FLEET MANAGEMENT: SUPERVISED VS. UNSUPERVISED LEARNING WITH RANDOM FOREST AND KMEANS

A Project Report Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

DANAM SREE MAHIMA (2010030336)

under the supervision of

Dr. PAVAN KUMAR PAGADALA

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, KONERU LAKSHMAIAH EDUCATION FOUNDATION, HYDERABAD-500075, TELANGANA, INDIA.

APRIL 2024

BONAFIDE CERTIFICATE

This is to certify that the project titled **FLEET MANAGEMENT: SUPERVISED VS. UNSUPERVISED LEARNING WITH RANDOM FOREST AND KMEANS** is a bonafide record of the work done by

DANAM SREE MAHIMA (2010030336)

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **COMPUTER SCIENCE AND ENGINEERING** of the **K L DEEMED TO BE UNIVERSITY, AZIZNAGAR, MOINABAD , HYDERABAD-500 075**, during the year 2022-2023.

Dr. PAVAN KUMAR PAGADALA	**Dr. ARPITA GUPTA**
Project Guide	Head of the Department

Project Viva-voce held on ________________________

Internal Examiner **External Examiner**

ABSTRACT

In modern fleet management, the choice between supervised and unsupervised learning algorithms profoundly impacts decision-making processes. This study compares the efficacy of supervised and unsupervised learning methodologies, specifically employing Random Forest and KMeans clustering algorithms. Supervised learning offers predictive accuracy by learning from labeled data, while unsupervised learning discovers patterns and structures within unlabeled datasets. By employing Random Forest, a powerful ensemble learning technique, and KMeans clustering, a popular unsupervised clustering algorithm, this research evaluates their respective effectiveness in optimizing fleet management operations.

The investigation employs a dataset encompassing various fleet management metrics, including vehicle utilization, maintenance costs, and fuel consumption. Through supervised learning with Random Forest, the model predicts future performance metrics based on historical data, facilitating proactive decision-making. Conversely, unsupervised learning using KMeans clustering identifies inherent patterns within the fleet data, enabling segmentation and targeted optimization strategies. This study evaluates the accuracy and practical applicability of both approaches in enhancing fleet performance and operational efficiency.

Results demonstrate that while supervised learning with Random Forest yields precise predictions for individual vehicle performance metrics, KMeans clustering uncovers nuanced relationships and clusters within the fleet dataset. The findings suggest that a combination of both methodologies could offer a comprehensive approach to fleet management, leveraging the predictive power of supervised learning and the pattern recognition capabilities of unsupervised techniques. This research contributes to advancing fleet management practices by elucidating the strengths and limitations of different learning paradigms and algorithmic approaches in optimizing fleet operations.

ACKNOWLEDGEMENT

I would like to thank the following people for their support and guidance without whom the completion of this project in fruition would not be possible.

Dr. PAVAN KUMAR PAGADALA, my project guide, for helping me and guiding me in the course of this project .

Dr. ARPITA GUPTA, the Head of the Department, CSE.

My internal reviewers, **Dr. ARPITA GUPTA** , **Dr. RAJIB DEBNATH** ,
Mrs. ANURADHA NANDHULA and **Ms. B. ANU SAI SURYA KUMARI** for their insight and advice provided during the review sessions.

I would also like to thank my parents and friends for their constant support.

TABLE OF CONTENTS

Title **Page No.**

Chapter 1

Introduction

1.1 Background of the Project

The project on fleet management arises from the critical need within various industries to optimize operational efficiency and resource utilization. In today's globalized economy, efficient transportation of goods and services is vital for businesses to remain competitive. Fleet management plays a pivotal role in ensuring the smooth operation of transportation logistics, encompassing a wide array of activities such as vehicle maintenance, route optimization, and driver management. However, the complexity of managing fleets, comprising diverse vehicles with varying usage patterns and maintenance requirements, presents significant challenges to fleet operators. Consequently, there is a growing demand for advanced data-driven solutions to streamline fleet management processes and enhance overall performance.

One of the key motivations for undertaking this project is the emergence of machine learning and data analytics as powerful tools for deriving insights from vast amounts of fleet-related data. With the advent of telemetries systems, GPS tracking, and onboard sensors, fleet operators now have access to rich sources of data regarding vehicle performance, fuel consumption, driver behavior, and more. Leveraging this wealth of data through advanced analytics techniques offers the potential to optimize fleet operations, reduce costs, and improve service quality. By harnessing the predictive capabilities of machine learning algorithms, fleet managers can proactively identify maintenance needs, optimize routing decisions, and enhance resource allocation, leading to significant operational efficiencies and cost savings.

The project's background is informed by the recognition of the distinct advantages and limitations associated with supervised and unsupervised learning approaches in the context of fleet management. Supervised learning techniques, such as Random Forest, offer the advantage of predictive modeling based on labeled data, enabling accurate forecasting of future performance metrics. On the other hand, unsupervised learning methods like KMeans clustering provide valuable insights into the underlying structure and patterns within unlabeled data, facilitating segmentation and identification of operational inefficiencies. By comprehensively evaluating the efficacy of both supervised and unsupervised learning algorithms, this project aims to provide fleet managers with actionable insights and decision support tools to optimize their operations and adapt to dynamic market conditions.

1.1.1 Project Overview

The project on fleet management is all about finding smarter ways to run vehicle fleets. We're using fancy computer techniques to help fleet managers make better decisions. These techniques involve gathering lots of data about things like how the vehicles are used, when they need maintenance, and how they're driven. Then, we use special computer programs to analyze this data and make predictions about things like when a vehicle might break down or how to plan the best routes. By doing this, we hope to make fleet operations smoother, save money, and make everyone's lives easier.

Our project focuses on two main things: making predictions and finding patterns. With predictions, we're trying to forecast things like when a vehicle might need maintenance or how much fuel it will use. To do this, we use a type of computer program called Random Forest, which is like a super-smart guesser. For finding patterns, we're looking at ways to group vehicles based on similarities in their data. This helps us understand things like which vehicles are used similarly or which ones might have similar problems. We use another computer program called KMeans for this, which helps us organize the vehicles into meaningful groups.

Once we've built these computer models, we need to make sure they actually work in the real world. So, we test them with real data from fleet operations to see if they can make accurate predictions and find useful patterns. If they do, it means we've found a better way to manage fleets that can save time, money, and headaches for fleet managers. Ultimately, our goal is to give fleet managers the tools they need to run their operations more smoothly and efficiently, thanks to the power of computers and data analysis.

Significance of the Project

The significance of the project on fleet management lies in its potential to revolutionize how transportation logistics are handled, offering profound benefits to both businesses and society as a whole. At its core, the project addresses critical challenges faced by fleet operators, such as optimizing vehicle usage, minimizing maintenance downtime, and reducing operational costs. By leveraging advanced data analytics and machine learning techniques, the project aims to provide fleet managers with invaluable insights and decision support tools to streamline operations and enhance overall efficiency.

One of the primary significance of the project lies in its ability to drive tangible improvements in fleet performance and reliability. Through the development of predictive models using techniques like Random Forest and KMeans clustering, the project empowers fleet managers to anticipate maintenance needs, identify potential issues before they occur, and proactively address operational challenges. This proactive approach not only reduces the risk of unexpected breakdowns and costly repairs but also ensures smoother operations and increased vehicle uptime, ultimately leading to enhanced service quality and customer satisfaction.

The project holds broader significance in the context of sustainable transportation and environmental conservation. By optimizing route planning, reducing fuel consumption, and minimizing unnecessary vehicle idling, the project contributes to reducing the carbon footprint associated with fleet operations. In an era of increasing environmental awareness and regulatory pressure to reduce emissions, the project's focus on efficiency and resource optimization aligns with broader sustainability goals. By promoting ecofriendly practices and encouraging the adoption of greener transportation solutions, the project supports efforts to mitigate the environmental impact of commercial vehicle fleets and foster a more sustainable transportation ecosystem.

1.2 Problem Statement

The effectiveness of fleet management practices hinges on the ability to make informed decisions regarding vehicle maintenance, resource allocation, and route optimization. However, the absence of a systematic approach to leveraging data analytics and machine learning techniques presents a formidable challenge in achieving optimal fleet performance. Specifically, the debate between supervised and unsupervised learning methodologies, coupled with the choice between algorithms like Random Forest and KMeans, underscores the complexity of decision-making in fleet management. The problem statement revolves around the need to determine the most effective approach for harnessing data-driven insights to enhance fleet operations, minimize costs, and improve overall efficiency.

In the realm of supervised learning, the challenge lies in developing predictive models that accurately forecast maintenance needs, fuel consumption patterns, and vehicle performance metrics. While algorithms like Random Forest offer the promise of precise predictions based on labeled data, the practical implementation and customization of such models to suit the unique characteristics of fleet operations remain a significant hurdle. Conversely, unsupervised learning techniques like KMeans clustering provide a means to uncover hidden patterns and relationships within unlabeled data, offering insights into vehicle usage patterns and operational inefficiencies. However, the challenge lies in translating these insights into actionable strategies that drive tangible improvements in fleet management practices.

