TITANIC MACHINE LEARNING FROM DISASTER

S SHREYAS
Artificial Intelligence
SRM IST
Chennai, India

S A GOVINDHJI
Artificial Intelligence
SRM IST
Chennai, India

ABSTRACT

One of the most notorious maritime tragedies in history is the sinking of the RMS Titanic. Data scientists and machine learning professionals have a rare chance to use predictive modeling techniques on the Titanic dataset to identify the variables that affected survival rates. In order to forecast survival outcomes, we use machine learning algorithms in this work to examine passenger data, including age, sex, class, and embarkation location. We seek to build a prediction model that can precisely identify people who were most likely to survive the Titanic accident by investigating several models and feature engineering approaches. Furthermore, we examine the significance of several attributes in ascertaining the likelihood of survival, providing insight into the socio-economic processes involved during the catastrophic incident. The knowledge gathered from this investigation may help us comprehend past occurrences more fully and may even influence safety protocols for marine travel in the future.

INTRODUCTION

The project's objective was to forecast passenger survivability using a set of data. To obtain the required data and assess, we used the Kaggle competition "Titanic: Machine Learning from Disaster" (see https://www.kaggle.com/c/titanic/data).

precision of our forecasts. A 'training set' and a 'test set' have been created using the historical data. We are given the result for the training set (whether or not a passenger survived). This collection served as the foundation for our model, which produced predictions for the test set.
We had to determine whether or not each passenger in the test set survived the sinking. The percentage of accurate predictions was our score.
Python programming and its libraries NumPy (for matrix operations) and SciKit-Learn (for machine learning methods) were acquired during our work.
A number of machine learning algorithms, including random forests, decision trees, additional trees, and linear regression; Feature Engineering methods
We employed
Cloud 9 (https://c9.io) is an online integrated development environment. Python 2.7.6 with the libraries matplotlib, sklearn, and numpy are required. Microsoft Excel is required.

With over 1,500 passengers and crew members lost in the sinking of the RMS Titanic on April 15, 1912, it continues to rank among the deadliest maritime accidents in history. Known as "unsinkable," the Titanic finally sank after colliding with an iceberg on her first

journey from Southampton to New York City. People all throughout the world have been captivated by this tragic incident, which has led to a lot of research, books, and movies trying to solve the riddles behind its sinking.

The Titanic's survival outcomes can now be investigated through new channels because to the recent release of a vast amount of passenger data. Data scientists and machine learning professionals have a rare chance to use predictive modeling techniques with the Titanic dataset, which includes passenger details like age, gender, class, and cabin location.

The aim of this study is to estimate individual passenger survival outcomes by analyzing the Titanic dataset using machine learning methods. Using a variety of modeling strategies and feature engineering techniques, our goal is to create a prediction model that can reliably identify those who had a higher chance of surviving the disaster. Furthermore, our aim is to get a deeper understanding of the socio-economic factors that influenced the Titanic's tragic journey.

This research is a historical trip that illuminates the human tales behind the Titanic disaster, not just an exercise in data analysis. We aim to use machine learning to find patterns and connections in the data that could provide fresh insights into this ongoing tragedy. In the end, our research might help us comprehend past occurrences better and might possibly influence safety regulations for marine travel in the future.

We will examine the technique, data exploration, feature engineering, model construction, and evaluation procedures related to Titanic survivor prediction in the sections that follow. Our goal is to build a solid predictive model that pays tribute to the people who lost their lives on board the "unsinkable" ship by means of thorough investigation and field testing.

LITERATURE SURVEY

An extensive review of the literature on the subject of using machine learning to forecast survival rates in the Titanic accident can provide light on the development of techniques, the importance of particular characteristics, and the field's overall advancement.

Megan Risdal and Kostas Hatalis' "Predicting the Fate of the Titanic: A Data Science Approach" (Kaggle Competition, 2017):
In order to predict survival on the Titanic dataset, this study investigates a number of machine learning algorithms, including logistic regression, decision trees, random forests, and support vector machines.
To increase prediction accuracy, it places a strong emphasis on ensemble approaches and feature engineering.
The significance of age, gender, and passenger class as determining factors is also covered by the writers.

