A

PROJECT REPORT

ON

LEARNING FROM THE DISASTER

Submitted In Partial Fulfillment Of TheRequirement For The Award Of the Degree of

BACHELOR OF TECHNOLOGY IN

ELECTRONICS AND COMPUTER ENGINEERING (ECM)

By

B.RISHI SRIVATHSAVA (19311A19D8)

B. SAI TEJA GOUD (19311A19D9)

P.VIKAS (19311A19G6)

Under the guidance of

Mrs. K. SREELATHA
Assistant Professor

Department Of Electronics & Computer Engineering

Sreenidhi Institute Of Science & Technology

2022-2023

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

SREENIDHI INSTITUTION OF SCIENCE AND TECHNOLOGY

(Autonomous)

CERTIFICATE

This is to certify that the Dissertation entitled **" LEARNING FROM THE DISASTER"** is bonafide work done and submitted by **B.RISHI SRIVATHSAVA (19311A19D8), P.VIKAS(19311A19G6)** and **B.SAITEJA GOUD(19311A9D9)** in partial fulfillment of the requirement for the award of Degree of Bachelor of Electronics and Computer Engineering, SREENIDHI INSTITUTE OF SCIENCE AND TECHNOLOGY, Affiliated to Jawaharlal Nehru Technological University, Hyderabad is a record of bonafide work carried out by him/her. The results embodied in this dissertation have not been submitted to any other university for the award of any other degree or diploma**.**

Mrs. K. SREELATHA
Associate Professor,
SNIST, Hyderabad

Dr. D. MOHAN
Department of ECM,
SNIST, Hyderabad

DR. M. SHAILAJA
Associate Professor
Project Coordinator

External Examiner

ACKNOWLEDGEMENTS

We convey our sincere thanks to **Dr. C. V. Tomy,** Director and **Dr. T. Ch Siva Redd**y, Principal, Sreenidhi Institute of Science and Technology, Ghatkesar, for providing resources to complete this project.

We are very thankful to **Dr. D. Mohan**, Head of ECM Department, Sreenidhi Institute of Science and Technology, Ghatkesar, for providing an initiative to this project and giving valuable timely suggestions over my project work and for their kind cooperation in the completion of the project.

We convey our sincere thanks to our internal guide**, Mrs. K. SREELATHA**, Associate Professor and to our project coordinator, **Dr. M. Shailaja**, Associate Professor, Sreenidhi Institute of Science and Technology, for their continuous help, cooperation and support to complete this project.

Finally, we convey our sincere thanks to all the faculties of the ECM department, for their timely suggestions and guidance in the completion of this project

DECLARATION

This is to certify the work reported in the present project titled **“LEARNING FROM DISASTER”** is a record work done by us in Department of Electronics and Computer Engineering, Sreenidhi Institute of Science and Technology, Yamnampet, Ghatkesar.

The report is based on the project work done entirely by us and not copied from any other source.

B. Rishi Srivathsava(19311A19D8)

B. Sai Teja Goud (19311A19D9)

P. Vikas (19311A19G6)

ABSTRACT

The main goal of this study, "**LEARNING FROM THE DISASTER**," is to examine the relationship between the survival and death rates of those who were travelling in the "titanic rms" by using conventional methods to analyse the rate as accurately as is practical.

The algorithm, however, is greedy and does not rely on a single algorithm. Instead, it analyses the findings in light of the data taken into account and makes use of the extremely precise values produced by each process.

This report's data set was taken from the "kaggle" website. based on the factors this website takes into account, such as passenger class, passenger age, passenger sex, passenger id, and siblings.

However, the subsequent comparison takes into account the prediction accuracy of each method, including decision trees, random forest, xg-boost, gradient boosting algorithm, K-Nearest neighbours, Logistic Regression

List of Contents

List of Figures

Contents **Page No**

Chapter 1 INTRODUCTION

This study, "LEARNING FROM THE DISASTER," examines the relationship between passenger survival and mortality rates on board the "rms Titanic" when the enormous ship sank in the "North Atlantic" ocean as a result of an unintentional "ship-wreck."

In order to achieve the optimum result, it is important to compare the results of different machine learning algorithms to the ideal values listed in the datasets made available by the "kaggle" website. Age, gender, passenger-class, passenger-id, and other significant criteria are taken into account as potential determinants of survival.

Accuracy comparisons are made between the algorithms Nave Bayes, Logistic Regression, Decision Tree, SVM, and Random Forest. The best result is then displayed after comparisons between each algorithm's most accurate output. The results are compared for each attribute using various combinations of the characteristics after each attribute has undergone an accuracy check.

1.1 EXISTING SYSTEM

Earlier, a number of data analysts tried to determine why certain passengers perished in the shipwreck while others managed to live. The majority of the previously examined data relate to a certain algorithm taking into account the greatest number of features in that algorithm.Lam & Tan et al. performed one of the fewest attempts to explore the combination of the algorithms, using the Naive Bayes, Decision tree, and SVM algorithms.

1.2 PROPOSED SYSTEM

It is considered that the accuracy of a forecast does not totally depend on the quantity of characteristics in a particular algorithm, according to reports by "lam and tang et al". The machine learning methods Naive Bayes, Logistic Regression, Decision Tree and Random Forest, K-Nearest Neighbours, XG-Boost, Gradient Boosting Classifier, and Logistic Regression are used to create this report.

Chapter 2 Literature survey

1. Titanic: Exploring Survival Analysis" by Kamil Sindi and Taimur Zahid (2020)

   In order to forecast the survival of Titanic passengers, this study investigates survival analytic methodologies. The authors make use of a dataset made up of 891 passenger records and various variables like passenger class, gender, and age. The findings demonstrate that the Cox proportional hazards model, with an accuracy rate of 76.44%, performs best in terms of accuracy.

2. "Predicting the Fate of the Titanic Passengers: A Comparative Study of Machine Learning Algorithms" by Abdelhadi Abderrahim and Ali Oussous (2020)

   In order to forecast the survival of Titanic passengers, this study evaluates the effectiveness of different machine learning techniques, including decision trees, random forests, and support vector machines. The authors make use of a dataset made up of 891 passenger records and various variables like passenger class, gender, and age. The outcomes demonstrate that, with an accuracy rate of 79.70%, the decision tree algorithm works best.

3. "Data Analysis of Titanic Passenger Survival Using Machine Learning Techniques" by Sai Sruthi Pothula, Supriya Kavuri, and P. Srinivas Kumar (2020)

   This study uses a variety of machine learning techniques to forecast the survival of Titanic passengers, including decision trees, random forests, and k-nearest neighbours. The authors make use of a dataset made up of 891 passenger records and various variables like passenger class, gender, and age. The outcomes demonstrate that the accuracy of the k-nearest neighbours algorithm is the highest, at 78.70%.

Chapter 3 System Analysis

3.1 OPERATING SYSTEM:

- ➢ Windows 10 (7/8)

3.1.1 HARDWARE ESSENTIALS:

- ✓ 4gb (and above) ram
- ✓ 160gb (and above) hard disk

3.1.2 SOFTWARE ESSENTIALS:

- ✓ Python (3.6.2/5/6/7/8)

3.2 LIBRARIES USED:

1. **Numpy:** A strong Python package for numerical computing is called NumPy. It offers support for sizable, multidimensional arrays and matrices, as well as a range of mathematical operations for effectively using these arrays. Due to its capability to effectively handle enormous datasets and carry out difficult mathematical operations, NumPy is frequently utilised in scientific and data-related applications. As a result of its reputation for quick and effective array operations, it is a key library in the Python data science ecosystem. Additionally, NumPy works well with other libraries like Pandas and Matplotlib, which improves its capacity for data manipulation and visualisation.

2. **Pandas:**Python has a robust data analysis and manipulation toolkit called the Panda library. It offers data structures and procedures that facilitate working with structured data in spreadsheets and SQL tables, among other formats. Pandas provides a DataFrame object, a two-dimensional data structure resembling a table with labelled rows and columns. Data cleaning, transformation, and investigation are made possible by it.

3. **Matplotlib:** A robust Python library for data visualisation is called Matplotlib. For making static, dynamic, and interactive plots, charts, and graphs, it offers a wide variety of capabilities. You can create line plots, scatter plots, bar plots, histograms, heatmaps, and a variety of other visualisations with Matplotlib. Almost every component of your plots, including the axes, labels, colours, markers, and legends, can be changed. Matplotlib is a well-liked option for data analysis and exploration since it easily interfaces with other libraries like NumPy and Pandas. It is the preferred tool for Python data visualisation because of its adaptability, simplicity, and thorough documentation.

4. **Seaborn:** On top of Matplotlib, Seaborn is a well-known Python data visualisation package. It offers a sophisticated user interface for producing visually appealing and educational statistical visuals. Seaborn features a variety of plotting functions, such as scatter plots, line plots, bar plots, histograms, and more, and makes it easier to create complicated visualisations. To improve the plots' visual appeal, it also has pre-installed themes and colour schemes. Pandas data structures and Seaborn work well together, making it simple to visualise data straight from dataframes. It is frequently used for data analysis, exploratory data visualisation, and producing graphics that are suitable for publication.

5. **Sklearn:** Python's scikit-learn, sometimes referred to as sklearn, is a well-liked machine learning library. It offers a wide range of tools and techniques for preprocessing data, choosing features, training models, and evaluating them. Machine learning applications including classification, regression, clustering, and dimensionality reduction are all simple to accomplish with sklearn. It is user-friendly for both novice and seasoned practitioners due to its consistent and clear API. Sklearn effectively integrates with other Python scientific libraries like NumPy, SciPy, and matplotlib to support effective data analysis and visualisation workflows. Sklearn is an all-around strong and flexible package that is essential to the Python machine learning environment.

Chapter 4 Design

4.1 DATA ANALYTICS

The kaggle website provided the datasets utilised in this analysis. A sample of 418 passengers was used to create the dataset because it is necessary to link the death and demise rates. When using the algorithms, information like age, gender, siblings, and passenger id is taken into account.

The table that is provided below represents the attributes that were taken into consideration for this report and were taken from the kaggle website. This data is cleaned to remove any missing information, and the data is predicted using various methods before being compared for the best outcome.

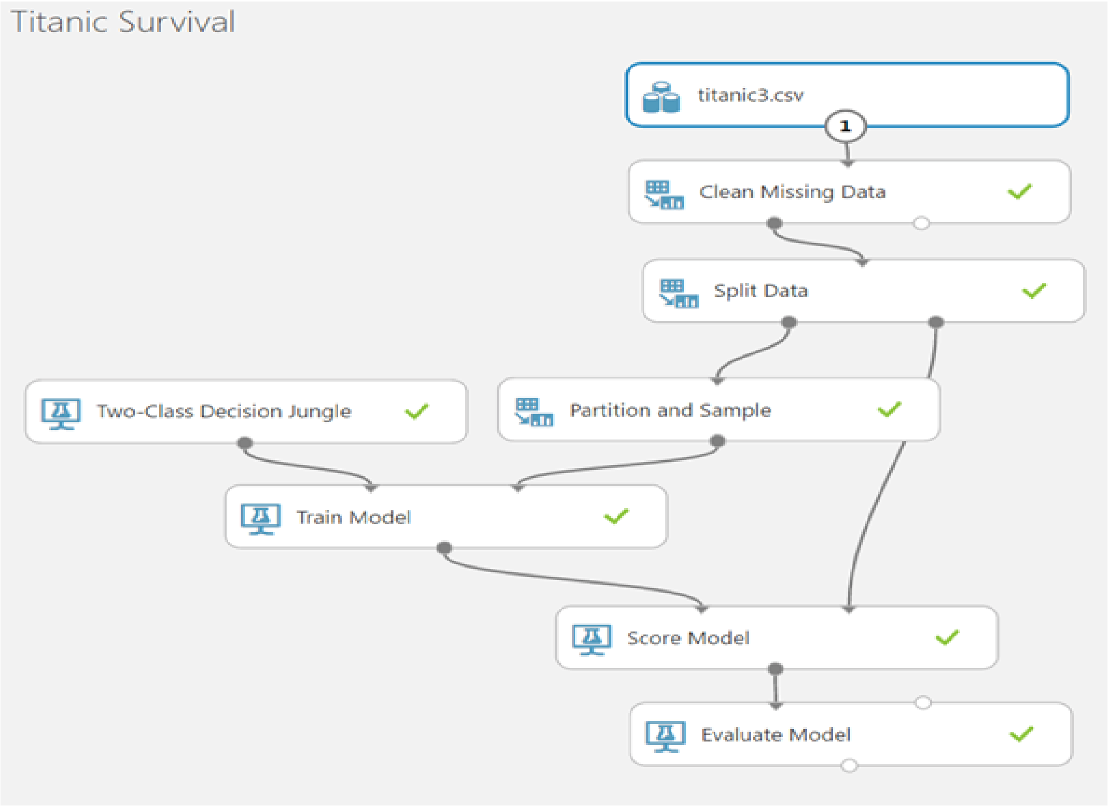
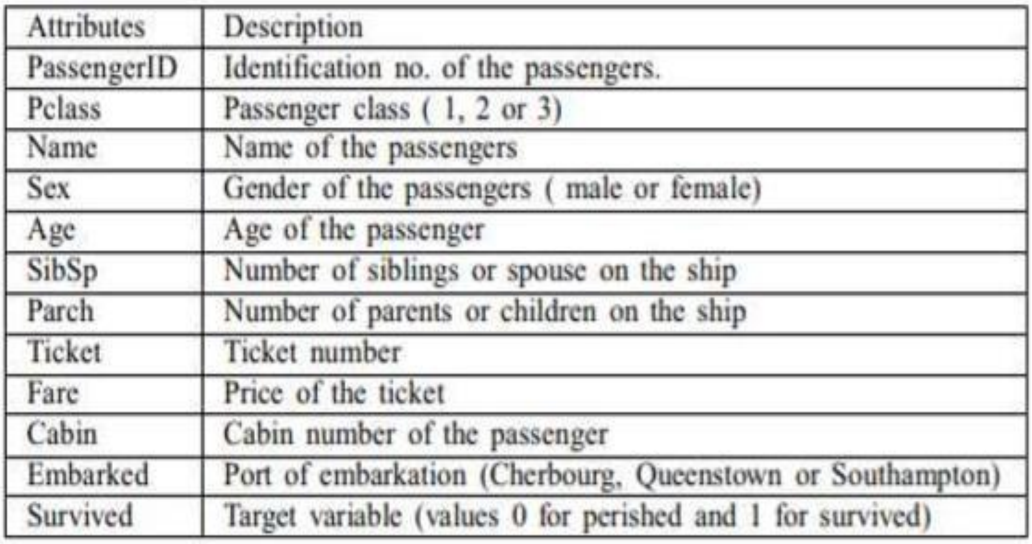


