ECS 171 Project Report: Titanic Survival

Team 5: Sivani Voruganti, Radhika Kulkarni, Mitchell Geringer,
Tabib Chowdhury, Zhixing Liu, Jean Pagaduan, Aly Kapasi, Haohan Ji,
Shiwei Hei, Seok Ki, Chenrui Zhang, Vicky Yin

July 2021

Project Code - Github Repository Link
Project Code - Zip File Google Drive Link

1 Introduction and Background

The Titanic Survival Project is based on the tragic historical event in which the RMS Titanic, a British passenger liner, sank in the Northern Atlantic Ocean after striking an iceberg while on its voyage from Southampton to New York City on April 14, 1912. There were an estimated 2,224 passengers on board and an estimated 68% of those who boarded the supposed "Unsinkable Ship" perished either in the sinking of the vessel or in the subzero waters of the North Atlantic Ocean (Star).

On April 14th, 1912, the crew members started to receive warnings regarding the emergence of an iceberg and floating ice. They unfortunately did not realize the importance of these messages and did not respond to the situation besides shifting the fairway a little bit further south without slowing down. Ultimately, the series of consequences that followed led to the tragic sinking of the Titanic ("Titanic in Nova Scotia").

The Titanic Survival project is a classic artificial intelligence problem which serves as an insightful window into exploring data science and machine learning. The primary goal of our research is to predict whether passengers survived the Titanic shipwreck based on demographic data such as their age, gender, socio-economic class, and specifically understand which attributes impacted their survival most. For instance, it was reported that first class women had the highest survival rate while the third class men had the lowest survival rate regardless of crew members, and we shall find out if this claim, along with other observations, is true (Zanchi). We also aim to apply and test the efficacy of several different Machine Learning models – specifically Logistic Regression, Artificial Neural Network, Random Forest, Support Vector Machine (SVM) with linear and rbf kernels, and K-Nearest Neighbor – in predicting whether passengers survived the shipwreck based on their demographic passenger data.

Beyond this study, our comparative research work will be useful in determining the most accurate and effective Machine Learning strategy to utilize in present and future applications. For instance, this research could be relevant in applications such as simulating and reporting the safety of modern-day ships and cruise ships before they are open to passengers. This research can also be extended to cases such as the recent COVID-19 outbreak aboard the Diamond Princess cruise ship, to perhaps simulate the likeliness of the contracting the virus amongst passengers and take necessary precautions.

The remaining part of the paper is structured as follows: review of some significant literature offering insights into the data, a dataset description and analysis of the Titanic Kaggle Dataset, our proposed methodology including data processing and models we created to predict survival, experimental results from model predictions, and lastly conclusion and discussion.

2 Literature Review

The Titanic Survival Machine Learning problem is relatively popular and has been studied extensively in recent times. In particular, most researchers conducted comparative studies to examine the performances of several different Machine Learning techniques and algorithms in predicting Titanic survivors. For instance, Ekinci et. al.(2018) applied fourteen prevalent Machine Learning techniques to this dataset, including Logistic Regression, k-Nearest Neighbors, Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, Voting, and Artificial Neural Networks, among other methods. There is some overlap in the utilized techniques across comparative studies on the Titanic dataset. For example, a study by Kakde and Agrawal (2018) incorporates the Logistic Regression, Decision Tree, Random Forest, and SVM techniques, while also placing special emphasis on a preliminary exploratory data analysis process to uncover new features of the data. Also, Balakumar et. al. (2019), Singh et. al. (2020), and Farag and Hassan (2018) similarly incorporate the Logistic Regression, Decision Tree, k-Nearest Neighbors, Random Forest, Naive Bayes, and SVM techniques. Additionally, most studies used Grid Search method for hyperparameter tuning (when not using methods that required more computing power and time than we could provide). These Machine Learning techniques appear to be most commonly employed in the context of the Titanic Survival problem, with the Random Forest algorithm usually performing best - an observation to be cognizant of while we embark on this problem.

3 Dataset Description & Analysis

After acquiring the dataset supplied by Kaggle for the Titanic Survival problem, we restrict and restructure the training dataset for reasonable analysis. We perform minor data cleaning as well as data imputation to deal with missing values, and drop variables which are not needed for predicting survival.

