

UNIVERSITY of RWANDA



AFRICAN CENTRE OF EXCELLENCE IN DATA SCIENCE

University of Rwanda

College of Business and Economics

APPLICATION OF MACHINE LEARNING TECHNIQUES FOR INCIDENT – ACCIDENT CLASSIFICATION PROBLEM IN AVIATION SAFETY MANAGEMENT

A Dissertation Submitted

In Partial Fulfilment of the Requirement for the MSc in Data Mining

Rukabu Oswald

Under supervision of

Dr. Japhet Niyobuhungiro

September 2020

Declaration

I declare that this dissertation entitled **application of machine learning techniques for incident – accident classification problem in aviation safety management** is the result of my own work and has not been submitted for any other degree at the University of Rwanda or any other institution.

Names: Rukabu Oswald

Signature



Approval sheet

This dissertation entitled **application of machine learning techniques for incident – accident classification problem in aviation safety management** written and submitted by Rukabu Oswald in partial fulfilment of the requirements for the degree of Master of Science in Data Science majoring in data mining is hereby accepted and approved. The rate of plagiarism tested using Turnitin is 18 % which is less than 20% accepted by the African Centre of Excellence in Data Science (ACE-DS).



Supervisor



Head of Training

Dedication

I dedicate my dissertation to God Almighty, my creator, my strong pillar, my source of inspiration, wisdom, knowledge, and understanding. I also dedicate my work to my whole family for the endless support and encouragement throughout my studies. Thank You. My love for you all can never be qualified. God Bless you.

Acknowledgements

I would like first to thank God for His protection. It is my privilege to release the feeling of my gratitude to my people who supported me directly or indirectly to complete my Master's Degree program. I express my heart full indebtedness and owe a deep sense of gratitude to my supervisor Dr. Japhet Niyobuhungiro for his sincere guidance and inspiration in completing my thesis.

I also thank all my classmates, friends and family members for their supports direct or indirect and cooperation and for their kind guidance and encouragement. I will always be indebted to them. God bless you all.

Abstract

In a global world, one of key contributors to business success is the aviation industry. Most business magnets rely on this to quickly travel from one place to another to conduct transactions. This has led to an increase in demand of flights with many more aviation entrepreneurs cropping up every now and then. A key challenge faced by this industry is transporting passengers and goods safely from departure point to their destination. It is not uncommon for flights to encounter an incident/accident problem which often perturbs the journey. Thus, companies continuously strive to mitigate such occurrence.

Aviation companies record historical data in various databases which is rich with insight that can be extracted and used to mitigate events that impede flights. In this research, we employed machine learning techniques to identify potential contributory factors and trained machine learning models using supervised learning. These models learn from existing data and are used to predict when an aircraft incident or accident would occur given some conditions. In our findings, the best model achieved an accuracy of 96% when tested on unseen data.

Keywords: Machine Learning, Data Mining, Aviation Safety, Analysis Method, Aircraft Accident.

Table of Contents

Declaration i
Approval sheet ii
Dedication iii
Acknowledgements iv
Abstract v
List of Figures viii
List of Tables ix
List of Equations x
1.0 General introduction 1
1.1 Background 1
1.2 Problem Statement 1
1.3 Research objectives 2
1.3.1 Main objectives 2
1.3.2 Specific objective 2
1.4 Overview of the data mining techniques 2
1.5 Causes of accidents and incidents 3
2.0 Literature review 5
2.1 Introduction 5
2.2 Importance of machine learning in aviation safety management 5
2.3 Contributory factors for aviation incidents or accidents 6
3.0 Methodology 7
3.1 Introduction 7
3.2 Data preparation 7
3.3 Machine leaning approaches 8
3.3.1 Predictive model 8
3.3.2 Support Vector Machine (SVM) 9
3.3.3 Naïve Bayes 10
3.3.4 Decision Tree 11
3.3.5 Random Forest 13
3.3.6 Parameter tuning 15
3.3.7 Metrics to evaluate machine learning algorithms 15
3.4 Description of dataset 19
3.4.1 Exploratory data analysis 20
3.4.2 Data pre-processing 20
3.4.3 Import Python libraries for data analysis. 21
3.4.4 Data cleaning 21
3.4.5 Handling missing values using pandas. 22

3.4.6 Data transformation 30
3.4.7 Feature selection 32
4.0 Analysis and results 36
4.1 Introduction 36
4.2 Feature importance 36
4.3 Direct classification analysis 37
4.3.1 Decision Tree 37
4.3.2 Random Forest 38
4.3.3 Support Vector Machine (SVM) 40
4.3.4 Naïve Bayes 41
4.3.5 Summary of performance evaluation on different classifiers - Directclassification 42
5.0 Conclusion and Recommendations 45
6.0 References 48

List of Figures

Figure 1: The maximum-margin hyperplane separating samples from two classes 9
Figure 2: Decision tree algorithm 11
Figure 3: Classify response to direct mailing using decision tree..... 12
Figure 4: Python libraries for data analysis 21
Figure 5: Summary of missing values in our dataset 23
Figure 6: Heat map function represents distribution of null values 24
Figure 7: Python codes illustrating investigation type 25
Figure 8: Python codes illustrating amateur built 26
Figure 9: Python codes illustrating phase of flight 27
Figure 10: Python codes illustrating variable number of engines..... 28
Figure 11: Python code replacing missing values with the modal value for the four attributes and the result..... 29
Figure 12: Dataset after cleaning, without null value 30
Figure 13: Python code for transformation of investigation type from categorical to numerical 30
Figure 14: Summary of investigation type after transformation from categorical to numerical 31
Figure 15: Summary of weather condition after transformation from categorical to numerical 31
Figure 16: Summary of amateur built after transformation from categorical to numerical 31
Figure 17: Summary of phase of flight after transformation from categorical to numerical .. 31
Figure 18: Summary of number of engine with numerical values..... 32
Figure 19: Summary of correlations between variables in our dataset. 34
Figure 20: Shape of our dataset 34
Figure 21: All variables in our dataset 35
Figure 22: Filtered features (potential features contributory to A/c Accident) 35
Figure 23: Feature importance..... 36
Figure 24: Classification report and accuracy score for Decision Tree classifier 37
Figure 25: Confusion matrix for Decision Tree classifier..... 38
Figure 26: Graph representing performance of Decision Tree classifier based on area under curve..... 38
Figure 27: Classification report and accuracy score for Random Forest classifier 39
Figure 28: Confusion matrix for Random Forest classifier..... 39
Figure 29: Graph representing performance of Random Forest classifier based on area under curve..... 39
Figure 30: Classification report and accuracy score for SVM classifier..... 40
Figure 31: Confusion matrix for SVM classifier..... 40
Figure 32: Graph representing performance of SVM classifier based on AUC..... 40
Figure 33: Classification report and accuracy score for Naïve Bayes classifier 41
Figure 34: Confusion matrix for Naïve Bayes classifier..... 41
Figure 35: Graph represents performance of Naïve Bayes classifier based on AUC..... 41
Figure 36: Evaluation performance of different classification models based on AUC on aircraft accident data..... 42

List of Tables

Table 1: Confusion matrix 17
Table 2: Summary of our dataset 20
Table 3: Summary of our dependent variable (investigation type) 24
Table 4: Summary of amateur built 25
Table 5: Summary of phase of flight 27
Table 6: Summary of number of engines 28
Table 7: Summary of weather condition 29
Table 8: Summary of all features in our dataset 33
Table 9: Summary on how independent variable contribute to the target variable 36
Table 10: Summary of direct classified accuracy 37
Table 11: Performance of Decision Tree classifier 43
Table 12: Performance of Random Forest classifier 43
Table 13: Performance of SVM classifier 44
Table 14: Performance of Naïve Bayes classifier 44
Table 15: Summary of accuracy for different classifier before and after parameter tuning 44

List of Equations

Equation 3-1 8
Equation 3-2 9
Equation 3-3 9
Equation 3-4 10
Equation 3-5 10
Equation 3-6 10
Equation 3-7 11
Equation 3-8 13
Equation 3-9 13
Equation 3-10 13
Equation 3-11 14
Equation 3-12 14
Equation 3-13 14
Equation 3-14 14
Equation 3-15 14
Equation 3-16 14
Equation 3-17 16
Equation 3-18 17
Equation 3-19 17
Equation 3-20 18
Equation 3-21 18
Equation 3-22 18
Equation 3-23 33
Equation 3-24 33

1.0 General introduction

1.1 Background

A significant number of Africans and the rest of the world population travel weather for business trips or leisure on a daily basis. Since 1950s the speed of traffic volume has been considerable worldwide [1]. Air transport can be considered as one of the most essential services that contribute to social and economic growth. The aviation industry is a key promoter of tourism – a sector many economies are focusing on because of its impacts. Safety is the underlying principle in aviation field. It is considered the most critical factor to the growth of aviation industry worldwide, as airlines which record numerous accidents are bound to lose their customers. More African countries, Rwanda inclusive, are witnessing fast growth in the aviation industry. Because of the great increase in air traffic congestion; the rate of flight delays; the rate of aircraft incidents and accidents are anticipated to rise if no measures are taken to guarantee the safe operation of aircraft.

In this research, we aim at predicting the occurrence of an aircraft accident/incident by extracting insight from historical data. This insight would be used to guide the decision-making process of aviation stakeholders. Like other aviation companies in Africa and worldwide, aviation companies in Rwanda need to employ data analytics approaches for sound decision making, other than relying solely on traditional methods which often than not, do not capture much of the insight found in data. It should be necessary to acquire and employ new methods and tools through the advancement of research capabilities to promote safety in aviation field and other areas related to aviation.

One of the key components of aviation safety management is information regarding previous unusual circumstances. In reality to foresee and prevent what might happen one needs to gather as much as information about what has happened. Machine learning is a combination of methods and tools which could provide possible answer to the mentioned issue.

Analysing safety related records is one of the most potential means of preventing aircraft incidents and accidents from occurring.

1.2 Problem Statement

Aviation industries worldwide regard safety as its first priority. They are committed to developing, implementing, maintaining and constantly improving safety management systems to achieve the highest level of safety performance and meet or exceed national and international

standards. All their employees, managers and staff alike, without any exception are accountable and responsible for safety while delivering the services.

To achieve this, they establish comprehensive procedures for collecting, analyzing, reporting and storing of data on hazards, incidents and accidents, and often strive for continual improvements through safety assurance.

Most aviation companies in Africa and worldwide rely on basic data analysis using tools like Excel and query tools to support their decision-making process. Ironically, the aviation industry is one of the sectors that accumulates large amounts of data pregnant with a lot of insight. The data could be structured and stored in a relational database or unstructured data. By using advanced analytics approaches, knowledge discovered from these data could be used to improve safety in the aviation industry.

In this dissertation, we apply machine learning algorithms to mine insight found in aviation data and predict when an accident or incident would occur given prevailing conditions. We use classification techniques including random forest, Decision Tree, SVM and Naives Bayes models to address this problem.

1.3 Research objectives

1.3.1 Main objectives

To improve aviation safety management by predicting incident/accident using machine learning techniques.

1.3.2 Specific objective

- ✓ To identify key attributes that contribute to aviation incidents and accidents.
- ✓ Develop machine learning models to learn from historical data and predict the occurrence of an aviation incident/accident given a data point.
- ✓ To provide some recommendations to the aviation industry to establish new procedures for collecting, analyzing and storing of data on hazards, incidents and accidents.

1.4 Overview of the data mining techniques

Data mining is characterized as a strategy of retrieving insight from big sets of data [2]. As such, we can state that data mining is mining information from data. The target of data mining is to search for relationships and global patterns that exist in large databases but are hidden in the vast amounts of data [3]. It might take years to identify these significant relationships embedded in the data with traditional techniques.

Data analysts with advanced machine learning technology can discover the relationships between features and hidden patterns. Useful patterns can be identified from large aviation datasets. To capture these patterns and make use of them in solving the problem at hand, the following four steps are conducted:

1. **Data description:** Data are described by its shape and statistical attributes, by visually reviewing charts and graphs and by searching for relationship among attributes [4].
2. **Exploratory data analysis:** Data Exploration is a critical part of data analysis. Through exploration, insight is obtained from the data and patterns and anomalies are discovered. Statistical summaries and graphical presentations are often used to present findings [5].
3. **Build a predictive model**: Predictive models are constructed, which learn from the patterns found in the data [4].
4. **Verify the model**: An unseen dataset is then used to test the performance of the model trained. Adjustments are made to the model until a satisfactory performance is attained [4].

1.5 Causes of accidents and incidents

In aviation industry, safety experts recognize that aircraft accidents or incidents are always as a result of a sequence of events including technical and human errors.
To identify the cause of an aircraft accident, a sequence of investigations needs to be conducted in order to arrive at the cause. Focus is not concentrated on the immediate cause only as many factors always build up to an aircraft event. For example, in the case of the 1977 air disaster between flight 8405 operated by KLM and flight 1736 operated by Pan Am on the runway at Rodeos airport in Spain, investigation revealed that sequences of events contributed to this world known aviation disaster. More than 5 critical causal event factors were involved. If any of these events did not occur, the disaster would have been avoided [5]. The most successful techniques to avoid aircraft incidents or accidents is to take into account all the linkages in the series of events leading to the disaster.

