Order number:

Dissertation submitted to the

UNIVERSITY OF MOHAMED BOUDIAF – MSILA

FACULTY OF MATHMATICS AND COMPUTER SCIENCE
DEPARTEMENT OF COMPUTER SCIENCE

In partial fulfillment of the requirements for the degree of

Master in **Computer science**

By

Slimani Tahar

Kroubi Rafik

Title of the dissertation

Developing a Predictive Maintenance Tool for the Company ETUSA

Under the supervision of

Dr.Tahar Mehenni

Composition of the jury

Dr.Malika Boudia	University of Msila	President
Dr.Rached Yagoubi	University of Msila	Examiner

*June, **2023***

DEDICATIONS

“With gratitude and deep appreciation, we dedicate this dissertation to our beloved parents, family members, and dear friends. Whom have been the driving force behind our success with their belief in our abilities, even in the face of challenges.

We are profoundly grateful for their invaluable guidance and sacrifices, as we owe them everything. This achievement would not have been possible without their love and support. It is with heartfelt affection and profound gratitude that we dedicate this work to them.”

Kroubi Rafik, Slimani Tahar

ACKNOWLEDGMENTS

We would like to extend our sincere gratitude to our supervisor Dr.Tahar Mehenni, without whom we would've been lost, and whose support, inspiration, and guidance has lead us through our studies and project with ease. We thank him for his efforts, and the enormous help he provided along the way.

We would also like to extend our gratitude to the University of "Mohamed Boudiaf – M'sila" for providing us with the countless opportunities, necessary education, and support to build our future.

Lastly, we are grateful to the jury members, Dr.Malika Boudia and Dr.Rached Yagoubi, for taking their valuable time to evaluate our project, and for their feedback.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

In recent years, the transportation industry has witnessed a significant shift towards optimizing operational efficiency and reducing downtime through the implementation of advanced maintenance strategies. As a prominent player in the bus transportation sector, ETUSM (Enterprise de Transport Urbain et Suburbain de M'sila) has recognized the importance of proactive maintenance practices to enhance the reliability and availability of its fleet. The effective utilization of resources, such as buses, not only improves customer satisfaction but also ensures the seamless operation of public transportation services.

Traditionally, maintenance activities have been carried out based on fixed schedules or reactive approaches, often resulting in unforeseen breakdowns and costly repairs. However, with the advent of data-driven technologies and the rise of the Internet of Things (IoT), new opportunities have emerged for transforming maintenance practices. Predictive maintenance, in particular, has gained significant attention as a methodology that can revolutionize the way organizations manage their assets by leveraging data and advanced analytics.

This master's dissertation aims to develop a predictive maintenance tool for ETUSM, the leading bus company serving the urban and suburban areas of M'sila. The tool will be designed to harness the power of data generated by ETUSM's fleet, enabling the identification of potential failures and proactive maintenance actions. By shifting from reactive to predictive maintenance, ETUSM will be able to minimize unexpected downtime, optimize maintenance costs, and ultimately enhance the overall reliability and availability of its bus fleet.

CHAPTER 1:

MAINTENANCE

1. Introduction to Maintenance:

In any production environment that relies on machinery and equipment, maintenance is an essential component of ensuring efficient operations. Maintenance can be broadly defined as any activity or set of processes that aim to the continuous and efficient functionality of equipment, machinery, and other assets. This can encompass a wide range of tasks, from routine tests and measurements to replacements, adjustments, and repairs.

The ultimate goal of maintenance is to retain material in a serviceable condition or to restore it to serviceability, minimizing the risk of unexpected downtime and ensuring that machinery and equipment continue to perform optimally. To achieve this goal, maintenance tasks may be scheduled at regular intervals, conducted on an as-needed basis, or guided by predictive data analysis and other tools.

By prioritizing maintenance practices, businesses can reduce costs, minimize downtime, and optimize their production processes. Which can help to ensure long-term success and profitability in a competitive environment. **[1]**

2. Applications:

Maintenance plays a critical role in any industry that relies on machinery and equipment. Some of the key sectors that depend heavily on industrial maintenance include:

- Manufacturing industries
- Energy industries such as oil, gas, and mining
- Land transportation, including trains, buses, and trucks
- Metalworking and fabrication
- Air transportation, including commercial airlines and private aviation
- Offshore structures, such as oil rigs and wind turbines

- Woodworking and forestry
- Industrial plants and facilities

In all of these industries, industrial maintenance is a significant part of both capital expenditures and operational budgets. By investing in regular maintenance, businesses can improve the efficiency and productivity of their operations, reduce the risk of costly downtime and repairs, and ultimately, maximize their return on investment. **[2]**

3. Maintenance Strategies:

Maintenance is a broad term that encompasses a range of different activities and approaches. To ensure that machinery and equipment continue to operate smoothly and efficiently, it is essential to select the right type of maintenance for the specific needs of each situation. Understanding the different types of maintenance available can help individuals choose the one that is best suited to their needs.

There are several types of maintenance, including:

- **Preventive maintenance**, also referred to as **Routine Maintenance** or **Planned Maintenance**, which involve regular inspections and maintenance to prevent equipment failures before they occur
- **Corrective maintenance**, which involves repairing equipment only after it has already failed
- **Predictive maintenance**, which uses data analysis and monitoring to predict when maintenance will be required

By understanding the differences between these maintenance types, individuals can choose the one that best serves their needs, taking into account factors such as the type and age of the equipment, the nature of the work being done, and the available resources. Ultimately, selecting the right maintenance approach can help maximize the lifespan and performance of equipment, minimize downtime and repair costs, and improve overall productivity and efficiency. **[1]**

3.1. Routine Maintenance:

In addition, referred to as preventive maintenance, implemented on a fixed schedule, typically chosen to be performed on downtime between shifts or on the weekends to avoid affecting serviceability goals, it includes inspections, cleaning, replacing, and checking parts or units. **[3]**

3.1.1. Key Elements:

There are several key elements concerning Routine Maintenance that must be considered, such as scheduling regular maintenance, conducting inspections, and replacing or repairing any broken parts identified during the inspection. By following this process, businessmen and companies can ensure the continuous functionality of their machinery, reduce the risk of unexpected downtime, and save money in the long run.

3.2. Planned Maintenance:

Planned Maintenance is similar to routine maintenance, but with less frequency of occurrence, more time-consumption, more expense, and usually requires the precision and services of a specialist, it is very thorough and could be a complete check-up of each essential part or component of a working unit. **[3]**

3.2.1. Example:

For an air conditioning unit, routine maintenance could include minor procedures such as taking out and cleaning the filters of an air conditioning unit, while planned maintenance would be hiring an HVAC (Heating, Ventilation, and Air Conditioning) specialist to check refrigerant levels, possible leaks, and measure airflow through the evaporator coil. **[3]**

3.3. Corrective Maintenance:

Corrective Maintenance can be a planned or unplanned process, it occurs after a routine inspection uncovers a potential issue or after the complete breakdown or malfunction of equipment, it is usually expensive, but it depends on the number of broken or worn-out parts that need to be repaired or replaced. **[4]**

3.3.1. Planned Corrective Maintenance:

Occurs in two scenarios:

- When an asset is allowed to run until it fails or breaks down, it is strictly applied to non-critical units that are easily and cheaply repaired or replaced.
- When a problem is found before, it causes equipment failure.

- **Example:**

Imagine an asset has several fans. The asset can still operate properly if one breaks and there are many extra fans in your storeroom, which means repairs are quick and inexpensive. Because of this, you decide to let the fans run until one of them fails and then replace it at that point. This is an example of run-to-fail corrective maintenance.

Let us say you perform a preventive maintenance inspection on a conveyor system every two weeks. During one of these inspections, you find that some bearings have been damaged, so you replace them. This is an example of preventive corrective maintenance. **[4]**

3.3.2. Unplanned Corrective Maintenance

Considered in two situations:

- When a breakdown occurs before a scheduled maintenance.
- When an asset shows signs of failure or reaches failure unexpectedly.

- **Example:**

Pretend your facility has a compressor. You plan for it to be inspected and repaired after every 100 hours of use in order to keep it functioning properly. However, the asset breaks after only 75 hours of operation and you have to perform an emergency repair.

Your facility can’t afford for one of its forklifts to break down, but there’s no preventive maintenance done on the vehicles. When it breaks and a technician scrambles to get it working again. **[4]**

3.4. Predictive Maintenance:

Depending on real-time monitoring of asset condition and performance, it is a technique that implements data analysis to detect patterns and anomalies in equipment and assets that may show signs of potential issues, so they can be fixed before they result in a failure or malfunction. **[5]**

3.4.1. Key Elements:

To implement predictive maintenance techniques, several key elements must be present, including data analysis, specialized software, artificial intelligence, Internet of Things (IoT) devices, and integrated systems. It allows businesses to collect and analyse data from their equipment and assets, identify patterns, and use predictive models to predict potential failures and schedule maintenance beforehand. **[5]**

3.4.2. Objectives:

The main objectives of predictive maintenance are to predict asset failure before it occurs, optimize maintenance frequency, and prevent unplanned corrective maintenance, reducing operational costs and improving efficiency. **[5]**

3.4.3. Benefits:

Predictive maintenance is a vital strategy with numerous benefits, including a significant reduction or near elimination of unscheduled equipment downtime due to failure, increased labour utilization, increased production capacity, reduced maintenance costs, and increased equipment lifespan. **[5]**

4. Importance of Maintenance:

Maintenance provides many benefits that can contribute to the success of any business, company, or industry. These benefits include reducing expenses by preventing breakdowns that can lead to an increase in labour costs per unit during machinery repairs, ensuring the

maintainability of assets and equipment, assuring quality of service, therefore it plays a pivotal role in the longevity of success.

5. Conclusion:

Maintenance represents a significant fraction of the total operating costs in many industry sectors, as it showed improvements and effectiveness in the overall performance of businesses, many industries incorporate at least one of the Maintenance Strategies mentioned above, with priorities on which strategy to use in which situation and the value of each strategy is significant as they help set a straight path to success with no cuts on serviceability, no shortage in productivity and no reduced functionality.

Maintenance strategies offer a huge advantage in an extremely competitive field, which is business; therefore, it plays one of the most important roles for success. **[6]**

CHAPTER 2:

PREDICTION & MACHINE LEARNING

1. Prediction:

1.1. Introduction to Prediction:

Prediction is the process of using available information or data to estimate or forecast future events or outcomes. It involves identifying patterns and trends in past or present data and using this information to make informed predictions about what might happen in the future. Accurate predictions are essential in many areas of research, including economics, social sciences, medicine, engineering, and technology. Predictive models are often used to make decisions, allocate resources, and mitigate risks.

There are different types of prediction models, including statistical models, machine learning models, and simulation models. These models require data collection, cleaning, selection, training, evaluation, and deployment. The accuracy and effectiveness of a predictive model depends on the quality of the data, the complexity of the model, and its suitability for the specific problem or application. Overall, prediction helps individuals and organizations make informed choices and take proactive measures to address potential risks and opportunities**. [7]**

1.2. Importance of prediction in various fields:

Predictive analytics plays a crucial role in resolving intricate business challenges and discovering new opportunities. There are several benefits to using this approach, including fraud detection, marketing campaign optimization, risk reduction, and operational improvements. To prevent fraudulent activity, various analysis techniques are used to enhance pattern recognition, identify illegal behaviour, and stop frequent fraud occurrences. Predictive models help organizations attract, retain, and grow their most important customers by analysing customer purchase behaviour and promoting cross-selling opportunities. Credit scores and insurance claims are used to minimize risks through predictive analytics. Additionally, predictive analytics streamlines operations and helps companies make better decisions by managing resources and

forecasting inventories to increase operational efficiency. Hotels use predictive analytics to optimize space occupancy and increase revenue by forecasting the number of visitors, while airlines use it to confirm a range of ticket prices. **[12]**

1.3. Applications of Predictive Analytics:

Predictive analytics has a wide range of applications across different industries, including fraud detection, marketing campaigns optimization, risk minimization, and improving business operations. It involves analysing data to identify patterns and trends, and using this information to make predictions about future outcomes.

Here are a few fields on which Predictive analytics has helped:

- **Marketing**: it can help businesses identify ideal consumers and plan how to reach them
- **Retail**: used it to manage inventory and offer customized promotions to customers
- **Manufacturing**: it can optimize the supply chain and identify potential equipment failures
- **Healthcare:** it can aid in specialized treatment and identify patients at risk of developing chronic diseases
- **Finance:** it helps prevent fraud, assess credit risk, and retain valuable customers. **[12]**

1.4. Predictive Analytics Process:

The predictive analytics process generally consists of seven steps, which are:

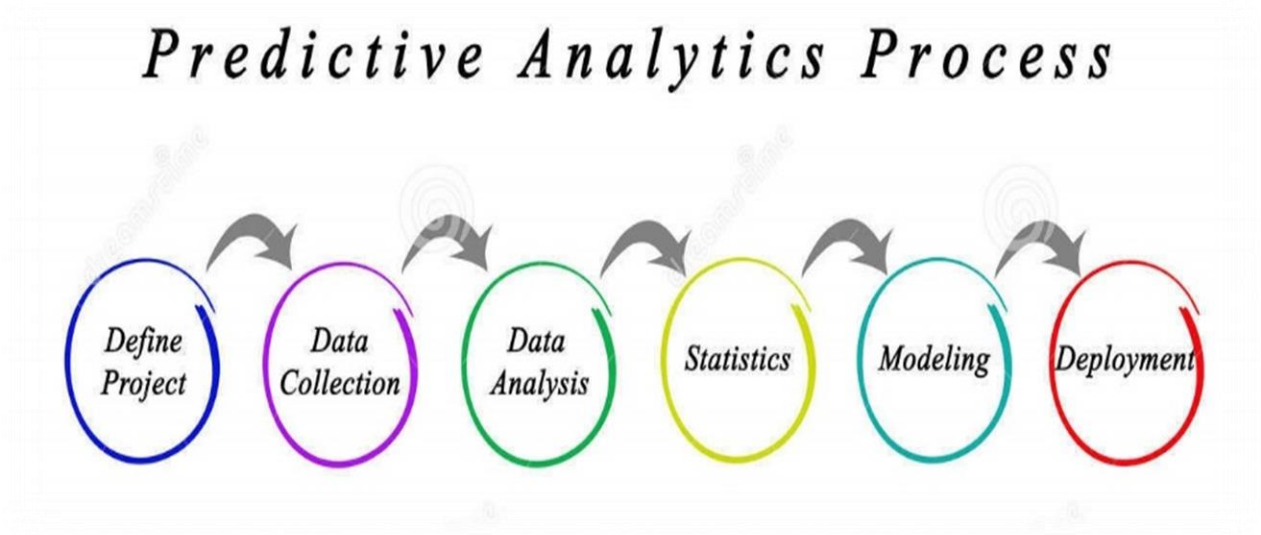

Figure 2.1 – Predictive Analytics Process [12]

Define Project: Establish project goals, scope, deliverables, business objectives, and identify data sets to be used.

Data Collection: Collect data from multiple sources to gain a comprehensive view of customer interactions.

Data Analysis: Analyse and model data to discover useful insights and draw conclusions.

Statistics: Use statistical analysis to validate assumptions and test hypotheses using standard models.

Modelling: Develop accurate predictive models for future outcomes and evaluate multiple models for the best solution.

Deployment: Deploy predictive models into everyday decision-making processes to automate decisions and obtain results and reports.

Model Monitoring: Manage and monitor models to ensure they are performing as expected and providing desired results. **[12]**

1.5. Prediction Issues:

The major issue is preparing the data for Classification and Prediction. Preparing the data involves the following activities:

1.5.1. Data Cleaning:

Data cleaning involves removing the noise and treatment of missing values. The noise is removed by applying smoothing techniques, and the problem of missing values is solved by replacing a missing value with the most commonly occurring value for that attribute. **[8]**

1.5.2. Relevance Analysis:

The database may also have irrelevant attributes. Correlation analysis is used to know whether any two given attributes are related. **[8]**

1.5.3. Data Transformation and reduction:

The data can be transformed by any of the following methods.

- **Normalization**: The data is transformed using normalization. Normalization involves scaling all values for a given attribute to make them fall within a small specified range. Normalization is used when the neural networks or the methods involving measurements are used in the learning step.
- **Generalization:** The data can also be transformed by generalizing it to the higher concept. For this purpose, we can use the concept hierarchies. **[8]**

2. Machine Learning:

2.1. Introduction to Machine Learning:

Machine learning is a subfield of artificial intelligence that enables machines to learn and improve over time without explicit programming. This is accomplished by feeding the machine labelled training data, allowing it to automatically identify patterns and make predictions.

Unlike traditional programming, machine learning is an automated process that enables machines to solve problems with minimal human input. While artificial intelligence encompasses machines making decisions and learning like humans, machine learning is a subset of AI that focuses on autonomous learning from data.

By leveraging machine learning, businesses can automate tasks, analyse large datasets, and improve decision-making accuracy. Ultimately, machine learning is a powerful tool that can save time and money while driving innovation and improving outcomes across a wide range of applications. **[10]**

2.2. Types of Machine Learning:

To understand how machine-learning works, you'll need to explore different machine learning methods and algorithms, which are basically sets of rules that machines use to make decisions. Below, you will find the five most common and most used types of machine learning: **[4]**

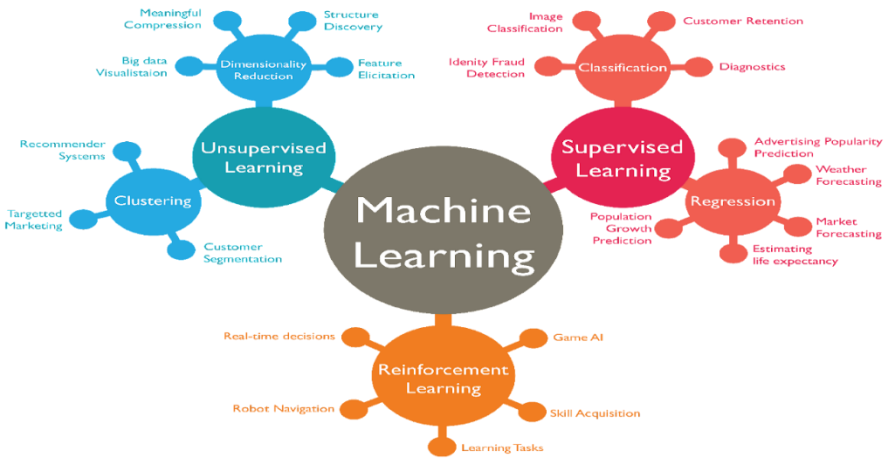

Figure 2.2 – Types of Machine Learning

2.2.1. Supervised Learning:

Supervised learning is the most commonly used machine learning approach where algorithms and models use labeled training data to make predictions. Training samples include an input and a desired output, which are analyzed by the algorithm to make educated guesses when labeling unseen data. This approach is called "supervised" because the models require manually labeled data to learn from. Labeled data helps the machine to recognize patterns and connections in similar data categories, images or words.