Our given training dataset, renamed "Titanic", had three variables with NA values, also known as missing values. Around 20% of the Age variable, 77% of the Cabin Number variable, and 0.2% of the Embarked variable had missing values. Since the Embarked variable had very few NA values, we removed the entries associated with these missing embarkments, as the effect of not having these values will likely be small on our predictions. We also removed the variable Cabin Number because most of the values were missing, making it unsuitable for prediction, since the possible influence of Cabin Number on survival is likely already encompassed in features like Class and Fare which both also dictate where passengers are seated on the ship.

From there, to be able to use the Age variable in our model we performed data imputation on Age using KNN imputator from *sklearn* library in Python to replace the missing values. We used the 5 neighbors of each NA Age value to create an average age for each missing age value. The weights are uniformly averaged by measure of distances from each neighbor. We thought this was a reasonable approach instead of eliminating the Age variable completely because we believe that age will play a rather important role in determining survival. We chose to use 5 nearest neighbors in our imputation because most people on the ship were in their 20s, but we also didn't want to lose too much complexity in the data, so we took five values so that the average would fall in the same range of values.

Furthermore, to increase ease in data processing, we removed the Name, Ticket, and PassengerId variables. We thought this was appropriate because these attributes would be more difficult to translate into quantifiable variables, such as Name, which could tell us during what decade the passenger was born or if they were wealthy. PassengerId was merely the ID of the passenger in the data and has no valuable additional information. Although the ticket can imply class level, we found that a lot of

information was missing in this column, and other attributes could provide more accurate information on passenger wealth, such as Fare and Class.

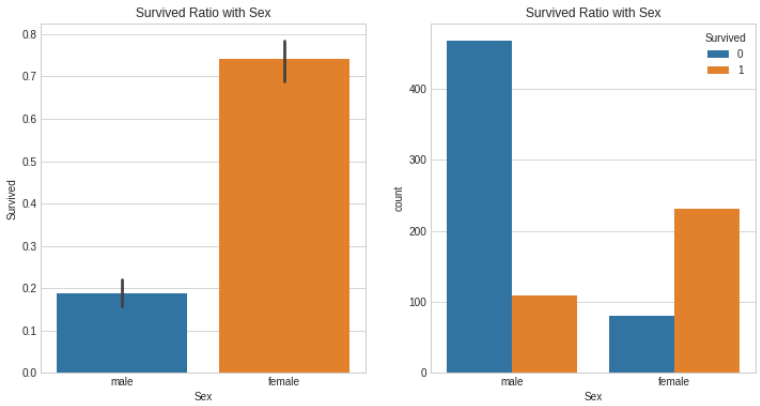
After getting reasonable hypothetical values for the missing Age values, we no longer have any missing values in our data set. Our finalized dataset consists of the following variables which will be used to predict survival: Pclass (type of ticket booked on the ship: 1st/2nd/3rd class), Sex (male/female), Age (passenger's age, if estimated it is xx.5), SibSP (number siblings or spouses of the passenger also on the ship, but does not include mistresses), Fare (amount the passenger paid for their ticket, in pounds), Parch (numbers of parents and children of the passenger also on the ship), and Embarked (port at which the passenger boarded the ship: Cherbourg, Queenstown, or Southampton). We will use several visualization techniques in order to gain a better understanding of the data we are working with. First, we display the linear correlations of each of the variables to find which variables shared a high linear correlation.

	Survived	Pclass	Age	SibSp	Parch	Fare
Survived	1.000	-0.336	-0.075	-0.034	0.083	0.255
Pclass	-0.336	1.000	-0.328	0.082	0.017	-0.548
Age	-0.075	-0.328	1.000	-0.217	-0.173	0.096
SibSp	-0.034	0.082	-0.217	1.000	0.415	0.161
Parch	0.083	0.017	-0.173	0.415	1.000	0.218
Fare	0.255	-0.548	0.096	0.161	0.218	1.000