If investigators want to identify the probable causes of aircraft accident or incident, the past incidents and accidents, and causal factors should be taken into consideration. The following are some potential causal factors: Human factors from personnel errors; Malfunction of aircraft systems; Engine failure, Flight controls failure; Pneumatic systems, Avionics, Poor

aircraft maintenance; Safety risks environment involving weather, volcanic ash, bird's activities, etc.; Air traffic control error or negligence; Non-compliance of air traffic control instructions; and any combination of different factors.

In order to conduct investigation on aircraft accidents or incidents the following definitions by the international civil aviation organisation (ICAO), have to be taken into consideration.

- **Causes:** "Causes are actions, omissions, events, conditions, or a combination thereof, that lead to an accident or incident"[6].
- **Accidents:** "Accident are occurrences associated with the operation of aircraft, from the time any person boards an aircraft with the intention of flight until the time all persons have disembarked, that results in one or more of the following:
  - ✓ A person is fatally or seriously injured.
  - ✓ The aircraft sustains damage or structural failure that adversely affects the structural strength, performance, or flight characteristics of the aircraft and would normally require major repair or replacement of the affected component.
  - ✓ The aircraft is missing or completely inaccessible"[6].
- **Incidents:** "Incidents are occurrences, other than accidents, associated with the operation of aircraft that affect or could affect the safety of operation"[6].
- **Investigation:** "Investigation is a process conducted for the purpose of accident prevention which includes the gathering and analysis of information, the drawing of conclusions, including the determination of causes and, when appropriate, the making of safety recommendations"[6].
- **Aircraft:** "Aircraft is defined as any machine that can derive support in the atmosphere from the reactions of the air other than the reactions of the air against the earth's surface"[6].

2.0 Literature review

2.1 Introduction

This section reviews previous work on the benefit of using machine learning approaches to analyse aviation incident and accident reports. Machine learning techniques for predicting topmost causes of aviation incidents are critically examined. Knowledge gotten from the literature guided the approaches selected to address our research problem.

2.2 Importance of machine learning in aviation safety management

It is clear that large datasets, like those containing aviation incidents and accidents and other unusual events, contain a large number of patterns. The process of aviation safety related data analysis starts with recording the reports from different aviators to the safety officer for further analysis, investigation and recommendation [2]. Aircraft accident and incident data include both structured and unstructured data. Data in structured format usually stored in databases, are accessed with queries and easily analysed. For structured data, analyses and visualization can easily be done using graphic tools once the data is queried [7]. This is totally different from unstructured data. Before the machine learning approaches were developed, we could not find methods to analyse unstructured data such as in textual, audio, etc. [3]. In many companies, the analysis relies on the observation and experience of the officer in charge of safety [2]. When using machine learning, one does not have to start with hypothesis or previous knowledge. With machine learning approaches, we identify within a dataset patterns, trends and anomalies not previously thought of [8].
Machine learning methods can be carried out in aviation safety management using a moderate, home- grown aviation safety recording systems and basic tools, or with sophisticated software and professional, commercially accessible aviation safety databases. The usage of machine learning in aviation management mainly is to establish patterns in safety data and occurrences that are unseen in large amounts of disparate information; Validity of identified trends to predict whether risk controls successfully anticipate expected, future events; Correlations between different safety elements; Identify patterns in both employee and managerial behaviours; and Extract important safety knowledge from acquired safety documentation to facilitate data-based decision-making processes.

2.3 Contributory factors for aviation incidents or accidents

According to the US Department of transportation (Bureau of Transportation Statistics) "aviation has a safe superior record compared to the other modes of transportation" [9]. Massoud Bazargan and Vital S. Guzhva in their study, applied Logistic Regression model on a vast sample of general aviation accident data from 1983 to 2002. They identified the following factors as potential contributors to the aviation accidents: *light condition, pilot's experience, wind conditions, aircraft characteristics, flight phase and retractable landing gear*. Although several variables were in line with expectations that were established as leading to the aircraft accident, some of them were unintuitive [10].

Most of time, experts in aviation safety have recognized that aircraft accidents and incidents usually result from a variety of occurrences, each of which is related to one or more causative factors [11].

According to Nal reports (2006) "*pilot error, mechanical and maintenance problems and other unknown factors are the primary causes for 76%, 16% and 8% of all general aviation accidents respectively*" [12].

According to Li et al (2001) "more than 85% of all general aviation accidents have pilot error as a cause or factor. Causative factors could be classified into categories as follows*: Human factors/personnel error; Malfunction or failure of aircraft structures, engines, or other systems; Deficient maintenance; Hazardous environment involving weather, volcanic ash, birds, etc.; Air traffic management errors; Any combination of the various factors*" [13].

Lisa M. Vetlman in his research on incident data analysis using data mining techniques, applied two predictive models: Logistic Regression and Decision Tree models on a large sample of aviation data from 1983 to 2002, aiming at identifying statistically significant factors that raise the likelihood of having injuries or not. Logistic Regression model correctly predicted 1.72% injuries, and Decision Tree model correctly predicted 21% of the injury events [14].

The current research is distinct from those mentioned above, in the sense that we apply machine learning techniques to discover the most potential factors contributing to aircraft incidents or accidents and predict the potential occurrence of an incident or accident based on insight from historical data. We apply four classification models, namely Random Forest, Decision Tree, Support Vector Machine and Naïve Bayes classifiers and then do comparison of results looking for better classifier.

3.0 Methodology

3.1 Introduction

As already mentioned, data mining is a set of techniques used to find trends in historical data, and apply the knowledge gotten to solve problems or make sound decisions [15]. Problems solved using data mining could be descriptive in nature or could involve prediction of some target. In this study we begin by describing the data used. We identify the variables and perform data cleaning to get high-quality data. We then employ graphs and charts to visualize meaningful correlations between features. Feature selection is performed based on intuition and insight found in the data on how the independent variables impact the dependent variable. Using the selected features, predictive models are developed to predict the possible occurrence of an aircraft accident given each data record. The models are tested on unseen data and their performance compared to see how well they generalize. The approach used to address our main problem is encapsulated in the steps below:

1. Clean the data and prepare it for analysis.
2. Look for patterns and relationships.
3. Modify dataset as necessary.
4. Apply models to the dataset.
5. Assess how well the models fit the data.
6. Compare the performance of developed models on the used data.

3.2 Data preparation

Data preparation is a very essential stage in data analysis as an approach employed to get better results. Alternatively, we may call it pre-modeling. During data preparation, one performs a lot of work before processing and analyzing the data. This includes cleaning the dataset; transforming the dataset in readable format, renaming columns for easy recognition; and do some corrections to the dataset. Proper data preparation is required to guarantee the quality of the data. High data quality leads to a greater chance of getting better results from predictive models. Good data preparation provides high data quality for further analysis [16]. Generally data preparation allows efficient data analysis.

The following are the various steps involved in data preparation: Loading data, cleaning data by removing unnecessary or erroneous data, transforming data formats, and rearranging data. In this project we have employed Python library “pandas” in preparing our dataset.

3.3 Machine leaning approaches

Machine learning is a sub-field of computer science which emerged from the study of artificial technology trends, recognition and computational learning theory [17]. It discovers the constructional and analysis of algorithms that learn from a dataset and make predictions[18]. Thus, algorithms work by model building. According to Arthur Samuel "Machine learning is a field of study that gives computers the ability to learn without being explicitly programmed" [18].

This section describes the machine learning approaches used. The problem being solved is a classification problem, our task being to build predictive models to predict if an aircraft witnesses an accident or not.

3.3.1 Predictive model

Predictive model is a method that uses knowledge and statistics to predict data model results. Predictive models are developed using a supervised approach to learning. Supervised learning is where we have input X and output Y variables and use an algorithm to learn the mapping function from input to output [19].

$$Y = f(X).$$

Equation 3-1

The intention is to estimate the mapping function so that the output value (Y) is predicted once a new input value (X) has been mounted. Supervised learning problems are divided into two categories: the problem of classification and the problem of regression. Our problem in this analysis is the problem of classification. Each model is trained using historic data. To measure a model's performance, we provide new inputs with unknown outputs and determine how well the machine learning model correctly predicts the labels (outputs). Using Scikit-Learn's test_train_split algorithm, the dataset was divided into training and test sets, the training set being 80% and test set comprising 20%. The following classification models will be used to power the prediction and do comparison of their accuracy scores to get best model to our data: Naive Bayes; Support Vector Machine (SVM); Random Forest; and Decision Tree.

3.3.2 Support Vector Machine (SVM)

Support vector machine is a supervised machine learning algorithm which handles classification and regression problems and can manage multiple categorical and continuous variables. SVM is a classifier that separates data points using hyperplane and finds an optimal hyperplane to classify new data points. The optimal hyperplane is also used to minimize error [20]. Mathematically speaking, SVM is a classifier that finds a function (hyperplane) $h(x) = w^T\ x + b$ that correctly separates two classes with a maximum margin. Given a set of points x_i that belong to two linearly separated classes w_1, w_2 the distance of any instance from hyperplane is equal to $\frac{|h(x)|}{\|w\|}$. SVM purpose it to find w and b such that the value of $h(x)$ equals 1 for the nearest data point that belongs to w_1 and -1 for the nearest data point that belongs to w . This can be viewed as having the margin of $\frac{1}{\|w\|} + \frac{1}{\|w\|} = \frac{2}{\|w\|}$,

whereas for $w^T\ x + b = 1$ for $x \epsilon w_1$ and $w^T\ x + b = -1$ for $x \epsilon w_2$.

This leads to an optimization problem to minimize the function

$$(w^T\ w/2)\ + c\sum_{i=1}^{N}\beta_i,$$

Equation 3-2

subject to the limitations

$$y_i(w^T\ (x_i) + b) \geq 1 - \beta_i, \beta_i \geq 0,\ i = 1,2,3, \ldots \ldots . N.$$

Equation 3-3

Here, w is expressed as vector coefficient, β explains the boundary for new data point, c expresses capacity constant, and b is a constant, i expresses index for level T cases of training set, x_i expresses independent variables, and y_i expresses class labels. The constant c varies depending on the optimization goal. If c is increases, then a tighter margin is obtained, and if it decreases, then the margin is maximized between two classes, which leads to the aim of SVM.

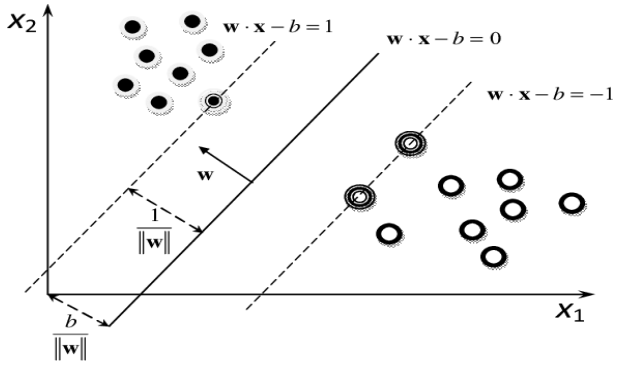


Figure 1: The maximum-margin hyperplane separating samples from two classes

3.3.3 Naïve Bayes

Naïve Bayes is a supervised learning algorithm based on Bayes theorem and used for classification problems. It is a classifier which predicts on the basis of the probability of an object. Naïve Bayes model makes a prediction for new input using Bayes theorem which is used to determine the probability of hypothesis with prior knowledge. It uses the condition probability .

The formula for Bayes' theorem is given as:

$$P(y_j|x_i) = \frac{P(x_i|y_j)P(y_j)}{P(x_i)}, \quad i = 1,2,3, \dots \dots . n,$$

Equation 3-4

where:

$P(y_j|x_i)$: is a posterior probability, the probability of event y_j (hypothesis) given event x_i (prior knowledge),

$P(x_i|y_j)$: is a likelihood probability, probability of the evidence given that the probability of hypothesis is true.

$P(y_j)$: is a prior probability, probability of hypothesis before observing the evidence

$P(x_i)$: is a marginal probability, probability of evidence.

We assume that a dataset contains n instances x_i, $i = 1,2,3, \dots . . n$ which consist of p variables, i.e., $x_i = x_{i1}, x_{i2}, x_{i3}, \dots \dots x_{ip}$. Each instance is assumed to belong to one and only one class, $y \in \{y_1, y_2, y_3, \dots . y_c\}$.

To simplify the notation assume that $X = x_i$, and $Y = y_j$. We obtain:

$$P(y_j|X) = \frac{P(x_i|y_j)P(y_j)}{P(X)}, \quad i = 1,2,3, \dots \dots . n.$$

Equation 3-5

This yields:

$$P(y_j|X) = \frac{P(y_j)\prod_{i=1}^{n} P(x_i|y_j)}{P(X)}.$$

Equation 3-6

When we are interested in a crisp classification rule, which is a rule that assigns each instance to exact one class, then we calculate the value of the numerator in Equation 3-6 for each class and select that class for which this value is maximal. This rule is called maximum a posterior

rule. The resulting “winning” class is known as Maximum A Posteriori (MAP) class, is calculated as:

$$\hat{y} = \underset{y_j}{\operatorname{argmax}} \prod_{1}^{k} P(x_k|y_j)\, P(y_j).$$

Equation 3-7

A model that implements Equation 3-7 is called a Naïve Bayes classifier.