To illustrate, in order to train a machine learning algorithm to automatically detect spam, a data set of emails that are either spam or not spam would be used.

Supervised learning is typically classified into two types of tasks: **Classification** and **Regression**. Classification algorithms, such as Support Vector Machines (SVM) and Naive Bayes, are used when the output value is a category with a finite number of options, while regression models are used when the expected result is a continuous number. For instance, regression can be used to predict the probability of an event happening or the value of a particular property. **[10]**

2.2.2. Unsupervised Learning:

Unsupervised learning algorithms explore and unveil patterns in data that lacks labels. Since the models receive input data without desired outputs, they must infer conclusions based on circumstantial evidence, without any guidance or training. These models are not trained with the "right answer," so they must find patterns independently.

Clustering is one of the most common types of unsupervised learning that groups similar data, mostly used for exploratory analysis. This approach can help detect hidden patterns or trends.

For instance, an e-commerce company's marketing team could use clustering to improve customer segmentation by grouping customers with similar spending behaviors. Based on this segmentation, marketers can develop tailored strategies, such as offering discounts or promotions to high-spending customers with low-income to reward their loyalty and improve retention. **[10]**

2.2.3. Semi-Supervised Learning:

"In semi-supervised learning, the classification is done on data with predefined classes as well as on data with non-predefined classes, this is a mixed type of learning. The objective of combining both types of learning is to significantly improve the quality of classification and learning in general." **[11]**

2.2.4. Reinforcement Learning:

"Reinforcement learning consists of learning from successive experiences in contact with the environment in order to find the best solution. It is therefore learning guided by the reaction of the environment and generally involves three main tasks:

- ✓ Observation of the effects of the agent's actions on the environment.
- ✓ Deduction of the quality of these actions based on observations (Positive/Negative)
- ✓ Improving future actions." **[11]**

2.2.5. Deep Learning:

Deep learning models, used by tech giants such as Google, Microsoft, and Amazon, are advanced machine learning algorithms based on Artificial Neural Networks (ANN) that emulate the human brain. With multiple layers of interconnected neurons, these models can process various forms of data and learn progressively through input. Deep learning is ideal for complex problems and large data sets, such as image recognition, speech recognition, and NLP, but requires extensive training data. **[10]**

3. Types of Prediction Problems in Machine Learning:

3.1. Python and Sci-Kit Learn:

3.1.1. Python:

Python is a popular high-level programming language known for its simplicity, readability, and versatility. It has a clean syntax, supports multiple programming styles, and has a large community contributing to its libraries and frameworks. Python is cross-platform and has an extensive standard library, making it suitable for various tasks. **[30]**

3.1.2. Sci-Kit Learn:

Scikit-learn is a popular Python library for machine learning. It provides a range of tools and algorithms for tasks like data analysis, pre-processing, model selection, and evaluation. With a user-friendly interface, scikit-learn is accessible to both beginners and experienced practitioners. It integrates well with other scientific computing libraries and offers a comprehensive collection of algorithms and utilities for various machine learning problems.

Scikit-learn's strengths lie in its simplicity and extensive functionality. It supports both supervised and unsupervised learning techniques, along with feature extraction, selection, and preprocessing capabilities. The library focuses on good software engineering practices and provides detailed documentation and examples for easy understanding and application.

Overall, scikit-learn is a valuable tool for machine learning tasks in Python. Its user-friendly interface, comprehensive functionality, and seamless integration with other libraries make it a popular choice for both beginners and experienced practitioners in the field of data analysis and machine learning. **[31]**

3.2. Regression:

Regression is a type of supervised machine learning algorithm used to predict a continuous numerical value. In the given example, the aim is to predict the stock value for tomorrow based on its past performance. This is done by analysing the relationship between the independent variables (past stock values) and the dependent variable and creating a mathematical model that can predict the value of the dependent variable for new data. **[9]**

3.2.1. Linear Regression:

Linear regression is a statistical technique utilized to estimate the association between two variables. This method supposes a linear connection between the independent and dependent variables, and endeavors to locate the most fitting line that characterizes this relationship. The line is established by decreasing the sum of squared deviations between the projected and actual values. Linear regression is extensively employed in a range of domains, such as finance, economics, and social sciences, for examining and projecting trends in data. It can also be extended to multiple linear regression, which involves multiple independent variables, and logistic regression, that is utilized for binary classification issues. **[15]**

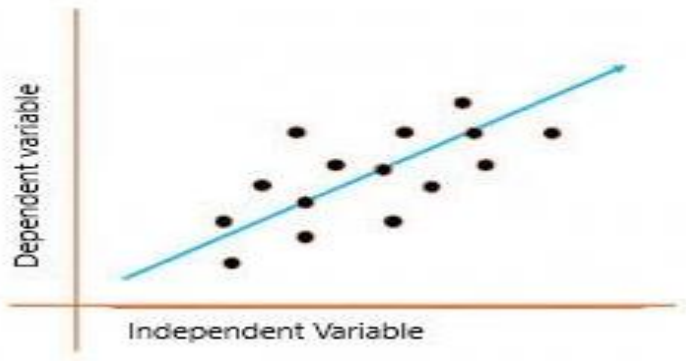

Figure 2.3 – Linear Regression Hyperbole [15]

The above graph depicts the linear relationship between the output variable (y) and the predictor variable (X). The best fit straight line is shown in blue, which is plotted based on the given data points to fit the points as best as possible.

To calculate the best-fit line, linear regression uses the traditional slope-intercept form as follows:

$$Yi = \beta 0 + \beta 1 Xi$$

Where Yi is the dependent variable, β0 is the constant/intercept, β1 is the slope/intercept, and Xi is the independent variable.

This algorithm explains the linear relationship between the dependent output variable y and the independent predictor variable X using a straight line Y = B0 + B1 X. **[15]**

- **Random Error(Residuals):**

In regression, the difference between the observed value of the dependent variable (yi) and the predicted value (predicted) is called the residuals.

$$\varepsilon_i = y_{predicted} - y_i$$

Where $y_{predicted} = B_0 + B_1 X_i$

The linear regression algorithm aims to obtain the optimal values for B0 and B1 to find the best-fit line. The best-fit line is the one with the least error, which means that the error between predicted and actual values should be minimized**. [15]**

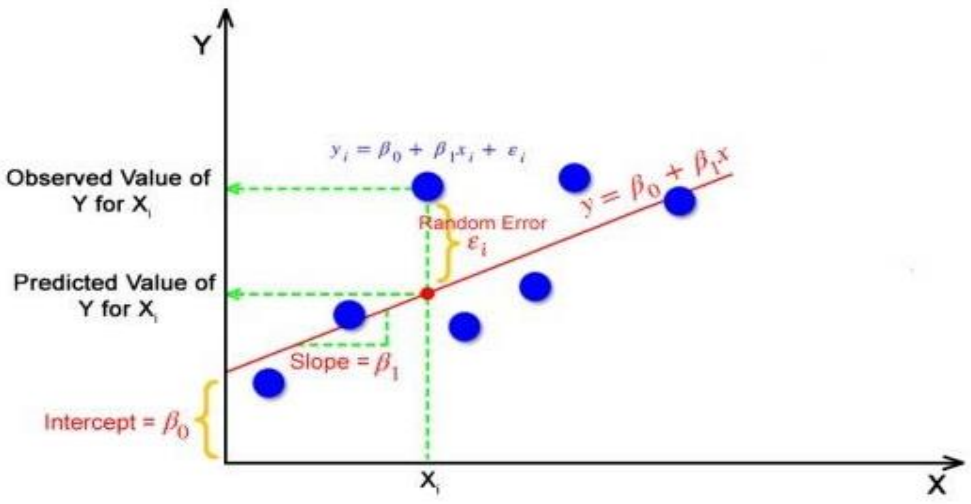

Figure 2.4 – Linear Regression Random Error [15]

- **Cost Function for Linear Regression:**

The cost function is essential for determining the optimal values of B0 and B1 that result in the best-fit line for the data points.

In Linear Regression, the Mean Squared Error (MSE) cost function is commonly used, which represents the average of the squared errors that occurred between the predicted output ($y_{predicted}$) and the actual output (y_i).

MSE is calculated using the simple linear equation y = mx + b:

$$MSE = \frac{1}{N}\sum_{i=1}^{n}\left(y_i - (B1x_i + B0)\right)^2$$

By utilizing the MSE function, we can adjust the values of B0 and B1 such that the MSE value converges to the minimum. These parameters can be computed using the gradient descent method, which minimizes the value of the cost function. **[15]**

3.2.2. Polynomial Linear Regression:

Polynomial regression is a form of linear regression that models the relationship between the independent variable x and dependent variable y using a polynomial function of degree n. The equation is denoted as E (y |x). The objective is to fit a nonlinear relationship between x and the conditional mean of y. The least-squares method is commonly used, which corresponds to minimizing the variance of the coefficients as per the Gauss Markov Theorem. In Polynomial Regression, a curvilinear relationship between the dependent and independent variables is observed and a polynomial equation is fitted to the data. Machine learning is a subset of Multiple Linear Regression, where we convert the equation into a Polynomial Regression by adding more polynomial terms. **[10]**

- **Types of Polynomial Regression:**

  - ✓ "Linear: If the degree is 1
  - ✓ Quadratic: if the degree is 2
  - ✓ Cubic: if the degree is 3 and goes on, on the basis of the degree." **[16]**

Polynomials	Form	Degree	Examples
Linear Polynomial	$p(x): ax+b, a \neq 0$	Polynomial with Degree 1	$x + 8$
Quadratic Polynomial	$p(x): ax^2+b+c, a \neq 0$	Polynomial with Degree 2	$3x^2-4x+7$
Cubic Polynomial	$p(x): ax^3+bx^2+cx, a \neq 0$	Polynomial with Degree 3	$2x^3+3x^2+4x+6$

Figure 2.5 – Types of Polynomial Regression [16]

- **Example:**

Here is an example about the difference between applying Linear Regression and Polynomial Regression applied on the same data set: **[17]**

Linear Regression Applied:

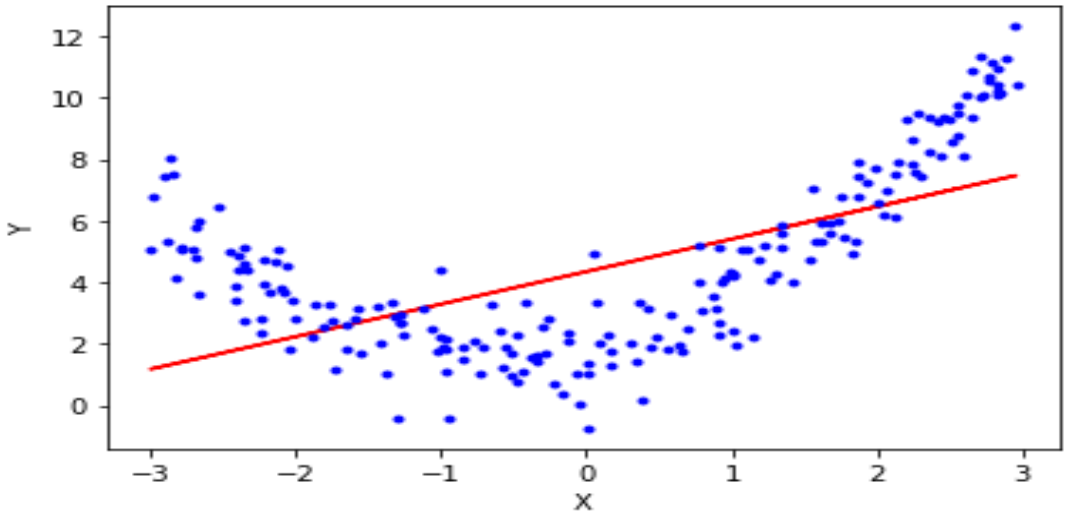

Figure 2.6 – Linear Regression Applied [17]

Polynomial Regression Applied:

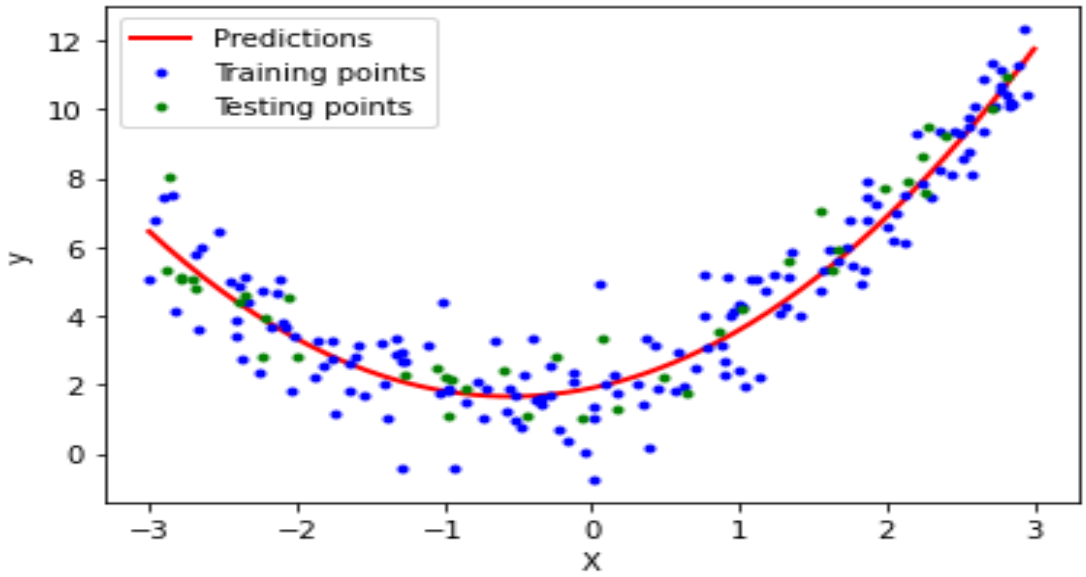

Figure 2.7 – Polynomial Regression Applied [17]

3.2.3. Logistic Regression:

Logistic regression is a data analysis technique that explores the relationships between two data factors to predict the value of one based on the other, typically with binary outcomes. For instance, it can be utilized to forecast whether a website visitor will click the checkout button in their shopping cart or not, based on prior visitor behaviour data such as browsing duration and the number of items in the cart. **[12]**

- **Logistic Regression Function:**

Logistic regression is a statistical model that uses the logistic function, or logit function, in mathematics as the equation between *x* and *y*. The logit function maps *y* as a sigmoid function of *x*.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Plotting of the equation will result in an S-curve, as shown in this figure: **[18]**

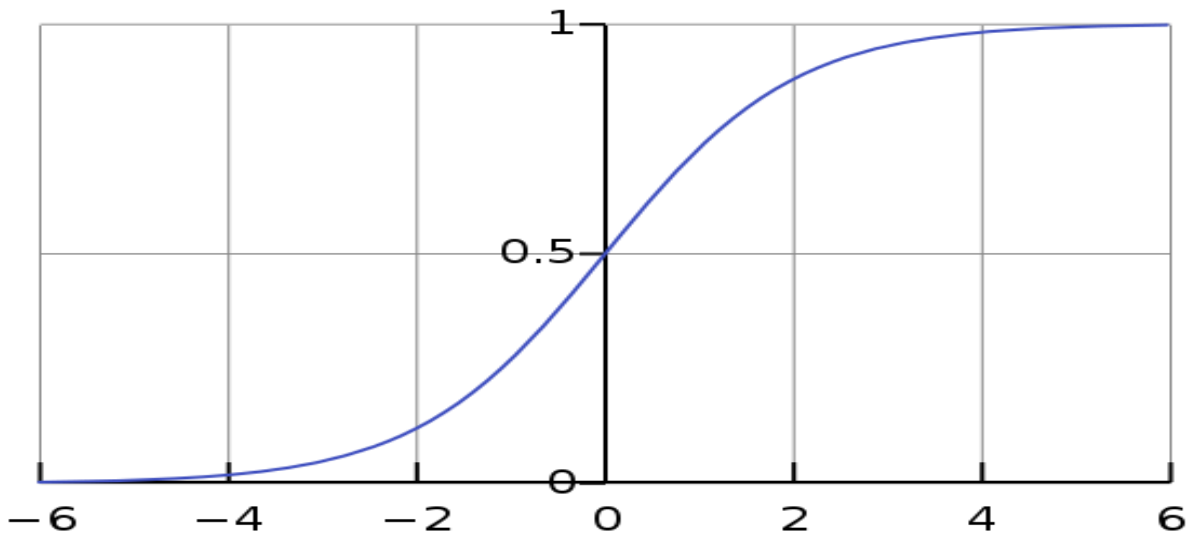

Figure 2.8 – Logistic Regression S-Curve [18]

3.3. Classification:

Classification is a type of predictive modelling task in machine learning where the goal is to predict a class label for a given input data. For instance, in tasks such as recognizing handwriting characters or identifying spam emails, classification models need to be trained on a large dataset, with known labels, and then validated using a new dataset to see how accurate the model can be in classifying new unlabelled data. **[9]**

3.3.1. Types of Classification:

There are four (4) different types of classification, which are:

- ✓ Binary Classification
- ✓ Multi-Class Classification
- ✓ Multi-Label Classification
- ✓ Imbalanced Classification

And each of them has its own use case, purpose, characteristics and results.

