Analysis on the survival of the Titanic dataset

Adilah Afroze Khan

*Abstract*—This paper explores various machine learning algorithms to predict the Titanic passenger survival rate based on demographic and travel-related features. It compares the effectiveness of kNN, SVM, Decision Trees, Random Forest, and Naïve Bayes in handling challenges like class imbalance and missing data. Results indicate that SVM outperforms others, achieving the highest accuracy of 80.99% and low prediction errors.

Keywords—titanic, survival prediction, machine learning, kNN, SVM, Decision Tree, Random Forest, Naïve Bayes.

# Introduction

The Royal Mail Ship (RMS) Titanic, the largest liner of its era, tragically sank during its maiden voyage from Southampton to New York City, resulting in the loss of over 1,500 lives among its 2,224 passengers and crew aboard[7].The disaster remains one of the most studied maritime tragedies, offering valuable insights into survival patterns based on passenger demographics. It presents a compelling data science challenge: predicting survival outcomes through binary classification based on demographic and travel features. Besides, understanding survival patterns in maritime disasters has important implications for safety protocols and emergency response planning.

Traditional statistical analysis requires extensive manual effort and domain expertise in both maritime safety and demography. Modern data science approaches offer powerful alternatives that can automatically identify survival patterns from passenger attributes, enabling more efficiency and scalability.

Titanic prediction is fundamentally a binary classification problem where machine learning algorithms predict survival outcomes based on the given demographic and travel-related features. This paper presents a comprehensive comparative study of various classification algorithms to determine which method yield optimal performance for this specific case.

The research domain has evolved significantly since the earlier statistical analysis of the disaster. Early studies relied on simple machine learning models, like logistic regression, Naïve Bayes, Decision tree, Random Forest [7,8] to predict the survival of passengers. Barhoom et al. [9] demonstrated improved accuracy using artificial neural network and studying the characteristics of passengers-cabin class, age, point of departure, and that relationship to the chance of survival of the disaster.

The problem addressed in this study holds particular challenges as it involves:

* Imbalanced dataset
* Numerous missing data points across critical features
* Complex interactions between socioeconomic factors and survival outcomes.

The main objectives of this research are to:

1. Assess how different approaches handle the inherent class imbalance and missing data.
2. Evaluate and compare five distinct classification algorithms (kNN, SVM, Decision Tree, Random Forest, and Naïve Bayes).
3. Identify which classifier performs the best given the dataset’s unique characteristics.

# Dataset description

## Data Collection

The dataset used in this study is primarily collected from the Kaggle competition platform, specifically from the ‘Titanic: Machine Learning from Disaster’ challenge [1]. It is a classic binary classification dataset used to predict passenger survival based on the demographic and ticket information of the passengers.

## Data Overview

The dataset includes a mixture of numerical, categorical and textual features. It contains 891 rows and 12 columns, where each row corresponds to a passenger on the Titanic. A brief description of all the features are given as follows:

1. PassengerId: A unique numerical identifier assigned to each passenger.
2. Survived: The target variable indicating survival status (1 = survived, 0 = did not survive).
3. Pclass: An ordinal variable indicating socio-economic class (1 = Upper, 2 = Middle, 3 = Lower).
4. Name: A textual feature indicating Passenger’s name.
5. Sex: A categorical variable indicating gender (male/female).
6. Age: A continuous variable indicating the age of the passenger.
7. SibSp: A discrete variable indicating number of siblings or spouses aboard.
8. Parch: A discrete variable indicating number of parents or children aboard.
9. Ticket: A textual feature indicating the ticket number.
10. Fare: A continuous variable indicating fare paid for the ticket.
11. Cabin: A categorical feature indicating the cabin number.
12. Embarked: A categorical variable representing the port of embarkation (C = Cherbourg, Q = Queenstown, S = Southampton).