Fig 4.1.1:DataFlow of project

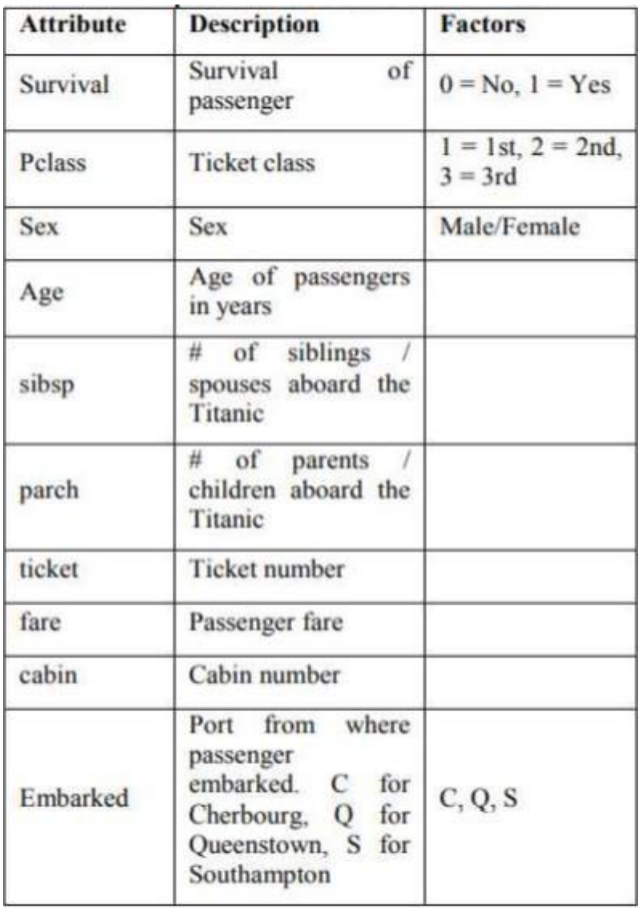
4.1.1 ATTRIBUTES IN TRAINING DATASET



Attributes	Description
PassengerID	Identification no. of the passengers.
Pclass	Passenger class (1, 2 or 3)
Name	Name of the passengers
Sex	Gender of the passengers (male or female)
Age	Age of the passenger
SibSp	Number of siblings or spouse on the ship
Parch	Number of parents or children on the ship
Ticket	Ticket number
Fare	Price of the ticket
Cabin	Cabin number of the passenger
Embarked	Port of embarkation (Cherbourg, Queenstown or Southampton)
Survived	Target variable (values 0 for perished and 1 for survived)

Fig 4.1.2:Attributes used in training dataset

4.1.2 STRUCTURE OF INPUT DATASET



Attribute	Description	Factors
Survival	Survival of passenger	0 = No, 1 = Yes
Pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
Sex	Sex	Male/Female
Age	Age of passengers in years	
sibsp	# of siblings / spouses aboard the Titanic	
parch	# of parents / children aboard the Titanic	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
Embarked	Port from where passenger embarked. C for Cherbourg, Q for Queenstown, S for Southampton	C, Q, S

Fig 4.1.3:Structure Of Input Dataset

4.1.3 AGE PLOT

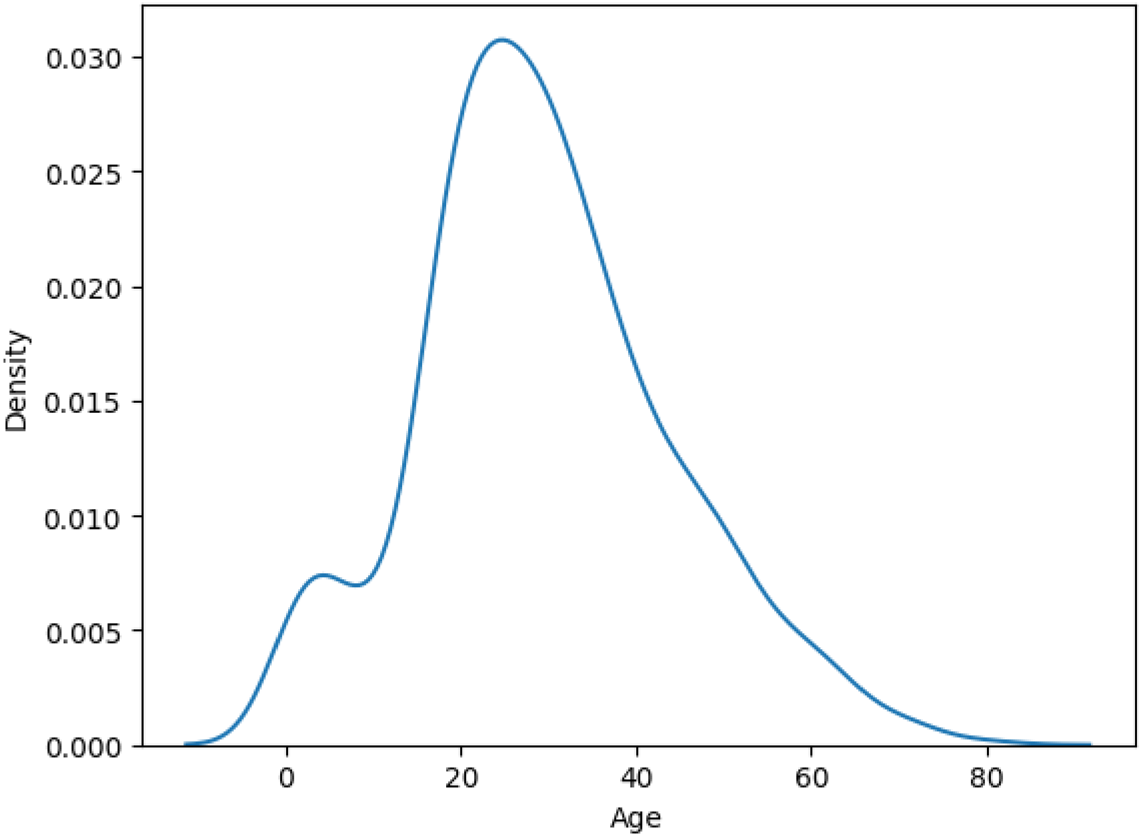


Fig 4.1.4: Age v/s Density

4.1.4 SEX BAR PLOT

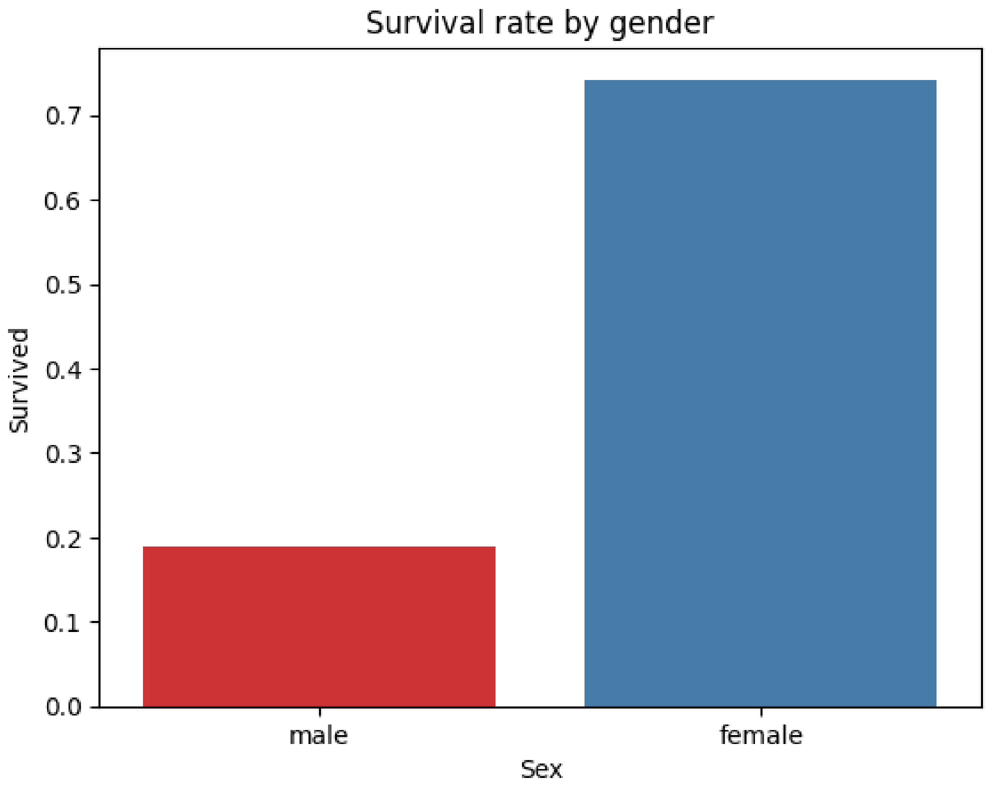
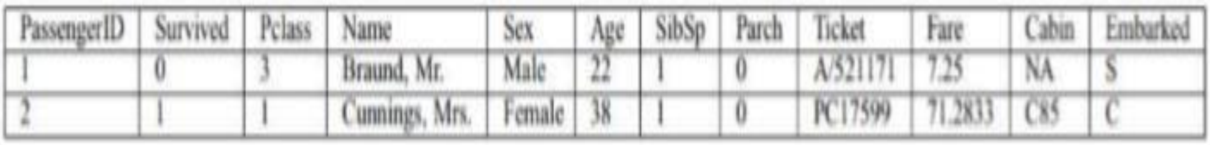


Fig 4.1.5: Sex v/s Survived

4.1.5 KAGGLE DATASET



PassengerID	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	Braund, Mr.	Male	22	1	0	A/521171	7.25	NA	S
2	1	1	Cunnings, Mrs.	Female	38	1	0	PC17599	71.2833	C85	C

Fig 4.1.6: Kaggle Dataset

4.2. ALGORITHMS USED

Naive Bayes, logistic regression, decision trees, random forests, K-Nearest, XG-boost, Gradient Boosting classifiers, and Random Forest Classifier are some of the qualities used in this research. **[4]** These algorithms forecast the outcomes, and as a result, the actual outcomes are compared among them for greater accuracy. The attributes used in this report are as follows, and they should be varied in order to demonstrate the relationship between survival and the passing rate of the passengers in the "titanic rms":

1. PassengerId
2. Survived
3. Pclass
4. Name
5. Sex
6. Age
7. SibSp
8. Parch
9. Ticket
10. Fare
11. Cabin
12. Embarked

As a result, the aforementioned algorithms are used in accordance with their functions. **[9]** The naive mathematician formula is a classification algorithm that won't be able to create a prediction model, hence it is useless in Python. The concept that "all the options that belongs to a category are independent" is usually addressed using the formula. **[11]** Once the chance is used and the outcomes are improved, a category's probability is obtained.

4.2.1 NAIVE BAYES

For classification and prediction problems, the Naive Bayes theorem is a probabilistic technique used in statistics and machine learning. It is predicated on the "naive" assumption of feature independence and Bayes' theorem.

The main application of Naive Bayes is in classification jobs. The algorithm determines the conditional probabilities of each class label given the observed data, then forecasts the most likely class label given a set of features or attributes (X) and a class label (Y).

The approach uses a labelled dataset to create a statistical model during the training phase**. [5]** It determines the likelihood probability of each feature given each class (P(X|Y)) and the prior probability of each class label (P(Y)).**[7]** Each class label's prior probability is a representation of the likelihood that it will appear in the dataset, whereas the likelihood.

The model can be used for prediction after training. The approach uses Bayes' theorem to determine the posterior probability of each class label given the observed data given a fresh set of features (X). **[6]** The predicted class is given the class label with the highest posterior probability.

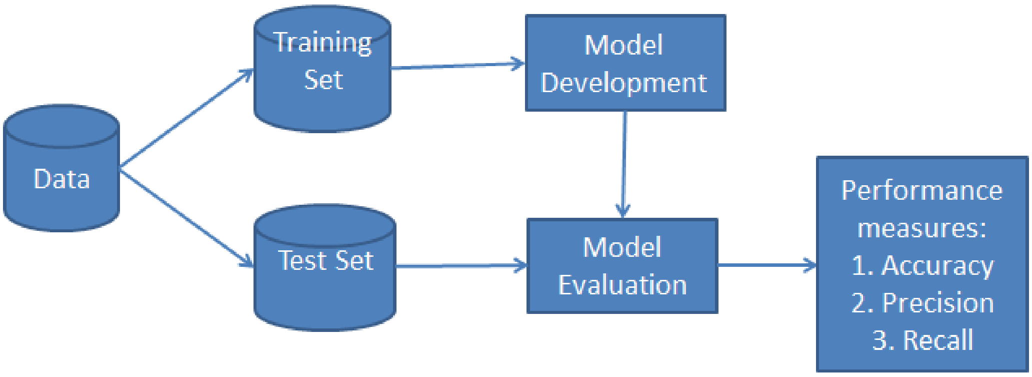


Fig 4.2.1: Navie Bayes Workflow

Numerous applications, including as text classification, spam filtering, sentiment analysis, and recommendation systems, make extensive use of naive bayes. It is renowned for being straightforward, scalable, and capable of managing enormous feature spaces. **[10]** However, in some situations where feature dependencies exist, the assumption of feature independence can limit its performance.