3.3.2. Binary Classification:

Binary classification is a type of problem where the aim is to categorize input data into two distinct categories. The training data used for this task is labelled as true and false, positive and negative, or 1 and 0, depending on the problem at hand. For example, we may want to determine

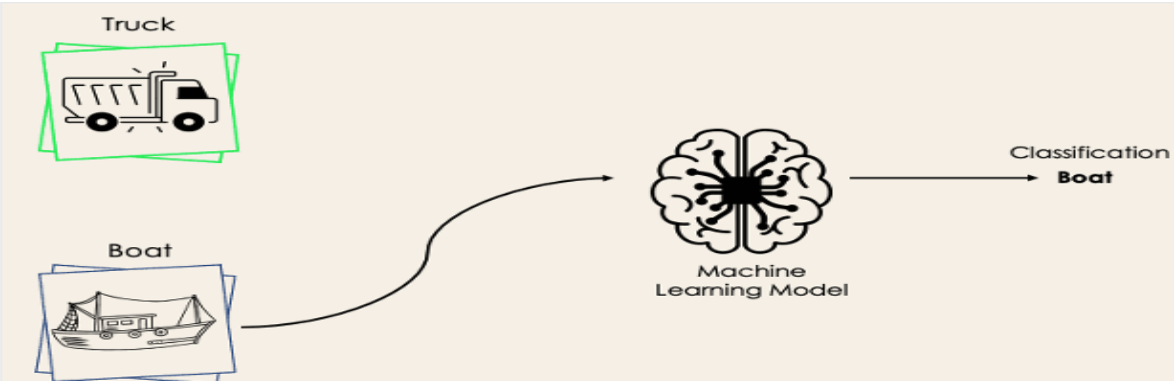

Figure 2.9 – Binary Classification [19]

whether an image represents a truck or a boat. While algorithms like Logistic Regression and Support Vector Machines are specifically designed for binary classification, other methods like Decision Trees and K-Nearest Neighbours can also be used for this task. **[19]**

3.3.3. Multi-Class Classification:

Multi-class classification involves predicting the class label of an input example, which can have at least two exclusive class labels. For example, we may want to identify whether a given image is a plane, car, or boat. In such cases, the model's goal is to predict the correct class to which the input example belongs. **[19]**

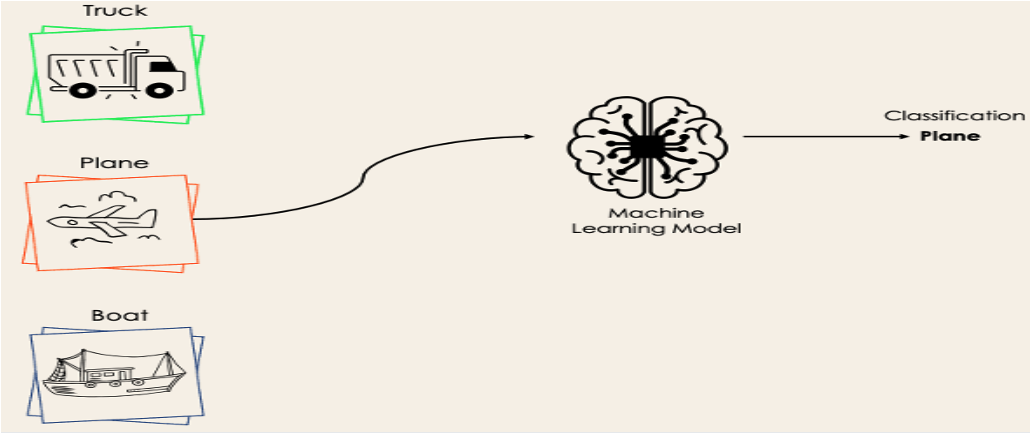

Figure 2.10 – Multi-Class Classification [19]

Many binary classification algorithms, including Random Forest, Naive Bayes, K-Nearest Neighbors, Gradient Boosting, SVM, and Logistic Regression, can also be used for multi-class classification. However, SVM and Logistic Regression do not natively support multi-class classification. To adapt these algorithms, binary transformation approaches such as one-versus-one and one-versus-all can be used. The one-versus-one strategy trains a classifier for each pair of labels, while the one-versus-all strategy trains a classifier for each label against all other labels. **[19]**

- **One versus one approach:**

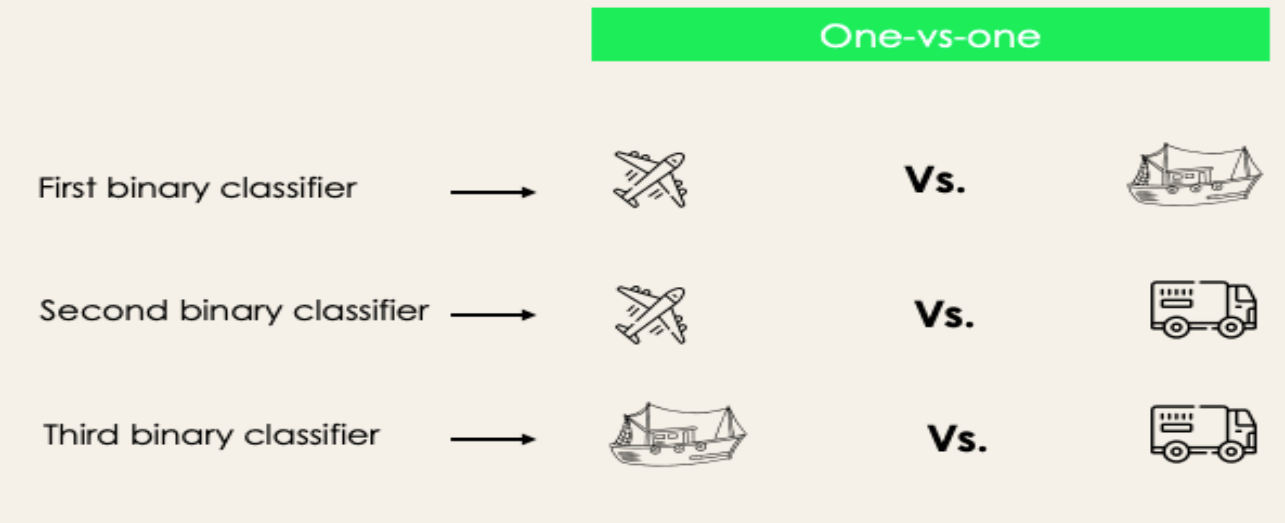

Figure 2.11 – One versus One Classification Approach [19]

- **One versus All approach:**

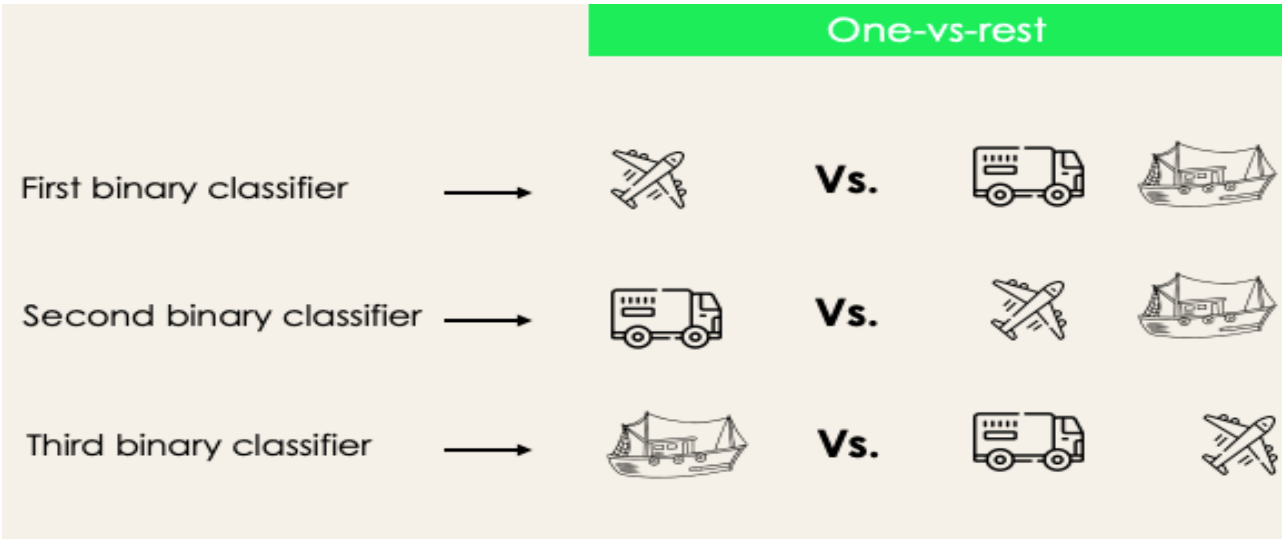

Figure 2.12 – One versus All Classification Approach [19]

3.3.4. Multi-Label Classification:

In multi-label classification, the objective is to predict zero or more classes for each input example, without mutual exclusion, since a single example can have more than one label. This type of classification is common in various domains such as auto-tagging in NLP or object recognition in computer vision, where input data can have multiple attributes or objects. For example, an image can have a plane, boat, truck, and dog all in one. **[19]**

Multi-label classification requires specialized algorithms as traditional multi-class or binary classification models cannot be used. However, specific algorithms have been developed for this task, including Multi-Label Decision Trees, Multi-label Gradient Boosting, and Multi-Label Random Forests. **[19]**

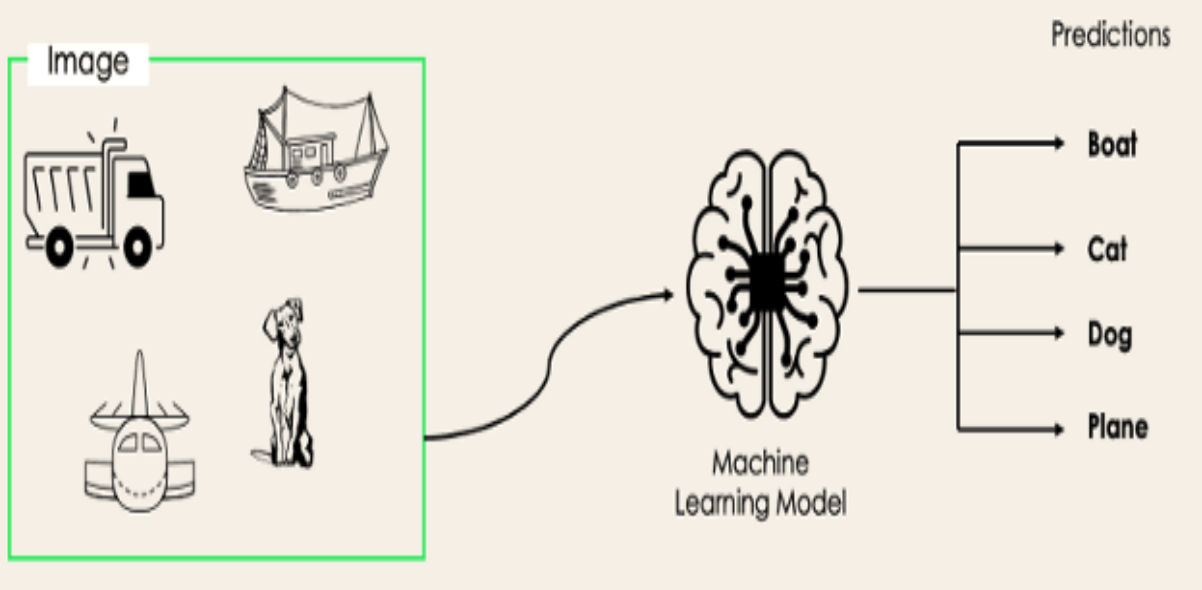

Figure 2.13 – Multi-Label Classification [19]

3.3.5. Imbalanced Classification:

In imbalanced classification, the distribution of examples across classes is uneven, with some classes having more examples than others in the training data. For instance, in a 3-class classification scenario, the training data may have 60% trucks, 25% planes, and 15% boats.

Imbalanced classification can arise in several real-world applications, such as detecting fraudulent transactions in financial industries, diagnosing rare diseases, or analyzing customer churn. Conventional predictive models such as Decision Trees, Logistic Regression, and others may not work well on imbalanced datasets since they may be biased towards predicting the majority class while treating the minority class as noise. **[19]**

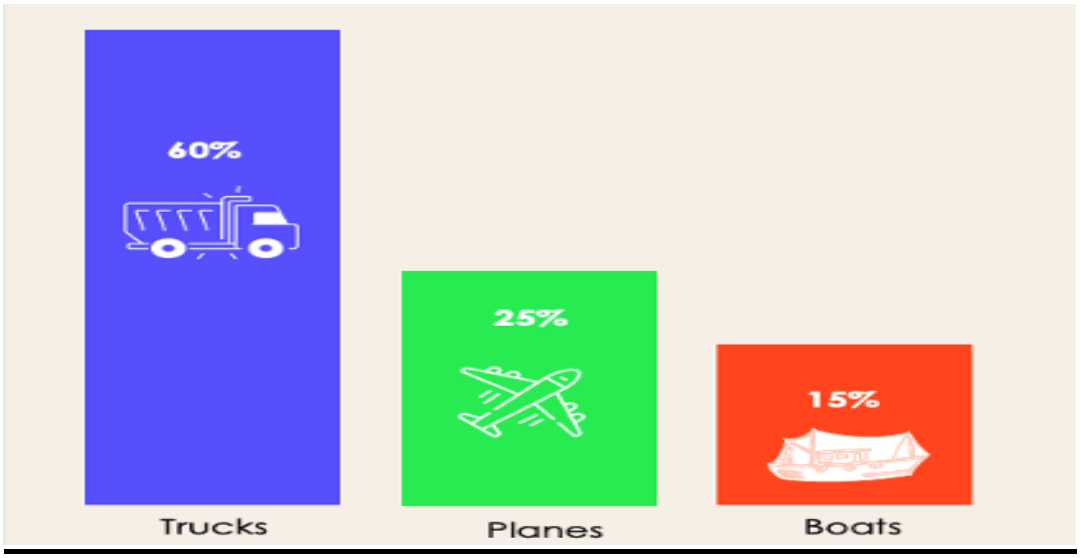

Figure 2.14 – Imbalanced Classification [19]

3.4. Machine Learning techniques:

3.4.1. Decision Trees:

A decision tree is a model that can be used for both **classification** and **regression**. It's a tree-like structure that represents decisions and their possible consequences. Each branch of the tree represents a choice between a number of alternatives and its leaves represent a decision. Decision trees partition data into subsets based on input variables and are popular due to their ease of understanding and interpretation. They are useful in decision analysis and can handle missing data and select preliminary variables. However, they have limitations in adapting to changes and lag behind in prediction accuracy compared to other models. The calculation can be complex, especially when dealing with uncertain data. **[13]**

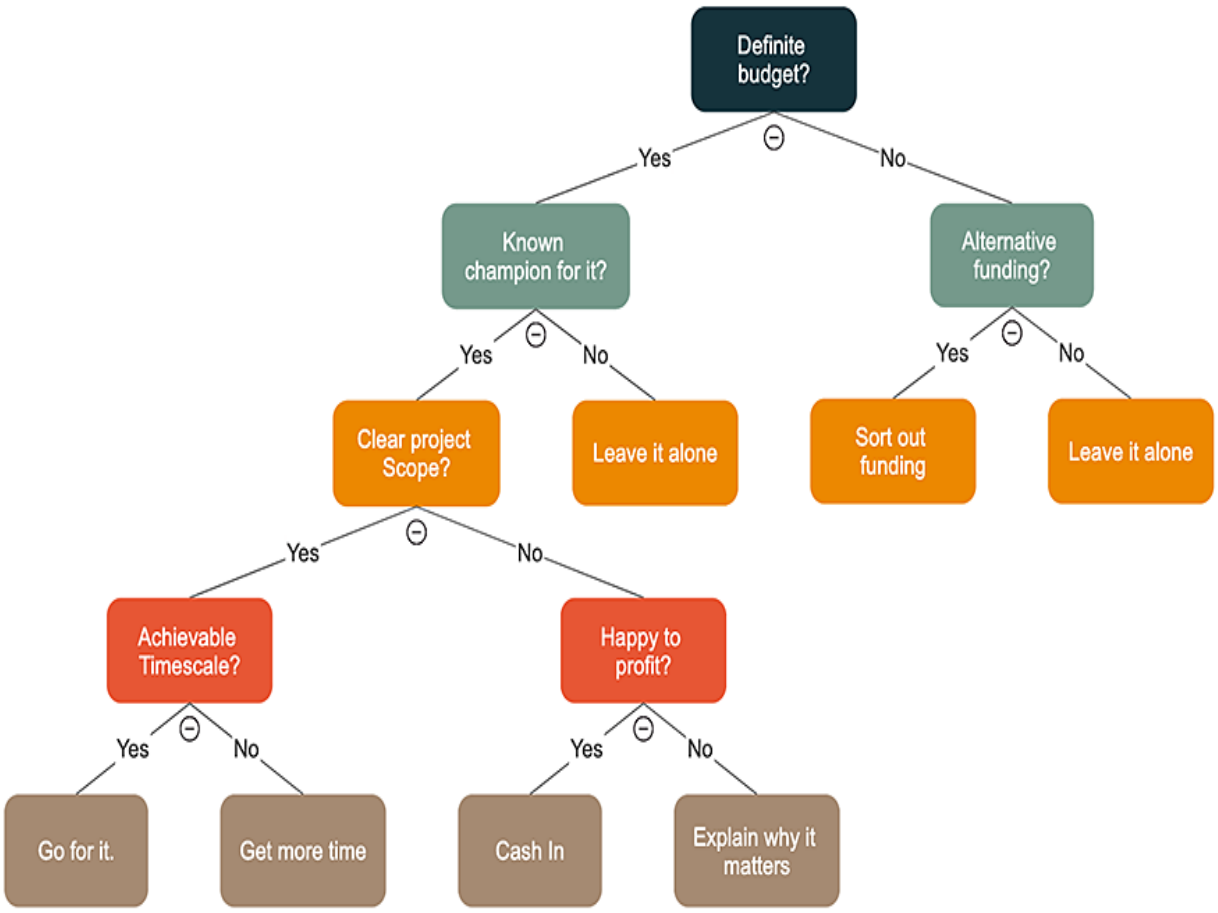

Figure 2.15 – Decision Trees Model [13]