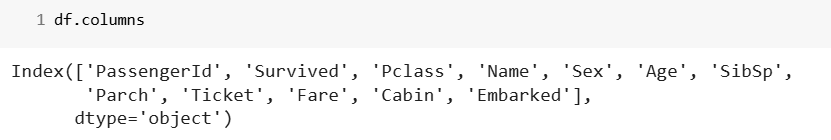


Fig. 1. Dataset Columns

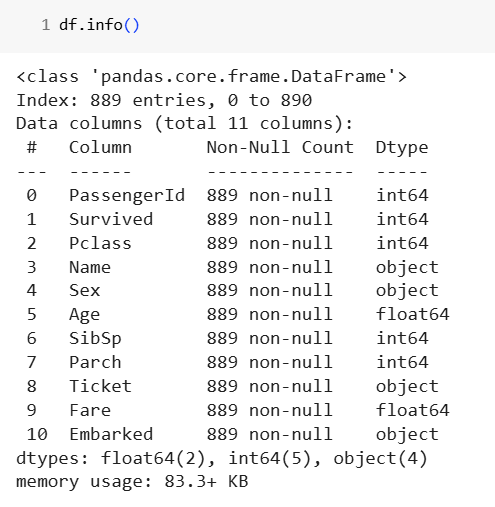


Fig. 2. Dataset Summary

## Data Distribution

The target feature ‘Survived’ has an imbalanced distribution, with a survival rate of only 38.3% (Fig. 3) suggesting a higher proportion of passengers did not survive the incident.

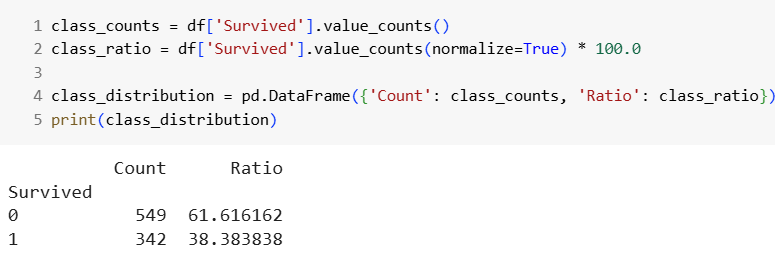


Fig. 3. Dataset Distribution

## Missing Values

The dataset contains missing values in several features, with ‘Age’ having 177 missing values, ‘Cabin’ showing a significant amount of missing data, with 687 missing entries, and ‘Embarked’ having only 2 missing entries (Fig. 4).

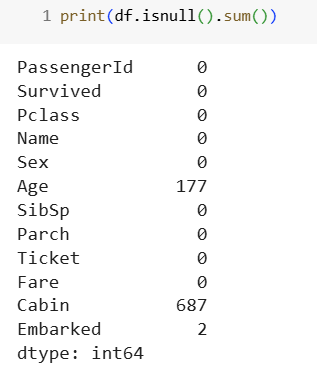


Fig. 4. Missing entries in each column

# methodology

## Software Tools Utilized

For this analysis, Python was the primary language used, along with several supporting libraries. The Python libraries utilized include:

1. Pandas: Used for data manipulation and analysis, including reading the dataset, handling missing data and extracting insights.
2. NumPy: Used for numerical operations and efficient array handling.
3. Matplotlib.pyplot and Seaborn: Used for data visualization, including plotting count plot, histogram, box plots, and other graphical representations.

In addition to Python, WEKA, a data mining and machine learning software developed in Java, was also used for exploratory analysis and comparison of model performance.

## Data Loading and Initial Inspection

* The dataset, named *titanic.csv,* was stored in Google Drive and accessed via Google Colab. It was loaded into the Python environment using the *pandas.read\_csv()* function.

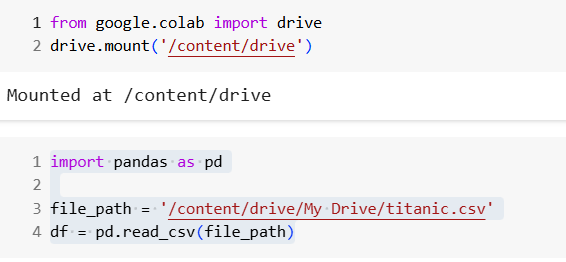


Fig. 5. Data Loading

* A description of the initial dataset inspection is provided using *df.head(),* which displays the first five rows of the dataset to give a quick overview of the data structure:

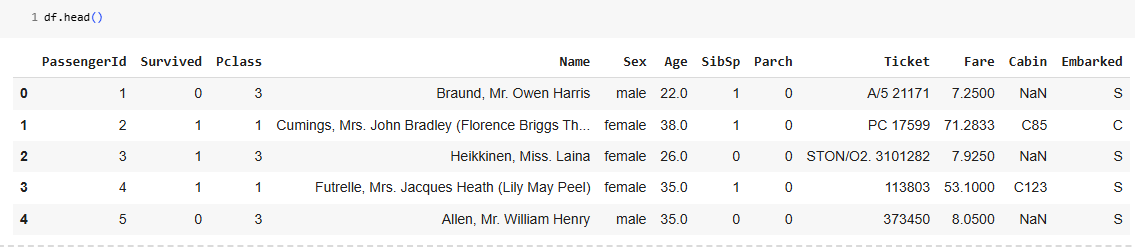


Fig. 6. First five rows of the dataset

Identify applicable funding agency here. If none, delete this text box.