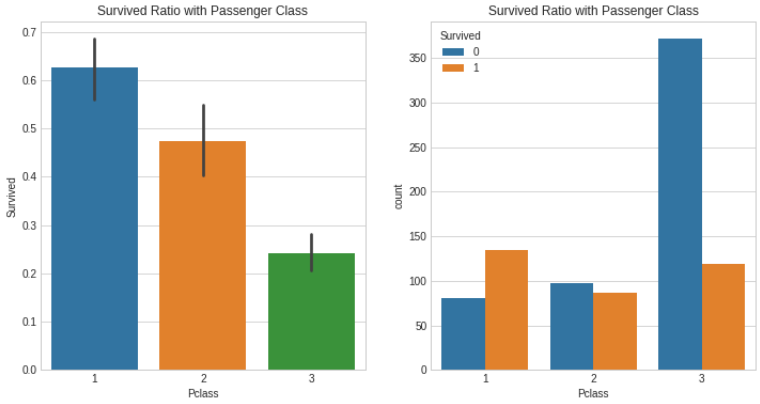
When looking at the correlations, we can see that the variables Fare and Pclass have the highest correlation coefficient of -0.548. It is also an inverse relationship, as Fare increases, the Passenger class decreases. Furthermore, we see that the variables Parch and SibSp also have a relatively high correlation coefficient of 0.415. However, this time it is a positive value, meaning that as the number of parents and children increases, the number of siblings/spouses increases. For each of these relationships, we checked to see if there were outliers by finding the z-score of each of the data points. We defined the outliers as being more that 3 standard deviations away from the mean and we did find some outliers. However, we decided to keep the outliers in our dataset because when testing both the dataset with the outliers removed and the dataset while keeping the outliers, we found that the outliers did not play a significant role in the outcome of our testing.

Next, we create the four histograms below in order to analyze our data in comparison with previous research that highlights the key features for surviving the Titanic incident as passenger Class and Sex.

In the first histogram we see that the female group had a significantly higher survival rate over the male group. Examining the survival rates within each sex group as depicted in the second histogram, the number of males who did not survive is significantly higher than the number of males who did survive, while the number of females who survived is much higher than the number of females who did not survive. These findings suggest that gender was a key feature in influencing survival, which is understandable due to the ship's order that women and children leave the boat before men during the Titanic's evacuation.



Through the next set of histograms, we examine survival rate vs passenger class. As per the first histogram, we observe the highest survival rate for first class, followed by second class, and finally third class with the lowest survival rate. Connecting this finding back to the Titanic situation, higher classes of passengers likely had access to closer/faster ways out during evacuation. Next, as per an in-group comparison depicted in the second histogram, first class is the only group in which the number of passengers who survived exceeds the number of passengers that did not survive.



From this comparison, it is reasonable to consider gender and passenger class as decisive features in indicating survival of the Titanic incident.

4 Proposed Methodology

4.1 Data Preprocessing

To pre-process the dataset, we first transform categorical data into numeric values for ease in future modeling. In the Titanic dataset, there are two categorical labels: Embarked and Sex. Considering the lack of natural ordering of these categories, we decided to use one-hot encoding to transform these two labels. The method we utilized is the pandas library's get_dummies(). This replaced "Embarked" and "Sex" with five new attributes: Embarked_S, Embarked_C, Embarked_Q, Sex_female and Sex_male.

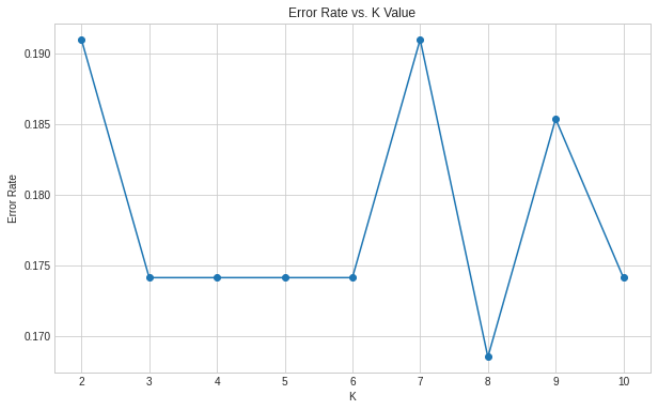
Now we have a dataset consisting of only numerical values named "titanic_encoded". However the range of these values varies greatly. For example, the Pclass label represents the ticket class of each passenger, which only have three values: 1, 2, 3. By contrast, the value of Age label is distributed from close to 0 until 74 years. To generate a useful and accurate model, we must apply a normalization or standardization procedure. To normalize the data, we imported MinMaxScaler from sklearn.preprocessing. After the scaling of the dataset, we have a clean and normalized titanic_clean.