3.4.2. Artificial Neural Networks:

A subset of machine learning models called neural networks, sometimes referred to as artificial neural networks (ANNs) or simulated neural networks (SNNs), created to imitate the structure and operation of the human brain. They are made up of a network of artificial neurons, or interconnected nodes, that analyse and send data using a set of weights and thresholds. Deep learning algorithms frequently employ neural networks to carry out operations like pattern identification, picture and speech recognition, natural language processing, and more. To learn and gradually increase their accuracy, they rely on training data. **[20]**

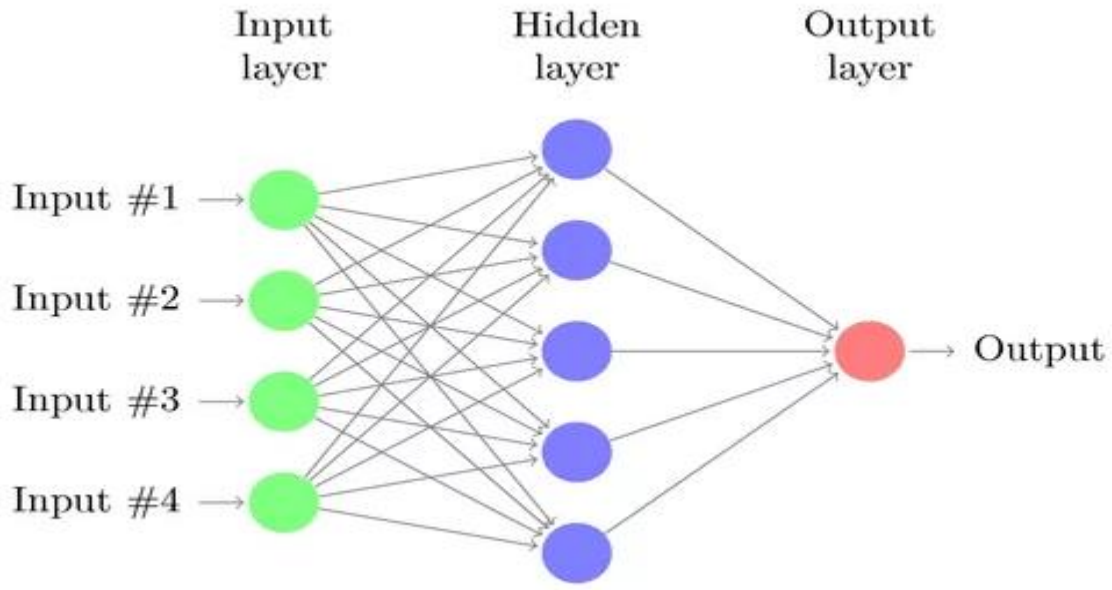

Figure 2.16 – Artificial Neural Networks Model [20]

Input data is sent to the neural network's input layer to begin the process. Each hidden node is connected to one or more output nodes in the output layer, and each input node in the input layer is connected to one or more hidden nodes in the hidden layers. The strength and relevance of the information travelling via the connections between nodes are determined by the weights and thresholds that are applied to such connections.

The weighted total of the inputs is calculated by the neural network upon receiving input data, and this result is then passed. **[20]**

3.4.3. K-Nearest Neighbour (KNN):

The k-nearest neighbours (KNN) algorithm is a supervised learning classifier that makes predictions or classifications based on the closeness of data points. Because it is non-parametric, no presumptions are made about how the data are distributed in general. Although KNN can be applied to classification and regression problems, classification challenges are where it is most frequently used. The algorithm decides which class to put a particular data point in based on a majority vote of its closes neighbours. The majority vote doesn't always require more than 50% of the vote to assign a class label in multi-class issues. **[21]**

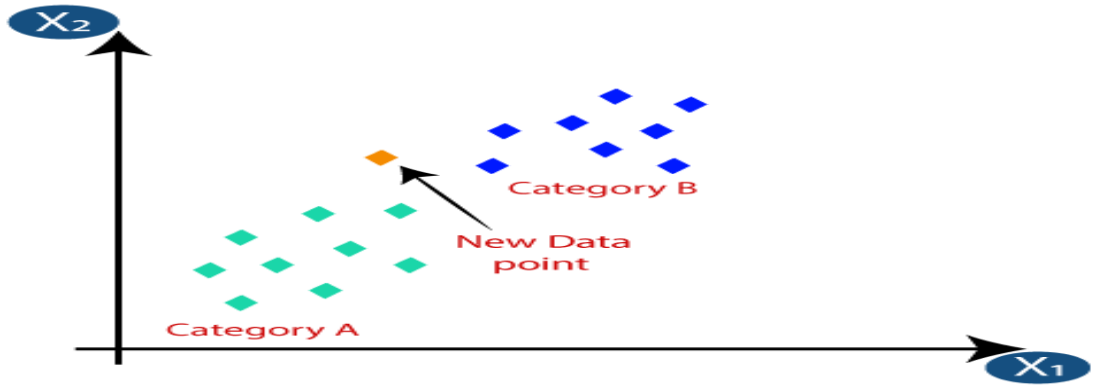

Figure 2.17 – K-Nearest Neighbor Model [21]

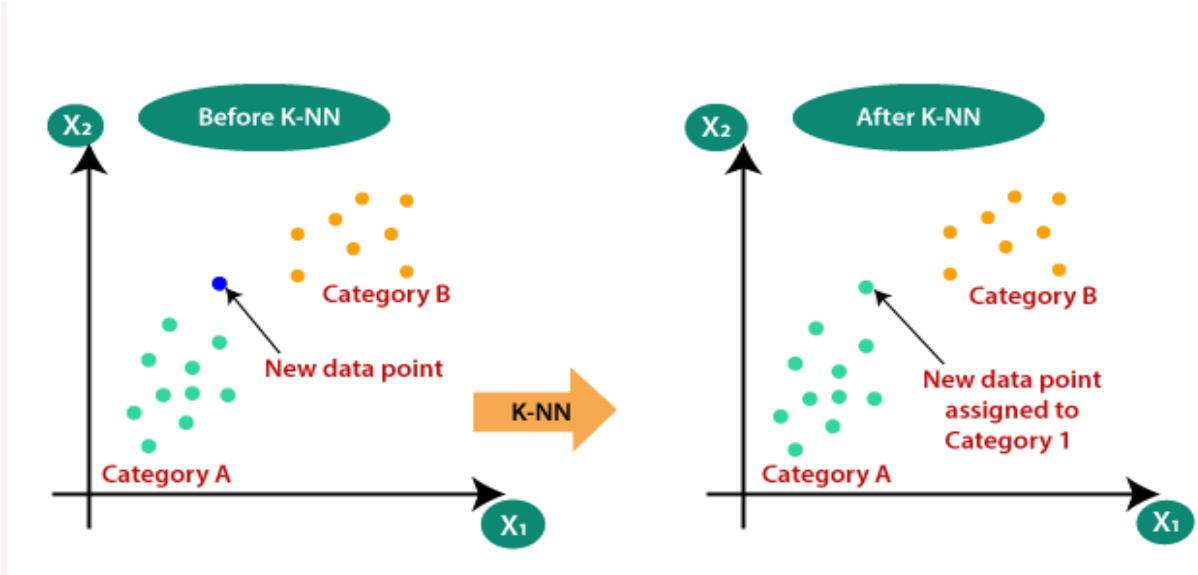

Figure 2.18 – K-Nearest Neighbor Applied [21]

3.4.4. Support Vector Machine (SVM):

A supervised machine learning technique that can be used for classification or regression problems. It works by determining the ideal hyperplane to use for dividing different data classes in a high-dimensional space. The model performs better in terms of generalization since the hyperplane is positioned to maximize the margin between the two classes. SVM is particularly helpful when there are many features and a limited amount of data points. The technique is frequently used in a variety of applications, including bioinformatics and the classification of text and images. **[22]**

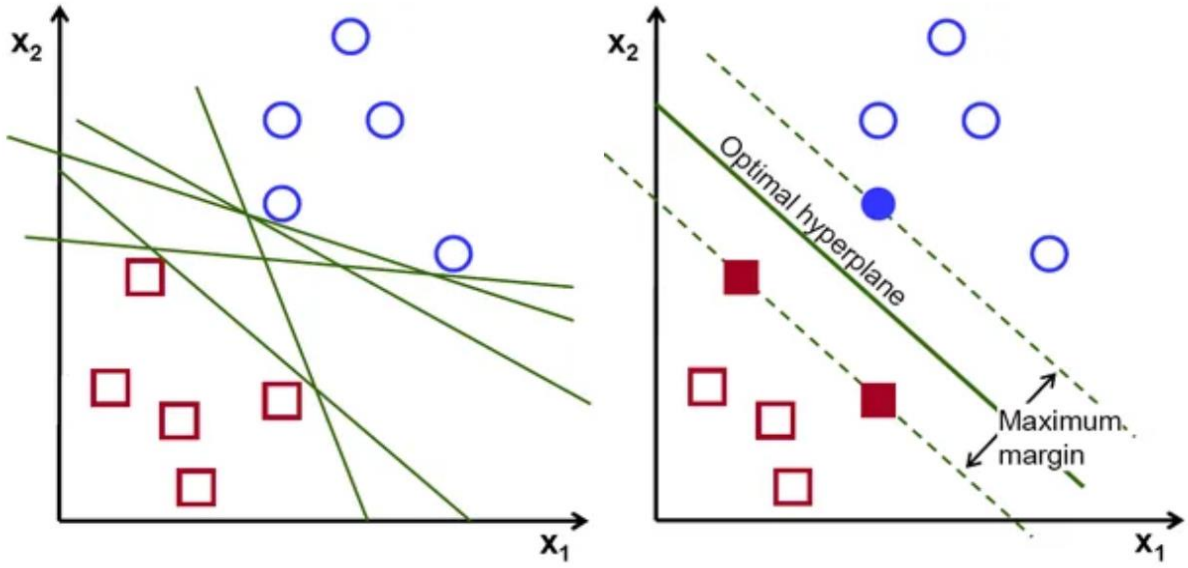

Figure 2.19 – Support Vector Machine (SVM) [22]

To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence. **[22]**

3.4.5. Naïve Bayes:

Naïve Bayes is a supervised learning algorithm designed for classification problems, especially in the realm of text classification with extensive datasets. A powerful and uncomplicated algorithm that facilitates the development of fast machine learning models capable of swift predictions. By functioning as a probabilistic classifier, Naïve Bayes predicts the probability of an object belonging to a particular class. Common applications of Naïve Bayes include spam filtering, sentiment analysis, and article classification. **[23]**

- **Bayes Theorem:**

Naïve Bayes algorithm uses prior knowledge and conditional probability to estimate the likelihood of a hypothesis based on available information.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where:

- **P (A|B):** Probability of hypothesis A on the observed event B.
- **P (B|A):** Probability of the evidence given that the probability of a hypothesis is true.
- **P (A):** Probability of hypothesis before observing the evidence.
- **P (B):** Probability of Evidence. **[23]**

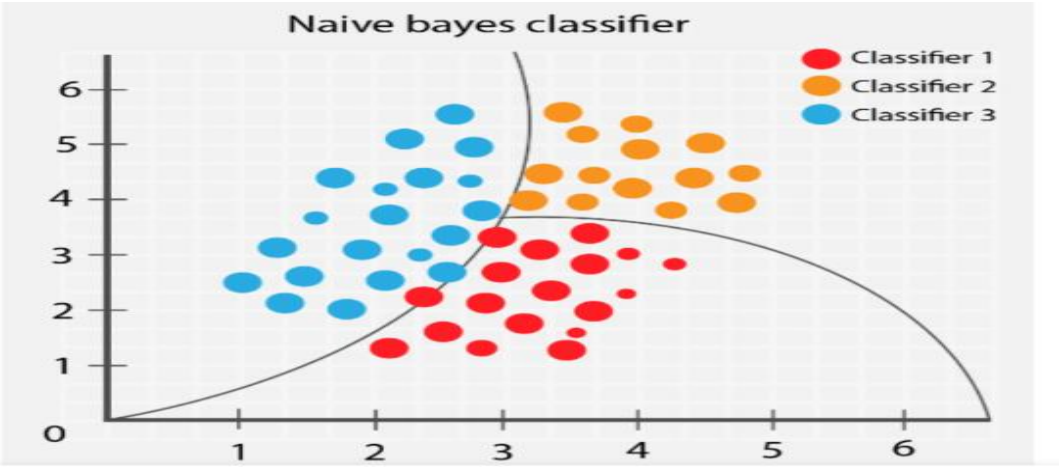

Figure 2.20 – Naïve Bayes Model [23]

CHAPTER 3:

PREDICTION OF MAINTENANCE AND STATE OF THE ART

1. Prediction of maintenance:

Maintenance prediction involves employing a range of techniques and methodologies to foresee the timing of maintenance requirements for systems, equipment, or infrastructure. Its primary objective is to enhance maintenance strategies, minimize downtime, boost operational efficiency, and reduce costs associated with unplanned or reactive maintenance.

Over the past years, the field of maintenance prediction has witnessed notable progress, primarily driven by the abundance of data, the development of machine learning and data analytics methodologies, and the increasing adoption of Internet of Things (IoT) technologies.

These advancements have paved the way for precise and proactive maintenance planning, ultimately resulting in enhanced reliability and performance of assets.

1.1. Stages:

Predictive maintenance encompasses the collection of asset data and extraction of relevant information to determine the optimal timing for maintenance activities. This process can be segmented into three key stages:

1.1.1. Data Collection

The foundation of predictive maintenance lies in acquiring high-quality data. Sensors are strategically deployed to gather real-time information about equipment performance and overall health. The specific data collected depends on the chosen monitoring techniques, which can encompass factors such as vibration, temperature, pressure, noise levels, or corrosion levels. **[24]**

1.1.2. Data Mining

Leveraging the capabilities of the Internet of Things (IoT), the collected data is transmitted from sensors to a centralized system or software for comprehensive analysis. By integrating data from diverse assets, the effectiveness and precision of predictive maintenance are greatly enhanced. The consolidated information provides a comprhensive view of the overall equipment condition and facilitates accurate predictions of maintenance requirements. **[24]**

1.1.3. Calculations and Machine Learning

Predictive maintenance goes beyond anomaly detection. Algorithms are developed and applied to offer prognoses. Initially based on equipment history, maintenance logs, and statistics, advanced Artificial Intelligence techniques can detect anomalies earlier, find correlations, and provide intelligent suggestions for preventing breakdowns. This gives rise to prescriptive maintenance. **[24]**

2. State of the art:

This section provides a comprehensive overview of the latest advancements, methodologies, techniques, and technologies in predictive maintenance. Drawing from recent academic literature, this section highlights the existing knowledge and progress in maintenance prediction. It highlights significant developments in data analytics, machine learning, and the integration of Internet of Things (IoT) technologies, highlighting their positive impact on enhancing maintenance strategies. In addition, this section examines successful case studies, industry best practices, and identifies potential research directions for the future. Through this exploration, we gain valuable insights into the current landscape of maintenance prediction, identify research gaps, and establish a solid foundation for the following chapter of this dissertation.

2.1. Work One:

[25] Özlem GÜVEN , 2Hasan ŞAHİN , 2022 , Predictive Maintenance Based On Machine Learning In Public Transportation Vehicles , Journal of Engineering Sciences and Researches, Vol 4 (1), Page 89 – 98 , https://dergipark.org.tr/en/pub/bjesr/issue/69536/1093519

2.1.1. Main Research Objective:

This research aims to implement a predictive maintenance approach in the public transport sector to mitigate issues such as service disruptions, delays, and accidents caused by unexpected vehicle breakdowns. By leveraging real-time vehicle health data from IoT sensors, the study applies classification techniques in machine learning to assess the probability of normal and malfunctioning vehicles. Fuzzy logic is employed for maintenance planning, generating fuzzy outputs to determine maintenance priorities. The overall goal is to detect a significant portion of vehicle faults through the application of the predictive maintenance approach. **[25]**

2.1.2. Application:

In their research, an integrated approach incorporating machine learning techniques and Fuzzy Logic was adopted to create a framework for predictive maintenance. The methodology encompassed the utilization of various classification algorithms such as random forest which is a combination of several decision trees, logistic regression, k-nearest neighbors, support vector machines, and naive Bayes (All of which were mentioned previously in Chapter 2). Through the synergistic application of these methods, the primary objective of this study was to augment the precision and dependability of maintenance predictions, leading to the optimization of maintenance planning and the mitigation of unexpected breakdowns in public transport vehicles. **[25]**

- **Fuzzy Logic:** A mathematical approach that allows for reasoning and decision-making under uncertainty by assigning degrees of truth and membership to variables, enabling more flexible analysis and decision-making processes.
- **Data-Division:** The data set used was divided into two parts, 75% for the training set, and 25% for the test set used for validation and accuracy calculation.

2.1.3. Obtained Results:

In this section, we explore the distinctions among the employed approaches and provide a comparative analysis of their performance in predictive maintenance.