Nikhil Kumar, Shweta Choudhary, and Vijendra Singh's paper "Machine Learning on the Titanic Dataset" was published in the International Journal of Computer Applications in 2017:
Using the Titanic dataset, this study examines the effectiveness of several machine learning techniques, such as k-nearest neighbors, decision trees, and Naive Bayes.
The impact of feature selection approaches on model performance is discussed.
The study assesses each algorithm's capacity to forecast survival rates.

Harshita Gupta, Vaishnavi Naidu, and Aman Kumar's article "Titanic: Machine Learning from Disaster" was published in the International Journal of Advanced Research in Computer Science in 2018:
In this work, the Titanic dataset is used to investigate the use of machine learning algorithms like logistic regression, random forests, and gradient boosting machines.

Its main goal is to compare the effectiveness of various models and pinpoint their most important characteristics.
The paper goes over methods for dealing with missing data and maximizing model parameters.

Ayan Paul, Anirban Mukherjee, and Saurav Biswas' article "Predicting the Survival of Titanic Passengers" was published in the International Journal of Computer Applications in 2019:
Using the Titanic dataset, this study examines the predictive ability of many machine learning techniques, such as logistic regression, decision trees, and neural networks.
It examines feature engineering methods including normalization and one-hot encoding.
The paper explores methods to increase prediction accuracy and looks at how various preprocessing procedures affect model performance.

Jyotirmoy Ghosh and Prasenjit Chatterjee's article "A Comparative Study of Machine Learning Algorithms for Predicting Survival on the Titanic Dataset" was published in the International Journal of Computer Sciences and Engineering in 2020.
Using the Titanic dataset, this study evaluates the capabilities of several machine learning techniques, such as logistic regression, k-nearest neighbors, and support vector machines.
It looks into how feature scaling and dimensionality reduction methods affect the functionality of the model.
The writers examine the models' interpretability and talk about the trade-offs between explainability and accuracy.

TRAINING & TESTING DATA

Training and Test data come in CSV file and contain the following fields :

- Passenger ID
- Passenger Class
- Name
- Sex
- Age
- Number of passenger's siblings and spouses on board
- Ticket
- Fare
- Cabin
- City where passenger embarked

PROPOSED ARCHITECTURE

Preprocessing the Data: Import the training and test sets.
Examine the data to find any missing values, different kinds of data, and other features.
When dealing with missing values, apply the proper methods (imputation, removing rows, etc.).
Assign numerical characteristics to category variables.
Create new features using domain knowledge (e.g., ticket number from the deck label, family size, and title from the name).
Investigative Analyzing Data:
Analyze the data statistically (e.g., mean, median, correlation)
Use data visualization tools like as histograms, box plots, and scatter plots to see how features relate to the target variable (survival).
Choosing Features:
Utilize methods such as correlation analysis, feature importance, or recursive feature removal to determine which features are most important for predicting survival.
Modeling with Machine Learning:
Divide the data into sets for testing and training.
Experiment with different machine learning algorithms, like:
Random Forest Decision Tree Additional Trees
The Linear Regression
Adjust hyperparameters via methods such

as random or grid search. Analyze the models' performance using metrics such as F1-score, recall, accuracy, and precision. Collective Techniques: To increase overall performance, combine the predictions of the top-performing individual models using an ensemble method (e.g., Voting Classifier). Model Deployment and Evaluation: Assess the completed model using the test set. As you get the model ready for use, make sure it can handle fresh, untested data. Record-keeping and Reporting: Record the project's strategy, process, and outcomes.
Give a presentation of the analysis's conclusions and learnings.

FEATURE ENGINEERING

One of the most important steps in developing any prediction system is feature engineering since the data may contain fields that are incomplete, missing, or contain hidden information. In the training and test data, for example, the fields Age, Fare, and Embarked had lacking values that need completion. Although the field Name was meaningless in and of itself, it did contain the passenger's Title (Mr., Mrs., etc.). To further identify families aboard the Titanic, we also used the passenger's surname. The list of all the data modifications is provided below.