3.3.4 Decision Tree

A decision tree is a classifier represented by the instance space as a recursive partition [21]. This classifier consists of nodes that form a rooted tree, meaning that it is a directed tree that has no incoming edges with a node called “root”. The remaining nodes have only single input edge. A node with outgoing edges is called an internal or test node. These nodes are called leaves (recognized as terminal or decision nodes). In a decision tree, each inner node splits the instance space into two or more sub-spaces with a certain distinct function of the input variable. Mostly in the simple and most commonly used instance, every test recognizes one attribute, such that the instance space is divided up as per attribute values.
The condition refers to a set, in the case of numeric attributes. Each leaf is assigned to one class which represents the most suitable target. Alternatively, a probability vector may be carried by the leaf indicating the probability that the target attribute has a certain value. Instances are classified by navigating them down to a leaf from the root of the tree, according to the result of the tests along the path.

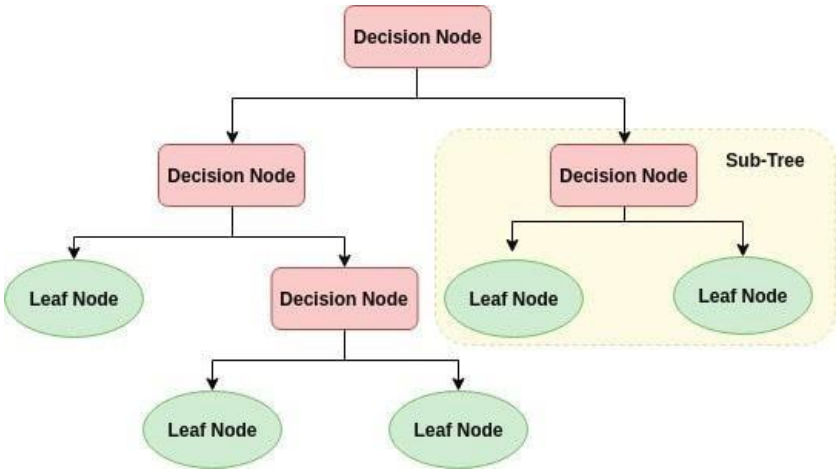


Figure 2: Decision tree algorithm

For example, Figure 2 describes a decision tree which explains whether a potential customer will respond to a direct mailing or not. As circles, internal nodes are represented, while leaves are denoted as triangles.

In view of this classifier, the analyst will predict a potential customer's response (by sorting it down the tree), and recognize the behavioural characteristics of the entire population of potential customers about direct mailing. Each node is labelled with the attribute that it checks, and its branches are labelled with the values that correspond.

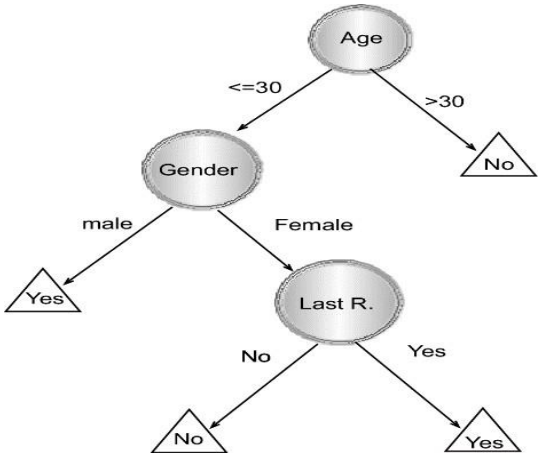


Figure 3: Classify response to direct mailing using decision tree

In the situation of numerical attributes, this classifier is represented geometrically as a set of hyperplanes, each orthogonal to one axis. Decision-makers naturally prefer less complicated decision trees, since they can be considered more comprehensible. In addition, according to Breiman et al. "the complexity of the tree has a critical impact on its precision"[18].

The complexity of the tree is managed directly by the stopping criteria used and the method of pruning used. Typically the complexity of the tree is calculated by using one of the following parameters: total number of nodes, total number of leaves, depth of the tree and number of attributes used. The induction of a decision tree is closely related to the induction of laws.

Each path can be translated into a law from the root of a decision tree to one of its leaves simply by joining the tests along the path to form the antecedent component and taking the class prediction of the leaf as the class value.

For example, Figure 3 can be translated into the rule: "If the customer's age is less than or equal to or equal to 30 and the customer's gender is" Male-then the "customer will reply to the mail." The resulting rule set can then be streamlined to improve a human user's understandability, and probably its accuracy [19].

The major challenge in the Decision Tree is to define the attribute for the root node at each step. This process is called selection of the attributes. We have two common selection steps for the attributes: Information gain; Gini index.

Information gain: Denoted by $Gain(S, A)$, it measures the relative change in entropy with respect to independent variables. $Entropy$ is defined as the amount of uncertainty or randomness in data. The higher the entropy the more content of the knowledge.

$Entropy(S) = \sum_{xsX} P(x) \log_2 \frac{1}{P(x)}$.

Equation 3-8

This yield:

$Gain(S, A) = Entropy(S) - \sum_{vgValue(A)} \frac{|S_v|}{|S|} . Entropy(S_v).$

Equation 3-9

Where

$P(x)$: is the probability of event x;

S : is a set of instances;

A : is an attribute;

S_v : is a subset of S with $A = v$, and

$Value(A)$: is a set of all possible values of A.

Gini index: Gini Index is a metric for calculating how frequently an element picked at random will be incorrectly identified. It means you should choose an attribute with lower Gini index.

Sklearn supports Gini Index parameters with “Gini”, and it takes “gini” value by default. The formula for the Gini Index calculation is given below.

$Gini\ Index = 1 - \sum_j P_j^2.$

Equation 3-10

3.3.5 Random Forest

Random forest was introduced by an ear-inspired Leo Breiman-Amit and Geman's Lier function [22]. , random forest is an evolution of Breiman’s idea and has been built as a boosting competitor. Random forest may be used either for a categorical data, recognized as "classification," or a continuous data, known as "regression"[23]. Random forest algorithm as the name implies, is based on assemble of trees, based on a set of random variables for each tree. More formally, for a p-dimensional random vector $X = (X_1, \dots\dots, X_p)^T$representing the

real-valued input or predictor, Variables and a random variable Y representing the real-valued answer, we assume an unknown joint distribution $P_{XY}(X,Y)$ to find out a predictive function $f(X)$ to predict Y. The prediction function is given by a loss function $L(Y,f(X))$ And defined to minimize the expected value of the loss. $E_{XY}\ (L(Y,f(X)))$ Where the subscripts denote expectation with respect to the joint distribution of X and Y Intuitively, $L(Y,f(X))$ is a measure of how close $f(X)$ is to Y; it penalizes values of $f(X)$ that are a long way from Y. Typical choices of L are squared error loss for regression: $L(Y,f(X)) = (Y - f(X)\ )^2$.

Equation 3-11

And zero-one loss for classification:

$$L(Y,f(X)) = I((Y \neq f(X)\ ))\begin{cases} 0\ if\ Y = f(X) \\ 1\ otherwise \end{cases}.$$

Equation 3-12

For regression function minimizing $E_{XY}\ (L(Y,f(X)))$ gives

$$f(x) = E(Y|X = x)$$

Equation 3-13

For classification case, if the set of possible Y variable is presented by φ minimizing $E_{XY}\ (L(Y,f(X)))$ zero-one loss gives:

$$f(x) = arg\max_{y s \varphi} P(Y = y|X = x)$$

Equation 3-14

Otherwise known as the Bayes rule.

Ensembles construct f in terms of a collection of so-called "base learners" $h_1(x)$, ... , $h_j(x)$ and these base learners are combined to give the "ensemble predictor" $f(x)$. In regression, the base learners are averaged

$$f(x) = \frac{1}{j}\sum_{j=1}^{j} h\ \ (x).$$

Equation 3-15

Although $f(x)$is the most generally predicted class in classification.

$$f(x) = arg\max_{y s \varphi} \sum_{j=1}^{j} I(y = h_x(x)).$$

Equation 3-16

The jth base learner in random forests is a tree represented by $h_j(X, \Theta_j)$ where Θ_j is a set of random variables and the elements of Θ_j are independent for j = 1... J. Although the concept of a random forest is very general, it is almost always applied in a specific way. To understand the random forest algorithm, it is important to have a basic knowledge of the type of trees used as base learners.

3.3.6 Parameter tuning

The parameter tuning sets the parameter values prior to the training process. Parameter tuning in machine learning is equally important as data cleaning and feature extraction. Tuning parameter is very important and interesting, is very sensitive to a small change in the rate of learning or the estimators will lead a big change in model accuracy [24]. We have two methods for parameter tuning, namely Grid search and Random search.

Grid Search

Grid Search is the best way for parameter tuning to make the best score. This is done by joining the values and search for the best ranking and find out the best combination. The classic approach for parameter tuning is grid scan. By joining the values, we make a grid and search for the best ranking, and find the best combination [24].

Grid Search often results in an optimal solution, but it takes time and expensive computation due to the large combination, which makes it costly.

Random Search

Random Search selects the parameter values randomly and checks for the best score. Number of iteration is significant in parameter tuning which defines the number of searches that will occur during the process [24].

Random Search is not time consuming as Grid Search, requires less computing power but does not guarantee the best model.

3.3.7 Metrics to evaluate machine learning algorithms

Evaluating the performance of a machine learning algorithm is considered as an integral aspect of every project. In this study we discuss different ways of testing the efficiency of our machine learning models and why to use one instead of the other. The choice of metrics used to measure and compare the performance of machine learning algorithms depends largely on the nature of the data (balanced or unbalanced) [25]. Much of the time we used to measure our model's efficiency, but it's not enough to really assess our model. We employed various assessment criteria in this study to measure the performance of our model. Specific metrics

used are: Classification accuracy; Confusion matrix; Precision; Recall; F1 score; and are under Curve (AUC).

In accuracy measurement, the concept of true positive (TP); false positive (FP); true negative (TN) and false negative (FN) readily come handy.

a) True positives are classes predicted positive and they are really positive
b) True negative are classes predicted negative and are really negative
c) False positive are classes predicted positive and they are really negative
d) False negative are classes predicted negative and they are really positive

Classification accuracy

Classification accuracy is a ratio of the correctly predicted data points to the total cases in the dataset. It measures how well the model has predicted data instances as belonging to their respective classes. For classification problems, this is the most common assessment metric.

But it is really only acceptable when there is an approximately equal number of findings in each class (which is seldom the case) and both classes are equally likely to be present. However, this is not frequently the case. Mathematically, a model's accuracy measurement is calculated using the formula below

$$Accuracy = \frac{Number\ of\ corrected\ pridictions}{The\ total\ number\ of\ pridictions\ made}.$$

Equation 3-17

Accuracy can be misleading in the case of data imbalance.

Example: Consider a dataset which comprises 100 records of two classes A and B, with class A making up 80% of the entire data and class B 20%. Let us assume that we are interested in predicting the outcome of the minority class B. Suppose we build a model which gave the following outcomes:

TP = 5 that is 5 records of the B class were correctly predicted as belonging to B;

FP = 10 that is 10 records of the A class were predicted as belonging to B;

TN = 70 - that is 70 records belonging to class A were predicted as class A;

FN = 15 - that is 15 records belonging to class B were predicted as A.

The accuracy of the model would be given by:

Accuracy = (70 + 5)/ (80 +20) = 0.75 or 75%. We see that the model's accuracy of 75% is "good". However, a closer look at this model shows that we are able to perform correct prediction of our class of interest only 5/20 or 25% of the times. This model does not really

meet our expectations since 75% of the times class B is misclassified. A good metric to focus on and improve in order to have a better model in the case of data imbalance would be "recall", defined below.

Confusion matrix

A confusion/uncertainty matrix is a matrix or table that can be used, usually in supervised learning, to calculate the output of a machine learning algorithm. Each row in the uncertainty matrix represents the real class instances, and each column represents the projected class instances [26].

For convenience, we're mainly going to address stuff in terms of a binary classification issue where we're going to have to figure out whether investigation is a positive (accident) or a negative. Or an incident has had fatal (positive) or non-fatal (negative) effects.

From our issue about classification, we have samples from two classes: Positive (1) or Negative (0).

Predicted / Actual	Negative (0)	Positive (1)
Negative(0)	TN	FP
Positive(1)	FN	TP

Table 1: Confusion matrix

Accuracy

The accuracy is determined by the average of the values appeared in the "main diagonal"

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN}.$$

Equation 3-18

Accuracy is the most widely used metric for judging a model but, as already shown, it is not a good performance measure in case of data imbalance.

Precision

The precision is defined by proportion of true positive out of the total forecasted positive. The denominator defines the model prediction of the entire dataset rendered as positive.

Calculating means how much the model predicted correctly and it was really correct.

$$precision = \frac{TP}{TP+FP}.$$

Equation 3-19

Recall

Recall is determined by the percentage of true positive out of the overall real positive instances in a dataset. Thus in Equation 3-20, the expression (TP + FN) represents the actual number of positive instances in the dataset. TP is the actual positive cases (accidents) in the dataset that were predicted as positive by the model. FN represents positive cases which the algorithm incorrectly predicted as negative, that is, for these accident cases, the algorithm predicted them incident.

$$recall = \frac{TP}{TP+FN} .$$

Equation 3-20

Specificity

Specificity is determined by the proportion of negative instances out of the total real negative instances. Thus the denominator (*TN + FP)* is the *actual* number of negative instances in the dataset. It's like a recall but the shift is on the negative instances.

$$specificity = \frac{TN}{TN+FP}$$

Equation 3-21

F1 score

F1 score is the harmonic mean of recall and precision. See that if one goes low, the final F1 score drops considerably because of the item in the numerator. So a model performs well in F1 score if the expected positive is actually positive (precision) and does not miss out on positive ones and predicts negative ones (recall).

$$\frac{2}{\frac{1}{precision}+\frac{1}{recall}} = \frac{2*precision*recall}{precision+recall}.$$

Equation 3-22

One downside is that both precision and recall are given equal importance, so we may need one higher than the other according to our application, and the F1 score may not be the exact metric for it. So either a weighted-F1 score or a view of the PR or ROC curve will help.