4.2.2 LOGISTIC REGRESSION

When attempting to predict a binary result (such as true/false, yes/no, or 0/1) in binary classification problems, a statistical model called logistic regression is used. It is a kind of generalised linear model that depicts the relationship between the features and the likelihood of the binary outcome using the logistic function, also referred to as the sigmoid function **[16].**

The assumption behind logistic regression is that the binary outcome's log-odds (also known as logit) and characteristics are linearly related. **[15]** The logistic function, which transfers the log-odds to a value between 0 and 1, transforms the log-odds into probabilities.

Using a labelled dataset, the logistic regression model is trained. The objective is to determine the coefficients' ideal values, which minimize the discrepancy between the anticipated probabilities and the actual labels. Usually, an optimization approach like gradient descent is used for this **[19].**

In logistic regression, the inaccuracy between the projected probabilities and the actual labels is measured by the cost function. A decision boundary is established based on a selected threshold (often 0.5) once the model has been trained. The outcome is anticipated to be positive if the predicted probability is higher than the threshold; otherwise, the outcome is predicted to be negative **[16].**

Using methods like one-vs-the-rest or softmax regression, logistic regression can be expanded to address multiclass classification issues. It is frequently employed in a variety of industries, including social sciences, finance, healthcare, and marketing.

4.2.3 SUPPORT VECTOR CLASSIFIER:

A machine learning approach for binary classification is called the Support Vector Classifier (SVC). Support Vector Machines (SVM), a potent method for both classification and regression applications, provide the foundation of this system.The best hyperplane in the feature space that divides the two classes is what SVC seeks to identify **[2].** The margin, or the separation between the hyperplane and the closest data points for each class, is a decision boundary that is maximised by the hyperplane. The SVC algorithm looks for the hyperplane with the highest achievable margin.

The data points nearest to the decision boundary (the hyperplane) are known as support vectors. The hyperplane is greatly influenced by these locations. They are the sole pieces of information used to calculate the margin and the decision boundary. **[22]**SVC can employ the kernel approach in situations where the classes are not linearly separable in the original feature space. In order to possibly become linearly separable, the kernel approach entails changing the data points into a higher-dimensional feature space. The linear kernel, polynomial kernel, radial basis function (RBF) kernel, and sigmoid kernel are examples of common kernel functions.

Based on the provided labelled data, SVC learns the ideal hyperplane parameters (coefficients and intercept) during the training phase.**[28]** Finding the hyperplane that maximizes margin while minimizing misclassification error is the objective.When the data points cannot be entirely separated, VC allows for some misclassification. It adds a soft margin that allows some data points to be misclassified but works to reduce the number of such mistakes.

The trade-off between maximising the margin and permitting misclassifications is controlled by the parameter C. When C is smaller, the margin is wider and there are more misclassifications; when C is greater, the margin is narrower and there are fewer misclassifications. Once trained, the SVC model can be used to forecast fresh, unobserved data points.**[30]** Based on which side of the decision boundary the data point lies, the algorithm determines the class label.

The reason SVC is a well-liked technique for classification problems is that it can manage complex decision boundaries, has efficient outlier handling procedures, and can be expanded to accommodate nonlinear relationships using various kernel functions. For large datasets, SVC can be computationally expensive and may be sensitive to the selection of hyperparameters.

4.2.4 K-NEAREST NEIGHBOURS:

K-Nearest Neighbours (KNN) is a simple and flexible technique used in machine learning for both classification and regression tasks. **[35]** It is a non-parametric method that bases its predictions on the k nearby feature space data points.KNN merely saves the labelled training data during the training phase. Model building or parameter estimates are not involved.

KNN measures the similarity between data points in the feature space using a distance metric, often the Euclidean distance. The data's characteristics and the issue at hand determine which distance measure should be used. KNN determines the k closest neighbours to a newly discovered, unlabeled data point using the selected distance measure. **[30]** K's value is a hyperparameter that must be predetermined. Based on the dominant class (in the case of classification) or the average value (in the case of regression) among the k closest neighbours, the predicted label or value for the new data point is established.

In KNN, the k value selection is crucial. A small value of k (such as 1) may result in overfitting and make the model more susceptible to data noise, whereas a large value of k may underfit the model and ignore local patterns. Techniques like grid search or cross-validation can be used to find the ideal value of k. Depending on how far away the neighbours are from the new data point, you might wish to give them different weights in some situations. This can be done by utilising weighted KNN, in which each neighbor's weight is inversely proportionate to how close to the new data point it is **[35].**

As a result, the forecast is more strongly influenced by nearby neighbours than by neighbours further away.To make sure that no single feature dominates the distance calculation, it is frequently required to scale or normalise the features in KNN. Techniques like z-score normalisation and min-max scaling can be used for this.

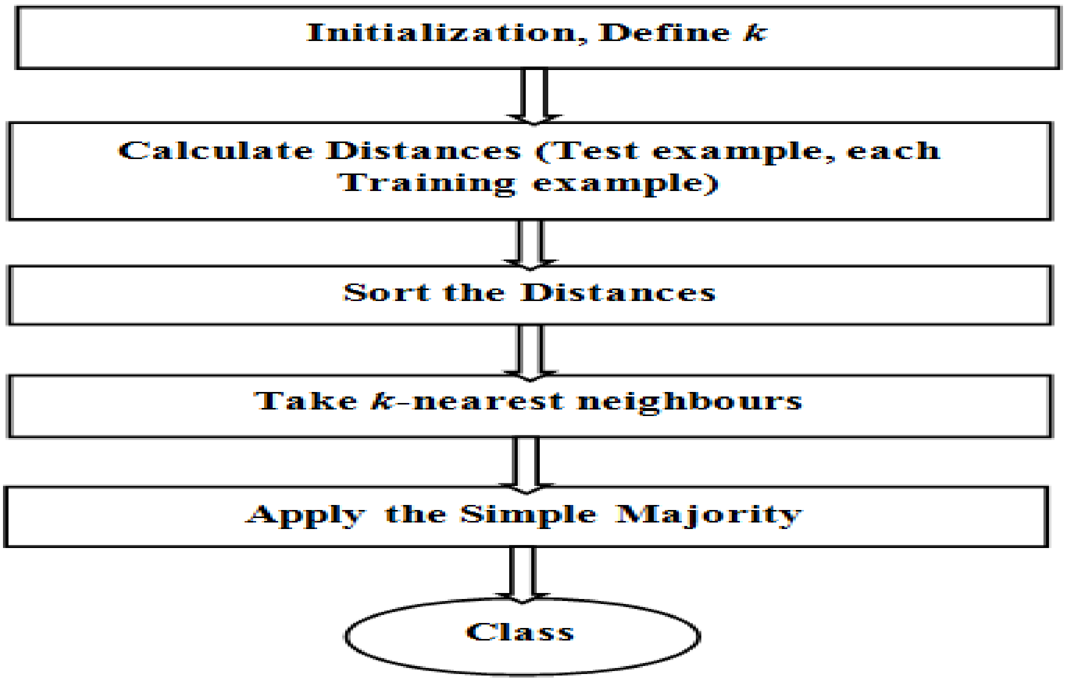

Initialization, Define k
Calculate Distances (Test example, each Training example)
Sort the Distances
Take k-nearest neighbours
Apply the Simple Majority
Class

Fig 4.2.2: KNN Workflow

4.2.5 <u>XG-BOOST ALGORITHM:</u>

A well-known and effective machine learning method that is a member of the gradient boosting family is called XGBoost (Extreme Gradient Boosting). It is intended to produce extremely accurate predictive models by fusing the advantages of several weak predictive models, often known as "weak learners" or "base models."

Gradient boosting is a method for creating ensemble models and is implemented in XGBoost**.[37]** In order to create a final, more accurate forecast, it systematically trains a number of weak models and then combines their predictions. Every model after that in the series fixes the errors introduced by the earlier ones. Decision trees serve as the main weak learners in XGBoost**. [40]** Simple hierarchical models called decision trees use unique regions to divide the input space into predictions. Both classification and regression issues can be handled by the algorithm.

To avoid overfitting and boost generalisation, XGBoost uses a variety of regularisation strategies. Regularisation lessens the possibility of the model fitting noise in the training data and aids in keeping the model's complexity under control. On the model's weights, the algorithm applies L1 and L2 regularisation terms, and it also has a term for regulating the complexity of the tree structures**.[33]** The parallel processing capabilities of XGBoost make it extremely effective and scalable. In order to train models more quickly and handle big datasets, it makes use of the parallel computing capabilities of contemporary technology, like as multi-core CPUs.

You can evaluate the applicability of various features in the prediction job using the measure of feature importance provided by XGBoost. You can use this information to guide feature selection or engineering efforts and to better understand the underlying trends in your data. We can adjust the behaviour of the algorithm with the help of the extensive selection of customizable hyperparameters provided by XGBoost. The learning rate, number of trees, maximum depth of the trees, regularisation terms, and other factors are all controlled by these parameters. The model's performance can be greatly impacted by tuning these hyperparameters.

Regression, classification, and ranking are just a few of the machine learning tasks that XG-Boost has been effectively used for. As a result of its accuracy, effectiveness, and interpretability, it has been employed in a variety of industries, including banking, healthcare, marketing, and internet advertising.

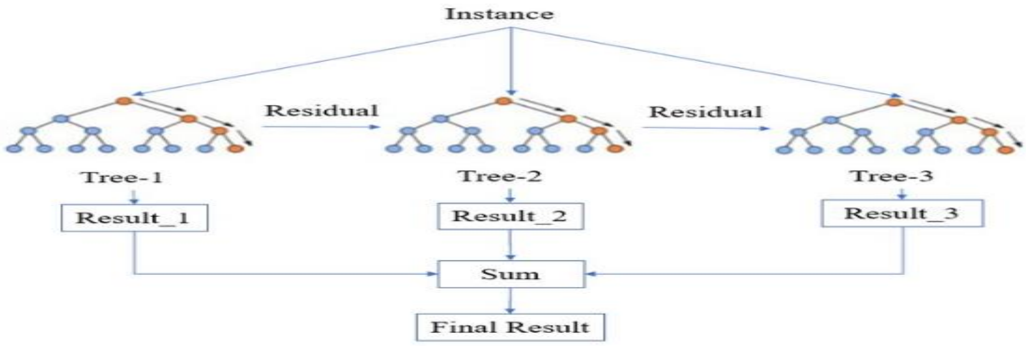


Fig 4.2.3: XG Boost Workflow

4.2.6 GRADIENT BOOSTING ALGORITHM:

A machine learning approach called gradient boosting combines a number of weak predictive models, often decision trees, to produce a stronger and more precise prediction model. It is a potent and popular algorithm that has excelled in a variety of fields and machine learning competitions.Gradient boosting sequentially trains a number of unreliable models. Each model is trained to fix the mistakes created by the earlier models, concentrating on the predictions that still include errors. The following models in the series are created to capture the loss function's residual errors or gradients**.[32]**

A final forecast is created by combining the predictions of all the weak models. Predictions are often averaged or summarised, with the weights being depending on the performance of each weak model. The accuracy of predictions is increased overall thanks to the ensemble method.Gradient boosting uses gradient descent to optimise the model's parameters. It makes use of an objective or loss function, which measures the discrepancy between expected and actual values. In order to minimise the error, the algorithm iteratively modifies the parameters in the direction of the loss function's steepest descent**.[40]**

Weak learner base models are used in gradient boosting. These weak learners are often shallow decision trees with few splits, sometimes known as decision stumps. Decision trees are used because they are easy to use, effective at capturing nonlinear relationships, and able to capture interactions between features.To avoid overfitting, gradient boosting uses regularisation techniques. Regularisation aids in keeping the model's complexity under control and increases generalizability. The boosting process can be controlled using strategies like shrinkage (learning rate), tree depth limits, and sampling.

The gradient boosting technique offers a feature importance metric that quantifies the relative weights given to various characteristics in the prediction task. It evaluates how each feature contributes to lowering the loss function during the boosting phase. The understanding of the underlying patterns in the data as well as feature engineering and feature selection can both benefit from this knowledge.

Gradient boosting includes adjusting a number of algorithm-controlling hyperparameters.

These comprise the learning rate, the quantity of unreliable models, the maximum tree depth, regularisation terms, and more. In order to maximise the performance of the gradient boosting model, proper hyperparameter adjustment is necessary.

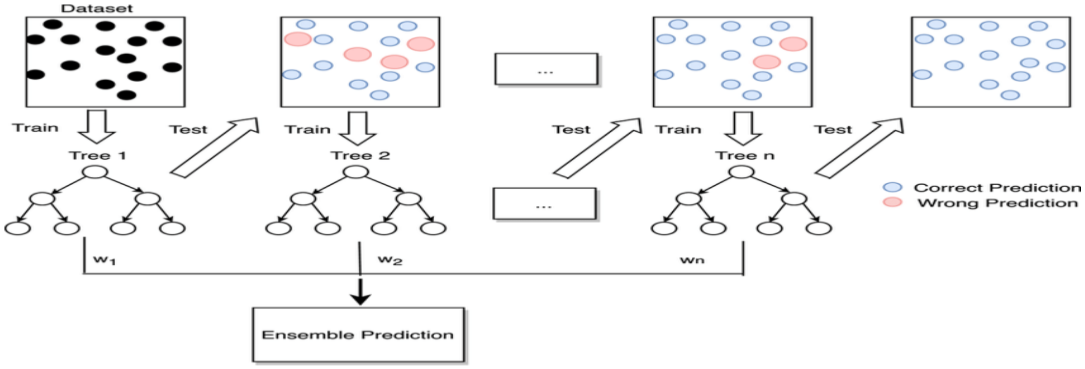


Fig 4.2.4: Gradient Boosting algorithm Workflow

4.2.7 DECISION TREE:

An efficient machine learning technique that may be utilised for both classification and regression applications is the decision tree algorithm. It is a supervised learning technique that creates a model for making predictions based on input features in the form of a tree structure.A decision tree is a type of hierarchy made up of nodes and branches. The root node, which is the top node, is what the entire dataset is represented by. The branches reflect the choices or results based on the intermediate nodes, which represent traits or attributes. The ultimate forecasts or results are represented by the leaf nodes.