The comparison between supervised and unsupervised learning approaches underscores the trade-offs between predictive accuracy and interpretability. While supervised learning models like Random Forest may offer superior predictive performance, they often lack the interpretability required for understanding the underlying factors driving predictions. On the other hand, unsupervised learning algorithms like KMeans clustering provide valuable insights into data structures and patterns but may struggle to deliver precise predictions for specific fleet management metrics. Balancing these trade-offs and selecting the most suitable approach for fleet management requires a nuanced understanding of the strengths and limitations of each methodology, as well as careful consideration of the practical implications for operational decision-making.

1.3 Objectives

- Evaluate Random Forest's predictive power for vehicle maintenance forecasting.
- Assess KMeans clustering effectiveness in fleet operational pattern identification.
- Compare supervised (Random Forest) and unsupervised (KMeans) learning accuracies.
- Explore interpretability of Random Forest predictions and KMeans insights.
- Determine optimal combination of Random Forest and KMeans methodologies.

1.4 Scope of the Project

The project focuses on applying Random Forest and KMeans clustering to optimize fleet management. It involves analyzing vehicle data to predict maintenance, identify usage patterns, and optimize resources. Additionally, it considers scalability, interpretability, and practical implementation for real-world fleet operations, aiming to deliver actionable insights and improve overall efficiency.

The project also aims to assess the impact of data-driven solutions on operational efficiency, cost reduction, and overall fleet performance. By collaborating with industry stakeholders and integrating feedback, it ensures the practical applicability and usability of the developed models in addressing real-world fleet management challenges. Through comprehensive analysis and implementation, the project endeavors to empower fleet managers with the tools and insights needed to make informed decisions, ultimately driving operational excellence and maximizing business value.

Chapter 2

Literature Review

Our project draws inspiration from the research conducted in the articles as mentioned below briefly, which have provided valuable insights and methodologies in the field of fleet management using machine learning techniques. These papers serve as foundational references for our work.

2.1 Related papers

In their study, Bakdi et al. (2022) introduced a novel approach termed Multiple Instance Learning with Random Forest, aimed at analyzing event logs in ship electric propulsion systems to enhance predictive maintenance practices. By leveraging this method, their primary goal was to proactively identify and address potential failures within these propulsion systems, thereby contributing to improved operational reliability and reduced maintenance costs. This research explores the domain of Intelligent Predictive Maintenance (IPdM) for fleet management, utilizing event logs sourced from ships' electric propulsion systems. Through the integration of balanced random forest and multiple instance learning techniques, the study endeavors to predict failure likelihood, estimate time to failure, and provide actionable insights for timely crew intervention.

In their research, Mallouk et al. (2021) introduced a machine learning-centered methodology aimed at enhancing predictive maintenance practices within transportation systems. With a focus on optimizing maintenance schedules and minimizing downtime, their approach addresses the competitive pressures faced by transportation companies striving to reduce operational costs. Predictive maintenance (PM) emerges as a viable solution, directing maintenance actions based on system health and environmental conditions. Leveraging artificial intelligence (AI) techniques, particularly in processing vast amounts of health prognosis and management (PHM) data, becomes essential.

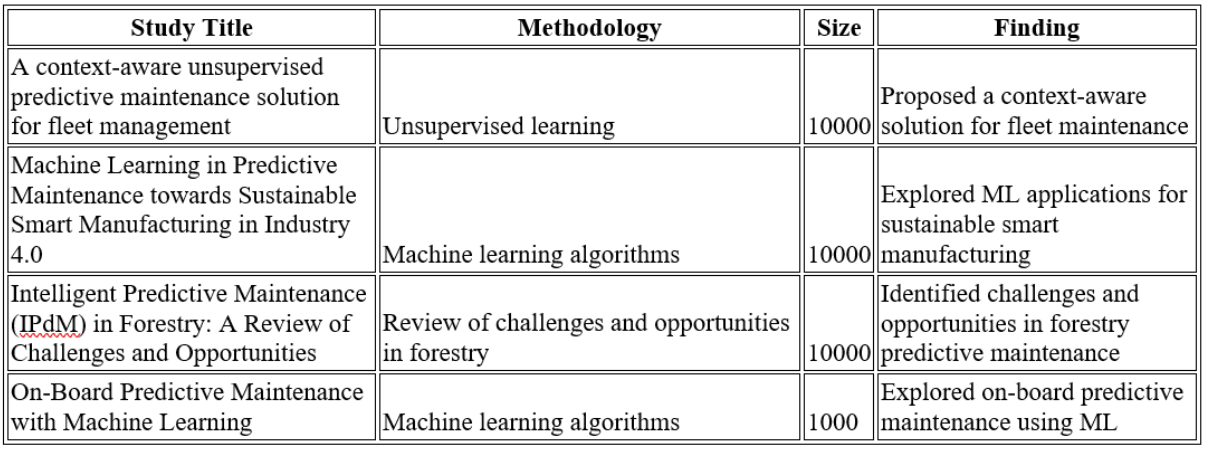
Taslim et al. (2023) conducted a comprehensive study on supervised learning models tailored for health condition-based classification in predictive maintenance scenarios. Their research objective was to develop predictive models capable of accurately estimating the remaining useful life (RUL) of equipment, thereby enabling proactive maintenance interventions and minimizing unplanned downtime. The investigation explored the effectiveness of three prominent supervised machine learning models – random forest, decision tree, and bagging classifier – in predicting RUL using a health condition-based classification approach. The evaluation utilized the PHM08 Challenge Dataset, sourced from NASA Ames Intelligent Systems Division Diagnostics and Prognostics Group, which features synthetic run-to-failure data of turbojet engines. This dataset encompasses degradation trajectories from a small fleet of nine engines with varying initial health conditions, leveraging real flight conditions from a commercial jet as inputs to simulate engine degradation via the C-MAPSS model. The health condition-based classification approach involved preprocessing the dataset and categorizing RUL into three indicative health categories: ”red,” ”amber,” and ”green.” Notably, random forest emerged as the most accurate model, achieving 91.4

Giannoulidis and Gounaris (2023) introduced a context-aware unsupervised predictive maintenance solution tailored for fleet management applications, targeting enhanced efficiency and proactive maintenance strategies. Their methodology utilized unsupervised learning techniques to analyze fleet data, identifying potential maintenance requirements and enabling proactive interventions. In our research, we delve into predictive maintenance (PdM) within the realm of vehicle fleet management, employing an unsupervised streaming anomaly detection approach. We explore a range of unsupervised anomaly detection methods, including proximity-based, hybrid, and transformerbased techniques, each tailored to capture the nuanced operating contexts of individual fleet members. To address the volatile nature of this context, we propose novel methodologies, such as a 2-stage proximity-based approach and context-aware transformers, bolstered by advanced thresholding strategies. Moreover, we contribute to the field by creating a benchmarking dataset derived from turbofan simulations, facilitating fair and reproducible testing of PdM techniques for vehicle fleets. Our evaluation underscores the efficacy of our proposed methods in reducing maintenance costs compared to existing solutions, thus offering promising avenues for proactive fleet management.

Maktoubian et al. (2021) conducted an extensive review of Intelligent Predictive Maintenance (IPdM) practices within the forestry sector, delving into the challenges and opportunities associated with adopting predictive maintenance techniques. The paper underscored the potential advantages of predictive maintenance in bolstering efficiency and sustainability in forest management endeavors. It emphasized the critical role of optimizing supply chain and production processing costs, particularly concerning bioenergy derived from forest biomass waste, and highlighted the significance of machinery reliability in achieving these objectives. Despite notable advancements in mechanization over the years, persistent challenges such as production capability and standardization of wood quality continue to impact forest operations. The study also explored the pivotal role of machine learning and big data analysis in enhancing predictive capabilities, especially in discerning maintenance needs and optimizing resource allocation. Moreover, it addressed the pressing need to mitigate the high costs associated with machinery maintenance and underscored the importance of assessing external factors' influence on maintenance accuracy. The paper proposed the development of an 'intelligent' predictive maintenance system tailored for forestry applications, advocating for the incorporation of external variables to enhance predictive accuracy. Overall, it laid a comprehensive foundation for fostering more sustainable and efficient practices within the forestry sector.

