International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V74I1P101

Volume 74 Issue 1, 1-24, January 2026
© 2026 Seventh Sense Research Group®

Original Article

Comparative Analysis of Classical Machine Learning Models for Predictive Maintenance in the Brewing Industry

Achille EBOKE[1], Richard NASSO TOUMBA[1], Samuel Maxime MOAMISSOAL[1], Timothée KOMBE[1]

[1]*Laboratory of Technology and Applied Science (LTSA), University Institute of Technology (IUT), University of Douala, Cameroon.*

[1]*Corresponding Author : ebokeachille@gmail.com*

Received: 09 June 2025 Revised: 21 November 2025 Accepted: 25 December 2025 Published: 14 January 2026

Abstract *- The article investigates the application of classical Machine Learning (ML) Algorithms for enhanced equipment for reliability and failure prognosis within complex industrial systems, specifically focusing on the brewing industry. The objective is to develop a Robust Machine Learning framework to anticipate equipment breakdowns and ensure optimal operational performance accurately. This methodology encompasses a dual approach, first leveraging comprehensive simulation data sets derived from a demanding manufacturing environment to identify operational irregularities and predict potential equipment failures. Second, undertake a rigorous comparative analysis of various supervised learning models, including Decision Trees, Random Forests, Logistic Regression, Support Vector Machines (SVM), and K-nearest neighbors. These models were implemented using Python and evaluated meticulously through metrics such as confusion matrices, classification reports, ROC curves, and stratified cross-validation. Results indicate that the Random Forest model demonstrates superior overall performance for binary classification in this context. This comparative assessment provides critical insights for selecting and implementing the most effective predictive maintenance strategies, aiming to optimize brewing operations. Future work will concentrate on refining the identified high-performing models and exploring class-specific performance metrics to mitigate further the costs associated with false positives and false negatives.*

Keywords *- Brewing Industry, Machine Learning, Predictive Maintenance, Model performance.*

1. Introduction

This century has seen a tremendous evolution in the industrial sector, leading to the emergence of the complex industry and all of its associated technologies [1-5]. However, when a portion of the production system is started, a fault that spreads cannot be quickly and accurately traced by the operator in the manufacturing sector [6]. Furthermore, once the procedure has been initiated, the machines cannot be interrupted. However, a tracing system is necessary to safeguard and preserve the plant and its output, which are expensive assets [7, 8]. In order to prevent production stoppages brought on by failures, flaws that spread during plant startup must also be identified and fixed. The use of Artificial Intelligence (AI) in industrial processes has revolutionized productivity and creativity [9]. Real-time defect diagnosis, prediction, and operation optimization are made possible by the integration of AI into automation systems and data analytics, which enhances industrial processes [10, 11]. AI makes it possible to put solutions like automated repetitive tasks, predictive maintenance, and better inventory management into practice [12, 13]. When paired with data analysis, these technologies provide a notable boost in operational decision-making and production performance. The use of Machine Learning (ML) lowers operating costs and allows for product personalization, which propels the digital transformation of numerous sectors [14-18]. Using linked data on equipment failures in the brewing sector, five machine learning models will be considered for classification in the rest of the work, including Decision Trees, Random Forests, Logistic Regression, Support Vector Machines (SVM), and k-nearest neighbors. The classification report, the ROC curve, stratified cross-validation, and confusion matrices are the instruments used to assess performance [19]. In order to determine the optimal model, we compare these tools at the conclusion.

2. Literature Review

Numerous techniques have been employed in publications that review the literature on fault propagation, including the development of a Neuro-Fuzzy Monitoring tool [20-22], stochastic modeling based on the MEE (State Space Model), and the ALTARICA DATA FLOW application,

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

which functions as a kind of black box that brings out all the necessary parameters to determine the OEE (Synthetic Rate Of Return), a performance indicator par excellence, while relying on the NFE 60-182 [23-26]. Later, using this same indication, additional models, including LSTM, were employed to assess dynamic reliability [27-29]. In his thesis, he employs the detection and localization of problems in electric motors as one of his methods and discusses this topic. He uses topological, nominal, and linear models for diagnostics and comes to the conclusion that the monitoring features improve equipment management by reducing downtime, streamlining maintenance, and boosting operational safety [30, 21]. Advanced defect diagnosis and detection methods are discussed by [31-33] , Here, he utilizes evolutionary algorithms to find leaks in water pipes and comes to the conclusion that the more search space there is, the less precise the results are. In his theme titled "Detection of Defects in a Predictive Maintenance Environment, [34-36] lays out a methodology that can predict a variety of defects using signals obtained from direct current motors. He uses MATLAB Simulink's predictive models, Deep Learning, FcN, Resnet, the encoder, and the LSTM as tools. He concludes that by adjusting the encoder model's hyperparameters, an average accuracy of 88.53% was achieved, which is higher than he had anticipated [37, 38].

Discusses fatigue harm and the management of additive manufacturing flaws; in his work, he will intentionally create model flaws and determine how detrimental they are to fatigue. It will produce deterministic submillimeter flaws with sizes ranging from 150 to 1000 µm using non-destructive porosity detection techniques in additive manufacturing. [39] were inspired by [40] talk about the use of Bayesian networks for fault diagnosis and detection in systems. In their article, Implementation of Bayesian networks in an embedded system to help diagnose failures of industrial machines, they suggest an embedded system that enables the collection, processing, and filtering of operating data from an industrial machine. This is so that a diagnostic support tool based on graphical probabilistic models (Bayesian networks) can be developed by utilizing vast amounts of data from several machine-embedded sensors. With the use of Bayesian networks produced by learning algorithms, this tool will enable the diagnosis of potential faults in the observed system without the need for knowledge of the physical models of the system's constituent parts. In his thesis, "Applications of Artificial Intelligence (AI) to the detection and isolation of multiple faults in a telecommunications network", [41-44], he suggests a general probabilistic approach that makes it easier to model how faults spread across big telecommunications networks. A model of the transmission of faults and alerts on the Fiber to the Home (FTTH) access network of the Gigabit Passive Optical Network (GPON) type is developed using this approach, which is based on the formalism of Bayesian networks. It demonstrates that the diagnostic performance of the "expert" model of the GPON-FTTH network is reasonably enhanced by the probabilistic model with optimized parameters. In their paper Adaptation of epidemiological models for the analysis of the propagation of failures in a production resource, [45, 46], the authors reviewed the literature. First, they examined the primary models of disruptive phenomenon propagation for industrial systems. They then presented the principles of epidemiological models and how they can be adapted to non-living systems. Similarly, [47] present a SIQS model that considers quarantine and vaccination. In his book "Mathematical Modelling of the Propagation of an Epidemic," [48] begins with the SIR model and its variations while considering a dynamic system. He then discusses the spatio-temporal model of the spread of an epidemic, develops some models like the percolation and forest fire models, compares biological and artificial neural networks in all of their aspects, and concludes by developing some known propagation networks. The project "Detection, classification, and localization of faults in transmission lines by Artificial Neural Networks" by [49-51], examines the application of Neural Networks for the rapid and accurate detection, classification, and localization of faults in electric power transmission lines in order to support a new generation of protection relay systems. In addition to causing equipment damage and system outages, faults also pose a serious threat to the power grid's stability and operability. For each of the three phases, a feedforward Neural Network and a backpropagation algorithm utilizing MATLAB Simulink will be used to identify whether a fault is present or absent, categorize it based on its transitory characteristics, and indicate where it is located on a transmission line. In the framework of Industry 4.0, [52], whose thesis focuses on the use of Machine Learning Algorithms for the detection of bearing failures on rotating machines, first applies techniques that enable us to extract features from a rotating machine's data. After that, he installed a mechanism to keep an eye on the machine's condition by establishing a threshold for appropriate operation and another to sound an alarm when the latter is achieved. Second, he classified the various failure levels using MATLAB's machine learning methods.

The estimating technique known as "Cross-Validation" yielded a reliability of 99.3% after the time and frequency domain features of the signals were extracted. The model parameters are optimized during this learning phase to match the data best. Next, it assesses "test set validation," another validation method. For huge datasets, this method is advised. Following a number of tests, we achieved a 100% categorization rate for the various defect levels taken into account. Contribution to the detection of faults for the predictive maintenance of mechatronic systems using methods based on observers [53-55]. Sparse reconstruction is a novel diagnostic technique that has been applied to gear transmissions to study and diagnose various errors that arise in dynamic systems. A dynamic algorithm that uses a few system observations to estimate a sparse fault vector is the foundation of the sparse reconstruction technique. A non-

linear global system that describes the mechatronic interactions is obtained by using an integrated modeling technique that respects the inputs and outputs of each subsystem. The fact that a limited "parsimonious" number of defects can exist simultaneously is one of its peculiarities. The number of sensors that must be installed determines this figure. We can identify more issues if we increase the number of sensors. The fact that this technology provides an online diagnostic is one of its benefits. In their paper Detection and classification of flaws for a GPV: Comparative research between the thresholding approach and neural networks, [56, 57], they did a comparison of the thresholding method to examine the neural approach's performance. Based on his simulation results, he finds that, in comparison to the thresholding method, Artificial Neural Networks are the most appropriate (simple and easy) strategy for GPV diagnosis. Patch analysis, the foundation of our system, enables the detection of even the most minute flaws.

A multi-class SVM technique and the usage of a GAN's discriminative component form the basis of the article by [58], titled One-Class detection and classification of faults on concrete surfaces. The findings are intriguing because they demonstrate how well a cGAN can employ descriptors in this case, LBPs to detect abnormalities. To be more specific, our approach allows for the detection of all kinds of flaws and resulting in a categorization of defects where the multi-OC-SVM performs well with a 91% accuracy rate. In his thesis, "Machine learning applied to the analysis and prediction of failures in HPC systems," [59].

More specifically, his thesis makes two major contributions: the first is about anticipating processor overheating in High-Performance Computing (HPC) systems, and the second is about analyzing and emphasizing the connections between the events that are recorded in the system logs. Real data from a sizable HPC system in use for production is utilized to assess these two contributions. In their paper Strategies for AltaRica modeling of fault propagation in dynamic systems, published in 2022 [60] provide advice on how to take full advantage of this dynamic modeling feature. As part of the System & Safety Continuity (S2C) project of the IRTs Saint Exupéry and System, a group of experts in traditional safety and Model Based Safety Assessment (MBSA) created the suggested guidelines [20, 22 ,61, 62]. In his essay titled "Improving Machine Availability with Machine Learning," [63] states that businesses can use machine learning models to examine machine data in real time, identify abnormalities, and foresee issues before they arise. As a result, unplanned downtime is decreased, MTTR and MTBF are optimized, productivity rises, and maintenance expenses fall.

3. Materials and Methods

3.1. Methodology for Acquiring Data

The preferred strategy for monitoring, steering, failure detection, and prediction at the moment is data-driven methodologies, sometimes referred to as data-driven methods, which have grown in favor [64-67]. These approaches have a distinct edge since they leverage the utilization of large data sets and intricate Machine Learning Techniques [68]. From the gathered data, this method enables the extraction of pertinent and valuable information [12]. As seen in Figure 1 in the illustration, this data is gathered in a serial configuration production line.

Fig. 1 Overall view of the brewing production system

The decoder, washer, filler, tester, labeler, packer, and coder are the seven major equipment blocks that make up the synoptic shown in Figure 2 of our brewing production system.