	AUC	ACC
RF	0.99999984	0.99991348
LR	0.99896336	0.99630868
KNN	0.99981788	0.99985580
SVM	0.99994755	0.99855808
NB	0.99695318	0.99604914

Figure 3.1 – Results from Article 1

The study obtained various results, Random Forest demonstrated the best performance among the classifiers as shown in **Figure 3.1**. These weights indicate the likelihood of a vehicle being normal or healthy, while their inverse values indicate the maintenance urgency. Using fuzzy logic, the maintenance speed was categorized into low (long term), medium (medium term), and high (urgent) based on the probabilities of vehicles being normal. Triangular membership functions were employed to represent the probability of normality and the maintenance rates. **[25]**

- **AUC:** Area Under The Curve.
- **ACC:** Accuracy.

2.2. Work Two:

[26] Rune Prytz, 2014, Machine learning methods for vehicle predictive maintenance using off-board and on-board data, [Licentiate Thesis, Halmstad University Dissertations no. 9], Halmstad University, https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A789498&dswid=8005

2.2.1. Main Discussion:

The thesis contains 4 papers, and discusses the increasing importance of vehicle uptime in the transport industry and the need for improved maintenance planning. The research explores unsupervised and supervised methods for predicting vehicle maintenance using large amounts of data. A telematics gateway allows vehicles to communicate with a back-office system, which analyses data for anomalies and associates them with repair history. The thesis investigates different data representations and deviation detection techniques. The first method, COSMO, is an unsupervised approach that identifies interesting data representations and isolates deviating vehicles using a consensus-based approach. The second method is a supervised classification that uses earlier collected and aggregated vehicle statistics, labelled with repair history, to train a classifier for predicting maintenance. The case study focuses on failures of the vehicle air compressor using AB Volvo's database of usage statistics. **[26]**

2.2.2. Application:

The research encompassed four papers and employed various machine learning techniques to develop a predictive maintenance framework. These techniques included consensus self-organizing models (COSMO), linear regression, support vector machines, and random forests. By integrating these methods, the study aimed to enhance the accuracy and reliability of maintenance predictions, ultimately optimizing maintenance planning and reducing unforeseen breakdowns in public transport vehicles. **[26]**

2.2.3. Obtained Results:

- **Paper I: Towards relation discovery for diagnostics**

The paper presents an unsupervised method for discovering relationships between measured signals in Volvo trucks during normal operations and faults. The method involves finding valid models based on a MSE threshold and studying model parameters over time. An ambient filtering method is also introduced. The evaluation on a controlled fault injection experiment shows successful detection of one out of four faults. **[27]**

- **Paper II: Wisdom of Crowds for Self-organized Intelligent Monitoring of Vehicle Fleets**

Using the COSMO algorithm, this paper applies it to detect failures in a real-world setting with 19 buses. Failures related to the cooling fan, heat load of the engine, and wheel speed sensors were detected. The algorithm outperformed existing on-board diagnostic algorithms in detecting upcoming failures. The paper also analyses downtime and highlights the need to reduce it from 11% to the operator's goal of 5%. **[26]**

COSMO: Consensus Self-Organizing Models

- **Paper III: Analysis of Truck Compressor Failures Based on Logged Vehicle Data**

This paper explores off-board data sources (LVD and VSR) for predicting air compressor failures. Three classifiers are evaluated, considering F-score and a cost function. The challenges of non-iid datasets and learning individual truck behaviour are discussed. The results suggest the viability of using off-board data for predicting maintenance, but more work is required. **[28]**

LVD: Logged Vehicle Data database.

VSR: The Vehicle Maintenance database.

- **Paper IV: Predicting the Need for Vehicle Compressor Repairs Using Maintenance Records and Logged Vehicle Data**

Introducing an off-board method using supervised machine learning, this paper focuses on predicting air compressor failures in Volvo FH13 vehicles. The method evaluates the vehicle before a scheduled workshop visit and flags it as faulty if the air compressor is predicted to fail. The evaluation considers a cost function and shows an economical benefit, but with a relatively high level of false repair claims. The classifier's performance trade-off is illustrated, and the optimal case is when the sensitivity is 0.4 and specificity is 0.9. **[29]**

3. Conclusion:

In conclusion, the exploration of the approaches in predictive maintenance has uncovered a wide range of possibilities for creating effective tools. Theoretical methodologies like COSMO provide a distinctive and unsupervised means of identifying patterns and anomalies within vehicle systems. Meanwhile, machine learning techniques such as linear regression, support vector machines, and random forests offer robust frameworks for predictive modelling and analysis.

By integrating these diverse approaches, an exciting opportunity arises to develop a comprehensive and dependable predictive maintenance tool. By capitalizing on the strengths of each method, we can enhance the precision and accuracy of maintenance predictions, resulting in improved planning strategies and the prevention of unexpected breakdowns in public transport vehicles.

CHAPTER 4:

RESULTS AND DISCUSSION

1. E.T.U.S(L'Etablissement de Transport Urbain et Suburbain):

ETUS (L'Etablissement de Transport Urbain et Suburbain) is a leading bus company in Algeria, specializing in providing urban and suburban transportation services. With a strong commitment to efficiency and passenger satisfaction, ETUS has established itself as an important company in the public transportation sector. Through its extensive bus network and dedication to quality service, ETUS serves as a vital transportation lifeline for thousands of passengers across the country, having a facility in each province.

- **E.T.U.S. M'sila:**

ETUSM (Enterprise de Transport Urbain et Suburbain de M'sila) is one of the prominent facilities operated by ETUS (Enterprise de Transport Urbain et Suburbain), the leading bus company in Algeria.

ETUSM is Located in the city of M'sila, and plays a vital role in providing efficient and reliable urban and suburban transportation services to the local population. As part of ETUS's network, it serves as a crucial transportation hub, facilitating seamless travel within M'sila and its surrounding areas. The facility encompasses a well-maintained fleet of buses, maintenance workshops, and administrative offices, all working together to ensure the smooth operation of public transportation services. With a dedicated team of skilled personnel, ETUSM upholds high standards of service quality, safety, and convenience for passengers relying on public transportation in the region.

2. Tool Proposition and Description:

2.1. Introduction:

This chapter presents a proposal and discussion of an AI-based predictive maintenance tool designed specifically for E.T.U.S. M'sila, a bus company operating in M'sila. The primary objective of this tool is to address the operational challenges faced by the company's bus fleet and unlock new possibilities for proactive maintenance strategies. Using advanced AI algorithms, the proposed predictive maintenance tool analyses data from buses breakdowns to predict potential faults or failures, enabling timely interventions and minimizing unplanned downtime. Through a validation process, the tool's effectiveness and reliability have been evaluated, providing valuable insights and performance assessments specific to E.T.U.S.M's bus fleet. This chapter provides an overview of the proposed tool, validation results, and a critical discussion of its implications for E.T.U.S.M's maintenance operations. By integrating AI technologies into maintenance practices, this tool has the potential to revolutionize their approach to maintenance, optimizing fleet reliability, reducing costs, and improving overall operational efficiency for the company.

2.2. Objectives:

The AI-based predictive maintenance tool developed for E.T.U.S. M'sila Bus Company aims to address specific operational challenges and enhance maintenance practices within their bus fleet. The tool is designed to achieve the following objectives:

2.2.1. Predictive Fault Detection:

The tool will develop algorithms and models to analyse real-time data from buses, enabling accurate prediction of potential faults or failures before they occur. By focusing on proactive maintenance, the tool will facilitate timely interventions and minimize unplanned downtime.

2.2.2. Optimize Maintenance Scheduling:

Utilizing the predictions generated by the tool, maintenance scheduling can be optimized. This objective aims to streamline resource allocation, minimize disruptions to operations, and enhance the utilization of maintenance personnel and equipment.

2.2.3. Cost Reduction:

By implementing proactive maintenance strategies, the tool seeks to reduce overall maintenance costs associated with reactive repairs and unplanned downtime. Resource allocation will be optimized, replacement part costs minimized, and the efficiency of maintenance operations improved.

2.2.4. Improve Fleet Reliability:

The tool will enhance the overall reliability and availability of E.T.U.S. M'sila's bus fleet by identifying and addressing potential issues in advance. This objective aims to increase the operational readiness of buses, reduce service disruptions, and improve customer satisfaction.

The AI-based predictive maintenance solution will be extremely helpful to E.T.U.S. M'sila for them to optimize their maintenance procedures, by achieving these goals, the organization will be able to quickly address possible problems, which will improve operational effectiveness, cut expenses, and the dependability of their bus fleet. E.T.U.S. M'sila will ultimately experience improved performance overall and increased customer satisfaction as a result of the tool's use, further solidifying their status as a dependable and effective bus service provider.

2.3. Process:

The following procedure was used to create the tool, and each component is explained along with how it is to be used:

2.3.1. Data-Collection:

The technology depends on the gathering and integrating of pertinent data, such as sensor data from buses and maintenance records. The goal is to produce a complete and trustworthy dataset, hence the scope includes designing and putting into practice data collection methods.

2.3.2. AI Algorithm Development:

The tool involves the development and implementation of advanced AI algorithms, such as machine learning or deep learning models, to analyze the collected data and generate predictive

insights. The scope includes selecting appropriate algorithms, training and fine-tuning models, and integrating them into the tool's framework.

2.3.3. Monitoring and Prediction:

The tool monitors data from buses and provides timely predictions of potential faults or failures. The scope covers the development of algorithms and techniques for data processing, anomaly detection, and fault prediction.

2.3.4. Integration with Maintenance Workflow:

The tool is designed to seamlessly integrate with E.T.U.S. M'sila's existing maintenance workflow and systems. This scope includes considerations for data exchange, communication protocols, and integration with maintenance management software or systems.

It is important to provide additional information specific to E.T.U.S. M'sila's bus fleet, such as the size of the fleet, the types of buses, and any unique maintenance challenges they face. This information will help establish the context and align the objectives and process of the tool with the specific needs of E.T.U.S. M'sila.

3. Description of the Models used:

3.1. Introduction:

The efficient operation of bus fleets is essential for providing reliable and sustainable public transport services in urban areas. Buses serve as an essential means of transportation, carrying a large number of passengers each day and contributing to the general mobility of the city. To ensure the smooth functioning of bus fleets, it is important to understand the data associated with bus operations and malfunctions.

This graduate note focuses on analysing a dataset that provides insight into bus operations and breakdowns. The dataset was obtained from a reputable public transportation agency and covers a period of one year. It contains information that is recorded periodically every day or when a malfunction occurs on the bus.

Within the data set, different variables provide a picture of bus operations. These variables include bus identifiers, driver identifiers, time and location of the malfunction, in addition to the cause, air temperature, number of passengers that day, up to the last date of the last malfunction. Analysis of these variables can help identify patterns, correlations, and potential areas for improvement in maintenance practices and operational efficiency.

However, it is important to consider the limitations of the data set. Accuracy and completeness of recorded information may vary, and some malfunctions may not be captured or reported. Despite these limitations, the dataset serves as a valuable resource for understanding the challenges facing bus fleets and exploring ways to improve their performance.

By analysing the dataset, this graduate note aims to provide insight into the factors affecting bus operations and breakdowns. The findings and recommendations from this analysis can contribute to improving the overall maintenance strategies, reliability and quality of service for bus fleets. Ultimately, this research aims to support the development of efficient and sustainable public transport systems, benefiting both commuters and urban communities as a whole.

3.2. Dataset Description:

The dataset covers a broad time frame that includes a full year of operations. This duration allows for a comprehensive analysis of seasonal variations, long-term trends, and the impact of external factors on bus operations and breakdowns. With an emphasis on accuracy and completeness, the dataset contains a wide range of variables that capture critical aspects of bus operations. These variables include:

3.2.1. Bus ID:

A Bus ID is a unique identifier that is assigned to each individual vector within the dataset. Allows tracking of bus information and performance metrics. From a reliable source of bus ID, it becomes possible to assess bus reliability and maintenance requirements and to study bus behavior, these identifiers are in the form of numbers.

3.2.2. Driver ID:

The dataset includes a variable called Driver ID, which serves as a unique identifier assigned to each bus driver within the fleet. This identifier allows for tracking and analysis of driver-specific performance metrics and potential correlations with operational outcomes.

3.2.3. Weather:

The dataset includes a weather column, which provides information about the weather conditions prevailing at the time the data was collected. This column provides valuable insights into the impact of weather on bus operations and potential associations between weather conditions and other variables within the dataset.

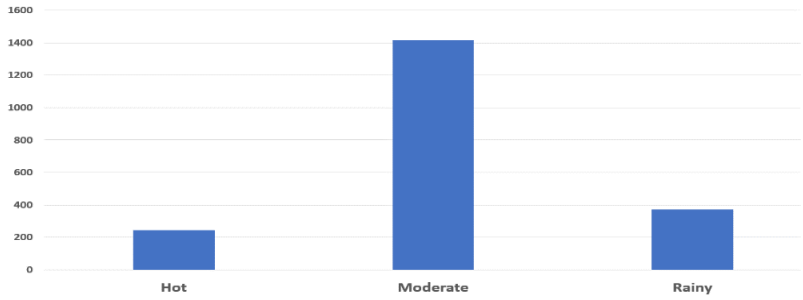

Figure 4.1 – Count of Weather

3.2.4. Date, Time, and Period:

The dataset includes a datetime column, which provides information about the week number for the specific month, time, and date that each record was collected. This column allows temporal analysis of different variables and events within the dataset.

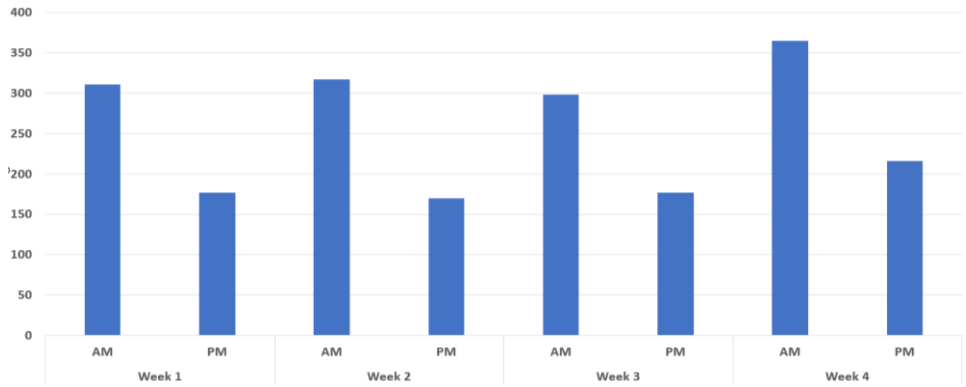

Figure 4.2 – Count of Date

3.2.5. Reason:

The dataset includes a Cause column, which provides information about the specific cause or reason associated with a recorded event or event. This column provides valuable insights into contributing factors to bus breakdowns, delays or other operational incidents.

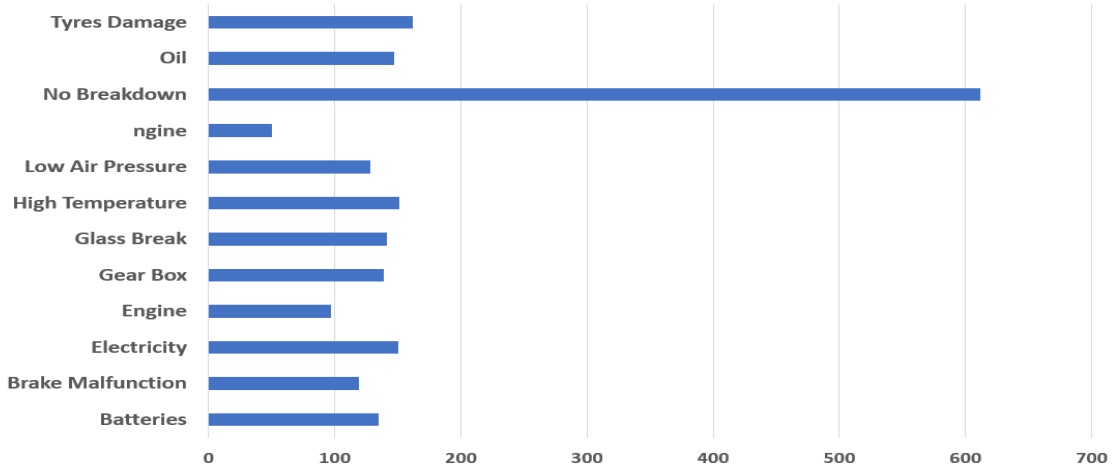

Figure 4.3 – Count of Reason

3.2.6. Place:

The dataset includes a Place column, which provides information about the specific location or place associated with each recorded event or observation. This column offers valuable insights into the spatial aspects of bus operations and allows for the analysis of location-specific patterns or trends.

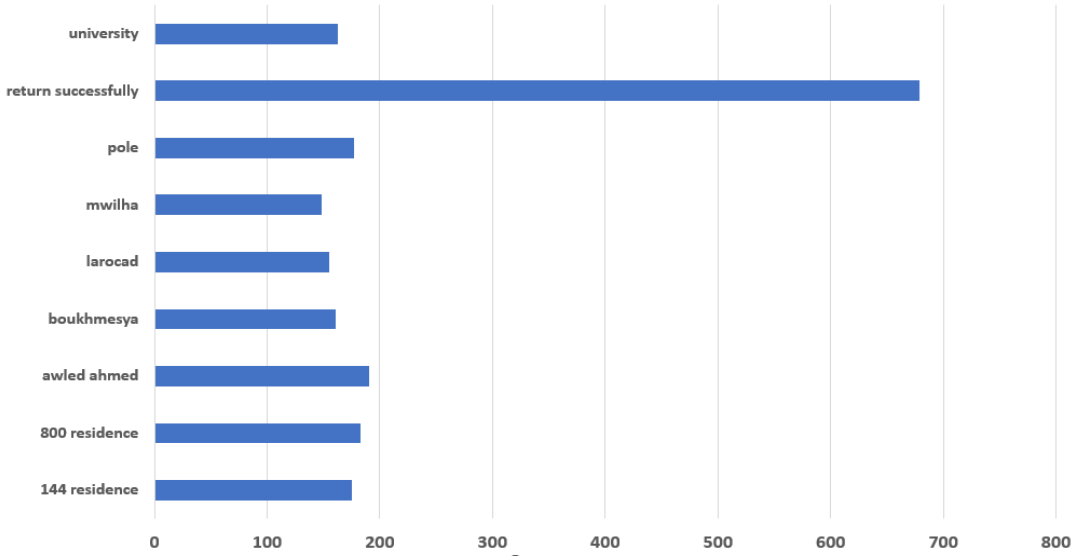

Figure 4.4 – Count of Place

3.2.7. NumOfPass:

The dataset includes a NumOfPass column, which provides information about the number of passengers on each bus during an event or the number of passengers that day if the bus returns safely. This column provides insights into passenger volume and allows for the analysis of trends and patterns regarding ridership.

