterization capabilities, with the quasi-static loading mode giving critical data on material reactions under constant load and the CSM mode providing dynamic testing capabilities.

2.3. Pop-in Events in Load-Displacement Curve

Nanoindentation provides load-displacement data as output, demonstrating the material's behavior, which changes based on alterations in the material or its characteristics [38]. Pop-in events in the load-displacement curve represent the material's response at different applied loads. A pop-in event is defined as an abrupt displacement bursting that is detected

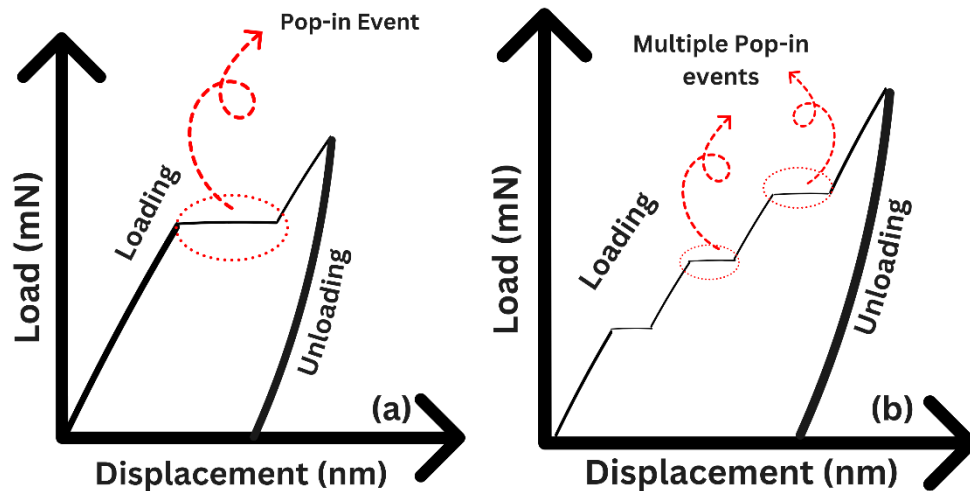


Figure 6: (a) First Pop-in Occurrence within the Load-Displacement Curve, (b) Multiple Subsequent Pop-in Events.

during a nanoindentation test under a constant load condition. This event happens due to variations in the properties of the material beneath the surface. [39]. These pop-in occurrences are attributed to various factors. For example, the very first pop-in event is typically the result of the elastic-to-plastic transition, as illustrated in Figure 6(a).

Additionally, multiple pop-in events can be caused by various factors such as phase transformations, dislocation nucleation and movement, cracking phenomena, and the transmission of dislocations from one grain to an adjacent grain. Figure 6(b) visually represents the occurrence of multiple pop-in events. These events serve as significant sources of information regarding the mechanical characteristics of materials, including hardness, fracture toughness, and elastic modulus. One prevalent factor contributing to the occurrence of pop-ins is the cracking phenomenon, which can manifest in diverse forms, including radial cracking, lateral cracking, and median cracking, among others [34].

2.4. Cracking in GaAs

This particular segment is dedicated to a comprehensive analysis of the characteristics and occurrence of the cracking phenomenon in GaAs. The cracking phenomenon ensures that a pop-in event will occur in the load vs displacement curve during nanoindentation [40]. The goal of this study is to forecast the initiation of microcracks in order to streamline the segmenting process of GaAs wafers, that's why it is an essential to discuss this section separately. This specialized inquiry not only emphasizes the importance of understanding the cracking characteristics of GaAs but also enables a focused analysis that provides subtle insights that are crucial for applications where material reliability and crack resistance are of the utmost importance. However, it is critical for understanding crack morphology and the manufacturing of high-quality GaAs-based devices, which are related to the physical features, appearance, and form of the microcracks in the material. As a result, nanoindentation is a well-known technology for investigating cracking evolution and material mechanical properties [31]. It describes the crack's formation, propagation, and interaction with pre-existing defects in the material. Many research and studies have previously been addressed to the investigation of indentation on GaAs and crack formation. The major purpose is to comprehend the distribution of radial cracks in various crystallographic orientations [41]. For example, the crack formation appears anisotropic when a sharp indenter is used to indent GaAs with varying doping conditions. Because of the radius of the indenter, the pop-in stress is greater for the spherical indentation than for

the sharp indentation. These insights are essential for material characterization and understanding mechanical behavior at a small scale.

2.5. Crack Observing Techniques

Cracking analysis can give significant insights into the mechanical behavior and failure mechanisms of materials. There are two distinct methodologies have been developed to examine the structure and behavior of cracking in GaAs: (1) controlled cleavage cross-sectioning at specific locations within the indentation zones; and (2) in-situ nanoindentation with continuous indentation load and displacement recording. [14]. The first approach is used to precisely characterize the orientations, shapes, and sizes of subsurface indentation cracks [42]. The second approach involves combining surface cracking observations with load-displacement data to determine cracking dynamics [43].

2.5.1. Ex-Situ Study

Ex-situ studies involve the segregation of the indentation process from the statistical procedures, which provides a greater degree of experimental flexibility and control. Researchers perform indentation in a specific environment or under specific conditions and later examine the effects, such as deformation, cracking, or phase transformations, using various analytical tools. Numerous researches have been conducted to understand the crack morphology using ex-situ nanoindentation. For example, Pharr et al. [44] successfully developed the most straightforward methodologies for quantifying the mechanical properties of thin films using nanoindentation. They extracted information by the impression of the indentation and utilized it in finding the characteristics of films. Using a nanoindentation test, Xia et al. [45] find out the influence of surface roughness on the mechanical property of a material. Moreover, Rodriguez et al. [46] devised an innovative technique utilizing instrumented indentation to determine the cohesive-frictional mechanical properties of amorphous materials. Stephania et al. [47] utilized deep learning algorithms to investigate pop-in identification in nanoindentation curves. The objective of this study was to create a binary classifier based on image detection that could accurately distinguish between the presence and absence of pop-in events in load-displacement curves obtained from nanoindentation experiments. Moreover, Baiocco et al. [48] have recently applied ANNs and other ML tools to the prediction of yield stress,

indentation curves, and the volume fraction of precipitated secondary phases for duplex stainless steel grade 2205. Similarly, Yescas et al. [49] also investigated the prediction of the Vickers hardness of austempered ductile metals in relation to their composition, austenitizing and austempering temperatures and durations, and in the design of such irons. In addition, Windsor et al.[50] conducted research on the relationship between heat treatment conditions, density, and residual stresses in powder metallurgy steels. Through this investigation, she developed a model that effectively predicted the yield stress of irradiated ferritic steels.

2.5.2. In-Situ Study

In the fields of materials science and nanotechnology, in-situ nanoindentation is a technique whereby nanoindentation investigations are performed under identical conditions or surroundings that are utilized for subsequent analyses or observations. In contrast to ex-situ nanoindentation, which involves the separation, preparation, and subsequent analysis of samples, in-situ nanoindentation enables the continuous monitoring and characterization of material responses while they are undergoing indentation. Basically, the examination of discontinuities in the loading and unloading curves prove beneficial in identifying discrete events that happen within the material during experimentation. However, ascertaining the deformation mechanisms responsible for these discrete events is inherently speculation due to the impossibility of observing the actual processes during testing.

2.5.2.1. Cracking Sequence

A significant number of researches have been conducted using in-situ nanoindentation, providing a unique understanding of the behavior of materials. This includes phenomena such as crack initiation and propagation, pile-up, elastic and plastic transitions, and sink-in. This understanding is obtained by combining nanomechanical testing with direct observation via electron microscopy. The primary goal of this research is the investigation of the cracking phenomenon; therefore, comprehending real-time microcrack initiation is crucial to understanding. The sequence of cracking observed in Figure 7 is as detailed below: (i) A preliminary set of radial cracks (1) aligned with the [100] or [010] orientations are initiated by early loading. These microcracks are also known as Palmqvist cracks; (ii)

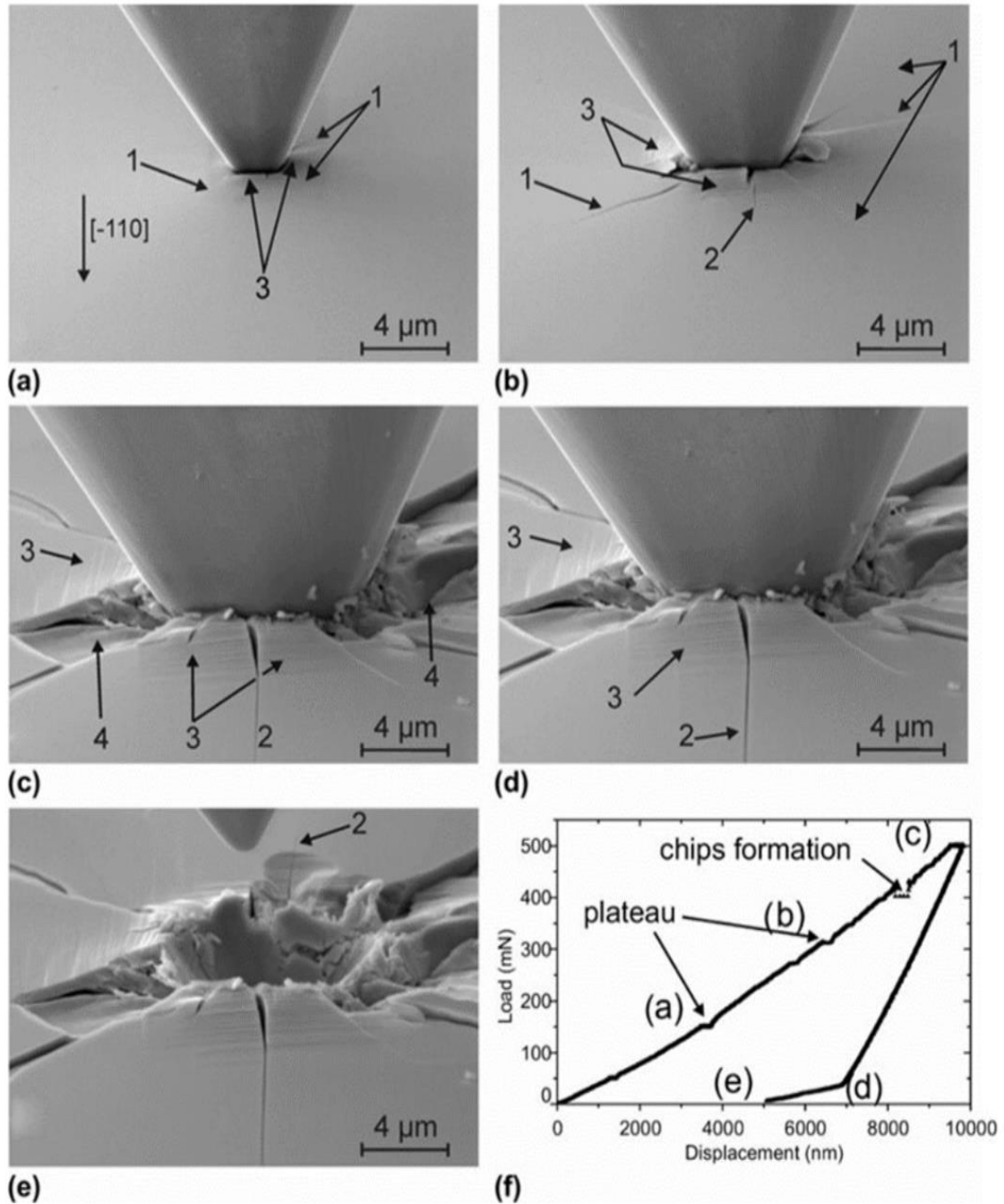


Figure 7: Sequential micrographs illustrating the cracking process with a conical 60° indenter under varying loads, captured during in situ nanoindentation: (a) Loading at 150 mN, (b) Loading at 300 mN, (c) Unloading at 50 mN, (d) Full load at 500 mN, (e) Unloading complete with indenter withdrawal, and (f) Corresponding Load–Displacement Curve. The arrows (1) and (2) indicate radial cracks, while the arrow (3) and arrow (4) indicate slide lines and fragments, respectively [14].

they undergo a transition, after which a subsequent set of radial cracks (2) aligned with the [110] or [010] directions form and expands upon the initial set. The microcracks under consideration are commonly known as half-penny cracks. (iii) Following this, chip formation (4) takes place either at the surface during loading or unloading. [14]. The aforementioned complementary methodologies provided significant insights into the sequence of cracking and the morphology of the cracks in the indentation area.