AUC-ROC

AUC-ROC also written as AUROC (Area Under the Receiver Operating Characteristics), is a performance measurement used for classification problems. Mostly is used when we need to visualize the performance of classification model or models. ROC is a probability curve that plot true positive rate (TPR) false positive rate (FPR) and separate signals from the noise.

AUC explains how much classification model is capable of distinguishing between classes. In our study for example, the higher the AUC the better the model is at predicting accident (1) as accident (1) and incident (0) as incident (0) [27].

3.4 Description of dataset

The aviation accident/incident dataset is obtained from the online data source for national tra nsportation safety board (NTSB) [28] .

This data set comprises 84262 records (rows) and 31 attributes (columns) of accidents and in cidents that occurred during the period 1962 to 2019. Initial preprocessing of the data resulted in the identification of 5 suitable attributes for further analysis. The attributes selected for ana lysis are *"investigation type", "amateur built", "weather condition", "phase of flight", and "number of engines".* The dependent variable is *"investigation type"* and comprises of two cl asses: 1 representing a data record corresponding to an aircraft accident and 0 for a data recor d corresponding to an aircraft incident. The accident/incident data is illustrated in Table 2, an d Table 8 summarizes all attributes from the original dataset.

Attributes	Attribute values	Code	Total	Null values
Investigation Type	Accident	1	80814	5
	Incident	0	3443	
Amateur Built	Yes	1	8083	
	No	0	75570	609
Weather Condition	VMC	1	78223	3103
	IMC	0	6039	
Phase of Flight	Approach	5	8163	
	Climb	3	2361	
	Cruise	7	11134	
	Descent	2	2305	
	Go-around	1	1696	
	Landing	10	20625	6608
	Maneuvering	6	10367	
	Others	10	168	
	Standing	0	1299	
	Take-off	9	16272	
	Taxi	4	2471	

	Unknown	10	793	
Number of Engines	00	0	1176	5034
	01	1	66570	
	02	2	10576	
	03	3	481	
	04	4	424	
	08	8	01	

Table 2: Summary of our dataset

3.4.1 Exploratory data analysis

Through analysis, truth present in data is found. Hidden facts are obtained when proper analysis is performed. The outcome of analysis is often a report that summarizes findings. In this project we used Python programming language and employed machine learning algorithms to perform data analysis.

With python we:

- Pre-processed the data and made it ready for analysis. This involved handling null values and feature selection based on their importance;
- Performed statistical analysis;
- Visualized insights using charts and graphs; and
- Trained and validated models to predict outcomes for new data records.

3.4.2 Data pre-processing

The importance of the quality of research data cannot be over emphasized. Pre-processing data is key as its quality needs to be guaranteed in order to arrive at meaningful results. We started by conducting data pre-processing as the first step after obtaining the dataset. The raw data was transformed into a form that can be easily and accurately analysed. Proper data preparation allows for efficient analysis. It can eliminate errors and inaccuracies that could have occurred during the data gathering process. Thus, can help in removing some bias resulting from poor data quality. Python libraries - Pandas and Numpy - were used for data preparation. After importing the data into the working environment, we tried to have general understanding of it, visualizing different insights. Cleaning was then done and high - quality data obtained for further analysis.

3.4.3 Import Python libraries for data analysis.

The dataset used in this project which is in comma separated values (csv) format was loaded and manipulated using Python programming language. Figure 4 shows code snippet used in this process. The figure contains Python libraries used in loading data, performing data cleaning, data exploration and visualization, and to build machine learning models. To get the data loaded into our working environment, pandas' function "read_csv" was used. This function imports data into a data frame - a two dimensional table where columns represent the data attributes and each row stands for a complete data record – presenting the data in an easy to manipulate way.

```
import numpy as np
import pandas as pd
import scipy as sp
import seaborn as sns
#import data processing
from sklearn.model_selection import train_test_split

#import models
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier as KNN
from sklearn.tree import DecisionTreeClassifier as DecisionTree
from sklearn.ensemble import RandomForestClassifier as RandomForest
from sklearn import svm

#import metrics
from sklearn import metrics
from sklearn.metrics import confusion_matrix,recall_score,precision_score,f1_score,roc_auc_score,accuracy_score, classificatic
from sklearn.model_selection import GridSearchCV

#import visualization libraries
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
```

Figure 4: Python libraries for data analysis

3.4.4 Data cleaning

Data cleaning is very important part of any data analysis project that involves machine learning [29]. The performance of developed models is greatly affected by the quality of the data used. During research, errors which may come from instruments used or from the operator of the instruments often lead to messy data. In order to obtain reliable results, the quality of the data needs to be assured before it is used.

Data cleaning refers to identifying and correcting errors in the dataset to ensure the quality of the data. In our data cleaning process, we performed the following tasks:

- Dropping unnecessary columns in a DataFrame. In machine learning, there exist the concept of curse of dimensionality where data with higher dimensions greatly affect the computational time of an algorithm. Dropping unnecessary columns of a dataset reduces the dimensionality and noise, thus improving the data quality. The approach used for feature selection is described in detail in a following section.

- Handling missing values: It is not uncommon to find missing values in a dataset. These could be as a result of non-respondent or omission to read an instrument at a set time. During the data cleaning process, we identified these values for each attribute and used insight found in the data to handle same.
- Renaming columns to a more recognizable set of labels.

3.4.5 Handling missing values using pandas.

In real life situations, missing data is always a problem. In machine learning model development, predictive models' accuracy are greatly affected by poor quality of data. It goes without saying that handling missing values which mess up data becomes a key component of machine learning model development. Missing data can arise in the dataset due to multiple reasons: the data for the specific field was not added by the user/data collection application; data loss which may occur while transferring data manually; a programming error, etc.

Checking for missing values in dataset.

This process involves navigating through each attribute of dataset, identifying and processing missing values encountered. Missing values for each attribute are treated independently using insight found in data. Depending on the percentage of missing valuesfor each attribute, we either decided to eliminate the attribute (in case where percentage of missing value was too large – greater than 50%) or they were handled based on the distribution of the non-null values.

The first thing we did when cleaning our data was to check missing values for every attributes. This guided us on how to proceed with feature selection. We used "isna()" and "sum()" functions of pandas DataFrame to count the missing values present in each attributethus: DataFrame.isna().sum().

In the dataset used, 5 features out of 31 have more than half of their values being nulls. Figure 5 presents a summary of the attributes and the count of the missing values in our dataset.

```
In [13]:  1  print(data.isna().sum())
```

```
Event.Id                    0
Investigation_Type          5
Accident.Number             0
Event.Date                  0
Location                   77
Country                   507
Latitude                54099
Longitude               54108
Airport.Code            36539
Airport.Name            33823
Injury_Severity             0
Aircraft_Damage          2709
Aircraft_Category       56754
Registration.Number      3850
Make                       70
Model                      99
Amateur_Built             609
Number_of_Engines        5034
Engine_Type              4256
FAR.Description         57056
Schedule                72419
Purpose_of_Flight        4880
Air.Carrier             80093
Total_Fatal_Injuries    27203
Total_Serious_Injuries  29878
Total_Minor_Injuries    28670
Total_Uninjured         14589
Weather_Condition        3103
Phase_of_Flight          6608
Report.Status               0
Publication.Date        14315
dtype: int64
```

Figure 5: Summary of missing values in our dataset

In Figure 6, a heat map is used to visualize the missing values. In the figure, areas with green colour represent the presence of data points while the areas with blue show missing values. As mentioned earlier, we can see from the heat map that six columns have more than half of their values being nulls. We rejected these attributes in a bit to improve on our data quality.

```
In [12]:  1  #for the heatmap blue indicates the columns with the NaN values,
          2  sns.heatmap(data.isnull(),cmap="winter_r")

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x169115ab7b8>
```

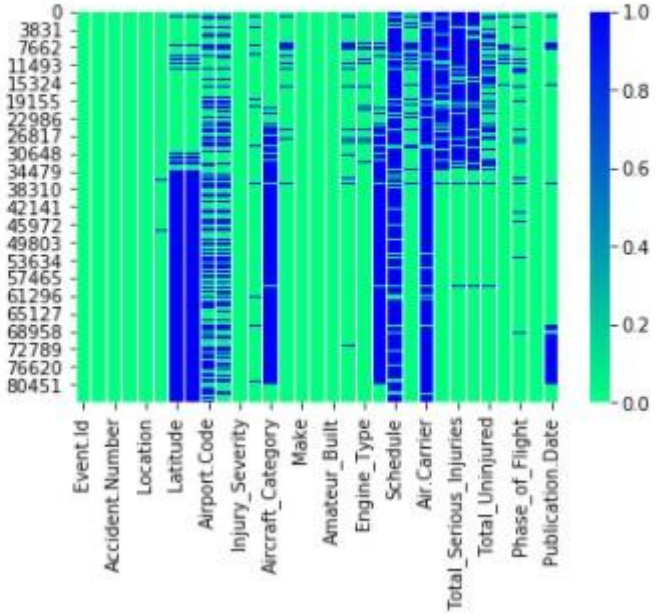


Figure 6: Heat map function represents distribution of null values

Filling in missing data

We used different methods to fill missing values within columns according to their nature. In our first approach, we replaced missing values with the most frequent (mode) value in the column. The second is to fill the missing values with the next or previous value (fill forward or fill backward).

Data cleaning focused on five attributes of the dataset which are investigation type (target variable), amateur built, weather condition, phase of flight, and number of engines. The 4 key independent attributes were selected for further analysis based on different criteria including expert knowledge; number of null values in attribute and complexity of categorical attribute composition.

a. Table 3 summarizes the investigation type which is our dependent variable with two values: accident and incident, and null values.

Code	Values	Total
01	Accident	80814
02	Incident	3443
03	Nan	5

Table 3: Summary of our dependent variable (investigation type)

Table 3 indicates that accident – represented by 1 as a state - is 96% of all values in investigation type column. Incident makes up only 3.94% with 5 missing values (making up 0.06%). Based on this modal value for the attribute, we replaced nulls with the value representing accident. Figure 7 shows the code snippet used to achieve this processing.

```
[3]: #data.Investigation_Type
investigation=data.groupby('Investigation_Type' ).size().reset_index(name="Total")
investigation
```

t[3]:

	Investigation_Type	Total
0	Accident	80814
1	Incident	3443

```
[4]: data.Investigation_Type.describe()
```

```
t[4]: count        84257
      unique           2
      top       Accident
      freq         80814
      Name: Investigation_Type, dtype: object
```

```
1 data31=data.Investigation_Type
2 NaN=data31.isna().sum().sum()
3 print("The Total number of NaN are",NaN)
```

```
The Total number of NaN are 5
```

```
In [26]: # missing value cells are filled with the highest value cell
         new_data2 = data2.fillna('Accident')
         new_data2.unique()

Out[26]: array(['Accident', 'Incident'], dtype=object)
```

```
In [27]: #new_data2=data1.Investigation_Type
         NaN=new_data2.isna().sum().sum()
         print("The Total number of NaN are",NaN)

         The Total number of NaN are 0
```

Figure 7: Python codes illustrating investigation type

b. Table 4 summarizes the amateur built as one of independent variable with two values: 'Yes' and 'No', and null values.

Code	Values	Total
01	Yes	8083
02	No	75570
03	Nan	609

Table 4: Summary of amateur built

The above table 4 indicates that ‘No’ – represented by ‘0’ as a state - is 89.7% of all values in amateur built column. ‘Yes’ makes up only 9.6 and missing values (making up 0.7%). Based on this modal value for the attribute, we replaced nulls with the value representing ‘No’. Figure 8 shows the code snippet used to achieve this processing.

```
In [29]:
1 amateur=data.groupby('Amateur_Built' ).size().reset_index(name="Total")
2 amateur
```

Out[29]:

	Amateur_Built	Total
0	No	75570
1	Yes	8083

```
In [28]:
1 data3=data1.Amateur_Built
2 NaN=data3.isna().sum().sum()
3 print("The Total number of NaN are",NaN)

The Total number of NaN are 609
```

```
In [3]:
1 data.Amateur_Built.describe()

Out[3]: count     83653
        unique        2
        top          No
        freq      75570
        Name: Amateur_Built, dtype: object
```

Figure 8: Python codes illustrating amateur built

c. Table 5 summarizes the phase of flight as one of independent variable with 10 values, unknowns and null values.