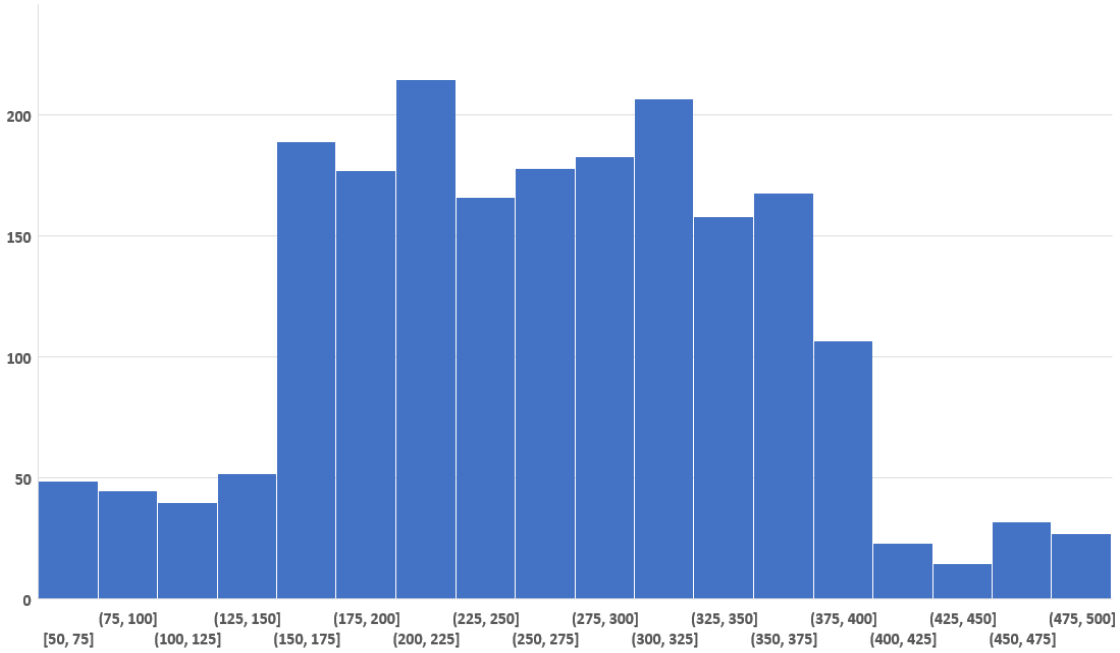

Figure 4.5 – Count of Number of Passengers

3.2.8. LastMalfunction:

The dataset includes a LastMalfunction column, which provides information about the last recorded failure for each bus. This column provides insights into the maintenance history and timing of the last reported failure for each bus.

It is important to note that the dataset undergoes regular updates and maintenance, ensuring the inclusion of the most recent data and preserving its relevance for ongoing analysis and decision-making processes.

3.3. sample from the dataset:

ID	Bus_Number	DriverId	Hour	Period	Route	DATE	Weather	Reason	NumOfPass	Place	LastMalfunction	Class
1	608	5	02:44	PM	16	Week 4	Moderate	No Breakdown	360	return successfully	164	Yes
2	403	17	08:44	AM	16	Week 2	Moderate	No Breakdown	184	return successfully	203	Yes
3	624	29	01:03	PM	16	Week 4	Hot	Low Air Pressure	101	return successfully	350	Yes
4	475	27	03:46	PM	11	Week 1	Hot	Electricity	74	awled ahmed	282	Yes
5	475	11	11:12	AM	13	Week 3	Moderate	Oil	337	mwilha	303	No
6	608	18	02:35	PM	13	Week 3	Moderate	No Breakdown	372	return successfully	61	Yes
7	608	12	11:47	AM	13	Week 4	Moderate	Tyres Damage	463	awled ahmed	274	No
8	622	17	09:31	AM	17	Week 2	Rainy	Engine	229	mwilha	262	No
9	492	18	02:12	PM	17	Week 2	Moderate	No Breakdown	316	return successfully	205	Yes
10	623	2	11:52	AM	16	Week 3	Moderate	Low Air Pressure	342	university	284	No
11	620	16	05:05	PM	17	Week 1	Moderate	No Breakdown	318	return successfully	44	Yes
12	489	12	08:02	AM	16	Week 2	Rainy	Brake Malfunction	200	university	355	No
13	608	23	04:06	PM	11	Week 2	Moderate	Electricity	361	university	282	Yes
14	494	7	08:00	AM	17	Week 4	Moderate	Tyres Damage	450	mwilha	390	No
15	622	11	10:37	AM	12	Week 2	Moderate	Oil	216	mwilha	257	No
16	619	2	11:48	AM	16	Week 1	Moderate	High Temperature	195	800 residence	309	No
17	621	18	09:51	AM	13	Week 2	Moderate	Gear Box	155	awled ahmed	321	Yes
18	619	10	08:51	AM	16	Week 2	Rainy	Electricity	320	800 residence	332	Yes
19	626	1	04:51	PM	17	Week 3	Rainy	Electricity	283	pole	131	No
20	626	16	10:30	AM	16	Week 4	Moderate	Tyres Damage	485	boukhmesya	291	No

Figure 4.6 – Data-Set Sample

3.4. data types:

Within a dataset, different data types are used to capture and represent different types of information. These data types provide structure and meaning to the data set, allowing the data to be stored, processed, and analyzed efficiently. Here are some common data types that are often found in datasets.

3.4.1. Numeric data:

Numeric data types represent numerical values such as integers or real numbers. They are used to specify measurements or quantities related to a data set. Examples include the number of passengers, the line number, or the hours the fault occurred. Numeric data types enable mathematical calculations, statistical analysis, and the visualization of trends or patterns.

3.4.2. Serial data:

Categorical data types represent separate and distinct categories or labels. They are used to classify data into specific groups or categories. Examples include bus ID, driver ID, weather

conditions, or fault types. Categorical data types make it easy to collect, group, and compare data based on specific categories.

3.4.3. Date and time data:

Date and time data types represent specific points in time or durations. They are used to capture temporal information, such as the date and time of bus operations, failure states, or data collection. Date and time data types enable temporal analysis, including identification of patterns over time, time-based comparisons, or trend analysis.

3.4.4. Boolean data:

Boolean data types represent Boolean values of true or false. They are used to indicate the presence or absence of a particular condition or trait. Examples include indicators of whether or not a failure has occurred. Boolean data types facilitate filtering, conditional analysis, or decision making based on Boolean conditions.

3.4.5. Spatial data:

Spatial data types represent geographic or spatial information, such as coordinates or boundaries. Used to capture location-specific details, spatial data types enable spatial analysis, mapping, or geospatial visualization to understand the relationships between data and geographic features.

By taking advantage of these different types of data, data sets become structured and meaningful, allowing comprehensive analysis and interpretation of the information they contain. Data types play an important role in shaping data management, analysis techniques, and decision-making processes based on the content and objectives of the dataset.

3.5. Relationships and connections:

Data analysis provides an opportunity to explore potential relationships and associations between variables, allowing valuable insights into their interrelationships. One example worth studying is the possible relationship between the age of the bus, the frequency of breakdowns, and weather and climatic conditions.

By investigating the relationships between bus age, breakdown frequency, and atmospheric and climatic conditions, we can gain insights into how external factors affect bus performance and maintenance requirements. This analysis enables us to understand whether certain weather or climatic conditions, such as temperature extremes or inclement weather, have an impact on the frequency of bus breakdowns.

For example, we can explore whether older buses are more prone to breakdowns during periods of extreme heat or cold. Such an analysis could provide insights into how climate affects the reliability and performance of older buses. It may reveal patterns indicating that certain weather conditions exacerbate the potential for breakdowns on older buses, requiring more maintenance and attention during these periods.

Understanding the relationships and correlations between bus life, failure frequency, and weather and climatic conditions supports effective resource allocation and decision-making processes. Transportation agencies can prioritize maintenance efforts, allocate resources based on specific weather conditions or seasons, and implement preventive maintenance strategies to mitigate the impact of bad weather on bus performance.

Furthermore, this analysis contributes to long-term planning and budgeting considerations, providing insights into the potential need for fleet renewal or the implementation of climate-specific maintenance protocols. By taking into account the relationships between bus age, breakdown frequency, and weather and climatic conditions, transportation agencies can improve their fleet management strategies to enhance operational efficiency, service reliability, and passenger satisfaction.

3.6. Data pre-processing:

Data preprocessing is a vital step in data analysis that involves cleaning, transforming, and organizing the data before analysis. It aims to improve data quality, address missing values and outliers, and ensure the data is in a suitable format for analysis. By performing data preprocessing, analysts can enhance the accuracy and reliability of their results and derive meaningful insights from the data

3.6.1. Data Cleaning:

In order to display the missing data, we write the following code:

```
print(pd.DataFrame({'missing:': data.isnull().sum()}))
```

Missing data results: We have 2 missing data in the Route column.

Since the missing data is not large, we will delete the row that contains missing data by writing the following code:

```
data = data[data['Route'].notna()]
```

In order to avoid analysis bias, we remove the duplicated lines we write the following code:

```
 data = data.drop_duplicates()
```

in order to discover Noise Data, we write the following code:

```
plt.figure()
data[['Bus_Number','DriverId':,'Place','Period','Route','NumOfPass','Weather','Class']].boxplot()
plt.title("Noise Data")
plt.show()
```

And by doing that process, we achieved the following result:

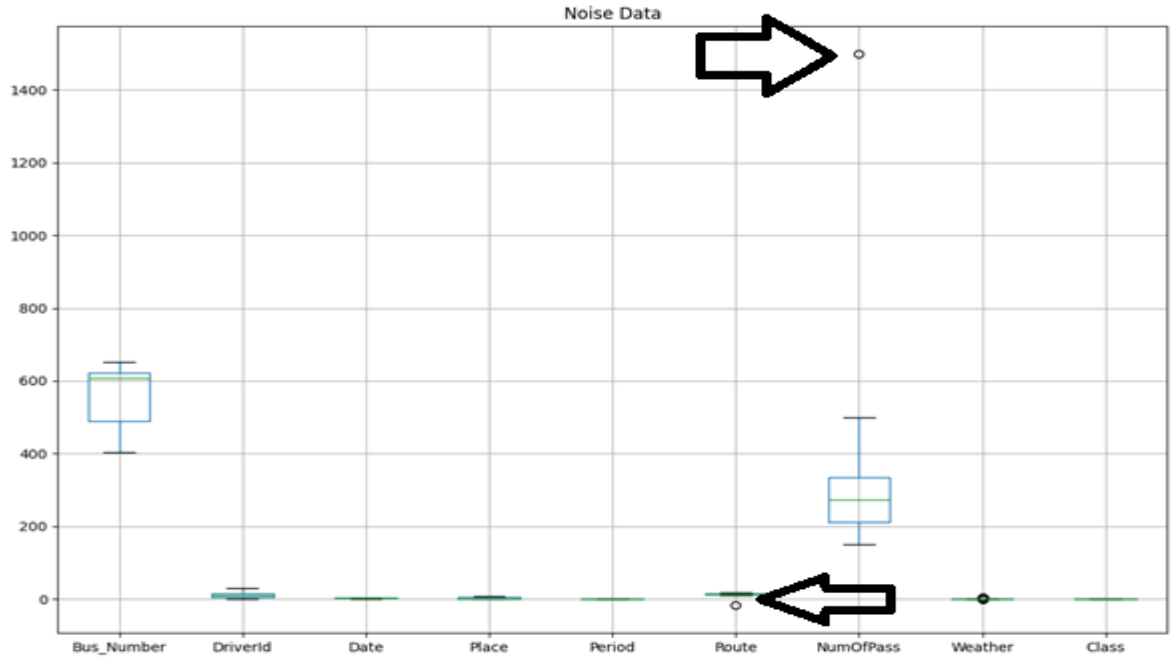

Figure 4.7 – Noise Data Results

We notice that there are two outliers in Route column and NumOfPass column, to know these values, we write the following code:

```
NoiseData1 = data[data.Route < 0]
NoiseData2 = data[data.NumOfPass > 600]

print(NoiseData1.Route)
print(NoiseData2.NumOfPass)
```

In order to handle the abnormal values, we write the following code:

```
data.at[NoiseData1.index[0],'Route'] = 17
data.at[NoiseData2.index[0],'NumOfPass'] = 150
```

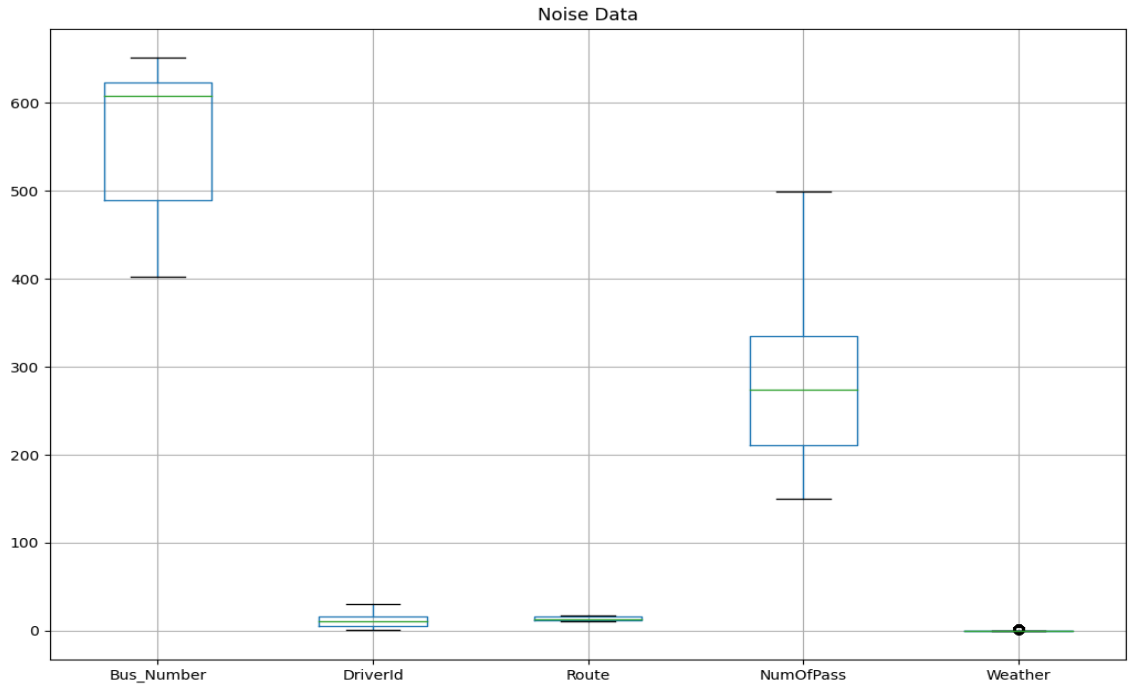

Figure 4.8 – Noise Data after Cleaning

3.6.2. Data Selection:

We selected a dataset focused on bus malfunctions from a transportation agency's maintenance records. This dataset provides valuable information for analyzing the frequency and types of bus malfunctions and improving maintenance strategies, and we delete the data we don't need with the following code:

```
data.drop(['ID'], axis=1, inplace=True)
data.drop(['Hour'], axis=1, inplace=True)
data.drop(['Reason'], axis=1, inplace=True)
data.drop(['Place'], axis=1, inplace=True)
```

3.6.3. Data Transformation:

Converting data into a digital format through preprocessing is important because it ensures compatibility with modern systems, enhances accessibility and efficiency, enables data integration, improves accuracy and reliability, and facilitates long-term data preservation.

This is a picture of the data after converting it into numeric data using Excel functions:

ID	Bus_Number	DriverId	Hour	Period	Route	DATE	Weather	Reason	NumOfPass	Place	LastMalfunction	Class
1	608	5	02:44	2	16	4	1	0	360	0	164	1
2	403	17	08:44	1	16	2	1	0	184	0	203	1
3	624	29	01:03	2	16	4	3	9	101	0	350	1
4	475	27	03:46	2	11	1	3	4	74	3	282	1
5	475	11	11:12	1	13	3	1	10	337	6	303	0
6	608	18	02:35	2	13	3	1	0	372	0	61	1
7	608	12	11:47	1	13	4	1	1	463	3	274	0
8	622	17	09:31	1	17	2	2	5	229	6	262	0
9	492	18	02:12	2	17	2	1	0	316	0	205	1
10	623	2	11:52	1	16	3	1	9	342	8	284	0
11	620	16	05:05	2	17	1	1	0	318	0	44	1
12	489	12	08:02	1	16	2	2	3	200	8	355	0
13	608	23	04:06	2	11	2	1	4	361	8	282	1
14	494	7	08:00	1	17	4	1	1	450	6	390	0
15	622	11	10:37	1	12	2	1	10	216	6	257	0
16	619	2	11:48	1	16	1	1	8	195	1	309	0
17	621	18	09:51	1	13	2	1	6	155	3	321	1
18	619	10	08:51	1	16	2	2	4	320	1	332	1
19	626	1	04:51	2	17	3	2	4	283	7	131	0
20	626	16	10:30	1	16	4	1	1	485	4	291	0

Figure 4.9 – Numeric Data-Set Transformation

3.6.4. Data Normalization:

Limiting the data to a range from 0 to 1, through data normalization, offers benefits such as improved variable comparison, enhanced algorithm convergence, avoidance of bias, stability, interpretability, and compatibility with certain models. It ensures fair comparison, prevents dominance of variables with larger scales, and improves the performance and interpretability of data analysis and modelling.