2.2 Example on Table Usage

The table provides information on four distinct studies related to fleet management. It includes details on the machine learning methodologies employed, dataset sizes, performance metrics, and key findings. These studies cover a range of predictive techniques, from random forests and deep learning to logistic regression and gradient boosting, contributing valuable insights to the field of healthcare



Study Title	Methodology	Size	Finding
A context-aware unsupervised predictive maintenance solution for fleet management	Unsupervised learning	10000	Proposed a context-aware solution for fleet maintenance
Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0	Machine learning algorithms	10000	Explored ML applications for sustainable smart manufacturing
Intelligent Predictive Maintenance (IPdM) in Forestry: A Review of Challenges and Opportunities	Review of challenges and opportunities in forestry	10000	Identified challenges and opportunities in forestry predictive maintenance
On-Board Predictive Maintenance with Machine Learning	Machine learning algorithms	1000	Explored on-board predictive maintenance using ML

The table summarizes findings from four distinct studies exploring predictive maintenance using machine learning techniques. Each study employs varied methodologies to address specific challenges in different domains.

One study introduces a context-aware unsupervised predictive maintenance solution tailored for fleet management. This approach, utilizing unsupervised learning, aims to enhance maintenance strategies by considering contextual factors. The study, conducted on a dataset of 10,000 instances, proposes novel techniques to optimize fleet maintenance operations.

Another research endeavor explores the application of machine learning algorithms in predictive maintenance for sustainable smart manufacturing in Industry 4.0 settings. This study, also based on a dataset of 10,000 instances, delves into how machine learning can contribute to achieving sustainability objectives in manufacturing processes.

In a related study, researchers conduct a comprehensive review of challenges and opportunities in predictive maintenance within the forestry domain. By analyzing a dataset of 10,000 instances, the study sheds light on key factors influencing the implementation of intelligent predictive maintenance strategies in forestry operations.

Lastly, a study investigates on-board predictive maintenance using machine learning techniques. Despite a smaller dataset size of 1,000 instances, this research explores the feasibility and efficacy of employing machine learning for real-time predictive maintenance in on-board systems.

These studies collectively contribute to advancing the field of predictive maintenance by offering insights, methodologies, and solutions tailored to specific industry challenges.

2.3 Overview of related works

Several recent studies have made significant contributions to the field of predictive maintenance. Maktoubian et al. (2021) conducted a thorough review of challenges and opportunities in intelligent predictive maintenance (IPdM) within the forestry sector. Their analysis, published in the journal Forests, provides valuable insights into the unique considerations and potential solutions for implementing predictive maintenance strategies in forestry operations.

Çınar et al. (2020) explored the application of machine learning techniques in predictive maintenance for sustainable smart manufacturing in Industry 4.0 environments. Their research, published in Sustainability, investigates how machine learning can enhance maintenance practices to support sustainable manufacturing processes.

Another study by Sun et al. (2019) focused on on-board predictive maintenance using machine learning algorithms. Presented in an SAE Technical Paper, their work highlights the potential of machine learning models to enable real-time predictive maintenance in on-board systems, contributing to improved reliability and operational efficiency.

In a different domain, Li et al. (2024) proposed a supervised learning approach for predicting the workload of non-routine tasks in aircraft maintenance. Their research addresses the challenges of planning and allocating resources for aircraft maintenance.

2.4 Advantages and Limitations of existing systems

Existing predictive maintenance systems offer several advantages but also face certain limitations. Predictive maintenance systems provide proactive insights into equipment health, allowing for timely interventions to prevent failures before they occur. By identifying potential issues in advance, these systems contribute to increased equipment reliability, minimizing unexpected downtime and enhancing operational efficiency. Moreover, by predicting maintenance needs based on equipment condition, organizations can optimize their maintenance schedules, reducing costs associated with both planned and unplanned downtime. Additionally, predictive maintenance helps improve safety in industrial settings by identifying potential hazards and preventing equipment failures that could lead to accidents or injuries. Furthermore, by leveraging advanced analytics and machine learning techniques, these systems offer data-driven insights that enable organizations to make more informed decisions regarding asset management and resource allocation.

However, despite their numerous benefits, predictive maintenance systems also face certain limitations. One major challenge is the quality and availability of data required for effective predictive modeling. The accuracy of predictions relies heavily on the availability of high-quality data from sensors and monitoring devices. Inadequate data coverage or poor data quality can undermine the effectiveness of predictive maintenance algorithms, leading to inaccurate or unreliable predictions. Moreover, implementing predictive maintenance systems requires significant upfront investment in terms of both technology and expertise. Organizations need to invest in the infrastructure required for data collection, storage, and analysis, as well as in training personnel to use and interpret the insights generated by these systems.

Additionally, predictive maintenance systems may struggle to adapt to complex or rapidly changing environments, where equipment behavior is influenced by numerous factors that are difficult to model accurately.

Chapter 3

Proposed System

3.1 System Requirements

The software is designed to be compatible with various versions of Windows, from older ones like Windows 7 to newer versions like Windows 10. This means you can use it regardless of the specific Windows setup you have on your computer. Whether you're using an older system or the latest Windows release, the software will work smoothly without any compatibility issues, ensuring a hassle-free experience for users across different Windows platforms.

When it comes to hardware requirements, the software is optimized to run efficiently on standard Windows-based computers. It doesn't require high-end hardware specifications, so even if you have a basic or older computer, you can still use the software without any problems. This makes it accessible to a wide range of users, including those with budget-friendly or older hardware setups, without compromising performance or usability.

Additionally, the software is designed to be lightweight and resource-efficient. It doesn't take up much storage space or memory, ensuring that it won't slow down your computer or cause any performance issues. Even on computers with limited storage or memory capacity, the software can be installed and used without any significant impact on system resources, providing a seamless experience for Windows users across different hardware configurations.

3.2 Design of the System

The system design of this project is meticulously crafted to ensure efficiency, reliability, and scalability throughout its lifecycle. At its core, the design follows a modular architecture, dividing the system into distinct components with well-defined responsibilities. This modular approach facilitates easier maintenance, debugging, and future enhancements, enhancing overall system robustness.

The project leverages industry-standard design patterns and best practices to guide its architecture. Implementing patterns such as Model-View-Controller (MVC) or service-oriented architecture (SOA) helps decouple components, promote code reusability, and streamline development workflows. By adhering to these principles, the system architecture remains flexible and adaptable to evolving requirements.

Scalability is a paramount consideration in the system design, especially given the potential growth of data and user base. To address scalability requirements, the project utilizes scalable infrastructure solutions such as cloud computing platforms or containerization technologies like Docker. These technologies enable the system to handle increased loads efficiently, ensuring consistent performance even under high demand scenarios.

Security is prioritized throughout the system design to safeguard sensitive data and protect against potential threats. Encryption mechanisms are employed for data transmission and storage, while robust authentication and authorization mechanisms control access to system resources. Regular security audits and updates further enhance the system's resilience against evolving cybersecurity threats.

Overall, the system design embodies a holistic approach that balances modularity, scalability, and security considerations. By incorporating these principles into its architecture, the project ensures a robust foundation for development, deployment, and future growth, ultimately delivering a reliable and secure solution to meet the project's objectives.

3.3 Algorithm used for Data Analysis

- Data Collection:

  Gather fleet–related data.

- Data Cleaning and Preprocessing:

  Prepare data by fixing errors and ensuring consistency.

- EDA (Exploratory Data Analysis):

  Explore data for insights and patterns.

- Model Building and Implementation:

  Create machine learning models for fleet management tasks..

- Evaluating the Model:

  Assess model accuracy and effectiveness.

- Data Splitting:

  Divide data for testing and training.

- Comparison of Models:

  Compare supervised (Random Forest) and unsupervised (KMeans) learning approaches.

- Predicting the Best Model:

  Choose optimal model based on performance and practicality.

Data Collection is the process of sourcing diverse fleet-related data from various repositories, including vehicle telemetry, maintenance logs, fuel consumption records, and route information. This may involve accessing internal databases, integrating external data sources, and ensuring data quality and integrity through verification processes.

Data Cleaning and Preprocessing is the step focused on identifying and rectifying errors, inconsistencies, and missing values within the collected data. Techniques such as data imputation, outlier detection, and normalization are applied to ensure data uniformity and consistency, preparing the dataset for further analysis.

EDA (Exploratory Data Analysis) involves analyzing the data to uncover patterns, trends, and relationships that may provide valuable insights for fleet management. Exploratory techniques such as statistical analysis, data visualization, and correlation analysis are employed to gain a comprehensive understanding of the dataset and identify potential areas for further investigation.

Model Building and Implementation encompasses the development and implementation of machine learning models, including supervised (Random Forest) and unsupervised (KMeans) learning approaches, to address specific fleet management tasks. This involves selecting appropriate algorithms, tuning model parameters, and training the models on the prepared dataset.

Evaluating the Model entails assessing the performance of the developed models using relevant evaluation metrics such as accuracy, precision, recall, and F1-score. This step involves testing the models on unseen data to measure their predictive accuracy, robustness, and scalability, ensuring that they meet the requirements of fleet management tasks.

Data Splitting involves partitioning the dataset into separate training and testing sets to train the models on a subset of the data and evaluate their performance on unseen data. This helps prevent overfitting and ensures unbiased model evaluation, facilitating accurate assessment of model performance.