2.6. Artificial Intelligence

Artificial Intelligence (AI) is a specialized domain within computer science that aims to create systems capable of outperforming human intelligence across a range of tasks, such as comprehending natural language and solving problems. ML is a subfield of AI,

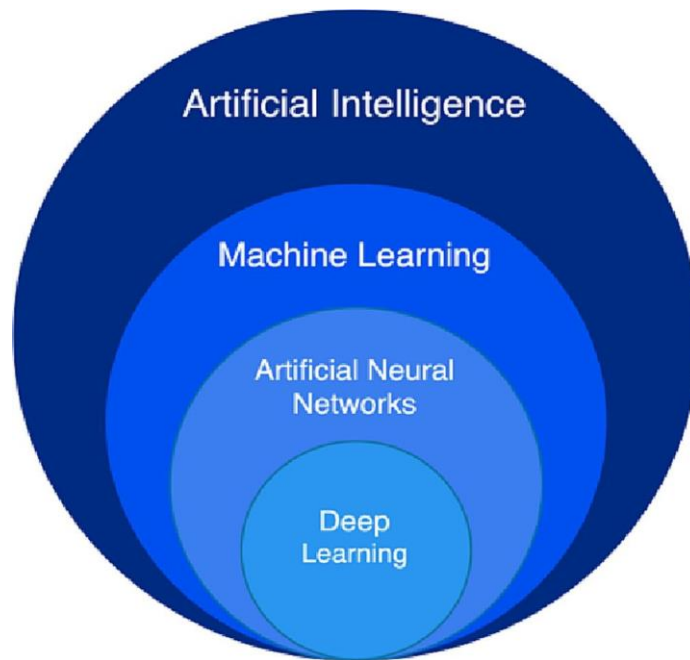


Figure 8: Visualization encompassing key elements of AI. ML, DL, and ANN [51].

concentrates on the development of various models that enable machines to improve their performance over time through data-driven learning. Deep Learning, a significant ML technique, makes use of multilayered neural networks that are especially effective at speech and image recognition [51]. Natural Language Processing facilitates computers in comprehending and generating human language, while Computer Vision interprets visual data. A comprehensive representation of AI and its subfields is mentioned in the figure 8

[51]. AI finds utility in a multitude of fields, encompassing expert systems, natural language generation, and robotics. General AI works to emulate human intellect more comprehensively, while narrow AI is specifically engineered to execute designated tasks. [52]. It is of critical importance to the development of intelligent systems and automation, influencing sectors ranging from finance to healthcare. Our data is based on numerical values, and, ML is particularly well-suited for handling numerical data, often excels in tasks that involve quantitative information. Therefore, it is better to use ML approach in order to develop a model for the prediction of cracking phenomenon. Moreover, ML is a purposeful endeavor rooted in the acknowledgment of its capacity for revolutionization. It emerges as a catalyst for revealing valuable insights, automating complex processes, and enhancing decision-making capabilities in a world filled with data. Exploring the domain of ML provides individuals with the necessary skills to effectively navigate the complexities of data analytics, predict future trends, and derive practical insights [53]. A detailed overview of machine learning is addressed in the subsequent section.

2.6.1. Machine Learning

A branch of artificial intelligence (AI) called machine learning (ML) is devoted to developing statistical models and algorithms. Without being explicitly programmed, these models enable computers to acquire knowledge independently and generate predictions or decisions [54]. ML techniques are used to analyze and interpret data, discover patterns, and make informed decisions based on the data [55]. Here's an overview of ML and its methods. It operates on the principle that computers can learn from data and improve their performance over time [56]. The key components of ML are as follows:

Data: ML algorithms need data to train itself with the help of patterns and make predictions. High-quality and relevant data is crucial for successful machine learning.

Features: Features are the specific characteristics or the columns of the input variables in the data that the ML algorithm uses to make predictions. Feature selection or engineering is an important step in ML.

Model: The purpose of a model is to depict the relationships between the objective variable and the data features through a mathematical representation. By means of training

on past data, the model acquires the capability to generate predictions on novel, unobserved data.

Training: During the training phase, the ML model learns from historical data by adjusting its internal parameters to minimize prediction errors.

Testing and Evaluation: After training, the model is tested on new data to assess its performance and accuracy. Various evaluation metrics are used, depending on the problem type (e.g., classification, regression) [57].

2.6.2. ML Models

ML comprises an extensive array of methodologies and strategies. The method selection is determined by the nature of the data available and the issues at hand. Total of four ML models SVM, GPR, ELT, and DT were used to train and forecasting the microcracks using load vs displacement data in MATLAB. Detailed descriptions of each of these models are presented in the subsequent sections:

2.6.2.1. Decision Tree

Decision trees are considered a fundamental technique in discriminant analysis for the purpose of knowledge discovery. The efficiency, promptness of operation, and uncomplicated implementation of the system all contribute to its elevated status. In this technique, an extensive arrangement of nodes and branches constitutes a tree-like structure. The division of individuals into groups is achieved through the utilization of specific functions for regression or classification by internal nodes that possess outgoing edges. In addition, values may be maintained at the terminal nodes or leaves or computed, depending upon the type of model [58]. The most basic form of regression tree is represented by a constant value at the root level, which is usually the mean value of the target attribute. In order to expand upon this notion, model trees substitute linear or nonlinear regression functions for these constants. A model proceeds from the root to a leaf to predict a value, with the instance's attributes guiding the path at each internal node. Then, the prediction is generated by the leaf's regression model [59]. The decision tree algorithm, which is composed of samples provided, is the core of the decision tree's functionality. By minimizing a fitness function, this algorithm attempts to locate the

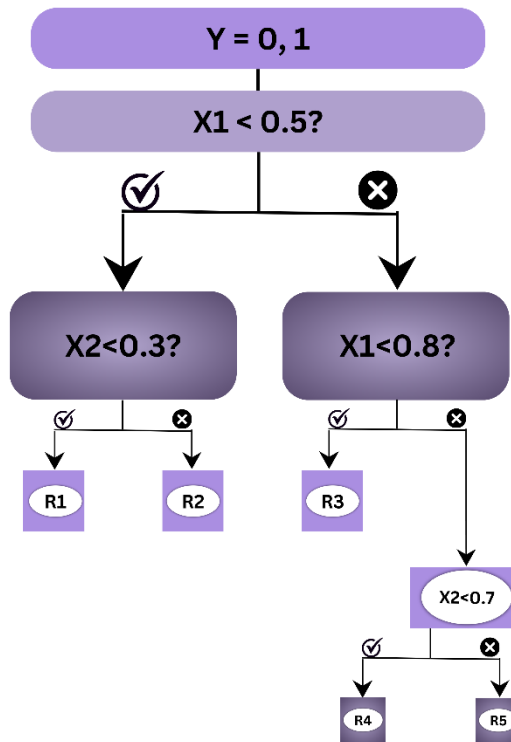


Figure 9: Representation of Decision Tree Structure.

optimal tree. When classes are not predefined, a regression model is fitted to the objective variable by utilizing independent variables. For each variable, the dataset is partitioned into multiple positions. The algorithm evaluates the variation between reported and predicted values at each division, utilizing the fitness function as its basis. The procedure proceeds recursively, selecting the division point that minimizes the error, as mentioned earlier [60]. Decision trees are effective when they are capable of processing complex datasets and generating results that are easily interpretable. Due to their capacity to streamline decision-making procedures and predict results satisfactorily, and they find extensive application across diverse domains, including finance, healthcare, and machine learning. A graphical depiction of the complete operational process of the DT algorithm is illustrated in Figure 9 [61].

2.6.2.2. Support Vector Machine

Support Vector Machines (SVM) is a supervised machine learning method that excels at both classification and regression. [62]. It manages the relationship between a output variable and one or more predictors as an extension of the SVM algorithm. The SVR algorithm acquires knowledge of a regression function that establishes a correspondence between input variables and observed response values through the formulation of an optimization problem [26]. SVR is renowned for its ability to precisely balance model complexity and prediction accuracy, which makes it exceptionally proficient in the management of high-dimensional data. This equilibrium is maintained by employing several fundamental principles derived from SVM and refined for regression. (a) the soft margin; (b) the kernel function; (c) the separating hyperplane; and (d) the maximum-margin hyperplane. In SVM, the separating hyperplane is a crucial concept, establishing a boundary to distinguish between classes in classification tasks. The maximum-margin hyperplane as mentioned in the figure 10, which enhances classification robustness by optimizing this boundary to maximize the distance from the closest data points of various classes. The soft margin approach further refines SVMs by accommodating real-world data imperfections and balancing margin width against misclassification levels to increase the model's generalizability. Additionally, both SVM and SVR depend heavily on the kernel function, a fundamental tool that addresses nonlinear challenges by transforming the input variables into a higher-dimensional space, thereby enabling linear discrimination

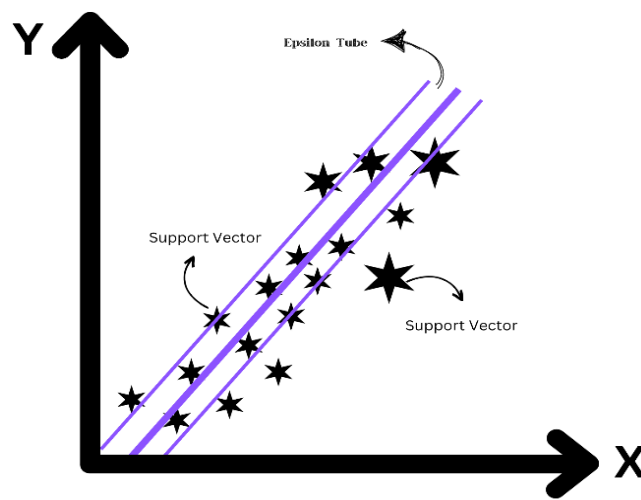


Figure 10: Representation of SVM Structure.

in more complex situations. Unlike SVM's focus on optimizing class margins, and fits a hyperplane that most data points are within a specified distance (ϵ -tube) in order to minimize the error. Whereas support vectors are critical in determining the position of the hyperplane, points that deviate from this tolerance are penalized [63]. Because of its methodology, SVR is an effective predictive analytics tool that can extract valuable insights from large, complicated datasets in a variety of industries, including environmental modeling, healthcare, and finance [64].

2.6.2.3. Gaussian Process Regression

GPR in the figure 11 is a highly adaptable supervised learning methodology that is utilized for both classification (with discrete outputs) and regression (with continuous outputs). Its utility covers an extensive range of situations, including the analysis of data sets and integration into more complex problem-solving procedures [65]. GPR is built upon the Gaussian process, which is a collection of random variables of any finite number that share a jointly consistent Gaussian distribution. The utilization of this statistical framework enables the imperfect prediction of continuous values through GPR [66]. GPR is a unique method because it is not linear or parametric. This makes it easier to connect data points in complex, high-dimensional spaces. GPR operates on the foundation of Bayesian probability theory, which facilitates the seamless integration of observed data and prior knowledge in order to produce predictions. GPR works well with other Bayesian framework regression methods, especially Kernel Ridge Regression (KRR) and linear

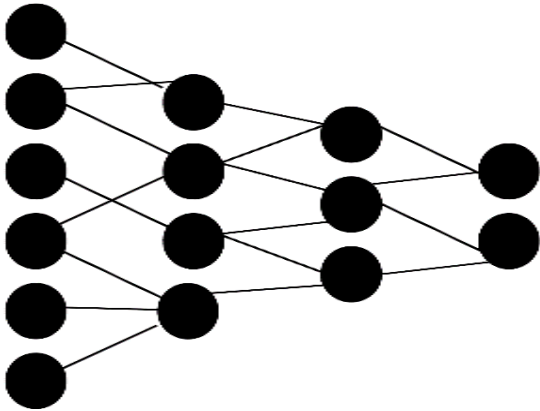


Figure 11: Representation of GPR Structure.

regression using radial basis functions. This exemplifies the adaptability and robustness of the framework in handling various predictive modeling scenarios [67]. The Bayesian approach, within the domain of neural networks, utilizes a prior distribution over weights that is analogous to a distribution over functions. A posterior is produced for prediction functions when this initial assumption is combined with a noise model. However, approximations are often necessary for practical implementations due to the intricate nature of this function distribution in neural networks [68]. The management of uncertainty and model complexity by the Bayesian framework of GPR differs from that of techniques like KRR. A comprehensive understanding of these differentiations is crucial for the selection of the most suitable tool for specific ML tasks, especially those that require precise interpolation and quantification of uncertainty.