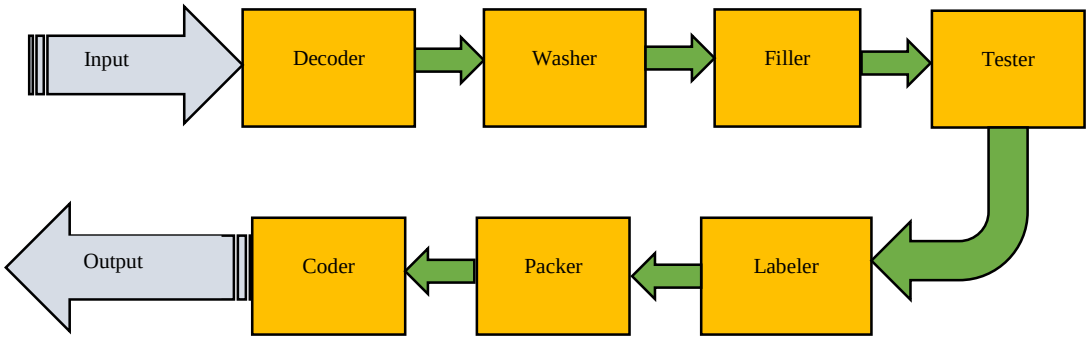


Fig. 2 Industrial brewing process diagram

	cl	Heures payées	Cad.	Cptr Soutireuse	Prod théorique	Entrées magasin	Rdt(EM sur Prod théo) %	Décaiss.	Laveuse	Soutireuse	Mireuse	Etiqueteuse	Encaiss.	Codeuse	defaillance	Mois
0	65	13.5	0	200651	283500	200640	0.707725	0	45	0	0	0	0	0	Laveuse	Janvier
1	33	11.0	1	157992	275000	157416	0.572422	0	60	0	0	0	0	0	Laveuse	NaN
2	65	10.0	0	202121	210000	200460	0.954571	0	0	0	0	0	0	0	aucune	NaN
3	33	9.5	1	190847	237500	190728	0.803065	0	0	0	0	0	0	0	aucune	NaN
4	65	9.0	0	171471	189000	171216	0.905905	0	0	0	0	0	0	0	aucune	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
289	65	13.5	1	231139	283500	230844	0.814265	0	0	30	0	0	0	0	Soutireuse	NaN
290	65	13.0	1	215550	273000	215280	0.788571	0	0	0	0	15	57	8	Encaisseuse	NaN
291	33	14.5	0	245520	362500	245520	0.677297	0	40	30	0	45	0	0	Etiqueteuse	NaN
292	65	11.5	1	214587	241500	214416	0.887851	0	0	0	0	0	0	15	Codeuse	NaN
293	60	11.5	-1	165788	241500	165576	0.685615	0	0	0	0	0	0	50	Codeuse	NaN

294 rows × 16 columns

Fig. 3 Dataset of Sobraga

Loading the libraries we previously downloaded into our workspace and then the file containing our dataset is the first step in the preprocessing procedure. The next stage is complete data cleansing, which involves changing data formats and handling missing values. While numerical codes can be preserved or altered by label coding, categorical variables, such as defect categories, are managed through techniques like one-hot coding or ordinal coding. Numerical variables are subjected to normalization. Lastly, we employ fault-based encoding to reverse dictionary keys and values and temporal aggregation to streamline and improve the analysis (see Figure 4). This methodical approach guarantees precision and effective implementation, providing a strong basis for the creation of advanced prediction models and well-informed operational decision-making. In image 5, she offers a summary of a supervised learning problem, drawing inspiration from the book titled Introduction to ML by Chloe Agathe Azencot [69].

	cl	Heures payées	Cad.	Cptr Soutireuse	Prod théorique	Entrées magasin	Rdt(EM sur Prod théo) %	Décaiss.	Laveuse	Soutireuse	Mireuse	Etiqueteuse	Encaiss.	Codeuse	defaillance
0	65	13.5	0	200651	283500	200640	0.707725	0	45	0	0	0	0	0	Laveuse
1	33	11.0	1	157992	275000	157416	0.572422	0	60	0	0	0	0	0	Laveuse
2	65	10.0	0	202121	210000	200460	0.954571	0	0	0	0	0	0	0	aucune
3	33	9.5	1	190847	237500	190728	0.803065	0	0	0	0	0	0	0	aucune
4	65	9.0	0	171471	189000	171216	0.905905	0	0	0	0	0	0	0	aucune
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
289	65	13.5	1	231139	283500	230844	0.814265	0	0	30	0	0	0	0	Soutireuse
290	65	13.0	1	215550	273000	215280	0.788571	0	0	0	0	15	57	8	Encaisseuse
291	33	14.5	0	245520	362500	245520	0.677297	0	40	30	0	45	0	0	Etiqueteuse
292	65	11.5	1	214587	241500	214416	0.887851	0	0	0	0	0	0	15	Codeuse
293	60	11.5	-1	165788	241500	165576	0.685615	0	0	0	0	0	0	50	Codeuse

294 rows × 15 columns

Fig. 4 Data following preprocessing

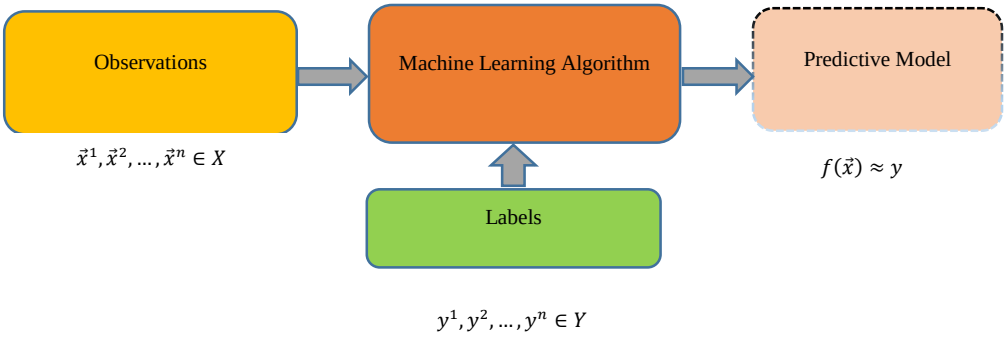


Fig. 5 An overview of a supervised learning problem

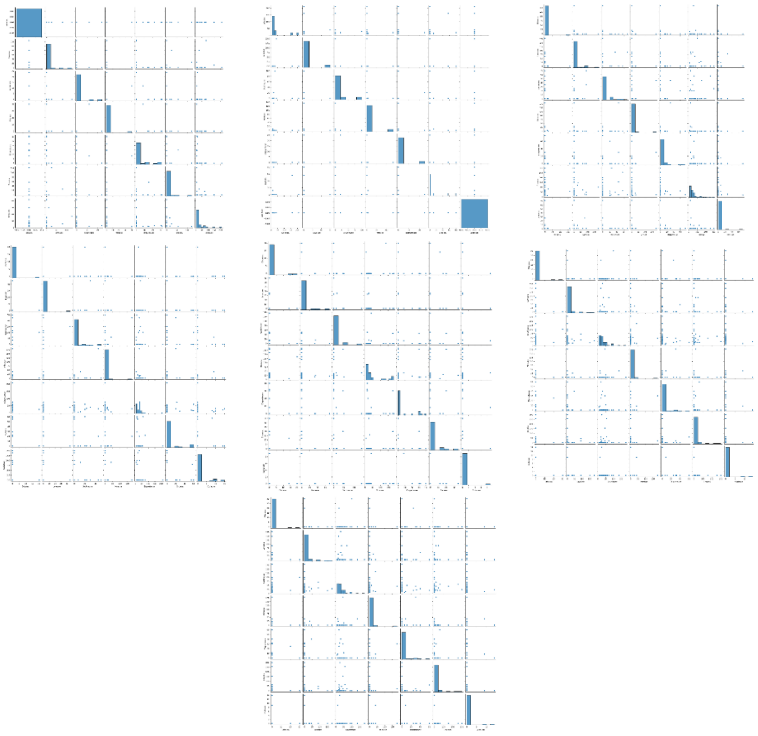


Fig. 6 Data collection failure distribution

The various distributions resulting from the failures documented in the collection of failures at the level of these seven pieces of equipment, as well as the months, which stand in for the days of information gathering, are flawlessly depicted in Figure 6. One way to formulate a supervised learning problem is as follows: The objective of supervised learning is to identify a function $f: X \rightarrow Y$ such that $f(\vec{x}) \approx y$ for all pairs $(\vec{x}, y) \in X \times Y$ having the same relationship as the observed pairs, given n observations, $\{\vec{x}^1, \vec{x}^2, \ldots, \vec{x}^n\}$, where each observation $\vec{x}^i$ Is an element of the space of observations X and their labels$\{y^1, y^2, \ldots, y^n\}$, where each label y^i Belongs to the space of labels, Y. The learning game is made up of everything. $D = \{(\vec{x}^i, y^i)\}_{i=1,\ldots,n}$. This article will examine a specific instance of $Y = \{1,2, \ldots, c\}$ that involves a multi-class categorization. The decision function for this case will be:

$g_c: X \rightarrow \mathbb{R}$ Such as

$$f(\vec{x}) = \underset{C=1,\ldots,c}{\operatorname{argmax}}\, g_c(\vec{x}) \quad (1)$$

The space can be divided into decision regions thanks to this idea of a decision function.

Then, we have C decision regions in the multi-class scenario.

$$R_c = \left\{\vec{x} \in X \middle| g_c(\vec{x}) = \max_k g_c(\vec{x})\right\} \quad (2)$$

However, a C-class classification problem can be solved with any binary classification technique, either one-against-all or one-against-one.

$$f(\vec{x}) = \underset{C=1,\dots,c}{\operatorname{argmax}} \sum_{k\neq c} g_c(\vec{x})$$

$$f(\vec{x}) = \underset{C=1,\dots,c}{\operatorname{argmax}} (\sum_{k\neq c} g_{ck}(\vec{x})) \quad (3)$$

3.2. The Many Algorithms that are Employed

3.2.1. Logistic Regression

One of the basic and widely used strategies for resolving classification issues is logistic regression. To observe a discrete set of classes, LR is a supervised machine learning classification algorithm. A statistical technique for modeling the likelihood of a binary event based on one or more independent variables is called Logistic Regression [70]. The reason for the term "Logistic Regression" is that the fundamental methodology is quite similar to that of linear regression. This categorization method's usage of the Logit function is where the term "Logistics" originates. Logistic regression forecasts the likelihood that an event will fall into either the positive class (1) or the negative class (0), in contrast to linear regression, which forecasts continuous values [71, 72]. The model is known as logistic regression:

$$f: x \rightarrow \sigma\left(\vec{\beta}^T \vec{x}\right) \quad (4)$$

The coefficients of which are derived by:

$$\underset{\vec{\beta}\in\mathbb{R}_{P+1}}{\operatorname{argmax}} \sum_{i=1}^{n} y^i log\sigma\left(\vec{\beta}^T \vec{x}^i\right) + (1-y^i)\log(1-\sigma\left(\vec{\beta}^T \vec{x}^i\right)) \quad (5)$$

3.2.2. Decision Tree

A decision tree is a type of prediction model that can be shown as a tree. Every child node in the tree represents a potential response to the condition that each node tests on a variable[73, 74]. Each label is represented by a leaf on the tree. We "follow" the test answers from the tree's root to forecast the label of an observation, then return the label of the leaf we arrive at. A decision tree divides the observation space *X* into as many regions as its leaves; all observations are then assigned the same label inside a single zone. This label is the most common label in the area when there is a categorization issue. We can write if *n* observations. $\vec{x}^1, \vec{x}^2, \dots, \vec{x}^n$ of *X* are labeled by $y^1, y^2, \dots, y^n$ and R regions $R_1, R_2, \dots, R_R$.

$$f(x) = \sum_{r=1}^{R} \delta_{x\in R_r} \underset{C=1,\dots,c}{\operatorname{argmax}} \sum_{i:.\vec{x}^i\in R_r} \delta(y^i, C) \quad (6)$$

Even if the latter results from the former of the two classes, we will address the multi-class issue in this article. This label represents the average label of the observations in this area for a regression problem:

$$f(x) = \sum_{r=1}^{R} \delta_{x\in R_r} \frac{1}{|R_r|} \sum_{i:.\vec{x}^i\in R_r} y^i \quad (7)$$

3.2.3. Random Forest

An ensemble learning technique for regression and classification is called Random Forest. The reason it is named "Forest" is that it grows a forest of decision trees. Through bootstrap sampling and random feature selection, this approach introduces diversity by building several decision trees. Either a majority vote (classification) or an average (regression) yields the final prediction. A forest ensures a more accurate result with more groups and decisions than a single decision tree, which only offers one outcome and a limited number of groups. The Random Forest algorithm's popularity stems from its ability to tackle classification and regression problems efficiently due to its versatility, high performance, robustness, and ease of use. This algorithm's strength rests in its capacity to work with intricate datasets while preventing overfitting, which makes it an effective tool for a range of Machine Learning predictive tasks [72**,** 75]**.** The Random Forest algorithm's capacity to handle data sets with both continuous variables, as in regression, and categorical variables, as in classification, is one of its primary characteristics. It is a favored option for classification and regression tasks due to its strong performance in these domains [76, 77].