The final preprocessing step we took is to partition the dataset into attributes and target. Since the goal of our Machine Learning models is to predict whether a passenger could survive, the target in this case is *Survived*, which we named y. The rest of labels are attributes which we named X.

4.2 Modeling

After the data preprocessing, the raw dataset now has been transformed into normalized input attributes, X and target, y, which can be easily utilized in the modeling process. As part of our comparative study, we built a total of 5 models to predict survival of Titanic passengers, including: Logistic Regression, K-nearest Neighbor, Neural Network, Random Forest, and Support Vector Machine (with linear and rbf kernels). We chose these models based on the nature of the Titanic Survival problem as a binary classification problem, and also based on observations from our visualization where key features included gender and class/fare. Although an additional testing dataset was provided by Kaggle, the main way to verify the testing predictions was to submit them in a csv file for the competition, which we are not partaking in. So, for each of the models, we decided to split the training data itself into train and test data following a ratio of 80:20.

Model Descriptions The first model is a basic Logistic Regression classification model using the LogisticRegression() function from the sklearn library. This model utilizes the Newton method solver. The next model we implemented is the K-Nearest Neighbors Classification model using sklearn's KNeighborsClassifier() function. As evidenced in the plot below, we found that when the number of clusters, k = 8, the error rate is lowest.



Our third model is a 4-layered Neural Network model using sklearn's MLPClassifier() with hidden layer sizes (9, 6), maximum iterations set to 300, a learning rate of 0.08, the stochastic gradient solver, and the logistic activation function. The fourth model is a Random Forest decision tree classifier. We used sklearn's RandomForestClassifier() function and set the number of trees in the forest to 100. Our fifth and final model type is Support Vector Machine (SVM) with both the linear and rbf kernels utilized. We justify these parameters with the tuning described below.

Hyperparameter Tuning The models and hyperparameters described above were updated using hyperparameter tuning methods in order to optimize model accuracy. To do this we applied the Grid Search algorithm to all of our models (with exception to the Random Forest decision tree since it had a high accuracy rate).

The first model we applied Grid Search to is the Logistic Regression model. The optimal parameters we found are: Inverse of regularization strength(C) = 0.01, penalty = l2, solver = 'newton-cg', with a best score of 0.800153. We chose to replace C with 1 and penalty = none, as it gave a higher accuracy.

The second Grid Search was on the KNN model and the optimal parameters were: leaf size = 2, number of neighbors = 10, Power parameter for the Minkowski metric = 2. With these parameters the best score of this model is 0.810968. We also decided to choose smaller parameters, since 10 neighbors was a little high. Instead we chose number of neighbors to be 8, since it had one of the smaller error rates as we saw in the error plot on the previous page.

For our Neural Network model, we used Grid Search again to tune the hyperparameters of Hidden layer Sizes, maximum iterations and learning rate. From our Grid Search algorithm we obtained the following values: hidden layer sizes = (9, 6), learning rate = 0.08, max iterations = 300 with a optimal score of 0.8132863581540024. We updated the our initial model with the parameters we observed from the Grid Search output.

Finally, we also used Grid Search to find the proper parameters for the Support Vector Machine models (for both the linear and rbf kernels). For the linear kernel, the best score we got was 0.787399 using the parameters: regularization parameter(C) = 10 and kernel coefficient(gamma) = 1. For the rbf kernel, the best score we got was 0.815540 using the parameters: regularization parameter(C) = 10 and kernel coefficient(gamma) = 1. We adapted these optimized parameters into our original SVM models.

5 Experimental Results

After training and testing our five models, we collected several metrics to measure their performance and compare them against one another. The metrics included are Accuracy, Classification Report, and Confusion Matrices. From these, we mainly utilized the Accuracy, F1 Score, and Confusion Matrices to compare and contrast the five different models.