* The column names of the dataset were retrieved using the *df.columns* method(see Fig. 1).

## Descriptive Statistics and Data Summary

The dataset was analyzed using the following methods to understand its structure and statistical properties:

* df.shape: The dataset comprises of 891 rows and 12 columns.
* df.info(): Provides a summary of the dataset, including the number of entries, column names, data types, and memory usage(see Fig. 2).
* df.describe(): Provides summary statistics for numerical features (e.g., mean age: 29.7 years, survival rate: 38.4%).

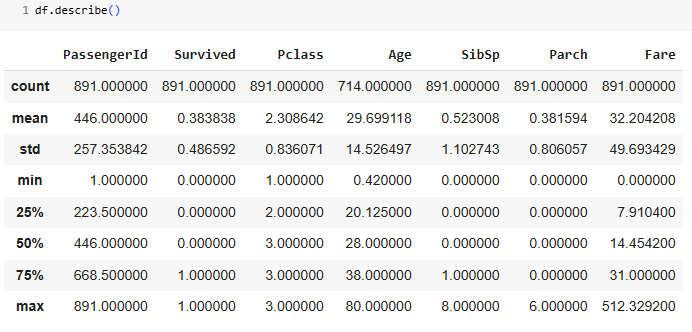


Fig. 7. Dataset summary

## Data Cleaning

Missing values were identified using *df.isnull().sum()* (see Fig. 4), revealing significant gaps in the ‘Age’ and ‘Cabin’ columns. These missing values were handled using a combination of imputation and column removal.

For the ‘Age’ feature, which had 177 missing values(see Section II), imputation was performed based on the average age within each passenger class, due to the observed correlation between Age and Pclass. This approach preserved valuable data that would have been lost if dropped. The ‘Cabin’ column, with 687 missing values, was dropped due to its limited usefulness. Similarly, the ‘Embarked’ column was dropped as it had only 2 missing entries, having insignificant impact on the analysis.

# modeling

Before training the models in WEKA, two essential filters were applied to the cleaned Titanic dataset:

* *Standardization*: It was used to scale all numeric features to have a mean of 0 and standard deviation of 1. This preprocessing was particularly important for distance-based classifiers like KNN and SVM.
* *Nominal to Binary Conversion*: The NominalToBinary filter was applied to convert categorical features into numeric binary attributes.

## Modelling Techniques Utilised

Five classification algorithms were implemented in

WEKA directly to predict passenger survival:

### k-Nearest Neighbour(kNN): Implemented via WEKA’s IBk classifer, classification is achieved by identifying the nearest neighbors to a query example and using those neighbors to determine the class of the query[2].

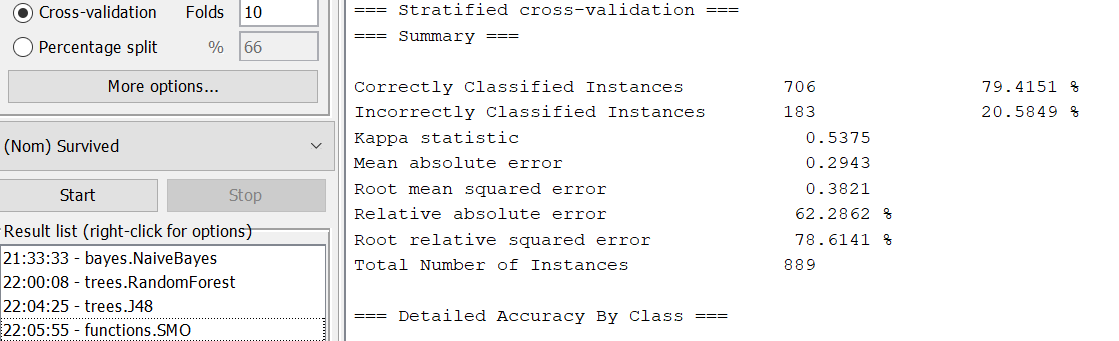


Fig. 8. Accuracy of kNN model

### Support Vector Machine(SVM): Implemented via WEKA’s SMO classifier, SVM transforms input data into a higher-dimensional feature space using kernel functions and constructs a hyperplane that acts as the best separator plan to separate the classes[3].