3.2.4. Support Vector Machine

The following optimization problem is what we refer to as SVM:

$$\underset{\vec{w}\in\mathbb{R}^P, b\in\mathbb{R}}{\operatorname{argmin}} \frac{1}{2}\|\vec{w}\|_2^2 t.q.\, y^i(\langle\vec{w}, \vec{x}^i\rangle + b) \geq 1, i = 1, \dots, n \quad (8)$$

Consider the answers to equation (8), $\vec{w}^*, b^*$. After that, the decision function is provided by:

$$f(x) = \langle\vec{w}^*, \vec{x}\rangle + b^* \quad (9)$$

Equation (9) defines the problem as follows:

$$\max_{\alpha\in\mathbb{R}^n} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{l=1}^{n} \alpha_i\alpha_l y^i y^l \langle\vec{x}^i, \vec{x}^l\rangle \quad (10)$$

$$t.q \sum_{i=1}^{n} \alpha_i y^i = 0;\ \alpha_i \geq 0, i = 1, \dots, n$$

3.2.5. KNN (k Nearest Neighbors)

Given a distance *d* on a set of *n* labeled observations, $N_K(\vec{x})$ is the collection of $\vec{x}$'s k nearest neighbors in $D = \left\{(\vec{x}^i, y^i)_{i=1,\dots,n}\right\}$:

- When we use the majority vote to solve a classification problem, $\vec{x}$ we adopt the label that is most common among its K nearest neighbors.

$$f(x) = \underset{C}{\operatorname{argmax}} \sum_{i:.\vec{x}^i\in N_K(\vec{x})} \delta(y^i, C) \quad (11)$$

- The average of the labels of k nearest neighbors is used as the label for a regression problem:

$$f(x) = \frac{1}{K}\sum_{i:.\vec{x}^i \in N_K(\vec{x})} y^i \quad (12)$$

3.3. Instruments Utilized

3.3.1. Python

The Python 3.11.7 programming language was selected for the development of our algorithms due to its rich ecosystem of specialized libraries. Tools such as scikit-learn, TensorFlow, pandas, NumPy, matplotlib, and PyTorch greatly facilitated the implementation, testing, and optimization of the models. Python's straightforward syntax and accessibility also fostered collaboration within our research team, accelerating the development cycle.

3.3.2. Jupyter Notebook

The Integrated Development Environment (IDE) Jupyter Notebook 7.0.8 was used for this study. Specifically designed for data scientists and engineers, it offers advanced features such as integrated data visualization, real-time code analysis, and an interactive variable explorer. Its ability to directly integrate scientific and machine learning libraries (NumPy, SciPy, Matplotlib, scikit-learn) within the same environment significantly improved our workflow and enabled efficient model exploration.

4. Results and Discussion

4.1. Metrics for Evaluation

4.1.1. The Matrix of Confusion

An important technique for assessing a classification model's performance is a confusion matrix. By contrasting the predicted and actual values, you can see and comprehend the model's output. We can see the categorization model's advantages and disadvantages in depth thanks to the confusion matrix. By examining the data and computing the relevant metrics like accuracy, precision, recall score, specificity, and F1-score, the number of samples correctly classified as not belonging to the class is represented by True Negatives (TN), which are represented by formulas based on concepts like row 0 and column 0. False Positives (FP) is the number of samples that were mistakenly assigned to the class in Row 0, Column 1. The number of samples that belong to the class but are mistakenly identified as NOT belonging to it is represented by Line 1, Column 0's False Negatives (FN), and the number of samples that are correctly classified as belonging to the class is represented by Line 1, Column 1's True Positives (TP). As a result, we will apply the following metrics' formulas :

- $\mathrm{Pr}ecision = \frac{TP}{TP+FP}$ Which indicates that among all the samples that the model predicted as positive, what proportion were actually positive?
- $\mathrm{Re}call(sensitivity) = \frac{TP}{TP+FN}$ Which indicates what percentage of all the genuinely positive samples the model will accurately detect?
- $Specificity = \frac{TN}{TN+FP}$ Which outlines the percentage of all truly negative samples that the model will accurately classify as negative?
- $F1-score = \frac{2\times \mathrm{Pr}ecision\times \mathrm{Re}call}{\mathrm{Pr}ecision+\mathrm{Re}call}$ Which is the precision and recall harmonic mean. It offers a fair assessment of the model's performance, particularly when there is a class disparity.
- $Accuracy = \frac{TP+TN}{TP+FN+TN+FP}$ This represents the overall percentage of samples that were correctly classified.

We can gain a better understanding of the model's operation and, if required, make adjustments.

4.1.2. The Graph-Based Classification Report

The performance of the various categorization models employed in this work is depicted in this graph. Because it is a grouped bar chart, comparing three distinct metrics, named precision, recall, and F1 score for several categories, is simple. The percentage of accurate positive predictions among all of the model's optimistic predictions is known as accuracy.

The percentage of real positive cases that the model correctly identified is known as recall. The categories "micro avg," "macro avg," "weighted avg," and "samples avg" represent averages of performance metrics computed in various ways across all devices. The x-axis represents various pieces of equipment or classes (numbered 0 to 8) for which the model was evaluated. The F1 score is a harmonic average of precision and recall.

4.1.3. The ROC Curve

A useful graphical tool for assessing the effectiveness of a binary classification model, in this case, determining whether or not equipment is defective, is the Receiver Operating Characteristic (ROC) curve. The ratio of inaccurate optimistic predictions to all actual negative observations is shown by the x-axis (X) or False Positive Rate (FPR). It is better if it is lower. The True Positive Rate (TPR), represented by the y-axis (Y), is the ratio of accurate optimistic forecasts to all actual positive observations. It is better if it is higher.

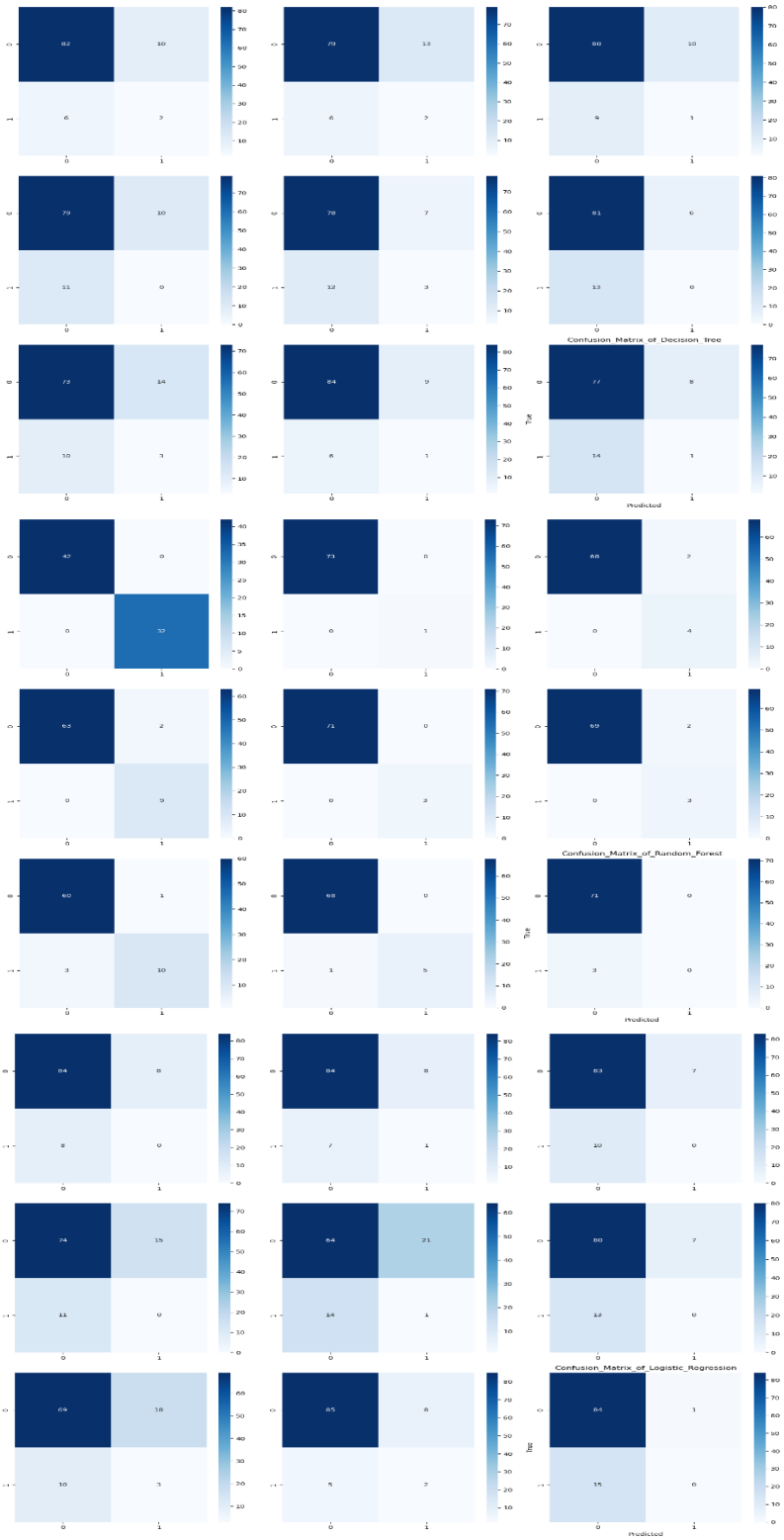
4.1.4. The Stratified Cross-Validation

Stratified cross-validation is an evaluation technique that preserves the proportion of classes in each subset (fold) of data used for training and testing. This is particularly important in the presence of class imbalance. It provides a robust estimate of the model's ability to generalize, reduces sampling bias, and helps detect overfitting. In the context of predictive maintenance, where failures are rare, it guarantees a meaningful and reliable evaluation [78].

4.2. Results of the Five Models' Performance Evaluation

4.2.1. Confusion Matrix

Figure 7 below displays the confusion matrix for our five models.



Following the completion of all the procedures related to precision, recall score, F1 score, specificity, and accuracy, the following tables represent the outcome :

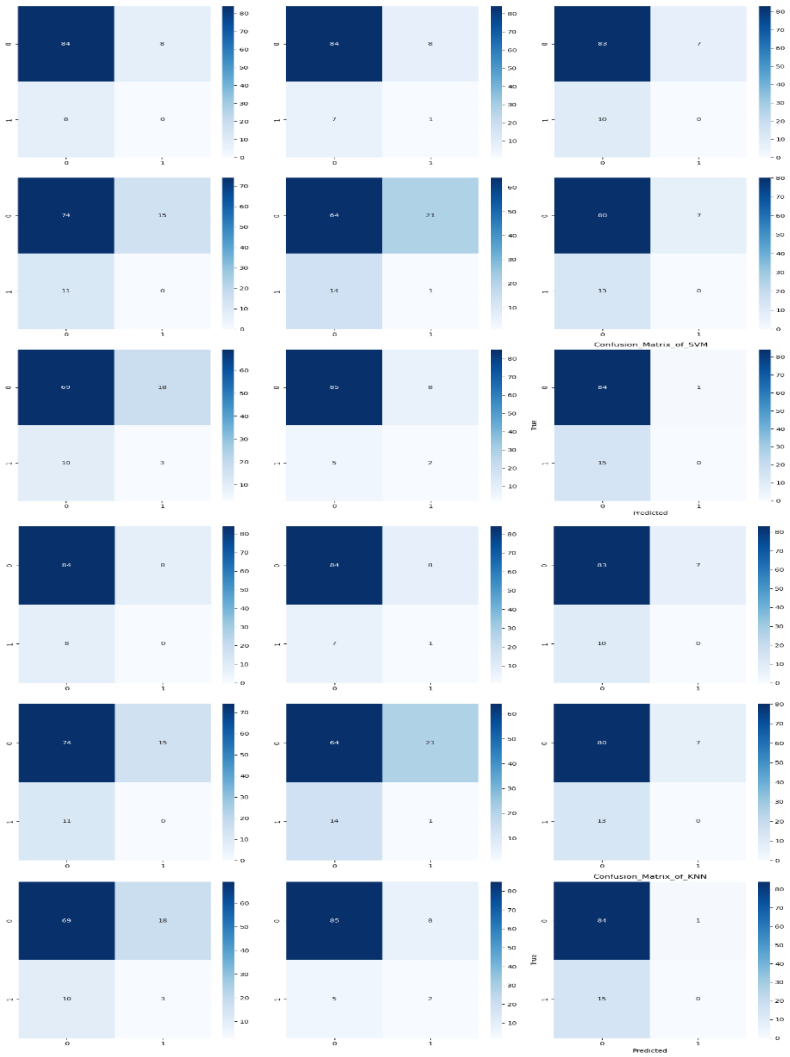


Fig. 7 Confusion matrix for our five models

Table 1. Overview of decision tree model metrics by class

Decision Tree					
Class	**Precision**	**Recall**	**Specificity**	**F1-Score**	**Accuracy**
Unpacker	0.16	0.25	0.89	0.20	0.84
Washer	0.13	0.25	0.86	0.17	0.81
Filler	0.09	0.10	0.89	0.16	0.81
Spotter	0.00	0.00	0.88	0.00	0.79
Cash boxer	0.30	0.20	0.92	0.24	0.81
Labeler	0.00	0.00	0.93	0.00	0.81
Coder	0.17	0.23	0.83	0.17	0.76
Normal mode	0.10	0.12	0.90	0.11	0.85
Gradient mode	0.11	0.06	0.91	0.07	0.78