In order to do that, we write the following code:

```
scaler = MinMaxScaler()
data = pd.DataFrame(scaler.fit_transform(data), columns=data.columns)
```

3.6.5. Data Partitioning:

Data partitioning is a critical step in model development, allowing for effective evaluation and validation. By splitting the dataset into different partitions, such as a training set, validation set, and test set, we can assess the performance and generalization capabilities of the model.

In our study, we allocated 80% of the dataset to the training set, and the remaining 20% to the test set. The larger proportion assigned to the training set ensures that the model has ample data to learn from and capture underlying patterns. The validation set is used to fine-tune the model's parameters and assess its performance during the training process. Finally, the test set, which remains unseen by the model during training, provides an unbiased evaluation of its ability to generalize to new, unseen data.

Below is the partition code:

```
attributes = data.drop('Class', axis=1)
target = data['Class']
```

The following display shows the training data and testing:

attributes :

	Bus_Number	DriverId	Period	...	Weather	NumOfPass	LastMalfunction
0	0.823293	0.137931	1.0	...	0.0	0.690423	0.396419
1	0.000000	0.551724	0.0	...	0.0	0.298441	0.496164
2	0.887550	0.965517	1.0	...	1.0	0.113586	0.872123
3	0.289157	0.896552	1.0	...	1.0	0.053452	0.698210
4	0.289157	0.344828	0.0	...	0.0	0.639198	0.751918
...	...	...	...	...	...	...	...
2024	0.361446	0.206897	1.0	...	0.0	0.142539	0.590793
2025	0.895582	0.137931	0.0	...	0.0	0.461024	0.383632
2026	1.000000	0.137931	0.0	...	0.0	0.224944	0.762148
2027	0.875502	0.724138	1.0	...	0.5	0.412027	0.987212
2028	0.365462	0.310345	0.0	...	0.0	0.556793	0.808184

target :

0	1.0
1	1.0
2	1.0
3	1.0
4	0.0
...	
2024	1.0
2025	1.0
2026	0.0
2027	0.0
2028	0.0

Figure 4.10 – Training and Testing Data

3.7. K-Nearest Neighbours Model (KNN):

3.7.1. Code:

```
# Create the K-Nearest Neighbors classifier
classifier = KNeighborsClassifier()

# Define the parameter grid for hyperparameter tuning
param_grid = {'n_neighbors': [3, 5, 7], 'weights': ['uniform',
'distance']}

# Perform K-Fold cross-validation
kfold = KFold(n_splits=5, shuffle=True, random_state=42)

# Perform grid search using cross-validation
grid_search = GridSearchCV(classifier, param_grid, cv=kfold)
grid_search.fit(X_train, y_train)

# Get the best parameters found during grid search
best_params = grid_search.best_params_


cv_results = cross_val_score(classifier, X_train, y_train,
cv=kfold)

# Calculate the mean accuracy across all folds
accuracy_scores = cv_results.mean()

# Create the best K-Nearest Neighbors classifier using the best
parameters
best_classifier = KNeighborsClassifier(**best_params)
best_classifier.fit(X_train, y_train)

# Predict on the test set using the best classifier
y_pred = best_classifier.predict(X_test)

# Calculate the accuracy on the test set
test_accuracy = accuracy_score(y_test, y_pred)

# Generate the classification report
cls_rep = metrics.classification_report(y_test.values, y_pred)
```

3.7.2. Model Description:

K-Nearest Neighbours (KNN) is a simple and versatile machine learning algorithm used for classification and regression tasks. It assigns a class label or predicts a value for a new data point based on the class labels or target values of its k nearest neighbours in the training data. KNN does not make any assumptions about the data distribution and works well with numerical and categorical features. However, it requires the choice of k and the distance metric, and its prediction time can increase with larger datasets. Overall, KNN is a popular choice for its simplicity and interpretability in various applications.

3.7.3. Data Pre-processing:

- **Data clean-up:**

Any missing values and outliers were addressed by deleting the missing values lines and adjusting the affected data points.

- **Feature scaling:**

To ensure equal importance of features, they are scaled to a similar range with all digits between 0 and 1.

- **Categorical coding variables:**

Categorical variables were converted into numeric representations

- **Feature selection:**

Relevant features were selected based on their association with the target variable

- **Split test train:**

The dataset was divided into training and test sets to evaluate the performance of the model. The split is 80% for training and 20% for testing.

By carrying out these pre-processing steps, the data is prepared for the KNN model, ensuring that it is clean, properly measured, and in a format suitable for effective training and prediction.

3.7.4. Training and Evaluation of the KNN Model:

- **Choosing the Value of k:**

Different values of k, such as 3, 5, and 7, are tested. The performance of the model is evaluated using cross-validation, where the training set is further divided into multiple folds. The value of k that yields the highest cross-validation accuracy is chosen.

- **Computing Nearest Neighbours:**

For each data point in the test set, the KNN algorithm calculates the distance to all data points in the training set. The nearest neighbours to the test data point are identified based on the chosen distance metric.

3.7.5. Evaluation:

Classification Report:

	precision	recall	f1-score	support
0.0	0.81	0.82	0.82	220
1.0	0.79	0.77	0.78	186
accuracy			0.80	406
macro avg	0.80	0.80	0.80	406
weighted avg	0.80	0.80	0.80	406

Figure 4.11 – KNN Evaluation

K-Nearest Neighbor algorithm achieved an accuracy rate of **80**% and an F1 score of **78**%.

Accuracy: means the algorithm's overall correctness in classifying instances, representing the percentage of accurately classified instances.

The F1 score: combines precision and recall, creating a balanced measure of the algorithm's performance. A 78% F1 score suggests a favorable trade-off between precision and recall, indicating accurate positive predictions and reduced false negatives.

3.7.6. Confusion Matrix:

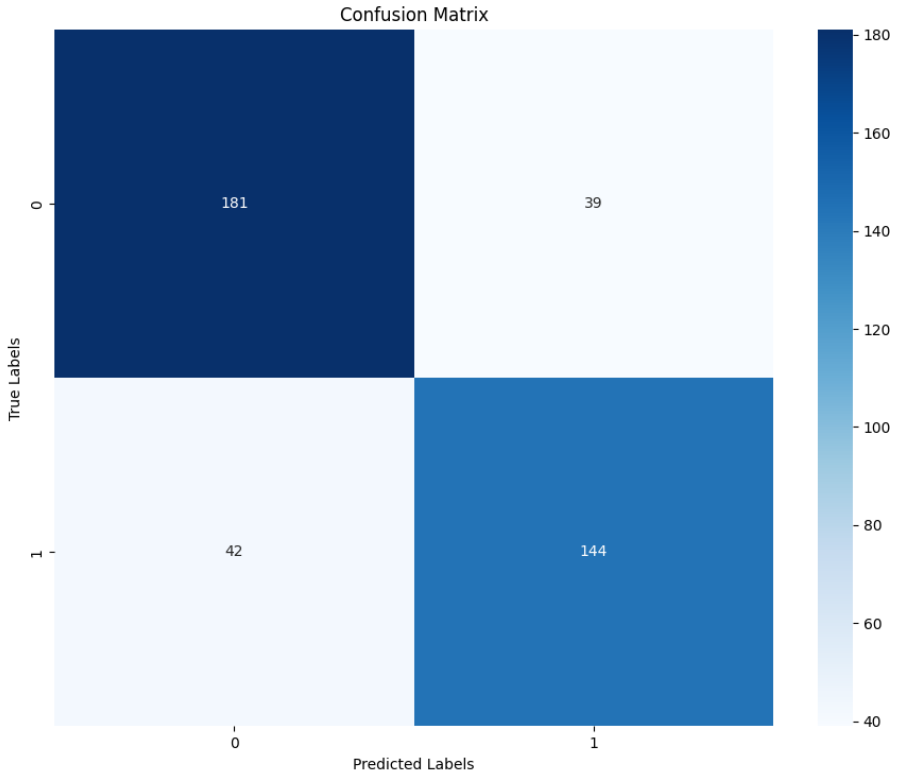

Figure 4.12 – KNN Confusion Matrix

From the confusion matrix, we observed that the KNN model correctly predicted 181 instances as positive and 144 instances as negative. However, it had 39 false positive predictions and 42 false negative predictions; this indicates that the model is performing well, although it may have some difficulty in identifying certain cases.

3.8. Decision Tree Model:

3.8.1. Code:

```
# Create the Decision Tree classifier
classifier = DecisionTreeClassifier()

# Define the parameter grid for hyperparameter tuning
param_grid = {'max_depth': [3, 5, 7], 'criterion': ['gini',
'entropy']}

# Perform K-Fold cross-validation
kfold = KFold(n_splits=5, shuffle=True, random_state=42)


# Perform grid search using cross-validation
grid_search = GridSearchCV(classifier, param_grid, cv=kfold)
grid_search.fit(X_train, y_train)

# Get the best parameters found during grid search
best_params = grid_search.best_params_

cv_results = cross_val_score(classifier, X_train, y_train,
cv=kfold)

# Calculate the mean accuracy across all folds
accuracy_scores = cv_results.mean()

# Create the best Decision Tree classifier using the best
parameters
best_classifier = DecisionTreeClassifier(**best_params)
best_classifier.fit(X_train, y_train)

# Predict on the test set using the best classifier
y_pred = best_classifier.predict(X_test)

# Calculate the accuracy on the test set
test_accuracy = accuracy_score(y_test, y_pred)

# Generate the classification report
cls_rep = metrics.classification_report(y_test.values, y_pred)
```

3.8.2. Model Description:

The decision tree model is a transparent and interpretable algorithm that uses a tree-like structure to make predictions based on input features. It splits the data into branches based on feature values and assigns class labels to leaf nodes. The model can handle different types of data and provides insights into the decision-making process.

3.8.3. Training and Evaluation of the Decision Tree Model:

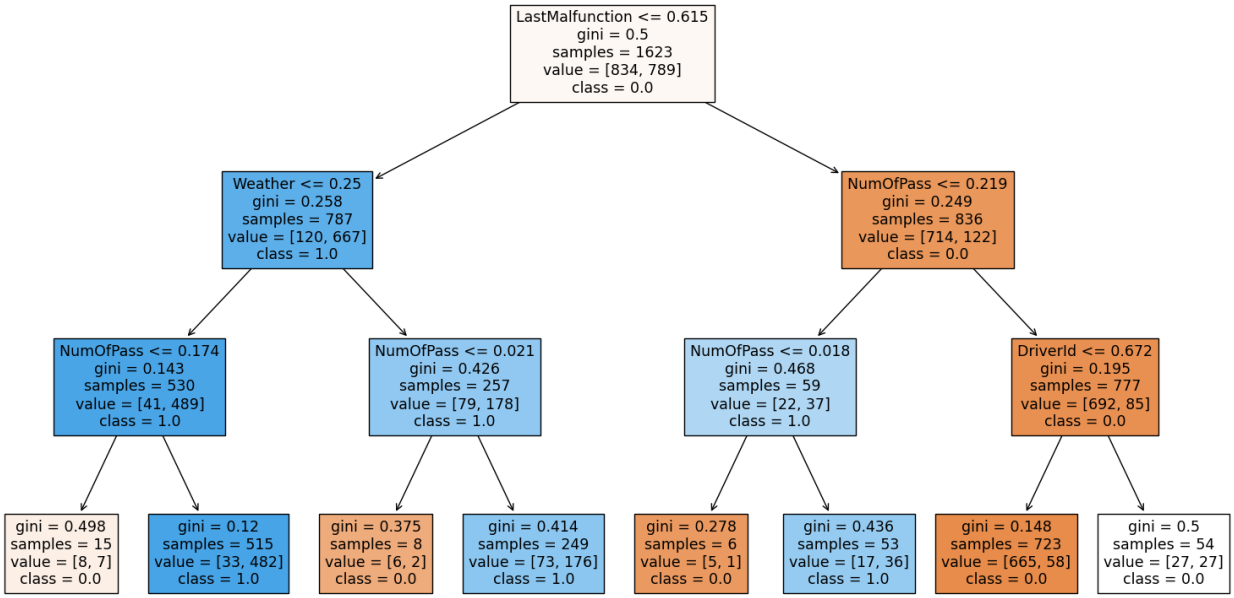

Figure 4.13 – Decision Tree Result

The decision tree algorithm works by recursively splitting the dataset based on features to create a tree-like model. Each node represents a decision based on a specific feature, leading to a predicted outcome at the leaf nodes. The model learns decision rules from the data and uses them to make predictions.

3.8.4. Evaluation:

Classification Report:

	precision	recall	f1-score	support
0.0	0.87	0.87	0.87	220
1.0	0.84	0.84	0.84	186
accuracy			0.86	406
macro avg	0.86	0.86	0.86	406
weighted avg	0.86	0.86	0.86	406

Figure 4.14 – Decision Tree Evaluation

The decision tree model achieved an impressive accuracy of **86**%, and an F1 score of **84**% on the test set, which is excellent.

3.8.5. Confusion Matrix:

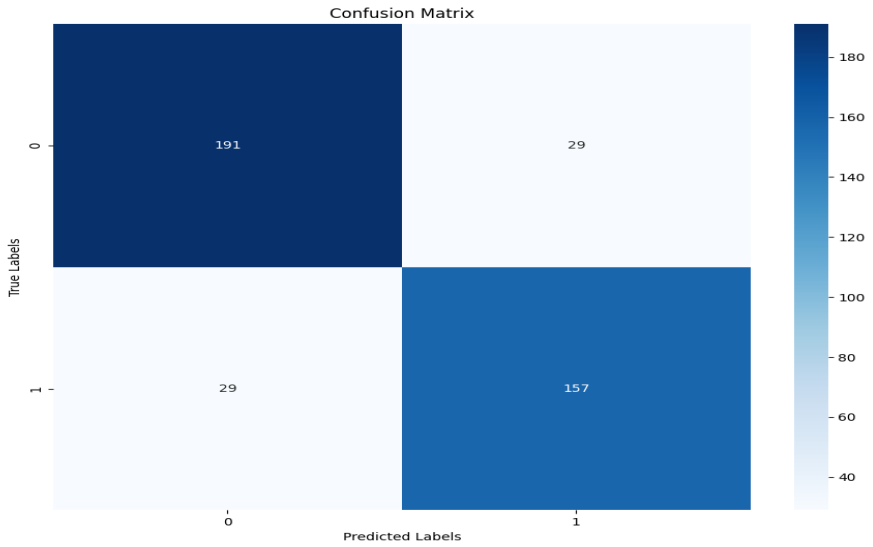

Figure 4.15 – Decision Tree Confusion Matrix

From the confusion matrix, we observe that the decision tree model correctly predicted 191 instances as positive and 157 instances as negative. However, it only had 29 false positive predictions and 29 false negative predictions, so the performance is excellent.

3.9. Naïve-Bayes Model:

3.9.1. Code:

```
# Create the Naive Bayes classifier
classifier = GaussianNB()

# Perform k-fold cross-validation
cv_scores = cross_val_score(classifier, X_train, y_train, cv=5)

classifier.fit(X_train, y_train)
# Predict on the test set
y_pred = classifier.predict(X_test)

# Calculate the accuracy
test_accuracy = accuracy_score(y_test, y_pred)

# Get the classification report
NB_cls_rep = classification_report(y_test, y_pred)
```

3.9.2. Model Description:

The Naive-Bayes model is a simple and interpretable probabilistic algorithm that is widely recognized for its effectiveness. It is built upon the principles of Bayes' theorem and operates under the assumption of feature independence, hence its "naive" designation. By computing the probabilities of different class labels given the input features, the model selects the label with the highest probability as the final prediction. Despite its simplistic assumption, the Naive-Bayes model showcases strong performance across diverse domains and excels particularly well in handling high-dimensional data. It offers a transparent understanding of how the features influence the prediction process and can be easily interpreted, making it a highly valuable tool in numerous applications.

3.9.3. Evaluation:

Classification Report:

	precision	recall	f1-score	support
0.0	0.82	0.87	0.85	220
1.0	0.83	0.78	0.81	186
accuracy			0.83	406
macro avg	0.83	0.82	0.83	406
weighted avg	0.83	0.83	0.83	406

Figure 4.16 – Naïve Bayes Evaluation

The performance of the Naive-Bayes model on the test set is impressive, achieving an accuracy of **83**%, an F1 score of **81**%

3.9.4. Confusion Matrix:

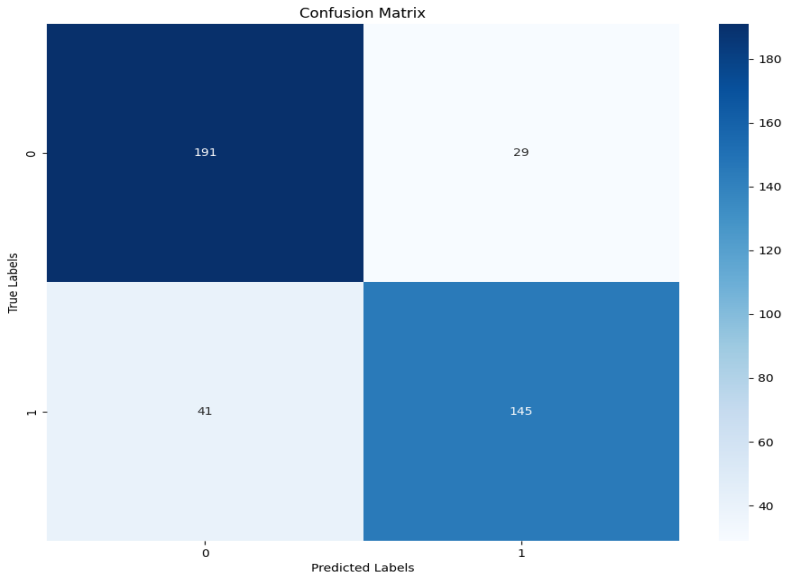

Figure 4.17 – Naïve Bayes Confusion Matrix

From the confusion matrix, we observe that the decision tree model correctly predicted 191 instances as positive and 145 instances as negative. However, it had 29 false positive predictions and 41 false negative predictions; this indicates that the model is performing well.