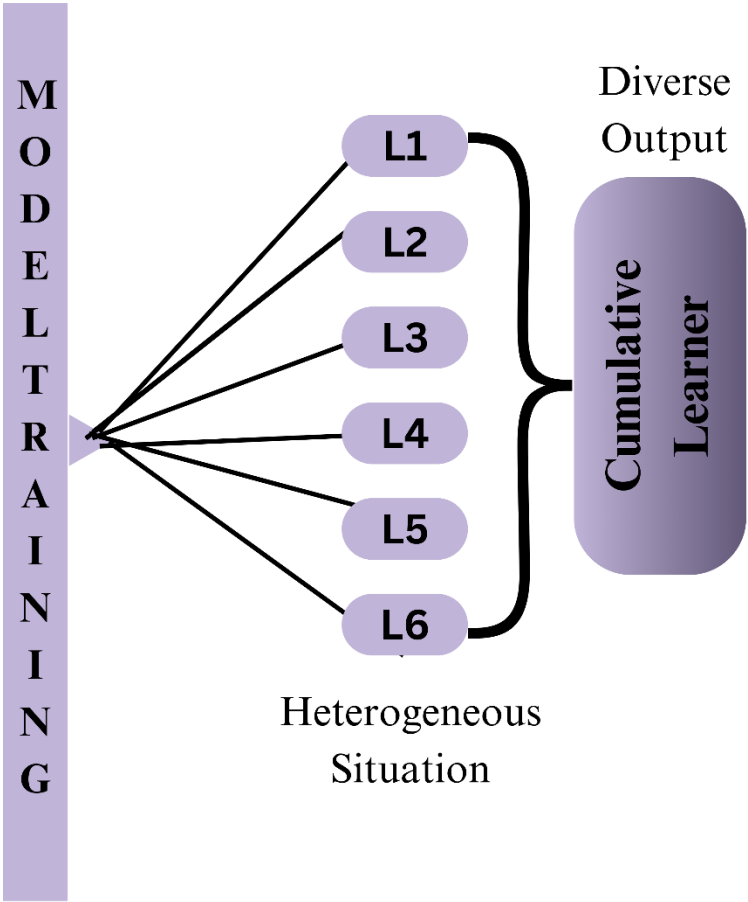


Figure 12: Representation of ELT Structure.

2.6.2.4. Ensemble Learning Tree

ELT methods are distinguished by the incorporation of a large number of learners, which are integrated in a specific manner to enhance the precision of predictions [69]. Tree-based machine learning models are characterized by their transparency and simplicity, which set them apart from earlier, more obfuscator-oriented algorithms. These models are particularly effective at handling both linear and nonlinear problems. The method under consideration is predicated on two essential model categories: meta-learners and base learners. Base learners have the duty of forecasting the posterior class probabilities of a provided sample. On the other hand, the meta-learner determines the final class designation by combining these predictions as visualized in the figure 12 [70]. The implementation of a dual-model approach enables more nuanced data processing, which in turn yields more precise predictions. An aspect that sets this approach apart is its extensive use of multi-objective optimization during the model development phase, placing particular emphasis on the combination and selection of models. This strategy helps to optimize ensemble complexity while minimizing accuracy and simultaneously ensuring precision and interpretability. The ensemble method additionally employs a hill-climbing algorithm to establish a stable collection of diagnostic principles. The method described above is based on strategic prioritization and selection of rules, which ensures that the ensemble delivers satisfactory results and offers valuable insights. This ensemble learning method is a big step forward in the field of ML because it uses base and meta-learners strategically and multi-objective optimization. It's especially useful for making diagnostic tools that are easy to understand [71].

2.6.3. Overview of Optimization Algorithm

A comparative analysis was carried out to identify significant features and optimize the hyperparameters with the help of two distinct optimization algorithms. The Genetic algorithm comes first, followed by Particle Swarm Optimization.

2.6.3.1. Genetic Algorithm

Generating computational algorithms is a technique that draws inspiration from the principles of natural evolution. It identifies the optimal solution from a set of potential solutions to a given problem. This methodology involves conceptualizing every possible

resolution as a "chromosome" and evaluating its excellence through the utilization of a specialized metric called the "fitness function." [72]. This fitness function is employed by GA in order to quantify the efficacy of every proposed solution. The algorithm iteratively evaluates these solutions and terminates execution when it identifies an adequate solution that satisfies the predetermined criteria. In the event that the intended solution remains unattainable, and GA proceeds to generate alternative solution candidates through processes including selection, crossover, and mutation, which collectively comprise the subsequent generation of possibilities [73]. During this iterative procedure, solutions that exhibit exceptional performance are retained, whereas those that fall short of perfection

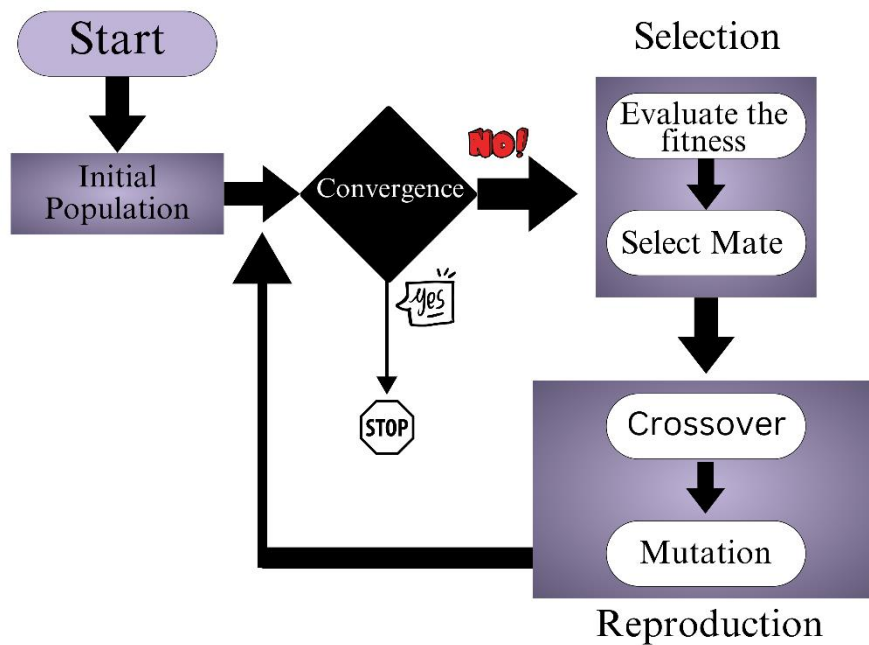


Figure 13: Representation of workflow of GA.

are discarded. As this process goes on, solutions that work well are kept, and the others are discarded. This way, over time, GA helps in finding ever-improving solutions to that problem. The key is to use these operations wisely to improve the solutions until the best one is found. Figure 13 visually describe the whole process effectively [74].

2.6.3.2. Particle Swarm Optimization

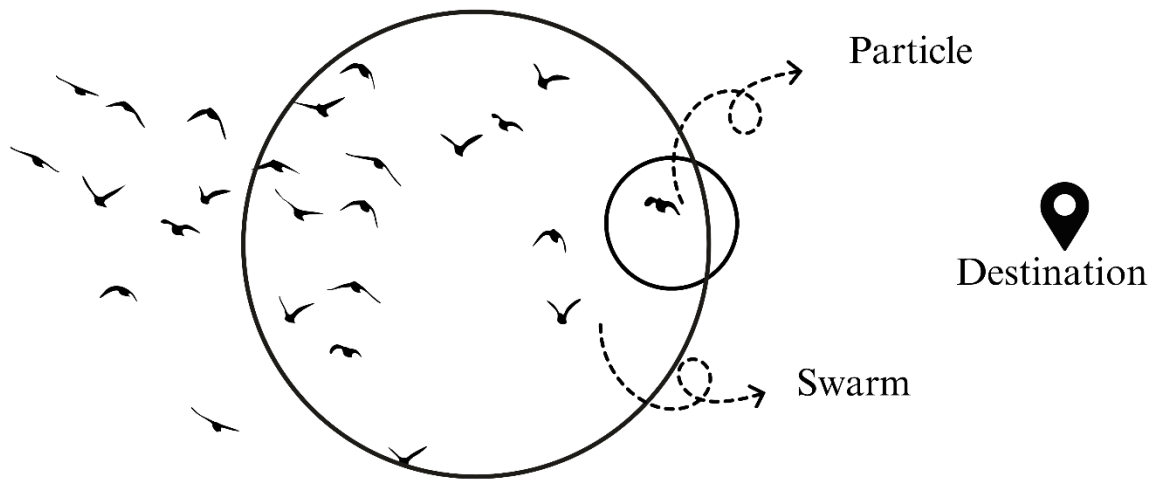


Figure 14: Nature inspired Technique (PSO).

PSO is a sophisticated optimization method that is characterized by its stochastic, self-adaptive, and population-based attributes as described in the figure 14. Particles are utilized to symbolize potential solutions within a specified search space in this methodology. Every individual particle establishes its own optimal position and function value by taking into account its present velocity and the optimal positions of its associates. This assessment guides the particles as they adjust their positions and velocities, guaranteeing that they approach the optimal solutions. By implementing regular adjustments to the positions, velocities, and adjacent interactions of particles, it is possible to constrain them to predetermined boundaries [75]. PSO operates on the fundamental principle of being capable of traversing vast solution spaces in pursuit of optimal values for various system characteristics, all the while considering cost limitations. This approach is widely employed across diverse scientific fields to tackle challenges that require optimal solutions while effectively controlling expenses. In PSO, every solution can be likened to a soaring particle or bird traversing the search space. These particles determine the optimal positions in concert as a swarm. In this multivariate space, particles are associated with position and velocity vectors. It is imperative to comprehend the relationship between the position and movement pattern of a particle, as this determines the limits that particles can

traverse. This approach has been useful in clarifying complex motion patterns and improving our comprehension of optimization challenges in a variety of scientific

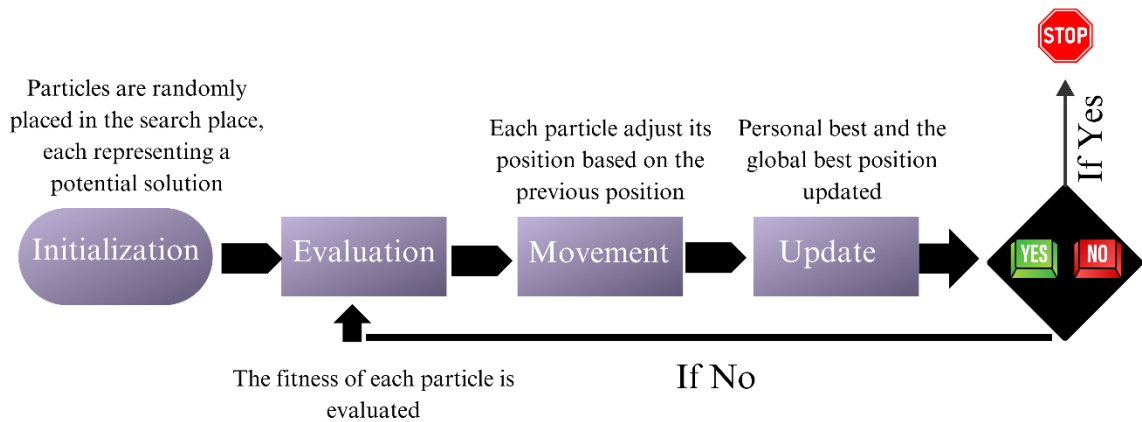


Figure 15: Workflow of PSO.

disciplines. It is imperative to acknowledge that the adaptability and collaborative nature of PSO among particles render it an excellent instrument for tackling practical challenges, guaranteeing effective resolutions while accommodating the intricacies included in various scientific investigations. Figure 15 illustrates the comprehensive visual representation of the PSO algorithm's operation [76].