Code	**Values**	**Total**
01	Approach	8163
02	Climb	2361
03	Cruise	11134
04	Descent	2305
05	Go-around	1696
06	Landing	20625
07	Manoeuvring	10367
08	Other	168
09	Standing	1299
10	Take-off	16272

11	Taxi	2471
12	Unknown	793
13	Nan	6608

Table 5: Summary of phase of flight

Phase of flight has 10 values with unknowns caused by writing errors or other reasons appeared in recording. Table 5 indicates that 'Landing' represented by '9' is the most frequent phase with 24% of all phases appeared in this column and missing values (making up 7.8%). Based on this modal value for the attribute, we replaced nulls with the value representing 'Landing'. Figure 9 shows the code snippet used to achieve this processing.

```
In [5]:  1 data.Phase_of_Flight.describe()

Out[5]: count       77654
        unique         12
        top       LANDING
        freq         20625
        Name: Phase_of_Flight, dtype: object
```

```
In [33]:  1 Phase_of_Flight=data.groupby('Phase_of_Flight' ).size().reset_index(nam
          2 Phase_of_Flight
```

Out[33]:

	Phase_of_Flight	Total
0	APPROACH	8163
1	CLIMB	2361
2	CRUISE	11134
3	DESCENT	2305
4	GO-AROUND	1696
5	LANDING	20625
6	MANEUVERING	10367
7	OTHER	168
8	STANDING	1299
9	TAKEOFF	16272
10	TAXI	2471
11	UNKNOWN	793

```
In [7]:  1 data32=data.Phase_of_Flight
         2 NaN=data32.isna().sum().sum()
         3 print("The Total number of NaN are",NaN)

         The Total number of NaN are 6608
```

Figure 9: Python codes illustrating phase of flight

d. Table 6 summarizes the number of engines as one of independent variable with 6 values, and null values and all values are numerical.

Code	Values	Total
01	00	1176
02	01	66570
03	02	10576
04	03	481

05	04	424
06	08	01
07	Nan	5034

Table 6: Summary of number of engines

Number of engines has 6 values. The above table 6 indicates that '1' is the most frequent with 79% of all values in number of engines column and missing values (making up 6%). Based on this modal value for the attribute, we replaced nulls with '1'represennting aircraft with one engine. Figure 10 shows the code snippet used to achieve this processing.

```
In [32]:
1 Number_of_Engines=data.groupby('Number_of_Engines' ).size().reset_index(name="Total")
2 Number_of_Engines
```

Out[32]:

	Number_of_Engines	Total
0	0.0	1176
1	1.0	66570
2	2.0	10576
3	3.0	481
4	4.0	424
5	8.0	1

```
In [4]:
1 data33=data.Number_of_Engines
2 NaN=data33.isna().sum().sum()
3 print("The Total number of NaN are",NaN)
```

```
The Total number of NaN are 5034
```

Figure 10: Python codes illustrating variable number of engines

For these faux attributes we used the same method, to replace null value with the most frequent, using fillna() function and we combined all of them for the replacement. Figure 11 Shows the code snippet used to achieve that processing.

```
In [34]:
new_data1 = data1.fillna(
                {'Investigation_Type':'Accident',
                 'Amateur_Built':'No',
                 'Make':'CESSNA','Phase_of_Flight':'LANDING','Number_of_Engines':1})
new_data1.head()
```

Out[34]:

	Investigation_Type	Amateur_Built	Make	Weather_Condition	Phase_of_Flight	Number_of_Engines
0	Accident	No	Cessna	VMC	TAKEOFF	1.0
1	Accident	No	Mooney	VMC	TAKEOFF	1.0
2	Accident	No	Cessna	NaN	LANDING	1.0
3	Accident	Yes	Vans	NaN	LANDING	1.0
4	Accident	No	Cessna	NaN	LANDING	1.0

```
In [35]:
print(new_data1.isna().sum())
```

```
Investigation_Type       0
Amateur_Built            0
Make                     0
Weather_Condition     3103
Phase_of_Flight          0
Number_of_Engines        0
dtype: int64
```

Figure 11: Python code replacing missing values with the modal value for the four attributes and the result

e. Table 7 summarizes the rest attribute, weather condition as one of independent variable with two values: 'VMC' and 'IMC' and null values. We replaced the null value with the previous values or next value, using fill forward method, with limitation of two null variables next to the real value or fill backward method, with limitation of two on one event and limitation of one for the last event to a fully cleaned dataset. Because usually the next or previous weather condition is similar to the current weather. Figure 12 shows the code snippet used to achieve this processing.

Code	Values	Total
01	VMC	78223
02	IMC	6039
03	Nan	3103

Table 7: Summary of weather condition

```
In [36]:
data_new = new_data1.fillna(method='ffill',limit=2)
data_new = data_new.fillna(method='bfill',limit=2)
```

```
In [37]:
data_new = data_new.fillna(method='bfill')
```

```
In [38]:
print(data_new.isna().sum())
```

```
Investigation_Type    0
Amateur_Built         0
Make                  0
Weather_Condition     0
Phase_of_Flight       0
Number_of_Engines     0
dtype: int64
```

```
In [39]:
data_new.head()
```

Out[39]:

	Investigation_Type	Amateur_Built	Make	Weather_Condition	Phase_of_Flight	Number_of_Engines
0	Accident	No	Cessna	VMC	TAKEOFF	1.0
1	Accident	No	Mooney	VMC	TAKEOFF	1.0
2	Accident	No	Cessna	VMC	LANDING	1.0
3	Accident	Yes	Vans	VMC	LANDING	1.0
4	Accident	No	Cessna	IMC	LANDING	1.0

Figure 12: Dataset after cleaning, without null value

3.4.6 Data transformation

Data transformation is defined as the mechanism by which data is transferred from one format to another. Usually from source format to the required format, so that thedata are readable easily by the preferred tool [30] . The choice of algorithms used in our research could perform better with numerical data. For this reason, we performeddata transformation in which all categorical variables were converted to numerical variables. Initially, the 5 selected variables, four were categorical with only one having numeric values. The four categorical variables were transformed to numericalvariables with each category/label assigned a number. In the following section, the actual transformations for each data attribute are described.

- ✓ For “investigation type”, we have two values in the form of string, ‘accident’ and ‘incident’. These were converted into numeric with ‘accident’ replaced by ‘1’ and ‘incident’ by ‘0’.

```
In [46]:
data_new.Investigation_Type = data_new.Investigation_Type.apply(lambda x: 1 if(x=='Accident') else 0)
```

Figure 13: Python code for transformation of investigation type from categorical to numerical

```
In [71]:  1  data_new.Investigation_Type.value_counts()

Out[71]: 1    80819
         0     3443
         Name: Investigation_Type, dtype: int64
```

Figure 14: Summary of investigation type after transformation from categorical to numerical

- ✓ The weather condition originally comprises of two categories: 'VMC' and 'IMC'. 'VMC' has been converted to '1' and 'IMC' to '0'.

```
In [20]:  1  data_new.Weather_Condition.value_counts()

Out[20]: 1    78223
         0     6039
         Name: Weather_Condition, dtype: int64
```

Figure 15: Summary of weather condition after transformation from categorical to numerical

- ✓ For Amateur built has two values in form of string, 'Yes' and 'No'. 'Yes' is represented by '1' and 'No' represented by '0'.

```
[18]:  1  data_new.Amateur_Built.value_counts()

t[18]: 0    76179
       1     8083
       Name: Amateur_Built, dtype: int64
```

Figure 16: Summary of amateur built after transformation from categorical to numerical

- ✓ For phase of flight has 10 values in form of string: Standing, Go-around, Descent, Climb, Taxi, Approach, Manoeuvring, Cruise, Take-off and Unknown. We converted this way standing to 0, Go-around to 1, Descent to 2, Climb to 3, Taxi to 4, Approach to 5, Manoeuvring to 6, Cruise to 7, Take-off to 8 and Landing to 9.

```
In [44]:  1  data_new.Phase_of_Flight.value_counts()

Out[44]: 9    28194
         8    16272
         7    11134
         6    10367
         5     8163
         4     2471
         3     2361
         2     2305
         1     1696
         0     1299
         Name: Phase_of_Flight, dtype: int64
```

Figure 17: Summary of phase of flight after transformation from categorical to numerical

✓ For number of engines has 6 values and all of them are numeric.

```
In [32]:
1 Number_of_Engines=data.groupby('Number_of_Engines' ).size().reset_index(name="Total")
2 Number_of_Engines
```

Out[32]:

	Number_of_Engines	Total
0	0.0	1176
1	1.0	66570
2	2.0	10576
3	3.0	481
4	4.0	424
5	8.0	1

Figure 18: Summary of number of engine with numerical values

3.4.7 Feature selection

For a given dataset, the contribution of independent variables to the dependent variable differs from one variable to another. Some attributes have a more predominant effect on the target than others. In feature selection, we aim at reducing the dimensionality of the dataset by picking out the independent variables that have more effect.

The method of choosing the most appropriate features from a given dataset is Feature Selection. Feature selection can increase the performance of a machine learning model in many cases, as this reduces noise found in the data, and algorithm runtimes and resource utilization. Our original dataset has 31 features as indicated in Figure 20. Among these 31 features, some are less significant based on our research purpose. The purpose of this study is to predict aircraft incidents. As such, our focus on the dataset was to identify attributes that significantly contribute to the aircraft accident and incident. Based on knowledge gained as an expert in the aviation industry having worked there for 8 years, some of the features in the dataset were observed to be reporting after effects of aircraft accident and incident. After consulting with other experts in the field of aviation, we were able to classify the attributes in the original dataset into three classes: contributory factors, effects of incident/accident and insignificant factors as shown in Table 8.

Code	**Contributory**	**Effects**	**Insignificant**
01	Aircraft category	Aircraft damage	Event id
02	Make	Injury severity	Accident number
03	Model	Total fatal injuries	Event date
04	Amateur built	Total serious injuries	Country

05	Number of engines	Total minor injuries	Airport name
06	Engine type	Total uninjured	Airport code
07	Purpose of flight	Report status	Registration number
08	Weather condition		Far description
09	Phase of flight		Schedule
10	Location		Air carrier
11	Latitude		Publication date
12	Longitude		

Table 8: Summary of all features in our dataset

As it is shown in the Table 8, the contributory factors are 12. Aircraft category, Latitude and Longitude were eliminated because they have null values (missing values) more than a half of all values as indicated in Figure 5.

Correlation between variables

In this project we have been calculating correlation between variables to check statistical relationship between each two variables in our data frame, to see how values of one variable depend on the value of another variable.

To check the correlations between two of variables in our dataset we used Pearson's coefficient and correlation matrix for visualization.

As shown in Figure 19, the values are the coefficients of correlation between the variables defined by the respective rows and columns. A coefficient of correlation is a number which denotes the intensity of the relation between two variables.

We used the coefficient of Pearson, as shown in Figure 19, and it is defined as the covariance between two variables divided by the product of the two variables' standard deviations.

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sigma_X \sigma_Y}$$

Equation 3-23

Once the covariance between two variables is further defined as the 'expected product value of those two variables deviations from their respective means. The covariance formula will make it simpler.

$$Cov(X,Y) = E[(X - \mu_X)(Y - \mu_Y)]$$

Equation 3-24

From Equation 3-24 the formula for Pearson's correlation will become:

$$\rho(X,Y) = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

Equation 3-25

Always σ value is between -1 and +1.

Values close to +1 indicate a strong positive relationship between the two variables, while those close to -1 indicate a strong negative relationship between the two variables.

The values close to zero mean, indicate that there is insignificant relationship between the two variables.

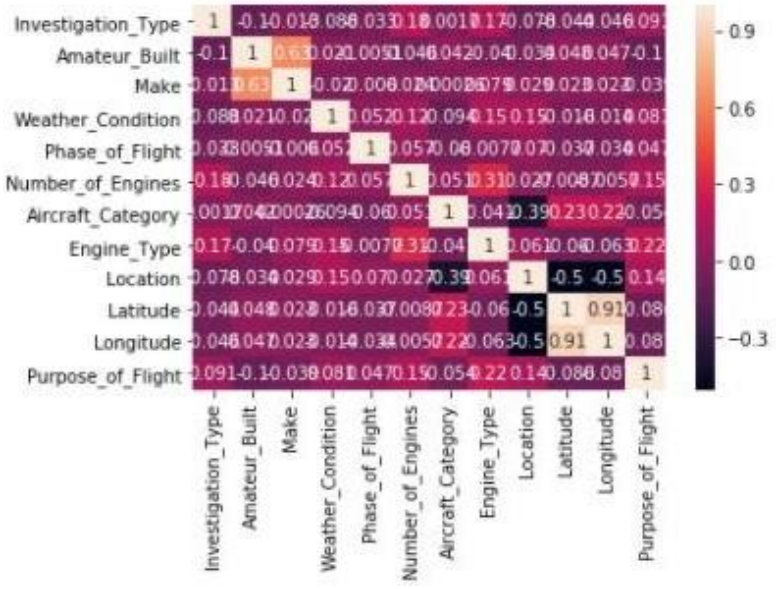


Figure 19: Summary of correlations between variables in our dataset.