3.10. Neural Networks Model:

3.10.1. Code:

```
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Define the classifier (MLP)
classifier = MLPClassifier(max_iter=5000,
learning_rate='adaptive')

# Define the parameter grid for grid search
param_grid = {'hidden_layer_sizes': [(50,), (100,), (50,50)],
'activation': ['relu', 'tanh'], 'solver': ['adam', 'sgd'],
'tol': [1e-4]}

# Create K-Fold cross-validation object
kfold = KFold(n_splits=5, shuffle=True, random_state=42)
# Perform grid search to find the best hyperparameters
grid_search = GridSearchCV(classifier, param_grid, cv=kfold)

grid_search.fit(X_train_scaled, y_train)
best_params = grid_search.best_params_

# Create MLP classifier with the best hyperparameters
best_classifier = MLPClassifier(max_iter=5000,
learning_rate='adaptive', **best_params)

# Perform cross-validation on the training set
cv_results = cross_val_score(best_classifier, X_train_scaled,
y_train, cv=kfold)
accuracy_scores = cv_results.mean()
best_classifier.fit(X_train_scaled, y_train)
y_pred = best_classifier.predict(X_test_scaled)
# Calculate the accuracy
test_accuracy = accuracy_score(y_test, y_pred)
# Get the classification report
NN_cls_rep = classification_report(y_test, y_pred)
```

3.10.2. Model Description:

Neural Networks are sophisticated and powerful models recognized for their effectiveness in various domains. They are constructed based on interconnected layers of artificial neurons, which collectively process and learn from data through a complex network architecture. Unlike simpler models, they can capture detailed relationships within the data without relying on assumptions like feature independence. By leveraging a combination of activation functions, weights, and biases.

Neural Networks can learn intricate patterns and make accurate predictions. They excel in handling high-dimensional data, allowing them to effectively extract meaningful features and uncover complex underlying structures. They are notable for their applications in image recognition, natural language processing, and time-series forecasting. While their internal workings can be complex and less interpretable compared to simpler models, Neural Networks remain highly valued and widely used for their flexibility, adaptability, and potential for advanced modeling and prediction tasks.

3.10.3. Evaluation:

Classification Report:

	precision	recall	f1-score	support
0.0	0.85	0.85	0.85	220
1.0	0.83	0.82	0.82	186
accuracy			0.84	406
macro avg	0.84	0.84	0.84	406
weighted avg	0.84	0.84	0.84	406

Figure 4.18 – Neural Networks Evaluation

The Neural-Networks model demonstrates a very good performance on the test set, achieving an accuracy of **84**%, an F1 score of **82**%.

3.10.4. Confusion Matrix:

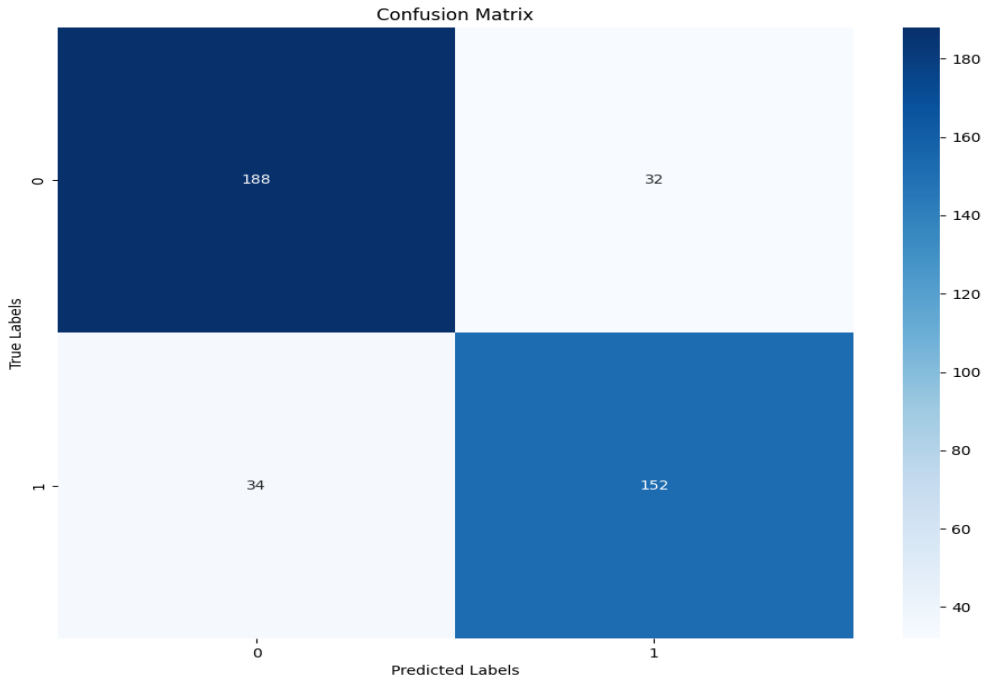

Figure 4.19 – Neural Networks Confusion Matrix

From the confusion matrix, we observe that the decision tree model correctly predicted 188 instances as positive and 152 instances as negative, it had 32 false positive predictions and 34 false negative predictions; which means the overall performance is very good.

4. Conclusion:

In conclusion, this dissertation has explored the development and evaluation of a predictive maintenance tool using Python and various machine learning algorithms, including K-Nearest Neighbours (KNN), Decision Tree, Naive-Bayes, and Neural Network,.

The use of Python as the primary programming language allowed for efficient and flexible implementation of the predictive maintenance tool. Its extensive libraries and frameworks facilitated data pre-processing, feature engineering, model training, and evaluation, enabling seamless integration of different machine learning algorithms.

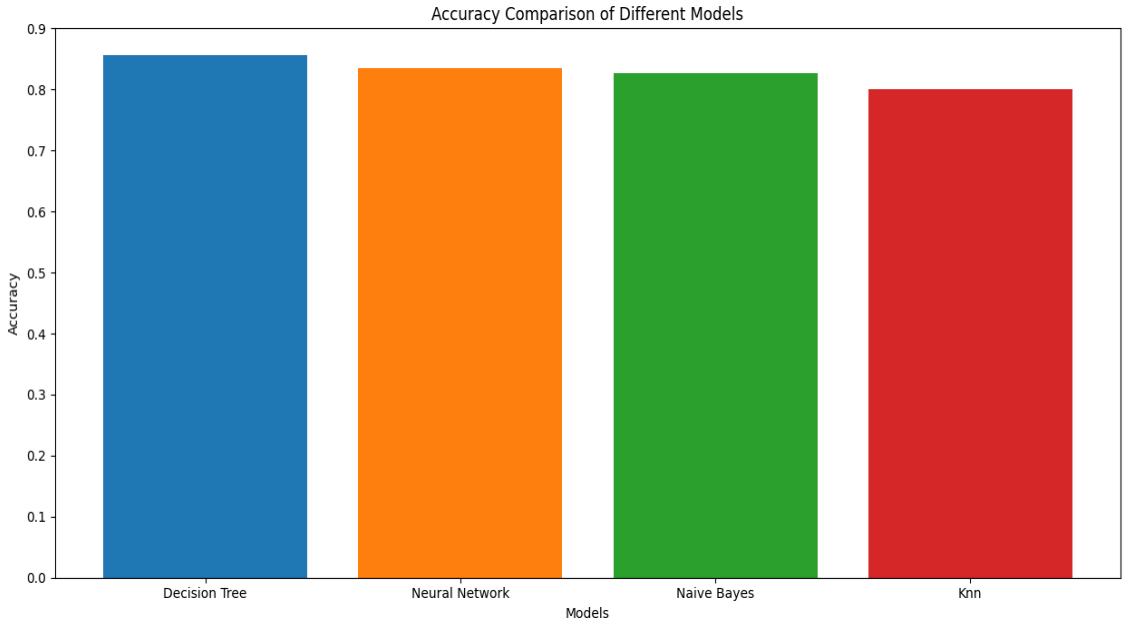

Figure 4.20 – Comparison of Model Performance

Throughout the research, different machine learning algorithms were applied and evaluated. The Decision Tree algorithm achieved an impressive 86% accuracy, which is the highest, shortly followed by the Neural Networks Algorithm, which achieved 84%, and then the Naïve Bayes Algorithm with an accuracy of 83%, and lastly, the K-NN Algorithm with 80%

The evaluation of these algorithms highlighted their respective strengths and limitations in the context of predictive maintenance. Each algorithm contributed valuable insights and

demonstrated its effectiveness in specific scenarios, emphasizing the importance of selecting the appropriate algorithm based on the nature of the problem and available data.

At last, the research displayed the significance of data pre-processing and feature engineering in improving the performance of predictive maintenance models. The careful selection and engineering of relevant features significantly enhanced the accuracy and robustness of the developed tool.

GENERAL CONCLUSION

In conclusion, this master's dissertation has successfully addressed the objective of developing a predictive maintenance tool for ETUSM, by leveraging advanced analytics techniques and machine learning algorithms. The research involved comprehensive data collection and analysis, leading to the identification of potential failures and degradation patterns in the bus fleet.

The study has made significant contributions to enhancing maintenance practices within ETUSM. By adopting proactive maintenance strategies enabled by the developed tool, the company can minimize unexpected downtime, optimize maintenance costs, and improve the reliability and availability of their buses. These outcomes directly benefit both ETUSM and its customers, ensuring a more efficient and reliable transportation service.

Among the various machine learning algorithms employed, Decision Trees demonstrated exceptional performance in predicting potential failures in the bus fleet. The flexibility and interpretability of Decision Trees make them highly suitable for this application. The model's ability to capture complex patterns and its robust performance makes it the preferred choice for implementing predictive maintenance in ETUSM's operations.

While Decision Trees proved to be the most effective algorithm for this specific predictive maintenance tool, it is important to acknowledge that further research and exploration of other machine learning algorithms can provide valuable insights and potentially enhance the tool's performance.

Overall, the development of this predictive maintenance tool marks a significant advancement in the transportation industry, promoting proactive maintenance strategies and improving the overall efficiency and reliability of ETUSM's bus fleet. The findings of this research lay the foundation for continued innovation in predictive maintenance practices and hold the potential to develop the maintenance operations of bus companies in Algeria.

REFERENCES

[1] "Maintenance: Definition, Types, Applications, and Pros & Cons." Retrieved from https://studentlesson.com/maintenance-definition-types-applications-pros-cons/. Accessed on: February 18th, 2023.

[2] "Industrial Maintenance at the Cusp of Industry Transformation." Retrieved from https://www.industr.com/en/industrial-maintenance-at-the-cusp-of-industry-transformation-2637064#:~:text=The%20broad%20industries%20that%20heavily,automotive%20industry%20and%20construction%20industry. Accessed on: February 21st, 2023.

[3] "Maintenance." Retrieved from https://safetyculture.com/topics/maintenance/. Accessed on: February 21st, 2023.

[4] "Corrective Maintenance." Retrieved from https://www.fiixsoftware.com/glossary/corrective-maintenance/. Accessed on: February 25th, 2023.

[5] "The Importance and Benefits of Predictive and Preventive Maintenance." Retrieved from https://www.tmasystems.com/resources/the-importance-and-benefits-of-predictive-and-preventive-maintenance/. Accessed on: February 27th, 2023.

[6] Mathias Wärja. (2005). Maintenance Management of Complex Industrial Systems - a methodology for renewal strategies (Doctoral thesis, Industrial Information and Control Systems Department of Electrical Engineering KTH, Royal Institute of Technology Stockholm, SWEDEN). Page 17, Retrieved from https://www.diva-portal.org/smash/get/diva2:14364/FULLTEXT01.pdf

[7] "Prediction." Retrieved from https://h2o.ai/wiki/prediction/. Accessed on: March 3rd, 2023.

[8] "Classification and Prediction in Data Mining." Retrieved from https://www.javatpoint.com/classification-and-predication-in-data-mining. Accessed on: March 4th, 2023.

[9] "Most Common Types of Machine Learning Problems." Retrieved from https://vitalflux.com/most-common-types-machine-learning-problems/. Accessed on: March 7th, 2023.

[10] "Machine Learning." Retrieved from https://monkeylearn.com/machine-learning/#:~:text=Machine%20learning%20(ML)%20is%20a,to%20make%20their%20own%20predictions. Accessed on: March 8th, 2023.

[11] Said Gadri, "Introduction to Artificial Intelligence "Chapter 5: Apprentissage Automatique et Apprentissage Profound, 2020-2021.

[12] "Role of Prediction Techniques in Various Fields." Retrieved from https://www.researchgate.net/publication/364929861_ROLE_OF_PREDICTION_TECHNIQUES_IN_VARIOUS_FIELDS. Accessed on: March 24th, 2023.

[13] "Predictive Analytics: A Review of Trends and Techniques." Retrieved from https://www.researchgate.net/publication/326435728_Predictive_Analytics_A_Review_of_Trends_and_Techniques. Accessed on: March 24th, 2023.

[14] "Natural Language Processing Techniques." Retrieved from https://monkeylearn.com/blog/natural-language-processing-techniques/. Accessed on: March 25th, 2023.

[15] "Everything You Need to Know About Linear Regression." Retrieved from https://www.analyticsvidhya.com/blog/2021/10/everything-you-need-to-know-about-linear-regression/. Accessed on: April 5th, 2023.

[16] "Understanding Polynomial Regression Model." Retrieved from https://www.analyticsvidhya.com/blog/2021/10/understanding-polynomial-regression-model/#:~:text=Polynomial%20regression%2C%20abbreviated%20E(y,the%20variance%20of%20the%20coefficients. Accessed on: April 5th, 2023.

[17] "All You Need to Know About Polynomial Regression." Retrieved from https://www.analyticsvidhya.com/blog/2021/07/all-you-need-to-know-about-polynomial-regression/#:~:text=A%20polynomial%20regression%20model%20is,the%20complexity%20of%20the%20relationship. Accessed on: April 5th, 2023.

[18] "Logistic Regression." Retrieved from https://aws.amazon.com/what-is/logistic-regression/#:~:text=In%20general%2C%20logistic%20regression%20explores,it%20has%20a%20known%20value. Accessed on: April 16th, 2023.

[19] "Classification in Machine Learning." Retrieved from https://www.datacamp.com/blog/classification-machine-learning. Accessed on: April 21st, 2023.

[20] "Neural Networks." Retrieved from https://www.ibm.com/topics/neural-networks. Accessed on: April 25th, 2023.

[21] "K-Nearest Neighbors (KNN)." Retrieved from https://www.ibm.com/topics/knn. Accessed on: May 8th, 2023.

[22] Towards Data Science. "Support Vector Machine: Introduction to Machine Learning Algorithms." Retrieved from https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47. Accessed on: May 9th, 2023.

[23] "Naive Bayes." Retrieved from https://www.ibm.com/topics/naive-bayes. Accessed on: May 11th, 2023.

[24] "Predictive Maintenance: What Is It and How Does It Work?" Retrieved from https://blog.infraspeak.com/predictive-maintenance/. Accessed on: May 14th, 2023.

[25] Özlem GÜVEN , 2Hasan ŞAHİN , 2022 , Predictive Maintenance Based On Machine Learning In Public Transportation Vehicles , Journal of Engineering Sciences and Researches, Vol 4 (1), Page 89 – 98 , https://dergipark.org.tr/en/pub/bjesr/issue/69536/1093519.

[26] Rune Prytz, 2014, Machine learning methods for vehicle predictive maintenance using off-board and on-board data, [Licentiate Thesis, Halmstad University Dissertations no. 9], Halmstad University, https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A789498&dswid=8005

[27] Prytz, Rune & Nowaczyk, Slawomir & Byttner, Stefan. (2011). Towards relation discovery for diagnostics, https://www.researchgate.net/publication/236669839_Towards_relation_discovery_for_diagnostics

[28] Prytz, Rune & Nowaczyk, Slawomir & Rognvaldsson, Thorsteinn & Byttner, Stefan. (2013). Analysis of Truck Compressor Failures Based on Logged Vehicle Data, https://www.researchgate.net/publication/256486984_Analysis_of_Truck_Compressor_Failures_Based_on_Logged_Vehicle_Data.

[29] Prytz, Rune & Nowaczyk, Slawomir & Rognvaldsson, Thorsteinn & Byttner, Stefan. (2015). Predicting the need for vehicle compressor repairs using maintenance records and logged vehicle data, https://www.researchgate.net/publication/273390872_Predicting_the_need_for_vehicle_compressor_repairs_using_maintenance_records_and_logged_vehicle_data

[30] **https://www.python.org/doc/essays/blurb/**

[31] **https://www.tutorialspoint.com/scikit_learn/scikit_learn_introduction.htm**

Abstract:

This master's dissertation focuses on developing a predictive maintenance tool for ETUSM, a bus company. The research involves data collection and analysis using advanced analytics techniques and machine learning algorithms to forecast potential failures in the bus fleet. The study contributes to enhancing maintenance practices, minimizing downtime, optimizing costs, and improving the reliability and availability of the buses. The developed tool serves as a significant step towards proactive maintenance strategies in the transportation industry, benefiting both ETUSM and its customers.

الملخص:

تركز هذه المذكرة على تطوير أداة صيانة تنبؤية لشركة النقل الحضري و الشبه حضري بولاية المسيلة ، و تضمن البحث جمع البيانات وتحليلها باستخدام تقنيات تحليل متقدمة وخوارزميات التعلم الآلي لتوقع الأعطال المحتملة في أسطول الحافلات ، تسهم هذا الدراسة في تعزيز ممارسات الصيانة، وتقليل أوقات التوقف، وتحسين تكاليف التشغيل، وتحسين الموثوقية وتوفر الحافلات، تعتبر الأداة المطورة خطوة هامة نحو استراتيجيات الصيانة الاستباقية في مجال النقل، وتعود بالفائدة على شركة النقل الحضري و الشبه حضري وزبائنها.