The most useful qualities for dividing the data at each node are chosen by the decision tree algorithm. The feature that produces the greatest impurity reduction or the largest information gain is chosen. Entropy and Gini impurity are two popular impurity measurements. **[39]** The data is recursively partitioned using the decision tree method according to the chosen features. The dataset is divided into subsets based on various feature values, and each subset's child nodes are created. Until a stopping requirement is satisfied, such as a maximum depth, a minimum number of samples per leaf, or a minimum reduction in impurities, this procedure is continued.

Following the path from the root node to a leaf node based on the feature values of the input data allows the decision tree to be used to generate predictions once it has been constructed. Each leaf node in the tree correlates to a prediction or class label, whereas each internal node in the tree refers to a judgement based on a feature.Models based on decision trees are quite comprehensible. The decision-making process may be easily understood and visualised thanks to the tree's structure. In addition to helping users understand the significance of various features in the prediction task, decision trees can also shed light on how the model behaves.

Both category and numerical features can be handled by decision trees. The method chooses the best split points for numerical features while creating distinct branches for categorical features.Decision trees can manage data with missing values. Based on the algorithm's best split prediction, missing values during the splitting process can be sent to either of the branches. **[15]** Decision trees can overfit, which causes the model to grow overly complicated and capture noise or unimportant patterns in the training data. Overfitting can be reduced by methods like pruning, limiting the maximum tree depth, and regulating the minimum amount of data per leaf.

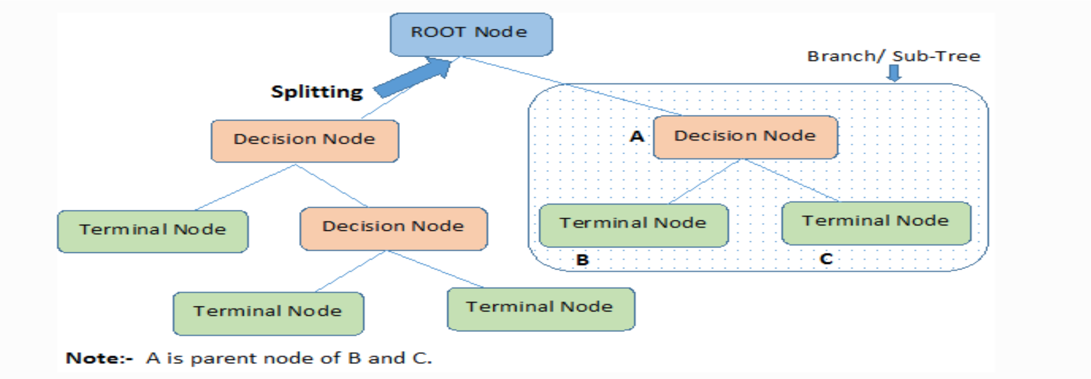


Fig 4.2.5: Decision Tree Algorithm Workflow

4.2.8 RANDOM FOREST:

The collective learning family of machine learning algorithms includes the well-known Random Forest algorithm. A strong and reliable predictive model is produced by combining various decision trees. The advantages of Random Forest include its high accuracy, handling of huge datasets, and resistance to overfitting. Each decision tree in a Random Forest ensemble is trained independently using a random subset of the training data. Bagging (bootstrap aggregating), a random sampling technique, contributes to the diversification of the trees and the reduction of model variance.**[10]**

In addition to randomly selecting features, Random Forest also randomly samples the training data. Only a portion of features are taken into account at each split in a decision tree. The trees' decorrelation and independence are increased by the selection of random features. Random Forest aggregates all of the different trees' forecasts during prediction and generates the final prediction based on majority voting (classification) or average (regression) as appropriate. This collective strategy lessens the impact of individual tree faults while enhancing the model's ability to generalise.

Comparing Random Forest to individual decision trees, the latter is less prone to overfitting. It lessens the chance of detecting noise and outliers in the data by mixing numerous trees and adding unpredictability to the training process. The forecasts are rounded off and strengthened by the ensemble averaging.

A measure of feature importance, offered by Random Forest, identifies the relative weights given to various features in the prediction job. **[11]** The algorithm evaluates each feature's contribution by calculating how much the prediction accuracy slips when it is randomly permuted.

This knowledge can aid in choosing features, locating important variables, and comprehending the data. When there are more features compared to samples in high-dimensional data, Random Forest can handle it well. At each split, it automatically selects features by taking into account a random subset of features. This characteristic makes it appropriate for scenarios involving feature engineering or tasks with numerous characteristics. Using the samples taken directly from the bag, Random Forest calculates the model's performance. Each tree is only exposed to a fraction of the training data during training, known as the out-of-bag samples.

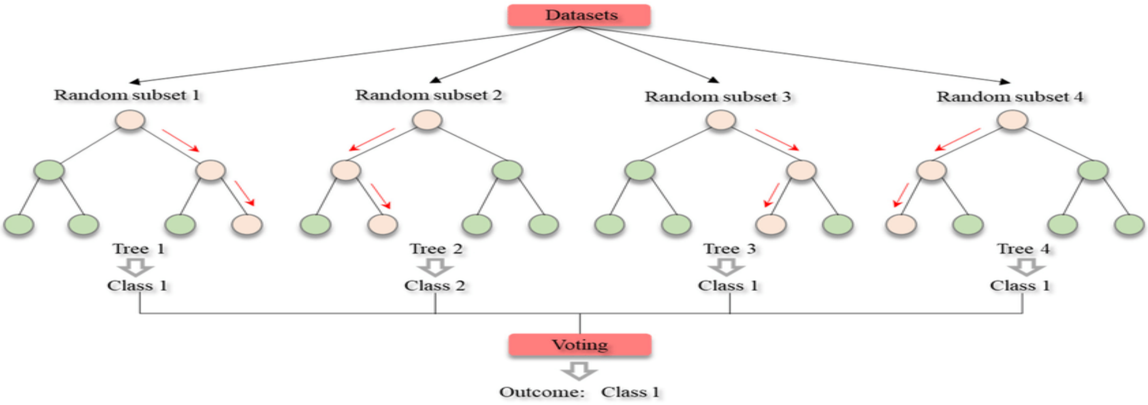


Fig 4.2.6: Random Forest Algorithm Workflow

4.2.9 LOGISTIC REGRESSION

The chance of a binary outcome—in this case, survival or not—based on one or more input factors (such as passenger class, age, and sex) is predicted using a linear model called logistic regression. The linear prediction is then converted to a probability value between 0 and 1 using the logistic function.Utilising the supplied training data, the logistic regression model can be trained, and a method like gradient descent can be used to optimise the model's coefficients. After the model has been trained, it may be used to forecast the likelihood that new passengers would survive based on their input parameters

Logistic regression can be used in the Learning from Disaster project to forecast whether a passenger would survive or not depending on factors like their age, sex, and passenger class. **[7]** The objective is to create a model that can correctly predict, depending on the characteristics of new passengers, whether they will survive. Missing values are removed from the dataset, the input variables are normalised, and categorical variables are encoded as numerical values. The test data are then used to assess the model's performance once the logistic regression model has been trained using the training data. Metrics including accuracy, precision, recall, and F1 score are frequently used to assess the model's performance

4.3 COMPARING THE ATTRIBUTES

4.31. Age v/s survival ,this showcase the survival and the demise ratio based on the age group of the people in the ship which shows age is directly/indirectly proportional of demise or survival.

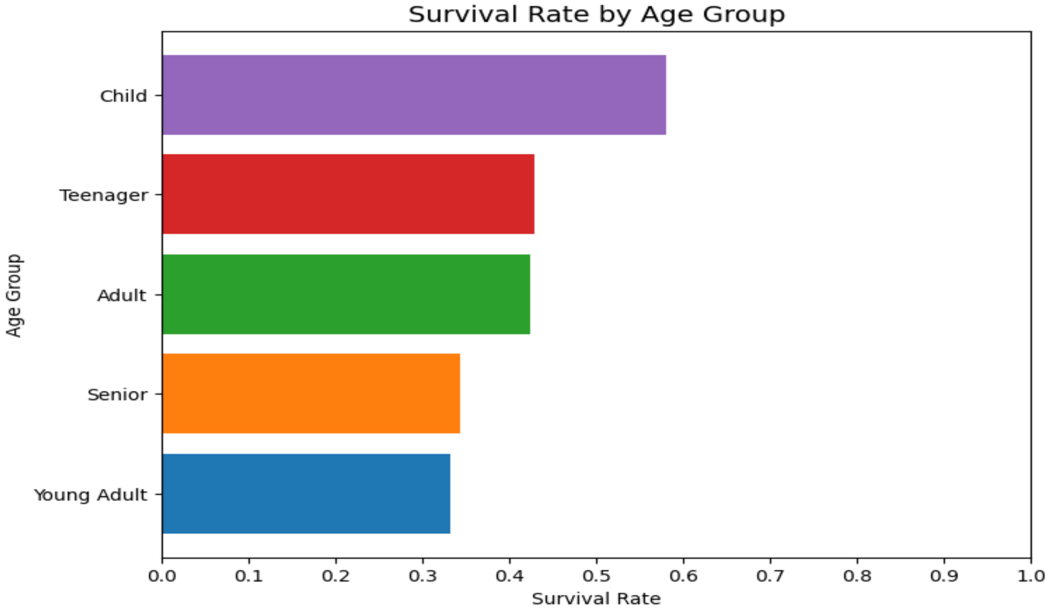


Fig 4.3.1: Age Group v/s Survival Rate

4.3.2 Sex v/s survival , this represents how the demise and survival rate is been affected with the age aspect. This shows the probability of survival or demise depending on the gender.