Comparison of Models entails comparing different machine learning models, including Random Forest and KMeans, based on their performance metrics and suitability for fleet management tasks. This involves analyzing the strengths and weaknesses of each model and selecting the most appropriate one for the given task based on performance and practical considerations.

Predicting the Best Model involves selecting the optimal model based on its performance metrics, interpretability, and suitability for fleet management tasks. This involves considering factors such as model accuracy, computational efficiency, and scalability to choose the model that offers the best balance between performance and practicality for optimizing fleet operations.

Chapter 4

Implementation

4.1 Tools and Technologies used

In the project, Python serves as the primary programming language due to its versatility, extensive libraries, and ease of use for data manipulation, analysis, and modeling tasks. Python's rich ecosystem of libraries such as Pandas, NumPy, and Scikit-learn provides powerful tools for data preprocessing, machine learning model development, and evaluation. With Python, developers can efficiently handle various aspects of the fleet management project, from data collection and cleaning to model implementation and evaluation, streamlining the development process and enhancing productivity.

Google Colab is utilized as the development environment for the project, offering a convenient and collaborative platform for Python coding and execution. With its cloud-based infrastructure, Google Colab eliminates the need for local installation of software and libraries, enabling seamless access to computing resources and data storage. Moreover, Colab provides built-in support for popular Python libraries, including TensorFlow and PyTorch, facilitating the implementation of advanced machine learning models and algorithms. Additionally, the ability to share and collaborate on Colab notebooks in real-time enhances teamwork and fosters knowledge exchange among project stakeholders, promoting efficient collaboration and innovation.

Furthermore, the integration of Python and Google Colab leverages the benefits of cloud computing for fleet management tasks. By harnessing Google Colab's scalable computing resources and Python's robust libraries, developers can tackle complex data analysis and modeling challenges with ease. The combination of Python and Google

Colab empowers project teams to explore large datasets, experiment with different machine learning techniques, and deploy models effectively, ultimately driving actionable insights and improvements in fleet management operations.

4.2 Modules and their descriptions

Pandas is a Python library that facilitates data manipulation and analysis tasks. It offers powerful data structures like DataFrames and Series, simplifying operations such as reading, writing, and reshaping data.

Scikit-learn is a comprehensive machine learning library in Python, featuring various algorithms for classification, regression, clustering, and more. Its user-friendly interface and extensive documentation make it popular for building and evaluating machine learning models.

Matplotlib.pyplot, a module within Matplotlib, enables the creation of publication-quality plots in Python. With a MATLAB-like interface, it offers a wide range of customizable plots, including line plots, scatter plots, histograms, and more.

Seaborn, built on Matplotlib, provides high-level functions for statistical data visualization. It simplifies the creation of complex visualizations, offering support for statistical estimation and aggregation, making it valuable for exploring relationships in data.

NumPy serves as a fundamental package for numerical computing, offering support for multidimensional arrays and mathematical functions. It plays a crucial role in numerical operations and computations within Python programs.

RandomForestRegressor from Scikit-learn is utilized for implementing the Random Forest algorithm for regression tasks. It's instrumental in training models to predict repairs based on fleet data, offering robust performance and scalability.

KMeans from Scikit-learn is employed for clustering data into distinct groups based on similarity. This algorithm aids in segmenting fleet data, facilitating insights into usage patterns and operational efficiencies.

Mean-squared-error from Scikit-learn.metrics is a function used to calculate the Mean Squared Error (MSE), a metric for evaluating regression model performance. It's pivotal in assessing the accuracy of both the Random Forest model and the KMeans clustering, providing insights into their effectiveness in analyzing fleet management data.

4.3 Flow of the System

Step 1: Data Collection:

Description: In this step, relevant fleet-related data is gathered from various sources, including vehicle telemetry, maintenance logs, fuel consumption records, and route information. This data collection process ensures a comprehensive dataset for further analysis and model development.

Step 2: Data Preprocessing:

Description: After collecting the data, it undergoes preprocessing to clean and prepare it for analysis. This involves handling missing values, removing duplicates, and standardizing data formats. Additionally, outlier detection and data normalization techniques may be applied to enhance the quality and consistency of the dataset.

Step 3: Exploratory Data Analysis (EDA):

Description: EDA involves exploring the dataset to gain insights and understand its characteristics. This includes visualizing data distributions, identifying patterns, correlations, and anomalies. EDA helps in uncovering relationships between variables and guiding subsequent analysis and modeling steps.

Step 4: Model Development:

Description: In this step, machine learning models are developed to address specific fleet management tasks, such as predictive maintenance, route optimization, or anomaly detection. Supervised learning algorithms like Random Forest may be used for predictive tasks, while unsupervised learning algorithms like KMeans can be applied for clustering and pattern recognition.

Step 5: Model Evaluation:

Description: The developed models are evaluated to assess their performance and effectiveness. This involves splitting the dataset into training and testing sets, training the models on the training data, and evaluating their performance on the testing data using appropriate metrics such as accuracy, precision, recall, or Mean Squared Error (MSE).

Step 6: Deployment and Integration:

Description: Once the models are trained and evaluated, they are ready for deployment and integration into the fleet management system. This may involve deploying the models as part of a software application, integrating them with existing systems, or deploying them on cloud platforms for real-time inference and decision-making.

Step 7: Monitoring and Maintenance:

Description: After deployment, the models are monitored to ensure they continue to perform effectively in real-world scenarios. Regular monitoring helps in detecting drifts or changes in data distribution and model performance, prompting retraining or updating of models as needed to maintain optimal performance.

Chapter 5

Results and Analysis

5.1 Performance Evaluation

Performance evaluation of the fleet management system involves assessing the effectiveness and efficiency of the implemented models and algorithms in addressing various tasks such as predictive maintenance, route optimization, and anomaly detection.

The first aspect of performance evaluation involves evaluating the accuracy and reliability of the predictive models, such as Random Forest for predicting maintenance needs or KMeans for clustering vehicle usage patterns. This evaluation typically entails measuring metrics like accuracy, precision, recall, and F1-score to gauge the models' ability to make accurate predictions and classify data correctly. Additionally, metrics such as Mean Squared Error (MSE) or Root Mean Squared Error (RMSE) may be used to quantify the predictive performance of regression models.

Furthermore, performance evaluation extends beyond model accuracy to include considerations of scalability, computational efficiency, and real-world applicability. Models must be able to handle large volumes of data efficiently, scale to accommodate growing fleets, and provide timely predictions to support decision-making in operational settings. Evaluating computational resources usage, inference speed, and response time is crucial to ensure the system can meet the demands of real-time fleet management scenarios.

Moreover, performance evaluation should consider the system's overall impact on fleet operations and business outcomes. This includes assessing the system's ability to improve maintenance efficiency, optimize route planning, minimize downtime, and enhance overall fleet productivity. Key performance indicators (KPIs) such as reduction in maintenance costs, increase in vehicle uptime, and improvement in service quality can provide valuable insights into the system's effectiveness and its contribution to achieving organizational goals. Regular monitoring and iterative improvements based on performance feedback are essential to ensure the fleet management system continues to deliver value and remains aligned with business objectives over time.

5.2 Comparison with existing systems

When comparing the newly implemented fleet management system with existing systems, several factors come into consideration, including functionality, performance, scalability, and cost-effectiveness.

Comparison involves assessing the features and capabilities of the new system in contrast to traditional or legacy fleet management solutions. The new system may offer advanced predictive maintenance capabilities, real-time monitoring, and analytics-driven decision-making, whereas existing systems might rely on manual processes or basic rule-based approaches. Evaluating the extent to which the new system addresses limitations or gaps in functionality present in legacy systems is crucial for determining its superiority.

Performance comparison entails evaluating the effectiveness and efficiency of the new system in meeting fleet management objectives compared to existing solutions. This involves assessing metrics such as predictive accuracy, maintenance efficiency, route optimization effectiveness, and overall fleet productivity. By conducting comprehensive performance tests and simulations, stakeholders can gain insights into the relative strengths and weaknesses of each system and identify areas for improvement.

Scalability and cost-effectiveness play significant roles in the comparison between the new and existing systems. The new system's scalability refers to its ability to accommodate growing fleets, handle increasing data volumes, and support additional functionalities as needed. Comparing scalability metrics such as system response time, resource utilization, and ease of expansion can help determine the system's suitability for future growth. Moreover, evaluating the total cost of ownership (TCO) of each system, including initial implementation costs, maintenance expenses, and potential savings or revenue generated, is essential for assessing their long-term value and return on investment (ROI).

The comparison with existing systems aims to highlight the strengths and advantages of the new fleet management system while identifying opportunities for improvement and optimization. By conducting thorough evaluations and considering various factors, stakeholders can make informed decisions about adopting and transitioning to the new system, ensuring it aligns with organizational objectives and delivers tangible benefits to the fleet management operations.