Table 2. Overview of the random forest model

Random Forest					
Class	**Precision**	**Recall**	**Specificity**	**F1-Score**	**Accuracy**
Unpacker	0.96	1.00	0.97	0.96	0.98
Washer	1.00	1.00	1.00	1.00	1.00

Filler	0.80	1.00	0.98	0.88	0.98
Spotter	0.82	1.00	0.97	0.90	0.97
Cash boxer	1.00	0.67	1.00	0.81	0.98
Labeler	0.50	1.00	0.96	0.67	0.92
Coder	0.92	0.85	0.98	0.88	0.96
Normal mode	1.00	0.67	1.00	0.81	0.97
Gradient mode	0.00	0.00	1.00	0.00	0.96

Table 3. Synopsis of logistic regression model metrics by class

Logistic Regression					
Class	**Precision**	**Recall**	**Specificity**	**F1-Score**	**Accuracy**
Unpacker	0.00	0.00	0.91	0.00	0.84
Washer	0.11	0.12	0.91	0.12	0.85
Filler	0.00	0.00	0.92	0.00	0.83
Spotter	0.00	0.00	0.83	0.00	0.74
Cash boxer	0.04	0.06	0.75	0.04	0.65
Labeler	0.00	0.00	0.92	0.00	0.80
Coder	0.14	0.23	0.79	0.17	0.72
Normal mode	0.20	0.28	0.92	0.23	0.87
Gradient mode	0.00	0.00	0.98	0.00	0.84

Table 4. Metrics summary by class of SVM model

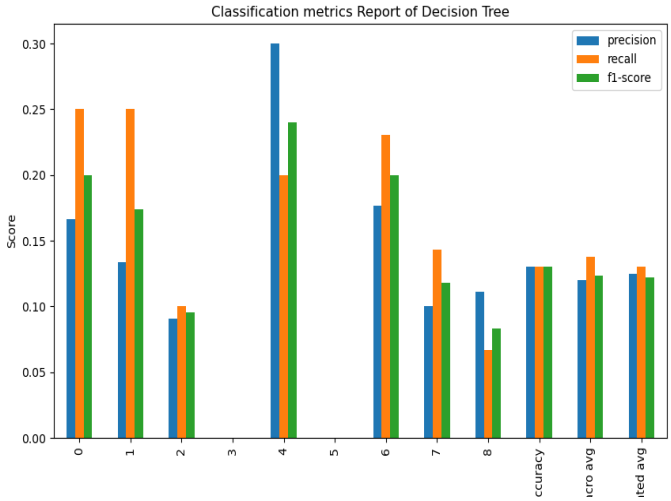
SVM					
Class	**Precision**	**Recall**	**Specificity**	**F1-Score**	**Accuracy**
Unpacker	0.00	0.00	0.91	0.00	0.84
Washer	0.11	0.12	0.91	0.12	0.85
Filler	0.00	0.00	0.92	0.00	0.83
Spotter	0.00	0.00	0.83	0.00	0.74
Cash boxer	0.04	0.06	0.75	0.04	0.65
Labeler	0.00	0.00	0.92	0.00	0.80
Coder	0.14	0.23	0.79	0.17	0.72
Normal mode	0.20	0.28	0.92	0.23	0.87
Gradient mode	0.00	0.00	0.98	0.00	0.84

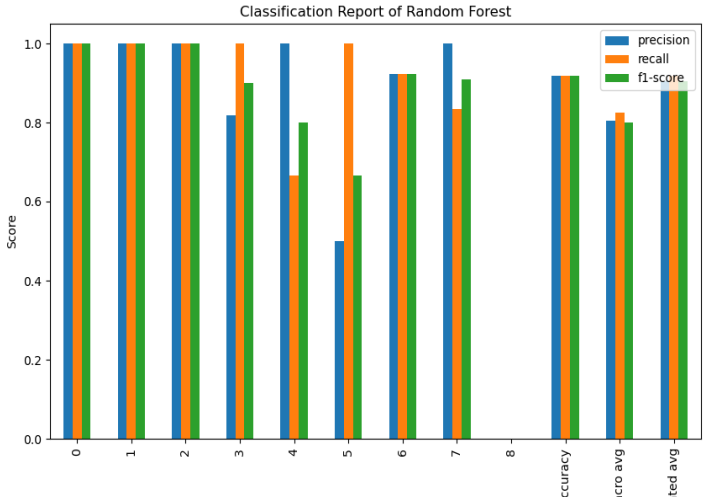
Table 5. Overview of KNN model metrics by class

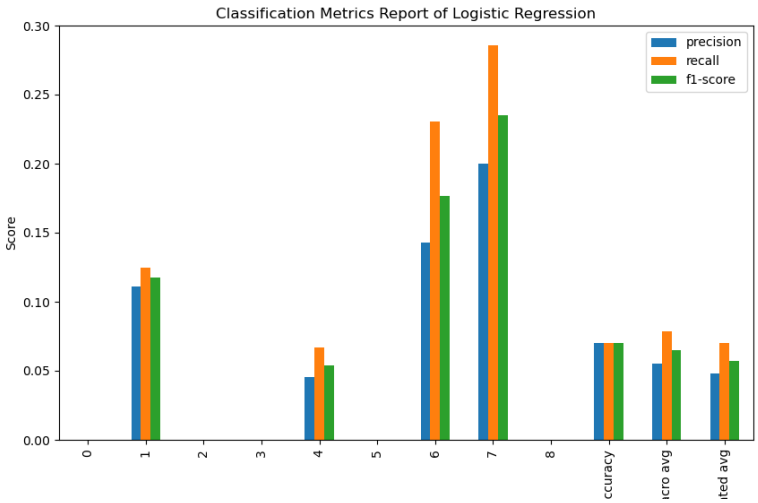
KNN					
Class	**Precision**	**Recall**	**Specificity**	**F1-Score**	**Accuracy**
Unpacker	0.00	0.00	0.91	0.00	0.84
Washer	0.11	0.12	0.91	0.12	0.85
Filler	0.00	0.00	0.92	0.00	0.83
Spotter	0.00	0.00	0.83	0.00	0.74
Cash boxer	0.04	0.06	0.75	0.04	0.65
Labeler	0.00	0.00	0.92	0.00	0.80
Coder	0.14	0.23	0.79	0.17	0.72
Normal mode	0.20	0.28	0.92	0.23	0.87
Gradient mode	0.00	0.00	0.98	0.00	0.84

4.2.2. Classification Report of the Five Models

The five-model histogram is shown in Figure 8 below.







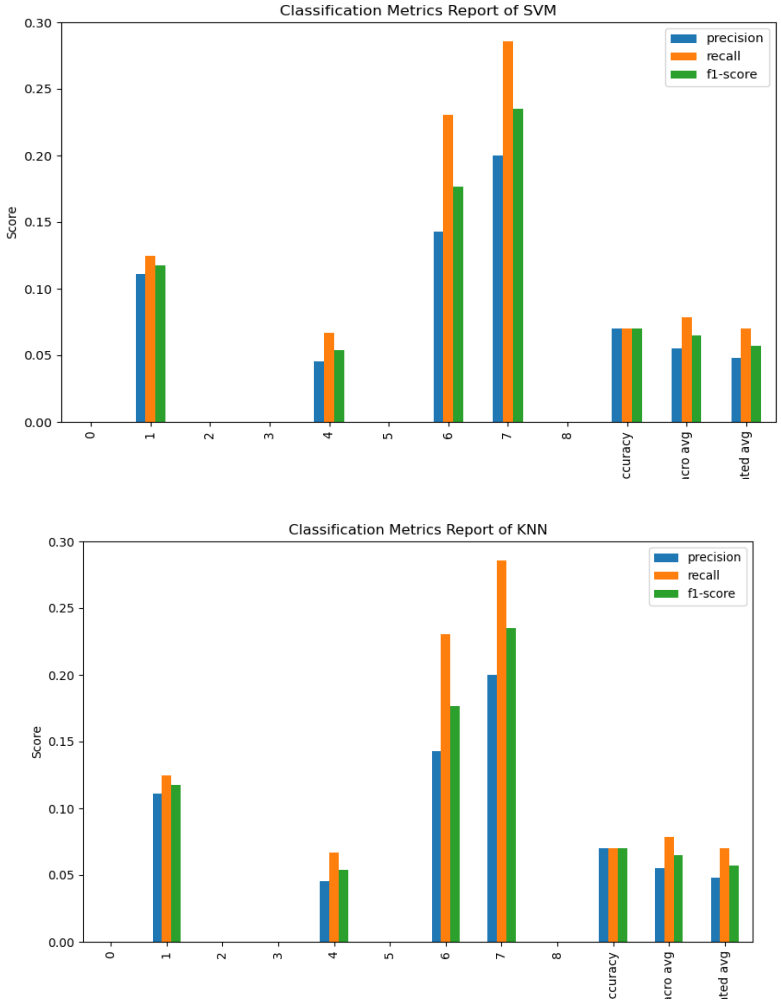
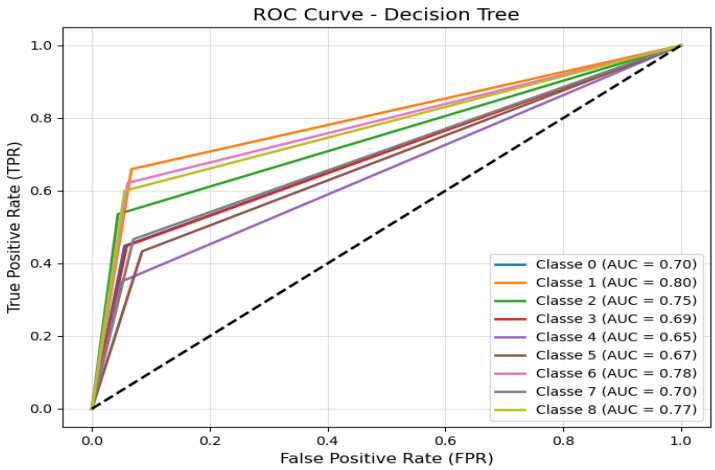
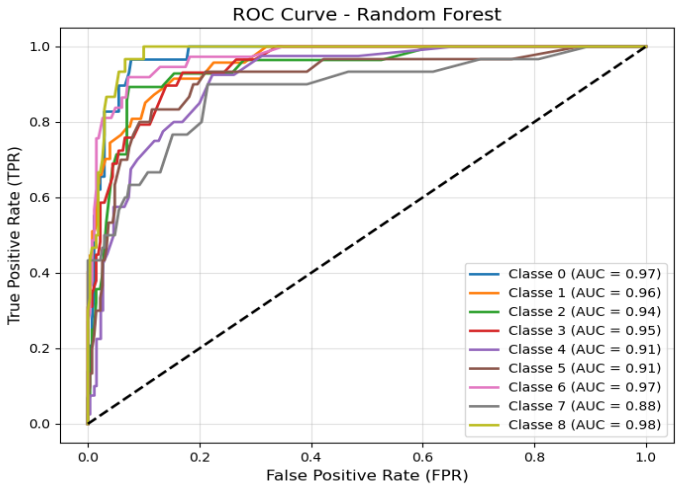


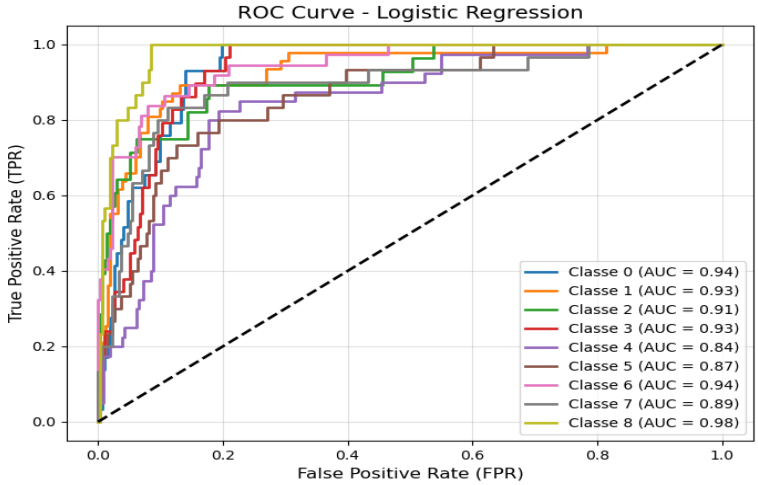
Fig. 8 Histogram of the five models

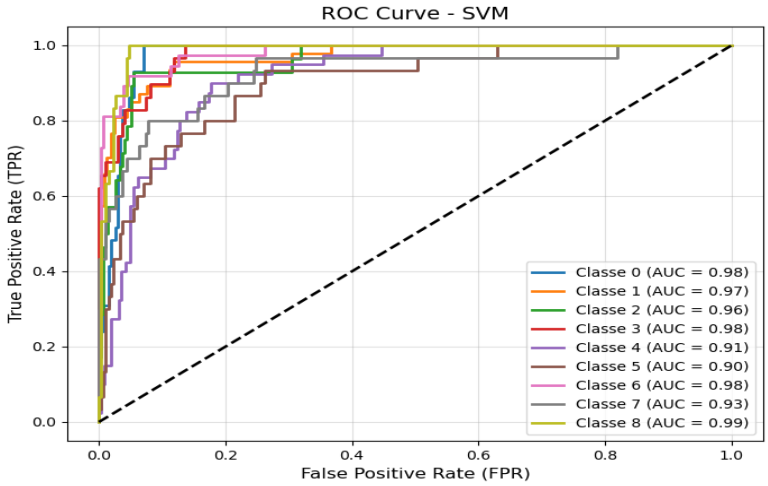
4.2.3. ROC Curve of the Five Models

The five models of the ROC Curve are shown in Figure 9 below.