Chapter 3:

Methodology

3.1. Data Collection

A detailed investigation of the existing literature unveils an all-encompassing investigation comprising a diverse collection of scholarly articles. A detailed review was conducted using reputable databases (Google Scholar, Web of Science, and Science Direct), incorporating key terms such as, nanoindentation, pop-in, crack initiation, and machine learning. This literature review studies multiple aspects of research, with a specific emphasis on the approaches utilized in ex-situ and in-situ. The discussion not only encompasses the wide range of papers that were examined but also thoroughly evaluates

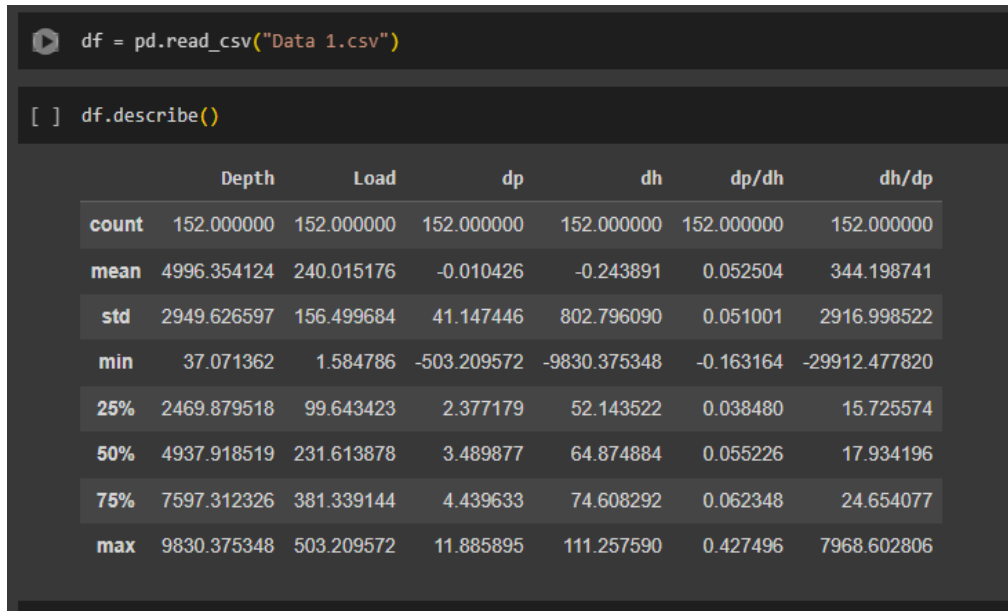


Figure 16: Data Classification Using Python Programming Language

each methodology's intricacies, illuminating the unique advantages and practical uses linked to ex-situ and in-situ strategies. A particular paper was selected due to its significant focus on in-situ techniques, to attain a more complicated comprehension [14]. The justification for selecting this particular paper is based on its credibility and dependability, ascribed to its real-time analysis implementation. In this paper, in-situ nanoindentation is

utilized for the analysis of the cracking phenomenon, which offers a singular and invaluable perspective. The findings are strengthened by the emphasis on real-time analysis, which contributes significantly to the wider discussion on methodologies for material analysis and characterization. After the successful selection of this prestigious paper, data points were carefully extracted from the graphical image and prepare for the subsequent steps. Secondly, these data points were classified using the python language by loading the data file in the CSV format as shown in the figure 16. A number of algorithms were executed on the provided data to attain the data insights in terms of box plots, pair plots, and Pearson correlation.

3.1.1. Overview of Python Plotting

Python plots have a significant impact on data exploration and analysis due to their ability to convert unorganized datasets into informative narratives. Python plots facilitate the discernment of correlations, trends, and patterns within the data through the utilization of visualization capabilities; they provide a dynamic lens through which information is brought to life. This functionality not only improves our comprehension of complex datasets but also enables us to convey discoveries in a persuasive and easily comprehensible manner. Various plots, such as box plots, pair plots, and Pearson correlation, can be obtained from the Python language [77].

3.1.1.1. Box Plots

Box plots in Python, also known as box-and-whisker plots, are highly effective graphical depictions employed to illustrate crucial statistical information concerning the distribution

```
#box plots
sns.boxplot(data=df[["Depth","Load","dp
","dh","dh/dp","dp/dh"]], orient="h")
```

Figure 17: Algorithmic Representation of Box Plots

of a provided dataset. The key insights derived from box plots include: outliers, skewness, symmetry, variability, and central tendency. A following algorithm in the figure 17 is required to attain the box plots on the basis of provided data sets [78].

3.1.1.2. Pair Plots

Pair plots, in Python often created with libraries like Seaborn provide a visual way to explore the relationships between variables in a dataset. The histograms on the show the distribution of each variable individually helping us understand their unique characteristics. Scatter plots on the display bivariate relationships allowing us to easily identify correlations, clusters and potential outliers. These diagrams serve as a tool for understanding how variables are interrelated and how their distributions are shaped. They simplify the analysis of datasets and offer guidance for further investigation. The process, for generating these plots is illustrated in Figure 18 [79].



Figure 18: Algorithmic Representation of Pair Plots.

3.1.1.3. Pearson Correlation

The Pearson correlation coefficient, abbreviated by the symbol "r," finds out the direction and intensity of a linear relationship between dependent and independent variables. A correlation of +1 indicates an ideal positive linear association on a scale from -1 to +1, while a correlation of -1 shows an ideal negative linear relationship; and a correlation of 0 implies the absence of any linear association. Pearson correlation algorithm as depicted in the figure 19, is used to assess the extent of association between variables, offers valuable insights into the correspondence between changes in one variable and another. It is imperative to acknowledge that correlation does not necessarily imply a connection, and

the measure's validity is predicated on the assumption of a linear association between the variables under investigation; furthermore, it may be susceptible to outliers in the data.

```
corrmat = df.corr()
plt.figure(figsize=(16, 8))
heatmap = sns.heatmap(corrmat,
vmax=1, annot=True, cmap="YlGnBu")
heatmap.set_title('Correlation Heatmap',
fontdict={'fontsize':30}, pad=12);
plt.show()
```

Figure 19: Algorithmic Representation of Box Plots

Different ML models were developed by integrating (1) GA and (2) PSO for the prediction of microcracks. The operational parameters include load and displacement. Before the data was imported from an Excel sheet in MATLAB R2021, the data was well classified in the Python programming language and identified the significance of each attribute with the independent variables. For the prediction of microcracks and feature selection, the optimization algorithms, which included GA and PSO, were integrated with ML models. Apart from that, a machine learning model was trained using the data set that predicted the initiation of microcracks by taking nanoindentation data under hypothetical parameters. In this study, a GPR-GA-based model was selected over all other models to predict crack initiation. Therefore, the workspace of the GA-based GPR model was used for building the GUI that fulfills the main objective of this study [80].

3.2. Data Pre-processing

After successfully classifying and getting visual insights of the extracted data using python language, it is ensured that it is ready for subsequent ML steps. Firstly, all data points were standardized to maintain logical values and prevent any single feature from exerting excessive influence. Careful scrutiny and the removal of unusual data points were

performed to maintain the prediction accuracy. The dataset was then split into two parts: 80% for training our models and 20% for accuracy testing, ensuring fairness through random selection. Thoughtful handling of missing data involved either filling in gaps or removing problematic sections. ML models, SVM, GPR, DT, and ELT, were trained and evaluated using metrics such as coefficient of determination (R^2) and root mean square error (RMSE) through specialized apps. Advanced techniques like GA and PSO were employed to fine-tune our models, and enhanced their predictive abilities significantly.

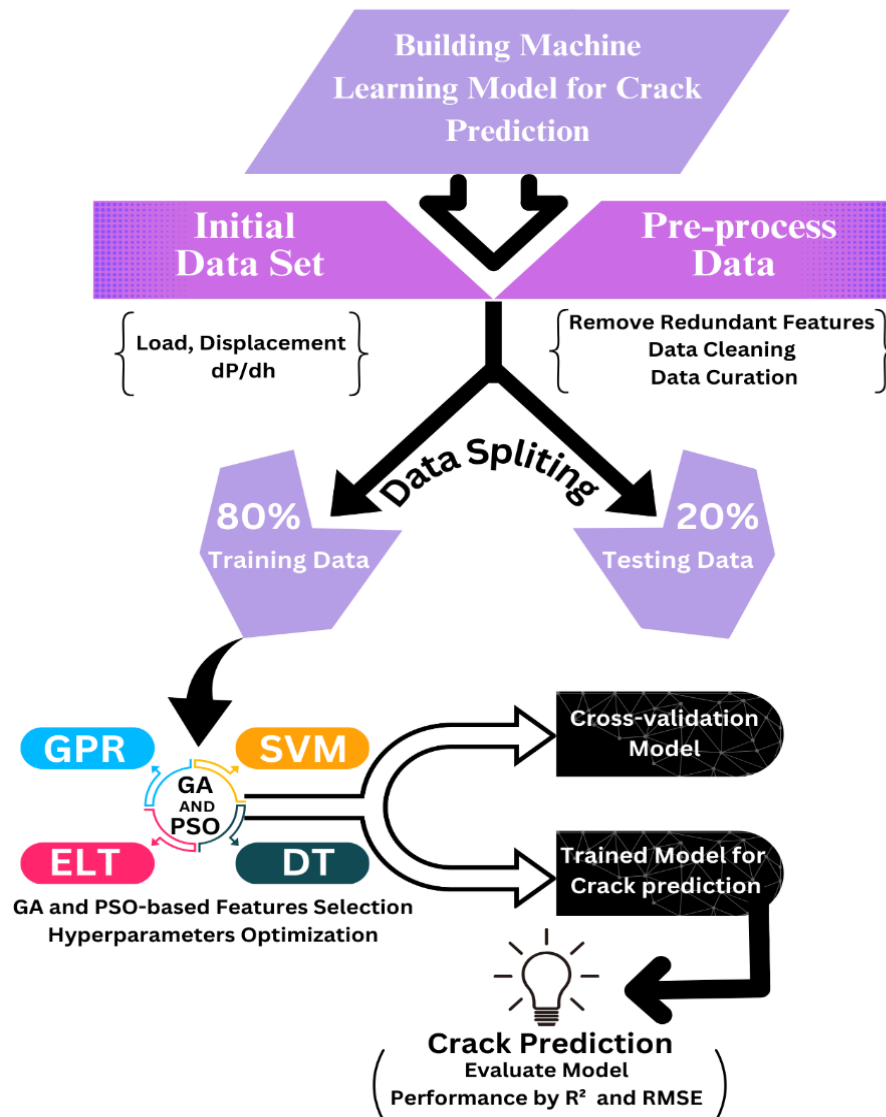


Figure 20: Machine Learning Workflow.

Thorough testing yielded impressive results, with R^2 and RMSE values close to 1 and 0, respectively, indicating the precision of our predictions. This meticulous approach not only ensured accuracy but also optimized the overall performance of our models. Figure 20 visually represents the entire process, from data collection to crack prediction.

Chapter 4:

Results and Discussion

4.1. Box plot, Pair plot, Correlation heatmap

A rigorous classification procedure using the Python programming language in the Google Colab environment is conducted for the classification of a comprehensive

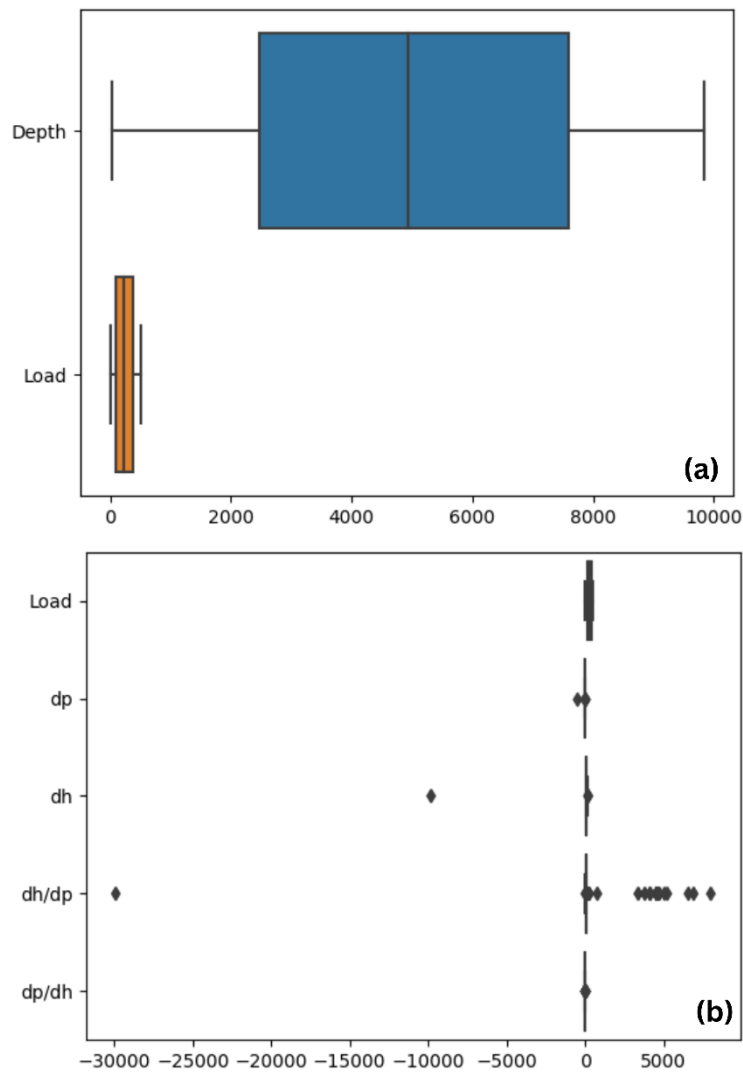


Figure 21: (a) Load (mN) and Displacement (μm) Box Plot representation (b) dp, dh, dh/dp, dp/dh Box Plot

exploratory data analysis using a variety of visualization techniques to obtain an in-depth comprehension of the dataset's complex characteristics. Additionally, box graphs, as shown in Figure 21, proved invaluable for visually summarizing the distribution of data across multiple categories or features. These plots facilitated the identification of median points, potential outliers, skewness, clusters, and spread in the load-displacement and dp/dh data sets, enriching the understanding of the dataset's nuances. The utilization of