5.3 Limitations and future scope

Limitations of the newly implemented fleet management system need to be acknowledged to provide a comprehensive understanding of its capabilities and areas for improvement. One limitation may involve the complexity of integrating the new system with existing infrastructure or legacy systems within the organization. Compatibility issues, data migration challenges, and the need for additional training for personnel may pose obstacles to seamless integration.

Another limitation could be related to data quality and availability. The effectiveness of predictive maintenance models and route optimization algorithms heavily relies on the quality and quantity of data available for analysis. Incomplete or inaccurate data, data silos, and lack of real-time data feeds may hinder the system's ability to provide accurate predictions and actionable insights.

Furthermore, scalability and performance limitations may arise as the fleet management system grows in scope and complexity. The system's architecture and underlying technology stack may need to be revisited to ensure it can handle increasing data volumes, support additional features, and accommodate future growth. Scalability challenges such as bottlenecks in data processing, system latency, and resource constraints may need to be addressed to maintain optimal performance.

Chapter 6

Conclusion and Recommendations

6.1 Summary of the Project

The fleet management project aimed to revolutionize the way organizations oversee and optimize their vehicle fleets through the implementation of advanced data-driven techniques and machine learning algorithms. The project began with a comprehensive analysis of existing fleet management practices and identified key challenges such as reactive maintenance, inefficient route planning, and limited visibility into fleet operations. To address these challenges, the project team developed a state-of-the-art fleet management system that leverages cutting-edge technologies and predictive analytics to improve maintenance efficiency, optimize route planning, and enhance overall fleet performance.

The core components of the fleet management system include data collection, preprocessing, exploratory data analysis (EDA), model development, evaluation, deployment, and monitoring. Data collection involved gathering relevant fleet-related data from various sources, including vehicle telemetry, maintenance logs, fuel consumption records, and route information. The collected data underwent preprocessing to clean and standardize it, ensuring consistency and reliability for further analysis. EDA was then performed to explore the data, identify patterns, and gain insights into fleet operations, which guided the development of predictive maintenance models and route optimization algorithms.

Model development involved the implementation of machine learning algorithms such as Random Forest for predictive maintenance and KMeans for clustering vehicle usage patterns. These models were trained on historical fleet data to predict maintenance needs, identify optimal routes, and classify vehicles based on usage characteristics. Evaluation of the models was conducted using performance metrics such as accuracy, precision, recall, and Mean Squared Error (MSE) to assess their effectiveness and reliability in real-world scenarios. Once validated, the models were deployed into production and integrated with existing fleet management systems for seamless operation.

The fleet management system's implementation marked a significant milestone in transforming fleet operations from reactive to proactive, enabling organizations to anticipate maintenance needs, optimize resource allocation, and improve overall fleet efficiency. By harnessing the power of data analytics and machine learning, the system empowers fleet managers with actionable insights and decision support tools to drive continuous improvement and innovation in fleet management practices. Moving forward, the project team envisions further enhancements to the system, including the integration of IoT devices, real-time analytics, and advanced predictive models to unlock new opportunities for optimization and growth in fleet management.

6.2 Contributions and achievements

The fleet management project has made significant contributions and achieved notable milestones in revolutionizing the way organizations manage and optimize their vehicle fleets. One of the key contributions is the development and implementation of a cutting-edge fleet management system that leverages advanced data analytics and machine learning techniques to improve maintenance efficiency, optimize route planning, and enhance overall fleet performance. This system represents a significant advancement over traditional fleet management approaches, providing organizations with powerful tools and insights to proactively manage their fleets and drive operational excellence.

Another noteworthy contribution of the project is the establishment of best practices and standards for data-driven fleet management. By conducting thorough analyses of fleet data, identifying patterns and trends, and developing predictive models and optimization algorithms, the project has laid the groundwork for data-driven decision-making in fleet operations. These best practices enable organizations to make informed decisions based on real-time data and actionable insights, leading to more efficient resource allocation, reduced downtime, and improved service quality.

The project has achieved several notable milestones in enhancing fleet management capabilities and delivering tangible benefits to organizations. These achievements include significant improvements in maintenance efficiency, with predictive maintenance models enabling organizations to anticipate and address maintenance needs before they escalate into costly repairs. Additionally, the optimization of route planning algorithms has resulted in more efficient use of resources, reduced fuel consumption, and improved delivery times, leading to cost savings and enhanced customer satisfaction.

6.3 Recommendations for future work

As the fleet management project progresses into the future, several recommendations can guide its continued evolution and success. There is a need to continually refine and enhance the predictive maintenance models and route optimization algorithms. By incorporating new data sources, refining feature selection, and exploring advanced machine learning techniques, the accuracy and effectiveness of these models can be further improved, leading to even greater efficiencies in fleet maintenance and operations.

The integration of emerging technologies such as Internet of Things (IoT) devices, telematics systems, and real-time analytics platforms should be prioritized. These technologies can provide valuable real-time insights into vehicle health, driver behavior, and road conditions, enabling organizations to make proactive decisions and optimize fleet operations in real-time. Moreover, exploring opportunities to leverage artificial intelligence (AI) and autonomous vehicle technologies can further enhance the capabilities and efficiency of fleet management systems.

There is a need to focus on data governance and cybersecurity measures to ensure the integrity, confidentiality, and availability of fleet data. Implementing robust data governance policies, data quality assurance processes, and cybersecurity protocols can safeguard sensitive fleet information and mitigate the risk of data breaches or cyberattacks. Additionally, fostering a culture of data-driven decision-making and continuous improvement within organizations is essential to maximize the value and impact of fleet management initiatives.

The collaboration and partnerships with industry stakeholders, technology providers, and research institutions can facilitate knowledge sharing, innovation, and the adoption of best practices in fleet management. By leveraging collective expertise and resources, organizations can accelerate the development and deployment of next-generation fleet management solutions and drive positive outcomes for the transportation industry as a whole. Overall, by embracing these recommendations and staying at the forefront of technological advancements, the fleet management project can continue to deliver value, drive innovation, and contribute to the sustainable and efficient management of vehicle fleets in the future.

Bibliography

[1] Bakdi, A., Kristensen, N. B., Stakkeland, M. (2022). Multiple Instance Learning With Random Forest for Event Logs Analysis and Predictive Maintenance in Ship Electric Propulsion System. IEEE Transactions on Industrial Informatics, 18(11), 7718-7728.

[2] Mallouk, I., Sallez, Y., El Majd, B. A. (2021). Machine learning approach for predictive maintenance of transport systems. In 2021 Third International Conference on Transportation and Smart Technologies (TST) (pp. 1-6). IEEE

[3] Taslim, H. A., Yahaya, N. A. Yahya, N. A. M. (2023). Supervised learning models for health condition-based classification of remaining useful life in predictive maintenance: A preliminary study. AIP Conf. Proc. 2808.

[4] Giannoulidis, A., Gounaris, A. (2023). A context-aware unsupervised predictive maintenance solution for fleet management. Journal of Intelligent Information Systems, 60, 521–547.

[5] Maktoubian, J., Taskhiri, M. S., Turner, P. (2021). Intelligent Predictive Maintenance (IPdM) in Forestry: A Review of Challenges and Opportunities. Forests, 12(11).

[6] Çınar, Z. M., Nuhu, A. A., Zeeshan, Q., Korhan, O., Asmael, M., Safaei, B. (2020). Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0. Sustainability, 12(19), 8211.

[7] Maktoubian, J., M. S., Turner, P. (2021). Intelligent Predictive Maintenance (IPdM) in Forestry: A Review of Opportunities. *Forests, 12*(11), 1495.

[8]] Sun Yong, Xu Zhentao, Zhang Tianyu. (2019). On-Board Predictive Maintenance with Machine Learning. SAE Technical Paper 2019-01-1048.

[9] Li Haonan, Ribeiro Marta, Santos Bruno, Tseremoglou Iordanis. (2024). Prediction of Non-Routine Tasks Workload for Aircraft Maintenance with Supervised Learning. AIAA Paper 2024-2529.

[10] Francesca Cipollini, Luca Oneto, Andrea Coraddu, Alan John Murphy, Davide Anguita(2017). Condition-Based Maintenance of Naval Propulsion Systems with supervised Data Analysis. 13, I-16145.