ROC Curve - Random Forest
True Positive Rate (TPR)
False Positive Rate (FPR)
Classe 0 (AUC = 0.97)
Classe 1 (AUC = 0.96)
Classe 2 (AUC = 0.94)
Classe 3 (AUC = 0.95)
Classe 4 (AUC = 0.91)
Classe 5 (AUC = 0.91)
Classe 6 (AUC = 0.97)
Classe 7 (AUC = 0.88)
Classe 8 (AUC = 0.98)


ROC Curve - Logistic Regression
True Positive Rate (TPR)
False Positive Rate (FPR)
Classe 0 (AUC = 0.94)
Classe 1 (AUC = 0.93)
Classe 2 (AUC = 0.91)
Classe 3 (AUC = 0.93)
Classe 4 (AUC = 0.84)
Classe 5 (AUC = 0.87)
Classe 6 (AUC = 0.94)
Classe 7 (AUC = 0.89)
Classe 8 (AUC = 0.98)


ROC Curve - SVM
True Positive Rate (TPR)
False Positive Rate (FPR)
Classe 0 (AUC = 0.98)
Classe 1 (AUC = 0.97)
Classe 2 (AUC = 0.96)
Classe 3 (AUC = 0.98)
Classe 4 (AUC = 0.91)
Classe 5 (AUC = 0.90)
Classe 6 (AUC = 0.98)
Classe 7 (AUC = 0.93)
Classe 8 (AUC = 0.99)

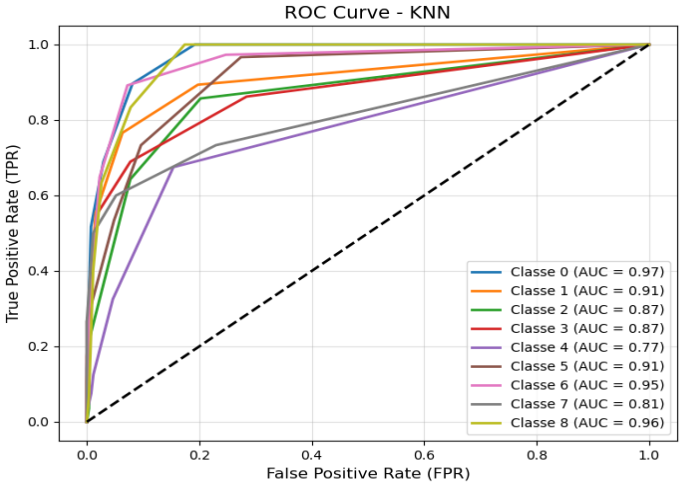
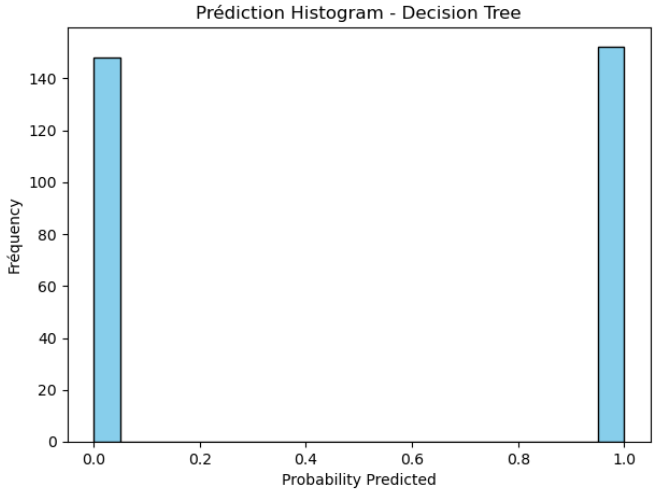
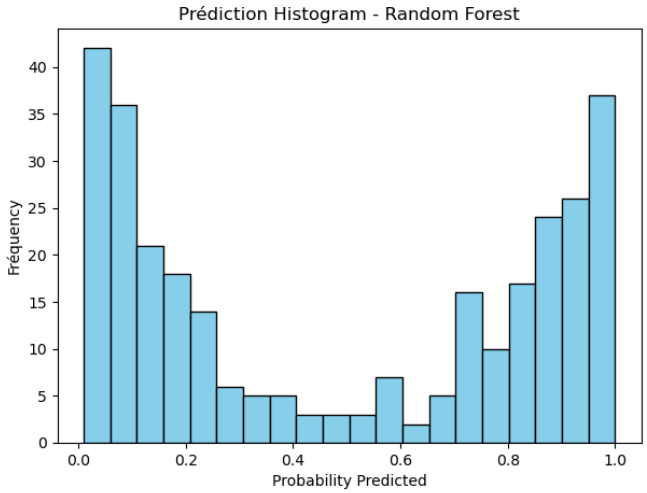


Fig. 9 The ROC curve for the five models

4.2.4. Comparison Results of Different Models

The histogram of the distribution probabilities of the various models is shown in Figure 10 below.





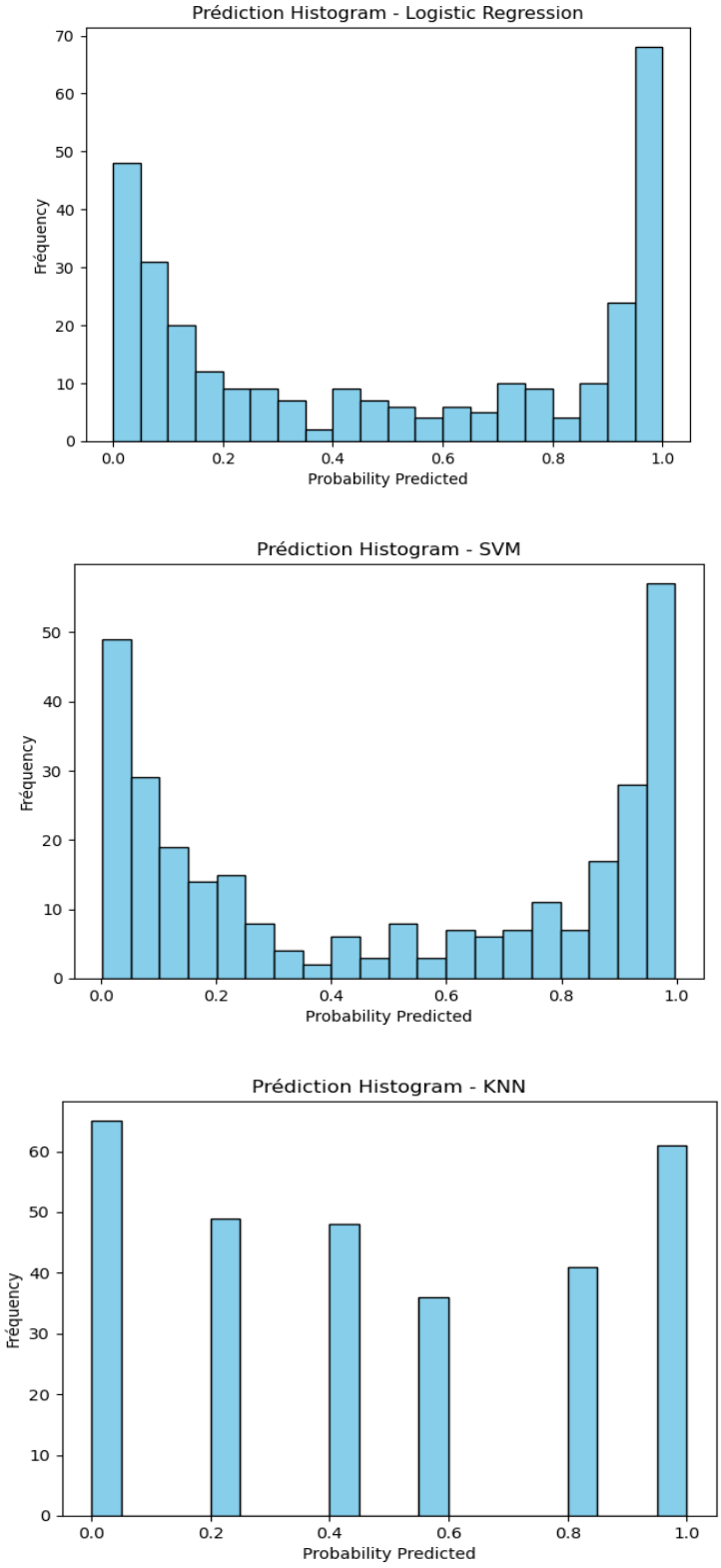
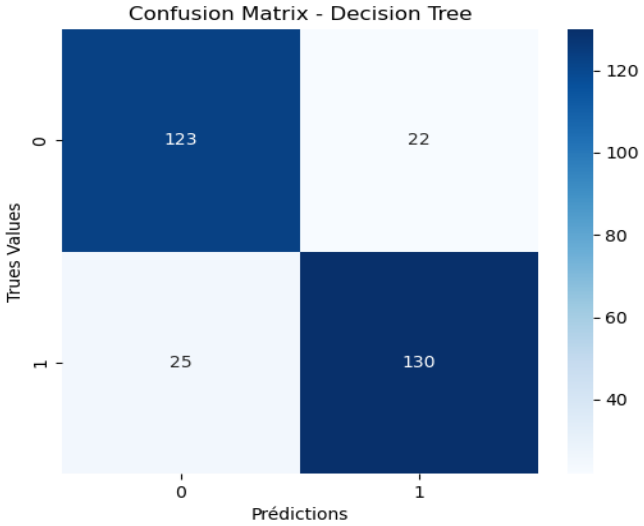
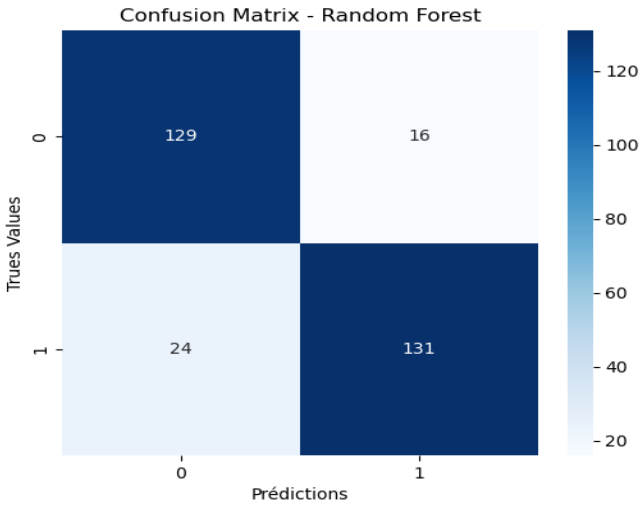
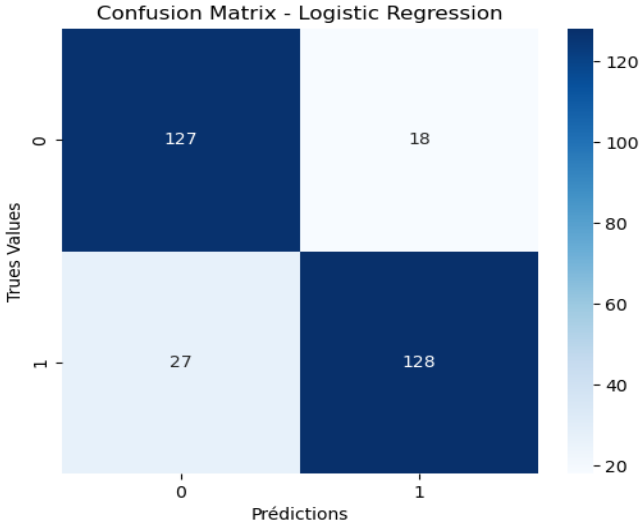