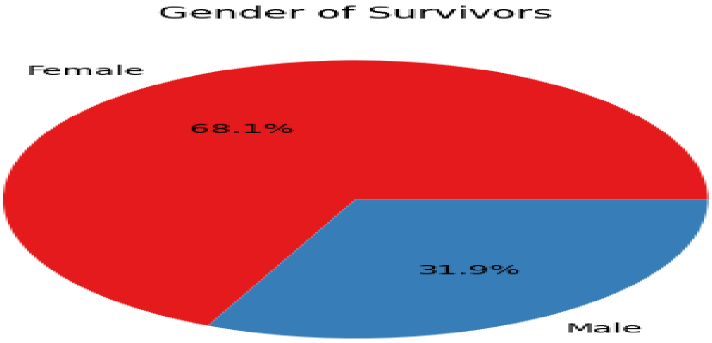


Fig 4.3.2: Sex v/s Survival Pie chart

4.3.3 The below histogram represents count of survived people having only one siblings/spouses is greater than who died.

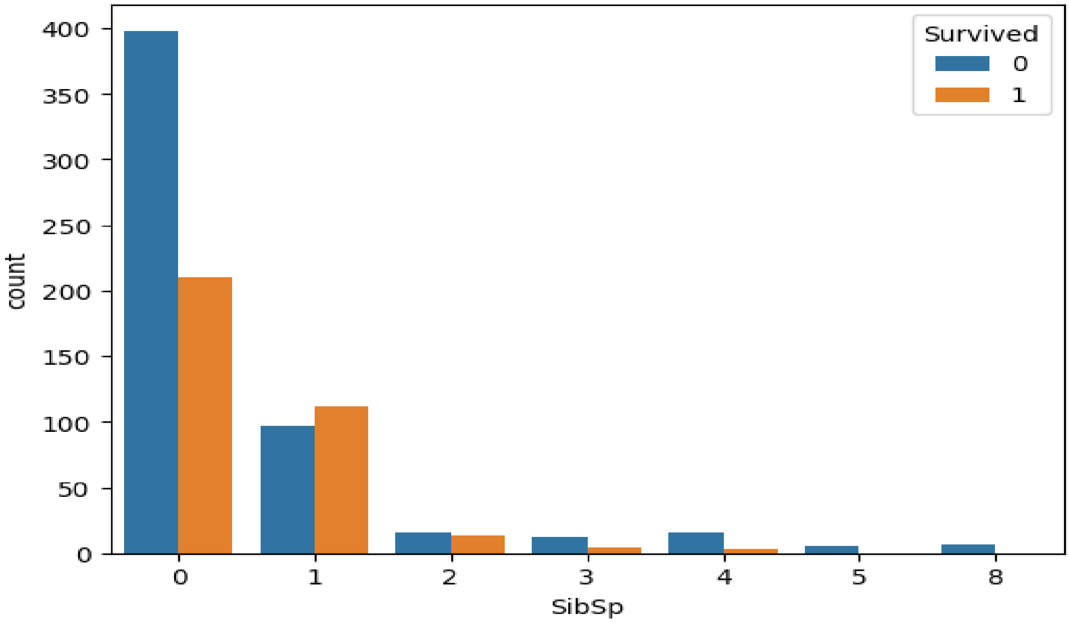


Fig 4.3.3: Siblings/spouse v/s No of people died

4.3.4 The below histogram shows count of survived people having only one parents/children is greater than who died.

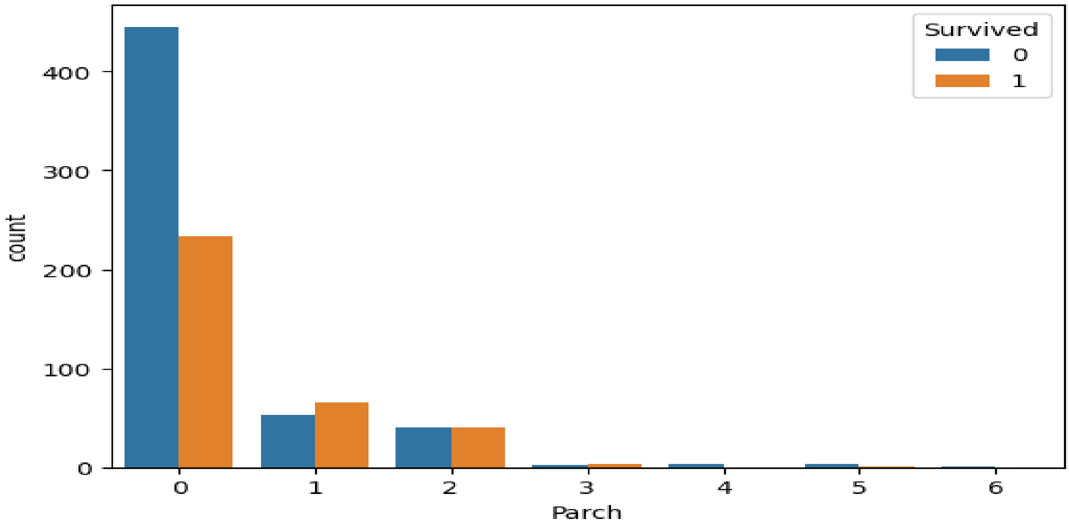


Fig 4.3.4: one parent/children v/s No of people died

4.3.5 Embarkation location v/s survival rates .The number of embarked survivals is greater than who number of who died.

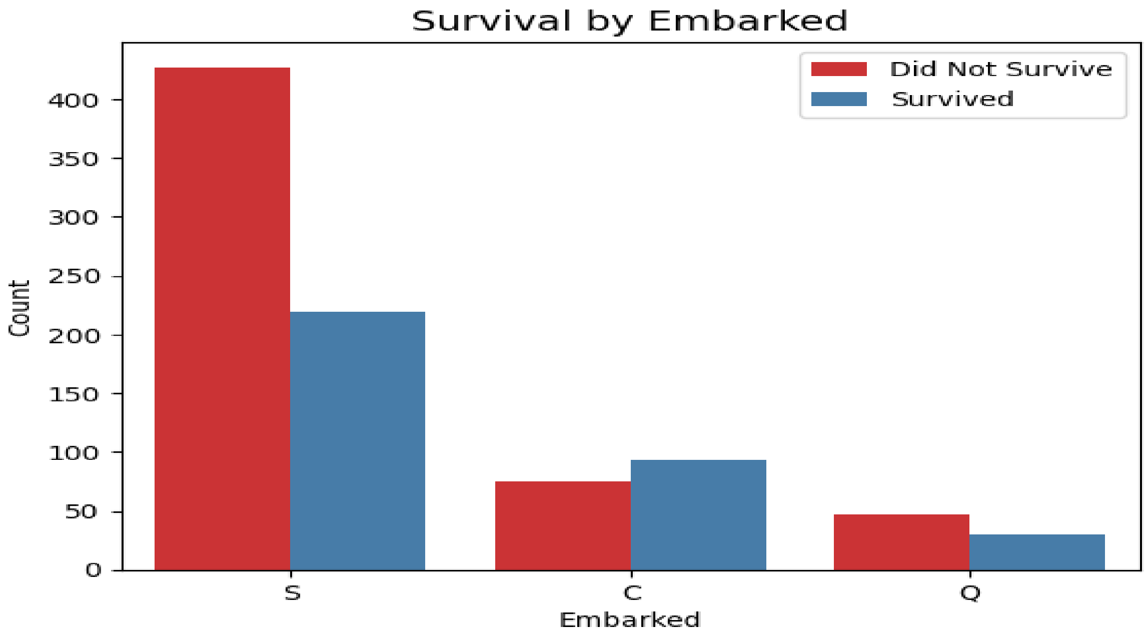


Fig 4.3.5: Location v/s Survival

4.3.6 Pclass v/s Survival rate**.** Passengers in 1st class are more likely to survive and Passengers in 3d class are more likely to die.

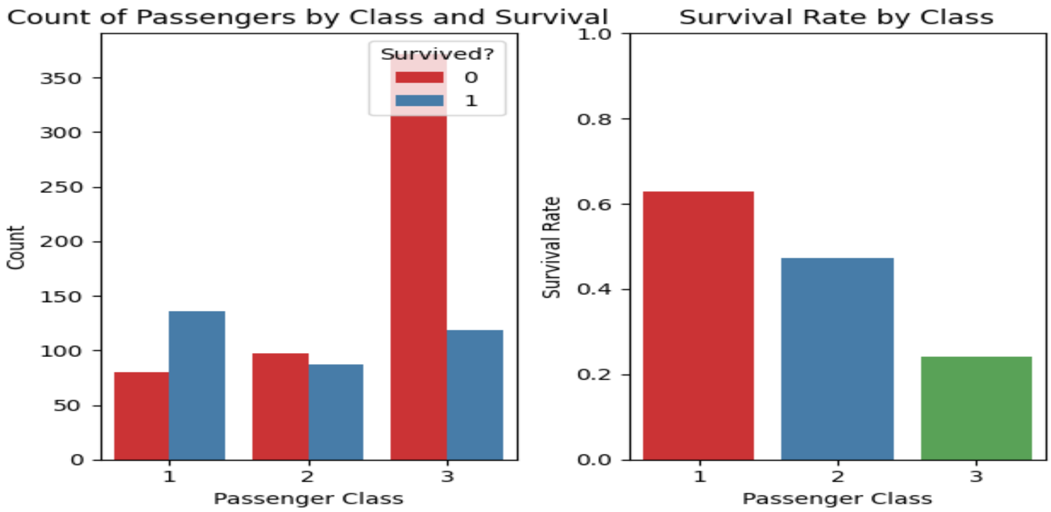


Fig 4.3.6: Passenger class v/s Survival Rate

4.4 ASSESSMENT OF MODEL

4.4.1 ERROR MATRIX

A method to verify the accuracy of the arrangement model's operation is the use of a disorderly network. When compared to the actual outcome of the information, it provides the actual number of forecasts that were accurate or inaccurate. The request is represented by the lattice n*n, where n is the total number of attributes. The information contained in the lattice is typically used to evaluate how well these models are implemented.

Affectability: The term "affectability" describes the quantity of true positives that can be accurately identified and is correlated with the rate of fake negatives. Sensitivity is defined as the ratio of real positive to real negative plus fake positive.

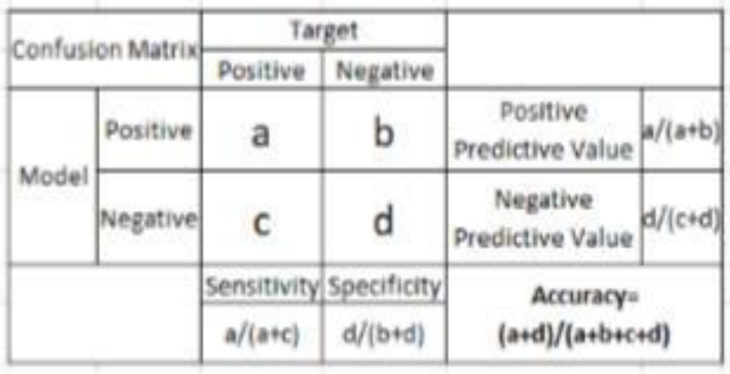
Particularity: It measures the extent of effectively recognised negatives and is inversely proportional to the false positive rate. Genuine negatives/(genuine negatives plus fake positives) is a measure of specificity.

Positive predictive value: The exhibition component of the factual test is provided. It is a combination of genuine positive (event that creates true anticipation and subject outcome is also true) and fake positive (occasion that creates false expectation and subject outcome is also false).

Negative predicted value: It is the sum of genuine negatives and bogus negatives (the event that creates bogus anticipation and subject outcome is certain) and the percentage of real negatives (the occasion that creates fraudulent expectation and subject outcome is additionally fake).

Exactness: It indicates the percentage of the model's or calculation's right expectations. "1.0" is the best value, and "0.0" is the value that is the most obviously bad.

4.4.2 **CONFUSION MATRIX**



Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	Positive Predictive Value	a/(a+b)
	Negative	c	d	Negative Predictive Value	d/(c+d)
		Sensitivity	Specificity	**Accuracy=**	
		a/(a+c)	d/(b+d)	**(a+d)/(a+b+c+d)**	

Fig 4.4.1: Confusion matrix

Mathematical calculations are carried out in "R," and accuracy errors in every model are discovered. Here are the accuracy levels that we typically attain for each model.

Chapter 5 Code

- Using different libraries, this piece of code handles data pretreatment and exploration duties. For the purposes of manipulating, analysing, and displaying data, it imports NumPy, Pandas, Matplotlib, and Seaborn. During execution, warnings can be disregarded using the warnings module. Training and test data are read from CSV files by the code.

1. It carries out tasks including displaying data using head() and tail(), examining the shape and information of the data, addressing missing values, and carrying out data transformations. Functions like kdeplot(), boxplot(), and hist() are used to build visualisations. Age group classification and name extraction using new columns are both done. By grouping comparable titles and replacing less prevalent ones, the data is made ready for study. Printing is done for the updated datasets.

EDA

1. The train_data dataset's correlation matrix is computed and displayed by this code. It produces a heatmap with annotations using Seaborn's heatmap() tool.It counts the number of survivors and computes the total number of passengers to determine the overall survival rate. By dividing the number of survivors by the total number of passengers and multiplying the result by 100, the survival rate is determined.

2. The result shows a moderate overall survival rate of 38.38%, which is indicative of passenger survivability. This research sheds light on the connections between several dataset factors and the overall survival result.

➢ Does passenger class after Survival Rates ?

1. In the train_data dataset, this line of code generates two graphs according to passenger class and survivability. Using the countplot() function from Seaborn, the first plot displays the distribution of passengers among several classes (Pclass), distinguished by survival (Survived). A legend and axis labels are also included.

2. The survival rate by passenger class is shown in the second plot, a bar graph. The mean survival rate for each class is computed using the groupby() function, and the results are then visualised using the barplot() function. The range of the survival rate is shown by the y-axis's restriction from 0 to 1.

➢ How does the survival rate differ by gender ?

1. In the train_data dataset, this function determines and displays the survival rate according to gender.

2. By dividing the data into groups based on "Sex" and taking the mean of the "Survived" column, it first determines the survival rate for each gender. The outcome is rounded to the nearest decimal point.

3. The survival rate by gender is then shown in a bar plot using Seaborn's barplot() function. Gender is represented on the x-axis, and survival rate is represented on the y-axis.

4. The count of men and women among the survivors is then determined by excluding everyone but the survivors from the datasetFinally, a pie chart is created to show the survivors' gender distribution.

5. Gender clearly played a key influence in influencing survival outcomes, as the data shows that women have a better chance of surviving than men.

➢ Any relationship between Age and Survival?

1. On the train_data dataset, this bit of code performs numerous visualisations and feature engineering procedures.Using Seaborn's violinplot() method, the first plot is a violin plot. It shows the age distribution (Age) for each category of survivors (Survived).

2. The survival rate by age group is then displayed on a bar plot. By age group (AgeGroup), the survival rates are categorised and represented using horizontal bars.To visualise the counts of parents/children (Parch) and siblings/spouses (SibSp) onboard, distinct by survival status (Survived), two count plots are developed.

3. Another count plot shows the survival rate according to the port of embarkation (Embarked), with various bars standing in for the various survival rates at each embarkation point.The survival rate by port of embarkation is shown on a bar graph.

4. To compare the fare (Fare) distribution between survivors and non-survivors, a box plot is made.The counts of passengers with various titles (Title), separated by survival status, are displayed on a count plot.

5. By developing new features, feature engineering is carried out. The number of wives and children plus one is added to the number of siblings to determine the family size (FamilySize).
6. The last few lines display the train_data and test_data dataset heads and save the passenger IDs from the test_data for future use.The research sheds light on the dataset's age distribution, age-specific survival rates, family size, port of embarkation, fare distribution, and title distribution of passengers.

➢ SUPPORT VECTOR CLASSIFIER

1. The Random Forest Classifier, SVC, KNeighbors Classifier, DecisionTree Classifier, and Logistic Regression classifiers are just a few of the classifiers imported into this code sample from the scikit-learn library. It also imports a number of evaluation measures and tools for choosing and preparing models.The characteristics and target variable for the classification task are first defined in the code. Using the train_test_split function, it then divides the data into training, validation, and test sets.

2. The training data is then preprocessed by utilising the pd.get_dummies function to encode categorical variables using one-hot encoding. In accordance, the test_data is likewise preprocessed. To maintain uniformity, the columns of the training and test sets of data are lined up.

3. Using the MinMaxScaler from sklearn.preprocessing, the training, validation, and test data's numerical columns are scaled.

4. On the preprocessed training data, a support vector classifier (SVC) model with a linear kernel is developed and trained. The SVC model is used as the estimator in the SelectFromModel function for feature selection.

5. The feature selection step's main features are printed. The accuracy, confusion matrix, and classification report are then calculated and produced using the SVC model to generate predictions on the validation data.

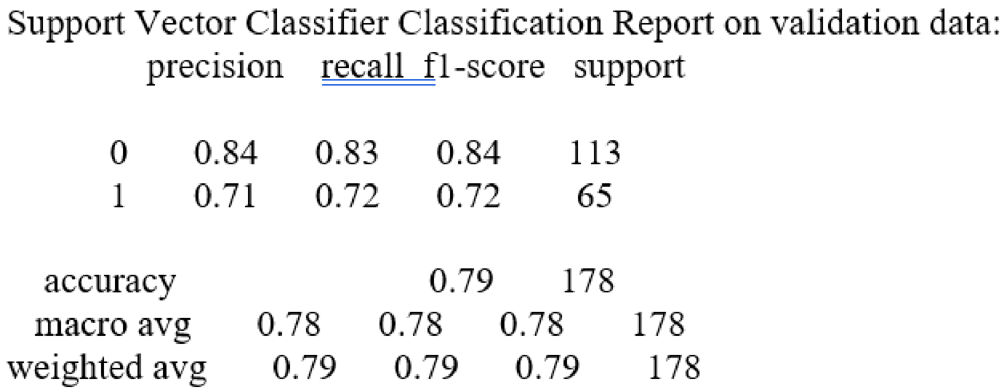
6. Finally, predictions are made on the test data using the trained SVC model, and the predicted labels are saved in the y_pred_svc variable.

7. In conclusion, this code use the SVC classifier to accomplish feature selection, model training, and evaluation. It serves as an example of how to prepare data, fit a

model, and make predictions.