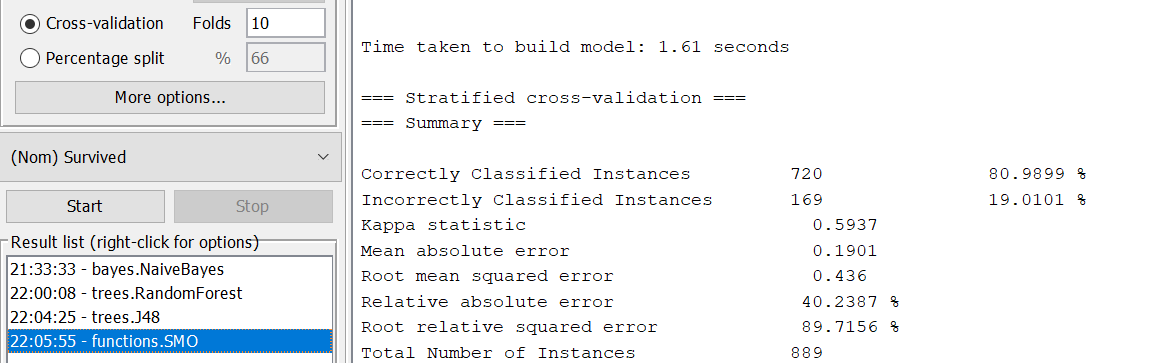


Fig. 9. Accuracy of SVM model

### Decision Tree(DT): The J48 classifier in WEKA implements decision tree which is a hierarchical classifier recursively splitting data based on feature values to make decisions, aiming for maximum information gain at each note[4].

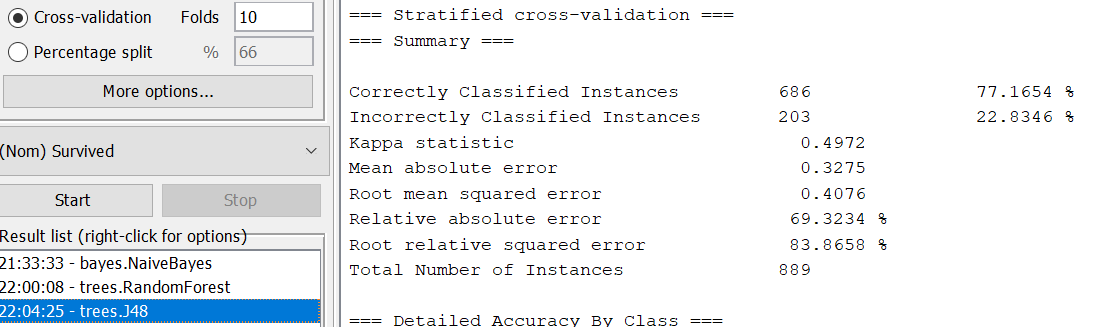


Fig. 10. Accuracy of DT model

### Random Forest(RF): Random Forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest[5].

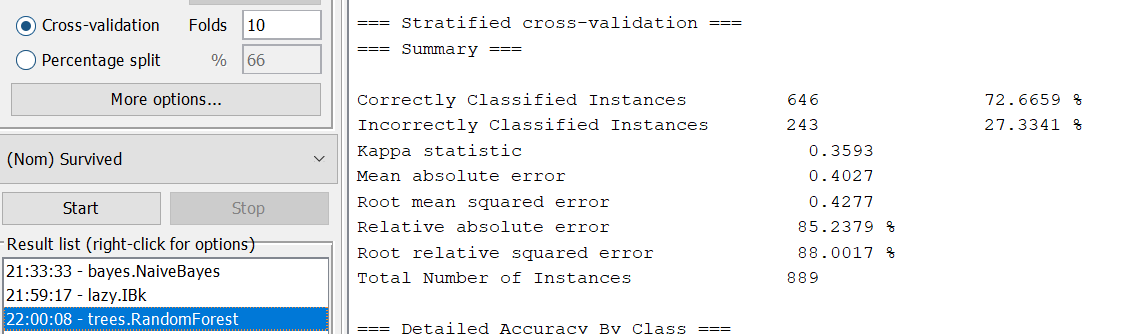


Fig. 11. Accuracy of RF model

### Naïve Bayes(NB): As a mathematical classification approach, this classifier involves a series of probabilistic computations for the purpose of finding the best-fitted classification[6].