Fig. 10 Predicted probability distribution histogram of the five models

4.2.5. Comparison of the Final Model (Based on Overall Accuracy)







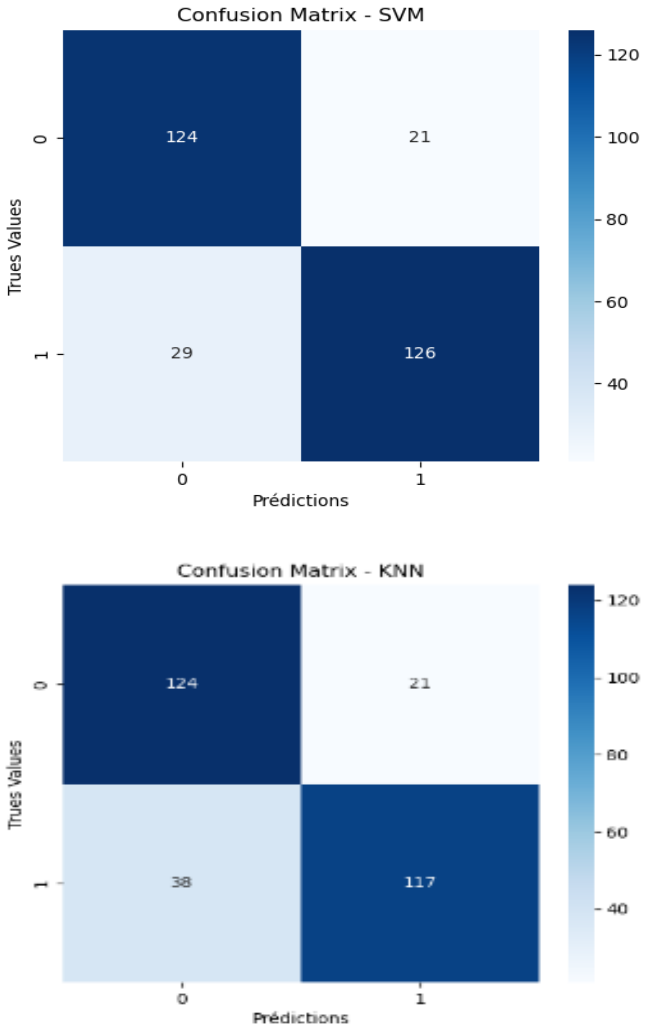


Fig. 11 Probability distribution histogram of the five models

4.2.6. Model Comparison Using the ROC Curve

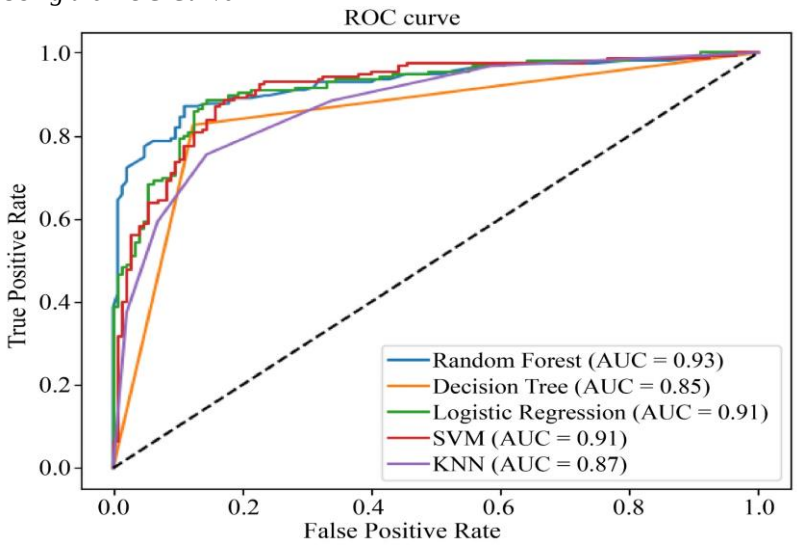


Fig. 12 Overall ROC Curve for the various models

4.2.7. Model Comparison using the Stratified Cross-Validation

Figure 13 presents the results of the stratified cross-validation as a histogram.

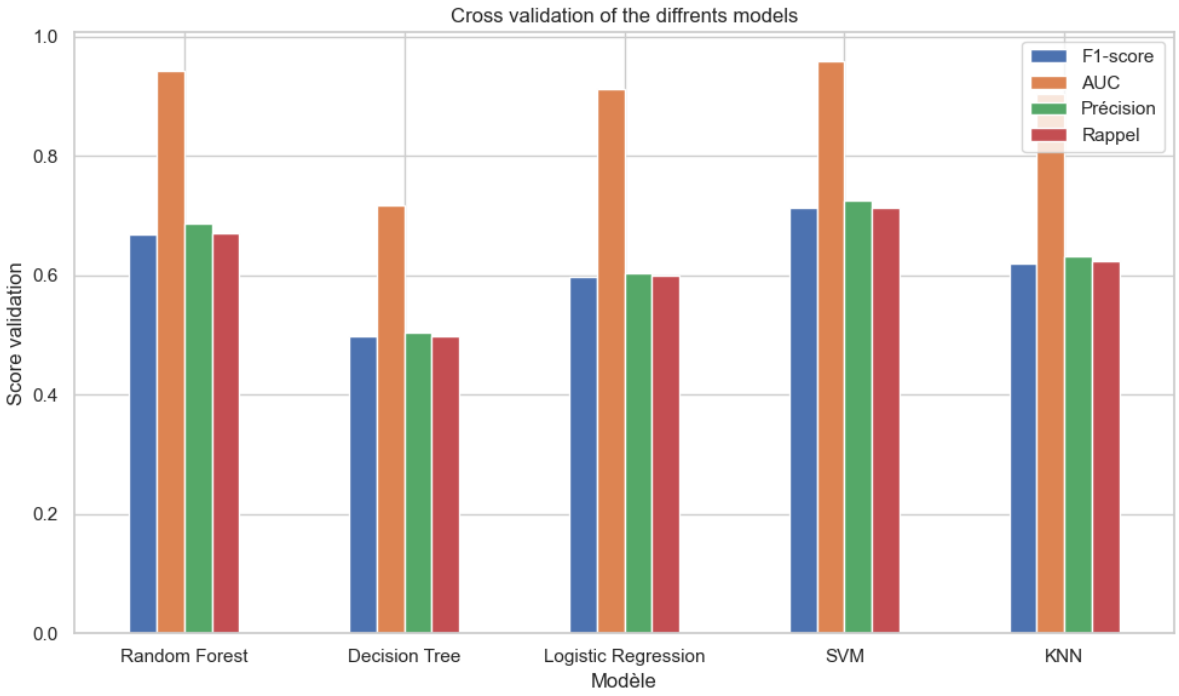


Fig. 13 Stratified Cross-validation histogram of the five models

4.2.8. A comparison between Contemporary Boosting Models and the Best Random Forest Model

Finally, Figure 14 compares the performance of our best model (Random Forest) with more recent boosting algorithms (Gradient Boosting, AdaBoost, XGBoost, LightGBM), highlighting their industrial applicability.

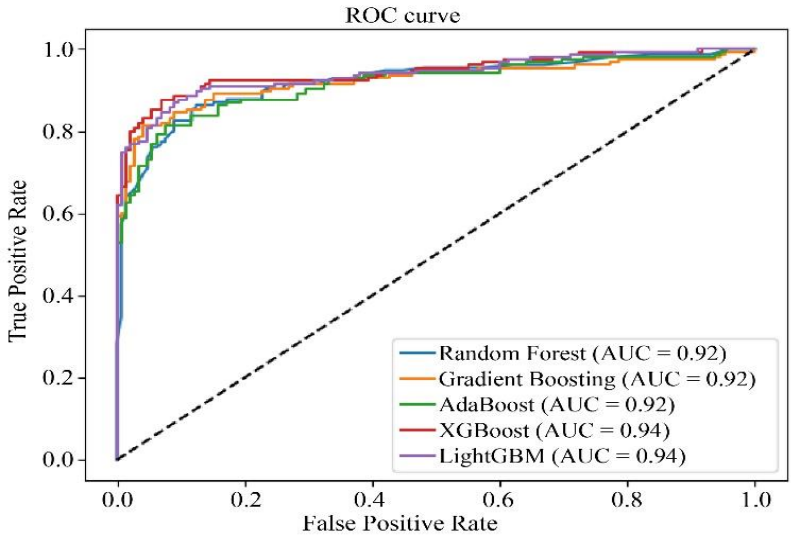


Fig. 14 Comparison of the best random forest model with recent boosting models

4.3 Discussion of the Confusion Matrix

4.3.1. Confusion Matrix of the Decision Tree

With greater F1 scores, the "Cash Boxer" and "Unpacker" classes appear to fare better overall. On this dataset, the "Casher" class in particular exhibits excellent accuracy and specificity. Certain classes, such as "Coder," have a high precision but a moderate recall, meaning that while the model is mostly correct when predicting "Coder," it occasionally

fails to identify real-world occurrences of the class. On the other hand, recall might be greater than precision for other classes. With 0% precision and recall, the "Sealer and Labeler" classes perform terribly. This implies that the model is experiencing trouble accurately recognizing "Scanner and Labeler" instances. For the majority of classes, the specificity is comparatively high, indicating that the model is generally effective in detecting instances that do not fall into the positive class.

4.3.2 Confusion Matrix of Random Forest

The precision, recall, and F1-Score of the "Unpacker," "Washer," and "Coder" classes run flawlessly, ranging from 0.85 to 1.00. Very high scores are also displayed by the "Normal Mode" and "Casher" classes. The model accurately detects all actual occurrences of the majority of classes, where the recall is 1.00, and the precision is less than 1.00. However, this results in a few false positives, which lowers precision a little bit in the "Filler" and "Splotter" situations.

With a recall and an F1-score of 0, the Random Forest model has trouble with the "Degraded Mode" class. This suggests that no instances of this class were accurately identified by the model. Since there were no successful predictions for this class, the accuracy is likewise zero. The model is effective at identifying occurrences that do not belong to the positive class, as evidenced by the generally very high specificity for all classes.

4.3.3. Confusion Matrix of Logistic Regression, SVM, and KNN

Precision and recall are zero for a number of classes (Unpacker, Filler, Spotter, Labeler, and Gradient Mode), suggesting that these models are unable to detect positive cases accurately. F1 scores are typically extremely low, indicating that these models struggle to use the one-versus-all method to rank many steps in our process. These models are generally effective at correctly identifying negative examples, as evidenced by the relatively high specificity for the majority of classes, despite low precision and recall. The “Encoder” and “Normal Mode” classes have slightly higher precision and recall than the others, while they are still low.

4.4. Discussion of the Classification Report

4.4.1. The Decision Tree Model Classification Report

Some classes, especially class 3 (Splotter) and, to a lesser extent, class 5 (Labeler), are difficult for the decision tree model to handle. The model performs exceptionally well for the Encoder0, Unpacker1, and Mireuse5 classes, exhibiting excellent recall and precision. Performances in the Labeler3, Filler6, and Normal Mode7 courses are mediocre.

4.4.2. The Random Forest Model Classification Report

As evidenced by the rise in precision, recall, and F1 scores, the Random Forest model outperforms the Decision Tree for the majority of classes. Random Forest performs better across classes because it is more resilient and less likely to overfit than a Decision Tree. Even with the Random Forest's development, Class 8 is still the hardest to forecast.

4.4.3. The Logistic Regression Model Classification Report

Depending on the classifications, the logistic regression model displays varying performance. He excels in some classes (1, 6, 7) but struggles in others (0, 2, 3, 5, 8).

Overall, we find that logistic regression performs worse than Decision Tree and Random Forest Models. This is due to the fact that logistic regression is a linear model, whereas decision trees and random forests are better suited to capture non-linear correlations in data.