➢ ACCURACY :

1. The required classifiers and evaluation metrics are imported by this code snippet. It uses SVC to carry out feature selection and outputs the key features. On previously cleaned up training data, a support vector classifier is trained. In addition to calculating accuracy, confusion matrix, and classification report, it then predicts labels for the validation data. Finally, it uses the trained SVC model to make predictions based on the test data.



```
Support Vector Classifier Classification Report on validation data:
              precision    recall  f1-score   support

           0       0.84      0.83      0.84       113
           1       0.71      0.72      0.72        65

    accuracy                           0.79       178
   macro avg       0.78      0.78      0.78       178
weighted avg       0.79      0.79      0.79       178
```

Fig 5.1: SVC accuracy

➢ RANDOM CLASSIFIER :

1. With 100 estimators and a maximum depth of 5, a Random Forest Classifier model is generated in the code. The preprocessed training data are used to train the model.

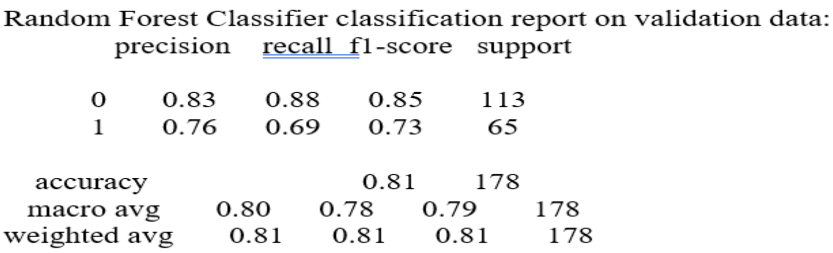
2. The Random Forest Classifier is used to choose the features, and the most significant features are printed. On the basis of the validation data, predictions are produced, and the accuracy, confusion matrix, and classification report of the model are calculated and reported. The Random Forest Classifier's performance on the validation data is printed.

3. The trained model is then used to make predictions on the test data, and the outcomes are saved in the variable "y_pred_rfc".

➢ ACCURACY :

1. With 100 estimators and a top depth of 5, the Random Forest Classifier model is trained. 'AgeClass', 'FamilySize', 'Fare', 'Pclass', 'Sex_female', 'Sex_male', 'Title_Miss', 'Title_Mr', and 'Title_Mrs' are some of the key traits that are found via feature selection.

2. An accuracy of 80.9% is obtained once the model has been assessed using the validation data. Using the confusion matrix, it can be shown that there are 45 true positives, 14 false positives, 20 false negatives, and 99 true negatives. On the whole, the performance of the Random Forest Classifier on the validation data is encouraging.



```
Random Forest Classifier classification report on validation data:
              precision    recall  f1-score   support

           0       0.83      0.88      0.85       113
           1       0.76      0.69      0.73        65

    accuracy                           0.81       178
   macro avg       0.80      0.78      0.79       178
weighted avg       0.81      0.81      0.81       178
```

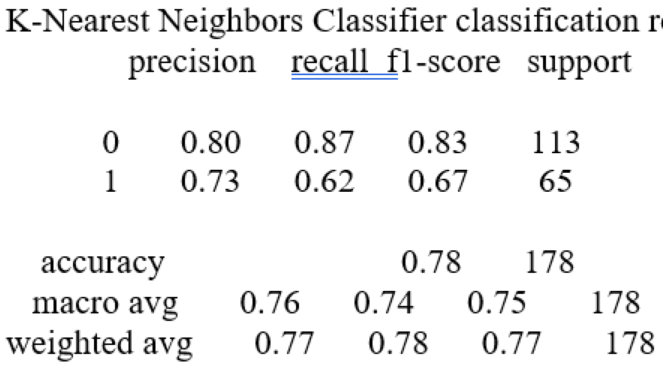
Fig 5.2: Random Forest Classifier accuracy

➢ K - NEAREST NEIGHBOUR :

1. With five neighbours, the K-Nearest Neighbours Classifier model is trained.Additional evaluation metrics for each class, such as accuracy, recall, and F1-score, are included in the classification report. The trained model is then employed to forecast the labels for the test data.

➢ ACCURACY :

1. On the validation data, the K-Nearest Neighbours Classifier has an accuracy of 77.53%. According to the confusion matrix, 98 of the 153 incidents were correctly classified as negatives, 40 as positives, while 25 were incorrectly labelled as negatives and 15 as positives.



```
K-Nearest Neighbors Classifier classification report on validation data:
              precision    recall  f1-score   support

           0       0.80      0.87      0.83       113
           1       0.73      0.62      0.67        65

    accuracy                           0.78       178
   macro avg       0.76      0.74      0.75       178
weighted avg       0.77      0.78      0.77       178
```

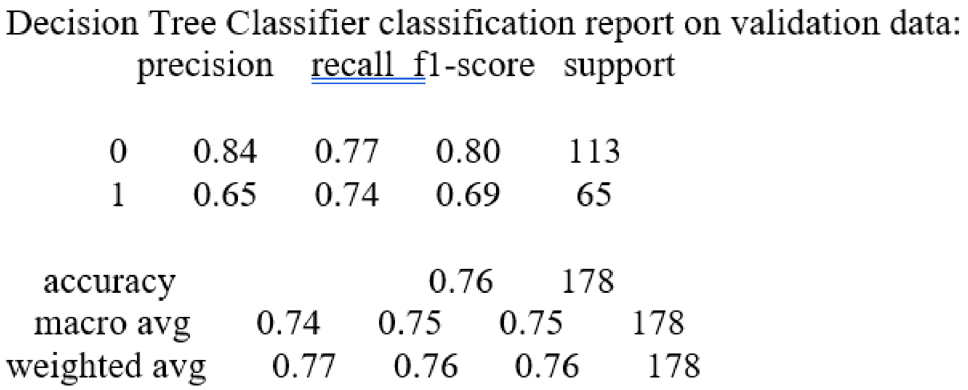
Fig 5.3: KNN accuracy

➢ DECISION TREE MODEL :

1. This code uses the Titanic dataset to fit a Decision Tree Classifier model. To determine the most crucial features, feature selection is done using the SelectFromModel function.
2. The crucial details are then printed. On the validation dataset, the model is then utilised to create predictions, and the accuracy, confusion matrix, and classification report are printed.
3. The model is then employed to make predictions on the test dataset, with the outcomes being stored in y_pred_dt.
4. The code offers a method for categorising Titanic passengers as either survivors or non-survivors using the decision tree paradigm. On the validation dataset, the model's accuracy is also presented.
5. This model can be utilised to improve understanding of the variables that affected Titanic survivability.

➢ ACCURACY :

1. The Titanic dataset is used to train the Decision Tree Classifier model, which includes crucial characteristics like AgeClass, FamilySize, Fare, Pclass, and Sex_female. The confusion matrix shows 48 correct predictions for survivors and 87 correct predictions for non-survivors, giving a validation data accuracy of 75.84%.



```
Decision Tree Classifier classification report on validation data:
              precision    recall  f1-score   support

           0       0.84      0.77      0.80       113
           1       0.65      0.74      0.69        65

    accuracy                           0.76       178
   macro avg       0.74      0.75      0.75       178
weighted avg       0.77      0.76      0.76       178
```

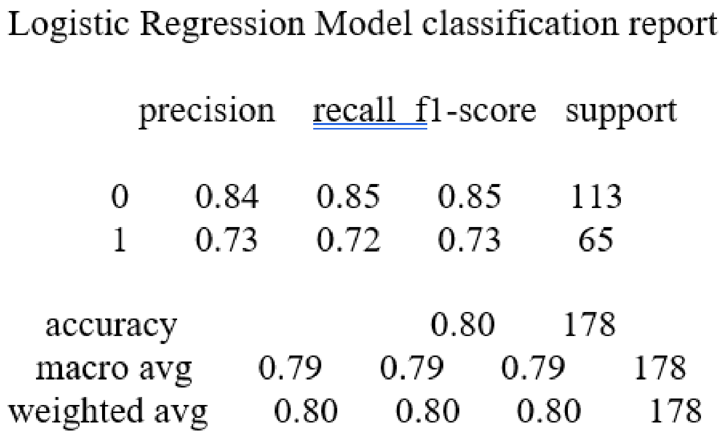
Fig 5.4: Decision Tree accuracy

➢ LOGISTIC REGRESSION :

1. On the Titanic dataset, the Logistic Regression model is trained using features like AgeClass, Fare, Pclass, Sex_female, and Sex_male. The model is assessed using the validation data, giving information on how well it performs. Through feature selection, the crucial aspects of the model are identified.
2. The classification report offers specific metrics for both the survivor and non-survivor groups, including precision, recall, and F1-score. Finally, survival outcomes on the test data are predicted using the trained Logistic Regression model.

➢ ACCURACY :

1. Important features including AgeClass, Deck_C, Deck_E, Deck_G, FamilySize, Pclass, Sex_female, Sex_male, Title_Master, Title_Mr, and Title_Mrs are identified by the Logistic Regression model.
2. With 96 accurate predictions for non-survivors and 47 accurate predictions for survivors, it obtains an accuracy of 80.34% on the validation data.



```
Logistic Regression Model classification report on validation data:

              precision    recall  f1-score   support

           0       0.84      0.85      0.85       113
           1       0.73      0.72      0.73        65

    accuracy                           0.80       178
   macro avg       0.79      0.79      0.79       178
weighted avg       0.80      0.80      0.80       178
```

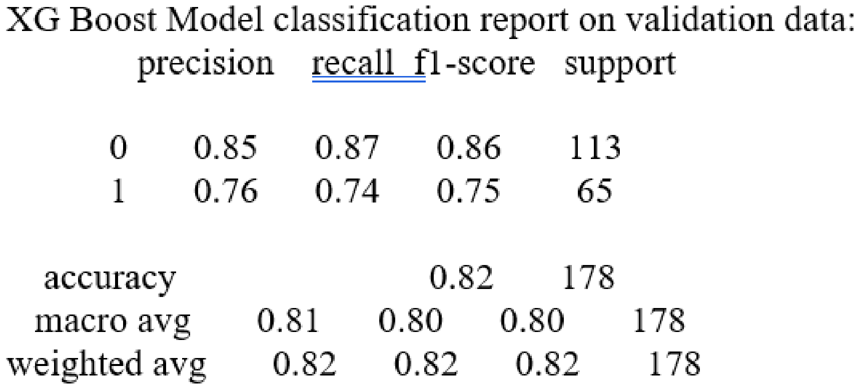
Fig 5.5: Logistic Regression accuracy

➢ XGBOOST AS XGB :

1. The programme implements an XGBoost model with a maximum depth of three, a learning rate of 0.05, and 100 estimators. The technique of feature selection is used to pinpoint crucial features. The provided training data is subsequently used to train the model.
2. On the basis of the validation data, predictions are formed, and the accuracy, confusion matrix, and classification report of the model are used to assess its performance.
3. Finally, predictions are made on the test data using the trained model.

➢ ACCURACY :

1. On the supplied data, the XGBoost model is trained using 100 estimators, a learning rate of 0.05, and a maximum depth of 3. 'Pclass', 'Sex_female', 'Title_Mr', and 'Title_Other' have been noted as key traits.
2. A confusion matrix reveals 98 accurate predictions for the non-survived class and 48 accurate predictions for the survived class, giving the model a validation data accuracy of 82.02%.



```
XG Boost Model classification report on validation data:
              precision    recall  f1-score   support

           0       0.85      0.87      0.86       113
           1       0.76      0.74      0.75        65

    accuracy                           0.82       178
   macro avg       0.81      0.80      0.80       178
weighted avg       0.82      0.82      0.82       178
```

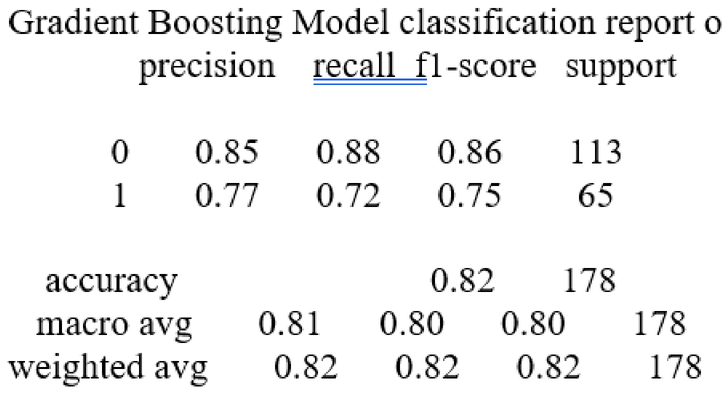
Fig 5.6: XG BOOST accuracy

➢ GRADIENT BOOSTING CLASSIFIER :

1. On the supplied data, a Gradient Boosting model with 100 estimators, a learning rate of 0.1, and a maximum depth of 3 is trained. The code sample omits explicitly referencing the significant characteristics determined by feature selection.
2. The confusion matrix and classification report are among the outcomes of the model evaluation using the validation data. Finally, predictions on the test data are made using the trained model.

➢ ACCURACY :

1. Important features including 'AgeClass', 'Fare', 'Pclass', 'Sex_male', and 'Title_Mr' are used by the Gradient Boosting model, which was trained with 100 estimators, a learning rate of 0.1, and a maximum depth of 3.
2. A confusion matrix showing 99 correct predictions for the non-survived class, 14 incorrect predictions for the non-survived class, 18 incorrect predictions for the survived class, and 47 correct predictions for the survived class is produced by the model, which achieves an accuracy of 82.02% on the validation.