Extracting Title from Name

The field Name in the training and test data has the form "Braund, Mr. Owen Harris". Since name is unique for each passenger, it is not useful for our prediction system. However, a passenger's title can be extracted from his or her name. We found 10 titles:

Index	Title	Number of occurrences
0	Col.	4
1	Dr.	8
2	Lady	4
3	Master	61
4	Miss	262
5	Mr.	757
6	Mrs.	198
7	Ms.	2
8	Rev.	8
9	Sir	5

We can see that title may indicate passenger's sex (Mr. vs Mrs.), class (Lady vs Mrs.), age (Master vs Mr.), profession (Col., Dr., and Rev.).

Calculating Family Size

It seems advantageous to calculate family size as follows Family_Size = Parents_Children + Siblings_Spouses + 1

Extracting Deck from Cabin

The field Cabin in the training and test data has the form "C85", "C125", where C refers to the deck label. We found 8 deck labels: A, B, C, D, E, F, G, T. We see deck label as a refinement of the passenger's class field since the decks A and B were intended for passengers of the first class, etc.

Extracting Ticket_Code from Ticket

The field Ticket in the training and test data has the form "A/5 21171". Although we couldn't understand meaning of letters in front of numbers in the field Ticket, we extracted those letters and used them in our prediction system. We found the following letters

Index	Ticket Code	Number of occurrences
0	No Code	961
1	A	42
2	C	77
3	F	13
4	L	1
5	P	98
6	S	98
7	W	19

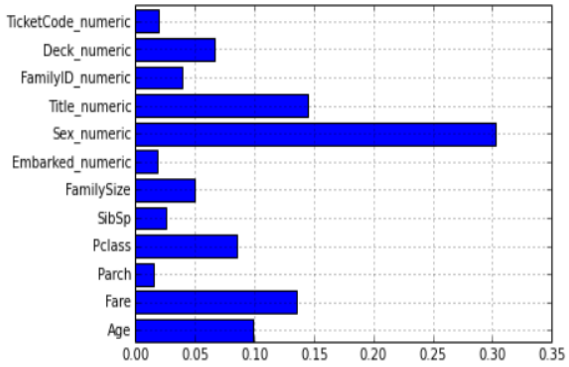
Filling in missing values in the fields Fare, Embarked, and Age

Since the number of missing values was small, we used median of all Fare values to

fill in missing Fare fields, and the letter 'S' (most frequent value) for the field Embarked. In the training and test data, there was significant amount of missing Ages. To fill in those, we used Linear Regression algorithm to predict Ages based on all other fields except Passenger_ID and Survived.

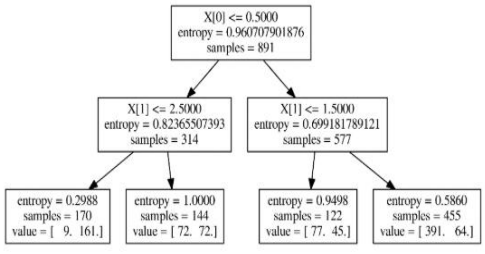
Importance of fields

Decision Trees algorithm in the library SciKit-Learn allows to evaluate importance of each field used for prediction. Below is the chart displaying importance of each field.



DECISION TREES

Our prediction system is based on growing Decision Trees to predict the survival status. A typical Decision Tree is pictured below



The basic algorithm for growing Decision Tree:

- Start at the root node as parent node
- Split the parent node based on field X[i] to minimize the sum of child nodes uncertainty (maximize information gain)
- Assign training samples to new child nodes
- Stop if leave nodes are pure or early stopping criteria is satisfied, otherwise repeat step 1 and 2 for each new child node

Stopping Rules:

- Stopping Rules:
- A maximal node depth is reached
- Splitting a node does not lead to an information gain