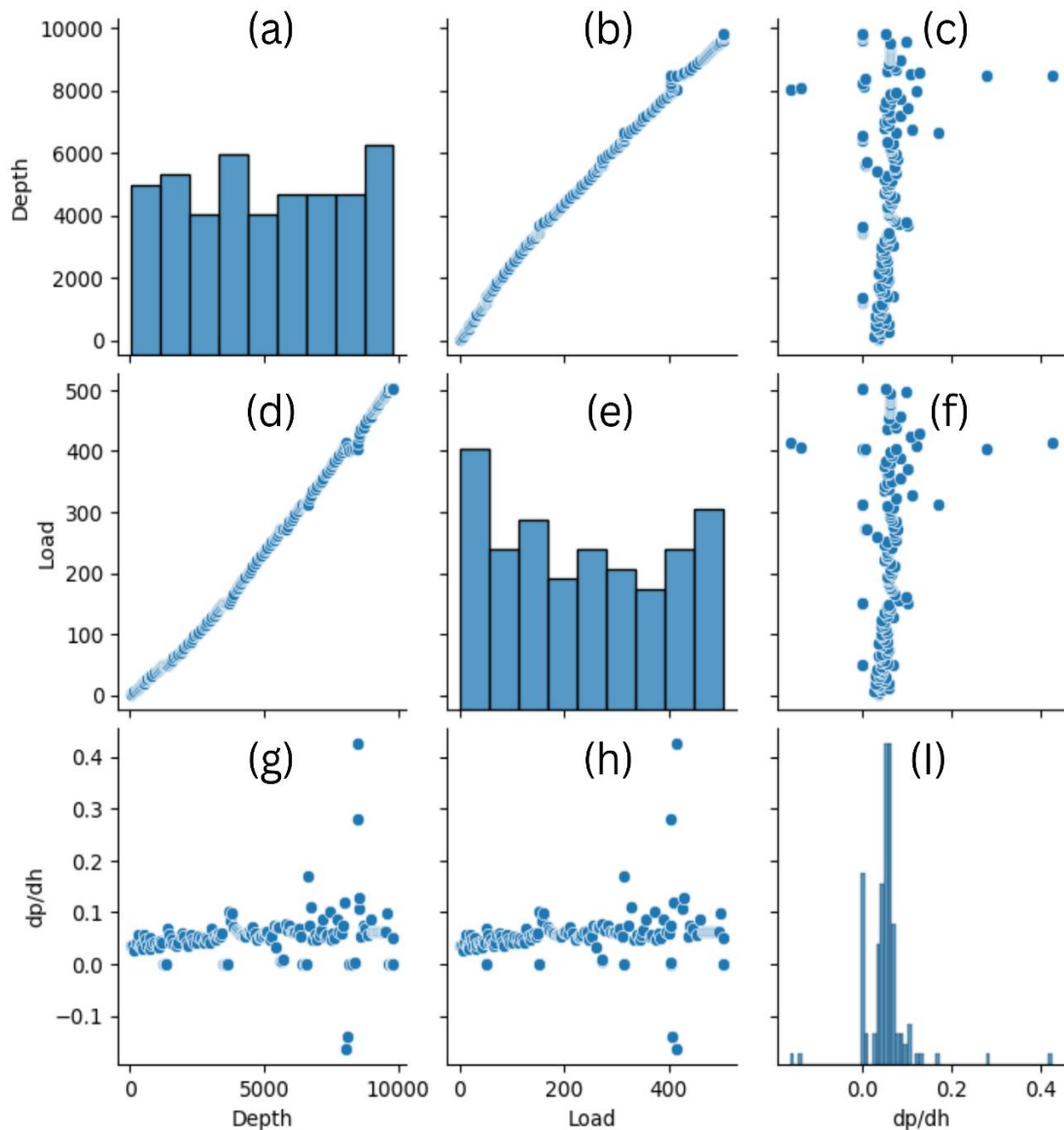


Figure 22: Pair Plot representation, (a), (e), (I), shows the histogram. (b), (c), (d), (f), (g), and (h), indicate bivariate relationships.

pair graphs, which are illustrated in figure 22, was integral to exhaustive analysis. The provided plots offer a graphic representation of data dispersion, recognizable patterns, histograms for individual variables, and an overall comprehension of the interconnections among numerous factors. A careful examination of these diagrams could enable us to discern possible connections, clusters, and correlations among the data points. The linear correlations between the variables in the data were subsequently ascertained using Pearson correlation analysis. It provides the correlation coefficient, and crucial details about the connections between variables such as load, depth, dp, dh, dp/dh, and dh/dp measures the strength and direction of linear correlations. It identifies significant connections through this technique, which may be utilized to direct future modeling or research efforts. It is

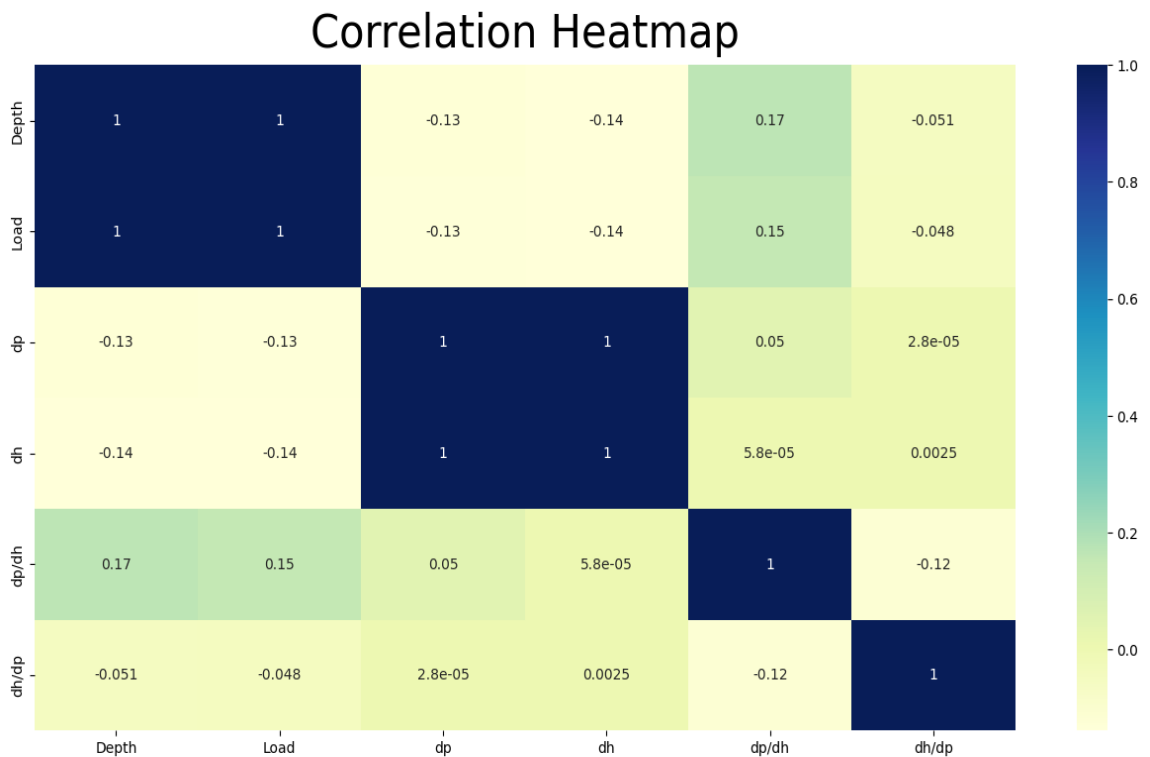


Figure 23: Pearson Correlation between inputs and output.

observed that the components dp, dh, dp/dh, or dh/dp have a significant correlation with the load and depth in our situation. Furthermore, it is obvious that the dp/dh relationship

with load and displacement is stronger than the dh/dp , dp , or dh . Because dp/dh is positive over the entire box, yet dh/dp , dh , and dp have negative values as mentioned in the figure 23. In this way, a machine learning model will be developed to predict the value of dp/dh on the basis of input parameters (load and displacement). So, a journey through data analysis in Google Colab with Python included sorting the data into groups, exploring it with box plots and pair plots, and finally a thorough look at the relationships between variables using the Pearson correlation coefficient. These efforts collectively aimed to enhance our understanding of the dataset's characteristics and pave the way for more informed decision-making in subsequent data-related tasks.

4.2. Performance Criterion of ML models

The coefficient of determination (R^2) and root mean square error (RMSE) are essential metrics employed for assessing the efficacy of various models, specifically in regression tasks. R^2 shows the amount of the output variable's variability that can be explained by the independent variables. A higher R^2 value indicates a stronger relationship between the variables and a more accurate fit of the model. On the other hand, RMSE finds out the average discrepancy between projected and observed values, where smaller values indicate higher levels of predictive precision [81]. The R^2 metric is commonly employed to evaluate the extent to which the features of a model explain the variability in the data. On the other hand, the RMSE value emphasis on the precision of the model's predictions. It is customary to employ both measures in conjunction, as a model that exhibits a high R^2 value and a low RMSE value achieves a harmonious equilibrium between elucidating variance and generating precise predictions. Nevertheless, the selection between these options should be in accordance with the particular objectives of the analysis, taking into account whether one prioritizes the adequacy of the model or the accuracy of predictions, or even both, depending on the given circumstances [82]. In this study, a range of ML models, including SVM, GPR, DT, and ELT, were utilized. These models were trained using the regression toolbox available in MATLAB R2021b. The objective of the training was to develop predictive capabilities for identifying the presence of cracks. In order to evaluate the predictive accuracy of these ML models, the graphs illustrating the correlation between predicted and actual crack occurrences were analyzed. A combination

of GA and PSO-based feature selection and hyperparameter optimization techniques were employed by the ML models to accurately predict the microcracks. The accomplishment was achieved by utilizing a training dataset, which accounted for 80% of the complete dataset consisting of more than 126 data points, alongside a testing dataset that formed the remaining 20% (31 data points). The ML models underwent training utilizing the features that were identified through the use of the GA and PSO, as outlined in Table 1. Furthermore, the machine learning models were trained using optimal hyperparameters, which were established through the process of feature selection and hyperparameter tuning. The data points obtained from the location

Table 1: GA and PSO parameters.

Algorithm Parameters			
Genetic Algorithm	Value	Particle Swarm Optimization	Value
Number of generations	100	Max. Iterations	100
Crossover	Scattered	Max. Velocity	-1-1
Crossover Probability	0.8	Inertia weight	0.4-0.9
Elite Count	3.95	Cognitive Coefficient	2.0
Population Size	50	Swarm Size	25
Population Type	Bitstring	Social Coefficient	2.0
Mutation	Uniform		
Mutation Probability	0.1		

of the microcracks exhibit a horizontal straight line along $x = y$, suggesting a high level of accuracy in the prediction performance. Nevertheless, it is crucial to acknowledge that ML models often demonstrate more accuracy in predicting outcomes on the training dataset as opposed to the testing dataset.

4.3. Features Selection

In certain scenarios, when constructing and training a model, the number of input parameters can be extensive. However, not all of these parameters contribute meaningfully to the output. In such cases, creating a user interface with a vast array of input parameters becomes impractical and overwhelming. To address this challenge, feature selection emerges as a crucial strategy for identifying and prioritizing the input parameters that genuinely influence output predictions. GA and PSO are particularly effective in this regard, as they aid in the removal of redundant or less impactful features

Table 2: GA and PSO based Features Selection.

GA and PSO based features selection	
Models	Selected Features
SVM	Load, Depth
GPR	Load, Depth
DT	Load, Depth
ELT	Load, Depth

from the input set. Essentially, the GA assesses the significance of each input parameter and evaluates its effectiveness in producing meaningful results. For example, in this study, both GA and PSO consistently selected load and displacement as essential input parameters across all models, as shown in Table 2. This decision was based on the crucial influence of these factors in accurately predicting the output. This technique proves exceptionally valuable when dealing with a large number of input parameters, enhancing model efficiency, and facilitating the creation of a user-friendly interface.