Appendices

Appendix A

Source code

```python
#step-1 Data Collection
# Import necessary libraries
 # Importing Pandas library for data manipulation
import pandas as pd
# Importing train_test_split function for splitting data
from sklearn.model_selection import train_test_split
  # Importing RandomForestClassifier for classification
from sklearn.ensemble import RandomForestClassifier
# Importing metrics for model evaluation
from sklearn.metrics import accuracy_score, classification_report
# Importing LabelEncoder for encoding categorical variables
from sklearn.preprocessing import LabelEncoder
# Importing metrics for model evaluation
from sklearn import metrics
#importing dataset
# Importing files module for file uploading
from google.colab import files
# Uploading dataset
uploaded = files.upload()
# Reading dataset into DataFrame

data = pd.read_csv('annual-fleet-maintenance-division-statistics-1 (1).csv')
# Printing column names
print(data.columns)

 # Displaying shape of the DataFrame
data.shape
# Displaying descriptive statistics of the DataFrame
data.describe()
 # Displaying the number of missing values in each column
data.isnull().sum()

# Visualize distribution of numeric features
# Importing matplotlib for data visualization
import matplotlib.pyplot as plt
 # Importing seaborn for enhanced data visualization
import seaborn as sns
# Setting figure size
plt.figure(figsize=(12, 8))
# Creating histogram
sns.histplot(data['Total Vehicles'], bins=30, kde=True, color='blue')
```

```
# Setting title of the plot
plt.title(’Distribution of Vehicles’)
# Setting x-axis label
plt.xlabel(’Number of Light Duty Vehicles’)
# Setting y-axis label
plt.ylabel(’Repairs’)
# Displaying the plot
plt.show()

# Splitting data into features and target variable
# Selecting features
X = data.drop(columns=[”Preventative Maintenance”, ”Repairs”]) # Selecting target variable
y = data[”Repairs”]

# Splitting the data into training and testing sets
# Splitting data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initializing and training the Random Forest model
# Initializing model
rf_model = RandomForestRegressor(n_estimators=100, random_state=42)
# Training model
rf_model.fit(X_train, y_train)

# Making predictions on the testing set
# Making predictions
y_pred = rf_model.predict(X_test)

# Calculating Mean Squared Error
# Calculating MSE
rf_mse = mean_squared_error(y_test, y_pred)
# Printing MSE
print(”Mean Squared Error for Random Forest:”, rf_mse)

# Importing required libraries
# Importing Pandas for data manipulation
import pandas as pd
# Importing KMeans for clustering
from sklearn.cluster import KMeans
 # Importing matplotlib for data visualization
import matplotlib.pyplot as plt

# Selecting relevant features for clustering
# Selecting features
X = data.drop(columns=[”Preventative Maintenance”, ”Repairs”])

# Initializing and fitting KMeans model
# Initializing model
kmeans = KMeans(n_clusters=3, random_state=42)
# Fitting model
kmeans.fit(X)

# Getting cluster labels
 # Getting cluster labels
cluster_labels = kmeans.labels_

```

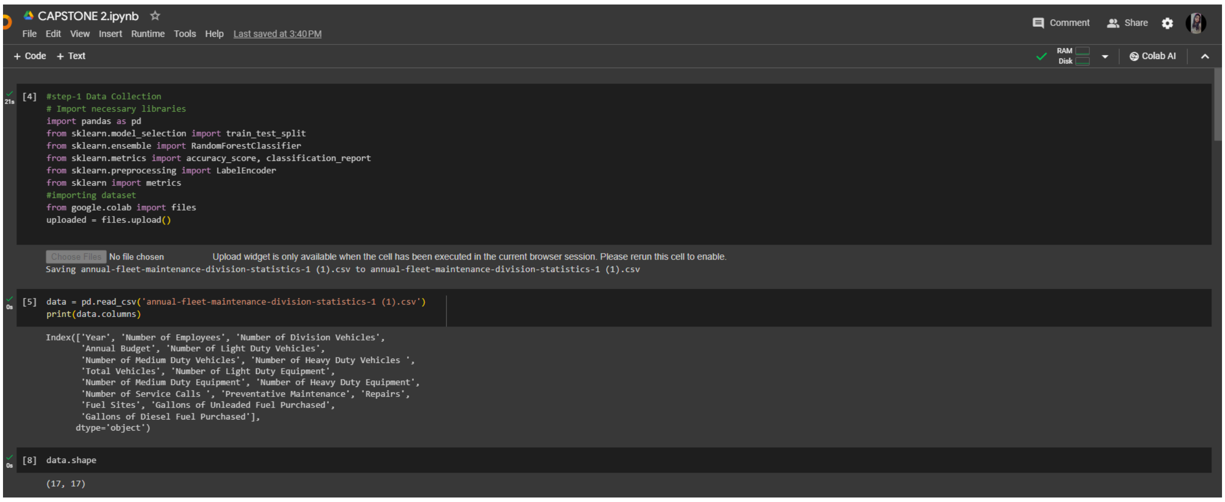
```
# Adding cluster labels to the fleet data
# Adding cluster labels as a new column
data[’Cluster’] = cluster_labels

# Visualizing clusters
# Creating scatter plot
plt.scatter(data[’Annual Budget’], data[’Repairs’], c=cluster_labels, cmap=’viridis’)
 # Setting x-axis label
plt.xlabel(’Annual Budget’)
# Setting y-axis label
plt.ylabel(’Repairs’)
# Setting title of the plot
plt.title(’KMeans Clustering of Fleet Data’)
# Displaying the plot
plt.show()

# Checking the distribution of clusters
# Printing distribution of clusters
print(data[’Cluster’].value_counts())

# Importing NumPy for numerical computations
import numpy as np
# Importing Pandas for data manipulation
import pandas as pd
# Importing KMeans for clustering
from sklearn.cluster import KMeans

# Assuming data is your DataFrame containing the data
kmeans = KMeans(n_clusters=3)  # Initializing model
kmeans.fit(data)  # Fitting model

# Get the centroids and labels
 # Getting centroids
centroids = kmeans.cluster_centers_
# Getting labels
labels = kmeans.labels_

# Calculate the MSE
# Initializing MSE variable
mse = 0
 # Looping through data points
for i in range(len(data)):
# Getting cluster index
    cluster_index = labels[i]
    # Calculating MSE
    mse += np.sum((data.iloc[i] - centroids[cluster_index]) ** 2)
     # Calculating mean MSE
kmeans_mse /= len(data)

# Printing MSE
print(”Mean Squared Error (MSE):”, kmeans_mse)

# Print MSE for both models
 # Printing Random Forest MSE
print(”Random Forest MSE:”, rf_mse)
# Printing K-means MSE
print(”K-means MSE:”, kmeans_mse)
```

```

# Determine the best model
# Checking if Random Forest MSE is less than K-means MSE
if rf_mse < kmeans_mse:
# Printing statement
    print(”Random Forest is the better model.”)
else:
    # Printing statement
    print(”K-means clustering is the better model.”)
```

Appendix B

Screen shots

B.1 Results


CAPSTONE 2.ipynb
File Edit View Insert Runtime Tools Help Last saved at 3:40 PM
Comment Share
+ Code + Text
RAM Disk
Colab AI

```
[4] #step-1 Data Collection
    # Import necessary libraries
    import pandas as pd
    from sklearn.model_selection import train_test_split
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.metrics import accuracy_score, classification_report
    from sklearn.preprocessing import LabelEncoder
    from sklearn import metrics
    #importing dataset
    from google.colab import files
    uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

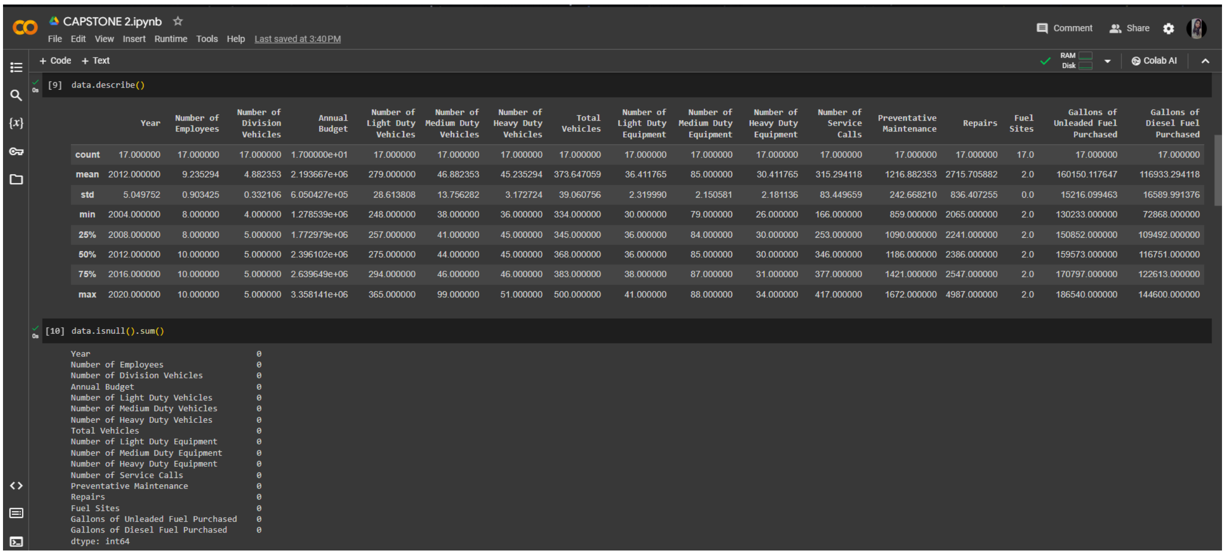
```
Saving annual-fleet-maintenance-division-statistics-1 (1).csv to annual-fleet-maintenance-division-statistics-1 (1).csv
```