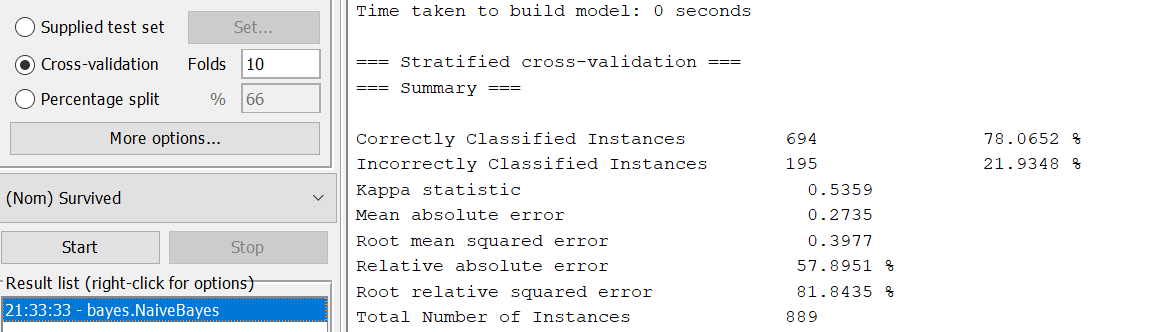


Fig. 12. Accuracy of NB model

## Evaluation Methodology

The performance of the classifiers was assessed using several evaluation metrics. The key metrics included:

### Accuracy: The proportion of correctly classified instances.

### Kappa Statistic: Measures the agreement between predictions and actual values.

### Mean Absolute Error(MAE): Measures the average error between predictions and actual values.

### Root Mean Squared Error(RMSE): Measures the standard deviation of prediction errors.

### Relative Absolute Error(RAE): Compares the absolute error of model’s predictions relative to a baseline model.

### Root Relative Squared Error(RRSE):Quantifies the error relative to a model predicting the mean.

### Precision, Recall, and F-measure: Measures for each class(survived and not-survived), providing insight into classification performance.

### Confusion Matrix:Provides a detailed view of classification performance measuring True Positive, True Negative, False Positive, and False Negative.

# Results

The performance of the classification models was evaluated on the cleaned Titanic dataset. The evaluation was conducted using various statistical metrics, and the results are summarized in Table I and Table II.

1. Overall performance metrics

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Classifier | Accuracy | Kappa Statistic | MAE | RMSE | RAE | RRSE |
| kNN  (k = 10) | 79.42% | 0.5375 | 0.2943 | 0.3821 | 62.29% | 78.61% |
| SVM | 80.99% | 0.5937 | 0.1901 | 0.436 | 40.24% | 89.72% |
| DT | 77.17% | 0.4972 | 0.3275 | 0.4076 | 69.32% | 83.87% |
| RF | 72.67% | 0.3593 | 0.4027 | 0.4277 | 85.24% | 88.00% |
| NB | 78.07% | 0.5359 | 0.2735 | 0.3977 | 57.90% | 81.84% |

Table I shows the overall classification performance of models based on accuracy, kappa statistic, MAE, RMSE, RAE, and RRSE.

Among all models, SVM achieved the highest accuracy (80.99%), followed by kNN (79.42%) and Naïve Bayes (78.07%). SVM also recorded the lowest MAE (0.1901) and RAE (40.24%), indicating better generalization and lower prediction error.

1. Class-wise evaluation metrics (Weighted avg.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | Precision | Recall | F-measure | FPR | ROC Area |
| kNN  (k = 10) | 0.800 | 0.794 | 0.785 | 0.288 | 0.847 |
| SVM | 0.809 | 0.810 | 0.809 | 0.222 | 0.794 |
| DT | 0.770 | 0.772 | 0.766 | 0.294 | 0.758 |
| RF | 0.742 | 0.727 | 0.699 | 0.402 | 0.832 |
| NB | 0.781 | 0.781 | 0.781 | 0.244 | 0.838 |

TABLE II compares the classifiers based on precision, recall, F-measure, False Positive Rate, and ROC area.