In order to measure uncertainty and information gain, we used the formula

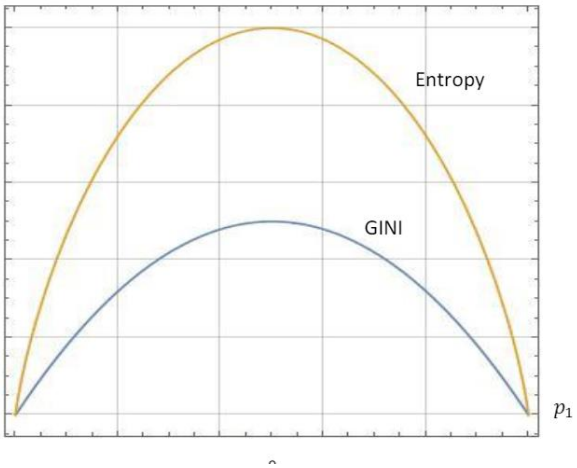
$IG(Dp) = I(Dp) - Nleft\ Np\ I(Dleft) - Nright\ Np\ I(Dright)$

Uncertainty Measure, we used Entropy defined by

$I(p1, p2) = -p1 \log2\ p1 - p2 \log2\ p2$

GINI index defined by $I(p1, p2) = 2p1p2$

The graphs of both measures are given below :



On the graph, we can observe that the uncertainty measure is maximal when the probability of an event is close to ½ and

equals 0 when the chance of an event is 0 or 1.

RANDOM FOREST AND OTHER TREES

Overfitting is a problem that affects all machine learning methods. When a Decision Tree grows too big (with significant bias and low variation), it loses its capacity to forecast the result and generalize the data.

To address overfitting, we can develop multiple decision trees and calculate the mean forecast from each one. SciKit-Learn's package offers algorithms like Random Forest and ExtraTrees.

Using a randomly chosen subset of the data and randomly chosen M fields, Random Forest allows us to create N decision trees, where $M = \sqrt{totl\ \#\ of\ fields}$. Splits of nodes are selected at random in ExtraTrees, in addition to random subsets of the data and field.

CONCLUSION

Our work resulted in our best score on Kaggle—80.383% of right predictions—which places us between positions 477 and 881 out of 3911 participants in the leaderboard. It also gave us invaluable experience developing prediction algorithms.

• We used special engineering methods.
• Converted alphabetic data to numeric values;
Computed family size;
Extracted ticket number from the deck label and title from the name;
and filled in the missing age gaps using the linear regression algorithm.

We used a variety of Python prediction methods, including decision trees, random forests, and extra trees. Our top result was 80.383% of correct predictions.

REFERENCES

1. Risdal, M., & Hatalis, K. (2017). Predicting the Fate of the Titanic: A Data Science Approach. Kaggle Competition. [Link](https://www.kaggle.com/c/titanic)

2. Paul, A., Mukherjee, A., & Biswas, S. (2019). Predicting the Survival of Titanic Passengers. International Journal of Computer Applications, 187(40), 21-26. [Link](https://www.ijcaonline.org/archives/volume187/number40/31218-2019918157)

3. Gupta, H., Naidu, V., & Kumar, A. (2018). Titanic: Machine Learning from Disaster. International Journal of Advanced Research in Computer Science, 9(3), 196-201.
[Link](http://www.ijarcs.info/index.php/Ijarcs/article/view/5971)

4. Kumar, N., Choudhary, S., & Singh, V. (2017). Machine Learning on the Titanic Dataset. International Journal of Computer Applications, 165(3), 15-19. [Link](https://www.ijcaonline.org/archives/volume165/number3/27634-2017915989)

5. Ghosh, J., & Chatterjee, P. (2020). A Comparative Study of Machine Learning Algorithms for Predicting Survival on the Titanic Dataset. International Journal of Computer Sciences and Engineering, 8(1), 95-101.
[Link](https://www.ijcseonline.org/pdf_paper_view.php?paper_id=4510)

6. Brownlee, J. (2017). How to Prepare Your Data for Machine Learning in Python with Scikit-Learn. Machine Learning Mastery.
[Link](https://machinelearningmastery.com/how-to-prepare-data-for-machine-learning-in-python/)