4.4.4. The SVM Model Classification Report

Precision, recall, and a low F1 score indicate that the model has trouble handling classes (0, 2, 3, 5, 8) (Unboxer, Filler, Mirage, Labeler, and Gradient Mode). With high scores, the model appears to work effectively for classes 6 (Coder) and 7, in particular (Normal mode). The micro average is likewise low, confirming the model's challenges, and the macro average is extremely low, suggesting that the model is having trouble overall with all classes.

4.4.5. The KNN Model Classification Report

The model struggles with classes (0, 2, 3, 5, 8) (Unboxer, Filler, Mirage, Labeler, and Gradient Mode), as evidenced by precision, recall, and a poor F1 score. The model seems to function well for classes 6 (Coder) and 7, especially in normal mode, with high scores. The macro average is incredibly low, indicating that the model is generally struggling with all classes, and the micro average is similarly low, confirming the model’s difficulties.

4.5. Discussion on the Roc curve

4.5.1. The ROC Curve of the Decision Tree Model

It seems that the Decision Tree model has trouble accurately forecasting the condition of the majority of the individual pieces of equipment, especially the cash box (Class 4). Although the coder (Class 6) performs the best, even their AUC is not very high. Additionally, the regular (Class 7) and deteriorated (Class 8) operational modes can only be predicted with a modest degree of accuracy.

4.5.2. The ROC Curve of the Random Forest Model

The Random Forest model is very good at determining the states of the coding machine (Class 6), depalletizer (Class 0), and degraded operational mode (Class 8). Additionally, it works incredibly well with the Class 1 bottle washer, Class 2 filling machine, and Class 3 inspection machine. Both the labeler (Class 5) and case packer (Class 4) continue to have excellent predictions. Although it is harder for the model to predict the regular operational mode (Class 7) than the other special states, the performance is still impressive.

4.5.3. The ROC Curve of the Logistic Regression Model

In addition to determining the states of the unpacker (Class 0) and the coder (Class 6), the Logistics Regression model is especially effective at determining the degraded mode (Class 8). Additionally, it performs admirably for the candulator (Class 3) and washer (Class 1). While still within an acceptable range, the case packer's (Class 4) and labeler's (Class 5) performance is marginally worse. Additionally, the typical mode (Class 7) prediction is accurate.

4.5.4. The ROC Curve of the SVM Model

The deteriorated mode (Class 8), unpacker (Class 0), sighter (Class 3), and coder (Class 6) states are all exceptionally well-identified by the SVM model. Additionally, it performs exceptionally well in regular mode (Class 7), filler (Class 2), and washer (Class 1). Although significantly worse, the case packer (Class 4) and labeler (Class 5) nevertheless have excellent performance.

4.5.5. The ROC Curve of the KNN Model

The KNN model performs exceptionally well in detecting the degraded mode (Class 8), coder (Class 6), and unpacker (Class 0). It provides both the labeler (Class 5) and the washer (Class 1) with good performance. Filler (Class 2), candulator (Class 3), case packer (Class 4), and regular mode (Class 7) all have poorer performance.

4.6. Discussion of Predicted Probability Distribution

With a noticeable bimodal distribution and a trough at probability 0.5, the Random Forest model's prediction probability histogram shows a propensity to provide forecasts with high confidence. This implies that the model frequently has a high degree of confidence in the classifications it makes for that particular class. However, the Decision Tree model shows a binary prediction behavior, with nearly all of the probabilities being either 0 or 1. This is consistent with decision trees, which assign a probability of 0 or 1 in their "pure" terminal leaves. This feature stands in stark contrast to the Random Forest model's more complex distribution, which we previously examined. Compared to the Decision Tree, the Logistic Regression model has a more complex distribution with a notable presence of intermediate probabilities, suggesting the capacity to convey uncertainty. It appears less extreme than the Random Forest, despite the concentration near the ends. More freedom to interpret predictions and modify the classification threshold is offered by this distribution. We observe that the SVM has a propensity to generate predictions with a distribution that implies bimodality and a comparatively high degree of confidence. While intermediate probabilities do exist, they are less frequent than those close to 0 or 1. This distribution illustrates how the SVM uses a separation margin to determine classification. The KNN model is characterized by significant frequencies for intermediate probabilities in addition to peaks at the endpoints (0.0 and 1.0). In contrast to models that exhibit more pronounced bimodality, this indicates a higher frequency of uncertainty in predictions and reflects the neighborhood-based structure of the model. This distribution may make it more challenging to choose the categorization threshold.

4.7. Discussion on the Overall Accuracy of the Confusion Matrices of our Different Models

Of the five, Random Forest has the highest overall accuracy (86.7%), followed by Logistic Regression (85.0%), Decision Tree (84.3%), SVM (83.3%), and Random Forest (86.7%). KNN (80.3%) has the lowest overall accuracy. The Random Forest is the best model for this binary classification job based on total accuracy. While KNN exhibits the lowest efficiency, Logistic Regression, Decision Tree, and SVM follow with marginally worse performance. Note that this conclusion is predicated on overall accuracy. Class-specific F1 scores may be the subject of future research, especially if the costs of false positives and false negatives vary by class. Nonetheless, Random Forest seems to outperform the other models in terms of sheer overall efficiency.

4.8. Discussion on Comparing the ROC Curve on Models

Following the use of many models, the data indicates that the best models for forecasting equipment failures and states in your brewery system are Random Forest, SVM, and Logistic Regression. Based on the total AUC, the Random Forest model seems to be marginally better. KNN may be less dependable across all types of failures and equipment conditions, despite its respectable performance. The Decision Tree appears to be the least successful and may need a significant overhaul or an alternative strategy.

4.9. Discussion on Comparing the Cross-Validation

The cross-validation results revealed distinct performance profiles across models: *Random Forest* achieved the highest scores across all metrics, with a particularly strong recall and AUC, indicating excellent class separation and minimal false negatives. *Logistic regression* showed stable and interpretable performance, making it a reliable choice for regulatory or traceability-sensitive applications. *Decision Tree* and *KNN* exhibited lower generalization capacity, with reduced F1 and AUC scores, suggesting sensitivity to data structure and potential overfitting. *SVM* delivered moderate results, but its effectiveness may depend on further tuning and preprocessing steps such as feature scaling or class balancing.

4.10. Discussion on Comparing Contemporary Boosting Models and the Best Random Forest Model

Significant performance and industrial adaptation differences are *found* when Random Forest and boosting models (Gradient Boosting, AdaBoost, XGBoost, and LightGBM) are compared in the context of predictive maintenance in breweries. With an AUC of 0.93, Random Forest is a dependable option for industrial settings where stability is crucial because it is resilient and not overly sensitive to hyperparameters. However, because boosting

models can manage class imbalances, which are common in failure data, and capture complex relationships, they perform somewhat better in accuracy than Random Forest (AUC = 0.94). Examples of these models include XGBoost and LightGBM. Boosting models offer a strategic advantage in brewing, where malfunctions are uncommon but crucial, as they enable more accurate identification of weak signals and more precise calibration of probabilities, both of which are necessary for setting intervention priorities. Specifically, LightGBM combines high precision, low memory usage, and speed of execution, making it ideal for industrial settings with limited resources. In conclusion, LightGBM is the most appropriate model for efficient predictive maintenance in breweries because of its accuracy, speed, and skillful handling of unbalanced data, even though Random Forest is still a reliable and interpretable option.

5. Conclusion

In summary, the Random Forest model seems to be a good classifier for this binary problem based on its overall confusion matrix. However, taking into account the problem's context and contrasting its performance with that of other models would be necessary for a more thorough assessment. Based on a direct comparison of their total confusion matrices, the Decision Tree model appears to be marginally less effective than the Random Forest model on this binary classification problem, despite achieving respectable accuracy and F1 scores. In this overall assessment, Random Forest produced fewer classification errors. KNN has the lowest efficiency, followed by Decision Tree and Logistic Regression, both of which perform marginally worse. Note that this conclusion is predicated on overall accuracy. Class-specific F1 scores may be the subject of future research, especially if the costs of false positives and false negatives vary by class. Nonetheless, Random Forest seems to outperform the other models in terms of sheer overall efficiency. It would be prudent to concentrate on improving and implementing Random Forest, SVM, or Logistic Regression models in the future. We may select one of these top-performing models based on the particular requirements of our brewery (such as interpretability and computational cost). Choosing the best model for your brewery's predictive maintenance plan will require additional analysis utilizing additional metrics like precision, recall, F1 score, and the cost of false positives and false negatives.

Declarations

The authors did not receive support from any organization for the submitted work. The authors have no competing interests to declare that are relevant to the content of this article.

Acknowledgments

The authors would like to thank the brewing production industry SOBRAGA and the Laboratory of Technology and Applied Sciences, University of Douala, Cameroon.

References

[1] Mike Hobday, "Product Complexity, Innovation and Industrial Organisation," *Research Policy*, vol. 26, no. 6, pp. 689-710, 1998. [CrossRef] [Google Scholar] [Publisher Link]

[2] Yang Lu, "Industry 4.0: A Survey on Technologies, Applications and Open Research Issues," *Journal of Industrial Information Integration*, vol. 6, pp. 1-10, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[3] Vasja Roblek, Maja Meško, and Alojz Krapež, "A Complex view of Industry 4.0," *Sage Open*, vol. 6, no. 2. pp. 1-11, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[4] Ulrich Schmoch et al., "Linking Technology Areas to Industrial Sectors," *Final Report to the European Commission*, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[5] Zhijia You, and Lingjun Feng, "Integration of Industry 4.0 Related Technologies in Construction Industry: A Framework of Cyber-Physical System," *IEEE Access*, vol. 8, pp. 122908-122922, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Jay Lee et al., "Industrial Artificial Intelligence for Industry 4.0-based Manufacturing Systems," *Manufacturing Letters*, vol. 18, pp. 20-23, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ardalan F. Khalil, and Sarkawt Rostam, "Machine Learning-Based Predictive Maintenance for Fault Detection in Rotating Machinery: A Case Study," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13181-13189, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Ameeth Kanawaday, and Aditya Sane, "Machine Learning for Predictive Maintenance of Industrial Machines Using IoT Sensor Data," *2017 8th IEEE International Conference on Software Engineering and Service Science*, pp. 87-90, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[9] Bo-hu Li et al., "Applications of Artificial Intelligence in Intelligent Manufacturing: A Review," *Frontiers of Information Technology and Electronic Engineering*, vol. 18, no. 1, pp. 86-96, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[10] Noor A. Mohammed et al., "Performance Analysis of Different Machine Learning Algorithms for Predictive Maintenance," *Al-Khwarizmi Engineering Journal*, vol. 20, no. 2, pp. 26-38, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Mohammad Abidur Rahman et al., "Enabling Intelligent Industrial Automation: A Review of Machine Learning Applications with Digital Twin and Edge AI Integration," *Automation*, vol. 6, no. 3, pp. 1-35, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[12] Max Kuhn, and Kjell Johnson, *Measuring Performance in Classification Models*, Applied Predictive Modeling, Springer, New York, NY, pp. 247-273, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[13] R. Keith Mobley, *An Introduction to Predictive Maintenance*, 2nd ed., Elsevier, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[14] Dorina Weichert et al., "A Review of Machine Learning for the Optimization of Production Processes," *The International Journal of Advanced Manufacturing Technology*, vol. 104, no. 5-8, pp. 1889-1902, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] Tobi Michael Alabi et al., "A Review on the Integrated Optimization Techniques and Machine Learning Approaches for Modeling, Prediction, and Decision Making on Integrated Energy Systems," *Renewable Energy*, vol. 194, pp. 822-849, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Ogagaoghene Uzezi Idhalama, and Elisha Ondieki Makori, "Artificial Intelligence, Deep Learning, Machine Learning, Robotics and Digital Transformation: Applications, Implications and Future," *Ukrainian Journal of Educational Studies and Information Technology*, vol. 12, no. 3, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Abbas Sharifi, Mohsen Ahmadi, and Ali Ala, "The Impact of Artificial Intelligence and Digital Style on Industry and Energy Post-Covid-19 Pandemic," *Environmental Science and Pollution Research*, vol. 28, no. 34, pp. 46964-46984, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Anirbid Sircar et al., "Application of Machine Learning and Artificial Intelligence in Oil and Gas Industry," *Petroleum Research*, vol. 6, no. 4, pp. 379-391, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Gian Antonio Susto et al., "Machine Learning for Predictive Maintenance: A Multiple Classifier Approach," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 812-820, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[20] Yassine Barakat et al., "What Contributions of Artificial Intelligence in Innovation?," *E3S Web of Conferences*, vol. 234, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] David Gonzalez-Jimenez et al., "Data-Driven Fault Diagnosis for Electric Drives: A Review," *Sensors*, vol. 21, no. 12, pp. 1-33, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] Nuritdin Yuldoshev, Bobir Tursunov, and Saidmuhtor Qozoqov, "Use of Artificial Intelligence Methods in Operational Planning of Textile Production," *Journal of Process Management and New Technologies*, vol. 6, no. 2, pp. 41-51, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[23] Timothée Kombé, "*Modeling Fault Propagation in Production Systems*," Phd Thesis, INSA de Lyon, 2011. [Google Scholar] [Publisher Link]