Below we present a table of the Confusion Matrices for all the models, then detailed discussions of each model's performance, and finally some analysis on all five models relative to each other and the overall best model. Random forest had the highest true results, while both SVMs had the lowest. K-Nearest Neighbors had the highest true positives. Logistic regression and artificial neural network had more average true and false results compared to the other models.

Confusion Matrices for all 5 Models

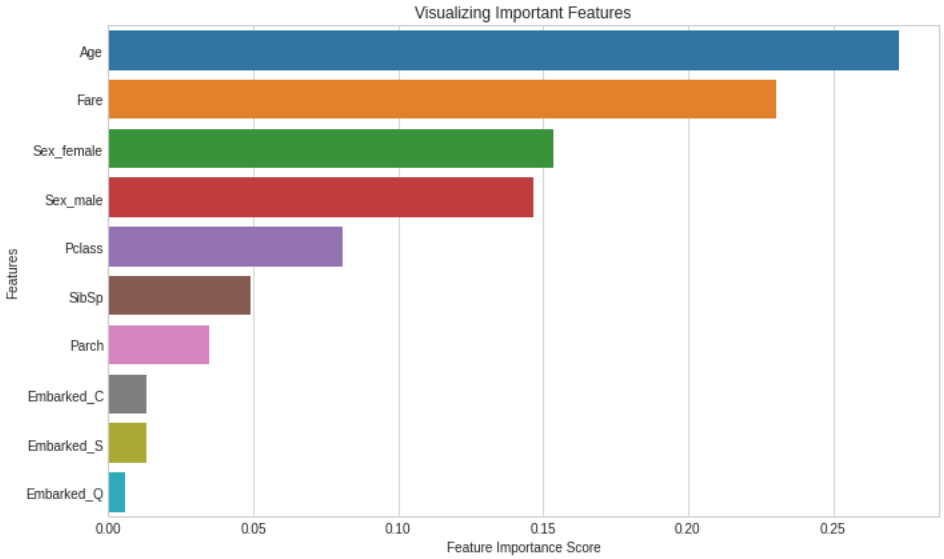
Logistic Regression	KNN	ANN	Random Forest	SVM (lin)	SVM (rbf)
[[109 17] 17 35]]	[[119 7] 23 29]]	[[114 12] 22 30]]	[[115 11] 15 37]]	[[106 20] 20 32]]	[[106 20] 20 32]]

Logistic Regression The Logistic Regression model had an overall accuracy of 0.80 and weighted average F1 score of 0.81, and proved to be the second-most accurate model out of the five. We chose this model as one of the five explored in this study since it is generally very well-applicable to binary classification problems, and it indeed performed well based on our results.

K-Nearest Neighbors The K-Nearest Neighbors Classification model had an overall accuracy of 0.83 and weighted average F1 score of 0.82, and is the fourth-most accurate model out of the five (tied with ANN). When we created the K-Nearest Neighbors Classifier, we found the optimal number of clusters by using elbow method and plotting error rates of the KNN model between 2 and 10 clusters. We found that the model with 8 clusters had the lowest error rates, especially compared to the other cluster sizes which had far greater error rates.

Artificial Neural Network The Artificial Neural Network model had an overall accuracy of 0.81 and weighted average F1 score of 0.82, and is the fourth-most accurate model out of the five (tied with KNN). This model was a feed-forward neural network trained with optimal hyperparameters found during grid search. We also ran cross-validation on this model, collecting accuracy and MSE information during each iteration. The average accuracy was 0.7895 and average MSE was 0.2105.

Random Forest The Random Forest model had an overall accuracy of 0.85 and weighted average F1 score of 0.85, and proved to be the most accurate model out of the five. Below we include a graph with data generated by the Random Forest model to rank the importance of each feature in the dataset during classification.