```
[5] data = pd.read_csv('annual-fleet-maintenance-division-statistics-1 (1).csv')
    print(data.columns)
```

```
Index(['Year', 'Number of Employees', 'Number of Division Vehicles',
       'Annual Budget', 'Number of Light Duty Vehicles',
       'Number of Medium Duty Vehicles', 'Number of Heavy Duty Vehicles ',
       'Total Vehicles', 'Number of Light Duty Equipment',
       'Number of Medium Duty Equipment', 'Number of Heavy Duty Equipment',
       'Number of Service Calls ', 'Preventative Maintenance', 'Repairs',
       'Fuel Sites', 'Gallons of Unleaded Fuel Purchased',
       'Gallons of Diesel Fuel Purchased'],
      dtype='object')
```

```
[8] data.shape
```

```
(17, 17)
```

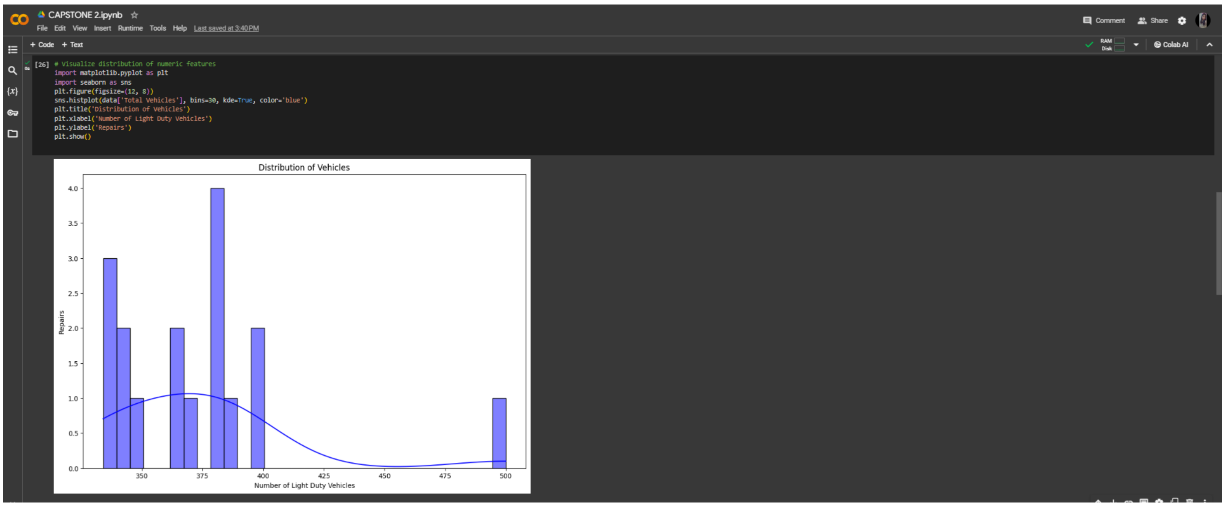

CAPSTONE 2.ipynb
File Edit View Insert Runtime Tools Help Last saved at 3:40 PM
Comment Share
+ Code + Text
RAM Disk
Colab AI

```
[9] data.describe()
```

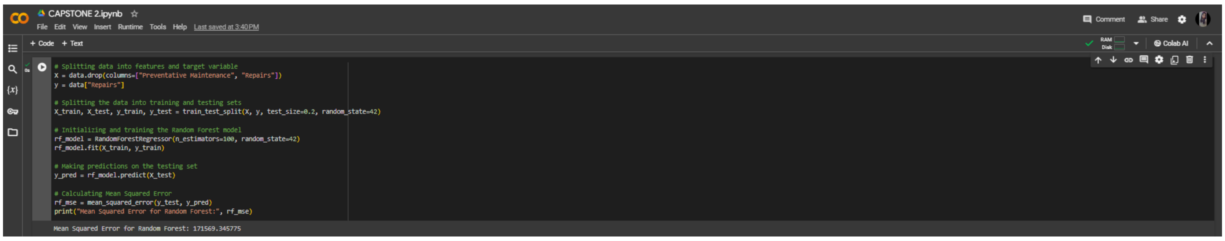
	Year	Number of Employees	Number of Division Vehicles	Annual Budget	Number of Light Duty Vehicles	Number of Medium Duty Vehicles	Number of Heavy Duty Vehicles	Total Vehicles	Number of Light Duty Equipment	Number of Medium Duty Equipment	Number of Heavy Duty Equipment	Number of Service Calls	Preventative Maintenance	Repairs	Fuel Sites	Gallons of Unleaded Fuel Purchased	Gallons of Diesel Fuel Purchased
count	17.000000	17.000000	17.000000	1.700000e+01	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000	17.0	17.000000	17.000000
mean	2012.000000	9.235294	4.882353	2.193667e+06	279.000000	46.882353	45.235294	373.647059	36.411765	85.000000	30.411765	315.294118	1216.882353	2715.705882	2.0	160150.117647	116933.294118
std	5.049752	0.903425	0.332106	6.050427e+05	28.613808	13.756282	3.172724	39.060756	2.319990	2.150581	2.181136	83.449659	242.668210	836.407255	0.0	15216.099463	16589.991376
min	2004.000000	8.000000	4.000000	1.278539e+06	248.000000	38.000000	36.000000	334.000000	30.000000	79.000000	26.000000	166.000000	859.000000	2065.000000	2.0	130233.000000	72868.000000
25%	2008.000000	8.000000	5.000000	1.772979e+06	257.000000	41.000000	45.000000	345.000000	36.000000	84.000000	30.000000	253.000000	1090.000000	2241.000000	2.0	150852.000000	109492.000000
50%	2012.000000	10.000000	5.000000	2.396102e+06	275.000000	44.000000	45.000000	368.000000	36.000000	85.000000	30.000000	346.000000	1186.000000	2386.000000	2.0	159573.000000	116751.000000
75%	2016.000000	10.000000	5.000000	2.639649e+06	294.000000	46.000000	46.000000	383.000000	38.000000	87.000000	31.000000	377.000000	1421.000000	2547.000000	2.0	170797.000000	122613.000000
max	2020.000000	10.000000	5.000000	3.358141e+06	365.000000	99.000000	51.000000	500.000000	41.000000	88.000000	34.000000	417.000000	1672.000000	4987.000000	2.0	186540.000000	144600.000000

```
[10] data.isnull().sum()
```

```
Year                                  0
Number of Employees                   0
Number of Division Vehicles           0
Annual Budget                         0
Number of Light Duty Vehicles         0
Number of Medium Duty Vehicles        0
Number of Heavy Duty Vehicles         0
Total Vehicles                        0
Number of Light Duty Equipment        0
Number of Medium Duty Equipment       0
Number of Heavy Duty Equipment        0
Number of Service Calls               0
Preventative Maintenance              0
Repairs                               0
Fuel Sites                            0
Gallons of Unleaded Fuel Purchased    0
Gallons of Diesel Fuel Purchased      0
dtype: int64
```

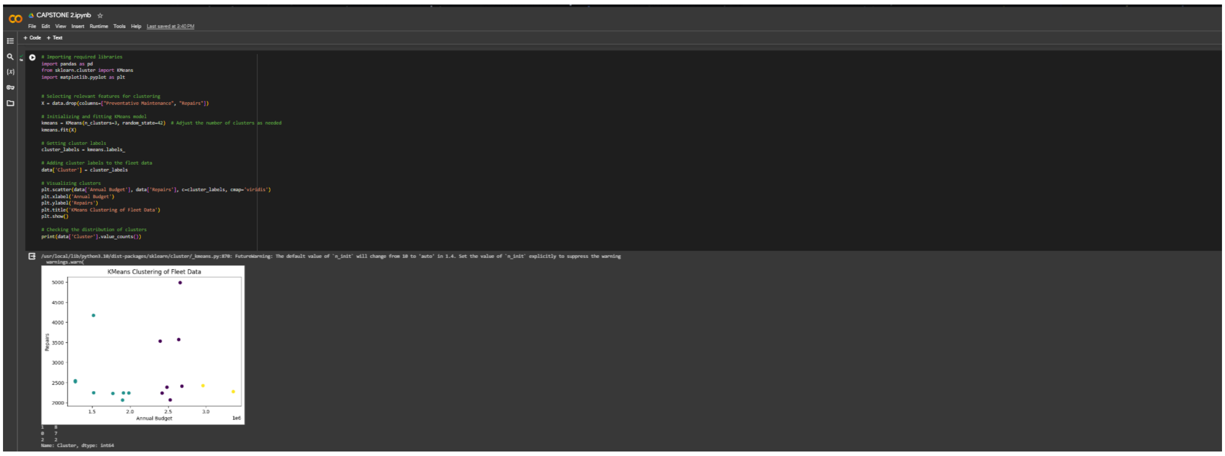