4.4. GA and PSO based hyperparameters tuning

To determine the tuning parameters for multiple ML models, the regression model toolkit was utilised. Utilizing standardized data and performing a 5-fold cross-validation comprised the tuning procedure. The selected hyperparameters, along with their corresponding ranges and the optimized values, are detailed in the table. The hyperparameters for the DT, SVM, GPR, and ELT models were fine-tuned and optimized using the GA and PSO approaches. In this study, the SVM model was optimized with specific values using GA and PSO. For GA, the optimal values included a box constraint of 915.2512, a kernel scale of 1.3608, a kernel function of Gaussian, and an epsilon value of 0.0656. In PSO, the values were a box constraint of 999.99, a kernel scale of 0.2239, a kernel function of Gaussian, and an epsilon value of 0.0071. The ensemble model hyperparameters were fine-tuned using GA and PSO as well.

Table 3: Parameter Ranges and Optimized Values in Selection Process.

Models	Hyperparameters	Ranges	Selected Range	
			GA	PSO
SVM	Box Constraint	0.001-1000	915.2512	999.9999
	Kernel Scale	0.001-1000	1.3608	0.2239
	Kernel Function	Gaussian, Linear, Quadratic, Cubic	Gaussian	Gaussian
	Epsilon	10-50	0.0656	0.0071
GPR	Sigma	0.0001-55.069	1.1767	13.7360

	Kernel Function	Non/iso-tropic Exponential, Non/iso-tropic Matern 3/2, Non/iso-tropic Matern 5/2 Non/iso-tropic Rational Quadratic, Non/iso-tropic Squared Exponential,		Squared Exponential
	Basic Function	Constant, Zero, Linear		
	Kernel Scale	0.523-523	324.9509	320.9508
DT	Leaf Size	1-61	8.3739	8.5496
ELT	No. of Learners	10-500	457.5542	133.5503
	Learning Rates	0.001-1	0.6554	0.001
	Methods	LSBoost, Bag	LSBoost	Bag

4.5. Models' performance

A number of ML models such, as GPR, DT, SVM and ELT were employed in order to predict the occurrence of microcracks. and successfully made predictions by utilizing input variables load vs displacement. All the models were selected through a feature selection approach based on GA and PSO methods. Table 4 illustrates a comparison of these models, indicating that overall GPR and SVM exhibited satisfactory performance in crack prediction compared to ELT and DT. R^2 values for GPR, ELT, and SVM in GA during testing were 0.9982, 0.9829, and 0.9780, with corresponding RMSE values of $6.25E^{-16}$, $3.26E^{-14}$, and $2.61E^{-14}$, respectively. In the case of PSO, GPR, DT, and SVM showed satisfactory performance in crack prediction compared to ELT. The R^2 and RMSE values obtained for GPR, DT, and SVM during testing were 0.9979, 0.7744, and 0.9721,

Table 4: Comparison of ML methods using PSO and GA.

Models	R ² in Training		R ² in Testing		RMSE in Training		RMSE in Testing	
	GA	PSO	GA	PSO	GA	PSO	GA	PSO
ELT	0.9854	0.9684	0.9829	0.9581	1.32E ⁻¹³	2.87E ⁻⁶	3.26E ⁻¹⁴	2.87E ⁻⁶
GPR	0.9729	0.9981	0.9982	0.9979	3.73E ⁻⁸	2.87E ⁻⁶	6.25E ⁻¹⁶	7.31E ⁻¹⁶
SVM	0.9693	0.9725	0.9780	0.9721	0.1703	0.0185	2.61E ⁻¹⁴	2.10E ⁻¹⁴
DT	0.9518	0.9518	0.7744	0.7744	7.53E ⁻¹⁴	7.53E ⁻¹⁴	6.10E ⁻⁴	5.10E ⁻⁴

with corresponding RMSE values of $7.31E^{-16}$, $5.10E^{-4}$, and $2.10E^{-14}$, respectively, as detailed in table 4. In comparison to alternative models, the comparatively inadequate performance of DT in GA and ELT in PSO can be attributed to the non-linear nature of crack occurrences with respect to input variables. The testing and training outcomes for the DT and ELT models were less than ideal, as indicated by the testing R^2 and RMSE values of 0.7744 and $6.10E^{-4}$ for DT in GA and 0.9581 and $2.87E^{-6}$, respectively, for ELT in PSO. As a whole, throughout the training and testing phases in GA, the GPR and ELT models exhibited superior performance compared to all other models. Similarly, in PSO, the GPR and SVM models outperformed the competition. By comparing the R^2 and RMSE values of the models, it was determined that the GPR in the GA model performed better than other ML models. Therefore, we selected the GPR workspace in order to develop the graphical user interface.

4.6. Partial dependence plots

Partial Dependence Plots (PDPs) in the figure 24, are used in machine learning to show the relationship visually between predictor variables and the expected result [83]. Distinct non-linear correlations with the goal variable 'Y' are shown by the two PDPs that are provided, one for 'Displacement' and the other for 'Load'. The 'Displacement' PDP has complicated effects on 'Y', with a steep early reduction, a plateau, and a continuing decrease. On the other hand, the connection between 'Y' and the 'Load' PDP is smoother. Y rises to a peak and then gently declines, indicating a different but equally significant effect on the result. These charts emphasize the complex dynamics at work in predictive models, which also emphasize how important it is to understand feature interactions to enhance and interpret models.

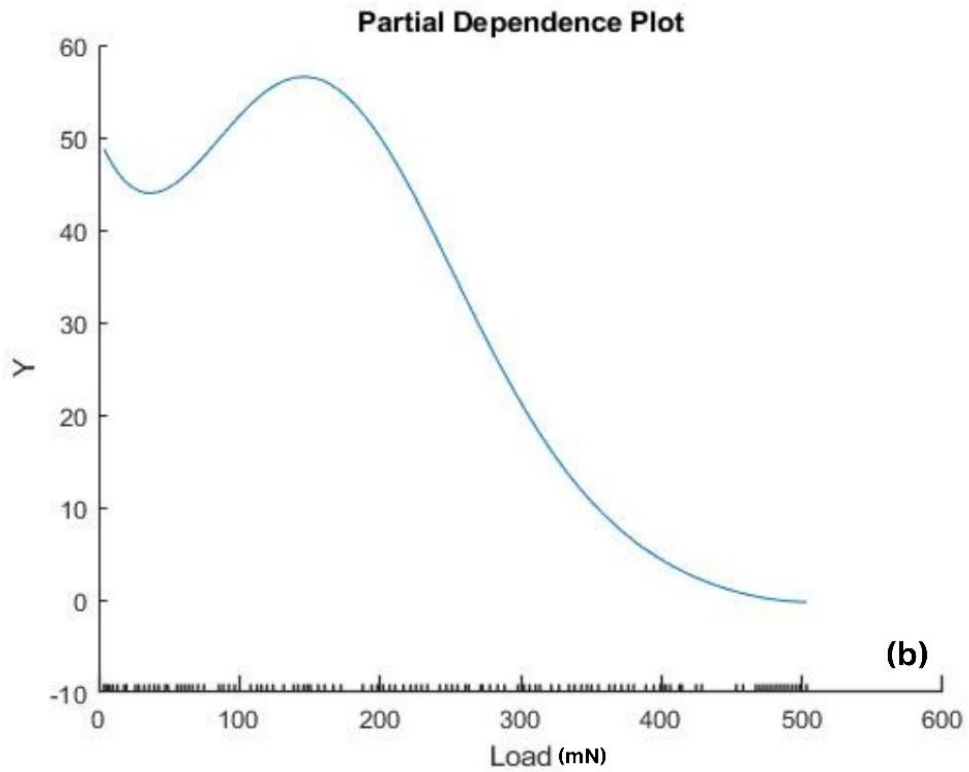
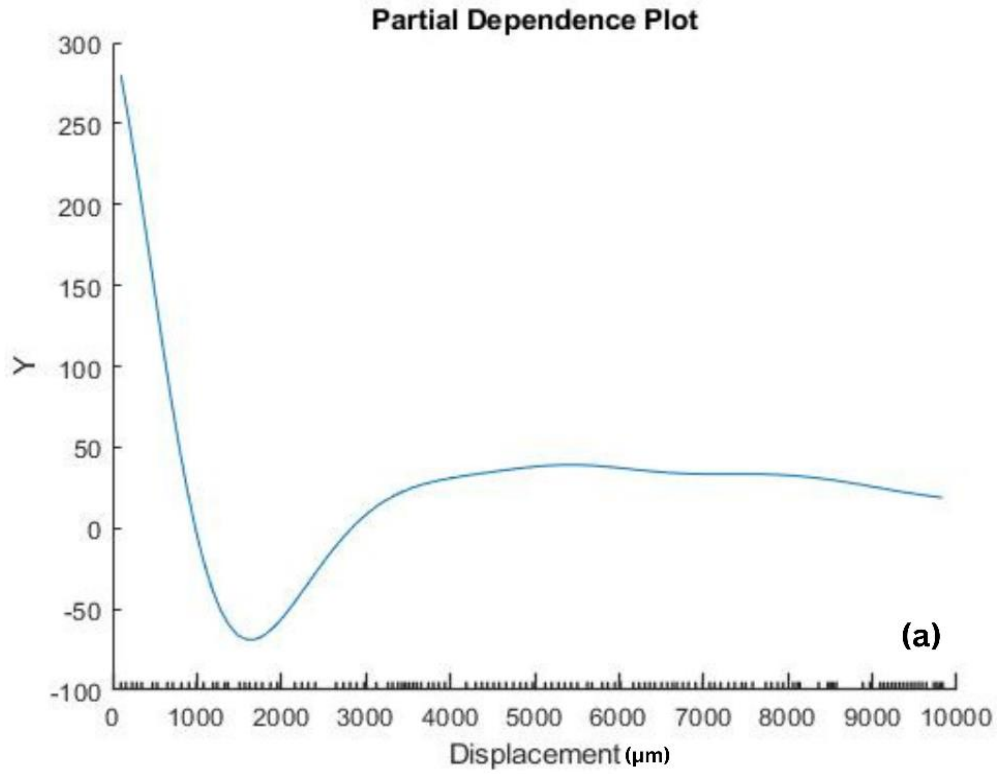
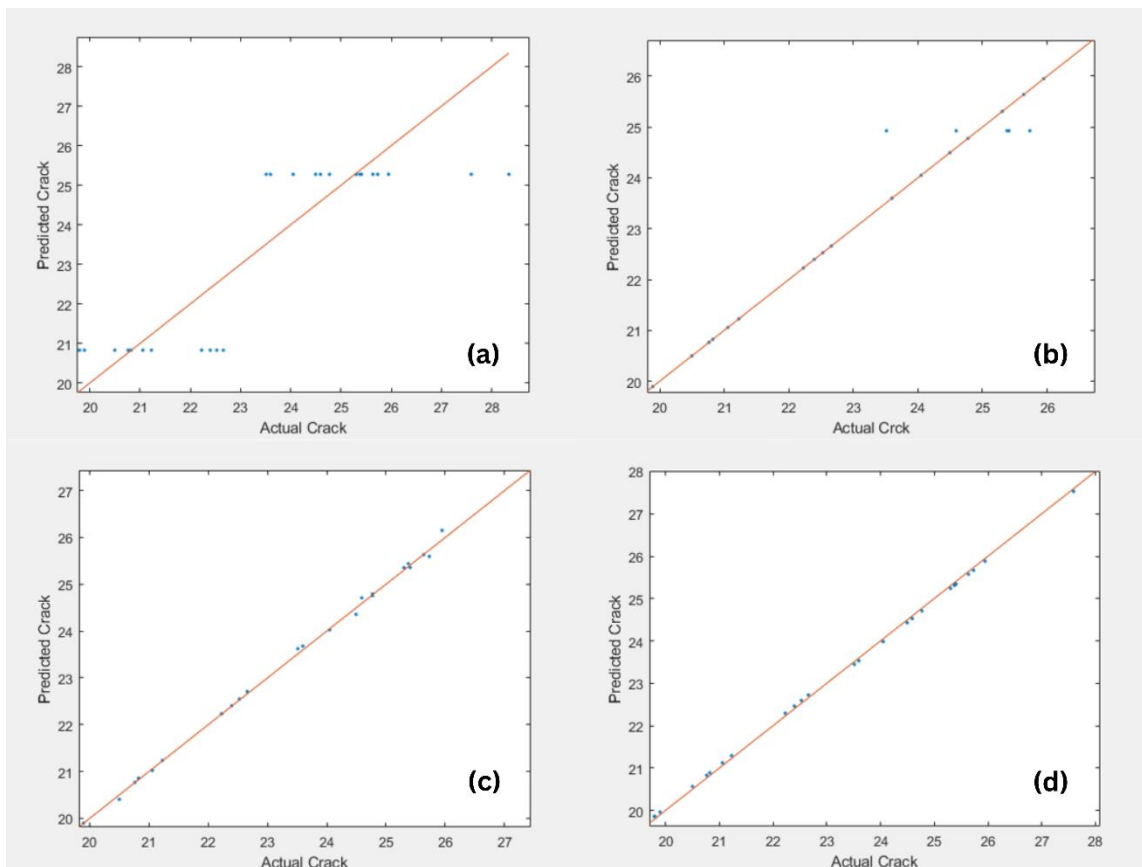


Figure 24: PDP'S Demonstrating the Influence of Inputs on Microcrack Prediction.