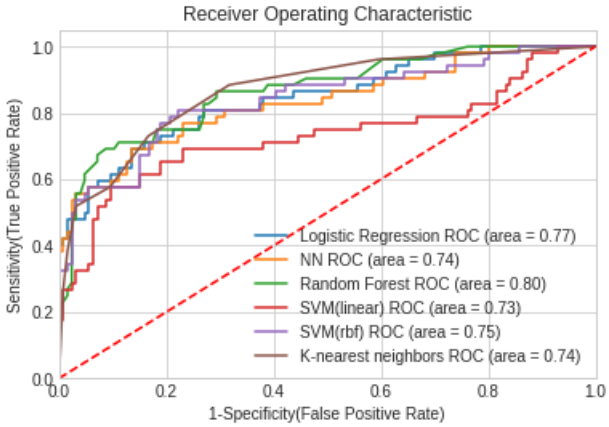


From the figure above, it is evident that the feature Age has the most impact in determining survival, with a feature importance score of over 0.25. This makes sense because most people who were on the Titanic were between the ages of 20-29 years, and people who are within this age range biologically are much more likely to survive a traumatic shipwrecking and prolonged exposure to subzero temperatures. The feature Fare closely follows Age with a score of nearly 0.23 and thus is also rather important. One reason that might be why Fare is more important in the model than Sex being female or male, or class of ticket is that Fare has the additional component of class and affordability of ticket factored in. A person who pays a high fare for their ticket may be of the first class, or perhaps they might have paid a high fare but were discriminated against when being placed in a class of ticket. For example, a mistress of a rich person may have a high costing ticket but is placed in second class, which was farther away from the deck of the ship. Therefore the complexities in Fare may take up more of an importance than features of ticket Pclass or Sex. Another observation that can be made is that Sex_female is only slightly more important than Sex_male, and this fits in with the research knowledge that men were less likely to survive than women, but the small gap could be because most of the variation in the data was already captured by the Fare and Age variables. Lastly, the variables of SibSp, Parch, Embarked_C, Embarked_S, and Embarked_Q all had feature importance of less than 0.05 for our model, which indicates that they did not have much to add. The three different types of Embarked variables refer to the ports at which the passengers got on the boat. When we dropped these last five variables from the model, however, accuracy dropped by 2%, so we decided to keep them in the model.

Support Vector Machine Finally, the SVM model with linear kernel had an overall accuracy of 0.775 and weighted average F1 score of 0.79 and proved to be the least accurate model out of the five. The SVM model with rbf kernel had an accuracy of 0.82 and weighted average F1 score of 0.83, and was the third-most accurate model.

Receiver Operating Characteristic (ROC) Curve For a side-by-side comparison of the abilities of all the binary classification models, we included a plot with all five models' Receiver Operating Characteristic Curves shown together. The ROC Curves depict the true positive rate (TP/(TP+FN)) plotted against the false positive rate (FP/TN+FP)). In the context of the Titanic Survival problem,

the true positive rate represents correctly predicted survival, while false positive rate represents incorrectly predicted survival. The false positive rate is the most crucial in this problem and should ideally be as low as possible (area under the curve should be as large as possible), because issuing a predicted result of survival when a passenger actually does not survive can have great repercussions for the Titanic problems and similar future applications.



Final Ranking of Models Based on the ROC curve above, the ranking of models from greatest area under the curve (AUC) (highest predictive accuracy) to least area under the curve (lowest predictive accuracy) is as follows: 1) Random Forest, 2) Logistic Regression, 3) SVM with rbf kernel, 4) K-Nearest Neighbors and Neural Network (tie), 5) SVM with linear kernel. The Accuracy and F1 scores for each model also echo and corroborate these results.This is different from our confusion matrix analysis earlier which was based solely on adding up true results and comparing them.

Overall, the model with the best performance for this problem seems to be Random Forest. This finding corresponds well with the past studies we examined before starting our own work, as many of them also reported the Random Forest algorithm to be the most effective for the Titanic Survival problem and dataset.

6 Conclusion & Discussion

Overall, during this project, we built, compared, and tested the efficacy of five different Machine Learning models on the Titanic Survival dataset to predict and gain insights regarding important features for survival. We found that the model using the Random Forest strategy had the highest accuracy for predicting the survival of passengers. Also, as seen in the Visualizing Features Graph generated during the Random Forest modeling, we found that Age, *Fare*, *Sex* were the most significant features in determining survival. This is consistent with the findings of the previous literature which we explored in our literature review.