Figure 19 summarise the correlation between all relevant variables to our target variable, it is shown that Amateur built and Make are highly positive correlated with correlation coefficient of 0.63. This led not to consider Make as our independent variable to our target for analysis. Based on the above feature selection techniques and based on knowledge gathered through in depth discussion with experts, we narrowed down the independent variables four: Amateur built, Number of engines, Weather condition, Engine type, Phase of flight, the target variable being Investigation type.

```
[4]:  1  data.shape
      2

[4]:  (84262, 31)
```

Figure 20: Shape of our dataset

```
In [3]: 1 data.columns

Out[3]: Index(['Event.Id', 'Investigation_Type', 'Accident.Number', 'Event.Date',
       'Location', 'Country', 'Latitude', 'Longitude', 'Airport.Code',
       'Airport.Name', 'Injury_Severity', 'Aircraft_Damage',
       'Aircraft_Category', 'Registration.Number', 'Make', 'Model',
       'Amateur_Built', 'Number_of_Engines', 'Engine_Type', 'FAR.Description',
       'Schedule', 'Purpose_of_Flight', 'Air.Carrier', 'Total_Fatal_Injuries',
       'Total_Serious_Injuries', 'Total_Minor_Injuries', 'Total_Uninjured',
       'Weather_Condition', 'Phase_of_Flight', 'Report.Status',
       'Publication.Date'],
      dtype='object')
```

Figure 21: All variables in our dataset

```
In [3]: 1 data1=data[['Investigation_Type','Amateur_Built','Make','Weather_Condition','Phase_of_Flight','Number_of_Engines']]
        2 data1.head()
```

Out[3]:

	Investigation_Type	Amateur_Built	Make	Weather_Condition	Phase_of_Flight	Number_of_Engines
0	Accident	No	Cessna	VMC	TAKEOFF	1.0
1	Accident	No	Mooney	VMC	TAKEOFF	1.0
2	Accident	No	Cessna	NaN	NaN	1.0
3	Accident	Yes	Vans	NaN	NaN	NaN
4	NaN	No	Cessna	NaN	NaN	1.0

Figure 22: Filtered features (potential features contributory to A/c Accident)

4.0 Analysis and results

4.1 Introduction

Machine learning approaches were implemented to predict aircraft accident or incident, and to identify potential contributory factors in aviation accidents/incidents. Prediction models used include:

I. Decision Tree;
II. Random Forest;
III. SVM;
IV. Naive Bayes.

4.2 Feature importance

Feature importance defined as a technique that allocate a score to the independent variables based on how useful they are at predicting a target variable. The achieved scores will illustrate which feature may be more important to the objective. As it is shown in Figure 23 among 4 input features, the phase of flight is the most potential contributory feature (factor) to the aircraft accidents in our dataset.

Phase of flight	Number of engine	Weather condition	Amateur built
0.797	0.084	0.071	0.047

Table 9: Summary on how independent variable contribute to the target variable

```
In [68]: feature_important = model.get_booster().get_score(importance_type='weight')
         keys = list(feature_important.keys())
         values = list(feature_important.values())

         data21 = pd.DataFrame(data=values, index=keys, columns=["score"]).sort_values(by = "score", ascending=False)
         data21.plot(kind='bar')
Out[68]: <matplotlib.axes._subplots.AxesSubplot at 0x169117f2c88>
```

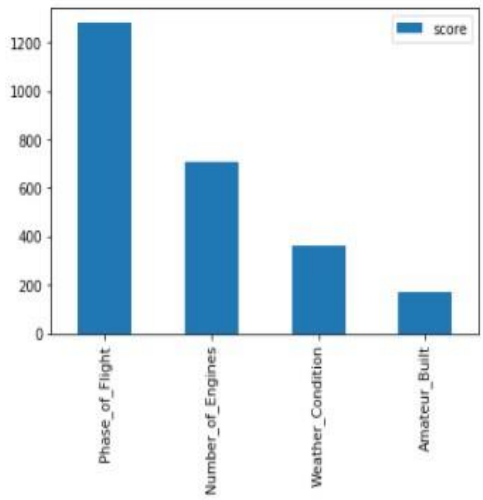


Figure 23: Feature importance

4.3 Direct classification analysis

We used four classification algorithms to predict if each record in our dataset represents a situation that led to an accident or not. The algorithms used are Support Vector Machine (SVM), Naïve Bayes, Random Forest, and Decision Tree. They have been evaluated based on their classification accuracies achieved and their performance based on AUC as can be seen in Table 10 and Figure 36.

classifier	Accuracy	AUC
Decision tree	95.81%	80%
Random forest	95.85%	80%
SVM	95.76%	56%
Naïve Bayes	93.23%	77%

Table 10: Summary of direct classified accuracy.

4.3.1 Decision Tree

Decision Tree classifier has been applied to classify data points on the basis of investigation type class (dependent variable) and this classifier classified data into 2 classes. The output of this classifier is determined on the basis of their precision, recall, F1 - score, area under curve (AUC). After training the model, we tested it on new data and it had an accuracy score of 96 %. Figure 24, provides a summary of the performance of the Decision Tree classifier. We observe that this model can correctly predict 96% of the unseen records that represent circumstances corresponding to an aircraft accident.

```
2]: 1 print ('Accuracy Score :',accuracy_score(y_test,y_dt))
    2 print(classification_report(y_dt,y_test))
```

```
Accuracy Score : 0.9581083486619593
             precision    recall  f1-score   support

          0       0.10      0.53      0.16       128
          1       1.00      0.96      0.98     16725

avg / total       0.99      0.96      0.97     16853
```

Figure 24: Classification report and accuracy score for Decision Tree classifier

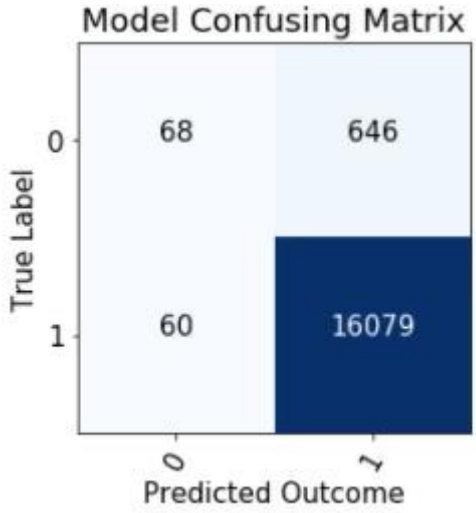


Figure 25: Confusion matrix for Decision Tree classifier

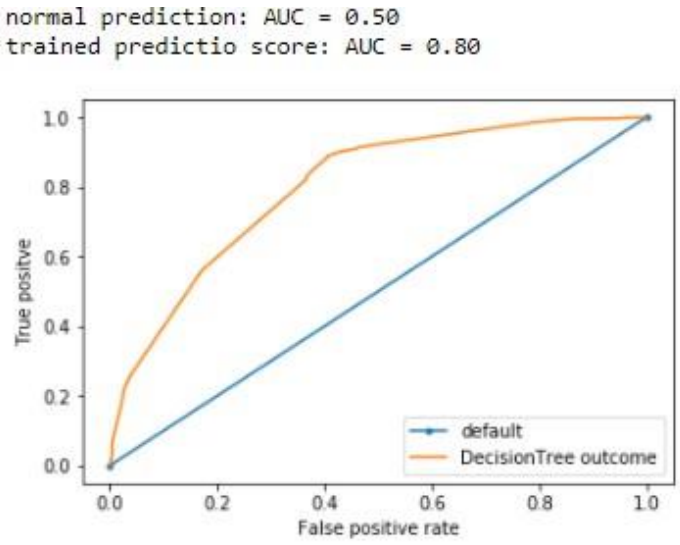


Figure 26: Graph representing performance of Decision Tree classifier based on area under curve

4.3.2 Random Forest

Random Forest classifier has been applied to classify data points on the basis of investigation type class (dependent variable) and this classifier classified data into 2 classes. The output of this classifier are determined on the basis of their precision, recall, f1-score, area under curve (AUC). After training the model, we tested it on new data and it had an accuracy score of 96 %. Figure 29 provides a summary of the performance of the Random forest classifier. We observe that this model can correctly predict 96% of the unseen records that represent circumstances corresponding to an aircraft accident.

```
n [33]:  1 print ('Accuracy Score :',accuracy_score(y_test,y_pred_rf ))
         2 print (classification_report(y_test,y_pred_rf))
```

```
Accuracy Score : 0.9585830415949682
             precision    recall  f1-score   support

          0       0.56      0.10      0.17       714
          1       0.96      1.00      0.98     16139

avg / total       0.94      0.96      0.94     16853
```

Figure 27: Classification report and accuracy score for Random Forest classifier

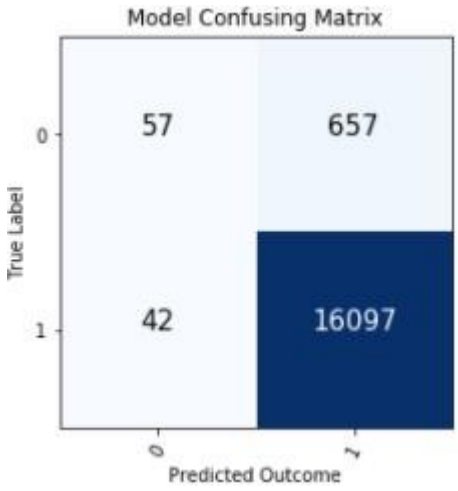


Figure 28: Confusion matrix for Random Forest classifier

```
normal prediction: AUC = 0.50
trained predictio score: AUC = 0.80
```

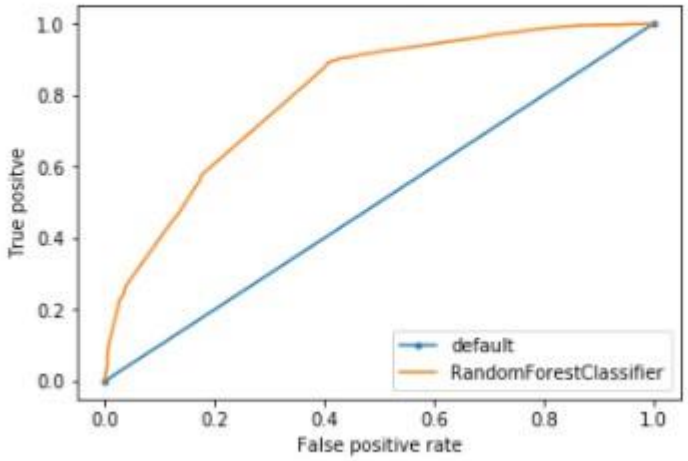


Figure 29: Graph representing performance of Random Forest classifier based on area under curve

4.3.3 Support Vector Machine (SVM)

SVM classifier has been applied to classify data points on the basis of investigation type class (dependent variable) and this classifier classified data into 2 classes. The output of this classifier are determined on the basis of their precision, recall, f1-score, area under curve (AUC). After training the model, we tested it on new data and it had an accuracy score of 96 % . Figure 32 provides a summary of the performance of the SVM classifier. We observe that this model can correctly predict 96% of the unseen records that represent circumstances corresponding to an aircraft accident.

```
In [26]:  1 print ('Accuracy Score :',accuracy_score(y_test,y_pred_svm ))
          2 print (classification_report(y_test,y_pred_svm))

Accuracy Score : 0.9576336557289503
             precision    recall  f1-score   support

          0       0.00      0.00      0.00       714
          1       0.96      1.00      0.98     16139

avg / total       0.92      0.96      0.94     16853
```

Figure 30: Classification report and accuracy score for SVM classifier

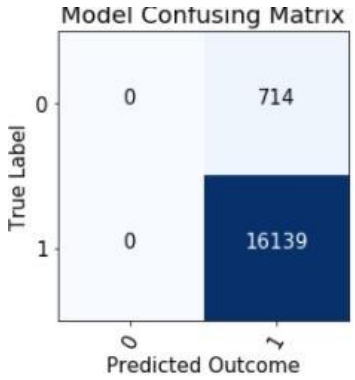


Figure 31: Confusion matrix for SVM classifier

```
normal prediction: AUC = 0.50
trained predictio score: AUC = 0.56
```

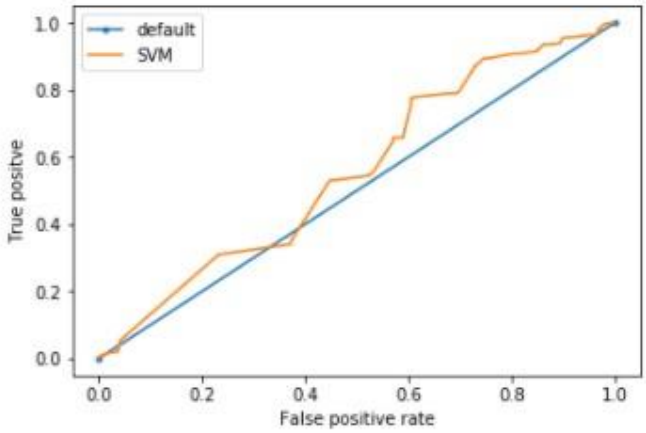


Figure 32: Graph representing performance of SVM classifier based on AUC

4.3.4 Naïve Bayes

Naïve Bayes classifier has been applied to classify data points on the basis of investigation type class (dependent variable) and this classifier classified data into 2 classes. The output of this classifier are determined on the basis of their precision, recall, f1-score, area under curve (AUC). After training the model, we tested it on new data and it had an accuracy score of 93 %. Figure 35 provides a summary of the performance of the Naïve Bayes classifier. We observe that this model can correctly predict 93% of the unseen records that represent circumstances corresponding to an aircraft accident.

```
69]:  1 print ('Accuracy Score :',accuracy_score(y_test,y_pred_nb ))
      2 print (classification_report(y_test,y_pred_nb))
```

```
Accuracy Score : 0.9323562570462233
             precision    recall  f1-score   support

          0       0.23      0.25      0.24       714
          1       0.97      0.96      0.96     16139

avg / total       0.94      0.93      0.93     16853
```