Naïve Bayes showed balanced performance across precision, recall, and F-measure (all at 0.781), while Random Forest achieved the highest ROC area (0.832) indicating better separability. However, Random Forest also had the highest FPR (0.402), suggesting more false positives.

# Discussion

From the experimental results summarized in TABLE I and TABLE II, it is evident that SVM emerged as the best-performing model overall. It achieved the highest accuracy of 80.99%, the best kappa statistic of 0.5937, and the lowest mean absolute error (MAE), and relative absolute error (RAE), indicating strong consistency between the predicted and actual classifications. The result reflects SVM’s robustness in handling the dataset’s complexity and its ability to generalize well.

In terms of precision, recall, and F-measure, SVM again performed strongly, suggesting a balanced ability to correctly classify both survivors and non-survivors.

Interestingly, NB and kNN also showed relatively strong performance, with accuracies of 78.07% and 79.42% respectively. NB demonstrated consistency across all metrics, particularly in precision, recall, and F-measure, making it a strong candidate where model simplicity is prioritized.

Despite being a strong performer typically, the ensemble learning model RF underperformed in this case with the lowest accuracy (72.67%) and the highest error rates.

The evaluation shows that even comparatively less precise models can still perform well in certain metrics (e.g., RF had a high ROC area of 0.832), emphasizing necessity of multi-metric evaluation method as accuracy alone would have overlooked these fine-grained performance characteristics.

# Conclusion

This study evaluated five machine learning models- k-Nearest Neighbor, Support Vector Machine, Decision Tree, Random Forest, and Naïve Bayes on the Titanic dataset to predict passenger survival. SVM demonstrated superior performance with the highest accuracy (80.99%) and lowest error rates, showcasing its robustness in handling classification tasks. While Naïve Bayes and kNN also performed well, ensemble learning model, Random Forest underperformed. This underperformance suggests the importance of model selection based on dataset characteristics. The findings highlight multi-metric evaluation is essential, as accuracy alone may overlook nuanced performance differences. Future work could explore feature engineering, and hyperparameter tuning to further enhance model efficiency.

##### References

1. Titanic: Machine Learning from Disaster, <https://www.kaggle.com/c/titanic>
2. Kataria, Aman & Singh, Mandeep. (2013). A Review of Data Classification Using K-Nearest Neighbour Algorithm.
3. Cortes, C., Vapnik, V. Support-vector networks. *Mach Learn* **20**, 273–297 (1995). https://doi.org/10.1007/BF00994018.
4. Song YY, Lu Y. Decision tree methods: applications for classification and prediction. *Shanghai Arch Psychiatry*. 2015;27(2):130-135. doi:10.11919/j.issn.1002-0829.215044.
5. Breiman, L. Random Forests. *Machine Learning* **45**, 5–32 (2001). https://doi.org/10.1023/A:1010933404324.
6. F. -J. Yang, "An Implementation of Naive Bayes Classifier," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018, pp. 301-306, doi: 10.1109/CSCI46756.2018.00065. keywords: {Probabilistic logic;Bayes methods;Tools;Python;Machine learning;Data mining;Engines;Naive Bayes Classifier, Probabilistic Classification, Bayesian Theory}.
7. Tabbakh, A., Rout, J.K., Rout, M. (2021). Analysis and Prediction of the Survival of Titanic Passengers Using Machine Learning. In: Tripathy, A.K., Sarkar, M., Sahoo, J.P., Li, KC., Chinara, S. (eds) Advances in Distributed Computing and Machine Learning. Lecture Notes in Networks and Systems, vol 127. Springer, Singapore. <https://doi.org/10.1007/978-981-15-4218-3_29>.
8. A. Singh, S. Saraswat and N. Faujdar, "Analyzing Titanic disaster using machine learning algorithms," 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India, 2017, pp. 406-411, doi: 10.1109/CCAA.2017.8229835. keywords: {Machine learning algorithms;Prediction algorithms;Algorithm design and analysis;Logistics;Decision trees;Predictive models;Classification algorithms;Titanic;Prediction;Classification;Data mining;R;Python;Logistic Regression;Random Forest;Decision Tree;Nave Bayes},
9. Barhoom, Alaa & Abu-Naser, Samy & Abu-Nasser, Bassem & Khalil, Ahmed & Musleh, Musleh. (2019). Predicting Titanic Survivors using Artificial Neural Network.