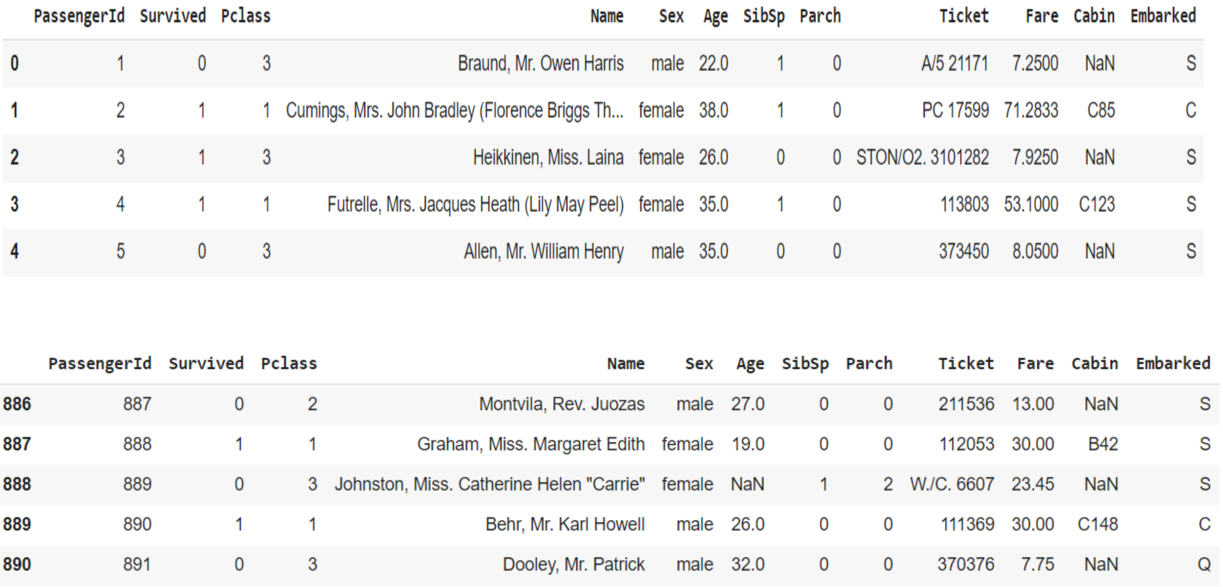


```
Gradient Boosting Model classification report on validation data:
              precision    recall  f1-score   support

           0       0.85      0.88      0.86       113
           1       0.77      0.72      0.75        65

    accuracy                           0.82       178
   macro avg       0.81      0.80      0.80       178
weighted avg       0.82      0.82      0.82       178
```

Fig 5.7: Gradient Boosting accuracy

Chapter 6 Results

1. Loading Data

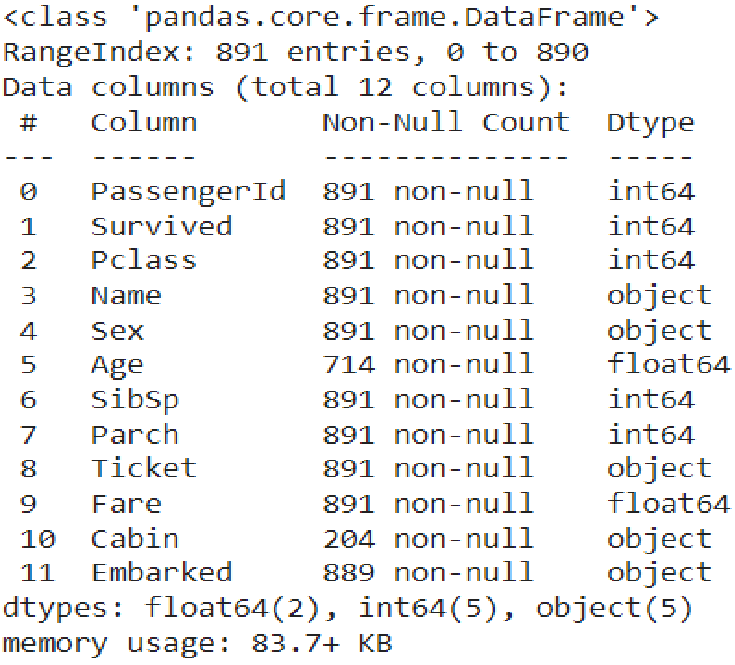


	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	NaN	Q

Fig 6.1: Loading Data into code

2. TABLE INFORMATION:



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

Fig 6.2: Table Information

3. SURVIVAL RATIO BY AGE:

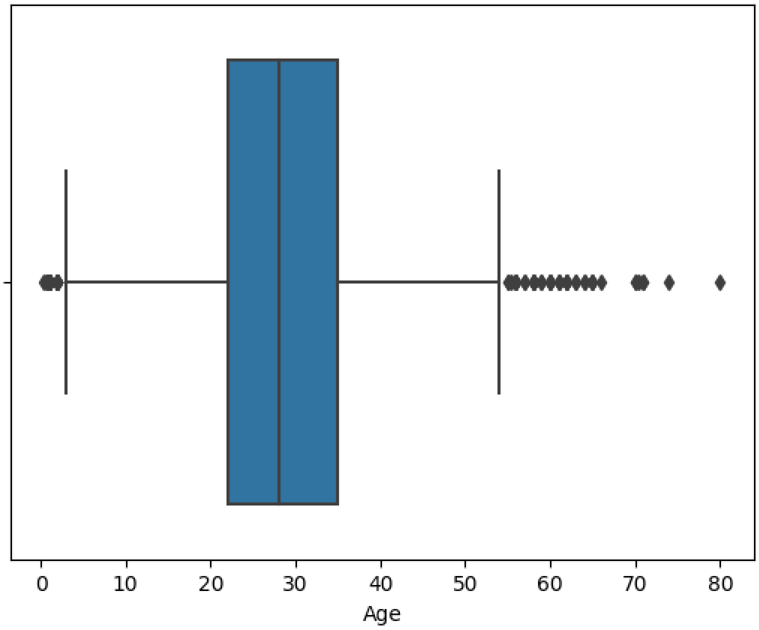


Fig 6.3: Survival Ratio by age

4. SURVIVAL RATE BY GENDER:

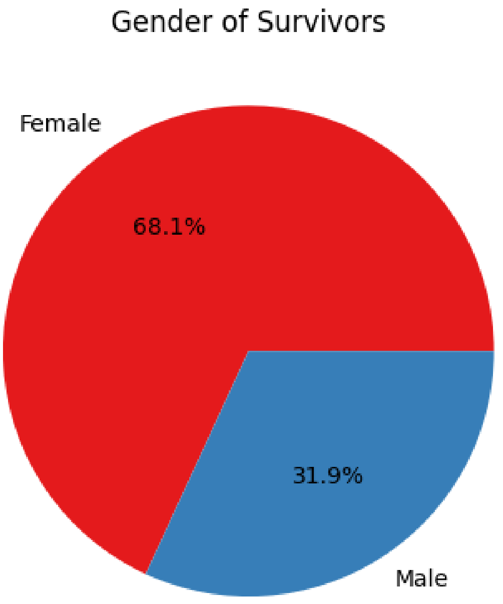
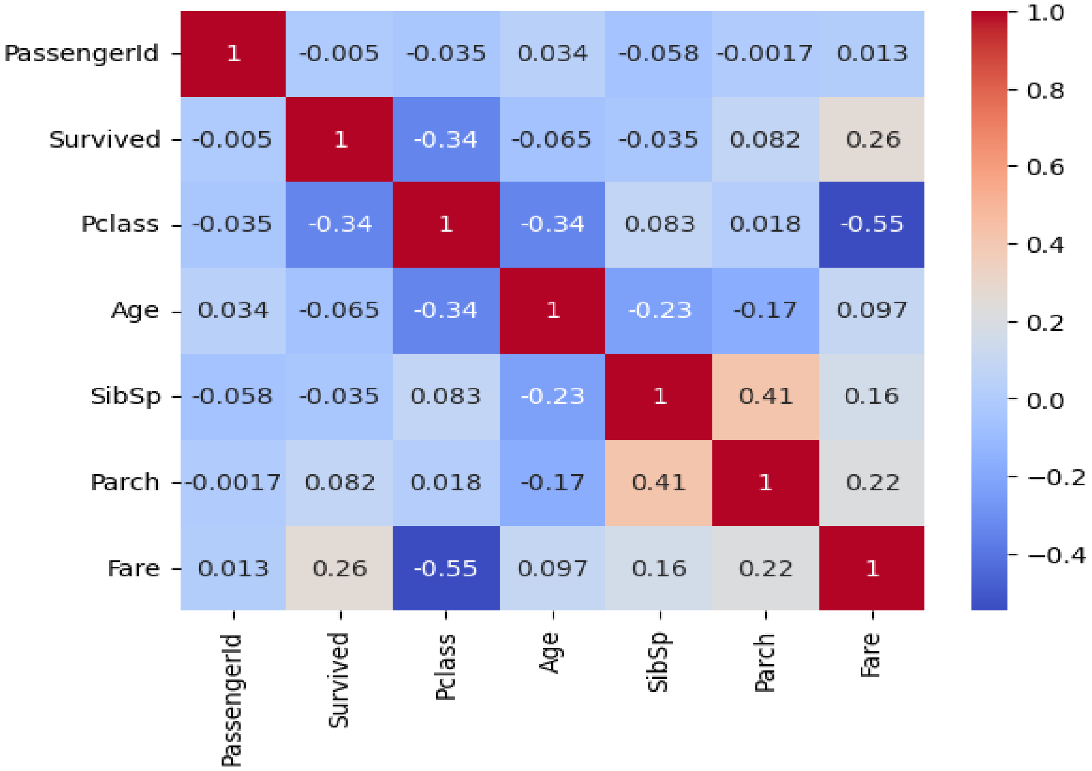


Fig 6.4: Survival Ratio by gender

5. CORRELATION OF FEATURES:



	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1	-0.005	-0.035	0.034	-0.058	-0.0017	0.013
Survived	-0.005	1	-0.34	-0.065	-0.035	0.082	0.26
Pclass	-0.035	-0.34	1	-0.34	0.083	0.018	-0.55
Age	0.034	-0.065	-0.34	1	-0.23	-0.17	0.097
SibSp	-0.058	-0.035	0.083	-0.23	1	0.41	0.16
Parch	-0.0017	0.082	0.018	-0.17	0.41	1	0.22
Fare	0.013	0.26	-0.55	0.097	0.16	0.22	1

Fig 6.5: Correlation Matrix

6. AGE DISTRIBUTION GRAPH:

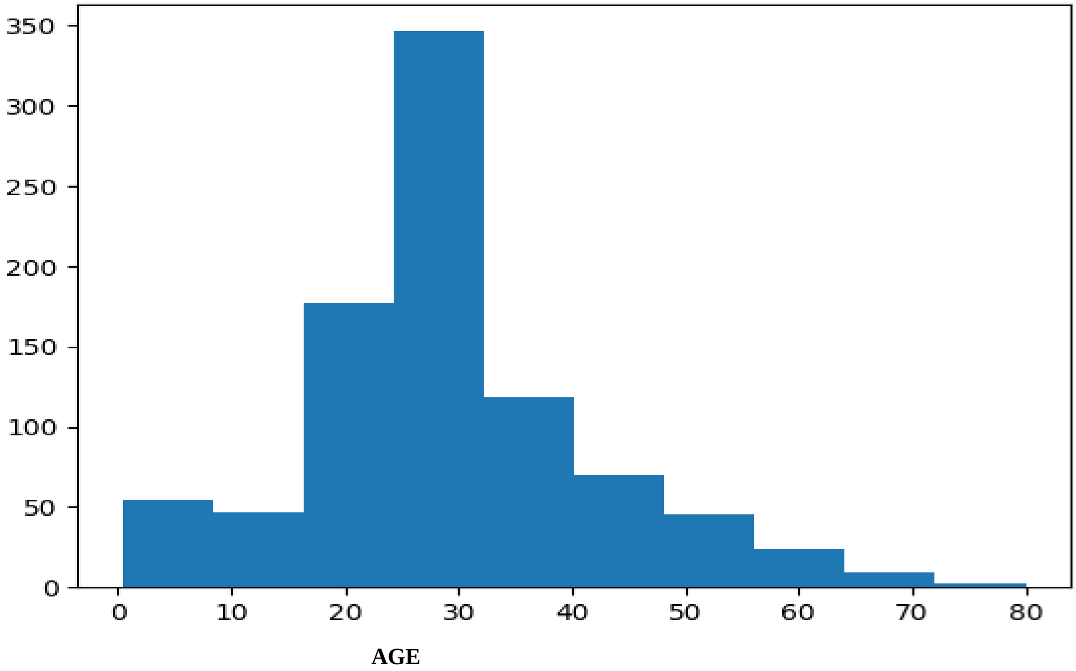


Fig 6.6: Age Distribution

6.1 ACCURACY:

Algorithm Used	Existing system Accuracy	Proposed System Accuracy
Logistic Regression	75%	80.34%
Support Vector Classifier	N/A	79.21%
K-Nearest Neighbor	N/A	77.53%
XG-Boost	N/A	82.02%
Gradient Boosting Classifier	N/A	82.02%
Decision tree	71%	75.84%
Random forest	75%	80.9%

Chapter 7 Conclusion and Future Scope

Conclusion:

This investigation was motivated by a desire to learn about, comprehend, and use the cart calculation. The vision showcase competition participant Titanic: ai from catastrophe serves as a platform for several vision evaluation techniques. This discussion focused on developing a prediction model to forecast passenger endurance aboard the RMS Titanic. It was quite time-consuming and labor-intensive to clean, organize, and categorise the informational components. The cart approach and its extensions, such as packing and uneven terrain, were employed successfully. Using the expected precision offered by Kaggle, we assessed the efficiency of each strategy in accurately arranging the traveller endurance in the deadly boat catastrophe. There were two distinct sets of findings.

We believe that the key test was the one that only required minimal adjustments based on the data provided by Kaggle. The second study re-envisioned the most perplexing elements, such as lodge and name, and utilised a relapse tree to estimate the missing age values. It was decided to display the primary grouping tree. There were no appreciable changes in accuracy between the approaches that we looked into. We were unable to get an accuracy rate that was significantly higher than the level of fundamental conviction with any feature combination, though. a Bayes classifier that only considers gender.

Future Scope:

The results we obtained can be expanded upon in future study by investigating new variables and improving the models to improve their prediction power. Additionally, investigating other machine learning algorithms or cutting-edge methods may enhance the precision of survival forecasts and offer new insights.In the end, the information gleaned from this study can help in unravelling the underlying causes that influenced Titanic survivorship, perhaps guiding future planning and reaction plans for disasters

References

[1] Kaggle.com, ‘Titanic:Machine Learning form Disaster’,[Online]. Available: http://www.kaggle.com/. [Accessed: 10- Feb- 2017].