Figure 33: Classification report and accuracy score for Naïve Bayes classifier

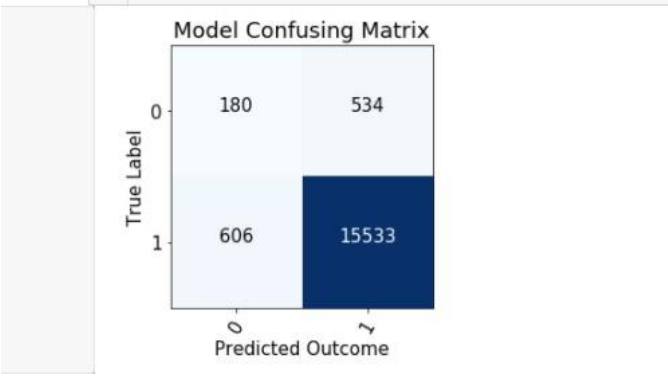


Figure 34: Confusion matrix for Naïve Bayes classifier

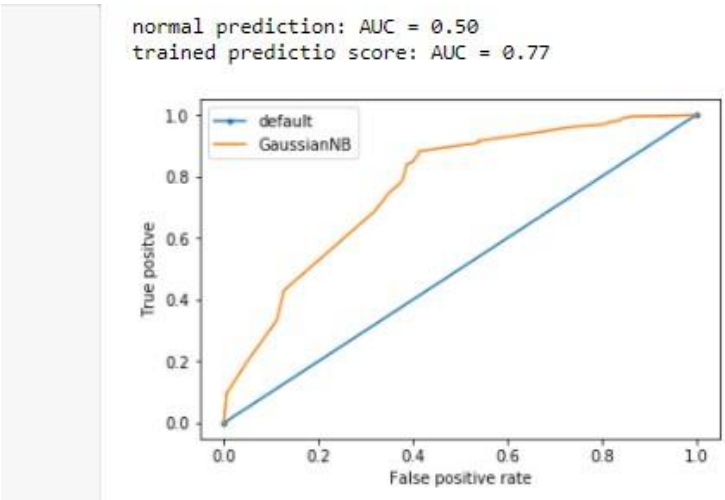


Figure 35: Graph represents performance of Naïve Bayes classifier based on AUC.

4.3.5 Summary of performance evaluation on different classifiers - Direct classification

In this project we have used four classification models looking for the best classification model for our dataset. Figure 36 is shows the performance of the models measured using the ROC-AUC curve. As the graph indicates, the random forest classifier and the decision tree classifier have outperform the other models with an AUC of 80%. These models both have an accuracy of 96% as is shown in Table 10.

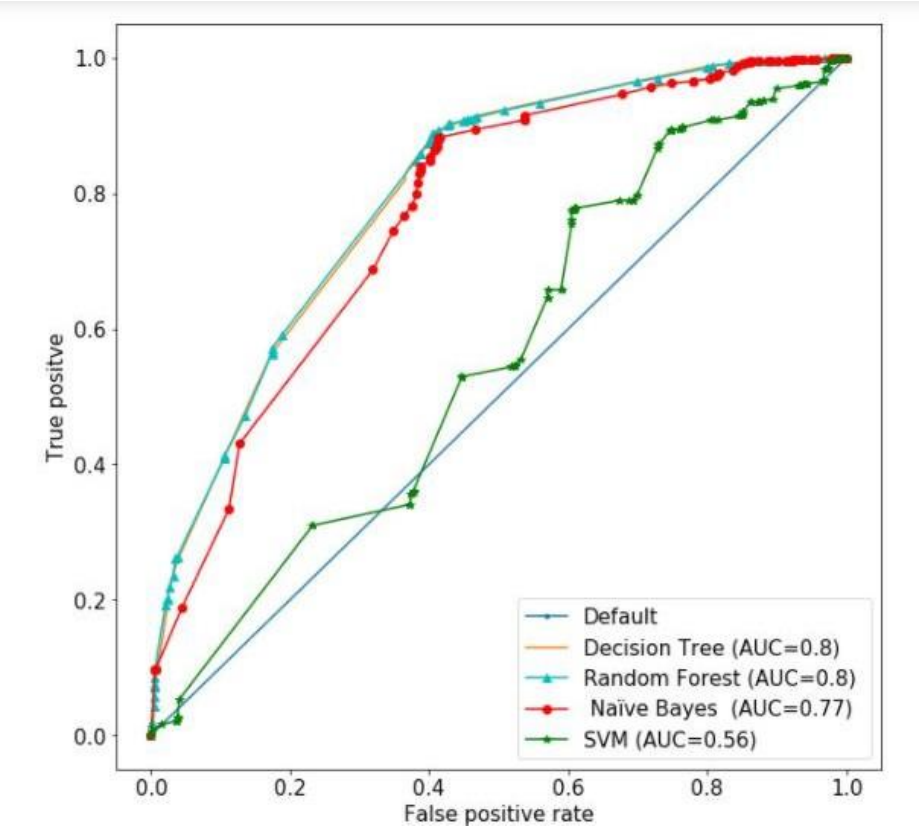


Figure 36: Evaluation performance of different classification models based on AUC on aircraft accident data.

Classification analysis with parameter tuning

In this project two analysis approaches have been implemented, direct classification analysis without parameter tuning and classification analysis with parameter tuning. Generally, the achieved accuracies of the models increased after applying parameter tuning. We employed Grid Search algorithm to tune parameters for some models while for others we used Random Search algorithm. Our intention was to use Grid Search with a wider range of parameters. However, after running into computational issues due to limited compute power, we resorted to Random Search method to tune the parameters.

Decision Tree classifier output

We used Random Search technique for parameter tuning. After obtaining the optimal parameter set and training the model, its accuracy on unseen data increased slightly from 95.8%to 96.0%. We note that this increase in accuracy resulted from an increase in the ability of themodel to predict incidents as incidents. We observed that the number of correctly predicted incidents increased from 128 to 148. The summary of the model's performance before and after parameters were tuned is present in Table 11.

Direct classification analysis					Classification analysis with parameter tuning			
Accuracy	0.9581				0.9601			
class	Precision	Recall	F1-score	Predicted	Precision	Recall	F1-score	Predicted
Incident(0)	0.10	0.53	0.16	128	0.12	0.57	0.19	148
Accident(1)	1	0.96	0.98	16725	1	0.96	0.98	16705
Avg/total	0.94	0.96	0.94	16853	0.9	0.96	0.97	16853

Table 11: Performance of Decision Tree classifier

Random Forest classifier output

We used Random Search technique for parameter tuning. After obtaining the optimal parameter set and training the model, the model's performance with regards to our class of interest (accidents) did not change. The slight change comes with an increase in the ability of the model to correctly recognize the negative class. The summary of the model's performance before and after parameters were tuned is present in Table 12 .

Direct classification analysis				Classification analysis with parameter tuning			
Accuracy	0.9585			0.9601			
class	Precision	Recall	F1-score	Precision	Recall	F1-score	Predicted
Incident(0)	0.56	0.10	0.17	0.57	0.12	0.18	714
Accident(1)	0.96	1	0.98	0.96	1	0.98	16139
Avg/total	0.94	0.96	0.94	0.94	0.96	0.95	16853

Table 12: Performance of Random Forest classifier

SVM classifier output

Random Grid Search was equally used for parameter tuning of the SVM. The model did not witness a significant change in its performance after parameter tuning. The recall of the

model remained unchanged as well at the accuracy. The predicted true positives were the same in both cases. We used Random Search technique for parameter tuning. The classified result increased slightly as it is shown in Table 13.

Direct classification analysis				Classification analysis with parameter tuning			
Accuracy	0.9576			0.9595			
class	Precision	Recall	F1-score	Precision	Recall	F1-score	Predicted
Incident(0)	0.0	0.0	0.0	0.57	0.12	0.18	714
Accident(1)	0.96	1	0.98	0.96	1	0.98	16139
Avg/total	0.92	0.96	0.94	0.94	0.96	0.95	16853

Table 13: Performance of SVM classifier

Naïve Bayes classifier output

The Naïve Bayes classifier was also trained and tested on the research data. We trained the model without setting a prior and thereafter, we used parameter tuning algorithm to identify a value for the prior. The two developed models – one without prior and another with a set prior – were trained and tested on new data. This led to an increase in accuracy from 93.2% to 96.0%. Table 14 shows a summary of the various parameters measured.

Direct classification analysis				Classification analysis with parameter tuning			
Accuracy	0.932			0.9601			
class	Precision	Recall	F1-score	Precision	Recall	F1-score	Predicted
Incident (0)	0.23	0.25	0.16	0.12	0.57	0.19	714
Accident (1)	0.97	0.96	0.96	1	0.96	0.98	16139
Avg/total	0.94	0.93	0.93	0.99	0.96	0.96	16853

Table 14: Performance of Naïve Bayes classifier

In Table 15, we illustrate the performance of considered classification models on the classification of aircraft accidents and incidents data before and after parameter tuning technique is applied. It is clear that classification accuracy after parameter tuning is better than before.

Code	Classifier	Accuracy before parameter tuning	Accuracy after parameter tuning
01	Decision Tree	95.81%	96.01%
02	Random Forest	95.85%	96.01%
03	SVM	95.76%	95.95
04	Naïve Bayes	93.23%	96.01%

Table 15: Summary of accuracy for different classifier before and after parameter tuning

5.0 Conclusion and Recommendations

In this research, we have used several machine learning approaches to find potential contributory factors to aircraft accident or incident based on feature importance. We employed different classification models, namely Support Vector Machine (SVM); Naïve Bayes classifier; Decision Tree classifier and Random Forest classifier to predict the potential occurrence of an incident or accident based on insight from historical data. We have also used two parameter tuning algorithms *GridSearchCV* and *RandomizedSearchCV* to achieve better accuracy for our models. By conducting parameter tuning, we obtained a good parameter combination for each algorithm trained, thereby enhancing the model's accuracy. By applying feature importance selection techniques, significant contributors to aircraft accident and incident were determined. As shown in Table 9 and Figure 23, "Phase of flight" was identified to be the most relevant feature to our target variable. We note from Table 9 that "Phase of flight" contributes 80% to determining if an aircraft would have an incident or an accident. From Table 5 there are four phases of flight with their contributions to aircraft incident/accident as follows: Landing phase has 24%; take-off phase with 19%; cruising phase 13% and manoeuvring phase with 12%. Thus, most aircrafts are likely to have an accident or incident during landing phase. This finding is a reflection of reality because this isthe phase where pilots are too busy checking all flight controls as they prepare to touch down [9]. The landing phase is prune to a number of factors which account for incident/accident occurring. These include a human error committed by any of the airmen during busy controls;poor visibility resulting from bad weather conditions that impede accurate landing; and wind direction. Another possible cause of an incident/accident during the landing phase is a technical problem from the malfunctioning of the aircraft. During take-off phase, there are three factors that mostly contribute to the aircraft accident which are poor visibility due to weather condition, direction of the wind, human error and technical issue from the malfunctioning of aircraft systems. During cruising phase, most accidents and incidents result from malfunctioning of pressurization system, engine failure or human error due to the non-compliance of instruction pilot is given from air traffic controller [4]. Lack of concentration from controller on traffic under his jurisdiction could also lead to an accident/incident during cruising. For manoeuvring phase, mostly the accident or incident happens on manoeuvring area due to the congestion of parking aircraft or poor visibility while aircraft is taxing or is pouched.

In this study we analysed aircraft accident dataset and developed machine learning models used to predict aircraft incident or accident. The models were trained using historical data and their performances measured on unseen data using different metrics: classification accuracy, confusion matrix, recall, precision, area under curve. As it is shown in Table 10 without parameter tuning, the best model in predicting the outcome variable was Random Forest with an accuracy score of 96% followed by Decision Tree with 96% accuracy, followed by SVM with 95%, and Naïve Bayes 93%. We conducted parameter tuning which improved the accuracy of our models. Table 15 summarizes the accuracy scores for the different models before and after parameter tuning. The models Random Forest, Decision Tree and Naïve Bayes all recorded similar performance with an accuracy score of 96.01%. The SVM however, scored a slightly lower accuracy of 95.59%. Comparing these two tables indicates that parameter tuning techniques provided better accuracy for all classification models we used. As seen from these two tables, we were able to achieve an appreciable level of incident/accident prediction using all the models trained. All classifiers used performed very well with accuracy score of around 96%.

In this research, we tested the performance of different classification models on our given dataset with and without parameter tuning. We performed parameter tuning in order to improve the accuracy of the models trained. Using machine learning techniques helps to discover unseen relationship between attributes in a dataset. In our case study, we used different machine learning techniques to find the most potential factors contributing to our target variable which is investigation type with two classes ‘accident’ and ‘incident’. The feature importance was the best method to provide the most relevant feature to our target variable. There are most common causes of accidents or incident associated to different phase of flight such as Landing, take-off, cruising and manoeuvring on ground. Those most common causes are human errors, malfunctioning of aircraft systems, poor visibility and wind direction [9]. In this research we limited ourselves to four independent variables due to the nature of dataset used including other feature in dataset could have led to different result, it is worth investigating the performance of this model by considering more independent features as new knowledge could be discovered. We would have loved to perform further parameter tuning for the models used. However, we were limited by computation capacity of our machines.