4.7. Graphical representation between actual and predicted cracks

The graphical representations in Figure 25 provide a comprehensive overview of the actual and predicted performance of GPR in GA and SVM in PSO. Additionally, other graphs showcasing SVM, DT, and ELT in the context of GA, and GPR, ELT, and DT within PSO, offer clear evidence of minimal deviation between predicted crack outcomes and the actual data.



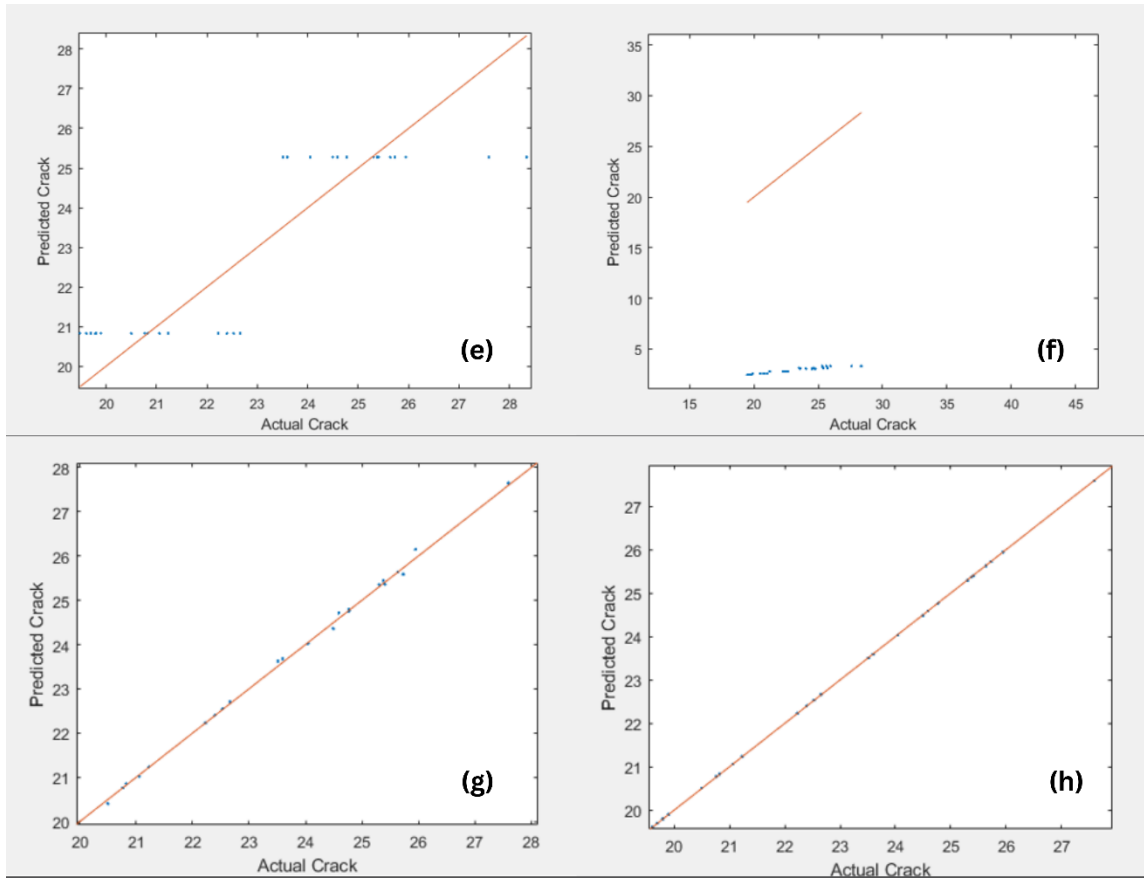


Figure 25: (a) DT-GA (b) ELT-GA (c) GPR-GA (d) SVM-GA predicted vs actual crack
 (e) SVM-GA (f) DT-PSO (g) ELT-PSO (h) GPR-PSO predicted crack against actual crack

4.8. Graphical user interface

A graphical-user interface (GUI) was constructed utilizing graphical images and figures derived from the GPR-GA model. It facilitates communication between operators and electronic equipment. The GUI developed in this study allowed for the input of derived data from the nanoindentation, specifically load and displacement. The GUI uses the GPR model's prediction function to forecast the existence of cracks. The MATLAB 2021b software was employed to develop the GUI. The model was activated via a push button, as illustrated in Figure 26, with load (mN) and displacement (μm) entered as inputs in the

GUI. The discrepancy between the output predicted by the GUI and the yield observed in the experiment was below 2%.

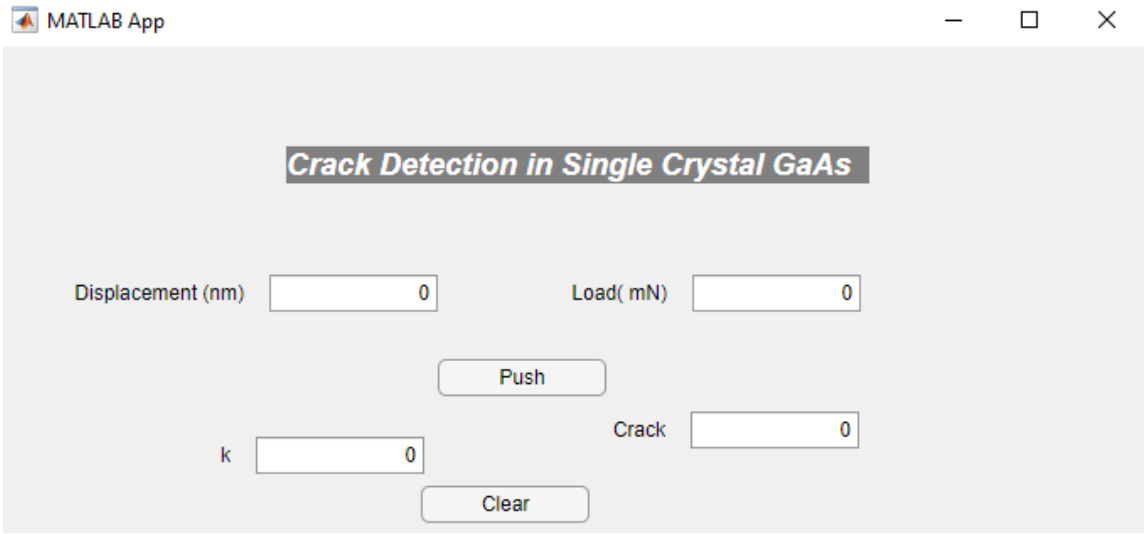


Figure 26: Screenshot of Graphical User Interface.

Conclusions and Recommendations

Conclusions

1. By utilizing nanoindentation data, four distinct categories of ML models were applied to the prediction of microcracks.
2. Employed GA and PSO to hyper-tune the GPR, SVM, DT, and ELT parameters.
3. The prediction accuracy of the GPR model in GA was superior to that of all other models, as evidenced by its R^2 value of 0.9982.
4. The analysis of partial dependence plots demonstrated that the parameters selected via GA-based optimization have a substantial influence on the prognostication of the fracture phenomenon.
5. A user-friendly graphical user interface was created in accordance with the GPR workspace in order to inspect the fracture phenomenon.

Recommendations

1. A novel hybrid of GA and PSO can be applied in pop-in prediction in future which will eliminate the need of ex-situ nanoindentation.
2. The application of Regression learner can be useful in the field of materials science because it has never been done before.
3. This method can be extended to the prediction of the type of crack which leads to the better understanding of materials' behavior which is not available in the literature.

References

- [1] M. Krivov *et al.*, "Properties of n-type gallium arsenide doped with germanium when single crystals are grown from the melt," *Soviet Physics Journal*, vol. 26, pp. 1047-1050, 1983.
- [2] K. Hjort, J. Soderkvist, and J.-A. Schweitz, "Gallium arsenide as a mechanical material," *Journal of Micromechanics and Microengineering*, vol. 4, no. 1, p. 1, 1994.
- [3] J. Wood and D. Morgan, "Gallium arsenide and related compounds for device applications," *Acta Physica Polonica A*, vol. 79, no. 1, pp. 97-116, 1991.
- [4] L. Peng *et al.*, "Achieving homogeneity of InGaN/GaN quantum well by well/barrier interface treatment," *Applied Surface Science*, vol. 505, p. 144283, 2020.
- [5] B. K. Gilbert and G.-W. Pan, "The application of gallium arsenide integrated circuit technology to the design and fabrication of future generation digital signal processors: Promises and problems," *Proceedings of the IEEE*, vol. 76, no. 7, pp. 816-834, 1988.
- [6] M. Kitsunai and T. Yuki, "How gallium arsenide wafers are made," *Applied Organometallic Chemistry*, vol. 8, no. 3, pp. 167-174, 1994.
- [7] M. Jurisch, "The development of lec technology for GaAs single crystal growth from laboratory scale to mass production," *Electronic Materials*, vol. 31, no. 1/2, pp. 7-17, 2003.
- [8] C. Frank-Rotsch, N. Dropka, A. Glacki, and U. Juda, "VGF growth of GaAs utilizing heater-magnet module," *Journal of crystal growth*, vol. 401, pp. 702-707, 2014.
- [9] E. Klimov *et al.*, "GaAs Molecular Beam Epitaxy on (110)-Oriented Substrates," *Crystals*, vol. 13, no. 1, p. 28, 2022.
- [10] S. P. Tobin *et al.*, "Assessment of MOCVD-and MBE-growth GaAs for high-efficiency solar cell applications," *IEEE Transactions on Electron Devices*, vol. 37, no. 2, pp. 469-477, 1990.
- [11] R. Fornari, C. Paorici, L. Zanotti, and G. Zuccalli, "LEC-growth and the major electrical and structural characteristics of semi-insulating gallium arsenide," *Acta Physica Hungarica*, vol. 57, no. 3-4, pp. 263-270, 1985.
- [12] H. Hobgood, G. W. Eldridge, D. L. Barrett, and R. N. Thomas, "High-purity semi-insulating GaAs material for monolithic microwave integrated circuits," *IEEE Transactions on Electron Devices*, vol. 28, no. 2, pp. 140-149, 1981.
- [13] D. Hartzell, L. K. Leung, and F. J. Towner, "Cost-effective, high-volume molecular beam epitaxy," *JOM*, vol. 50, pp. 37-39, 1998.
- [14] K. Wasmer, C. Pouvreau, J.-M. Breguet, J. Michler, D. Schulz, and J. H. Giovanola, "Nanoindentation cracking in gallium arsenide: Part I. In situ SEM nanoindentation," *Journal of Materials Research*, vol. 28, no. 20, pp. 2785-2798, 2013.
- [15] M. Cooke, "Scribe and dice," *III-Vs Review*, vol. 19, no. 4, pp. 20-24, 2006.
- [16] K. Wasmer *et al.*, "Cleavage fracture of brittle semiconductors from the nanometre to the centimetre scale," *Advanced Engineering Materials*, vol. 7, no. 5, pp. 309-317, 2005.