This research has broad future implications as the models and insights found can be extended to gauge and improve survival rate for modern-day modes of transportation. With pre-generated survival information on large-scale passenger vehicles like planes and cruises for instance, it would be possible to make safer vehicles and more effective evacuation plans. Furthermore, the features which were used in this project only consider individual influences. Future data sets and machine learning models can be extended to also consider outside features such as the environmental conditions and resource availability, which can have an impact on determining survival. With our study and similar research on the Titanic Survival problem, scientists and engineers can make plans that could save countless lives through proper planning and learn from the tragic maiden voyage of the RMS Titanic.

7 References

Balakumar, B., Raviraj, P. & Sivaranjani, K. (2017). Prediction of survivors in Titanic dataset: A comparative study using Machine Learning algorithms (retrieved on July 2020 from https://pdfs.semanticscholar.org/545a/9e5da57058cf08e32eae6b5816839505ac3c.pdf).

Brownlee, Jason. "Tune Hyperparameters for Classification Machine Learning Algorithms." Machine Learning Mastery, Machine Learning Mastery Pty. Ltd., 27 Aug. 2020, machinelearningmastery.com/hyperparameters-for-classification-machine-learning-algorithms/.

"Classification Report." Classification Report - Yellowbrick v1.3.post1 Documentation, The Scikit-Yb Developers, 2016, www.scikit-yb.org/en/latest/api/classifier/classification_report.html.

Ekinci, Ekin & Omurca, Sevinc & Acun, Neytullah. 2018, "A Comparative Study on Machine Learning Techniques Using Titanic Dataset".

Fandom, "List of crew members on board RMS Titanic", Nov, 15th, 2006 https://titanicdatabase.fandom.com/wiki/Crew_of_the_RMS_Titanic

Kaggle. "Titanic - Machine Learning from Disaster." Kaggle: GettingStarted Prediction Competition, Kaggle, www.kaggle.com/c/titanic/data.

Kakde, Yogesh and S. Agrawal. "Predicting Survival on Titanic by Applying Exploratory Data Analytics and Machine Learning Techniques." International Journal of Computer Applications 179 (2018): 32-38.

Morales, Carlos Raul. "Random Forest on Titanic Dataset" Medium, Analytics Vidhya, 17 Aug. 2020, medium.com/analytics-vidhya/random-forest-on-titanic-dataset-88327a014b4d.

Nadine Farag and Ghada Hassan. 2018. Predicting the Survivors of the Titanic Kaggle, Machine Learning From Disaster. In Proceedings of the 7th International Conference on Software and Information Engineering (ICSIE '18). Association for Computing Machinery, New York, NY, USA, 32–37. DOI:https://doi.org/10.1145/3220267.3220282

Star, Charlie. "RMS Titanic: facts from beneath," 4 July, 2017. https://www.kiwireport.com/rms-titanic-facts-beneath/. Accessed 25 July 2021.

Singh, R. Nagpal and R. Sehgal, "Exploratory Data Analysis and Machine Learning on Titanic Disaster Dataset," 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2020, pp. 320-326, doi: 10.1109/Confluence47617.2020.9057955.

Titanica, "Titanic Crew List" https://www.encyclopedia-titanica.org/titanic-crew-list/, March 25th, 2003

"Titanic in Nova Scotia." Province of Nova Scotia. https://novascotia.ca/titanic/wireless-transcript.asp, Accessed 25 July 2021.

Zanchi, Lisa. ShiftComm. https://www.shiftcomm.com/insights/never-let-go-titanic-survival-101/. Accessed 25 July 2021.

Martulandi, Adipta. "K-Nearest Neighbors in Python + Hyperparameters Tuning." Medium, DataDrivenInvestor, 24 Oct. 2019, medium.datadriveninvestor.com/k-nearest-neighbors-in-python-hyperparameters-tuning-716734bc557f.

"How to Plot Multiple ROC Curves in One Plot with Legend and AUC Scores in Python?" Stack Overflow, 20 Mar. 2017, stackoverflow.com/questions/42894871/how-to-plot-multiple-roc-curves-in-one-plot-with-legend-and-auc-scores-in-python.