Currently all over the world our airspace is congested and as we have increasing number of new airlines, there is strong growth in traffic volume [16]. The number of flight delays, incidents and accidents is expected to increase if no measures are taken to improve safety even further. However, an increase in the number of accidents would be totally unacceptable. Aviation data analysts are recommended to use machine learning techniques to discover hidden causes of aircraft accidents and incidents, and further analysis on aviation data to protect our air travellers for future trends.

6.0 References

[1] A. Care and I. Sheets, “free puBliCatioNs.” pp. 1–11, 2008.

[2] Z. Nazeri, “Application of Aviation Safety Data Mining Workbench at American Airlines Proof-of-Concept Demonstration of Data and Text Mining Application of Aviation Safety Data Mining Workbench at American Airlines Proof-of-Concept Demonstration of Data and Text Mini,” no. November, 2003.

[3] “Reima Suomi,” pp. 178–198.

[4] S. Koteeswaran, N. Malarvizhi, E. Kannan, S. Sasikala, and G. Subbiah, “Data mining application on aviation accident data for predicting topmost causes for accidents,” *Cluster Comput.*, vol. 22, Sep. 2019, doi: 10.1007/s10586-017-1394-2.

[5] N. Malarvizhi and E. Kannan, “mining application on aviation accident data for predicting ost causes for accidents,” pp. 1–12, 2020.

[6] ICAO, “Accident Investigation and prevention divisional meeting,” pp. 2–3, 2008.

[7] N. Tulechki, “Natural language processing of incident and accident reports : application to risk management in civil aviation,” 2015.

[8] “What is an operational data store? | Stitch resource.” [Online]. Available: https://www.stitchdata.com/resources/operational-data-store/.

[9] M. Bazargan and V. S. Guzhva, “Factors contributing to fatalities in General Aviation accidents,” *World Rev. Intermodal Transp. Res.*, vol. 1, no. 2, pp. 170–182, 2007, doi: 10.1504/WRITR.2007.013949.

[10] V. S. Guzhva, M. Bazargan, and D. A. Byers, “Entrepreneurial spirit in government managed enterprises: evidence from the U.S. general aviation airports,” *J. Public Budgeting, Account. Financ. Manag.*, vol. 20, no. 4, pp. 419–438, 2008, doi: 10.1108/jpbafm-20-04-2008-b001.

[11] “5 Human Factors | Improving the Continued Airworthiness of Civil Aircraft: A Strategy for the FAA’s Aircraft Certification Service | The National Academies Press.” [Online]. Available: https://www.nap.edu/read/6265/chapter/7.

[12] AOPA Air Safety Foundation, “2009 Nall Report: Accident Trends and Factors for 2008,” *Statistics (Ber).*, 2009.

[13] G. Li, S. P. Baker, J. G. Grabowski, and G. W. Rebok, “Factors associated with pilot error in aviation crashes,” *Aviat. Sp. Environ. Med.*, vol. 72, no. 1, pp. 52–58, 2001.

[14] L. M. Veltman, “Incident Data Analysis using data mining techniques,” no. August, p. 69, 2008.

[15] K. Seerangan, N. Malarvizhi, E. Kannan, D. Kala, and G. Subbiah, “Data mining application on aviation accident data for predicting topmost causes for accidents,” *Cluster Comput.*, vol. 22, Sep. 2019, doi: 10.1007/s10586-017-1394-2.

[16] Y.-L. Hsu, “Airline Safety Management: The development of a proactive safety mechanism model for the evolution of safety management system,” *Airl. Saf. Manag. Dev. a proactive Saf. Mech. Model Evol. Saf. Manag. Syst. Coll. Aeronaut. AIR Transp. Gr.*, no. air safety management, pp. 1–343, 2004, [Online]. Available: https://core.ac.uk/download/pdf/139739.pdf.

[17] P. Dönmez, “Introduction to Machine Learning, 2nd ed., by Ethem Alpaydın. Cambridge, MA: The MIT Press2010. ISBN: 978-0-262-01243-0. $54/£ 39.95 + 584 pages.,” *Nat. Lang. Eng.*, vol. 19, no. 2, pp. 285–288, 2013, doi: 10.1017/s1351324912000290.

[18] A. Cutler, D. R. Cutler, and J. R. Stevens, “Ensemble Machine Learning,” *Ensemble*

Mach. Learn., no. February 2014, 2012, doi: 10.1007/978-1-4419-9326-7.
[19] "How to Create an Unsupervised Learning Model with DBSCAN." [Online]. Available: http://www.dummies.com/programming/big-data/data-science/how-to-create-an-unsupervised-learning-model-with-dbscan/.
[20] M. Awad and R. Khanna, "Efficient learning machines: Theories, concepts, and applications for engineers and system designers," *Effic. Learn. Mach. Theor. Concepts, Appl. Eng. Syst. Des.*, no. January, pp. 1–248, 2015, doi: 10.1007/978-1-4302-5990-9.
[21] E. G. Galitskaya and E. B. Galitskkiy, "Classification trees," *Sotsiologicheskie Issled.*, no. 3, pp. 84–88, 2013, doi: 10.4018/978-1-60960-557-5.ch006.
[22] A. A. Q. Doan, "Intro To Random Forest."
[23] L. Breiman, "ST4_Method_Random_Forest," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1017/CBO9781107415324.004.
[24] V. Jatana, "Hyperparameter tuning Tuning process," no. August, pp. 1–3, 2019.
[25] J. Thulin, "Master ' s Thesis in Computer Science : Machine learning-based classi ers in the Direkt Pro l grammatical pro ling system Supervisors : Jonas Granfeldt and Pierre Nugues," *Mach. Learn.*, no. January, 2007.
[26] Data to Fish, "Example of Confusion Matrix in Python." [Online]. Available: https://datatofish.com/confusion-matrix-python/.
[27] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognit.*, vol. 30, no. 7, pp. 1145–1159, 1997, doi: 10.1016/S0031-3203(96)00142-2.
[28] "Search | Kaggle." [Online]. Available: https://www.kaggle.com/search?q=sentiment+analysis+Movie+Reviews.
[29] M. Luckert and M. Schaffer-Kehnert, "Using Machine Learning Methods for Evaluating the Quality of Technical Documents," 2015, [Online]. Available: http://www.diva-portal.org/smash/get/diva2:920202/FULLTEXT01.pdf.
[30] "What is Data Independence? - Definition from Techopedia." 2017, [Online]. Available: https://www.techopedia.com/definition/1178/data-independence.

doc44

ORIGINALITY REPORT

18%	13%	9%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	link.springer.com Internet Source	1%
2	Submitted to Ain Shams University Student Paper	1%
3	Submitted to University of Huddersfield Student Paper	1%
4	Submitted to University of Sheffield Student Paper	1%
5	www.datacamp.com Internet Source	1%
6	Daniel Berrar. "Bayes' Theorem and Naive Bayes Classifier", Elsevier BV, 2019 Publication	1%
7	www.scribd.com Internet Source	1%
8	www.mii.lt Internet Source	1%
9	hdl.handle.net	

	Internet Source	1%
10	towardsdatascience.com Internet Source	<1%
11	Wang, Zhenghai, Guangdao Hu, YongZhang Zhou, Xin Liu, Xia Li, Kai Liu, and Xinchang Zhang. "", Geoinformatics 2008 and Joint Conference on GIS and Built Environment Advanced Spatial Data Models and Analyses, 2009. Publication	<1%
12	doc.lagout.org Internet Source	<1%
13	www.datasciencelovers.com Internet Source	<1%
14	arrow.tudublin.ie Internet Source	<1%
15	Submitted to Korea University of Technology and Education Student Paper	<1%
16	Submitted to University of Rwanda Student Paper	<1%
17	Adele Cutler. "Random Forests", Ensemble Machine Learning, 2012 Publication	<1%

18	books.openedition.org Internet Source	<1%
19	Submitted to Cummins College of Engineering for Women, Pune Student Paper	<1%
20	www.essay.uk.com Internet Source	<1%
21	mafiadoc.com Internet Source	<1%
22	Lin Li, Yixiang Huang, Jianfeng Tao, Chengliang Liu. "Internal leakage identification of hydraulic cylinder based on intrinsic mode functions with random forest", Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2019 Publication	<1%
23	docplayer.net Internet Source	<1%
24	Olawale Adepoju, Julius Wosowei, Shiwani lawte, Hemaint Jaiman. "Comparative Evaluation of Credit Card Fraud Detection Using Machine Learning Techniques", 2019 Global Conference for Advancement in Technology (GCAT), 2019 Publication	<1%

25	opus4.kobv.de Internet Source	<1%
26	theses.gla.ac.uk Internet Source	<1%
27	Submitted to GGS IP University Delhi Student Paper	<1%
28	Submitted to TechKnowledge Student Paper	<1%
29	meetnavpk.github.io Internet Source	<1%
30	ru.scribd.com Internet Source	<1%
31	Somtochukwu Godfrey Nnabuife, Boyu Kuang, James F. Whidborne, Zeeshan Rana. "Non-Intrusive Classification of Gas-Liquid Flow Regimes in an S-shaped Pipeline Riser Using a Doppler Ultrasonic Sensor and Deep Neural Networks", Chemical Engineering Journal, 2020 Publication	<1%
32	Submitted to University of Hong Kong Student Paper	<1%
33	Submitted to University of Limerick Student Paper	<1%
34	epubs.surrey.ac.uk Internet Source	<1%

35	Submitted to uhasselt Student Paper	<1%
36	tees.openrepository.com Internet Source	<1%
37	"Fuzzy Model based Classifiers", Cluster Analysis for Data Mining and System Identification, 2007 Publication	<1%
38	open.library.ubc.ca Internet Source	<1%
39	www.mitre.org Internet Source	<1%
40	"Ensemble Machine Learning", Springer Science and Business Media LLC, 2012 Publication	<1%
41	Submitted to University of Lancaster Student Paper	<1%
42	Submitted to TAR University College Student Paper	<1%
43	elib.suub.uni-bremen.de Internet Source	<1%
44	Mariette Awad, Rahul Khanna. "Efficient Learning Machines", Springer Nature America, Inc, 2015	<1%

	Publication	
45	ijireeice.com Internet Source	<1%
46	Submitted to Deakin University Student Paper	<1%
47	"Factors contributing to fatalities in General Aviation accidents", World Review of Intermodal Transportation Research, 2007 Publication	<1%
48	Submitted to Visvesvaraya Technological University Student Paper	<1%
49	www.ntsb.gov Internet Source	<1%
50	www.studytonight.com Internet Source	<1%
51	epdf.pub Internet Source	<1%
52	"Data Mining and Knowledge Discovery Handbook", Springer Science and Business Media LLC, 2005 Publication	<1%
53	Submitted to University of Warwick Student Paper	<1%
	www.caymanairports.com	

54	Internet Source	<1%
55	Submitted to University College London Student Paper	<1%
56	www.springerprofessional.de Internet Source	<1%
57	Submitted to uvt Student Paper	<1%
58	machinelearningmastery.com Internet Source	<1%
59	Submitted to National College of Ireland Student Paper	<1%
60	Submitted to Cardiff University Student Paper	<1%
61	www.nlr.nl Internet Source	<1%
62	Mahardhika Pratama, Jie Lu, Sreenatha G. Anavatti, Jose Antonio Iglesias. "A recurrent meta-cognitive-based Scaffolding classifier from data streams", 2014 IEEE Symposium on Evolving and Autonomous Learning Systems (EALS), 2014 Publication	<1%
63	manualzz.com Internet Source	<1%

64	www.mdpi.com Internet Source	<1%
65	Submitted to University of Newcastle Student Paper	<1%
66	Fred Otieno, Nathan Williams, Patrick McSharry. "Forecasting Energy Demand for Microgrids Over Multiple Horizons", 2018 IEEE PES/IAS PowerAfrica, 2018 Publication	<1%
67	Submitted to hotelschool Student Paper	<1%
68	Submitted to BAC International Study Centre Student Paper	<1%
69	eprints.kfupm.edu.sa Internet Source	<1%
70	epdf.tips Internet Source	<1%
71	www.nap.edu Internet Source	<1%
72	researchbank.rmit.edu.au Internet Source	<1%
73	arxiv.org Internet Source	<1%
	www.hindawi.com	

74	Internet Source	<1%
75	Submitted to Babes-Bolyai University Student Paper	<1%
76	www.nature.com Internet Source	<1%
77	wiredspace.wits.ac.za Internet Source	<1%
78	Submitted to Nottingham Trent University Student Paper	<1%
79	Truong Son Pham, Tuan Hao Hoang, Van Canh Vu. "Machine learning techniques for web intrusion detection — A comparison", 2016 Eighth International Conference on Knowledge and Systems Engineering (KSE), 2016 Publication	<1%
80	Reima Suomi, Olli Sjöblom. "chapter 11 Data Mining in Aviation Safety Data Analysis", IGI Global, 2009 Publication	<1%
81	Md. Aquil Khan. "Chapter 5 Multiple-Source Approximation Systems, Evolving Information Systems and Corresponding Logics: A Study in Rough Set Theory", Springer Science and Business Media LLC, 2016 Publication	<1%

82	scholar.afit.edu Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		