[2] Eric Lam, Chongxuan Tang, “Titanic Machine Learning From Disaster”, LamTang-TitanicMachineLearningFromDisaster, 2012.

[3] Cicoria, S., Sherlock, J., Muniswamaiah, M. and Clarke, L, “Classification of Titanic Passenger Data and Chances of Surviving the Disaster”, Proceedings of Student-Faculty Research Day, CSIS, pp. 1-6, May 2014

[4] Vyas, Kunal, Zeshi Zheng, and Lin Li, “Titanic-Machine Learning From Disaster”, Machine Learning Final Project, UMass Lowell, pp. 1-7, 2015.

[5] Mikhael Elinder.(2012). ‘Gender, social norms, and survival in maritime disasters’, [Online]. Available: http://www.pnas.org/content/109/33/13220.full. [Accessed: 8- March - 2017].

[6] Frey, B. S., Savage, D. A., and Torgler, B, “Behavior under extreme conditions: The Titanic disaster”, The Journal of Economic Perspectives, 25(1), pp. 209-221, 2011.

[7] Trevor Stephens. (2014), ‘Titanic: Getting Started With R - Part 3: Decision Trees’, [Online]. Available: http://trevorstephens.com/kaggletitanic-tutorial/r-part-3-decision-trees/. [Accessed: 11- March- 2017].

[8] Trevor Stephens. (2014). ‘Titanic: Getting Started With R - Part 3: Decision Trees’, [Online]. Available: http://trevorstephens.com/kaggletitanic-tutorial/r-part-3-decision-trees/. [Accessed: 8- March - 2017].

[9] Rex Morgan. (2016). Titanic [Online]. Available:http://www.because.uk.com/wp-content/uploads/Because2016-03w.pdf. [Accessed: 9- March - 2017].

[10] Jason Brownlee. (2014). How to implement Nave Bayes in Python from scratch [Online]. Available: http://machinelearningmastery.com/naivebayes-classifier-scratch-python/. [Accessed: 9- March - 2017].

[11] Santos, K.C.P, Barrios, E.B, “Improving Predictive accuracy of logistic regression model using ranked set sample,” Communication in statistic.

[12] Zhao, H.; Chen, B.; Zhu, C. Decision Tree Model for Rockburst Prediction Based on Microseismic Monitoring. *Adv. Civ. Eng.* **2021**, *2021*, 8818052.

[13] Feng, X.-T.; Yashun, X.; Guangliang, F. Mechanism, warning and dynamic control of rockburst evolution process. In Proceedings of the ISRM Regional Symposium—7th Asian Rock Mechanics Symposium, Seoul, Korea, 15–19 October 2012.

[14] Sun, Y.; Li, G.; Zhang, J.; Huang, J. Rockburst Intensity Evaluation by a Novel Systematic and Evolved Approach: Machine Learning Booster and Application. *Bull. Eng. Geol. Environ.* **2021**, *80*, 8385–8395.

[15] Cai, M. Principles of Rock Support in Burst-Prone Ground. *Tunn. Undergr. Space Technol.* **2013**, *36*, 46–56.

[16] Cai, X.; Cheng, C.; Zhou, Z.; Konietzky, H.; Song, Z.; Wang, S. Rock Mass Watering for Rock-Burst Prevention: Some Thoughts on the Mechanisms Deduced from Laboratory Results. *Bull. Eng. Geol. Environ.* **2021**, *80*, 8725–8743.

[17] Pu, Y.; Apel, D.B.; Liu, V.; Mitri, H. Machine Learning Methods for Rockburst Prediction-State-of-the-Art Review. *Int. J. Min. Sci. Technol.* **2019**, *29*, 565–570.

[18] Mark, C. Coal Bursts in the Deep Longwall Mines of the United States. *Int. J. Coal Sci. Technol.* **2016**, *3*, 1–9.

[19] Pu, Y.; Apel, D.B.; Wei, C. Applying Machine Learning Approaches to Evaluating Rockburst Liability: A Comparation of Generative and Discriminative Models. *Pure Appl. Geophys.* **2019**, *176*, 4503–4517.

[20] Zhang, J.; Jiang, F.; Yang, J.; Bai, W.; Zhang, L. Rockburst Mechanism in Soft Coal Seam within Deep Coal Mines. *Int. J. Min. Sci. Technol.* **2017**, *27*, 551–556.

[21] Zhou, Z.; Cai, X.; Li, X.; Cao, W.; Du, X. Dynamic Response and Energy Evolution of Sandstone Under Coupled Static–Dynamic Compression: Insights from Experimental Study into Deep Rock Engineering Applications. *Rock Mech. Rock Eng.* **2020**, *53*, 1305–1331.

[22] Carter EJ, Pouch SM, Larson EL. The relationship between emergency department crowding and patient outcomes: a systematic review. *J Nurs Scholarsh.* 2014;46(2):106–15. doi: 10.1111/jnu.12055.

[23] Johnson KD, Winkelman C. The effect of emergency department crowding on patient outcomes: a literature review. *Adv Emerg Nurs J.* 2011;33(1):39–54. doi: 10.1097/TME.0b013e318207e86a.

[24] Pines JM, Iyer S, Disbot M, Hollander JE, Shofer FS, Datner EM. The effect of emergency department crowding on patient satisfaction for admitted patients. *Acad Emerg Med.* 2008;15(9):825–31. doi: 10.1111/j.1553-2712.2008.00200.x

[25] Sun BC, Hsia RY, Weiss RE, Zingmond D, Liang L-J, Han W, et al. Effect of emergency department crowding on outcomes of admitted patients. *Ann Emerg Med.* 2013;61(6):605–11. e6. doi: 10.1016/j.annemergmed.2012.10.026.

[26] Chiu I-M, Lin Y-R, Syue Y-J, Kung C-T, Wu K-H, Li C-J. The influence of crowding on clinical practice in the emergency department. *Am J Emerg Med.* 2018;36(1):56–60. doi: 10.1016/j.ajem.2017.07.011.

[27] Farrohknia N, Castren M, Ehrenberg A, Lind L, Oredsson S, Jonsson H, et al. Emergency department triage scales and their components: a systematic review of the scientific evidence. *Scand J Trauma Resuscitation Emerg Med.* 2011;19:42. doi: 10.1186/1757-7241-19-42.

[28] Christ M, Grossmann F, Winter D, Bingisser R, Platz E. Modern triage in the emergency department. *Deutsches Arzteblatt Int.* 2010;107(50):892–8.

[29] McHugh M, Tanabe P, McClelland M, Khare RK. More Patients Are Triaged Using the Emergency Severity Index Than Any Other Triage Acuity System in the United States. *Acad Emerg Med.* 2012;19(1):106–9. doi: 10.1111/j.1553-2712.2011.01240.x.

[30] Torabi M, Moeinaddini S, Mirafzal A, Rastegari A, Sadeghkhani N. Shock index, modified shock index, and age shock index for prediction of mortality in Emergency Severity Index level *Am J Emerg Med.* 2016;34(11):2079–83. doi: 10.1016/j.ajem.2016.07.017.

[31] Levin S, Toerper M, Hamrock E, Hinson JS, Barnes S, Gardner H, et al. Machine-Learning-Based Electronic Triage More Accurately Differentiates Patients With Respect to Clinical Outcomes Compared With the Emergency Severity Index. *Ann Emerg Med.* 2018;71(5):565–74. doi: 10.1016/j.annemergmed.2017.08.005.

[32] Torabi M, Mirafzal A, Rastegari A, Sadeghkhani N. Association of triage time Shock Index, Modified Shock Index, and Age Shock Index with mortality in Emergency Severity Index level 2 patients. *Am J Emerg Med.* 2016;34(1):63–8. doi: 10.1016/j.ajem.2015.09.014.

[33] Arya R, Wei G, McCoy JV, Crane J, Ohman-Strickland P, Eisenstein RM. Decreasing Length of Stay in the Emergency Department With a Split Emergency Severity Index 3 Patient Flow Model. *Acad Emerg Med.* 2013;20(11):1171–9. doi: 10.1111/acem.12249.

[34] Rajkomar A, Oren E, Chen K, Dai AM, Hajaj N, Hardt M, et al. Scalable and accurate deep learning with electronic health records. *npj Digital Med.* 2018;1(1):18. doi: 10.1038/s41746-018-0029-1.

[35] Stewart J**,** Sprivulis P**,** Dwivedi G. Artificial intelligence and machine learning in emergency medicine. Emerg Med Australas. 0(0).

[36] Coslovsky M, Takala J, Exadaktylos AK, Martinolli L, Merz TM. A clinical prediction model to identify patients at high risk of death in the emergency department. *Intensive Care Med.* 2015;41(6):1029–36. doi: 10.1007/s00134-015-3737-x.

[37] Pearl A, Bar-Or R, Bar-Or D. An artificial neural network derived trauma outcome prediction score as an aid to triage for non-clinicians. *Stud Health Technol Inform.* 2008;136:253.

[38] Dugas AF, Kirsch TD, Toerper M, Korley F, Yenokyan G, France D, et al. An Electronic Emergency Triage System to Improve Patient Distribution by Critical Outcomes. *J Emerg Med.* 2016;50(6):910–8. doi: 10.1016/j.jemermed.2016.02.026.

[39] Teubner DJ, Considine J, Hakendorf P, Kim S, Bersten AD. Model to predict inpatient mortality from information gathered at presentation to an emergency department: The Triage Information Mortality Model (TIMM) *Emerg Med Australas.* 2015;27(4):300–6. doi: 10.1111/1742-6723.12425.

[40] Schuetz P, Hausfater P, Amin D, Haubitz S, Fassler L, Grolimund E, et al. Optimizing triage and hospitalization in adult general medical emergency patients: the triage project. *BMC Emerg Med.* 2013;13:12. doi: 10.1186/1471-227X-13-12.

updated_final_document-converted-4.docx

ORIGINALITY REPORT

8%	7%	4%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	2%
2	www.ijraset.com Internet Source	1%
3	doi.org Internet Source	<1%
4	www.researchgate.net Internet Source	<1%
5	www.irjmets.com Internet Source	<1%
6	github.com Internet Source	<1%
7	Saeed Shurrab, Abdulkarem Almshnanah, Rehab Duwairi. "Tool Wear Prediction in Computer Numerical Control Milling Operations via Machine Learning", 2021 12th International Conference on Information and Communication Systems (ICICS), 2021 Publication	<1%

8	Melchizedek Alipio, Miroslav Bures. "Intelligent Network Maintenance Modeling for Fixed Broadband Networks in Sustainable Smart Homes", IEEE Internet of Things Journal, 2023 Publication	<1 %
9	mdpi-res.com Internet Source	<1 %
10	centaur.reading.ac.uk Internet Source	<1 %
11	sro.sussex.ac.uk Internet Source	<1 %
12	www.scribd.com Internet Source	<1 %
13	pdfs.semanticscholar.org Internet Source	<1 %
14	www.ijser.org Internet Source	<1 %
15	medinform.jmir.org Internet Source	<1 %
16	Jonnalagadda Kensarin, Arul Xavier V M, Uriti Jaswanth Venkata Sai, Sanga Sai Srujan, Kommana Viswa Prakash. "Prediction of Cardiovascular Disease Risk using Machine Learning Models", 2023 9th International	<1 %

	Conference on Advanced Computing and Communication Systems (ICACCS), 2023 Publication	
17	www.hindawi.com Internet Source	<1 %
18	Federica Origo, Manuela Samek Lodovici. "Chapter 6 Temporary Help Workers in Italy. Where Do They Come From and Where Do They Go?", Springer Science and Business Media LLC, 2012 Publication	<1 %
19	Yuanyuan Pu, Derek B. Apel, Chong Wei. "Applying Machine Learning Approaches to Evaluating Rockburst Liability: A Comparation of Generative and Discriminative Models", Pure and Applied Geophysics, 2019 Publication	<1 %
20	iopscience.iop.org Internet Source	<1 %
21	bmjopen.bmj.com Internet Source	<1 %
22	dokumen.pub Internet Source	<1 %
23	Taher M. Ghazal, Amer Ibrahim, Ali Sheraz Akram, Zahid Hussain Qaisar, Sundus Munir, Shanza Islam. "Heart Disease Prediction Using Machine Learning", 2023 International	<1 %

	Conference on Business Analytics for Technology and Security (ICBATS), 2023 Publication	
24	essay.utwente.nl Internet Source	<1 %
25	unbscholar.lib.unb.ca Internet Source	<1 %
26	Christopher Meiring, Abhishek Dixit, Steve Harris, Niall S. MacCallum et al. "Optimal intensive care outcome prediction over time using machine learning", PLOS ONE, 2018 Publication	<1 %
27	Hannah, Marsha J.. "", Techniques and Applications of Image Understanding, 1981. Publication	<1 %
28	Mostafa Rezapour, Muhammad Khalid Khan Niazi, Metin Nafi Gurcan. "Machine Learning-based Analytics of the Impact of the Covid-19 Pandemic on Alcohol Consumption Habit Changes Among United States Healthcare Workers", Research Square Platform LLC, 2023 Publication	<1 %
29	Sakari Tuominen, Roope Näsi, Eija Honkavaara, Andras Balazs et al. "Assessment of Classifiers and Remote Sensing Features of Hyperspectral Imagery and Stereo-	<1 %

	Photogrammetric Point Clouds for Recognition of Tree Species in a Forest Area of High Species Diversity", Remote Sensing, 2018 Publication	
30	Saskia Locke, Anthony Bashall, Sarah Al-Adely, John Moore, Anthony Wilson, Gareth B. Kitchen. "Natural language processing in medicine: A review", Trends in Anaesthesia and Critical Care, 2021 Publication	<1 %
31	dspace.cvut.cz Internet Source	<1 %
32	library.samdu.uz Internet Source	<1 %
33	stax.strath.ac.uk Internet Source	<1 %
34	thesis.unipd.it Internet Source	<1 %
35	www.ijert.org Internet Source	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		