```
[26] # Visualize distribution of numeric features
     import matplotlib.pyplot as plt
     import seaborn as sns
     plt.figure(figsize=(12, 8))
     sns.histplot(data['Total Vehicles'], bins=30, kde=True, color='blue')
     plt.title('Distribution of Vehicles')
     plt.xlabel('Number of Light Duty Vehicles')
     plt.ylabel('Repairs')
     plt.show()
```


CAPSTONE 2.ipynb
File Edit View Insert Runtime Tools Help Last saved at 3:40 PM
+ Code + Text
Comment Share
Colab AI

```
# Splitting data into features and target variable
X = data.drop(columns=["Preventative Maintenance", "Repairs"])
y = data["Repairs"]

# Splitting the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initializing and training the Random Forest model
rf_model = RandomForestRegressor(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

# Making predictions on the testing set
y_pred = rf_model.predict(X_test)

# Calculating Mean Squared Error
rf_mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error for Random Forest:", rf_mse)
```

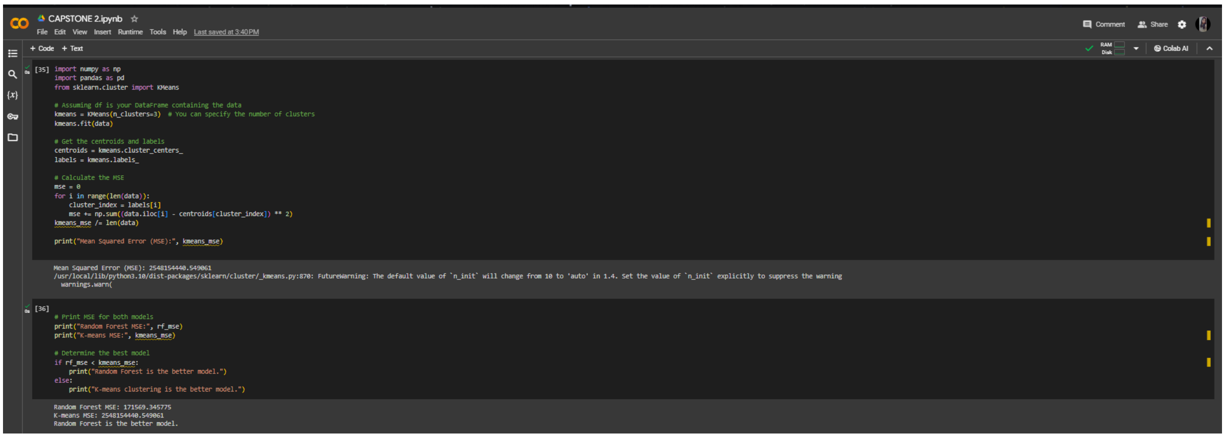
```
Mean Squared Error for Random Forest: 171569.345775
```


CAPSTONE 2.ipynb
File Edit View Insert Runtime Tools Help Last saved at 3:40 PM
+ Code + Text
KMeans Clustering of Fleet Data
Repairs
Annual Budget
2000
2500
3000
3500
4000
4500
5000
1.5
2.0
2.5
3.0
1e6

```
# Importing required libraries
import pandas as pd
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt


# Selecting relevant features for clustering
X = data.drop(columns=["Preventative Maintenance", "Repairs"])

# Initializing and fitting KMeans model
kmeans = KMeans(n_clusters=3, random_state=42)  # Adjust the number of clusters as needed
kmeans.fit(X)

# Getting cluster labels
cluster_labels = kmeans.labels_

# Adding cluster labels to the fleet data
data['Cluster'] = cluster_labels

# Visualizing clusters
plt.scatter(data['Annual Budget'], data['Repairs'], c=cluster_labels, cmap='viridis')
plt.xlabel('Annual Budget')
plt.ylabel('Repairs')
plt.title('KMeans Clustering of Fleet Data')
plt.show()

# Checking the distribution of clusters
print(data['Cluster'].value_counts())
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
```

```
1    8
0    7
2    2
Name: Cluster, dtype: int64
```


CAPSTONE 2.ipynb
File Edit View Insert Runtime Tools Help Last saved at 3:40 PM
+ Code + Text
Comment Share
Colab AI

```
[35] import numpy as np
     import pandas as pd
     from sklearn.cluster import KMeans

     # Assuming df is your DataFrame containing the data
     kmeans = KMeans(n_clusters=3)  # You can specify the number of clusters
     kmeans.fit(data)

     # Get the centroids and labels
     centroids = kmeans.cluster_centers_
     labels = kmeans.labels_

     # Calculate the MSE
     mse = 0
     for i in range(len(data)):
         cluster_index = labels[i]
         mse += np.sum((data.iloc[i] - centroids[cluster_index]) ** 2)
     kmeans_mse /= len(data)

     print("Mean Squared Error (MSE):", kmeans_mse)
```

```
Mean Squared Error (MSE): 2548154440.549061
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
```

```
[36]
     # Print MSE for both models
     print("Random Forest MSE:", rf_mse)
     print("K-means MSE:", kmeans_mse)

     # Determine the best model
     if rf_mse < kmeans_mse:
         print("Random Forest is the better model.")
     else:
         print("K-means clustering is the better model.")
```

```
Random Forest MSE: 171569.345775
K-means MSE: 2548154440.549061
Random Forest is the better model.
```

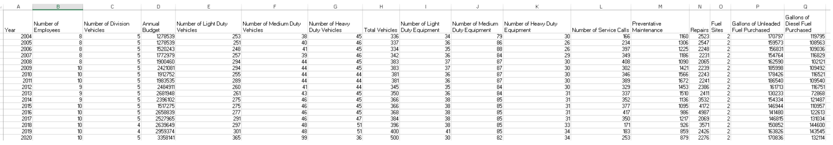
Appendix C

Data sets used in the project

The dataset utilized in this project was sourced from Kaggle, accessible via the following link:

https://data.world/city-of-bloomington/df8b2bf5-ef28-403a-90a9-d21f276e7961

It provides a comprehensive collection of the Fleet Maintenance Division is responsible for the safe and efficient maintenance and repair, as well as the distribution of unleaded and diesel fuel, for the City's fleet of vehicles and equipment. These services ensure that City departments have the vehicles and equipment necessary to provide their daily services to the residents of Bloomington. Annual statistics about operations of the Fleet Maintenance division of the Public Works Department includes annual budget figures, total employees, number and type of vehicles and equipment, number of repairs completed, service calls, preventative maintenance activities and more.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
Year	Number of Employees	Number of Division Vehicles	Annual Budget	Number of Light Duty Vehicles	Number of Medium Duty Vehicles	Number of Heavy Duty Vehicles	Total Vehicles	Number of Light Duty Equipment	Number of Medium Duty Equipment	Number of Heavy Duty Equipment	Number of Service Calls	Preventative Maintenance	Repairs	Fuel Sites	Gallons of Unleaded Fuel Purchased	Gallons of Diesel Fuel Purchased
2004	8	5	1278539	253	38	45	336	34	79	30	166	1160	2523	2	170797	119795
2005	8	5	1278539	251	40	46	337	36	86	26	234	1306	2547	2	159573	108563
2006	8	5	1520243	248	41	45	334	35	88	26	397	1225	2248	2	156831	109036
2007	8	5	1772979	257	39	46	342	36	84	29	349	1186	2231	2	154764	116829
2008	8	5	1900460	294	44	45	383	37	87	30	408	1090	2065	2	162590	102121
2009	10	5	2421081	294	44	45	383	37	87	30	302	1421	2239	2	185998	109492
2010	10	5	1912752	255	44	44	381	36	87	30	346	1566	2243	2	178426	116521
2011	10	5	1983535	289	44	44	381	36	87	30	389	1672	2241	2	186540	109540
2012	9	5	2484911	260	41	44	345	35	84	30	329	1453	2386	2	161713	116751
2013	9	5	2681948	261	43	45	350	36	84	31	337	1510	2411	2	130233	72868
2014	9	5	2396102	275	46	45	366	38	85	31	352	1136	3532	2	154334	121487
2015	10	5	1517275	275	46	45	366	38	85	31	377	1095	4172	2	146944	110957
2016	10	5	2658839	277	46	45	368	38	85	31	417	986	4987	2	141480	122613
2017	10	5	2527965	291	46	47	384	38	85	31	350	1217	2069	2	146815	131034
2018	10	4	2639649	297	48	51	396	38	85	33	171	926	3571	2	150852	144600
2019	10	4	2959374	301	48	51	400	41	85	34	183	859	2426	2	163826	143545
2020	10	5	3358141	365	99	36	500	30	82	34	253	879	2276	2	170836	132114