- [17] J. Woïrgard, C. Tromas, J. Girard, and V. Audurier, "Study of the mechanical properties of ceramic materials by the nanoindentation technique," *Journal of the European Ceramic Society*, vol. 18, no. 15, pp. 2297-2305, 1998.
- [18] C. Pouvreau, K. Wasmer, J. Giovanola, J. Michler, J. Breguet, and A. Karimi, "In-situ scanning electron microscope indentation of gallium arsenide," in *Fracture of Nano and Engineering Materials and Structures: Proceedings of the 16th European Conference of Fracture, Alexandroupolis, Greece, July 3–7, 2006*, 2006: Springer, pp. 61-62.
- [19] M. Bagheripoor and R. Klassen, "The effect of crystal anisotropy and pre-existing defects on the incipient plasticity of FCC single crystals during nanoindentation," *Mechanics of Materials*, vol. 143, p. 103311, 2020.
- [20] V. Trabadelo, S. Pathak, F. Saeidi, M. Parlinska-Wojtan, and K. Wasmer, "Nanoindentation deformation and cracking in sapphire," *Ceramics International*, vol. 45, no. 8, pp. 9835-9845, 2019.
- [21] D. Harding, W. Oliver, and G. Pharr, "Cracking during nanoindentation and its use in the measurement of fracture toughness," *MRS Online Proceedings Library (OPL)*, vol. 356, p. 663, 1994.
- [22] B. R. Lawn, A. G. Evans, and D. Marshall, "Elastic/plastic indentation damage in ceramics: the median/radial crack system," *Journal of the American Ceramic Society*, vol. 63, no. 9-10, pp. 574-581, 1980.
- [23] H. Nili, K. Kalantar-zadeh, M. Bhaskaran, and S. Sriram, "In situ nanoindentation: Probing nanoscale multifunctionality," *Progress in Materials Science*, vol. 58, no. 1, pp. 1-29, 2013.
- [24] H. Huang and H. Zhao, "In situ nanoindentation and scratch testing inside scanning electron microscopes: opportunities and challenges," *Science of Advanced Materials*, vol. 6, no. 5, pp. 875-889, 2014.
- [25] R. K. Vasudevan *et al.*, "Materials science in the artificial intelligence age: high-throughput library generation, machine learning, and a pathway from correlations to the underpinning physics," *MRS communications*, vol. 9, no. 3, pp. 821-838, 2019.
- [26] Z. U. Haq, H. Ullah, M. N. A. Khan, S. R. Naqvi, A. Ahad, and N. A. S. Amin, "Comparative study of machine learning methods integrated with genetic algorithm and particle swarm optimization for bio-char yield prediction," *Bioresource Technology*, vol. 363, p. 128008, 2022.
- [27] F. Javid, H. Pouriayevali, and K. Durst, "Dislocation–grain boundary interactions: Recent advances on the underlying mechanisms studied via nanoindentation testing," *Journal of Materials Research*, pp. 1-13, 2021.
- [28] A. C. Fischer-Cripps and A. C. Fischer-Cripps, "Scaling Relationships in Nanoindentation," *Nanoindentation*, pp. 119-123, 2011.
- [29] S. Hainsworth, H. Chandler, and T. Page, "Analysis of nanoindentation load-displacement loading curves," *Journal of Materials Research*, vol. 11, pp. 1987-1995, 1996.
- [30] A. Tiwari and S. Natarajan, *Applied nanoindentation in advanced materials*. John Wiley & Sons, 2017.
- [31] Y. I. Golovin, "Nanoindentation and mechanical properties of materials at submicro-and nanoscale levels: Recent results and achievements," *Physics of the Solid State*, vol. 63, pp. 1-41, 2021.

- [32] A. C. Fischer-Cripps and A. C. Fischer-Cripps, *Nanoindentation test standards*. Springer, 2011.
- [33] H. Wang, L. Zhu, B. Xu, H. Wang, L. Zhu, and B. Xu, "Principle and methods of nanoindentation test," *Residual Stresses and Nanoindentation Testing of Films and Coatings*, pp. 21-36, 2018.
- [34] A. C. Fischer-Cripps and D. Nicholson, "Nanoindentation. Mechanical engineering series," *Appl. Mech. Rev.*, vol. 57, no. 2, pp. B12-B12, 2004.
- [35] T. Chudoba, P. Schwaller, R. Rabe, J.-M. Breguet, and J. Michler, "Comparison of nanoindentation results obtained with Berkovich and cube-corner indenters," *Philosophical Magazine*, vol. 86, no. 33-35, pp. 5265-5283, 2006.
- [36] Y. Gao, "Beer and diaper," ed: Tsinghua University Press, Beijing, 2008.
- [37] X. Li and B. Bhushan, "A review of nanoindentation continuous stiffness measurement technique and its applications," *Materials characterization*, vol. 48, no. 1, pp. 11-36, 2002.
- [38] M. L. Oyen and R. F. Cook, "A practical guide for analysis of nanoindentation data," *Journal of the mechanical behavior of biomedical materials*, vol. 2, no. 4, pp. 396-407, 2009.
- [39] S. Pathak, J. L. Riesterer, S. R. Kalidindi, and J. Michler, "Understanding pop-ins in spherical nanoindentation," *Applied Physics Letters*, vol. 105, no. 16, 2014.
- [40] Y. Lin, Y. Weng, D. Pen, and H. Li, "Deformation model of brittle and ductile materials under nano-indentation," *Materials & Design*, vol. 30, no. 5, pp. 1643-1649, 2009.
- [41] M. Arafin and J. Szpunar, "A new understanding of intergranular stress corrosion cracking resistance of pipeline steel through grain boundary character and crystallographic texture studies," *Corrosion Science*, vol. 51, no. 1, pp. 119-128, 2009.
- [42] B. R. Lawn, "Indentation of ceramics with spheres: a century after Hertz," *Journal of the American ceramic society*, vol. 81, no. 8, pp. 1977-1994, 1998.
- [43] R. Rabe *et al.*, "Observation of fracture and plastic deformation during indentation and scratching inside the scanning electron microscope," *Thin Solid Films*, vol. 469, pp. 206-213, 2004.
- [44] G. Pharr and W. Oliver, "Measurement of thin film mechanical properties using nanoindentation," *Mrs Bulletin*, vol. 17, no. 7, pp. 28-33, 1992.
- [45] Y. Xia, M. Bigerelle, J. Marteau, P. E. Mazeran, S. Bouvier, and A. Iost, "Effect of surface roughness in the determination of the mechanical properties of material using nanoindentation test," *Scanning: The Journal of Scanning Microscopies*, vol. 36, no. 1, pp. 134-149, 2014.
- [46] M. Rodríguez, J. M. Molina-Aldareguía, C. González, and J. LLorca, "Determination of the mechanical properties of amorphous materials through instrumented nanoindentation," *Acta Materialia*, vol. 60, no. 9, pp. 3953-3964, 2012.
- [47] S. Kossman and M. Bigerelle, "Pop-in identification in nanoindentation curves with deep learning algorithms," *Materials*, vol. 14, no. 22, p. 7027, 2021.
- [48] G. Baiocco and N. Ucciardello, "Neural network implementation for the prediction of secondary phase precipitation and mechanical feature in a duplex stainless steel," *Applied Physics A*, vol. 125, pp. 1-8, 2019.

- [49] M. Yescas, "Prediction of the Vickers hardness in austempered ductile irons using neural networks," *International Journal of Cast Metals Research*, vol. 15, no. 5, pp. 513-521, 2003.
- [50] A. Allen, M. Hutchings, C. Windsor, and C. Andreani, "Neutron diffraction methods for the study of residual stress fields," *Advances in Physics*, vol. 34, no. 4, pp. 445-473, 1985.
- [51] E. S. Puchi-Cabrera, E. Rossi, G. Sansonetti, M. Sebastiani, and E. Bemporad, "Machine learning aided nanoindentation: A review of the current state and future perspectives," *Current Opinion in Solid State and Materials Science*, vol. 27, no. 4, p. 101091, 2023.
- [52] A. Kaplan and M. Haenlein, "Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence," *Business horizons*, vol. 62, no. 1, pp. 15-25, 2019.
- [53] J. Wei *et al.*, "Machine learning in materials science," *InfoMat*, vol. 1, no. 3, pp. 338-358, 2019.
- [54] A. Smola, "Introduction to machine learning," ed, 2008.
- [55] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, pp. 273-297, 1995.
- [56] Y. Baştanlar and M. Özuysal, "Introduction to machine learning," *miRNomics: MicroRNA biology and computational analysis*, pp. 105-128, 2014.
- [57] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *Journal of Big Data*, vol. 7, no. 1, p. 52, 2020.
- [58] W. Y. Loh, "Classification and regression trees," *Wiley interdisciplinary reviews: data mining and knowledge discovery*, vol. 1, no. 1, pp. 14-23, 2011.
- [59] E. Pekel, "Estimation of soil moisture using decision tree regression," *Theoretical and Applied Climatology*, vol. 139, no. 3-4, pp. 1111-1119, 2020.
- [60] D. Malerba, F. Esposito, M. Ceci, and A. Appice, "Top-down induction of model trees with regression and splitting nodes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 612-625, 2004.
- [61] S. S. Rathore and S. Kumar, "A decision tree regression based approach for the number of software faults prediction," *ACM SIGSOFT Software Engineering Notes*, vol. 41, no. 1, pp. 1-6, 2016.
- [62] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in neural information processing systems*, vol. 9, 1996.
- [63] N. Ancona, "Classification properties of support vector machines for regression," *Technical Report, RIIESI/CNR-Nr. 02/99*, 1999.
- [64] A. Ben-Hur, C. S. Ong, S. Sonnenburg, B. Schölkopf, and G. Rätsch, "Support vector machines and kernels for computational biology," *PLoS computational biology*, vol. 4, no. 10, p. e1000173, 2008.
- [65] C. K. Williams, "Prediction with Gaussian processes: From linear regression to linear prediction and beyond," in *Learning in graphical models*: Springer, 1998, pp. 599-621.
- [66] M. Abe *et al.*, "Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics): Preface,"

- Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3960, pp. V-VI, 2006.
- [67] A. Banerjee, D. B. Dunson, and S. T. Tokdar, "Efficient Gaussian process regression for large datasets," *Biometrika*, vol. 100, no. 1, pp. 75-89, 2013.
- [68] K. Kim, D. Lee, and I. Essa, "Gaussian process regression flow for analysis of motion trajectories," in *2011 International Conference on Computer Vision*, 2011: IEEE, pp. 1164-1171.
- [69] T. G. Dietterich, "Ensemble learning," *The handbook of brain theory and neural networks*, vol. 2, no. 1, pp. 110-125, 2002.
- [70] X. Lu, W. Zhou, X. Ding, X. Shi, B. Luan, and M. Li, "Ensemble learning regression for estimating unconfined compressive strength of cemented paste backfill," *IEEE access*, vol. 7, pp. 72125-72133, 2019.
- [71] Z. U. Haq, H. Ullah, M. N. A. Khan, S. R. Naqvi, and M. Ahsan, "Hydrogen production optimization from sewage sludge supercritical gasification process using machine learning methods integrated with genetic algorithm," *Chemical Engineering Research and Design*, vol. 184, pp. 614-626, 2022.
- [72] S. Mirjalili and S. Mirjalili, "Genetic algorithm," *Evolutionary Algorithms and Neural Networks: Theory and Applications*, pp. 43-55, 2019.
- [73] A. Lambora, K. Gupta, and K. Chopra, "Genetic algorithm-A literature review," in *2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon)*, 2019: IEEE, pp. 380-384.
- [74] J. Shapiro, "Genetic algorithms in machine learning," in *Advanced Course on Artificial Intelligence*: Springer, 1999, pp. 146-168.
- [75] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, 1995, vol. 4: IEEE, pp. 1942-1948.
- [76] S. Kiranyaz, T. Ince, and M. Gabbouj, *Multidimensional particle swarm optimization for machine learning and pattern recognition*. Springer, 2014.
- [77] M. F. Sanner, "Python: a programming language for software integration and development," *J Mol Graph Model*, vol. 17, no. 1, pp. 57-61, 1999.
- [78] K. Sahoo, A. K. Samal, J. Pramanik, and S. K. Pani, "Exploratory data analysis using Python," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 12, pp. 4727-4735, 2019.
- [79] Y. Jadoul, B. Thompson, and B. De Boer, "Introducing parselmouth: A python interface to praat," *Journal of Phonetics*, vol. 71, pp. 1-15, 2018.
- [80] Z. Gu, R. Eils, and M. Schlesner, "Complex heatmaps reveal patterns and correlations in multidimensional genomic data," *Bioinformatics*, vol. 32, no. 18, pp. 2847-2849, 2016.
- [81] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Computer Science*, vol. 7, p. e623, 2021.
- [82] W. Wang and Y. Lu, "Analysis of the mean absolute error (MAE) and the root mean square error (RMSE) in assessing rounding model," in *IOP conference series: materials science and engineering*, 2018, vol. 324: IOP Publishing, p. 012049.
- [83] B. M. Greenwell, "pdp: An R package for constructing partial dependence plots," *R J.*, vol. 9, no. 1, p. 421, 2017.