[24] Fan Yang, and Deyun Xiao, "Progress in Root Cause and Fault Propagation Analysis of Large-Scale Industrial Processes," *Journal of Control Science and Engineering*, vol. 2012, no. 1, pp. 1-10, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[25] Xi Chen, and Jian Jiao, "A Fault Propagation Modeling Method based on a Finite State Machine," *2017 Annual Reliability and Maintainability Symposium (RAMS)*, Orlando, FL, USA, pp. 1-7, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[26] Zhi Hou, and Zhonghua Yu, "Two-Layer Model of Equipment Fault Propagation in Manufacturing System," *Quality and Reliability Engineering International*, vol. 37, no. 2, pp. 743-762, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[27] Cheng-Yuan Sun et al., "A Quality-Related Fault Detection Method based on the Dynamic Data-Driven Algorithm for Industrial Systems," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3942-3952, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[28] Shen Yin et al., "Data-based Techniques Focused on Modern Industry: An Overview," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 1, pp. 657-667, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[29] Kai Zhong, Min Han, and Bing Han, "Data-Driven based Fault Prognosis for Industrial Systems: A Concise Overview," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 330-345, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[30] Sarahi Aguayo-Tapia et al., "Physical Variable Measurement Techniques for Fault Detection in Electric Motors," *Energies*, vol. 16, no. 12, pp. 1-21, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[31] Steven X. Ding, *Advanced Methods for Fault Diagnosis and Fault-Tolerant Control*, 1st ed., Springer Berlin, Heidelberg, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[32] Anam Abid, Muhammad Tahir Khan, and Javaid Iqbal, "A Review on Fault Detection and Diagnosis Techniques: Basics and Beyond," *Artificial Intelligence Review*, vol. 54, no. 5, pp. 3639-3664, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[33] Yasser Gritli, "*Diagnosis and Fault Detection in Electrical Machines and Drives based on Advanced Signal Processing Techniques*," Dissertation Thesis, University of Bologna, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[34] Wenjin Yu et al., "A Global Manufacturing Big Data Ecosystem for Fault Detection in Predictive Maintenance," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 183-192, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[35] Nagdev Amruthnath, and Tarun Gupta, "A Research Study on Unsupervised Machine Learning Algorithms for Early Fault Detection in Predictive Maintenance," *2018 5th International Conference on Industrial Engineering and Applications (ICIEA)*, Singapore, pp. 355-361, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[36] Alexios Lekidis et al., "Predictive Maintenance Framework for Fault Detection in Remote Terminal Units," *Forecasting*, vol. 6, no. 2, pp. 239-265, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[37] Lai-Chang Zhang et al., "Additive Manufacturing of Titanium Alloys by Electron Beam Melting: A Review," *Advanced Engineering Materials*, vol. 20, no. 5, pp. 1-90, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[38] Antonella Sola, and Alireza Nouri, "Microstructural Porosity in Additive Manufacturing: The Formation and Detection of Pores in Metal Parts Fabricated by Powder Bed Fusion," *Journal of Advanced Manufacturing and Processing*, vol. 1, no. 3, pp. 1-21, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[39] Nan Liu et al., "Fault Detection and Diagnosis using Bayesian Network Model Combining Mechanism Correlation Analysis and Process Data: Application to Unmonitored Root Cause Variables Type Faults," *Process Safety and Environmental Protection*, vol. 164, pp. 15-29, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[40] Mohamed Amine Atoui, "*Contributions to Fault Detection and Diagnosis in Systems by Bayesian Networks*," Doctoral Dissertation, University of Angers, 2015. [Google Scholar] [Publisher Link]

[41] Enerst Edozie et al., "Artificial Intelligence Advances in Anomaly Detection for Telecom Networks," *Artificial Intelligence Review*, vol. 58, no. 4, pp. 1-40, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[42] Marius Minea, Cătălin Marian Dumitrescu, and Viviana Laetitia Minea, "Intelligent Network Applications Monitoring and Diagnosis Employing Software Sensing and Machine Learning Solutions," *Sensors*, vol. 21, no. 15, pp. 1-32, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[43] Sara Ahmed Hazim, and Ahmed Faleh Mahmood, "Developments in Optical Fiber Network Fault Detection Methods: An Extensive Analysis," *International Journal of Computing and Digital Systems*, vol. 17, no. 1, pp. 1-15, 2025. [CrossRef] [Google Scholar]

[44] Serge Romaric Tembo Mouafo, "*Applications of Artificial Intelligence to the Detection and Isolation of Multiple Faults in a Telecommunications Network*," Doctoral Dissertation, Ecole nationale supérieure Mines-Télécom Atlantique, 2017. [Google Scholar] [Publisher Link]

[45] ChukwuNonso H. Nwokoye, and V. Madhusudanan, 'Epidemic Models of Malicious-Code Propagation and Control in Wireless Sensor Networks: An Indepth Review," *Wireless Personal Communications*, vol. 125, no. 2, pp. 1827-1856, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[46] F. Ongowe et al., "Adaptation of Epidemiological Models for the Analysis of Failure Propagation in a Production Resource," *Industrial Engineering and Production*, vol. 1, no. 1, pp. 1-15, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[47] Yan Wang et al., "Propagation Dynamics of an Epidemic Model with Heterogeneous Control Strategies on Complex Networks," *Symmetry*, vol. 16, no. 2, pp. 1-22, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[48] Aitouche Moh-Amokrane, Baddari Kamel, and Djeddi Mabrouk, Mathematical Modeling of Epidemic Propagation in Memory of all the Victims of the Covid-19 Pandemic, Office of University Publications, 2025. [Online]. Available: https://fhc.univ-boumerdes.dz/wp-content/uploads/2020/04/Mod%C3%A9lisation-math%C3%A9matique-de-propagation-dune-%C3%A9pid%C3%A9mie.pdf

[49] Shazia Kanwal, and Somchat Jiriwibhakorn, "Advanced Fault Detection, Classification, and Localization in Transmission Lines: A Comparative Study of ANFIS, Neural Networks, and Hybrid Methods," *IEEE Access*, vol. 12, pp. 49017-49033, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[50] Shazia Kanwal, and Somchat Jiriwibhakorn, "Artificial Intelligence based Faults Identification, Classification, and Localization Techniques in Transmission Lines-A Review," *IEEE Latin America Transactions*, vol. 21, no. 12, pp. 1291-1305, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[51] R. Resmi et al., "Detection, Classification and Zone Location of Fault in Transmission Line using Artificial Neural Network," *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Coimbatore, India, pp. 1-5, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[52] Zoungrana Wend-Benedo Arnaud Bienvenue, "*Application of Machine Learning Algorithms for the Detection of Bearing Defects on Rotating Machines Within the Framework of Industry 4.0*," Master's Thesis, University of Quebec at Chicoutimi, 2020. [Google Scholar] [Publisher Link]

[53] A.J. Robinson, "*Development of a Predictive Maintenance Model which Provides Fault Identification and Diagnostics on Electrical Gearmotor Systems*," Master Thesis, Delft University of Technology, 2022. [Google Scholar] [Publisher Link]

[54] Md Naeem Hossain, Md Mustafizur Rahman, and Devarajan Ramasamy, "Artificial Intelligence-Driven Vehicle Fault Diagnosis to Revolutionize Automotive Maintenance: A Review," *CMES-Computer Modeling in Engineering & Sciences*, vol. 141, no. 2, pp. 951-966, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[55] Gyuwon Kim et al., "Neural Fault Observer-Based On-Board Prognostics for Automotive Electric Power Steering Systems," *IEEE Access*, vol. 13, pp. 42312-42321, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[56] Jaouher Ben Ali et al., "Application of Empirical Mode Decomposition and Artificial Neural Network for Automatic Bearing Fault Diagnosis based on Vibration Signals," *Applied Acoustics*, vol. 89, pp. 16-27, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[57] Faiza Belhachat, Cherif Larbes, and Rachid Bennia, "Recent Advances in Fault Detection Techniques for Photovoltaic Systems: An Overview, Classification and Performance Evaluation," *Optik*, vol. 306, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[58] Yannick Faula, Véronique Eglin, and Stéphane Bres, "One-Class Detection and Classification of Defects on Concrete Surfaces," *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Melbourne, Australia, vol. 26, pp. 826-831, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[59] Marc Platini, "*Machine Learning Applied to Failure Analysis and Prediction in HPC Systems*," Doctoral Thesis, Grenoble Alpes University, 2020. [Google Scholar] [Publisher Link]

[60] Tatiana Prosvirnova et al., "Strategies for Modelling Failure Propagation in Dynamic Systems with AltaRica," *International Symposium on Model-Based Safety and Assessment*, Munich, Germany, vol. 1, pp. 101-115, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[61] Germán González Rodríguez, Jose M. Gonzalez-Cava, and Juan Albino Méndez Pérez, "An Intelligent Decision Support System for Production Planning based on Machine Learning," *Journal of Intelligent Manufacturing*, vol. 31, no. 5, pp. 1257-1273, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[62] Sara Bunian, Meshari A. Al-Ebrahim, and Amro A. Nour, "Role and Applications of Artificial Intelligence and Machine Learning in Manufacturing Engineering: A Review," *Engineered Science*, vol. 29, pp. 1-14, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[63] Hugo Pelletier, Optimize Machine Availability with Machine Learning, 2025. [Online]. Available: https://www.niagara.tech/blog/ameliorer-la-disponibilite-machine-avec-le-machine-learning

[64] S. Joe Qin, "Survey on Data-Driven Industrial Process Monitoring and Diagnosis," *Annual Reviews in Control*, vol. 36, no. 2, pp. 220-234, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[65] Zuyu Yin, and Jian Hou, "Recent Advances on SVM based Fault Diagnosis and Process Monitoring in Complicated Industrial Processes," *Neurocomputing*, vol. 174, pp. 643-650, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[66] Steve J. Bickley, Alison Macintyre, and Benno Torgler, "Artificial Intelligence and Big Data in Sustainable Entrepreneurship," *Journal of Economic Surveys*, vol. 39, no. 1, pp. 103-145, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[67] Houda KHOMSI et al., "The Role of AI and Big Data in the Evolution of Econometric Methods: A Literature Review," *International Journal of Scientific Research (Revue-IRS)*, vol. 3, no. 2, pp. 1786-1801, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[68] Gareth James et al., *An Introduction to Statistical Learning*, 1st ed., Springer New York, NY, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[69] Chloé-Agathe Azencot, *Introduction au Machine Learning*, 3rd ed., Dunod, 2025. [Google Scholar] [Publisher Link]

[70] David W. Hosmer Jr., Stanley Lemeshow, and Rodney X. Sturdivant, *Applied Logistic Regression*, John Wiley & Sons, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[71] Shivani Jain, and Anju Saha, "Improving Performance with Hybrid Feature Selection and Ensemble Machine Learning Techniques for Code Smell Detection," *Science of Computer Programming*, vol. 212, pp. 1-34, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[72] Tlamelo Emmanuel, Dimane Mpoeleng, and Thabiso Maupong, "Power Plant Induced-Draft Fan Fault Prediction using Machine Learning Stacking Ensemble," *Journal of Engineering Research*, vol. 12, no. 2, pp. 82-90, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[73] Oded Maimon, and Lior Rokach, *Data Mining and Knowledge Discovery Handbook*, 2nd ed., Springer New York, NY, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[74] Lior Rokach, and Oded Maimon, *Data Mining with Decision Trees: Theroy and Applications: Theory and Applications*, Series in Machine Perception and Artificial Intelligence: Volume 81, World Scientific Publishing, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[75] Sanhita Mishra et al., "Tree Based Fault Classification in Underground Cable," *Procedia Computer Science*, vol. 218, pp. 524-531, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[76] Leo Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001. [CrossRef] [Google Scholar] [Publisher Link]

[77] Weiting Zhang, Dong Yang, and Hongchao Wang, "Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2213-2227, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[78] Achmad Widodo, and Bo-Suk Yang, "Support Vector Machine in Machine Condition Monitoring and Fault Diagnosis," *Mechanical Systems and Signal Processing*, vol. 21, no. 6, pp. 2560-2574, 2007. [CrossRef] [Google Scholar] [Publisher Link]